A Novel Smartphone Application for Indoor Positioning of Users based on Machine Learning

Elaheh Homayounvala[*]
Department of Computing,
Goldsmiths University of London, UK.
e.homayounvala@gold.ac.uk

Mohammad Nabati
Cognitive Telecommunication Research Group, Department of Telecommunications, Faculty of Electrical Engineering, Shahid Beheshti University, G.C., Tehran, Iran.
mo.nabati@mail.sbu.ac.ir

Reza Shahbazian
Cognitive Telecommunication Research Group, Department of Telecommunications, Faculty of Electrical Engineering, Shahid Beheshti University, G.C., Tehran, Iran.
r_shahbazian@sbu.ac.ir

Seyed Ali Ghorashi
Cognitive Telecommunication Research Group, Department of Telecommunications, Faculty of Electrical Engineering, Shahid Beheshti University, G.C., Tehran, Iran.
a_ghorashi@sbu.ac.ir

Vahideh Moghtadaiee
Cyberspace Research Institute, Shahid Beheshti University, G.C. Tehran, Iran.
vmoghtadaiee@gmail.com

ABSTRACT

Smartphones are linked with individuals and are valuable and yet easily available sources for characterizing users' behavior and activities. User's location is among the characteristics of each individual that can be utilized in the provision of location-based services (LBs) in numerous scenarios such as remote health-care and interactive museums. Mobile phone tracking and positioning techniques approximate the position of a mobile phone and thereby its user, by disclosing the actual coordinate of a mobile phone. Considering the advances in positioning techniques, indoor positioning is still a challenging issue, because the coverage of satellite signals is limited in indoor environments. One of the promising solutions for indoor positioning is fingerprinting in which the signals of some known transmitters are measured in several reference points (RPs). This measured data, which is called dataset is stored and used to train a mathematical model that relates the received signal from the transmitters (model input) and the location of that user (the output of the model). Considering all the improvements in indoor positioning, there is still a gap between practical solutions and the optimal solution that provides near theoretical accuracy for positioning. This accuracy directly impacts the level of usability and reliability in corresponding LBSs. In this paper, we develop a smartphone app with the ability to be trained and detect users' location, accurately. We use Gaussian Process Regression (GPR) as a probabilistic method to find the parameters of a non-linear and non-convex indoor positioning model. We collect a dataset of received signals' strength (RSS) in several RPs by using a software which is prepared and installed on an Android smartphone. We also find the accurate 2σ confidence interval in the presented GPR method and evaluate the performance of the proposed method by measured data in a realistic scenario. The measurements confirm that our proposed method outperforms some conventional methods including KNN, SVR and PCA-SVR in terms of accuracy.

[*]Corresponding Author

AppLens, September 09–10, 2019, QEII Centre in London, UK

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/1122445.1122456

KEYWORDS

Positioning, Fingerprint, Gaussian Process Regression, Learning, Crowd Source data.

ACM Reference Format:
Elaheh Homayounvala, Mohammad Nabati, Reza Shahbazian, Seyed Ali Ghorashi, and Vahideh Moghtadaiee. 2019. A Novel Smartphone Application for Indoor Positioning of Users based on Machine Learning. In *Proceedings of AppLens 2019: The Workshop on Mining and*

Learning from Smartphone Apps for Users (AppLens). ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

In recent years, indoor location based services (ILBs) have attracted much attention from government regulations, commercial applications and industry [6, 18]. Positioning can be performed by using different methods including global positioning system (GPS) or global navigation satellite system (GNSS). However, in indoor environments because of limited coverage or non-line-of-sight (NLOS) error, satellite based methods do not provide the acceptable accuracy [6]. Therefore, many positioning algorithms are introduced that normally benefit from ranging information such as the received signal strength (RSS), time of arrival (TOA) and angle of arrival (AOA) techniques. TOA is a ranging technique to extract the distance between a transmitter and receiver by measuring the time of transferred signals [14]. AOA is normally obtained by antenna arrays in which the time-difference of received signals in antennas is used to extract signal's angle information [3].

However, in commercial mobile devices, the space is limited and typically the number of antennas is less than three. Both TOA and AOA methods force huge computational and hardware complexities while do not provide expected accuracy in indoor environments because of practical issues, including fading [8]. On the other hand, almost all portable devices are equipped with energy detector hardware that may be used to extract the ranging information from the received signals. The accuracy of this technique is highly affected by multi-path effect, however it is still the most popular method to measure the distance between transmitter and receiver, especially in crowd source applications [7].

One of the promising solutions for indoor positioning is using fingerprinting [7, 20]. This positioning method is generally divided into two phases: 1- offline (or training) phase and 2-online (or test) phase. In offline phase, first the RSS values of base stations are measured and stored in certain coordinates of environment that are known as reference points (RPs)[1, 9]. The base station (BS) signal could be WiFi [16], Bluetooth [17, 21], ZigBee [10, 22] and even light [5]. The RSS values are known as "fingerprints" of RPs and the data collection method is called "fingerprinting". Among different signals, WiFi has become more popular, as it is available in almost all indoor environments and in all communication devices including smartphones which are the main sources of crowd sourcing.

