Distance Estimation on Moving Object using BLE Beacon

Ching Hong Lam* and James She*†
*HKUST-NIE Social Media Lab, Department of Electronic and Computer Engineering
Hong Kong University of Science and Technology
†College of Science and Engineering, Hamad Bin Khalifa University, Qatar
Email: *{chlamaq,eejames}@ust.hk, †pshe@hbku.edu.qa

***Abstract*—The development of Internet of Things technology has connected the smart things to the Internet, enabling users to interact for different applications such as indoor positioning or location-based notification service. To improve the user experience, an accurate distance estimation is required to ensure the interaction can be delivered precisely. For general beacon-based application, the objects keep moving while they are interacting with the beacons. Therefore, their mobility needs to be considered for distance estimation. In this paper, comprehensive experiments are conducted to study the relationship between distance estimation accuracy and packet received rate from two angels, the beacon advertising interval and the object moving speed. Moreover, an improved distance estimation method using Kalman filter and support vector regression is proposed, which has archived at least 40% improvement comparing to current approaches. The proposed idea is also implemented in real-world application which archive less than $100\mu s$ computation time.**

***Index Terms*—Distance Estimation, Packet Receiving Rate, Advertising Interval, Moving Speed, BLE Beacon, Internet of Things**

(a) Cleaning robot (medium speed)

(b) Metro tracking (fast speed)

(c) Customized notification for mobile users (slow speed)

Fig. 1: Distance-based mobility-related application examples with different moving speeds.

I. Introduction

In the ecosystem of the Internet of Things (IoT), the things are able to interact with each other smartly through different IoT technologies [1] [2]. The smart things can be any electronic devices, which can detect or broadcast signal for interaction to provide certain services. Among all of the services, distance-based IoT application is one of them that are easiest to experience by the general public [3]. Examples like location-based service by delivering specific notifications to users when they are nearby some IoT devices [4] or the positioning system on users or robots [5]. Bluetooth Low Energy (BLE) Beacon has attracted lots of development on beacon related mobile applications since the iBeacon [6] is announced by Apple.Inc in 2013 due to its compatibility to the smartphone. BLE beacon is a device that broadcasts its identity Bluetooth signal periodically. Users' smartphones can detect the signal and perform specific actions which are designed by the service provider. As a battery-powered device, the beacon is easy to be deployed in different environments comparing to other technologies like RFID [7] or WiFi [8] which provide flexibility for the installation process. BLE beacon can be configured into different advertising interval, which decides how frequent the beacon broadcasts a packet. Shorter advertising interval can guarantee higher packet receiving rate, but it would trade off the battery lifetime. Usually, the beacon signal can cover around $10-15m$ depends on the configuration [9]. In some scenario, the users may not be able to locate the voucher if the promotion message is received by detecting the beacon. To enhance the user experience and the service delivery accuracy, the exact distance between the users and the BLE beacons is necessary to be estimated.

Received Signal Strength (RSS) is widely used for measuring the distance, which can be obtained from the smartphone directly. The RSS value indicates the strength of the signal transmitted from the beacon, and it decay when the distance towards the smartphone increases [10] [11]. RSS signal suffers from serious fluctuation due to effects like multi-path fading or shadowing [12] [13] that leads to distance estimation error. Generally, the path-loss model is used for estimating distance using RSS value as input for different wireless communication networks [14]. The path-loss model is based on an ideal assumption that the RSS would decrease exponentially with the actual distance under a line-of-sight condition. A path-loss model with given coefficients is provided in the *ALTBeacon open-source library* [15] for beacon related application development on the Android platform. However, in reality, the distribution of RSS is not precisely following the exponential curve, and the noisy signal will lead to serious estimation error by using the path-loss model directly.

978-1-7281-3316-4/19/$31.00 ©2019 IEEE

To reduce the signal fluctuation, a filtering process is usually applied for smoothing the signal from the noisy measurements [16]. Therefore, the packet receiving rate will directly affect how many samples can be used for filtering. By configuring the beacon to shorter advertising interval, it allows the smartphone to have more RSS samples to perform noise reduction and distance estimation. Besides, user mobility also affects the packet receiving rate on the receiver side. Considering the real-world situation, the smartphones or users are moving with certain speeds while they are interacting with the beacons. The number of packets received is limited for estimating distance. Therefore, precise distance is required to be computed with a few samples but still maintain accuracy.

