

# Reliable indoor location prediction using conformal prediction

Khuong An Nguyen · Zhiyuan Luo

Published online: 27 October 2013  
© Springer Science+Business Media Dordrecht 2013

**Abstract** Indoor localisation is the state-of-the-art to identify and observe a moving human or an object inside a building. However, because of the harsh indoor conditions, current indoor localisation systems remain either too expensive or not accurate enough. In this paper, we tackle the latter issue in a different direction, with a new conformal prediction algorithm to enhance the accuracy of the prediction. We handle the common indoor signal attenuation issue, which introduces errors into the training database, with a reliability measurement for our prediction. We show why our approach performs better than other solutions through empirical studies with two testbeds. To the best of our knowledge, we are the first to apply conformal prediction for the localisation purpose in general, and for the indoor localisation in particular.

**Keywords** Conformal prediction · Fingerprinting · Indoor localisation · Bluetooth tracking

## 1 Introduction

There are many new and exciting context-aware applications such as pedestrian navigation, and public safety. All these applications depend on indoor localisation. The purpose of indoor localisation is to identify and observe a user inside a building. Global Positioning System (GPS) has long been an optimal solution for outdoor localisation, yet the indoor counterpart remains an open research problem, because of the harsh and complex indoor building structure. Current indoor localisation systems remained either too expensive or not accurate enough [36, 37]. In this

---

K. A. Nguyen (✉) · Z. Luo  
Department of Computer Science, Royal Holloway, University of London  
Egham, Surrey TW20 0EX, UK  
e-mail: me@khuong.vn

Z. Luo  
e-mail: zhiyuan@cs.rhul.ac.uk

paper, we tackle the latter issue in a different direction, with a new confidence machine algorithm to enhance the accuracy of the real-time prediction. Our approach uses a pre-constructed database mapping the physical co-ordinates to the recorded wireless signal strength at that location. The problem is how to match a real-time signal strength pattern with the correct location with minimum error. We use the Conformal Prediction algorithm (CP) to achieve a better location accuracy, and to provide a reliability measurement for our prediction. We also designed a new underlying algorithm based on the Weighted K-nearest neighbour. Empirical studies showed that our algorithm alone performed 5–10 % better than other systems. To the best of our knowledge, we are the first to apply CP for the localisation purpose in general, and for the indoor localisation context in particular.

The remainder of the paper is organised as follows. In the next section, we introduce the indoor localisation problem, and the fingerprinting solution. We then introduce the ideas of conformal prediction, and how to apply it to solve the indoor localisation problem in Section 3. Experiments of conformal predictions on two testbeds and the performance of our new nonconformity measure are evaluated in Section 4. Conclusions and possible future work are discussed in Section 5.

## 2 The indoor localisation problem

A user can be coarsely identified at room-level or precisely localised at sub-room level. The finest-grained systems offer up to 3 cm accuracy, 95 % of the time, using intensive hardware implementation [36, 37]. However, they demand massive investment to install and maintain. Thus, recent approaches concentrate mainly on software solutions to reduce the cost. This section overviews the current states of the indoor localisation, emphasising their benefits and shortcomings, as a background to be enhanced in the next section.

### 2.1 Coarse-grained tracking

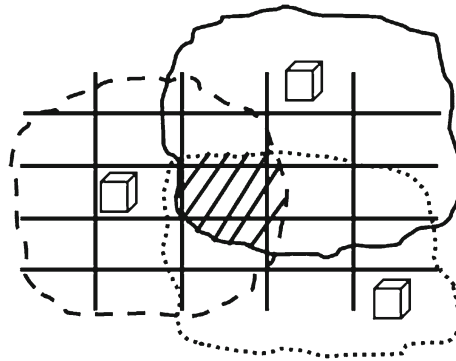
The most basic indoor localisation comes from the proximity-based tracking idea. Whenever a user and a base station can communicate, the user's location is determined as the location of the station. This method is further enhanced by dividing the tracking space into grids. The stations are strategically placed in such a way that each grid block is overlapped by the signal from as many different stations as possible (Fig. 1). Thus, instead of coarsely predicting the user's location to be somewhere within the station's broadcasting range, the accuracy is improved by interpreting the user's location to be the overlapped portion of the stations the user sees. However, many stations must be deployed to have a good signal coverage.

Overall, coarse-grained tracking idea is great for localising user at the room-level resolution. The next section discusses how to identify the user at much finer-grained sub-room resolution.