Dataset generation by gathering the measured data is very important step in positioning algorithms. The dataset is normally collected by crowd source techniques. By training, we aim to find the parameters of the mathematical model which describes the relation between model input and output. In fact, we normally choose a non-linear mathematical description to model the relation between the RSS as the input and user's location as the output of the positioning process. We try to extract all parameters of this mathematical model, according to the gathered dataset. When training is finished, the system is ready to be used by users in that environment.

The procedure of online phase is as follows: when the user of this system receives multiple RSS values from surrounding BSs, for instance WiFi APs, it feeds the values to the application. According to the mathematical model with extracted parameters in offline phase, the location of user is then estimated. Normally, maximum likelihood (ML) based methods are used to estimate the user's location based on the available fingerprints. The logical process of fingerprint based positioning is straight-forward, however, the optimal dataset gathering and improving the accuracy are still open issues that researchers try to address. The positioning accuracy is highly affected by model selection and its parameter estimation.

Some of the most frequently used maximum likelihood based methods are, k-nearest neighbor (KNN) [8], neural networks (NN) [2], deep learning [22] and support vector regression (SVR) [15] (which sometimes is called support vector machine regression). All these mentioned methods are generally called deterministic approaches in which the distribution of output (the coordinates of user's location) is not estimated. Unlike the deterministic approaches, probabilistic methods such as Horus, expectation-maximization, Kullback-Leibler divergence, Gaussian process regression (GPR), Bayesian network and conditional random filed, may be used to estimate the probability distribution of the system output. Therefore, by using probabilistic methods, it is possible to compute the confidence interval of truly localized inputs [6].

Here in this paper, we study the fingerprint based positioning and the goal is to improve the accuracy of indoor positioning by using the smartphones. Considering the noise model, we employ a probabilistic method called Gaussian process regression (GPR) to find the parameters of the non-linear mathematical model between RSS values and the user's location. In the first step, we employ a preprocessing for GPR that standardizes the input data. The model to estimate unknown parameters of the model is non-linear and non-convex. Therefore, we apply a gradient based algorithm to estimate the parameters by using our gathered dataset. The dataset is composed of RSS values and physical address of available access points in several reference points of the test environment. This dataset is collected using our developed smartphone application that could be used in crowd sourcing. We further compare the most common kernels to find the best kernel that maximizes the positioning accuracy.

The rest of this paper is organized as follows: in section 2 we present the related works and in section 3 we explain the system model and conventional GPR. In section 4, we present our proposed method and in section 5, the performance evaluation results using real measurements are presented. Finally, section 6 concludes the paper.

2 RELATED WORKS

Some of the recently proposed methods with higher accuracy, are normally used for target tracking [19]. However, these methods use additional hardwares such as magnetometer, gyroscope and accelerometer beside the RSS that are used to measure direction of magnetic field, angular velocity and 3D linear acceleration, respectively. This sensors are generally called Micro Electro Mechanical Systems (MEMS) [6]. Nowadays, many smartphones are equipped with these sensors and could be used when higher accuracy is mandatory [23]. Most of the recently presented methods only use RSS as fingerprint of RPs. These methods employ maximum likelihood and statistical properties of received signals in the environment and therefore, do not require any additional hardware. Maximum likelihood based methods are implemented in several studies with different accuracies and complexities. In [8], authors present KNN for indoor positioning. In KNN method, choosing K neighbor just based on minimum error, does not provide an acceptable system performance. Therefore, authors in [4], divide the dataset into three groups. In this paper, we also use the same three groups.

In [11], authors utilize path-loss model to calculate RSSs in each RP that is used for system training. Although this approach does not consider the complexity of the environment, it removes the dataset collection procedure in offline phase. However, in high-noise environments, this method does not provide an acceptable accuracy. In [16], authors consider a small dataset. They first consider the coordinates of users as input and RSS as the model output. They use GPR in offline phase to find the relation between RSS and user's location. They finally use KNN method to perform the positioning. Although many has contributed to the topic, but there is still a gap to reach a positioning accuracy near the theoretical one.

3 SYSTEM MODEL

General Notations

Here, we use small letters to present scalars, bold small letters to present vectors and bold capital letters to present matrices. For example: p shows a scalar number or a scalar function. $\mathbf{p}$ shows a vector, p_i or $[\mathbf{p}]_i$ shows the i'th element of $\mathbf{p}$. $\mathbf{P}$ shows a matrix, p_{ij} and $[\mathbf{P}]_{ij}$ shows the element in i'th row and the j'th column of $\mathbf{P}$, $\mathbf{p}_i$ shows i'th row of $\mathbf{P}$ and P_i shows i'th column of $\mathbf{P}$. The (˜) and (ˆ) correspond to train and test (validation), respectively.