In this paper, we conducted comprehensive experiments for analyzing the relationship between the packet receiving rate and distance estimation accuracy under different beacon advertising intervals and object moving speeds which covered almost all the possible BLE beacon applications. Nonetheless, a novel distance estimation method is proposed for improving the distance estimation accuracy on moving object. We applied the Kalman Filter (KF) to deal with the fluctuating RSS signals on the smartphone. After the filtered RSS value is input to a trained support vector regression (SVR) model for distance estimation. The experimental results show the estimation accuracy is improved at least 40% compared to the two benchmark functions and the proposed idea is implemented on a mobile application to demonstrate its feasibility. The proposed idea is abbreviated as KF-SVR. The contribution of this paper can be summarized by the following:

- Comprehensive experiments are conducted in a real-world setting to study the distance estimation accuracy under different packet receiving rate;
- An improved distance estimation method is proposed using KF to filter the fluctuated RSS and SVR for estimating the distance;
- The proposed idea is implemented in the real application to demonstrate the feasibility, and the performance is evaluated to verify its accuracy and real-time response.

The rest of the paper is organized as follows. Section II formulates the problem and describes the proposed idea. Section III explains the experimental setup and implementation details. Section IV presents the experimental results. Section V concludes the paper.

II. KF-SVR

In this section, the problem between the distance estimation accuracy and receiving rate is formulated. After that, the proposed KF-SVR is also described in detail. The notations used in this paper are shown in Table. I.

A. Problem Formulation

Before any computation process, signal acquisition is required for detecting the beacon signal to obtain the RSS value.

$$\mathbf{r}_{I,d_0,v} = (r_1, r_2, \cdots, r_N) \tag{1}$$

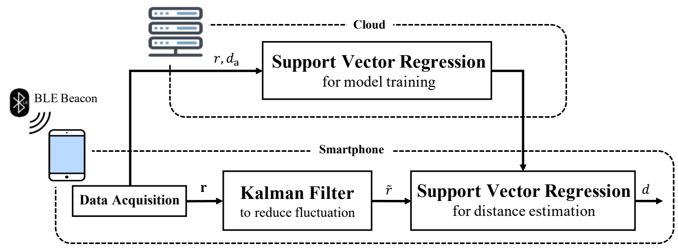


Fig. 2: The proposed KF-SVR. The KF and the trained SVR model are deployed on the smartphone for distance estimation.

where N is the number of packets detected over the scanning duration t_s. Denote that each RSS vector $\mathbf{r}$ are unique with the corresponding object moving speed v and beacon advertising interval I. After the RSS vector is obtained, the packet receiving rate can be calculated and expressed as the following:

$$\varphi = \frac{N}{t_s} \tag{2}$$

which indicate the average number of packets received every second.

As mentioned, a filtering process is needed for retrieving a less fluctuated RSS value, which can be any technique like averaging, min-max. In this paper, we used KF as the filter to deal with the noisy signal and obtain a filtered RSS $\tilde{r}$ for distance estimation. Once the filtered RSS value is obtained, the distance can be estimated by inputting to a distance model that is expressed as follows:

$$d_t = f(\tilde{r}) \tag{3}$$

where d_t is the distance of time $t < t_s$ and the model can be any kind of estimation model such as path-loss or trained non-linear model. Here, we used a trained support vector regression model for distance estimation, that will be further discussed in section II-C.

TABLE I: Related variables and their notations

Symbol	Meaning
$\mathbf{r}$	RSS vector
$\tilde{r}$	filtered RSS
r	RSS
t_s	scanning duration
N	packet number
φ	packet receiving rate
v	object moving speed
I	beacon advertising interval
d_m	moving distance
d_0	shortest distance between smartphone and beacon
d_t	estimated distance at time t
$\mathbf{F}$	transition matrix
K	Kalman gain
$\mathbf{z}$	measurement residual
$\mathbf{\Sigma}$	measurement noise covariance
σ	measurement noise
$\mathbf{H}$	transformation matrix
$\boldsymbol{\mathcal{E}}$	error covariance

B. Kalman Filter

This paper adopted KF to filter the fluctuated RSS measurements before performing distance estimation. KF is a lightweight filtering process that could effectively reduce the noise from the RSS samples by looping the RSS values with multiple iterations. The KF is deployed on the smartphone to reduce unnecessary network traffic to an external server, so the filtered the RSS value can be obtained once the RSS measurement arrives and input to the distance estimation model.

