RampScope: Ramp-level Localization of Shared Mobility Devices using Sidewalk Ramps

Jonghyuk Yun*
jhyoun96@yonsei.ac.kr
Yonsei University

Gyuyeon Kim*
msmhi52@yonsei.ac.kr
Yonsei University

Soundarya Ramesh
sramesh@comp.nus.edu.sg
National University of Singapore

Jun Han
jun.han@yonsei.ac.kr
Yonsei University

ABSTRACT

Short-term rentals of *shared mobility devices (SMDs)* including bikes, e-bikes, and e-scooters are gaining significant popularity across different countries. These services equip their SMDs with GPS receivers which allows the riders the flexibility to park their SMDs anywhere, and the next user simply finds the nearest parked SMDs. However, GPS accuracy decreases significantly in urban areas and causes real-world problems (e.g., users and chargers not being able to locate the SMDs). To overcome this problem, we propose *RampScope* that utilizes physical characteristics of *ramps* on the sidewalks – which are prevalent in urban areas – to correct for GPS error. As the user rides over a sidewalk ramp, the SMD equipped with a gyroscope captures the motion signal to uniquely identify the ramp and localizes the SMD to the nearest driven ramp. As a proof-of-concept, we present a preliminary evaluation of *RampScope* with real-world experiments by driving three different SMD types over 800 m to demonstrate an average ramp prediction accuracy of 98.1%.

CCS CONCEPTS

• **Information systems** → **Global positioning systems**; • **Human-centered computing** → **Ubiquitous and mobile computing systems and tools**.

KEYWORDS

Shared Mobility Device, Outdoor Localization, Sidewalk Ramp

ACM Reference Format:
Jonghyuk Yun, Gyuyeon Kim, Soundarya Ramesh, and Jun Han. 2023. *RampScope*: Ramp-level Localization of Shared Mobility Devices using Sidewalk Ramps. In *The 24th International Workshop on Mobile Computing Systems and Applications (HotMobile '23), February 22–23, 2023, Newport Beach, CA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3572864.3580334

*Both authors contributed equally to this research.

HotMobile '23, February 22–23, 2023, Newport Beach, CA, USA

ACM ISBN 979-8-4007-0017-0/23/02…$15.00
https://doi.org/10.1145/3572864.3580334

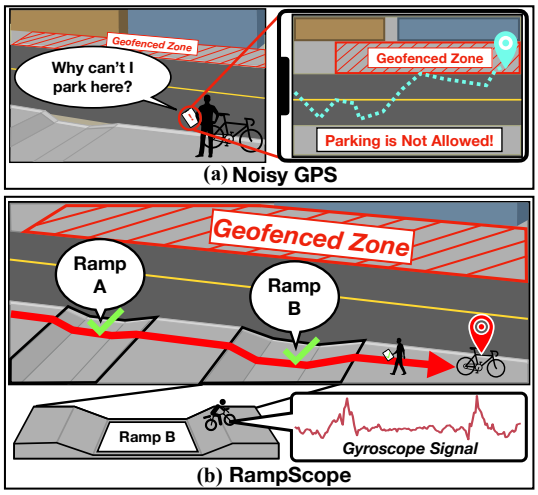

Figure 1: Figure depicts an SMD parking scenario, comparing (a) the *noisy GPS* and (b) *RampScope*. (a) GPS mislocates the SMD to the opposite sidewalk, i.e., in the geofenced area, hence preventing the rider from parking. (b) *RampScope* leverages gyroscope signals from the SMD traversing over sidewalk ramps to identify the correct sidewalk and the nearest driven ramp, thereby enabling successful parking of the SMD.

1 INTRODUCTION

Short-term rentals of *shared mobility devices (SMDs)* including bikes, e-bikes, and e-scooters– also known as *shared mobility services* – are gaining significant popularity, and their market is projected to increase to $500 billion by 2030 [9]. This is because they facilitate a convenient, low-cost, and environment-friendly solution for the first/last-mile commute compared to taxis or other ride-sharing services [3].

To further maximize user convenience, many of the shared mobility companies employ a *dockless system*, where the users are free to park the SMDs at any public locations. To accommodate the dockless system, however, the companies equip each of their vehicles with GPS receivers for accurate localization. This is so that the next user or their chargers can find the parked SMDs in their vicinity, utilizing the companies' smartphone applications. Furthermore, many companies implement *geofencing* to notify and prevent their users from parking or driving in restricted areas [17].