Fingerprints

As depicted in figure 1, we measure RSS values received from access points in selected RPs called fingerprints. Reference Points could be chosen in real-world scenarios based on the scenario types and movements of users. As an example reference points could be different points of interests in an interactive museum scenario. In remote health care scenarios for the elderlies, the RPs could be defined close to critical locations in home such as the bed or sitting point in living room. We further use the gathered dataset to train the system model. The user stands in each reference points, using our software that is installed on users' handset, collects the RSS values of surrounding access points. Each access points has a unique MAC ID that makes it distinguishable from others. Our installed software on users' handset samples the values with a predefined and adjustable frequency and that the RSS values are recorded for a specific period, for each access point. Due to the variation of measured RSS values during the time, the measured samples at each time period are averaged out and used as the RSS values for further calculations.

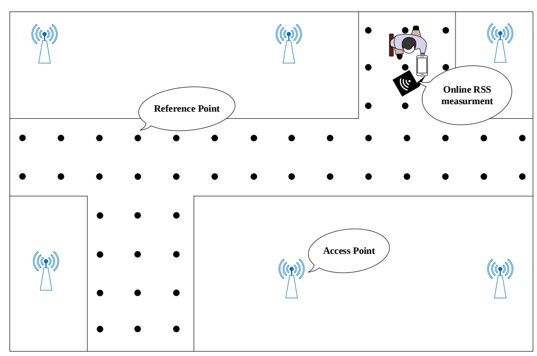

Figure 1: Typical environment of fingerprint positioning. The RPs are shown with circles and access points transmit WiFi signals. The users should collect data from PRs to form the dataset.

The goal is to find a mathematical relation between RSS values and user's location. We assume that the positioning is performed in two dimensional (2D) environment. Finding the parameters for both x and y, is similar. Therefore, without loss of generality, we only present the process for x.

Considering the Gaussian Process Regression, the non-linear relationship between input and output values (RSS values and user's location) are modeled as follows:

$$f_x(.) \sim N(0, \mathbf{C}_x(.,.)) \tag{1}$$

where $\mathbf{C}_x(.,.)$ is the covariance matrix between RSS values. Each element of this matrix shows the similarity between reference points. There are different algorithms in the literature to find the $\mathbf{C}_x(.,.)$ matrix and in most of them a cost function called kernel [12] is used. Some of the most common kernels are square exponential (SE), Linear (Lin) and Noise kernel [12]. In this paper, we have the most common kernels to compare their performance in GPR based fingerprint positioning.

Training System

We use the combination of three kernels to train the system. The combination of SE, Lin and Noise kernel is presented in equation (2). Other available kernels are compared in section 5.

$$\begin{aligned} &\mathbf{C}_x(\mathbf{p}_i,\mathbf{p}_j) = K_{SE}(\mathbf{p}_i,\mathbf{p}_j) + K_{Lin}(\mathbf{p}_i,\mathbf{p}_j) + K_n(\mathbf{p}_i,\mathbf{p}_j) \\ &= \gamma_{SE}^2 \exp\left(-\frac{d^2(\mathbf{p}_i,\mathbf{p}_j)}{l_{SE}^2}\right) + \gamma_{Lin}^2\omega(\mathbf{p}_i,\mathbf{p}_j) + \sigma_n^2\delta_{ij} \\ &\delta_{ij} = \{1 \;\; if \;\; i=j, \;\; 0 \quad otherwise\} \end{aligned} \tag{2}$$

where $\omega(.,.)$ and $d(.,.)$ are inner product and Euclidean distance between Two RSS vectors, respectively. In equation (2), there are four parameters that should be estimated. We use a learning based optimization process to estimated these parameters, $\theta = [\gamma_{SE},\ l_{SE},\ \gamma_{Lin},\ \sigma_n]$.

We assume that the train dataset and RSS values are as follows:

$$\tilde{\mathbf{x}} = [\tilde{x}_1\, \tilde{x}_2 \ldots\ \tilde{x}_{\tilde{L}}]^T \tag{3}$$

$$\tilde{\mathbf{P}} = \begin{bmatrix} \tilde{p}_{11} & \tilde{p}_{12} & \cdots & \tilde{p}_{1M} \\ \tilde{p}_{21} & \tilde{p}_{22} & \cdots & \tilde{p}_{1M} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{p}_{\tilde{L}1} & \tilde{p}_{\tilde{L}2} & \ldots & \tilde{p}_{\tilde{L}M} \end{bmatrix} \tag{4}$$

where M is number of APs, $\tilde{L}$ is the number of RPs, $\tilde{\mathbf{x}}$ shows the coordinates of RPs and $\tilde{\mathbf{P}}$ shows the RSS values of RPs. In Gaussian Process Regression, we assume that the relation between inputs and output is Gaussian as presented in equation (5):

$$\tilde{\mathbf{x}} \sim N(0, \mathbf{C}_{\tilde{x}}) \tag{5}$$

The parameter θ can be estimated as follows:

$$\tilde{\theta} = \arg\max_{\theta} \log(p(\tilde{\mathbf{x}})) = \arg\min_{\theta}(-\log(p(\tilde{\mathbf{x}}))) \tag{6}$$

where

$$\log p(\tilde{\mathbf{x}}) = \frac{1}{2}\log|\mathbf{C}_{\tilde{x}}| + \frac{\tilde{L}}{2}\log(2\pi) + \frac{1}{2}\tilde{\mathbf{x}}^T\mathbf{C}_{\tilde{x}}^{-1}\tilde{\mathbf{x}} \tag{7}$$

optimization problem presented in (6) is non-linear and non-convex. However, some gradient based algorithms such as conjugate gradient could be used to solve the problem.