The filtering process can be divided into two phases, prediction and update. When the RSS sample is detected by the smartphome, the KF will first predict a filtered value based on the previous estimation, which can be expressed as the following equation,

$$\tilde{\mathbf{r}}_{\tau'} = \mathbf{F}\tilde{\mathbf{r}}_{\tau} + \sigma_{\tau} \tag{4}$$

where $\tilde{r}_\tau$ is the filtered RSS value of τ-th samples and σ_τ is the noise. The objective here is to predict the filtered value for the next iteration based on the current estimation. After the prediction phase, the next RSS measurement arrives and the KF will update the previous predicted value as output for distance estimation. The equations of the update phase are list below [17]:

- Update current estimation,

$$\tilde{\mathbf{r}}_{\tau} = \tilde{\mathbf{r}}_{\tau}^{-} + K_{\tau}(\mathbf{z}_{\tau} - \mathbf{H}_{\tau}\tilde{\mathbf{r}}_{\tau}^{-}) \tag{5}$$

- Update current error covariance,

$$\boldsymbol{\mathcal{E}}_{\tau}^{+} = (\mathbf{I} - K_{\tau}\mathbf{H}_{\tau})\boldsymbol{\mathcal{E}}_{\tau}^{-} \tag{6}$$

- Compute the Kalman gain, which decides how much the current estimation will be affected by previous result,

$$K_{\tau} = \boldsymbol{\mathcal{E}}_{\tau}^{-}\mathbf{H}_{\tau}^{T}(\boldsymbol{\Sigma}_{\tau} + \mathbf{H}_{\tau}\boldsymbol{\mathcal{E}}_{\tau}^{-}\mathbf{H}_{\tau}^{T})^{-1} \tag{7}$$

As mentioned, the KF is implemented on the smartphone so the filtered RSS can be obtained at nearly the same time after the raw RSS measurement arrived. Once the filtered value is obtained, the model will compute the distance right away. For the applications that the objects are moving, such kind of fast response time definitely could benefit the overall performance.

C. Support Vector Regression model

An estimation model is required for computing the distance using the filtered RSS from the KF. Instead of using traditional path-loss model, this paper adopted a machine learning approach - SVR for distance estimation. The model needs to be trained using the RSS samples and the actual distance d_a. In this paper, the Radial Basis Function (RBF) kernel is used, which can be stated as the following:

$$\kappa = e^{-\gamma\|\mathbf{r}-\mathbf{r}'\|^2} \tag{8}$$

and the training process is a minimization problem as shown below,

$$\begin{aligned} C(\alpha) = \min_{\alpha,\alpha^*} & \frac{1}{2}(\alpha-\alpha^*)^T Q(\alpha-\alpha^*) \\ & + \epsilon\sum_{i=1}^{l}(\alpha_i-\alpha_i^*) + \sum_{i=1}^{l} d_i(\alpha_i-\alpha_i^*) \\ \text{subject to} \quad & e^T(\alpha-\alpha^*) = 0 \\ & \forall l: 0 \le \alpha_i, \alpha_i^* \le C, i = 1,\ldots,l \\ \text{where} \quad & Q_{ij} = \kappa(\mathbf{r}_i,\mathbf{r}_j) = \phi(\mathbf{r}_i)^T\phi(\mathbf{r}_j) \end{aligned} \tag{9}$$

where C is the regularization term that controls the penalty of the RSS samples lie outside of the tolerance ϵ [18]. The dual problem is solved by a two-point optimization method provided by *scikit* [19]. As shown in Fig. 2, the model training process is performed on the cloud due to its better computation capability that allows the model to be retrained or improved quickly. The trained model will be deployed on the smartphone for distance estimation. The result from the estimation model can be expressed as follow,

$$d_t = \sum_{i=1}^{l}(-\alpha_i + \alpha_i^*)\kappa + b \tag{10}$$

Thus, the proposed KF-SVR can compute the estimated distance without any Internet connection which provides the flexibility for different application scenarios and highly reduced the network traffic required comparing to some existing approaches [20]. Besides the network condition, as the model is deployed on the smartphone, the estimated distance can be responded in real-time. Such fast response time is desirable for applications that users only have limited time to interact with the beacons. Details on the computation performance will be discussed later in section III-A

III. Implementation and Experimental Setup

In this section, the performance of the improved architecture and experimental setup are discussed.

