



SateLoc: A Virtual Fingerprinting Approach to Outdoor LoRa Localization Using Satellite Images

YUXIANG LIN, WEI DONG, and YI GAO, Zhejiang University and Alibaba-Zhejiang University Joint Institute of Frontier Technologies
 TAO GU, Macquarie University

With the increasing relevance of the Internet of Things and large-scale location-based services, LoRa localization has been attractive due to its low-cost, low-power, and long-range properties. However, existing localization approaches based on received signal strength indicators are either easily affected by signal fading of different land-cover types or labor intensive. In this work, we propose SateLoc, a LoRa localization system that utilizes satellite images to generate virtual fingerprints. Specifically, SateLoc first uses high-resolution satellite images to identify land-cover types. With the path loss parameters of each land-cover type, SateLoc can automatically generate a virtual fingerprinting map for each gateway. We then propose a novel multi-gateway combination strategy, which is weighted by the environmental interference of each gateway, to produce a joint likelihood distribution for localization and tracking. We implement SateLoc with commercial LoRa devices without any hardware modification, and evaluate its performance in a 227,500-m² urban area. Experimental results show that SateLoc achieves a median localization error of 43.5 m, improving more than 50% compared to state-of-the-art model-based approaches. Moreover, SateLoc can achieve a median tracking error of 37.9 m with the distance constraint of adjacent estimated locations. More importantly, compared to fingerprinting-based approaches, SateLoc does not require the labor-intensive fingerprint acquisition process.

CCS Concepts: • Networks → Wide area networks; Location based services;

Additional Key Words and Phrases: LoRa localization and tracking, land-cover information

ACM Reference format:

Yuxiang Lin, Wei Dong, Yi Gao, and Tao Gu. 2021. SateLoc: A Virtual Fingerprinting Approach to Outdoor LoRa Localization Using Satellite Images. *ACM Trans. Sen. Netw.* 17, 4, Article 43 (June 2021), 28 pages.
<https://doi.org/10.1145/3461012>

A preliminary version of this article appears in *Proceedings of the 19th International Conference on Information Processing in Sensor Networks (IPSN 2020)*. This work is supported by the National Science Foundation of China (No. 62072396 and No. 61772465), Zhejiang Provincial Natural Science Foundation for Distinguished Young Scholars under No. LR19F020001, and Australian Research Council (ARC) Discovery Project Grants, DP190101888 and DP180103932.

Authors' addresses: Y. Lin, W. Dong (corresponding author), and Y. Gao, College of Computer Science, Zhejiang University, 38 Zheda Road, Hangzhou, Zhejiang, China and Alibaba-Zhejiang University Joint Institute of Frontier Technologies, Hangzhou, Zhejiang, China; emails: {linyx, dongw}@zju.edu.cn, qhgaoyi@gmail.com; T. Gu, Department of Computing, Macquarie University, 4 Research Park Drive, North Ryde, NSW 2109, Australia; email: tao.gu@mq.edu.au.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2021 Association for Computing Machinery.

1550-4859/2021/06-ART43 \$15.00

<https://doi.org/10.1145/3461012>

1 INTRODUCTION

The vision of the **Internet of Things (IoT)** has been to communicate with everyday objects around us. It has been reported that the number of IoT devices will reach 75.44 billion by 2025 [44]. The location information of these IoT devices can be exploited in various city-scale applications, such as finding lost objects (e.g., keys), navigating, and tracking pets. Over the years, various solutions have been proposed for such open-air localization. The traditional **Global Positioning System (GPS)** can provide accurate locations of the IoT devices. However, a GPS receiver typically consumes 30 to 50 mA of power and needs additional power to forward data [20], which is incompatible with the low power consumption constraints of IoT systems [38]. In addition, GPS tends to lose connectivity in indoor environments. To remedy these defects, over the years, many localization approaches have been proposed based on wireless technologies such as Wi-Fi [47, 48], BLE [5, 50], and Zigbee [7, 45]. These approaches have been shown to be effective; however, they require dense deployment of access points due to short radio range, leading to high deployment and maintenance costs. Long-range radio-based localization has recently attracted attention from researchers.

Among existing long-range radios, LoRa has been widely deployed in many city-scale IoT networks [11]. Exploiting LoRa for localizing IoT devices over long distances has attracted increasing research interests [2, 15, 20, 22, 25, 38]. An important property of LoRa to enable remote sensing is its multipath and Doppler resistance [8, 36]. In addition, its low cost, low power, and good scalability make LoRa suitable for IoT localization.

Although the research community has made significant advances in LoRa localization, many limitations exist. Existing LoRa localization algorithms rely on either calculating the **Time Difference of Arrival (TDoA)** [20, 38] or measuring the **Received Signal Strength Indicator (RSSI)** [2, 25]. TDoA-based approaches use the time differences obtained from different gateways to estimate node location. These approaches can only achieve sub-kilometer accuracy due to the limitation of the inherent time resolution of LoRa chips [18]. RSSI-based approaches can be further divided into two categories: (1) directly obtain the location by comparing the received RSSI values with an RSSI fingerprinting database (i.e., fingerprinting-based approaches [2, 14, 42]), and (2) first estimate the distance from the signal source with a known signal propagation model, then calculate the location using trilateration algorithms (i.e., model-based approaches [8, 16, 25, 26]). Fingerprinting-based approaches can usually achieve good accuracy. However, they are labor intensive since they require a site survey to build a database and update the database regularly to reflect changes in dynamic environments. As for the model-based approaches, one big issue is that they have not considered complex signal fading of different land-cover types (e.g., buildings, trees, and water) along LoRa links.

Signal fading on a LoRa link may vary depending on different land-cover types. From our benchmark experiments in Section 3.2, we observe that different land-cover types indeed cause significantly different path loss to LoRa signal. The actual path loss of a LoRa link may deviate from its theoretical value at a scale of tens of decibels. This motivates us to investigate how to accurately quantify path loss with respect to various land-cover types in the real world. If we can obtain the actual path loss in any location, we can then use it as a virtual fingerprint and further build a virtual fingerprint database for localization.

In this article, we present SateLoc, a novel LoRa Localization system based on virtual fingerprints obtained from *Satellite* images. Specifically, SateLoc first utilizes the high-resolution satellite image to identify the land-cover type of each pixel in the image. SateLoc then automatically estimates the path loss of an arbitrary LoRa link with the path loss parameters of the traversed land-cover types. We face two technical challenges in quantifying the path loss. The first challenge is to extract the

LoRa link of an arbitrary slope from the image and further divide it into segments, each of which belongs to the same land-cover type. The second challenge is how to combine the path losses of these segments with their lengths and the corresponding path loss parameters. A simple sum-up of the path losses of all segments may not work because these segments are extracted at different distances from the gateway and will have different impacts on the overall path loss.

To address the preceding challenges, we first propose an iterative path loss model. Compared to the standard log-distance model, our model can characterize the overall path loss more accurately since it considers both land-cover types and distances. We then propose an algorithm that can automatically segment an arbitrary LoRa link and calculate its overall path loss (see details in Algorithm 1). For each gateway, SateLoc will generate a virtual fingerprinting map, which consists of the path loss between each pixel in the satellite image and the gateway. Obviously, this virtual fingerprinting map can be generated with little human effort. More importantly, SateLoc uses the **Expected Signal Power (ESP)** instead of RSSI for fingerprinting since ESP characterizes signal energy even when the signal power is below the noise floor after a long propagation or traveling through walls. Subsequently, we can combine the virtual fingerprinting maps of multiple gateways to get the final location. However, in reality, the measurements of different gateways may suffer from different environmental interference, making it inappropriate to utilize these measurements equally. To adaptively combining the fingerprinting maps of gateways, we propose a weighted combination strategy, which takes the environmental interference of LoRa links as the weight. Finally, SateLoc will produce a joint location likelihood distribution to localize a LoRa node. For tracking applications, SateLoc will use the distance constraint of adjacent locations to reduce the search space of subsequent locations, thus further improving the tracking accuracy and reducing the system latency.

We implement SateLoc with commercial LoRa devices without any hardware modification. We evaluate its performance in a $350 \times 650\text{-m}^2$ urban area. Experimental results show that SateLoc achieves a median localization error of 43.5 m, which is improved by 56.6% and 53.0%, respectively, compared to two state-of-the-art model-based approaches (i.e., standard log-normal model [8, 25, 26] and Okumura-Hata model [16]). The localization accuracy of SateLoc is also comparable to existing fingerprinting-based approaches [2], and importantly we do not require any labor-intensive fingerprint acquisition process. For tracking applications, SateLoc further reduces the median tracking error to 37.9 m while significantly reducing the tracking delay.

In summary, we make the following key contributions:

- We propose SateLoc, which utilizes satellite images to achieve accurate LoRa localization with little human effort. SateLoc can automatically extract the land-cover information to generate virtual fingerprinting maps (i.e., ESP maps) for localization and tracking.
- We propose an iterative path loss model and empirically analyze the path loss parameters of different land-cover types. With this model, we can adaptively get a more accurate path loss of an arbitrary LoRa link for fingerprinting.
- We propose a multi-gateway combination strategy, which quantifies the environmental interference of each gateway as its weight, to jointly estimate node location.
- We propose a location-constrained approach to further improve the localization performance for tracking applications.
- We implement SateLoc with commercial LoRa devices and conduct comprehensive experiments in the field. Results show that SateLoc achieves a median localization error of 43.5 m and a median tracking error of 37.9 m in a $227,500\text{-m}^2$ area.

The rest of the article is organized as follows. Section 2 reviews the related work of LoRa localization. Section 3 introduces the background and motivation of SateLoc. Section 4 describes an

overview. In Section 5, we show how to automatically extract land-cover information. We present the ESP map extraction process and location estimation process in Section 6 and Section 7, respectively. We show how to achieve large-scale tracking with SateLoc in Section 8. We give the implementation details and evaluation results of SateLoc in Section 9. SateLoc is discussed in Section 10, and finally, Section 11 concludes this article.

2 RELATED WORK

The related work of LoRa localization can be mainly divided into the following two categories: TDoA-based localization and RSSI-based localization.

2.1 TDoA-Based Localization

TDoA-based approaches localize a LoRa node by the time differences when the same signal arrives at multiple gateways. In the work of Fargas and Petersen [20], a LoRaWAN network with TDoA capabilities was deployed to perform localization for stationary nodes. Carvalho et al. [10] evaluate the feasibility of using LoRaWAN for mobile applications. Podevijn et al. [38] further implement a TDoA system to evaluate the tracking performance in mobile scenarios.

However, TDoA-based approaches usually have poor localization performance because, in the current commodity LoRa hardware implementation, the resolution a timestamp can reach is one microsecond (μs) only [18, 22]. Such a resolution can easily result in a localization error of several hundred meters. We have shown that TDoA-based approaches cannot perform well using commercial LoRa devices with low clock precision in Section 9.2.1. In addition, these approaches require strict clock synchronization among gateways, making it more difficult for real deployment in the field.

μ Locate, in the work of Nandakumar et al. [34], designs a dedicated multi-band LoRa backscatter system to achieve meter-level localization accuracy in a multi-room environment. In contrast, SateLoc aims to achieve accurate remote localization with off-the-shelf LoRa devices.

2.2 RSSI-Based Localization

RSSI-based approaches are usually very efficient for remote LoRa localization with low cost since RSSI values are widely available in commodity LoRa devices.

Prior model-based approaches use RSSI measurements to estimate the distance from node to gateway using propagation models, which characterize the relationship between path loss and distance. Bor et al. [8] and Petajajarvi et al. [37] employ the standard log-normal model to describe path loss. Lam et al. [25, 26] further propose algorithms based on the log-normal model to eliminate noise interference for outdoor localization. Recently, Demetri et al. [16] first identified the type of surrounding environments (*urban* or *suburban*) and then used the corresponding equation of the Okumura-Hata model to characterize path loss. The main drawback of these approaches is that most signal propagation models work better in theory but work poorly in reality due to the complex signal fading. Specifically, they directly use unified models to characterize the path loss of a whole LoRa link and do not consider different land-cover types along the link, introducing deviation to the estimation of the overall path loss. Improving localization accuracy with existing propagation models remains a challenging task.