When the parameter estimation is performed, the model could be used in a smartphone software or other devices to estimate the user's coordinates with the received RSS values (also called test RSS) as follows:

$$\begin{bmatrix} \tilde{\mathbf{x}} \\ \hat{\mathbf{x}} \end{bmatrix} \left[\left(\begin{array}{c} 0 \\ 0 \end{array}\right), \left(\begin{array}{cc} \mathbf{C}_{\tilde{x}} & \mathbf{C}_{\tilde{x},\hat{x}} \\ \mathbf{C}_{\hat{x},\tilde{x}} & \mathbf{C}_{\hat{x}} \end{array}\right)\right] \tag{8}$$

In equation (8), $\mathbf{C}_{\tilde{x}} \in \mathbf{R}^{\tilde{L}\times\tilde{L}}$ is the covariance matrix between the train RSS values, $\mathbf{C}_{\hat{x},\tilde{x}} \in \mathbf{R}^{\hat{L}\times\tilde{L}}$ is the covariance matrix between the test and train RSSs and $\mathbf{C}_{\hat{x}} \in \mathbf{R}^{\hat{L}\times\hat{L}}$ is covariance matrix between test RSS values. $\hat{\mathbf{x}}$ is the new inputs coordinates that we want to estimate. The conditional distribution of equation (8) with respect to $\tilde{\mathbf{x}}$ is as follows:

$$\begin{aligned} &\hat{\mathbf{x}} |\, \tilde{\mathbf{x}} = \ [\mu_{\hat{x}|\tilde{x}}, \mathbf{C}_{\hat{x}|\tilde{x}}] \\ &\mu_{\hat{x}|\tilde{x}} = \mathbf{C}_{\hat{x},\tilde{x}}\, \mathbf{C}_{\tilde{x}}^{-1}\tilde{\mathbf{x}} \\ &\mathbf{C}_{\hat{x}|\tilde{x}} = \mathbf{C}_{\hat{x}} - \mathbf{C}_{\hat{x},\tilde{x}}\, \mathbf{C}_{\tilde{x}}^{-1}(\mathbf{C}_{\hat{x},\tilde{x}})^T \end{aligned} \tag{9}$$

where, $\mu_{\hat{\mathbf{x}}|\tilde{\mathbf{x}}}$ and $\mathbf{C}_{\hat{x}|\tilde{x}}$ are vector of estimated coordinates and estimation of covariance matrix, respectively. Diagonal elements of $\mathbf{C}_{\hat{x}|\tilde{x}}$ represent the variance of estimation coordinates (i.e. variance of $\mu_{\hat{\mathbf{x}}|\tilde{\mathbf{x}}}$)

4 PROPOSED METHOD

Dataset Gathering

The overview of our software is depicted in figure 2. This software is available in APK format, installable on all smartphones with Android operating system. As depicted in figure 2, by clicking on search, the software finds all available access points and records their unique MAC ID. All mentioned information could be saved in a simple text file in the application folder. As can be seen in figure 3, the record tab is used for data gathering configuration. First, we need to load the MAC ID stored file which could be feed to the application by a different user. Then, we need to indicate the sample time, samples number, RP number and iteration. The description to each of these parameters is as follows:

- Sample time: indicates the time taken for sampling the access point's RSS.
- Samples number: indicates the number of samples that are measured in a specific RP.
- RP number: shows the number we assign to each RP. This value is a key to indicate the RPs.
- Iteration: This value is set when the sampling is performed for multiple times, during multiple days and the default value is set to 1.

To collect the dataset, we measure 75 samples (N=75) in 100 seconds at 250 points. Then we randomly divide our

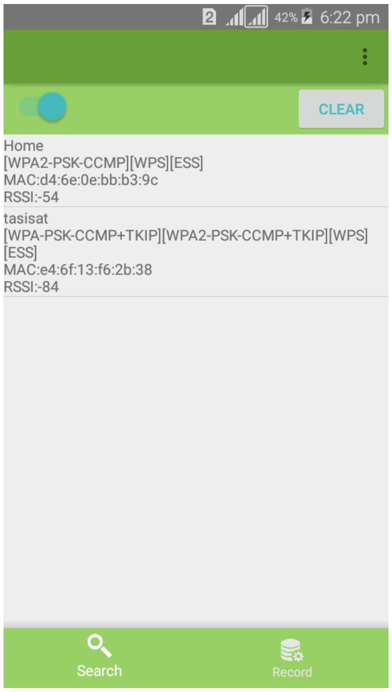

Figure 2: The search tab of our data gathering software. In this tab, the user can select and save the MAC ID of available access points.