A. Implementation

The proposed KF-SVR is implemented on a real-world mobile application for experiment and demonstrating its feasibility. The KF and SVR are both deployed on the smartphone side, KF is a lightweight filtering method that can be used to obtain the filtered RSS value right after the raw packet is detected. Hence, if the KF is deployed on the cloud, every RSS measurement is required to be sent to the server, which is not practical as it generates too much network traffic. For the SVR model, we only implemented the training phase on the cloud as this process is compute-intensive. The trained model will be downloaded to the mobile application for estimation. The experimental data are divided into 70% and 30% for cross-validation. The average computation time of the proposed KF-SVR, ALT-PL and NLPL is verified by using thousands of samples. The KF-SVR requires $28.7\mu s$ for KF and $36\mu s$ for SVR, and the general path-loss model used $26.6\mu s$. Therefore,

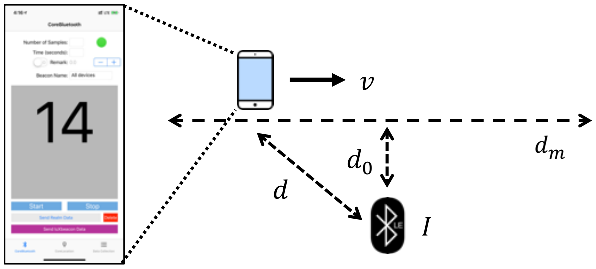


Fig. 3: Experiment setup and the screenshot of the mobile application for experiment.

the calculation time is nearly real-time and the time difference between the three methods is minimal.

B. Experimental Setup

To investigate the distance estimation accuracy under different packet receiving rate φ, we have conducted a huge set of experiments from two angles, the beacon advertising interval and the object moving speed to study the impact of the packet receiving rate.

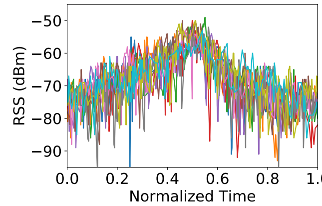
As shown in Fig. 3, the experiment is conducted with a smartphone moving with speed v for detecting the beacon signal and the beacon is broadcasting with interval I. A custom iOS mobile application is developed for data acquisition, and the user interface shows how many packets are recorded. For the experiments under high moving speed, we used a camera to capture the packet detection to make sure the preciseness of d_m, which is shown in Fig. 4. To ensure the accuracy of the speed in the experiment, the speed v is calculated by $v = d_m/\Delta t_s$, where the d_m is the distance that the object moved. Considering in real-world deployment, the users may interact with the beacons in different distances, so we have conducted the experiments with a set of 3 shortest distance $d_0 = \{1m, 2m, 5m\}$. While the actual speed may vary over each run, the average speed of 10 runs is used as the results.

IV. Experimental Result

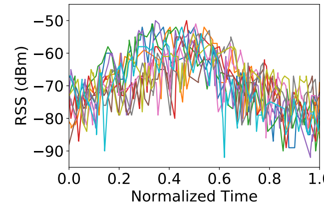
In this section, we first investigate the distance estimation accuracy under different settings, then compare the proposed KF-SVR with two benchmark functions.

A. Advertising Interval

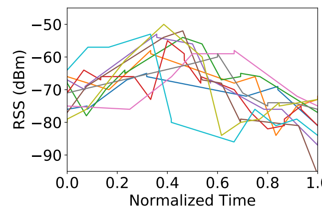
The advertising interval decides how fast the beacon broadcasts that could affect the number of packets re-



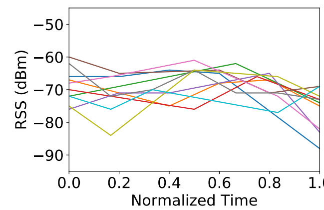
(a) $I = 20ms$



(b) $I = 100ms$



(c) $I = 1000ms$



(d) $I = 2000ms$

Fig. 5: RSS changes over time when the smartphone is moving towards and away from the beacon with different I.

ceived by the smartphone. We have conducted the experiment in total 6 different advertising interval $I = \{20ms, 50ms, 100ms, 500ms, 1000ms, 2000ms\}$. For most of the applications, beacons are configured to broadcast with an interval between $100ms$ to $1000ms$. We used the path-loss model from the *Alt-Beacon library* to compare the distance estimation accuracy by calculating the absolute distance that can be expressed as the following:

$$e(d_t) = ||d_t - d_a|| \tag{11}$$

where d_t is the estimated distance in time t and d_a is the actual distance.