Unfortunately, GPS suffers from *urban canyon effect*, where high-rise buildings block line-of-sight paths from the satellites to the GPS receivers as well as causing multi-path effects that result in localization errors up to about two hundred meters [2, 10]. This causes critical problems for the shared mobility services. As the SMDs are frequently ridden and parked on the bike lanes on sidewalks, the noisy GPS readings often incorrectly localize them to opposite sidewalks of a road or even to adjacent roads. This is reported to cause significant confusion to the users and chargers trying to find the SMDs. Furthermore, the users may be misguided to park or drive into the geofenced locations (as depicted in Figure 1(a)), leading to numerous illegal parking problems for municipalities and ultimately significant monetary loss for shared mobility services (§2.1).

While solutions exist to remedy GPS errors, none of them sufficiently resolve the problem. For example, *map-matching*, which snaps noisy GPS coordinates to the most likely point on the road, is still error prone, especially in dense urban areas where the sidewalks are compactly co-located [19].

To overcome the aforementioned problems, we ask the following question – *can we suggest an efficient solution that can help the shared mobility services to more accurately localize their SMDs?* To answer this question, we propose *RampScope* that utilizes physical characteristics of sidewalk ramps that can augment the noisy GPS readings, to ultimately localize the SMDs more accurately. Specifically, we make use of an observation that the different ramps on a sidewalk yield relatively unique *signatures* that can be captured by the gyroscopes equipped on the SMDs. Given the prevalence of these ramps, *RampScope* proposes a ***ramp-level localization***, namely accurately localizing the SMDs to the nearest driven ramp of the correct sidewalk. Figure 1(b) depicts the exemplary scenario. As the user rides over a ramp, the captured gyroscope signal pinpoints the exact ramp location, and resets the corresponding GPS error.

Designing *RampScope*, however, comes with difficult challenges. First, it is extremely difficult to obtain a representative but unique *signature* of each ramp from the gyroscope data collected from different SMD types – i.e., bikes, e-bikes, and e-scooters– with varying drivers and driving speeds. To solve this challenge, *RampScope* employs a *Bootstrapping phase* that collects multiple ramp signals and utilizes Dynamic Time-Warping Barycenter Averaging (DBA) to preserve the underlying characteristics of the ramp signals. Second, given many sources of noise from the ramp signals alone (e.g., different driving speed, pattern, and entrance/exit points on ramps), the prediction may not always be accurate. We correct the possible prediction errors by utilizing the phenomenon that the ramps are geographically positioned in a distinctive manner on each sidewalks.

To demonstrate the feasibility of *RampScope*, we present a preliminary evaluation with a total of three different SMD types, namely bike, e-bike, and e-scooter, driven across an 800 m route, consisting of eight different ramps (with a total of 24 ramps in the vicinity of the route) and obtain an average ramp prediction accuracy of 98.1%.

2 BACKGROUND

We present the relevant background and a feasibility study.

2.1 Real-World Problems

We present real-world problems of noisy GPS on shared mobility services. First, GPS inaccuracies can misguide the users riding the SMDs to ***enter geofenced areas***. These services enforce geofencing to prevent parking as well as driving in certain geofenced areas (due to government mandates as well as unserviced areas instigated by the mobility companies). Figure 1(a) illustrates the scenario where the SMD is incorrectly localized to a geofenced area, hence incorrectly prohibiting the user from parking the SMD. In addition to the parking scenario, misidentifying geofenced areas may significantly degrade usability while riding the SMDs. If the SMD is incorrectly localized to a geofenced (i.e., unserviced) area, many services automatically reduce the SMD speed or even stop its operation while the user is riding it [17, 22]. Both scenarios degrade the user experience leading to potential monetary loss for the shared mobility services.

Second, the GPS errors increase the time taken by next users or chargers [12] in ***locating parked SMDs***. In particular, the app may incorrectly pinpoint the location of SMDs on the other side of the sidewalk or in the middle of the roadway, causing increased user confusion and frustration. Additionally, such errors may also lead to monetary losses for users who *reserve* SMDs ahead of time, as they may have to pay extra charges while searching the incorrectly located SMD [4]. To address such problems, *RampScope* proposes to utilize the uniqueness of ramps to improve the localization accuracy to the nearest driven ramp.