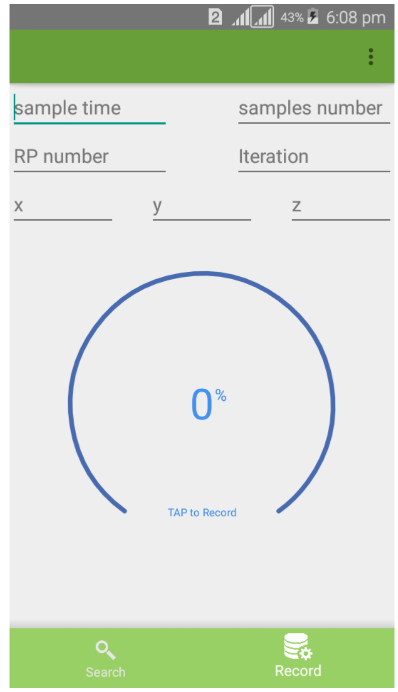

Figure 3: The Record tab of our data gathering software. In this tab, the user sets the variable parameters including sampling duration, number of samples, number of RP and the iteration number.

dataset into three groups: 1-Train (or Reference points), 2-Validation Points and 3-Test Points. The number of RPs should be enough to train the system. There is no predefined categorization for size of each group and it depends on the dataset [4]. In this paper, we use 210 points for training, 20 points for validation and 20 points for test set. We use RPs for training of the system. We also use validation points for two purposes: 1- choosing proper value for σ_n in Noise kernel that directly affects the confidence interval estimation and 2- choosing proper kernel for our presented dataset. Finally, the test points are used to evaluate the performance of our proposed method.

Preprocessing

In this paper, we standardize all inputs and outputs to make the evaluation process more meaningful. By standardization process of b, we calculate $b^{\text{standardize}} = \frac{b-\mu}{\sigma}$, where μ indicates the average value and σ shows the variance of b. In figure 4, we describe this process for both offline and online phases. In offline phase, we compute the mean and variance from each column (or AP) of training dataset separately, and keep the parameter for scaling inputs in online phase. We also compute the mean and variance of all outputs in training dataset and keep them to scale the output results.

Kernel Selection for Noise Estimation

To choose the proper value of σ_n in Noise kernel we swipe the σ_n with 0.005 steps from 0 to 0.3 (this interval depends on the dataset) and compute True Estimate Region Percentage (TERP) as presented in equation (10).

$$\begin{aligned} TERP_x &= \left(\frac{1}{\hat{L}} \sum_{i=1}^{\hat{L}} r_i \right) \times 100 \\ where,\ r_i &= \begin{cases} 1 & \left| \hat{\mathbf{x}}_i - [\mu_{\hat{x}|\tilde{x}}]_i \right| \leq 2\sqrt{[\mathbf{C}_{\hat{x}|\tilde{x}}]_{ii}} \\ 0 & otherwise \end{cases} \end{aligned} \tag{10}$$

After calculation of changing TERP in front of σ_n (in validation dataset) we can estimate a proper value for choosing σ_n in Noise kernel. Choosing σ_n in Noise kernel depends on data. Our proposed method is to keep all value of σ_n for which TERP is between $TERP_d$ and $TERP_u$, then choose the median of those. If data is enough, we can choose $TERP_d = 85$ and $TERP_u = 95$. However, when data is not enough, less values are chosen. The reason is, if we choose a large value for σ_n, the confidence interval will be larger than the real value, such that 100 percent of estimated coordinates will be in confidence interval region. Also this is important that Cramer-Rao-Lower-Bound (CRLB) that is proposed in [11] be less than RMSE. CRLB directly relates to $\mathbf{C}_{\hat{x}|\tilde{x}}$ and the percent of confidence interval must be less or equal to 95