Fig. 5 shows the RSS measurements with different packet receiving rates under the same speed of movement, i.e., human walking ($v = 2.5km/h$). It is expected that the smartphone can detect more signal from the $20ms$ beacon. As mentioned above, the user is moving towards and then away from the beacon. Hence, the measured RSS should increases in the first half while the distance d between the smartphone and beacon is getting closer. From the result, it is obvious that beacon with $20ms$ and $100ms$ are able to tell the users mobility comparing to $1000ms$ and $2000ms$. The experimental result of $d_0 = 1m$ is shown in Fig. 6a. The estimation accuracy is similar between $I = 20ms - 500ms$, then it starts to decrease when I is larger than $1000ms$. To further verify the impact from the advertising interval, the results of $d_0 = 2m, 5m$ are also shown in Fig. 6b and Fig. 6c.

B. Object Moving Speed

The object moving speed v also affect the packet receiving rate φ. According to the result from the previous section IV-A, any advertising interval shorter than $1000ms$ can obtain relatively accurate distance. In this section, we used a beacon with advertising interval $I = 100ms$ to conduct the experiments with various moving speed to ensure we can obtain enough

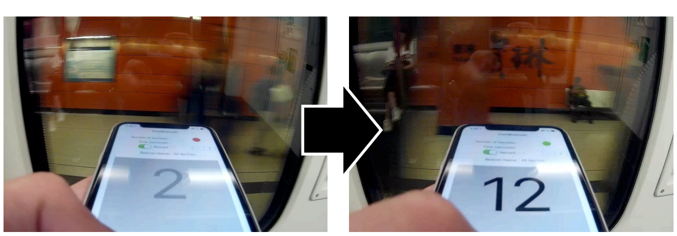
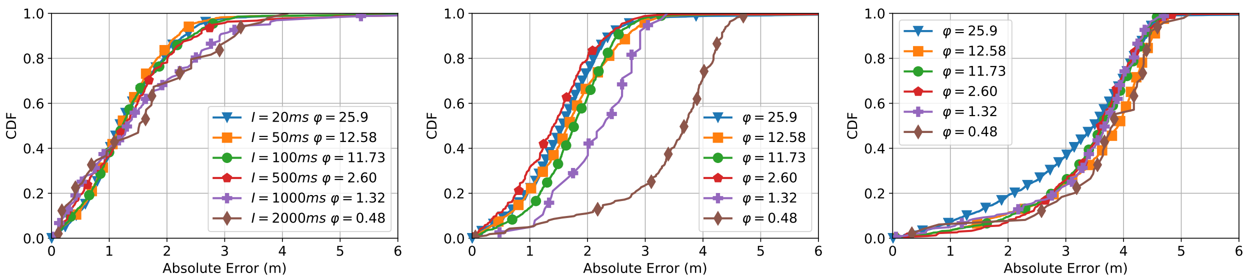


Fig. 4: Video screenshot of experiments with fast speed inside metro station. The number of received packets is indicated.



(a) $d_0 = 1m$ (b) $d_0 = 2m$ (c) $d_0 = 5m$

Fig. 6: Distance estimation accuracy with different packet received rate φ with different advertising intervals I.

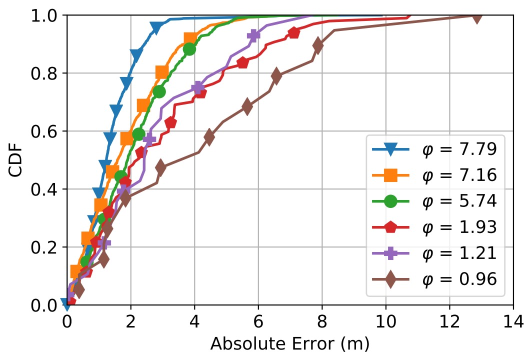


Fig. 7: Distance estimation accuracy with different packet received rate φ with different object moving speed v.

TABLE II: Moving speed and corresponding packet receiving rate

Moving Speed v	**Packet Receiving Rate φ**
2.5 km/h	7.79
2.9 km/h	7.16
3.8 km/h	5.74
7.0 km/h	1.93
23.6 km/h	1.21
45.4 km/h	0.96

RSS samples but still maintain the accuracy. The speed test covers mobility of human walking, remote-controlled vehicle and public transportation for simulating in a total of 6 speeds in a real-world setup. The moving speed v and its corresponding packet receiving rate φ is shown in Table. II.