Fingerprinting-based approaches work more effectively in reality. In these approaches, RSSI values of known locations are manually collected into a database in an offline training phase, then RSSI measurements will be compared with the database for online localization. Machine learning techniques, such as the Bayesian inference method [14, 28], SVM [42], and **k-Nearest-Neighbor** (kNN) [2], are typically applied on collected RSSI measurements to estimate node location. Although fingerprinting-based approaches can be more accurate than model-based techniques, the

construction of an effective fingerprint database usually requires a significant amount of manual efforts, especially for large-scale scenarios. Moreover, regular updates are essentially required to keep fingerprints up to date.

In SateLoc, we aim to combine the strength of the aforementioned two approaches, as well as address their drawbacks. Specifically, we utilize the land-cover information to enhance the existing propagation models to be more realistic and adaptive to real environments. With this enhanced iterative path loss model, we can then generate a more effective virtual map for fingerprint matching with little human effort.

3 BACKGROUND AND MOTIVATION

In this section, we first give the background of LoRa and then describe the motivation of using land-cover information to improve localization performance.

3.1 LoRa Background

LoRa is a physical layer technique that operates in sub-gigahertz ISM bands. LoRa adopts chirp spread spectrum modulation, in which a chirp is encoded using a linear variation of frequency over time. Different chirps (“0” and “1”) differ from each other in the initial frequency. A typical LoRa system consists of LoRa gateways and LoRa nodes. Given a central frequency f_c , the frequency of a transmitted up chirp increases from $f_c - BW/2$ to $f_c + BW/2$. At the receiver side, an inverse down chirp whose frequency decreases from $f_c + BW/2$ to $f_c - BW/2$ can be utilized to decode the chirp. A LoRa receiver can decode LoRa transmissions up to 20 dB below the noise floor, enabling very long communication distances. In addition, as LoRa uses its entire allocated bandwidth to transmit data, it is more robust to channel noise, multipath, and Doppler effects. Therefore, LoRa can play an important role in remote localization.

3.2 Motivation

In practice, the deployment of LoRa may not be in *open space* only but most likely in a combination of open space, forest, rangeland, and so on. In this case, Friis’ law [16] cannot be applied to model its propagation fading. When traversing different land-cover types, LoRa signal propagation can be complicated and may exhibit different behaviors.

We intend to study the influence of different land-cover types on LoRa links. We first use the log-distance path loss model [41] to describe the large-scale path loss $PL(d)$ in decibels:

$$PL(d) = \overline{PL(d_0)} + 10n \log\left(\frac{d}{d_0}\right) + N_\sigma, \quad (1)$$

where d is the distance between LoRa node and gateway, $\overline{PL(d_0)}$ is the mean path loss at a known reference distance d_0 , n is the path loss exponent of the corresponding environment, and N_σ is a zero-mean Gaussian random variable with a standard deviation of σ . The key advantage of this model is that it can model the path loss well in different environments with appropriate path loss parameters n and σ .

To present the impact of different land-cover types quantitatively, we conduct the following field experiments to identify the path loss parameters of each land-cover type.

Setup. In this experiment, we select five representative land-cover types as illustrated in Table 1. Each type may have different obstacles that cause different effects on LoRa signal propagation. Note that these types can be easily identified using multi-spectral satellite images [33].

We use a commercial Dragino LG01 gateway and a Dragino LoRa shield [19] in our experiments, as shown in Figure 1. Both the LoRa gateway and node are equipped with omnidirectional antennas. In the experiments, we can roughly consider that the antenna height is the same as the device

Table 1. Five Representative Land-Cover Types

Type	Description
BUILT-UP	Built-up areas with human artifacts
FOREST	Trees
FIELD	Open space, farms
WATER	Rivers, oceans, lakes
RANGELAND	Green land, grassland

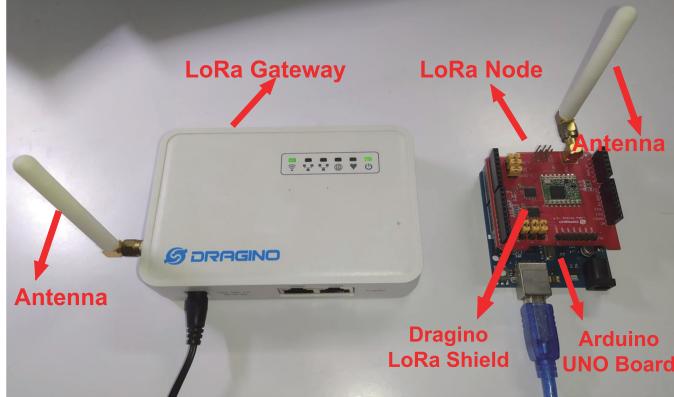


Fig. 1. The commercial Dragino LG01 gateway and LoRa node. The LoRa node consists of a Dragino LoRa shield and an Arduino UNO microcontroller board [3].

height. These devices are embedded with standard Semtech SX1278 LoRa RF-fronts [43]. As shown in Figure 2, we deploy the LoRa devices in five areas of different land-cover types near the Yuquan Campus, Zhejiang University. Note that these locations are not the same as the actual deployment in our evaluation part. In this experiment, we mainly aim to obtain reliable path loss parameters for land-cover types that may appear in the target evaluation area. In all land-cover types, both the gateway and the node are placed at 1 m above the ground. In the case of the water type, the gateway and the node are placed at 1 m above the water surface. As for the built-up type, we deploy the gateway in a concrete teaching building and move the node along a wall outside the building. On the campus, all buildings have a very similar internal structure and are all made of concrete. As a result, the estimated path loss parameters of built-up type are consistent in our experiments and can be used for localization on our campus. The LoRa node transmits packets periodically with the parameters configured as follows: **Spreading Factor (SF)** = 12,¹ **Bandwidth (BW)** = 125 kHz, **Coding Rate (CR)** = 4/8, and **Transmission Power (TP)** = 13 dBm. We use a common reference distance $d_0 = 1\text{m}$ to obtain a stable and accurate path loss reference $\overline{PL}(d_0)$ [21, 41]. We conducted this experiment on a sunny day in summer with low moisture in the air and plenty of leaves on the trees.

Dataset. In each land-cover type, we deploy the LoRa node at seven different distances to the gateway: 2, 13, 34, 60, 92, 200, and 300 m for packet collection. In particular, we set the maximum distance of the built-up type to 92 m because a few walls in a built-up area can make the LoRa link disappear within 100 m, which is consistent with some existing work [8, 29]. At each location, we

¹SF is the ratio between the symbol rate and chip rate and takes a value between 7 to 12. A higher value of SF results in a longer time for each symbol transmission and yields a longer communication range.

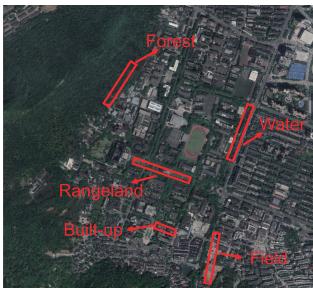


Fig. 2. Five experimental areas of different land-cover types for the LoRa path loss parameter study.

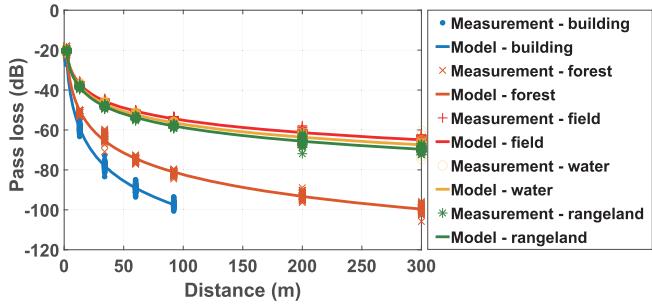


Fig. 3. Path loss vs. distance in the five land-cover types. Dots are the measurements and solid lines show the theoretical values calculated from Equation (1).

collect 50 packets and then record the RSSI and SNR of each packet for later analysis. In total, 1,650 packets (50 packets \times 7 locations \times 4 types + 50 packets \times 5 locations) are collected for parameter study.

Parameter study. Although LoRa can receive transmission with a signal power that is up to 20 dB below the noise floor, the common RSSI cannot capture the path loss of these extremely weak signals. LoRa uses the ESP to obtain the energy of the signal [16]:

$$ESP = RSSI + SNR - 10 \log(1 + 10^{0.1SNR}). \quad (2)$$

For each packet, we use its SNR and RSSI to calculate the ESP and further the path loss.² With sufficient packets collected at the reference distance d_0 , we can first obtain the reference path loss $PL(d_0)$ of each land-cover type. Then for each type, we use the curve fitting function in MatLab to obtain its essential path loss parameters n and σ . Figure 3 shows both measured and theoretical path loss at different distances in the five land-cover types. Each fitting curve is represented with a unique color of the corresponding measurements. We use the corresponding n and set $\sigma = 0$ to generate these curves. As seen, the log-normal model can well capture the characteristics of LoRa links in different land-cover types. Taking a closer look at the figure, different types can lead to significantly different path losses. Both *built-up* and *forest* types will cause a relatively more severe path loss than the other types. As a result, land-cover information of the LoRa link needs to be considered when estimating its path loss. We can also observe that even if the propagation of the LoRa radio signal is similar in open areas (e.g., field, water, rangeland), their path loss differences can be up to 5 to 10 dB when the propagation distance is 300 m. As these land-cover types are commonly seen in an urban area, the path loss differences among these similar types cannot be ignored when estimating the path loss of a LoRa link.

Table 2 shows the reference path loss $PL(d_0)$, path loss parameters n , and σ of all land-cover types we have measured. The table also shows the average error between the modeled value and the measurements. As seen, path loss exponent n is larger in a more complex land-cover type, which contains more **Non-Line-of-Sight (NLoS)** paths and thus more severe fading. Standard deviation σ and the fitting error both depend on the path loss variation due to dynamic obstacles (e.g., pedestrians, vehicles) and static obstacles (e.g., buildings, trees) across the link [41]. Therefore, these two values are relatively higher in the *built-up* and *forest* types with more obstacles. As these land-cover types on the campus are basically consistent, we directly use these path loss parameters

²To achieve the conversion between *ESP* and *path loss*, we have pre-measured the *ESP* value ESP_0 when the transceivers are placed together. Thus, we have $PathLoss = ESP_0 - ESP$. In the following, these two terms can be used interchangeably.

Table 2. Reference Path Loss $\overline{PL}(d_0)$, Path Loss Exponent n , Standard Deviation σ , and the Average Model Fitting Error in Different Land-Cover Types

Type	$PL(d_0)$	n	σ	Error (dB)
Built-up	23.091	4.499	1.887	2.351
Forest	21.254	3.616	0.891	1.691
Field	20.478	2.054	0.577	0.907
Water	20.709	2.158	0.623	0.977
Rangeland	20.392	2.275	0.657	1.057

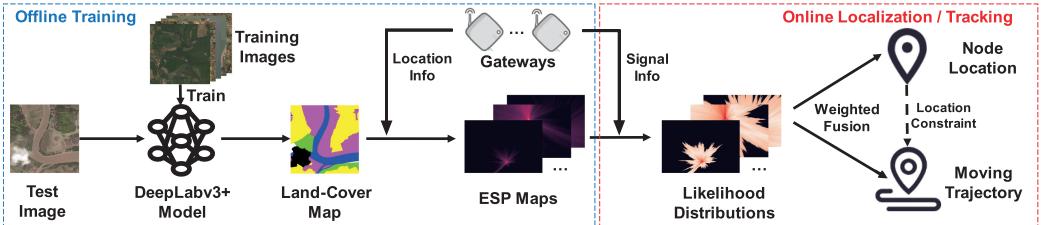


Fig. 4. Overview of SateLoc and the workflows for LoRa localization and tracking.

in our experiments. It is worth noting that the measured pass loss parameters may not hold across different environments even with the same land-cover type. For example, buildings made of wood and concrete will have different path loss parameters. In addition, trees with different numbers of leaves may also lead to different pass loss parameters. As a result, when using SateLoc in areas with significantly different land-cover settings, a locally specific tuning is needed. Specifically, we should conduct a similar local parameter estimation experiment to tune the path loss parameters of land-cover types that may appear in the area of interest.