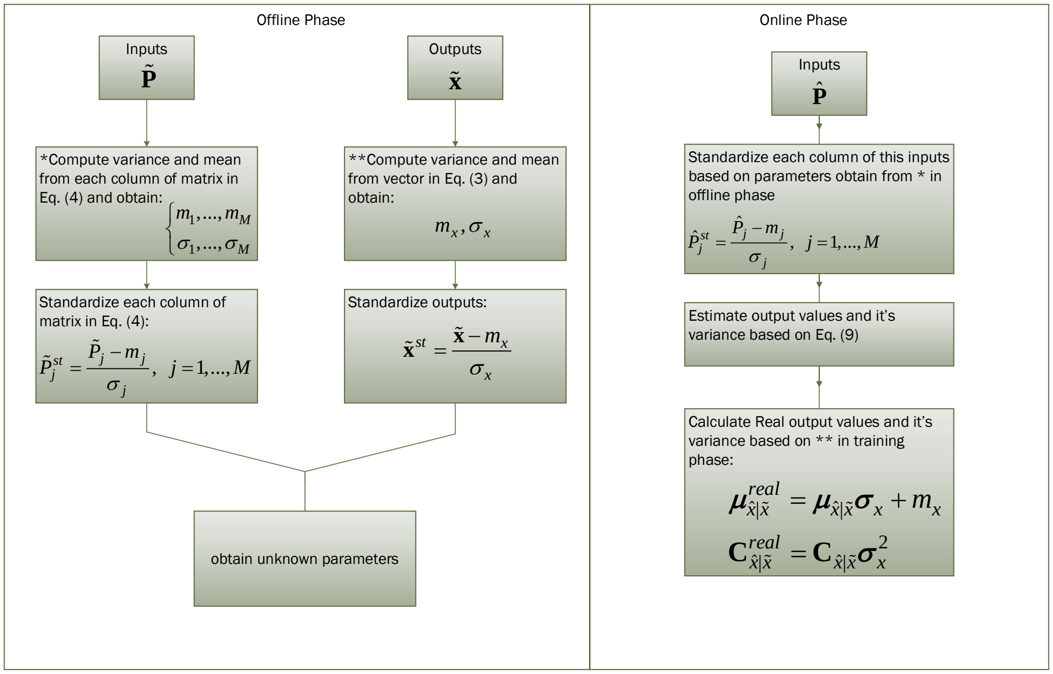

Figure 4: Standardization process of inputs and outputs for both offline and online phases.

percent. This explanation for choosing σ_n can be seen in the following equations:

$$\mathbf{a} = \{[\sigma_n]_i\}_{i=1}^{k} = \{\sigma_n \mid TERP_d \leq TERP_x \leq TERP_u\} \quad (11)$$

$$[\sigma_n]_{opt} = median\,\{\mathbf{a}\} \quad (12)$$

Here, we use Root Mean Squared Error (RMSE) to evaluate the performance of presented method. We use validation set to choose the best kernel where the kernel with minimum RMSE is selected while the noise increases. The RMSE is calculated as follows:

$$RMSE = \sqrt{\frac{1}{\hat{L}}\left(\sum_{i=1}^{\hat{L}}(\hat{x}_i - [\mu_{(\hat{x}|\tilde{x})}]_i)^2 + \sum_{i=1}^{\hat{L}}(\hat{y}_i - [\mu_{(\hat{y}|\tilde{y})}]_i)^2\right)} \quad (13)$$

5 EVALUATION RESULTS

To perform the evaluations, we have used the third floor of electrical engineering department of Shahid Beheshti University and the dataset is collected by using an Android software installed on a smart phone (Samsung Grand-Prime). We have measured the RSS values of 10 WiFi access point in 250 points with 75 samples of each access point for 100 consecutive seconds. As the number of reference points and access points increases, the positioning resolution can be improved. We first present the TERP while the Noise kernel parameter is changing for some kernels. As depicted in figure 5 and figure 6, increasing the Noise kernel parameter, σ_n, increases the TERP for all kernels. Figure 5 and figure 6 present the results for x and y coordinates, respectively.

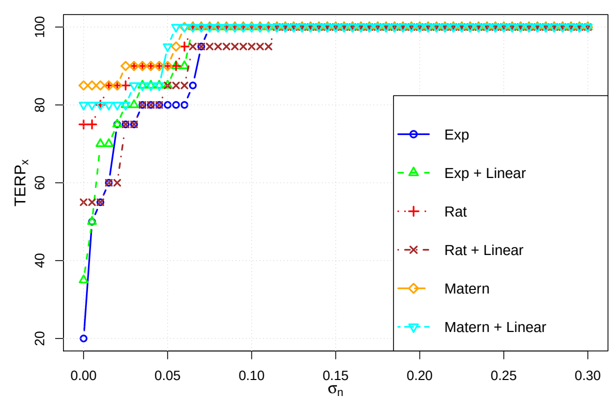

Figure 5: The amount of TERP for x coordinates versus Noise kernel parameter for EXP, Matern, Linear and some combinations of these kernels.

Here, we choose σ_n based on the TERP value when it stays on 75% and 85% interval. We use median value of mentioned TERP interval.

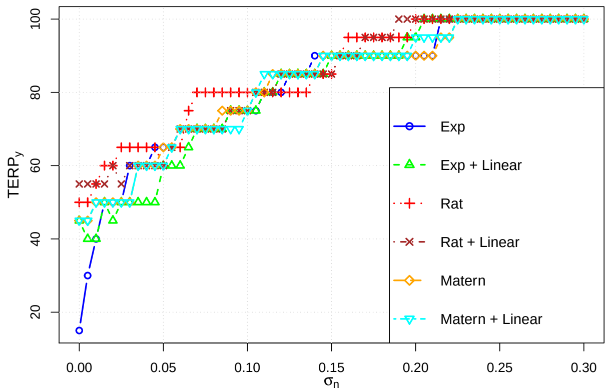

Figure 6: The amount of TERP for y coordinates versus versus Noise kernel parameter for EXP, Matern, Linear and some combination of these kernels.