From the result shown in Fig. 7, the distance estimation accuracy decreases with the packet receiving rate. Thus, more RSS samples can improve the estimation accuracy in overall. When the object moves faster than $45km/h$, the average number of packet received at the smartphone is less than one, where the result shows that more than 50% of time the absolute error is larger than $4m$.

C. Benchmark Functions

From the previous two sets of experiments, it is found that the distance estimation accuracy is related to the packet received rate. In this paper, we proposed a novel method using KF and SVR for distance estimation that aims to archive improved distance estimation accuracy on moving objects. To evaluate the performance of proposed KF-SVR, the following two benchmark functions (ALT-PL and NLPL) are used for comparison,

1) Alt-Beacon open-sourced library (ALT-PL): The *Alt-Beacon library* provides a distance estimation model based on the path-loss model with provided coefficient, the general form of the model is provided below,

$$d_t = \alpha \left(\frac{r}{r_0} \right)^{\beta} + \gamma \tag{12}$$

where r_0 is the reference RSS measured in $1m$. The corresponding value or α, β and γ are given in the library's document.

2) Non-linear Path-loss model (NLPL): Beside the open-sourced model, the same set of data is used to train a non-linear model which is the same as Eq. 12 using the curve-fitting technique. Instead of using the suggested value from *Alt-Beacon*, we use the coefficient obtained by our dataset using curve-fitting technique to construct a more accurate model as a benchmark.

D. Experimental Result

The proposed KF-SVR and the two benchmark functions are used to evaluate the accuracy with the same dataset. The 50^{th} and 90^{th} percentile of the CDF are used to compare the methods with the tested I, which are shown in Table. III. Comparing to ALT-PL and NLPL, the KF-SVR has in average more than 60% and 20% improvement on error reduction.

To further evaluate the performance of KF-SVR, the model is also tested under different object moving speed v. Besides the speed of human walking, we investigate the distance estimation performance for five more speed settings. The absolute error results of three estimation methods are shown in Table. IV. Comparing to ALT-PL, the KF-SVR has obtained more than 40% improvement on error reduction, and 20% comparing to NLPL.

V. Conclusion

Precise distance estimation is definitely essential for BLE beacon-based application. The service can be delivered ac-

TABLE III: Absolute distance error with different I at 50th and 90th percentile.

φ	I	Percentile	KF-SVR (Proposed)	NLPL	ALT-PL
25.9	$20ms$	90th	**1.92m**	2.01m	4.07m
		50th	**0.67m**	0.98m	1.82m
12.58	$50ms$	90th	**1.88m**	1.91m	4.29m
		50th	**0.94m**	1.93m	2.03m
11.73	$100ms$	90th	1.99m	**1.95m**	4.13m
		50th	**0.72m**	1.13m	2.11m
2.6	$500ms$	90th	2.18m	**1.93m**	4.08m
		50th	**0.74m**	1.21m	2.00m
1.32	$1000ms$	90th	**1.59m**	1.90m	4.33m
		50th	**0.74m**	1.52m	3.67m
0.48	$2000ms$	90th	**1.54m**	1.65m	4.33m
		50th	**0.88m**	1.26m	2.60m

TABLE IV: Absolute distance error with different v at 50th and 90th percentile.

φ	v	Percentile	KF-SVR (Proposed)	NLPL	ALT-PL
7.79	$2.5km/h$	90th	**1.01m**	1.55m	3.68m
		50th	**0.41m**	0.63m	1.58m
7.16	$2.9km/h$	90th	**1.88m**	1.91m	4.29m
		50th	**0.94m**	1.93m	2.03m
5.74	$3.8km/h$	90th	**1.18m**	1.70m	4.02m
		50th	**0.46m**	0.63m	1.89m
1.93	$7.0km/h$	90th	**5.82m**	**5.82m**	6.50m
		50th	**2.01m**	2.41m	2.25m
1.21	$23.6km/h$	90th	5.03m	**4.96m**	5.85m
		50th	**1.70m**	2.09m	2.44m
0.96	$45.4km/h$	90th	8.81m	**7.62m**	8.38m
		50th	**2.38m**	8.81m	4.13m

curately when an improved distance estimation can be applied for knowing how far the users are from the beacon. For most of the real-world applications, the user mobility leads to limited packet receiving rate for distance estimation. In this paper, comprehensive experiments are conducted to study the distance estimation accuracy with different packet receiving rate. An improved distance estimation method is also proposed and evaluated by multiple real-world settings that have archived more than 40% error reduction comparing to existing development framework. The proposed KF-SVR is also implemented for demonstrating the feasibility. Also, the response time is less than $100\mu s$ that is desirable for many applications, especially when the objects move quickly.