In summary, this experiment shows that different land-cover types can result in significantly different path loss effects. This strongly motivates us to utilize the land-cover information to improve LoRa localization accuracy. Our idea is first to identify the land-cover types from a satellite image like Demetri et al. [16], then automatically generate an ESP map for localization.

4 OVERVIEW OF SATELOC

Figure 4 gives an overview of SateLoc. SateLoc integrates land-cover information from satellite images and location and signal information from gateways to localize a LoRa node. The workflow of SateLoc is described as follows:

- (1) With the training image set, SateLoc first trains a DeepLabv3+ [13] model to classify different land-cover types in satellite images.
- (2) When the DeepLabv3+ model is trained, it can be applied to generate a land-cover map according to the satellite image of the testing area, which covers both the gateways and the node to be localized.
- (3) With the land-cover map, SateLoc further produces an ESP map for each gateway based on its location in the testing area. The ESP map essentially serves as a virtual fingerprinting map.
- (4) Every time a LoRa node to be localized transmits a packet, SateLoc records all surrounding gateways that receive the packet. Using the signal information (i.e., RSSI and SNR) extracted from the LoRa packets, SateLoc can produce a location likelihood distribution for each gateway based on its ESP map.

(5) SateLoc uses a weighted combination strategy to combine the likelihood distributions of all gateways. Finally, a joint location likelihood distribution can be generated to determine node location. To get the moving trajectory for tracking applications, SateLoc will further use a location constraint, such as the distance between adjacent target locations, to improve the tracking performance.

The first three steps are offline processes that need to be performed only once. The last two online localization/tracking steps will be performed each time the LoRa signal is transmitted to the gateways.

5 LAND-COVER CLASSIFICATION

In this section, we show how to identify land-cover types with the high-resolution satellite image datasets provided via DeepGlobe [17]. SateLoc uses DeepLabv3+ [13] for the multi-class image segmentation task. The DeepLabv3+ model employs an encoder-decoder structure, which has been successfully applied to many semantic segmentation tasks [6, 30]. The encoder module encodes multi-scale contextual information of the images, whereas the decoder module refines the image segmentation results along object boundaries. We select DeepLabv3+ as our land-cover classification approach due to its capabilities of extracting rich land-cover features and recovering the boundaries of different land-cover types.

5.1 Building the Model

The inputs to train the DeepLabv3+ model are the 803 satellite images at a spatial resolution of 50 cm and the corresponding mask images in the DeepGlobe dataset. We extract the three raw RGB values of each pixel from satellite images and the associated labeled land-cover types from mask images as the training data. The trained DeepLabv3+ model can be applied to the satellite image of the area of interest to identify the land-cover type of each pixel and further help output a land-cover map.

We use TensorFlow [1] to implement the DeepLabv3+ model on a single computer with one NVIDIA GPU. We employ a standard Residual Network architecture with 101 layers (ResNet-101) [23] as the network backbone in the model. We set some important hyperparameters of the model according to Chen et al. [13]. Specifically, we employ the “poly” learning rate policy. The learning rate is a tuning parameter in an optimization algorithm that determines the step size at each iteration while moving toward a minimum of a loss function. In setting a learning rate, there is a trade-off between the rate of convergence and overshooting. A too high learning rate will make the learning jump over minima, but a too low learning rate will either take too long to converge or get stuck in an undesirable local minimum. We set the initial learning rate to 0.007 and the cut-off learning rate to 10^{-6} [27]. The crop size is set to 513×513 . The output stride is the ratio of the input image spatial resolution to the final output resolution. For the image segmentation task, the output stride is set to 16 by applying the corresponding atrous convolution [12]. The batch size is the total number of training samples present in a single batch, and an iteration is the number of batches to complete one epoch. Since a large batch size is required to train batch normalization parameters and get better segmentation performance [12], we set the batch size to 10 and the maximum iteration to 30,000. In each epoch, the training data is sampled uniformly and randomly shuffled.

We use 723 satellite images and their mask images as the training set and the remaining 80 images as the test set. During training, we sample the same number of pixels (i.e., around 10 million pixels) for each land-cover type to eliminate the imbalance of sample sizes of different land-cover types in the training set. We perform a fivefold cross-validation strategy to get the final DeepLabv3+ model with the highest classification accuracy.

Table 3. Classification Accuracy of the DeepLabv3+ Model

Type	Built-up	Forest	Field	Water	Rangeland
Precision	0.92	0.81	0.74	0.59	0.38
Recall	0.82	0.57	0.64	0.90	0.60
OA	0.69				

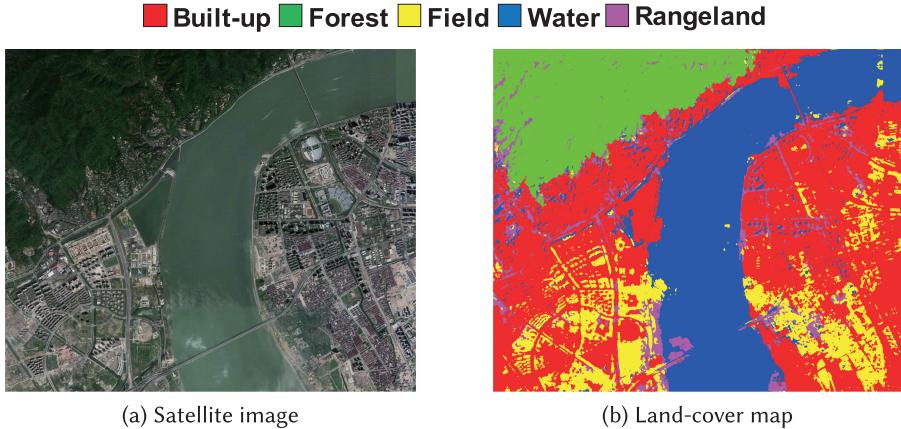


Fig. 5. An example of land-cover type classification: the ground truth satellite image in RGB (a) and the corresponding land-cover map generated by SateLoc (b) in a $5 \times 5\text{-km}^2$ area of Hangzhou, China.

5.2 Predicting Land-Cover Type

To evaluate classification accuracy, we assess the experimental results with **Overall Accuracy (OA)**, precision, and recall. OA is the percentage of correctly classified pixels of all pixels, whereas precision and recall are at a granularity of each type. The values of OA, precision, and recall are in the range of 0 to 1, and the higher value indicates better classification performance.

Table 3 shows the classification results of the trained DeepLabv3+ model. The OA is 0.69. This is because the labels provided in the dataset are far from perfect [17]. In the dataset, many masks ignore terrain details and small structures not annotated in the ground truth. Incomplete and often inaccurate labeling presents a significant barrier for model development and evaluation. However, compared to the baseline CNN-based approach proposed in the work of Demir et al. [17] (with an OA of 0.433) and SateLoc (IPSN 20) [31] that uses a random forest model for land-cover classification (with an OA of 0.51), our DeepLabv3+ model still improves the accuracy by 59.4% and 35.3%, respectively. Moreover, we have not carefully preprocessed the DeepGlobe dataset and have not tuned the hyperparameters very hard, as they are not the key contributions of our work. We believe the land-cover identification performance of our work can be further improved with a prior selection of accurately labeled samples and finer-tuned hyperparameters.

Figure 5(a) shows an example satellite image of a $5 \times 5\text{-km}^2$ area in Hangzhou, China. The image is directly sampled from Google Earth, and its corresponding land-cover map is shown in Figure 5(b). We see that the classification and segmentation quality is good enough, except for some uncertainty between water (blue) and forest (green) in the urban area. However, these misclassified pixels are sparsely distributed, and each covers an area of $0.6 \times 0.6\text{ m}^2$. These pixels are a small number of the long LoRa link, and the induced path loss error is still acceptable. We have also found that some large areas may be misclassified among water, built-up, and field land-cover types. These classification errors will certainly affect the subsequent localization performance of SateLoc. When

deploying SateLoc in these areas, we need to carefully improve the DeepLabv3+ model by using a finer-grained encoder feature resolution [13] to get better classification accuracy. In addition, the land-cover map has a property that each land-cover area is composed of a number of clustered pixels with clear boundaries. This property helps speed up the generation of the corresponding ESP map, which will be described in the next section.

6 GENERATING ESP MAPS

In this section, we will first introduce our iterative path loss model. Then we describe how to utilize the land-cover map to generate the virtual fingerprinting map (i.e., ESP map) for each gateway.

6.1 Iterative Path Loss Model

Considering different land-cover types along LoRa links, we propose an iterative path loss model based on the log-distance model in Equation (1). We first introduce the following notations:

- d_1, d_2, \dots, d_K are the distances between a gateway and K consecutive positions along a LoRa link, where $d_1 < d_2 < \dots < d_K$. Each segment between adjacent positions (i.e., (d_{k-1}, d_k)) belongs to the same land-cover type.
- $n_{(d_{k-1}, d_k)}$ denotes the path loss exponent of the land-cover type between adjacent positions d_{k-1} and d_k .
- $\sigma_{(d_{k-1}, d_k)}$ denotes the standard deviation of the Gaussian random variable of the land-cover type between adjacent positions d_{k-1} and d_k .

Our iterative path loss model can be formulated as follows:

$$\begin{aligned} PL(d_K) &= \overline{PL(d_0)} + 10n_{(d_0, d_1)} \log \frac{d_1}{d_0} + 10n_{(d_1, d_2)} \log \frac{d_2}{d_1} \\ &\quad + \dots + 10n_{(d_{K-1}, d_K)} \log \frac{d_K}{d_{K-1}} \\ &\quad + N_{\sigma_{(d_0, d_1)}} + N_{\sigma_{(d_1, d_2)}} + \dots + N_{\sigma_{(d_{K-1}, d_K)}} \\ &= \overline{PL(d_0)} + \sum_{k=1}^K \left(10n_{(d_{k-1}, d_k)} \log \frac{d_k}{d_{k-1}} + N_{\sigma_{(d_{k-1}, d_k)}} \right). \end{aligned} \tag{3}$$

The path loss $PL(d_K)$ at a distance d_K from the gateway can be calculated by iteratively adding the path loss of all segments of different land-cover types on the LoRa link. Compared to the standard log-distance model, our iterative path loss model considers land-cover types of segments at different distances on a LoRa link and thus can estimate the overall path loss more accurately. The detailed performance of our iterative path loss model is evaluated in Section 9.

6.2 ESP Map Generation

Figure 6 shows an example land-cover map, where each square represents a pixel in the map. We aim to generate the ESP map for the gateway (shown as the yellow circle). The basic idea of ESP map generation is to first select the pixels on the wireless link between the LoRa gateway and each candidate location, then calculate the path loss with the land-cover types of these pixels. Taking Pixel 1 as an example, we first connect it with the gateway using a straight line. Then we extract the coordinates and label of each pixel on the line. Since these pixels all belong to the *built-up* type, SateLoc directly uses the path loss model in Equation (1), along with the distance $d_{(\text{Gateway}, \text{Pixel}_1)}$ and the corresponding path loss exponent $n_{\text{built-up}}$, to calculate the ESP value at Pixel 1. It is worth noting that we set $\sigma_{\text{built-up}} = 0$ to eliminate the randomness of the generated fingerprinting maps.