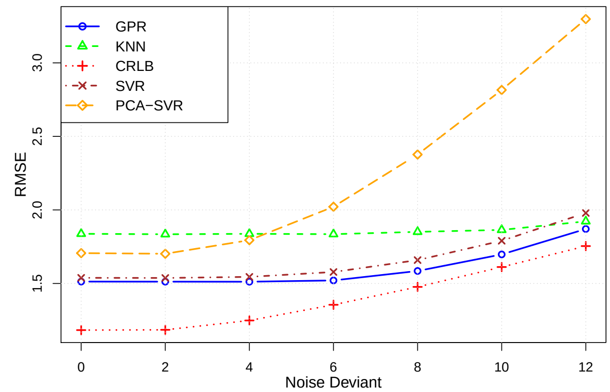

Figure 8: Localization error of proposed method (GPR) in compared with KNN, SVR, PCA-SVR and CRLB in terms of RMSE.

We also evaluate the positioning error for our proposed method with different kernels. In figure 7, the RMSE of positioning versus noise is presented for different types of kernels. As can be seen in this figure, the proposed method gains the minimum error by using the Matern kernel.

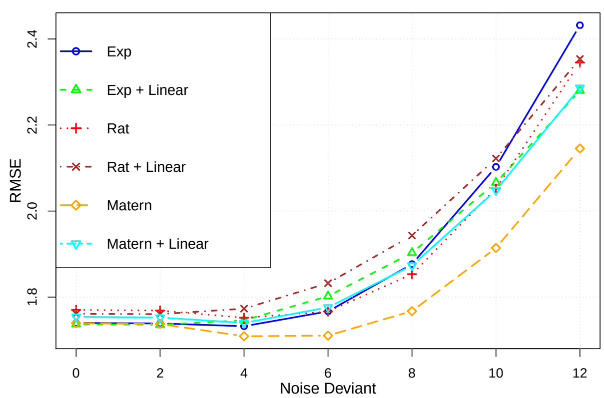

Figure 7: Positioning error of proposed method in terms of RMSE for different types of Noise kernels.

In another evaluation, we use Matern kernel and compare the performance of the proposed GPR based positioning with some of recent and known algorithms including KNN [8], SVR[15], PCA-SVR [13] and theoretical Cramer-Rao lower bound (CRLB) in terms of RMSE. As can be seen in figure 8, the proposed method performs best among others with nearest distance to theoretical bounds. In our final simulation, we evaluate the TERP of evaluated the test points with considered 2σ confidence interval. As depicted in figure 9, at-least 85% of tested points lie on the confidence interval by using the proposed method.

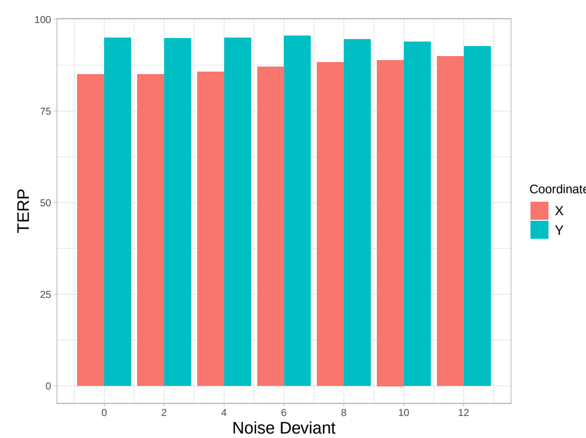

Figure 9: The amount of TERP versus the noise that shows the percentage of correctly estimated locations in 2σ confidence interval.

6 CONCLUSION

In this paper we developed a new smartphone app to detect user's position in an indoor scenario. We studied fingerprint positioning for indoor environments. We briefly explained the ranging techniques and maximum likelihood approaches used for fingerprint positioning. We used a probabilistic method called GPR to localize the users. We first gathered the RSS values of some fixed WiFi access points in some selected points with known coordinates called RP. The procedure has been performed using a mobile application installed on a smart-phone. We proposed a GPR based method to model the relation of input (RSS) and output (user's location) with gathered dataset. This model was non-linear and non-convex. Therefore, gradient based methods were used to extract the system parameters. We evaluated the performance of our proposed method by real measurements in computer simulations. We have used 250 points with 10 APs and 75 samples for each point. We evaluated the proposed method with some of known kernels. We have also compared the performance of the proposed method in terms of RMSE with some of known methods including KNN, SVR, PCA-SVR and theoretical CRLB. The simulation results confirm that the proposed method performs better and near to CRLB compared with other similar methods. Improving accuracy of indoor positioning, the users can be distinguished even in crowded environments without breaking the privacy rules. Therefore, their behavior and activity information can be analyzed based on their location gaining more meaningful insights. The result of this study has the potential to be used in real-world scenarios involving locations-based services such as interactive museums or remote healthcare scenarios.

REFERENCES

[1] Nasim Alikhani, Shima Amirinanloo, Vahideh Moghtadaiee, and Seyed Ali Ghorashi. 2017. Fast fingerprinting based indoor localization by Wi-Fi signals. In *2017 7th International Conference on Computer and Knowledge Engineering (ICCKE)*. IEEE, 241–246.

[2] Huan Dai, Wen-hao Ying, and Jiang Xu. 2016. Multi-layer neural network for received signal strength-based indoor localisation. *IET Communications* 10, 6 (2016), 717–723.