2.2 Utilizing Ramps for Localization.

Prevalence of Ramps. We refer to a ***ramp*** as a road element that consists of a downward slope, a flat road, and an upward slope, while occasionally including other components such as curbs and gutters (Figure 2(a)). Ramps facilitate a safe transition between a sidewalk and a roadway, and are prevalent on sidewalks (e.g., building/parking entrances, crosswalks, intersections; see ramp images depicted in Figure 2(b)). To provide an intuition of the prevalence of ramps, we randomly calculate the average number of ramps in densely populated cities to be about 23 ramps within a kilometer segment – i.e., observing a ramp in every 40 meters on a sidewalk[1].

Ramp-level Localization. *RampScope* utilizes the prevalence of ramps to provide *ramp-level localization*. *RampScope* localizes the SMD to the nearest driven ramp and *bounds* the errors to a single (and correct) sidewalk. As depicted in Figure 1(a), the GPS readings may erratically vary in two dimensions, namely along the width and length of the road, hence mislocating the SMD to the geofenced sidewalk. Contrarily, *RampScope* successfully identifies the ramps ridden by the user (Figure 1(b)), and correspondingly locates the SMD to the *nearest ramp* on the *correct sidewalk*.

[1]Given the lack of existing documents, we randomly pick one-kilometer segments of sidewalks from three densely populated cities, namely San Francisco, Seoul, and Singapore, and count the number of ramps in the segments from Google Maps to be 29, 21, and 20, respectively.

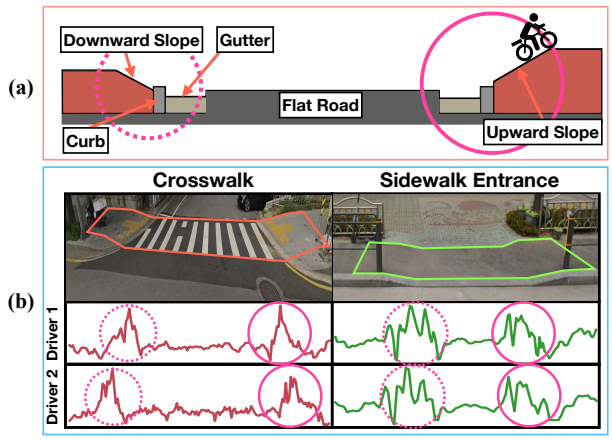

Figure 2: Figure depicts (a) components of the ramp that causes significant variations along with entry and exit points and (b) the uniqueness of gyroscope readings, where signals from the same ramp are alike while being distinct across two ramps.

2.3 Feasibility Study

To demonstrate *RampScope*'s feasibility, we perform a set of experiments on two types of ramps, namely a crosswalk and a sidewalk entrance as depicted in Figure 2(b). We observe the uniqueness of ramp signals (captured by the equipped gyroscope), having significant variations generated around the entry and exit points of the ramp. The ramp signals of two drivers riding over the same ramp are similar while being distinct across the two ramps. We attribute this phenomenon to the differences in ramp components including the angles, lengths, and heights of slopes, curbs, and gutters [14]. These components induce minute angular differences that are most efficiently captured by a gyroscope. We note that accelerometers counterintuitively incur more noise in *RampScope*'s use cases as they are more responsive to lateral movements.

3 SYSTEM DESIGN

RampScope leverages the uniqueness of ramp signals captured from a gyroscope sensor to augment the noisy GPS readings in order to achieve ramp-level localization of *shared mobility devices (SMDs)*. Figure 3 depicts the overview of *RampScope*'s design, which consists of two phases, namely – *Bootstrapping* and *Localization* phases. *Bootstrapping phase* (§3.1) utilizes gyroscope data from multiple traversals of each ramp, to learn the representative *ramp signature*. *Localization* phase (§3.2) takes as input – gyroscope and GPS data from an SMD traversing an unknown route, as well as *ramp signatures* (from *Bootstrapping* phase), to output the ramp-level localization (i.e., nearest driven ramp).

3.1 Bootstrapping Phase

Bootstrapping phase takes as input gyroscope signals from SMDs to compute *ramp signatures* for all ramps of interest along with the known ramp lat-lon coordinates. On a high level, the *Pre-processing module (§3.1.1)* performs noise-removal and extracts *ramp signals* corresponding to the individual ramps. Subsequently, the *Ramp Signature Generation* module (§3.1.2) computes representative *ramp signatures* by incorporating multiple ramp signals of each ramp. Finally, the *ramps' signatures* along with their ground-truth (lat, lon) location are saved to the cloud.