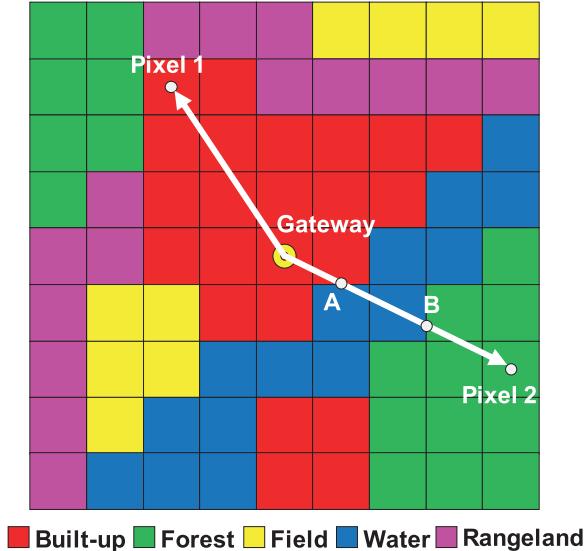


Fig. 6. Illustration of ESP map generation with an example land-cover map. LoRa node at Pixel 1 transmits its packets in the same land-cover type (i.e., built-up) while node at Pixel 2 needs to transmit across multiple types.

In a more complicated case shown in Figure 6, the link between Pixel 2 and the LoRa gateway goes through several land-cover types. We can use Equation (3) to calculate the path loss between Pixel 2 and the LoRa gateway. Specifically, there will be two intersections A and B at the boundaries between different types. Starting from the gateway, we first calculate the path loss $PL_{(Gateway, A)}$ on segment $(Gateway, A)$ with Equation (1). For the next step, we replace the reference distance d_0 with the distance $d_{(Gateway, A)}$ between gateway and intersection A in the log-distance path loss model. The reference path loss $PL(d_0)$ will also be replaced by $PL_{(Gateway, A)}$ correspondingly. Now we can calculate the path loss $PL_{(A, B)}$ on segment (A, B) with the updated log-distance model and the path loss exponent n_{water} . We repeat the preceding steps until we obtain the path loss of the last segment (i.e., segment $(B, Pixel_2)$) of the link. After that, we sum up the path loss of all segments and get the ESP value at Pixel 2.

Algorithm 1 shows the details of our ESP map generation algorithm. For each pixel in the land-cover map, SateLoc first extracts all traversed pixels from it to the gateway (not included) using a form of Bresenham's line drawing algorithm [9]. Then SateLoc divides these pixels into segments, each of which belongs to the same land-cover type. The path loss of each segment can be iteratively calculated with the corresponding path loss exponent and its distance to the gateway. Finally, SateLoc calculates the overall path loss and further the ESP value of the pixel. Although pixels belonging to the same type are usually clustered in an area (illustrated in Section 5), there will be reasonable ESP calculations (each corresponds to a segment) for each pixel. Therefore, the overall map generation cost (including time cost and computing resource overhead) will be acceptable.

To make the map more accurate, we utilize the receiver sensitivity S of the gateway. Specifically, pixels with an ESP value less than S will not receive the LoRa signal from the gateway. Therefore, for pixels whose ESP value less than S , we just assign $-\infty$ to them in the ESP map. This will also help significantly reduce the computation overhead in the following likelihood distribution generation process. Taken from Semtech [43], we set $S = -136$ dBm according to the LoRa configuration in our experiments. Figure 7(a) and (b) show two example ESP maps over the 2D space when gateways

ALGORITHM 1: ESP map generation algorithm

Input: Land-cover map LM with $X \times Y$ pixels; Gateway location (x_G, y_G) ; Basic ESP value ESP_0 ; Path loss exponents n 's of five land-cover types; Receiver sensitivity S ; Pixel resolution R ; The reference path loss $PL(d_0)$ (set with Table 2 according to the land-cover type of the pixel where the gateway is placed) at the reference distance d_0

Output: ESP map $EM = \{EM(1, 1), \dots, EM(X, Y)\}$

```

1: for  $i = 1$  to  $X$  do
2:   for  $j = 1$  to  $Y$  do
3:     // Get the pixel array  $P$  that consists of the coordinates  $[x, y]$  and land-cover types  $c$  of all pixels on
       link  $((x_G, y_G), (i, j))$ . The getTraversedPixels function is designed based on the Bresenham's algorithm.
4:      $P = getTraversedPixels(x_G, y_G, i, j, LM)$ 
5:     // Initialize the link length  $L$  and the path loss  $PL$  using the first pixel  $P_1$  in  $P$ .  $P_{1x}, P_{1y}, P_{1c}$  are
       the attributes  $x, y, c$  of  $P_1$ , respectively.
6:      $x_{end} = P_{1x}, y_{end} = P_{1y}, n = n_{P_{1c}}$ 
7:      $D = R\sqrt{(x_{end} - x_G)^2 + (y_{end} - y_G)^2}$ 
8:      $PL = PL(d_0) + 10n \log \frac{D}{d_0}$ 
9:      $L = D, x_{start} = P_{1x}, y_{start} = P_{1y}$ 
10:    // Search for the segments whose pixels belong to the same type, and iteratively calculate their
        path losses with their distances to the gateway.  $P_{kx}, P_{ky}, P_{kc}$  indicate the attributes  $x, y, c$  of the  $k$ -th
        pixel in  $P$ , respectively.  $|P|$  is the total pixel number of array  $P$ .
11:    for  $k = 2$  to  $|P| - 1$  do
12:      if  $P_{kc} == P_{k+1c}$  then
13:        continue
14:      else
15:         $x_{end} = P_{kx}, y_{end} = P_{ky}, n = n_{P_{kc}}$ 
16:         $D = R\sqrt{(x_{end} - x_{start})^2 + (y_{end} - y_{start})^2}$ 
17:         $PL = PL + 10n \log \frac{L+D}{L}$ 
18:         $L = L + D, x_{start} = P_{kx}, y_{start} = P_{ky}$ 
19:    // Add the path loss of the last segment to get the overall path loss.
20:     $x_{end} = P_{|P|x}, y_{end} = P_{|P|y}, n = n_{P|P|c}$ 
21:     $D = R\sqrt{(x_{end} - x_{start})^2 + (y_{end} - y_{start})^2}$ 
22:     $PL = PL + 10n \log \frac{L+D}{L}$ 
23:     $EM(i, j) = ESP_0 - PL$ 
24:    if  $EM(i, j) < S$  then
25:       $EM(i, j) = -\infty$ 
26: return  $EM$ 

```

are placed in the campus with sparse and dense buildings nearby, respectively. As seen, the path loss does not spread uniformly from the gateway. The area with dense buildings nearby suffers from more severe signal fading, and the corresponding LoRa coverage degrades more significantly.

7 FROM ESP MAP TO LOCATION

7.1 ESP Map to Likelihood Distribution

We have obtained the ESP map for each gateway, and we now present our algorithm to infer the location of the node. As we record the RSSI and SNR of each transmitted packet from a LoRa node, the corresponding ESP_{node} can be calculated using Equation (2). Assume there are M gateways in the testing area that may receive the LoRa signal, which means there are M ESP maps for

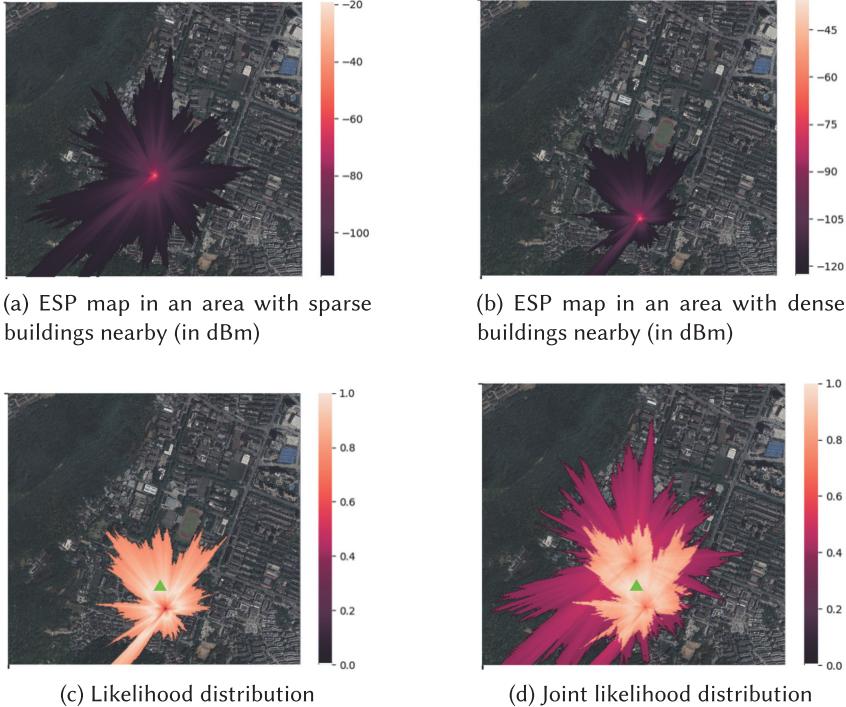


Fig. 7. From ESP to location in a $1.5 \times 1.4\text{-km}^2$ area of Yuquan Campus, Zhejiang University. (a) An example ESP map of $gateway_1$ placed in an area with sparse buildings. (b) An example ESP map of $gateway_2$ placed in an area with dense buildings. (c) An example likelihood distribution of $gateway_2$ when receiving an ESP value of ≈ -106 dBm from the LoRa node. The white pixels show the most possible locations estimated using Equation (4). (d) The joint likelihood distribution of the LoRa node produced using Equation (7) when $gateway_1$ receives an ESP value of ≈ -97 dBm and $gateway_2$ receives an ESP value of ≈ -106 dBm. A brighter pixel indicates a higher likelihood. The green triangle marks the actual location of the node.

localization. The likelihood $L_{m,i}$ of the node locating at the i -th pixel of the m -th ESP map can be computed as

$$L_{m,i} = 1 - \frac{|\overline{ESP_{node}} - ESP_{m,i}|}{\max(|\overline{ESP_{node}} - ESP_m|) - \min(|\overline{ESP_{node}} - ESP_m|)}, \quad (4)$$

where $\overline{ESP_{node}}$ is the mean ESP value measured from the LoRa node. $ESP_{m,i}$ is the ESP value at the i -th pixel in the m -th ESP map. ESP_m refers to the ESP values of all pixels in the m -th ESP map. $\max(|\overline{ESP_{node}} - ESP_m|)$ and $\min(|\overline{ESP_{node}} - ESP_m|)$ are the maximum and minimum ESP differences between $\overline{ESP_{node}}$ and the ESP value of a pixel in the m -th ESP map, respectively. Therefore, the denominator calculates the value range of ESP differences in the m -th ESP map and normalizes the likelihood. For pixels with an ESP value of $-\infty$, we directly set their likelihoods as 0. Figure 7(c) shows an example likelihood distribution of $gateway_2$ over the 2D space when receiving a LoRa packet with an ESP value of ≈ -106 dBm in our experiments. As seen, the shape of possible locations is not a circle since the LoRa links are affected by different land-cover types.

7.2 Likelihood Distribution Fusion

Figure 7(c) also shows that the pixel with the maximum likelihood can be far away from the ground truth when using only one gateway to localize a LoRa node. With only one gateway, the localization error can be up to several hundred meters due to its approximate circular likelihood distribution. To get a more accurate location, SateLoc usually requires at least two gateways. The localization error can be significantly reduced to less than 100 m when using two gateways to localize a LoRa node (shown in Figure 14(e)). The more gateways used, the higher the localization accuracy. In practice, however, there may not be many LoRa gateways covering the same area. To make a trade-off between deployment cost and localization accuracy, we set $M = 3$ in our experiments. The detailed performance with varying numbers of gateways will be evaluated in the evaluation.

A straightforward way to get the joint likelihood distribution is to equally add the likelihood obtained from all gateways. However, the performance of this method deteriorates due to different environmental interference along different LoRa links. The reason is that each link contains various land-cover types, each of which has a specific level of environmental interference. Based on Table 2, we use variance σ^2 to quantitatively represent the level of environmental interference. This is because σ^2 can characterize the uncertainty of the path loss and can be directly summed up when the path loss is independent. A larger σ^2 indicates more severe environmental interference. We hence develop a weighted combination algorithm to build the joint likelihood distribution and mitigate the effects of the environmental interference. The algorithm involves the following three steps.