### 2.2 Location fingerprinting

Coarse-grained tracking is improved further by analysing the wireless signal strength, based on the fact that the radio signal attenuates and gets weaker as it travels in

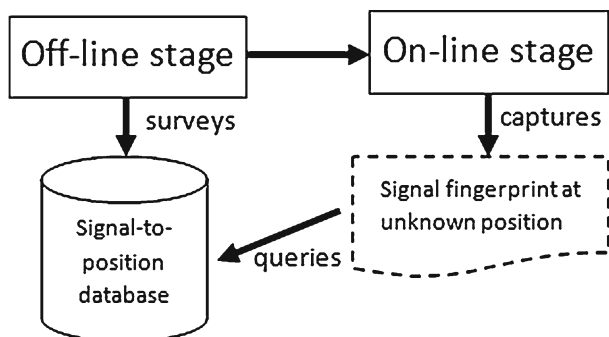
**Fig. 1** Overlapping signal of three stations



the air. There are two popular measurements to roughly represent the distance between a user and a station: the Received Signal Strength Indication (RSSI) and the Link Quality (LQ). However, two distinct locations might not have a linear relationship in terms of RSSI/LQ and the distance between them. This phenomenon is caused by the signal being blocked by the indoor objects, known as the multi-path issue [27]. An elegant solution—the ‘Location Fingerprinting’ method has been widely adopted for its simplicity and efficiency [2]. This method utilises the built-in wireless signal of the building, such as the Wireless LAN or Bluetooth, to survey a signal-to-position mapping database beforehand, which is known as the off-line stage. Having captured the signal strength at each individual location, the system applies the pattern-matching algorithm of the real-time signal and the database record to estimate the most probable position during the on-line stage. The Location Fingerprinting method is depicted in Fig. 2.

In comparison to the proximity-based tracking, this solution offers much higher fine-grained tracking even with a few stations. The overall accuracy of the system relies on two criteria. First, if the mapping database is poorly constructed during the off-line stage, the correct estimation will degrade. Second, the prediction algorithm chosen in the on-line stage will determine how good the estimated position is.

**Fig. 2** Location fingerprinting method



Our previous works were concerned with the first criterion, by showing the strategies to set up the transceivers and to collect the wireless signal more efficiently in different situations [26, 27]. This paper is more concerned with the second criterion, by discussing and evaluating machine learning algorithms which can cope with the uncertainty and the noises in the mapping database.

The main challenge is that RSSI measurement at the same location can vary from time to time, and distinct locations might have a similar RSSI combination due to human movements, humidity, furniture re-arrangement, as well as the multi-path fading of the indoor environment. Many algorithms have been applied to the problem, such as the K-nearest neighbours (K-NN) [2], Bayesian approach [41], Neural network [13], yet most of them assumed the training database is completely trustworthy. Our algorithms go a further step to provide an extra confidence information for our prediction location. For the indoor localisation problem, this information is useful because the mapping database is noisy. First, the overall accuracy is improved by combining the whole prediction region. Second, a current prediction with high confidence level can be used to improve a previous prediction with low confidence level in the on-line setting.

### 2.3 Modelling the indoor localisation problem

Without any loss of generality, let us formally model the Location Fingerprinting method. A single location  $L$  is modelled in a 3-dimensional space  $L = (d^x_L, d^y_L, d^z_L)$ . The signal strength RSSI between the user at a location  $L$  and all  $N$  stations is modelled as an  $n$ -tuple  $RSSI_L = (s_1, s_2, \dots, s_n)$ , where  $s_i$  is the signal strength received from the station  $i$  ( $1 \leq i \leq N$ ).

To reflect the indoor signal attenuation at each location, the RSSI is captured many times, which results in many duplicated examples in the Location label set  $Y$ . Two distinct examples in this Location set  $Y$  might have a similar signal strength reading, also because of the indoor signal attenuation. A database  $B$  organises the surveyed locations in terms of RSSI measurement and the corresponding physical co-ordinate as shown in Table 1.