3.1.1 Pre-processing. Given the gyroscope signals from an SMD traversing a route with several ramps, this module outputs their *individual ramp signals*. First, we reduce high-frequency noise in the gyroscope signal ($x[t]$) by utilizing *Exponentially Weighted Moving Average* to obtain a smoothed signal ($y[t]$) as follows: $y[t] = \alpha \times y[t-1] + (1-\alpha) \times x[t]$. We set the smoothing factor, α, to 0.99, as it achieves a good trade-off in retaining ramp's uniqueness, while removing noise due to differing riding patterns and mobility types. Subsequently, we identify the *ramp signals* by determining their boundaries based on the significant variations observed in gyroscope signal (as depicted in the signals from Figure 2(b)). Finally, we equalize the duration of ramp signals and normalize their amplitudes using z-score normalization, to facilitate subsequent signal comparisons. In particular, we equalize the duration of each ramp signal by re-sampling it to the average duration (namely, $\mathcal{L}$) of bootstrapping ramp signals.

3.1.2 Ramp Signature Generation. We compute the representative *ramp signatures* per ramp, leveraging *ramp signals* from multiple traversals (as depicted in the corresponding module of Figure 3). Learning a good ramp representation is challenging due to varying riding speeds, causing peaks in *ramp signals* to occur at different times. Hence, using a simple averaging technique result in distorted patterns. We solve this by leveraging Dynamic Time-Warping Barycenter Averaging (DBA) to compute the *ramp signature*, as it preserves the underlying characteristics of the *ramp signals* [16]. In particular, the DBA algorithm finds the optimal average signal that minimizes the sum of squared DTW distance among ramp signals [21]. Furthermore, we apply *Sakoe-Chiba* band (window size = $0.2\mathcal{L}$), to narrow the warping window, thereby improving computational efficiency [7].

3.2 Localization Phase

This phase predicts accurate ramp-level location of a SMD traversing an unknown route. First, in the *Pre-processing* module (§3.1.1), we detect *ramp signals* from the SMD's sensor data. Then, in the *Search Space Reduction* module (§3.2.1), we leverage the rider's noisy GPS readings to reduce the search space to *candidate ramps*. Subsequently, in the *Ramp Prediction* module (§3.2.2), we compare the *ramp signal* with the *ramp signatures* of candidate ramps, to predict the ramp with *maximum* similarity. Finally, the *Connectivity-based Correction* (§3.2.3) module further improves ramp predictions by incorporating the geographical *connectivity* of ramps.

3.2.1 Search Space Reduction. This module utilizes the noisy GPS to reduce the search space to candidate ramps by finding the ramps within the GPS error boundary. By reducing the search space, we improve the ramp prediction accuracy and reduce the overall computation cost in subsequent modules. Figure 4(a) depicts an example of the rider traversing a route and riding over three ramps, R_1, R_3, and R_5. The first detected ramp, R_1, has the GPS error boundary, B_1, which includes candidate ramps, R_1, R_2. This in turn yields four