First, we estimate the environmental interference of each pixel in the M likelihood distributions (one for each gateway). The environmental interference $\Gamma_{m,i}$ of the LoRa link between the m -th gateway $Gateway_m$ and the i -th pixel $Pixel_i$ can be obtained by

$$\Gamma_{m,i} = \sum_{k=1}^{|P_{m,i}|} \sigma_k^2, \quad (5)$$

where $P_{m,i}$ is the pixel array on link ($Gateway_m, Pixel_i$), $|P_{m,i}|$ is the total pixel number in $P_{m,i}$, and σ_k^2 is the interference value at the k -th pixel in $P_{m,i}$. Obviously, a larger Γ indicates that we should assess less weight on the corresponding link.

Second, we calculate the corresponding weight $w_{m,i}$ at the i -th pixel of the m -th likelihood distribution:

$$w_{m,i} = \frac{\frac{\prod_{\theta=1}^M \Gamma_{\theta,i}}{\Gamma_{m,i}}}{\sum_{\theta=1}^M \frac{\prod_{\theta=1}^M \Gamma_{\theta,i}}{\Gamma_{m,i}}}. \quad (6)$$

For a specific pixel $Pixel_i$, a relatively larger environmental interference $\Gamma_{m,i}$ among the M links (one for each gateway) will lead to a lower weight $w_{m,i}$. In addition, Equation (6) normalizes each weight $w_{m,i}$ to a range of 0 to 1. For each pixel, the sum of the M gateways' weights equals to 1.

Finally, SateLoc calculates the weighted sum of M^3 likelihood distributions to form the joint likelihood distribution. In the joint likelihood distribution, the likelihood of each pixel is discrete and can be different from each other. We need to search the joint likelihood distribution and find the desired pixel whose theoretical path loss is the most similar to the measured path loss values. In other words, SateLoc just picks the pixel with the highest likelihood as the final location:

$$Location = \arg \max_i \sum_{m=1}^M w_{m,i} L_{m,i}. \quad (7)$$

³ M can be an arbitrary positive integer (i.e., $M \geq 1$).

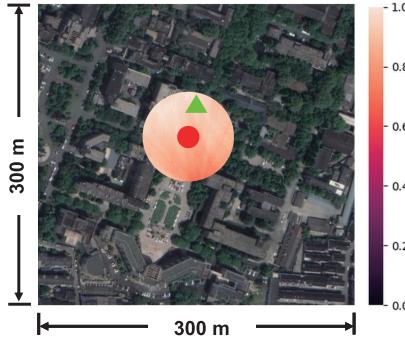


Fig. 8. Improved joint likelihood distribution of Figure 7(d). The maximum distance of adjacent locations (i.e., the radius of circular likelihood distributions) is set to 50 m.

Note that the first two steps are offline performed after getting the traversed pixels in Algorithm 1, whereas the last step needs to be performed in the online localization phase. Figure 7(d) shows an example joint likelihood distribution when $gateway_1$ receives an ESP value of ≈ -97 dBm and $gateway_2$ receives an ESP value of ≈ -106 dBm.

8 LARGE-SCALE TRACKING

Tracking applications usually have higher requirements for localization accuracy and system responsiveness. SateLoc has the potential to help realize large-scale tracking based on the continuity of target movement in the real world. Specifically, for target movement (e.g., human/pet walking, vehicle driving, bicycle riding), the distances between adjacent locations within a certain small time interval (e.g., 5 seconds in our experiments) are likely to be short, due to natural limitations on targets' moving speed. We can use this distance constraint to reduce the search space of subsequent locations and thus improve the tracking performance.

Figure 8 illustratively demonstrates the effect of the location constraint. In SateLoc, we set the maximum distance of adjacent locations to 50 m since we collect tracking data at an interval of 5 seconds through human walking and bicycle riding on campus. Assuming that the last estimated location of the LoRa node is the red dot, to estimate the current location (shown as the green triangle), we first calculate the joint likelihood distribution of a circular area whose center is the red dot with a radius of 50 m. Then we only seek the pixel with the highest likelihood within the circular area, which can help significantly reduce the system overhead and latency. The detailed performance of SateLoc in tracking applications will be shown in Section 9.2.

We also analyze the amount of error propagation that can be safely handled in the case of inaccurate localization of the previous location. Assuming that the human walking speed is 2 m/s in our experiments, the distance between adjacent locations is around 10 m. Figure 9 illustrates the error propagation between two adjacent locations. If the previously estimated location falls in the gray area (e.g., the grey dot), it will affect the current localization result since its search space cannot cover the current node location. Fortunately, due to the continuity of the path loss changes in the ESP map, the subsequently estimated location will also approach the current node location, thereby reducing the impact of the previous localization error. Other locations in the previous search space will have little effect on the current location estimation process. According to the geometric relationship, the white area where the error propagation can be safely handled accounts for 87.3% in our experimental settings. This area can be further expanded with a larger search space and a smaller LoRa data collection interval to reduce the impact of error propagation.

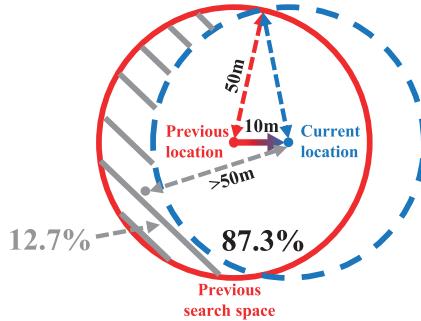


Fig. 9. Error propagation between adjacent locations at a moving distance of 10 m.

9 EVALUATION

In this section, we first present the implementation of SateLoc. We then describe our evaluation and present the experimental results in both localization accuracy and system latency.

9.1 Implementation

We set the LoRa gateway shown in Figure 1 as the receiver and the LoRa node as the transmitter. The RadioHead library [39] provides a LoRa communication library for embedded microprocessors and is installed in both the gateway and node. The transceivers use one channel in the 433-MHz band for transmitting or receiving LoRa signals. We carefully study the typical LoRa configuration parameters: TP, SF, BW, and CR. To ensure the LoRa transmission range and its resilience to noise, we select a set of parameters (TP = 13 dBm, SF = 12, BW = 125 kHz, CR = 4/8) in our experiments. Detailed localization performance of different parameters will also be evaluated. The node continuously broadcasts LoRa packets with unique IDs to all gateways at a period of 5 seconds. Note that we set such a short time period just for experimental analysis. In practice, the period can be much longer to provide a better lifetime. Each gateway is connected to a laptop through Ethernet to monitor and collect LoRa packets. When packets with the same ID are collected at the gateways, we record their RSSIs and SNRs. The core localization algorithms of SateLoc are implemented on a desktop with an Intel i7 6700 CPU and 8 GB of RAM.

Figure 10 shows the site deployment of SateLoc in Yuquan Campus, Zhejiang University. The terrain of Yuquan Campus is generally flat with slight fluctuation. The test satellite image (Figure 10, left, a $1.5 \times 1.4\text{-km}^2$ area) used in our experiments is directly extracted from Google Earth with a spatial resolution of 0.6 m. Three LoRa gateways and one LoRa node are placed in a $350 \times 650\text{-m}^2$ area on a college campus (Figure 10, right). The advertised communication range of LoRa is expected to be more than 10 km for suburban environments [29]. However, LoRa coverage can degrade significantly with the presence of obstacles in NLoS scenarios. Our experimental area covers a large number of buildings and trees, and is full of NLoS paths. As already shown in Figure 7, the LoRa coverage in our experiments can be only a few hundred meters, which is consistent with the experimental results in other works [8, 24, 29, 35]. Therefore, it is reasonable to place three gateways in such complex outdoor scenario, which is also the deployment environments in urban areas, to maintain an acceptable LoRa link quality.

All LoRa gateways are placed at 10 m above the ground in buildings. We manually record the GPS locations of LoRa gateways and convert them into the corresponding 2D pixel coordinates in the satellite image. This process only needs to be performed once because LoRa gateways usually remain stationary after deployment. For each gateway, we pre-measure the path loss at the reference distance (i.e., 1 m) for analysis. The LoRa node is powered by a 5V portable

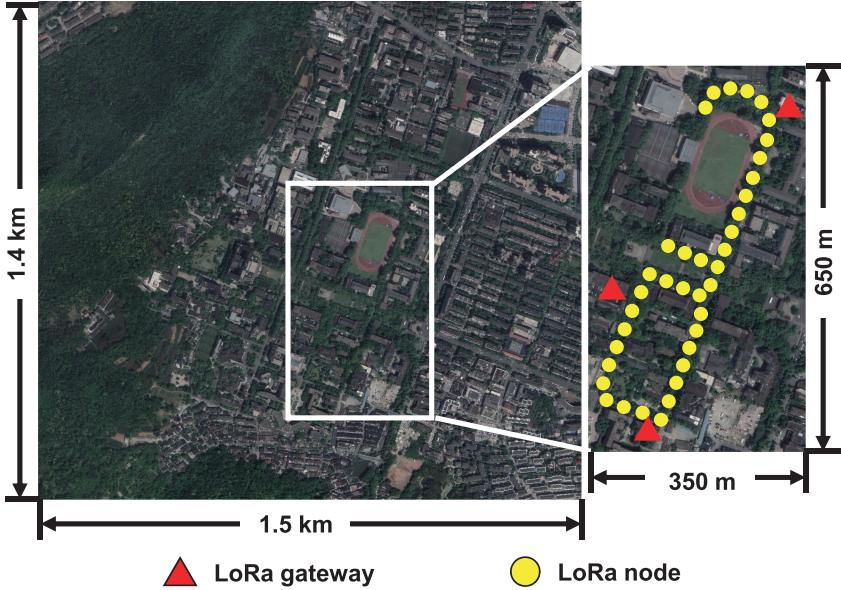


Fig. 10. Deployment of SateLoc in Yuquan Campus, Zhejiang University. The left image is the overall satellite map (2816×3072 pixels) we used in SateLoc. The right image is the zoom-in map of a $350 \times 650\text{-m}^2$ urban area, which shows the deployment of the Dragino gateways (red triangles) and the measurement sites (yellow dots). The three gateways are placed at the 2D pixels of (1170, 2130), (1780, 1550), and (2240, 1660) in the satellite map.

power supply and is placed at a height of 1 m when sending LoRa packets. We place the LoRa node at 38 test locations (shown in Figure 10) on the campus. Similarly, we manually label these locations in the form of 2D pixels in the satellite map as the ground truth. We use the flat distance rather than the three-dimension distance between gateway and node to generate the ESP maps for the following two reasons. First, in our experiments, the height difference between LoRa gateways and the LoRa node is only 9 m. Such a device height difference has little effect on the overall transmission distance that is usually several hundred meters. The impact of device height difference can be further ignored if SateLoc is deployed in commercial LoRa networks covering kilometer ranges. Second, in practice, it is difficult to continuously monitor the relative height of a LoRa node, especially for a mobile one. At each location, we collect the RSSIs and SNRs of 20 LoRa packets, which are aimed to calculate the average ESP to further mitigate the environmental interference. We have also shown the performance with varying numbers of packets in our evaluation.

9.2 Performance Evaluation

As a target for precise and fast remote localization, the two most important metrics are used to testify SateLoc: accuracy and latency.

9.2.1 Accuracy. We first compare the performance of SateLoc with RSSI-based and TDoA-based approaches. Then, we show the effectiveness of the ESP map and our path loss model. Finally, we evaluate the effects of different LoRa configuration parameters.