Acknowledgment

This work was initiated by HKUST-NIE Social Media Lab. at Hong Kong University of Science and Technology. Part of the research works was completed when Prof. James She was visiting the ICT Division in the College of Science and Engineering at Hamad Bin Khalifa University, Qatar.

References

[1] K. E. Jeon, J. She, P. Soonsawad, and P. C. Ng, “Ble beacons for internet of things applications: Survey, challenges and opportunities,” *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–1, 2018.

[2] S. Raza, P. Misra, Z. He, and T. Voigt, “Bluetooth smart: An enabling technology for the internet of things,” in *2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct 2015, pp. 155–162.

[3] R. Want and B. N. Schilit, “Interactive digital signage,” *Computer*, no. 5, pp. 21–24, 2012.

[4] P. C. Ng, J. She, and S. Park, “Notify-and-interact: A beacon-smartphone interaction for user engagement in galleries,” in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, July 2017, pp. 1069–1074.

[5] C. Feng, W. S. A. Au, S. Valaee, and Z. Tan, “Received-signal-strength-based indoor positioning using compressive sensing,” *Mobile Computing, IEEE Transactions on*, vol. 11, no. 12, pp. 1983–1993, 2012.

[6] ibeacon for developers. [Online]. Available: https://developer.apple.com/ibeacon/

[7] S. Evdokimov, B. Fabian, O. Gunther, L. Ivantysynova, and H. Ziekow, *RFID and the internet of things: Technology, applications, and security challenges*. Now Publishers Inc, 2011.

[8] W. Xue, W. Qiu, X. Hua, and K. Yu, “Improved wi-fi rssi measurement for indoor localization,” *IEEE Sensors Journal*, vol. 17, no. 7, pp. 2224–2230, April 2017.

[9] M. Sie and C. Kuo, “Indoor location estimation using ble beacon with multiple transmission power levels,” in *2017 IEEE International Conference on Consumer Electronics - Taiwan (ICCE-TW)*, June 2017, pp. 323–324.

[10] F. Zafari, I. Papapanagiotou, and T. Hacker, “A novel bayesian filtering based algorithm for rssi-based indoor localization,” in *IEEE ICC*, 2018.

[11] R. Faragher and R. Harle, “An analysis of the accuracy of bluetooth low energy for indoor positioning applications,” in *Proceedings of the 27th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+'14)*, 2014.

[12] D. Puccinelli and M. Haenggi, “Multipath fading in wireless sensor networks: Measurements and interpretation,” in *Proceedings of the 2006 International Conference on Wireless Communications and Mobile Computing*, ser. IWCMC '06. New York, NY, USA: ACM, 2006, pp. 1039–1044.

[13] M. Ayadi and A. B. Zineb, “Body shadowing and furniture effects for accuracy improvement of indoor wave propagation models,” *IEEE Transactions on Wireless Communications*, vol. 13, no. 11, pp. 5999–6006, Nov 2014.

[14] P. Kumar, L. Reddy, and S. Varma, “Distance measurement and error estimation scheme for rssi based localization in wireless sensor networks,” in *2009 Fifth International Conference on Wireless Communication and Sensor Networks (WCSN)*, Dec 2009, pp. 1–4.

[15] Android-beacon-library. [Online]. Available: https://github.com/AltBeacon/android-beacon-library

[16] C. H. Lam, P. C. Ng, and J. She, “Improved distance estimation with ble beacon using kalman filter and svm,” in *2018 IEEE International Conference on Communications (ICC)*, May 2018.

[17] A. Assa and F. Janabi-Sharifi, “A kalman filter-based framework for enhanced sensor fusion,” *IEEE Sensors Journal*, vol. 15, no. 6, pp. 3281–3292, June 2015.

[18] C.-C. Chang and C.-J. Lin, “Libsvm: A library for support vector machines,” *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, May 2011. [Online]. Available: http://doi.acm.org/10.1145/1961189.1961199

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[20] F. Zafari, I. Papapanagiotou, M. Devetsikiotis, and T. J. Hacker, “Enhancing the accuracy of ibeacons for indoor proximity-based services,” in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–7.