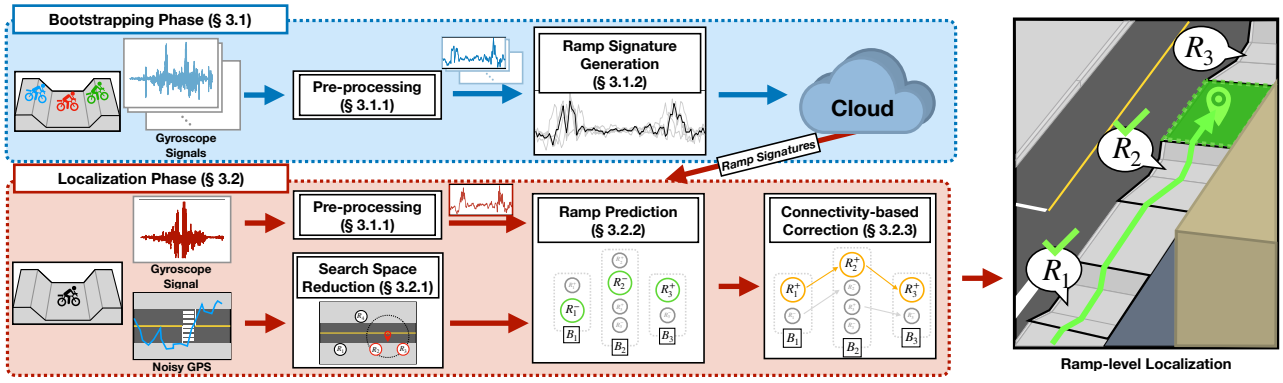

Figure 3: Figure depicts the overview of *RampScope*'s system design, consisting of two phases, namely the *Bootstrapping* and *Localization* phases. In the *Bootstrapping* phase, we leverage gyroscope signals from multiple traversals of each ramp to compute a representative *ramp signature*, while in the *Localization* phase, we input gyroscope signals from SMDs traversing an unknown route to output their accurate ramp-level location.

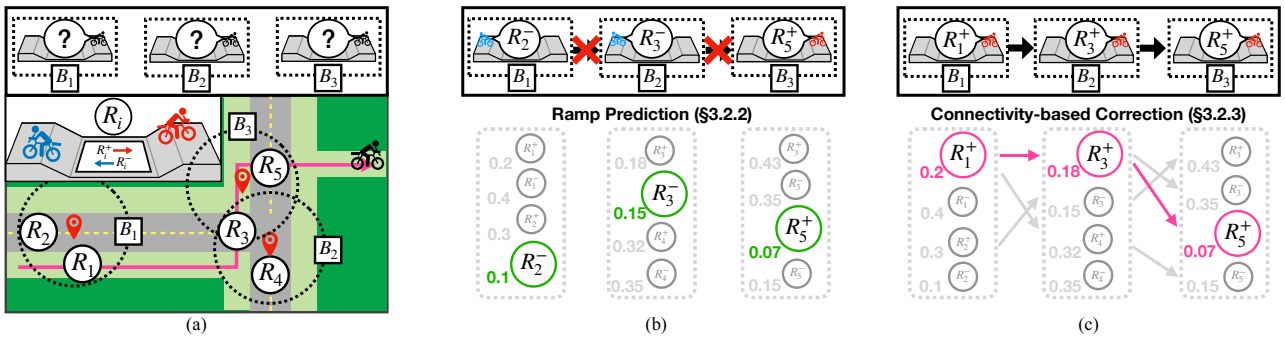

Figure 4: (a) depicts the scenario of an SMD rider traversing through three ramps, and their respective GPS error boundaries. (b) depicts the *ramp prediction* module that outputs the most likely ramp based on the highest similarity to underlying *ramp signatures*, and (c) depicts the *connectivity-based correction* module that further takes the geographical connectivity of ramps to provide a more accurate ramp-level location.

candidate *ramp signatures*, namely, R_1^+, R_1^-, R_2^+, R_2^-, where '+' and '-' denotes the two directions of ramp traversals. Note that we compute separate *ramp signatures* for the two different driving directions of the same ramp due to asymmetricity of ramp signals (assuming that the direction of travel is unknown) [2].

3.2.2 Ramp Prediction. We predict the most likely ramp, by taking as input the *ramp signal* and the candidate *ramp signatures*. We leverage DTW distance to compare the *ramp signal* with each of the candidate *ramp signatures*, and output the ramp with the least DTW distance (i.e, highest similarity). For example, Figure 4(b) depicts prediction of three ramps – R_2^-, R_3^-, and R_5^+, for the three detected ramps, respectively.

3.2.3 Connectivity-based Correction. We improve the ramp prediction by incorporating the underlying *connectivity* between ramps. Although the previous module outputs the ramp with the least DTW distance, it may still result in incorrect predictions due to noise in *ramp signals*, owing to different driving patterns, speed or entrance/exit points of ramps. We solve this problem by predicting a *sequence of connected ramps* that minimizes the overall DTW distance. Specifically, we generate a *route graph*, where for each detected ramp, we include nodes corresponding to ramps within its GPS boundary, $\mathcal{B}_t$. Each node has an associated *weight*, equal to the normalized DTW distance of the *ramp signal* with this node's (or ramp's) *signature*. Furthermore, we add a directed edge between nodes of adjacent boundaries (i.e., $\mathcal{B}_i$ and $\mathcal{B}_{i+1}$), if they are geographically connected. For example, in the route graph depicted in Figure 4(c), we connect nodes, R_1^+, and R_3^+, as forward (rightward) traversal ('+') of ramp, R_1, is connected to forward (upward) traversal ('+') of ramp, R_3 (from Figure 4(a)). Finally, we leverage Dijkstra's algorithm to identify a *sequence* of connected ramps with the least sum of DTW distances. Figure 4(c) highlights the final prediction, namely, ramps R_1^+, R_3^+, and R_5^+, which represent the rider's accurate ramp-level location.