Comparison study. *Comparison with model-based approaches.* We compare SateLoc with two state-of-the-art approaches that use the standard log-normal model [8, 25, 26] and the

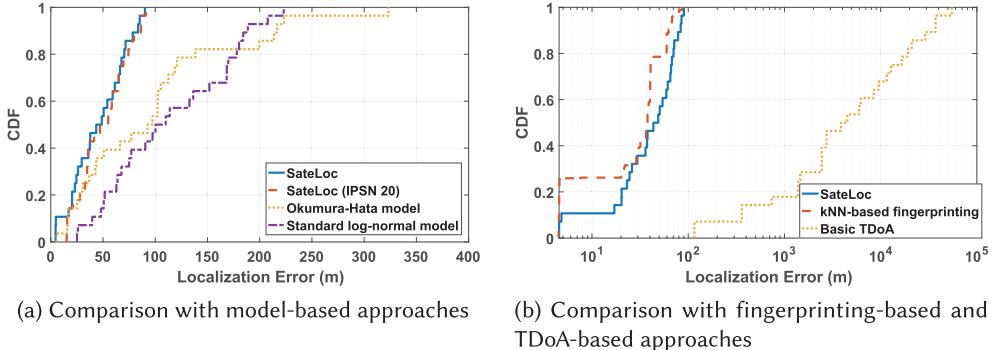


Fig. 11. Overall performance of SateLoc.

Okumura-Hata model [16] to characterize the path loss. We use the parameters estimated in the work of Bor et al. [8] and Demetri [16] for the standard log-normal model and Okumura-Hata model, respectively. The parameters of the standard log-normal model in the work of Bor et al. [8] are estimated in a dense built-up area that is similar to our experimental environments and thus can be directly used in the comparison study. In particular, the Okumura-Hata model is also related to the surrounding environment and consists of an urban equation and a suburban equation [16]. We use its *urban equation* in our experiments since the link path is dominated by the *built-up* and *forest* types on the campus. Although the Okumura-Hata model is expected to hold with large link distances and high base station antenna positions, it can also well characterize the LoRa links below 10 m and in the vicinity of LoRa devices [16]. Based on the estimated path losses, we employ a trilateration method to localize the node in these approaches. For a fair comparison, we use the same number of gateways (i.e., three) and LoRa configuration parameters during benchmarking. We also compare SateLoc with SateLoc (IPSN 20) [31] to show the localization performance improvement with different land-cover classification accuracy.

Figure 11(a) shows that the median localization error of SateLoc is 43.5 m, whereas the median localization errors of the standard log-normal model and the Okumura-Hata model are 100.2 m and 92.6 m, respectively. Compared to the two model-based localization approaches, SateLoc improves the localization accuracy by 56.6% and 53.0%, respectively. Compared to SateLoc (IPSN 20) that achieves a median localization error of 47.1 m, SateLoc further reduces the error by 7.6% with the help of the efficient DeepLabv3+ land-cover classification model. Although there is still a gap compared to high-precision GPS, we believe that SateLoc can help achieve some prime applications such as parked vehicle monitoring [15] with low power consumption. Specifically, the service provider may need to compute how much time each vehicle has been parked. GPS-based vehicle tracking is not widely deployed since GPS is power hungry and requires a connection to the vehicle battery, which increases the complexity of vehicle installation [15]. The localization accuracy of SateLoc allows the provider to monitor whether the vehicle is in the parking lot and record a rough parking area. In this case, the provider can use SateLoc to detect the start and end time of parking. Moreover, GPS cannot work in indoor parking lots due to loss of connection. On the contrary, LoRa has the capability of transmitting through walls indoors [35, 49]. Reviewing the parameter study of built-up areas in Section 3.2, we have shown that LoRa signals can transmit indoors for nearly 100 m when passing through several walls. Therefore, SateLoc can still work indoors due to the good penetration of LoRa signals. Another appealing use case of LoRa localization is the anti-theft of a large number of stationary assets in construction sites, utility yards, airports, and so on, due to the low cost of LoRa nodes.

Comparison with fingerprinting-based approaches. We further compare the performance of SateLoc with a state-of-the-art fingerprinting-based approach [2], which uses kNN for localization. We randomly select 70% (i.e., 14) LoRa packets of each location as the training set and the rest as the test set. During training, we calculate the ESP value of each gateway at each location to construct the fingerprint database. In total, there are 532 fingerprints (14 packets \times 38 locations), each consisting of three ESP values of the corresponding gateways. We adopt Euclidean distance to measure fingerprint similarity. For each test packet, we extract candidate locations with the $k = 3$ smallest distances and use the centroid as its location. We get the final localization error of a test location by averaging the localization errors of packets collected there.

Figure 11(b) shows that SateLoc achieves comparable localization accuracy compared to the kNN-based approach (a median localization error of 37.8 m). In addition, we find that the localization errors of the kNN-based approach are mainly distributed around 40 m, which is roughly the distances between neighbor test locations in our experiments. This indicates that the accuracy of these fingerprinting-based approaches is highly dependent on the site deployment when collecting fingerprints. However, it is labor intensive to construct and maintain an effective fingerprint database, especially in a large-scale area. Instead, SateLoc manages to remedy the defect using more effective virtual ESP fingerprints.

Comparison with TDoA-based approaches. We also make a comparison between SateLoc and a basic TDoA-based localization approach. As shown in Figure 11(b), TDoA-based approaches cannot perform well due to the low clock precision of the low-cost commercial LoRa devices used in our experiments. Specifically, the clock error of our devices can reach up to tens of microseconds, leading to a localization error of thousands of meters. Two lessons learned from these comparisons to make good use of TDoA information are (1) adding modules with high-precision clocks in the gateways or (2) deploying relatively expensive devices with high clock precision.

Effectiveness of the ESP map and path loss model. The ESP map boundary of a gateway can be viewed as its signal coverage area. We move the LoRa node around the gateway and mark the farthest locations where the LoRa signal can be received. Then we connect these locations to get the measured coverage boundary. Figure 12 shows the theoretical boundary of the ESP map and the measured LoRa coverage of a deployed gateway. As seen, the measured boundary can roughly match the automatically generated ESP map. Taking a closer look, the measured boundary is consistently smaller than the theoretical boundary. This is because the minimum measured ESP values (i.e., -123.02 dBm on average) are slightly above the theoretical ESP value (i.e., -136 dBm). We can also observe that there is an overestimation of the gateway coverage in the bottom left part of Figure 12. This is because the forest land-cover type in this area has been identified as water, thus underestimating the path loss.

We also perform controlled experiments to validate our iterative aggregation path loss model and compare it with other state-of-the-art path loss models. We deploy the LoRa devices at 75 different locations on the campus with different numbers of land-cover types in the middle and record the distances of each type. For each location, we collect 30 LoRa packets at an interval of 5 seconds at three different times (i.e., 9 AM, 3 PM, and 9 PM) and calculate the average path loss for model validation. The environmental variables, such as temperature and humidity, are different at these times. In total, there are 2,250 samples used in the experiments. Note that it is difficult to get a complete ground truth ESP map since we need to take these samples site by site. Hence, for each experimental location, we use the average path loss value of 30 samples as the ground truth. Figure 13 shows the detailed path loss estimation errors of SateLoc and other existing models in our experiments. As seen, SateLoc achieves a median path loss estimation error of 7.17 dB, reducing the error by 25.6%, 40.3%, and 77.6% compared to the Okumura-Hata model (i.e., 9.64 dB), standard log-normal model (i.e., 12.00 dB), and free space model (i.e., 32.05 dB), respectively.



Fig. 12. Effectiveness of the ESP map of Yuquan Campus, Zhejiang University. The white lines show the measured signal coverage boundary.

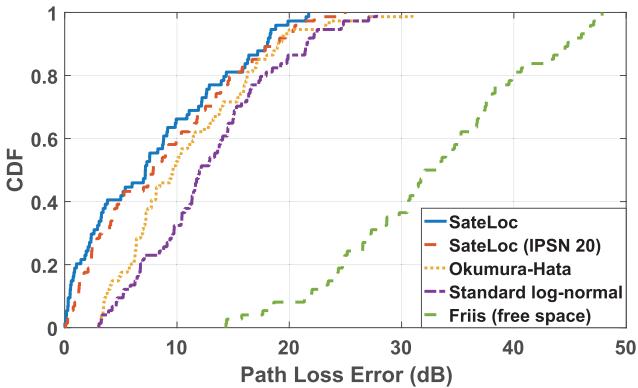


Fig. 13. Path loss estimation errors of our iterative aggregation path loss model and existing models.

Compared to SateLoc (IPSN 20), SateLoc further reduces the path loss estimation error by 8.9% due to the improved land-cover classification accuracy. Results show that SateLoc can get more accurate path losses and further a more accurate ESP map with the novel map generation algorithm.

Impact factors. In the following, we evaluate the performance of SateLoc with different LoRa configurations by varying one of the parameters at a time during the packet collection process. We also evaluate the localization accuracy of SateLoc when using different numbers of gateways and packets, with different gateway heights, and in different weather conditions and times.

Impact of transmission power. Considering that different transmission power will lead to different transmission ranges, localization accuracy may be greatly affected. To evaluate the influence of transmission power, we set TP to 13 dBm, 10 dBm, and 7 dBm, respectively. Figure 14(a) shows the localization error of different TPs. Results show that when the TP drops from 13 dBm to 7 dBm, the median localization error increases by 84.4%. In addition, the localization error in some locations can reach up to 250 m. This is because some gateways cannot successfully receive packets from these locations with a limited transmission power and thus cannot be used for localization.

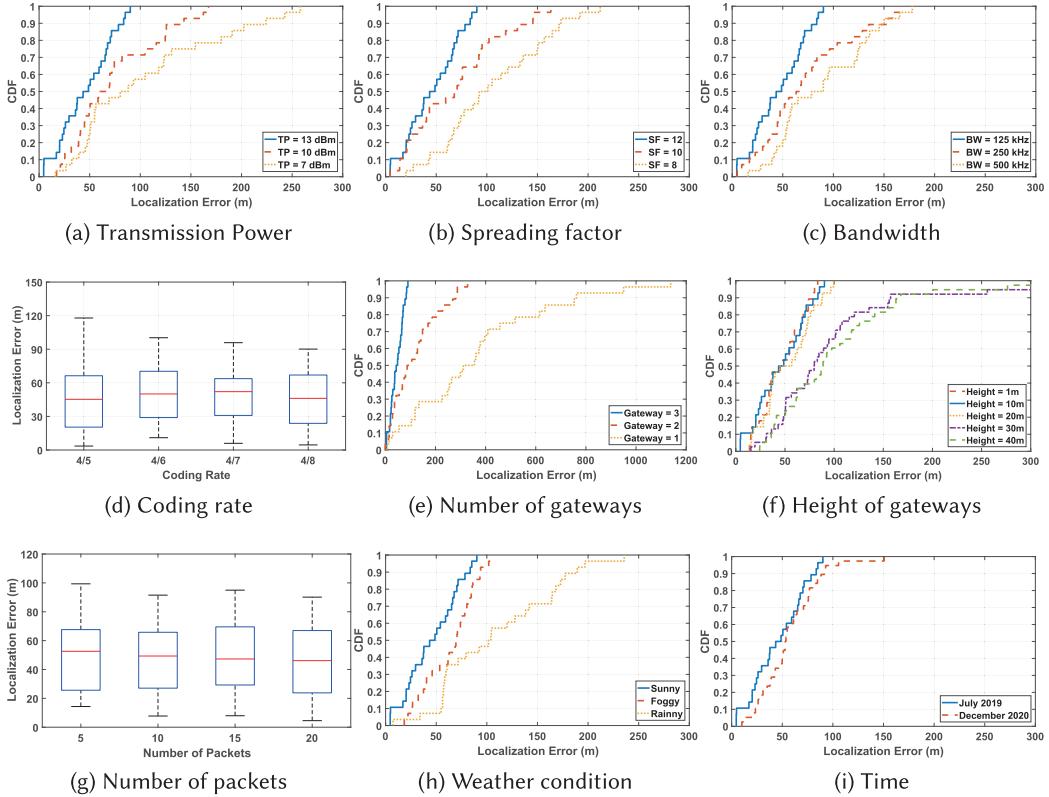


Fig. 14. Localization accuracy of SateLoc with different factors.