[3] Mohammad Fardad, Seyed Ali Ghorashi, and Reza Shahbazian. 2014. A novel maximum likelihood based estimator for bearing-only target localization. In *2014 22nd Iranian Conference on Electrical Engineering (ICEE)*. IEEE, 1522–1527.

[4] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. 2001. *The elements of statistical learning*. Vol. 1. Springer series in statistics New York.

[5] Xiansheng Guo, Sihua Shao, Nirwan Ansari, and Abdallah Khreishah. 2017. Indoor localization using visible light via fusion of multiple classifiers. *IEEE Photonics Journal* 9, 6 (2017), 1–16.

[6] Suining He and S-H Gary Chan. 2015. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys & Tutorials* 18, 1 (2015), 466–490.

[7] Changju Kan, Guoru Ding, Qihui Wu, and Tao Zhang. 2018. Robust localization with crowd sensors: A data cleansing approach. *Mobile Networks and Applications* 23, 1 (2018), 108–118.

[8] Ali Khalajmehrabadi, Nikolaos Gatsis, and David Akopian. 2017. Modern WLAN fingerprinting indoor positioning methods and deployment challenges. *IEEE Communications Surveys & Tutorials* 19, 3 (2017), 1974–2002.

[9] Zahra Ezzati Khatab, Amirhosein Hajihoseini, and Seyed Ali Ghorashi. 2017. A fingerprint method for indoor localization using autoencoder based deep extreme learning machine. *IEEE sensors letters* 2, 1 (2017), 1–4.

[10] Kevin Merchant, Shauna Revay, George Stantchev, and Bryan Nousain. 2018. Deep learning for RF device fingerprinting in cognitive communication networks. *IEEE Journal of Selected Topics in Signal Processing* 12, 1 (2018), 160–167.

[11] KNR Surya Vara Prasad, Ekram Hossain, and Vijay K Bhargava. 2018. Machine Learning Methods for RSS-Based User Positioning in Distributed Massive MIMO. *IEEE Transactions on Wireless Communications* 17, 12 (2018), 8402–8417.

[12] Carl Edward Rasmussen. 2003. Gaussian processes in machine learning. In *Summer School on Machine Learning*. Springer, 63–71.

[13] Ahmed H Salamah, Mohamed Tamazin, Maha A Sharkas, and Mohamed Khedr. 2016. An enhanced WiFi indoor localization system based on machine learning. In *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 1–8.

[14] Reza Shahbazian and Seyed Ali Ghorashi. 2017. Distributed cooperative target detection and localization in decentralized wireless sensor networks. *The Journal of Supercomputing* 73, 4 (2017), 1715–1732.

[15] Ke Shi, Zhenjie Ma, Rentong Zhang, Wenbiao Hu, and Hongsheng Chen. 2015. Support vector regression based indoor location in IEEE 802.11 environments. *Mobile Information Systems* 2015 (2015).

[16] Wei Sun, Min Xue, Hongshan Yu, Hongwei Tang, and Anping Lin. 2018. Augmentation of Fingerprints for Indoor WiFi Localization Based on Gaussian Process Regression. *IEEE Transactions on Vehicular Technology* 67, 11 (2018), 10896–10905.

[17] Chao Xiao, Daiqin Yang, Zhenzhong Chen, and Guang Tan. 2017. 3-D BLE indoor localization based on denoising autoencoder. *IEEE Access* 5 (2017), 12751–12760.

[18] Jun Yan, Lin Zhao, Jian Tang, Yuwei Chen, Ruizhi Chen, and Liang Chen. 2017. Hybrid kernel based machine learning using received signal strength measurements for indoor localization. *IEEE Transactions on Vehicular Technology* 67, 3 (2017), 2824–2829.

[19] Ali Yassin, Youssef Nasser, Mariette Awad, Ahmed Al-Dubai, Ran Liu, Chau Yuen, Ronald Raulefs, and Elias Aboutanios. 2016. Recent advances in indoor localization: A survey on theoretical approaches and applications. *IEEE Communications Surveys & Tutorials* 19, 2 (2016), 1327–1346.

[20] Yuxin Zhao, Chao Liu, Lyudmila Mihaylova, and Fredrik Gunnarsson. 2018. Gaussian Processes for RSS Fingerprints Construction in Indoor Localization. https://doi.org/10.23919/ICIF.2018.8455842

[21] Yuxin Zhao, Feng Yin, Fredrik Gunnarsson, Mehdi Amirijoo, and Gustaf Hendeby. 2016. Gaussian Process for Propagation modeling and Proximity Reports Based Indoor Positioning. In *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*. IEEE, 1–5.

[22] Kan Zheng, Huijian Wang, Hang Li, Wei Xiang, Lei Lei, Jian Qiao, and Xuemin Sherman Shen. 2016. Energy-efficient localization and tracking of mobile devices in wireless sensor networks. *IEEE Transactions on Vehicular Technology* 66, 3 (2016), 2714–2726.

[23] Han Zou, Zhenghua Chen, Hao Jiang, Lihua Xie, and Costas Spanos. 2017. Accurate indoor localization and tracking using mobile phone inertial sensors, WiFi and iBeacon. In *2017 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*. IEEE, 1–4.