[2]We set the '+' direction as (1) driving bottom to top (for lateral ramps), or (2) driving left to right (for horizontal ramps), and the other direction as '-'.

4 PRELIMINARY EVALUATION

We now present *RampScope*'s preliminary evaluation results.

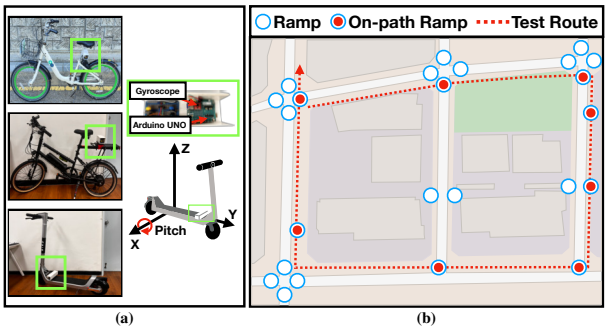

Figure 5: Figure depicts (a) the sensor module, as well as its affixed location on the different SMDs, and (b) the 800 m segment consisting of a total of 24 ramps (where eight are the *on-path* ramps).

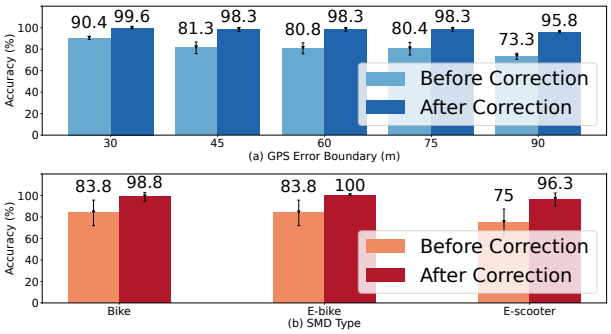

Figure 6: Figure depicts the *accuracy* for the ramps predicted by the *Ramp Prediction* (i.e., before correction; §3.2.2) and *Connectivity-based Correction* (i.e., after correction; §3.2.3) modules across (a) different GPS error boundaries, and (b) different SMD types.

4.1 Experiment Setup

Apparatus. We use three different types (totaling nine instances) of SMDs: ***bike*** - Alton City 260X Classic, ***e-bike*** - Alton Benzo 20 ST, and ***e-scooter*** - Bird Two and Ninebot ES2, with 24, 20, and 10 inch wheels, respectively. We affix the sensor module (i.e., MEMS gyroscope, LPY503AL, interfaced with an Arduino) on the SMDs (Figure 5(a)), to collect angular velocity (500Hz sampling rate) from its pitch axis.

Data Collection. We recruit two participants to collect data by riding the SMDs. For the *Bootstrapping* phase, one participant rides across 24 ramps (i.e., all ramps depicted in Figure 5(b)) for a total of five trials on all three SMD types. For the *Localization* phase, the other participant rides each SMD type, for a total of ten trials, along an 800-meter route containing eight ramps, as depicted by the trajectory in red in the figure. We enforce an upper-bound speed limit of 20km/h. We conduct the experiments by adhering to our university's Institutional Review Board (IRB) approval.

4.2 Preliminary Results

4.2.1 Differing GPS Error Boundaries. Recall that *RampScope* leverages GPS boundaries in order to reduce the space of *ramp signatures* (§3.2.1). In this experiment, we report the accuracy of ramp prediction across all three SMD types, as we arbitrarily vary the GPS boundary diameters from 30 m to 90 m. In particular, we compute the *ramp prediction accuracy* as the percentage of all ramps that are correctly identified. As depicted in Figure 6, *RampScope* achieves an average accuracy of 81.2% before correction, and an improved average accuracy of 98.1% after connectivity-based correction. Although the accuracy drops with increasing GPS error boundary, the after-correction accuracy is still above 95%, depicting the robustness of *RampScope* to GPS error.