Impact of spreading factor. Spreading factor can influence not only transmission range but also the resistance to interference noise [4]. We evaluate the performance of SateLoc with three different SFs: 12, 10, and 8. Figure 14(b) shows that the median localization errors are increased to 75.9 m and 92.6 m when the SFs are set to 10 and 8, respectively. Results show that a higher SF can significantly improve the localization performance of SateLoc, since it can reduce the interference of environmental noise.

Impact of bandwidth. A higher bandwidth gives a higher data rate but decreases the sensitivity due to additional noise [8]. Figure 14(c) shows the localization accuracy when the BWs are set to 125, 250, and 500 kHz. It is as expected that the localization errors increase with higher bandwidths. Compared to the 125-kHz BW, the localization errors of the 250-kHz BW and 500-kHz BW increase by 45.3% and 80.0%, respectively.

Impact of coding rate. Coding rate offers protection against the burstiness of interference and helps reduce the packet error rate. CR can be set to either 4/5, 4/6, 4/7, or 4/8. Figure 14(d) shows that the median localization errors of the four CRs are 43.2 m, 48.6 m, 51.4 m, and 43.5 m, respectively. Results show that SateLoc achieves consistent localization accuracy across different CRs. This is because changing the CR will not significantly influence both the RSSI and SNR of the received LoRa packets.

Impact of number of gateways. We also evaluate the performance of SateLoc when using the collected data from different numbers of gateways. As described in Section 7.2, we first use one, two, and three distributions (i.e., gateways) to form the joint likelihood distribution, respectively. Then

we pick the pixel with the highest likelihood as the final location. The localization results are shown in Figure 14(e). As seen, a larger number of gateways will significantly decrease the localization error. Considering the practical coverage and the deployment overhead of LoRa gateways, we use three gateways in our experiments. We believe the localization accuracy can be further improved with more gateways.

Impact of gateway height. The height at which the transceivers are placed plays a significant role in the interplay between the signal propagation and the environment. Although the LoRa nodes as user devices are usually placed close to the ground, the gateway can be pre-deployed at locations with various heights to favor communication and increase the coverage. In this experiment, we deploy LoRa gateways at five different heights: 1, 10, 20, 30, and 40 m. The LoRa node is placed 1 meter above the ground. We use the path loss parameters of Table 2 in this experiment. Figure 14(f) shows that height will affect path loss and localization performance. As seen, heights lower than 20 m show similar localization accuracy. However, when the deployment height is higher than 30 m, the localization error will increase. This is because a higher LoRa link will traverse less land-cover types. Although the land-cover types below the link will still be identified in the satellite images, they will be incorrectly used to estimate the path loss, thus leading to a relatively less accurate ESP map. We can also see that even when the gateways are placed at a height of 40 m, SateLoc can achieve comparable localization accuracy with existing model-based approaches (see Figure 11(a)).

Impact of number of packets. In our experiments, we collect 20 packets to calculate the average ESP values and localize the node around every 2 minutes (20 packets \times 5 seconds/packet). In practice, we find that the ESP values of packets collected at the same location are stable. In addition, SateLoc can complete the online localization phase within 3 seconds (evaluated in Section 9.2.2). These indicate that we can use fewer packets to localize the node. Figure 14(g) shows that using fewer packets will not significantly decrease the localization accuracy. When the number of packets drops from 20 to 5, the median localization error only increases from 43.5 to 51.4 m. For cases where response time is limited, SateLoc can still perform well using all available LoRa packets.

Impact of weather condition. Weather condition is one of the main influence factors in wireless communication. We collect experimental data on sunny, foggy, and rainy days. In these weather conditions, rainy days have the most moisture in the air, foggy days are the second, and sunny days are the least. Figure 14(h) shows that bad weather conditions (i.e., more moisture in the air) can significantly reduce localization accuracy. This is because the path loss parameters in Table 2 are extracted on sunny days and cannot work well in other conditions with different volumes of moisture in the air. A straightforward solution is to extract the corresponding parameters in all weather conditions.

Impact of time. We use two datasets that were collected far apart in time: one in July 2019 and the other in December 2020. We use the path loss parameters obtained in July 2019 (Table 2) for both datasets. Figure 14(i) shows that the median localization errors will increase by 11.3% when using path loss parameters collected at other times. This is because the environmental conditions, such as the presence of leaf on the trees, are different at different times and will affect the path loss estimation accuracy.

Tracking performance. We evaluate the tracking performance of SateLoc on the campus. We place some markers on the ground and ask a volunteer to ride at a speed of 4 m/s along these masks, which are the ground truth of the trajectory. Figure 15(a) shows an example of the tracking results of SateLoc with the help of location constraint. We also compare the overall tracking performance of SateLoc with and without the location constraint. As shown in Figure 15(b), if we only use the estimated locations to track the target, the median tracking error is 43.8 m. With the added location constraint, SateLoc can achieve a median tracking error of 37.9 m, improving the performance by

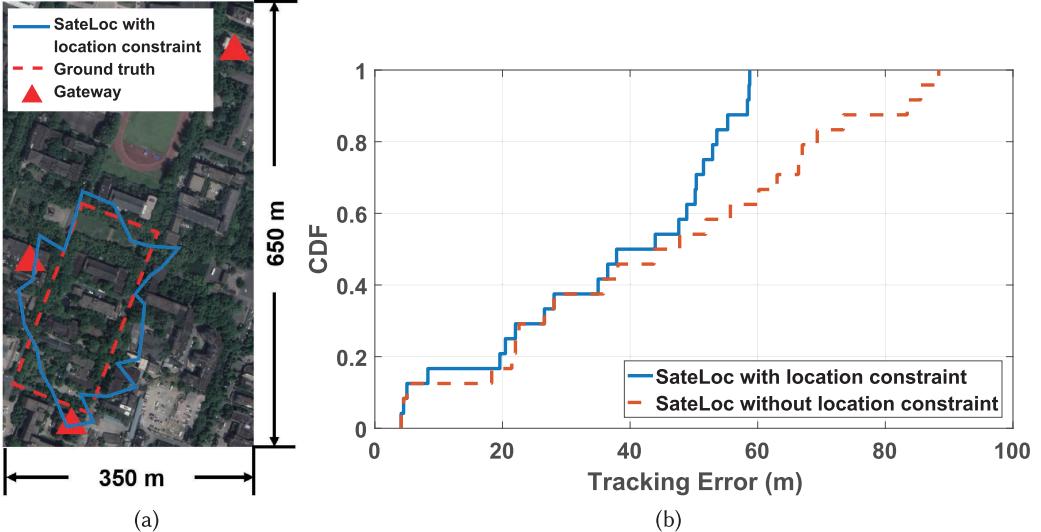


Fig. 15. Tracking accuracy of SateLoc. (a) An example of the tracking results of SateLoc on the campus with location constraint. (b) The overall tracking performance of SateLoc w/o location constraint.

Table 4. Online Localization/Tracking Delay of SateLoc

Application Type	Generation Delay	Fusion Delay	Overall Delay (with three gateways)
Localization (whole map)	5.85 s	0.38 s	17.93 s
Localization (zoom-in map)	0.75 s	0.30	2.55 s
Tracking	0.22 s	0.27 s	0.93 s

13.5%. This is because the localization errors of some estimated locations that deviate a lot from the ground truth are effectively calibrated with the location constraint.

9.2.2 Latency. In the following, we will show the system latency of SateLoc in localization and tracking applications.

Localization delay. Two main phases that contribute to the latency of SateLoc in online localization are the likelihood distribution generation and the fusion phase. Table 4 shows the detailed time delay of each phase for localization applications.

In the generation phase, time delay mainly contributes to the likelihood calculation process. For each gateway, SateLoc takes about 5.85 seconds to generate its likelihood distribution with the whole satellite image (2816×3072 pixels) we used in SateLoc. In the fusion phase, it takes around 0.38 seconds to produce the joint likelihood distribution and get the maximum likelihood.

It is worth noting that time delay is highly relevant to image size. If we only use the pixels of the zoom-in deployment map in Figure 10, the generation time of each gateway can be further reduced to 0.75 seconds and the fusion time can be reduced to 0.30 seconds. Therefore, the overall time delay ($0.75 \text{ s/gateway} \times 3 \text{ gateways} + 0.30 \text{ seconds}$) of the online localization phase should be within 3 seconds. In SateLoc, the time interval between adjacent packets is 5 seconds, which is sufficient for continuous localization. In a general LoRa network, the packet rate of a LoRa node

can even be less than 30 packets/hour to prolong its lifetime [37]. As a result, SateLoc is compatible with existing LoRa networks.

Tracking delay. Table 4 also shows the detailed time delay for tracking applications. In these applications, system latency can be further reduced since SateLoc only uses the joint likelihood distribution of circular areas with a specified radius. In our experiments, the likelihood generation time of each gateway is only 0.22 seconds and the fusion time is 0.27 seconds when we set the radius to 50 m. Therefore, the overall time delay ($0.22 \text{ s/gateway} \times 3 \text{ gateways} + 0.27 \text{ seconds}$) of the online tracking phase is within 1 second, which should be sufficient for most large-scale tracking applications.

10 DISCUSSION

In this section, we first discuss two possible ways to further improve the localization accuracy by improving land-cover classification accuracy and integrating additional information such as TDoA and road information. Then we present discussions on the limitations and future directions of SateLoc.

High-accuracy land-cover classification. Land-cover classification from high-resolution satellite images is known to be a challenging problem. In addition, the labels are usually far from perfect due to the cost of annotating multi-class segmentation masks [17]. Quantifying the path loss of LoRa links accurately much depends on classification quality. There are many other promising deep learning models (e.g., DFCNet [46], U-Net [40], DDCM-Net [32]) that can be used for accurate land-cover classification. We will further investigate a more efficient deep learning approach in our future work. In addition, since the land-cover may change over time, it is necessary to update the test satellite image and reclassify the land-cover types periodically (although not frequently, e.g., updating every 6 months) to maintain good localization accuracy. However, it is also promising to use multi-spectral satellite images with visible, near-infrared, and short-wave infrared bands [16] to improve land-cover classification accuracy.

Integrating additional information. Although SateLoc significantly reduces the localization accuracy to around 50 m, more efforts are required to apply this system for applications with high-accuracy localization requirements such as localizing goods in a warehouse. The localization accuracy can be further improved with multi-dimensional information available in off-the-shelf LoRa devices, such as TDoA. In our future work, we plan to investigate how to fuse the locations extracted from TDoA and RSSI. For tracking applications, the tracking accuracy can be further improved with road information extracted from satellite images.

Limitations of our path loss model. The terrain morphology can affect the propagation environment and path loss. SateLoc cannot be directly used in uneven areas such as hill and mountain areas without the height information of each land-cover type on the LoRa link. For example, when the LoRa transceivers are deployed on both sides of a hill, the parameters of a soil/rock/sand land-cover type should be used to estimate the path loss. For mountain areas, the key effect of path loss will be blockage due to Fresnel Zones [41], not limited to the land-cover types. However, SateLoc will still identify these areas as rangeland/forest and falsely use the corresponding parameters to calculate the path loss. In the future, we plan to explore LoRa localization in uneven areas with a contour map. We believe there is still space to further improve our path loss model with the height information of each land-cover type on the LoRa link. In addition, Figure 14(i) shows that the time variability of LoRa links is another relevant aspect that can increase the deviation of path loss from the model. Therefore, it is also necessary to periodically update the path loss parameters to maintain a good model performance.

More real-world evaluations. In this article, we analyze the performance of SateLoc in a flat urban area of around 0.2 km². Our current evaluation includes land-cover types of buildings, fields, forest,

and rangeland. Due to the limitations of deployment environments and the number of devices, we only use three LoRa gateways in our experiments. In the future, we plan to deploy SateLoc in more environments with more gateways to further evaluate the performance of SateLoc. To deal with the temporal fading effects of LoRa links, another important future direction is to collect a larger dataset in the time domain (e.g., different days or seasons) and investigate an effective machine learning approach to adaptively learn the path loss parameters.