4.2.2 Differing SMD Types. We evaluate *RampScope* for each of the aforementioned SMD types with a fixed GPS error boundary of 50 m. As depicted in Figure 6, *RampScope* achieves a high accuracy above 96% in all three cases, after the connectvity-based correction. Furthermore, we observe that e-scooter achieves the least accuracy of 75% and 96.3%, in the before and after correction settings, respectively. We attribute this result to their smaller wheel diameter of e-scooters which results in reduced damping, and subsequently increased noise in the gyroscope signals [5].

5 DISCUSSION

We discuss deployment considerations and future direction.

5.1 Deployment Considerations

Bootstrapping Data Collection. Recall from §3.1 that *RampScope* requires collecting data from multiple riders with known route information (i.e., lat-lon coordinates of the driven ramps). We envision utilizing the following resources to collect such data. The shared mobility service already hires employees or volunteer chargers (e.g., Lime Juicers [12]) to recharge their SMDs on a daily basis. Furthermore, if *RampScope* is deployed, we also envision crowdsourcing users' data and utilize online learning techniques to keep updating the *ramp signatures*. Moreover, we also envision utilizing autonomous delivery robots [13]. These data collection processes are synonymous to collecting and updating Google Street View data, but at a much smaller scale, as *RampScope* only needs to be operational in dense urban areas.

Saving Ramp Coordinates. *RampScope* locates SMDs to the lat-lon coordinate of the nearest ramp, saved in map data such as OpenStreetMap [15]. We envision that including ramp coordinates in the map data is certainly possible; some governments already release open-source data of the lat-lon coordinates of ramps [14]. Also, one study develops the ramp detection method from Google Street View, facilitating ramp coordinate inclusion in map data [8].

Driving on Sidewalks. *RampScope* localizes the SMDs only when they are driven over the ramps on sidewalks. While some countries allow SMDs to also be ridden on the roads, many riders choose to ride them on the sidewalks for safety reasons [20]. We can also incentivize the users to ride on the sidewalks to gain *RampScope*'s localization benefits.

Impact of Environmental Changes. We conjecture that as long as environmental changes (e.g., light snow or ice) do not distort the

differences in ramp components, the motion of SMDs is hardly affected. Furthermore, even if there are significant changes that incur failure of the ramp prediction (e.g., heavy snow), the connectivity-based correction algorithm further revises the mispredictions.

5.2 Future Directions

Through this work, we hint at the possibility of utilizing abundant yet seemingly insignificant road context (i.e., the unique shape of each ramp) captured by SMDs to provide additional functionality, namely reducing the GPS localization error. As a future direction, we envision utilizing additional sensors (e.g., accelerometers, magnetometers, and barometers) to capture a variety of road contexts – including elevation, slope, bumpiness, and manhole covers – for other novel applications. One such application is the detection of illegal parking without the need for additional infrastructure. To accurately identify illegal parking of SMDs, it is important to estimate the last location of the SMDs. This can be achieved through the implementation of dead-reckoning techniques, but these methods are prone to accumulate errors over time. We anticipate that incorporating road context information can reset these errors and enhance the precision of last location estimation of SMDs.

6 RELATED WORK

Several works utilize road context for improving localization accuracy and monitoring road conditions [1, 6, 11, 18]. For example, CARLOC, utilizes different road landmarks to augment dead-reckoning error to accurately localize cars on a road. While we are inspired by this work, we extend this further and capture the seemingly insignificant uniqueness of each ramps utilizing a gyroscope sensor. Furthermore, we extend *RampScope* from our prior work [18] to incorporate correction based on ramp *connectivity* making the localization system more robust against a variety of noise sources. Furthermore, we design *RampScope* to be robust against different SMD types.

7 CONCLUSION

We present *RampScope*, a novel ramp-level localization system for *shared mobility devices (SMDs)*, that is able to accurately localize the devices to the nearest driven ramp. *RampScope* utilizes the physical phenomenon that the sidewalk ramps are prevalent and are unique in their shape. *RampScope* leverages the gyroscope sensors equipped on the SMDs to capture the motion signal to uniquely identify the ramp and match it to its physical location (i.e., lat-lon coordinate). We present a preliminary evaluation to verify that *RampScope* is able to accurately identify and localize the ramps driven on a 800 m segment with an average ramp prediction accuracy of 98.1 %.

ACKNOWLEDGMENTS

We thank our shepherd and the reviewers for their insightful feedback. We also thank Bird Korea and Seojun Hong for providing experimental resources. This work is supported by Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2022-0-00420). Jun Han is the corresponding author of this work.