11 CONCLUSION

In this article, we propose SateLoc, which uses high-resolution satellite images for remote LoRa localization. We first analyze the path loss parameters in different land-cover types. With these parameters, we can capture the path loss of an arbitrary LoRa link. SateLoc consists of two phases: an offline training phase and an online localization phase. In the training phase, SateLoc trains a DeepLabv3+ model for automatically generating a land-cover map according to the satellite image of the area of interest. Next, SateLoc produces a virtual fingerprinting map (i.e., ESP map) for each gateway without the labor-intensive fingerprint acquisition process. Then in the localization phase, SateLoc uses a novel weighted combination strategy to combine the fingerprinting maps of multiple gateways and output a joint location likelihood distribution for node localization. We implement SateLoc with commercial LoRa devices in a 227,500-m² urban area. The experimental results show that SateLoc achieves a median localization error of 43.5 m and a median tracking error of 37.9 m, while requiring little human effort.

REFERENCES

- [1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. TensorFlow: A system for large-scale machine learning. In *Proceedings of OSDI'16*. 265–283.
- [2] Michiel Aernouts, Rafael Berkvens, Koen Van Vlaenderen, and Maarten Weyn. 2018. Sigfox and LoRaWAN datasets for fingerprint localization in large urban and rural areas. *Data* 3, 2 (2018), 13.
- [3] Arduino. 2019. ARDUINO UNO REV3. Retrieved June 5, 2021 from <https://store.arduino.cc/usa/arduino-uno-rev3>.
- [4] Aloÿs Augustin, Jiazi Yi, Thomas Clausen, and William Townsley. 2016. A study of LoRa: Long range and low power networks for the Internet of Things. *Sensors* 16, 9 (2016), 1466.
- [5] Roshan Ayyalasomayajula, Deepak Vasisht, and Dinesh Bharadia. 2018. BLoc: CSI-based accurate localization for BLE tags. In *Proceedings of CoNEXT'18*. ACM, New York, NY, 126–138.
- [6] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. 2017. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 12 (2017), 2481–2495.
- [7] Jan Blumenthal, Ralf Grossmann, Frank Golatowski, and Dirk Timmermann. 2007. Weighted centroid localization in Zigbee-based sensor networks. In *Proceedings of WISP'07*. IEEE, Los Alamitos, CA, 1–6.
- [8] Martin C. Bor, Utz Roedig, Thiemo Voigt, and Juan M. Alonso. 2016. Do LoRa low-power wide-area networks scale? In *Proceedings of ACM MSWiM'16*. ACM, New York, NY, 59–67.
- [9] Jack Bresenham. 1977. A linear algorithm for incremental digital display of circular arcs. *Communications of the ACM* 20, 2 (1977), 100–106.
- [10] D. Fernandes Carvalho, A. Depari, P. Ferrari, A. Flammini, S. Rinaldi, and E. Sisinni. 2018. On the feasibility of mobile sensing and tracking applications based on LPWAN. In *Proceedings of IEEE SAS'18*. IEEE, Los Alamitos, CA, 1–6.
- [11] Marco Centenaro, Lorenzo Vangelista, Andrea Zanella, and Michele Zorzi. 2016. Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios. *IEEE Wireless Communications* 23, 5 (2016), 60–67.
- [12] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. 2017. Rethinking atrous convolution for semantic image segmentation. arXiv:1706.05587.
- [13] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of ECCV'18*. 801–818.
- [14] Wongeon Choi, Yoon-Seop Chang, Yeonuk Jung, and Junkeun Song. 2018. Low-power LoRa signal-based outdoor positioning using fingerprint algorithm. *ISPRS International Journal of Geo-Information* 7, 11 (2018), 440.
- [15] LoRa Alliance Strategy Committee. 2018. *LoRaWAN Geolocation Whitepaper*. LoRa Alliance.
- [16] Silvia Demetri, Marco Zúñiga, Gian Pietro Picco, Fernando Kuipers, Lorenzo Bruzzone, and Thomas Telkamp. 2019. Automated estimation of link quality for LoRa: A remote sensing approach. In *Proceedings of ACM/IEEE IPSN'19*.

- [17] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raskar. 2018. DeepGlobe 2018: A challenge to parse the earth through satellite images. In *Proceedings of IEEE CVPR Workshops'18*.
- [18] Adwait Dongare, Craig Hesling, Khushboo Bhatia, Artur Balanuta, Ricardo Lopes Pereira, Bob Iannucci, and Anthony Rowe. 2017. OpenChirp: A low-power wide-area networking architecture. In *Proceedings of IEEE PerCom Workshops'17*. IEEE, Los Alamitos, CA, 569–574.
- [19] Dragino. 2017. LoRa Wireless Mote, Gateway. Retrieved June 5, 2021 from <http://www.dragino.com/products/lora.html>.
- [20] Bernat Carbonés Fargas and Martin Nordal Petersen. 2017. GPS-free geolocation using LoRa in low-power WANs. In *Proceedings of IEEE GIoTS'17*. IEEE, Los Alamitos, CA, 1–6.
- [21] Andrea Goldsmith. 2005. *Wireless Communications*. Cambridge University Press.
- [22] Chaojie Gu, Linshan Jiang, and Rui Tan. 2018. LoRa-Based localization: Opportunities and challenges. arXiv:1812.11481.
- [23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of IEEE CVPR'16*. 770–778.
- [24] Sokratis Kartakis, Babu D. Choudhary, Alexander D. Gluhak, Lambros Lambrinos, and Julie A. McCann. 2016. Demystifying low-power wide-area communications for city IoT applications. In *Proceedings of ACM WiNTECH'16*. ACM, New York, NY, 2–8.
- [25] Ka-Ho Lam, Chi-Chung Cheung, and Wah-Ching Lee. 2017. LoRa-based localization systems for noisy outdoor environment. In *Proceedings of IEEE WiMob'17*. IEEE, Los Alamitos, CA, 278–284.
- [26] Ka-Ho Lam, Chi-Chung Cheung, and Wah-Ching Lee. 2018. New RSSI-based LoRa localization algorithms for very noisy outdoor environment. In *Proceedings of IEEE COMPSAC'18*, Vol. 2. IEEE, Los Alamitos, CA, 794–799.
- [27] Yao Li and Luo Chen. 2019. Land cover classification for high resolution remote sensing images with atrous convolution and BFS. In *Proceedings of IEEE ICCC'19*. IEEE, Los Alamitos, CA, 1808–1813.
- [28] You Li, Zhe He, Yuqi Li, Hongliang Xu, Ling Pei, and Yu Zhang. 2018. Towards location enhanced IoT: Characterization of LoRa signal for wide area localization. In *Proceedings of IEEE UPINLBS'18*. IEEE, Los Alamitos, CA, 1–7.
- [29] Jansen Liando, Amalinda Gamage, Agustinus Tengoutius, and Mo Li. 2019. Known and unknown facts of LoRa: Experiences from a large-scale measurement study. *ACM Transactions on Sensor Networks* 15, 2 (2019), Article 16, 35 pages.
- [30] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. 2017. RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of IEEE CVPR'17*. 1925–1934.
- [31] Yuxiang Lin, Wei Dong, Yi Gao, and Tao Gu. 2020. SateLoc: A virtual fingerprinting approach to outdoor LoRa localization using satellite images. In *Proceedings of ACM/IEEE IPSN'20*.
- [32] Qinghui Liu, Michael Kampffmeyer, Robert Jenssen, and Arnt-Børre Salberg. 2020. Dense dilated convolutions' merging network for land cover classification. arXiv:2003.04027.
- [33] Lei Ma, Manchun Li, Xiaoxue Ma, Liang Cheng, Peijun Du, and Yongxue Liu. 2017. A review of supervised object-based land-cover image classification. *ISPRS Journal of Photogrammetry and Remote Sensing* 130 (2017), 277–293.
- [34] Rajalakshmi Nandakumar, Vikram Iyer, and Shyamnath Gollakota. 2018. 3D localization for sub-centimeter sized devices. In *Proceedings of ACM SenSys'18*. ACM, New York, NY, 108–119.
- [35] Pierre Neumann, Julien Montavont, and Thomas Noël. 2016. Indoor deployment of low-power wide area networks (LPWAN): A LoRaWAN case study. In *Proceedings of IEEE IPSN WiMob'16*. IEEE, Los Alamitos, CA, 1–8.
- [36] Yao Peng, Longfei Shangguan, Yue Hu, Yujie Qian, Xianshang Lin, Xiaoqiang Chen, Dingyi Fang, and Kyle Jamieson. 2018. PLoRa: A passive long-range data network from ambient LoRa transmissions. In *Proceedings of ACM SIGCOMM'18*. ACM, New York, NY, 147–160.
- [37] Juha Petajajarvi, Konstantin Mikhaylov, Antti Roivainen, Tuomo Hanninen, and Marko Pettissalo. 2015. On the coverage of LPWANs: Range evaluation and channel attenuation model for LoRa technology. In *Proceedings of IEEE ITST'15*. IEEE, Los Alamitos, CA, 55–59.
- [38] Nico Podevijn, David Plets, Jens Trogh, Luc Martens, Pieter Suanet, Kim Hendrikse, and Wout Joseph. 2018. TDoA-based outdoor positioning with tracking algorithm in a public LoRa network. *Wireless Communications and Mobile Computing* 2018, 7 (2018), 1–9.
- [39] RadioHead. 2019. RadioHead Packet Radio Library for Embedded Microprocessors. Retrieved June 5, 2021 from <https://github.com/dragino/RadioHead>.
- [40] Alexander Rakhlis, Alex Davydow, and Sergey I. Nikolenko. 2018. Land cover classification from satellite imagery with U-Net and Lovasz-softmax loss. In *Proceedings of CVPR Workshops'18*. 262–266.
- [41] Theodore S. Rappaport. 1996. *Wireless Communications: Principles and Practice*. Vol. 2. Prentice Hall PTR, Hoboken, NJ.

- [42] Hazem Sallouha, Alessandro Chiumento, and Sofie Pollin. 2017. Localization in long-range ultra narrow band IoT networks using RSSI. In *Proceedings of IEEE ICC'17*. IEEE, Los Alamitos, CA, 1–6.
- [43] Semtech. 2019. Semtech SX1278. https://www.semtech.com/uploads/documents/DS_SX1276-7-8-9_W_APP_V6.pdf.
- [44] Statista. 2019. Internet of Things—Number of Connected Devices Worldwide 2015–2025. Retrieved June 5, 2021 from <https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/#statisticContainer>.
- [45] Masashi Sugano, Tomonori Kawazoe, Yoshikazu Ohta, and Masayuki Murata. 2006. Indoor localization system using RSSI measurement of wireless sensor network based on Zigbee standard. *Wireless and Optical Communications* 538 (2006), 1–6.
- [46] Chao Tian, Cong Li, and Jianping Shi. 2018. Dense fusion classmate network for land cover classification. In *Proceedings of IEEE CVPR Workshops'18*. 262–264.
- [47] Jin Wang, Nicholas Tan, Jun Luo, and Sinno Jialin Pan. 2017. WOLoc: WiFi-only outdoor localization using crowd-sensed hotspot labels. In *Proceedings of IEEE INFOCOM'17*. IEEE, Los Alamitos, CA, 1–9.
- [48] Chenshu Wu, Jingao Xu, Zheng Yang, Nicholas D. Lane, and Zuwei Yin. 2017. Gain without pain: Accurate WiFi-based localization using fingerprint spatial gradient. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 2 (2017), 29.
- [49] Fusang Zhang, Zhaoxin Chang, Kai Niu, Jie Xiong, Beihong Jin, Qin Lv, and Daqing Zhang. 2020. Exploring LoRa for long-range through-wall sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 2 (2020), 1–27.
- [50] Yuan Zhuang, Jun Yang, You Li, Longning Qi, and Naser El-Sheimy. 2016. Smartphone-based indoor localization with Bluetooth Low Energy beacons. *Sensors* 16, 5 (2016), 596.

Received July 2020; revised April 2021; accepted April 2021