REFERENCES

[1] Heba Aly and Moustafa Youssef. 2013. Dejavu: an accurate energy-efficient outdoor localization system. In *ACM SIGSPATIAL*.
[2] A Angrisano, S Gaglione, and C Gioia. 2013. Performance assessment of GPS/GLONASS single point positioning in an urban environment. *Acta Geodaetica et Geophysica* (2013).
[3] Kwangho Baek, Hyukseong Lee, Jin-Hyuk Chung, and Jinhee Kim. 2021. Electric scooter sharing: How do people value it as a last-mile transportation mode? *Transportation Research Part D: Transport and Environment* (2021).
[4] Bird. 2022. Bird. https://www.bird.co/how/.
[5] Stefania Boglietti, Andrea Ghirardi, Chiara Turri Zanoni, Roberto Ventura, Benedetto Barabino, Giulio Maternini, and David Vetturi. 2022. First experimental comparison between e-kick scooters and e-bike's vibrational dynamics. *Transportation research procedia* (2022).
[6] Jakob Eriksson, Lewis Girod, Bret Hull, Ryan Newton, Samuel Madden, and Hari Balakrishnan. 2008. The pothole patrol: using a mobile sensor network for road surface monitoring. In *ACM MobiSys*.
[7] Zoltan Geler, Vladimir Kurbalija, Mirjana Ivanović, Miloš Radovanović, and Weihui Dai. 2019. Dynamic time warping: Itakura vs sakoe-chiba. In *2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*. IEEE.
[8] Kotaro Hara, Jin Sun, Robert Moore, David Jacobs, and Jon Froehlich. 2014. Tohme: detecting curb ramps in google street view using crowdsourcing, computer vision, and machine learning. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. 189–204.
[9] Kersten Heineke, Benedikt Kloss, Darius Scurtu, and Florian Weig. 2019. Micromobility's 15,000-mile checkup. *McKinsey & Company* (2019).
[10] Li-Ta Hsu. 2018. Analysis and modeling GPS NLOS effect in highly urbanized area. *GPS solutions* (2018).
[11] Yurong Jiang, Hang Qiu, Matthew McCartney, Gaurav Sukhatme, Marco Gruteser, Fan Bai, Donald Grimm, and Ramesh Govindan. 2015. Carloc: Precise positioning of automobiles. In *ACM SenSys*.
[12] Lime. 2022. Lime Juicer. https://lime.bike/juicer.
[13] Bernard Marr. 2020. Demand for these autonomous delivery robots is skyrocketing during this pandemic. https://www.forbes.com/sites/bernardmarr/2020/05/29/demand-for-these-autonomous-delivery-robots-is-skyrocketing-during-this-pandemic/?sh=4fac20fc7f3c.
[14] Department of Transportation. 2022. Pedestrian Ramp Locations. https://data.cityofnewyork.us/Transportation/Pedestrian-Ramp-Locations/ufzp-rrqu.
[15] OpenStreetMap contributors. 2017. Planet dump retrieved from https://planet.osm.org. https://www.openstreetmap.org.
[16] François Petitjean, Alain Ketterlin, and Pierre Gançarski. 2011. A global averaging method for dynamic time warping, with applications to clustering. *Pattern recognition* (2011).
[17] John Radcliff. 2020. Lime Introduces New Geofencing Technology, Setting Industry Standards For Scooters. https://www.li.me/en-es/blog/lime-introduces-new-geofencing-technology-setting-industry-standards-for-scooters.
[18] Christian August Reksten-Monsen and Jun Han. 2019. POSTER: Towards Precise Localization of E-Scooters Using Sidewalk Ramps (poster). In *ACM MobiSys*.
[19] Ming Ren and Hassan A Karimi. 2012. A fuzzy logic map matching for wheelchair navigation. *GPS solutions* (2012).
[20] et al. Sadeghinasr. 2021. Mining dockless bikeshare data for insights into cyclist behavior and preferences: Evidence from the Boston region. *Transportation research part D: transport and environment* (2021).
[21] Hiroaki Sakoe and Seibi Chiba. 1978. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing* (1978).
[22] Sonja Sharp. 2019. Did your rented e-scooter suddenly shut down? Blame the invisible geofence. https://www.latimes.com/california/story/2019-09-16/geofencing-scooters-westside-bird-lime.