The indoor localisation task is as follows: given an RSSI tuple  $RSSI_{\text{unknown}} = (s_1, s_2, \dots, s_n)$  of an unknown location inside the tracking zone, the system compares this tuple with all recorded samples in the database  $B$  to estimate a closest match,

**Table 1** Off-line database snapshot

3-D location ( $Y$ )	RSSI measurement ( $X$ )
...	.....
(51, 136, 70)	(-57, -41, -62, -59, -86, -81)
(51, 136, 70)	(-57, -42, -60, -58, -87, -81)
(51, 136, 70)	(-56, -41, -60, -59, -87, -81)
(51, 146, 70)	(-59, -36, -65, -58, -82, -75)
(51, 156, 70)	(-56, -38, -69, -58, -84, -73)
(51, 156, 70)	(-54, -35, -69, -57, -84, -72)
(51, 166, 70)	(-63, -32, -73, -57, -87, -68)
(51, 176, 70)	(-67, -33, -68, -60, -91, -69)
...	.....

which will predict the co-ordinate  $(d^x, d^y, d^z)$  of the unknown location. Table 2 summarises the main notations used in the paper.

## 2.4 Related works

There have been numerous attempts to solve the indoor tracking problem within the past decade with extensive hardware implementations such as the Active Badge system [36, 37], the Cricket system [29] which measure the time-of-flight from a tag to the beacons using ultrasonic sensors. Despite an extreme accuracy up to 3 cm, 95 % of the time, these systems are very expensive and very hard to maintain and deploy. Other hardware-based solutions include the foot-mounted sensor [31, 39], indoor mobile phone tracking [20, 23], which uses the compass and the gyroscope sensors to detect the direction the user is facing.

Recent approaches have solely focused on the software-side to improve the location accuracy based on the wireless signals such as Bluetooth [1, 3, 7, 9, 14, 15, 17, 18, 24, 25, 28, 42], Wireless LAN [4, 6, 16, 22, 30, 34, 40], Infrared [19], FM [8]. There have been attempts to localise the indoor users based on wireless signal propagation models [21, 38]. However, it was difficult to fit a model due to the indoor attenuation problem. The Fingerprinting solution has been most widely employed recently because it uses the attenuation problem as its advantage to build a signal mapping database. Selecting the wireless signal to build the database for the Fingerprinting method has been widely discussed. Many researchers employed the Wireless LAN signal for its stability over long range distance. Others used the Bluetooth signal because it uses less power, and is easy to install and configure. In our previous work, we also showed that the Bluetooth signal is more stable than the Wireless LAN over short range distance in a crowded environment. Many different approaches have been attempted to predict an unknown location using the Fingerprinting database, such as the K-nearest neighbours [2], Bayesian approach [41], or Neural network [13]. We realise that in such noisy environment, the prediction is not accurate, regardless of the implemented algorithm. Thus, we use the

**Table 2** Notations summary

Symbol	Meaning
$L$	A single location $(d^x_L, d^y_L, d^z_L)$
$(d^x_L, d^y_L, d^z_L)$	3-dimensional physical location
$RSSI_L$	The signal strength sequence at location $L$
$N$	Number of stations (APs)
$s_i$	A Single signal strength reading
$B$	A database
$z_i$	An example (an entry) in the database—a tuple $(RSSI, L)$
$K$	Number of nearest neighbours
$X$	The object set—the collection of all signal strengths
$Y$	The label set—the collection of all locations in 3-D
$\alpha$	Nonconformity measurement
$\varepsilon$	Significant level

CP to provide a confidence parameter for our prediction, as well as improving our prediction accuracy with our new nonconformity measurements.

### 3 Predicting an indoor location with confidence

Having discussed the uncertainty problem of the indoor localisation training database, this chapter explains the idea of confidence prediction we used to solve such issue. Based on the Conformal Prediction (CP) method, we designed a new underlying algorithm to predict an indoor location. Particularly, we look at the problem from both classification and regression perspectives, based on the size of the position set (the label set) and the signal strength set (the object set). The performance of the implementation will be evaluated in the next section.

#### 3.1 Conformal prediction

Conformal Prediction is a relative new machine learning algorithm, which uses experiences in the past to confidently and precisely predict the outcome of a new sample [33, 35]. However, what differentiates CP from other similar machine learning algorithms is the ability to customise the confidence level parameter, thus, either tightens the prediction region or looses it up to allow more samples at the cost of a lower confidence. It has been mathematically proved that the prediction region generated by CP is valid in online setting [35]. In other words, for a confidence level at 95 %, the correct estimated position is expected to be included in the prediction region at least 95 % of the time. Achieving the prediction result in such a powerful sense, however, CP demands a relatively weak assumption that the training database and the new sample to be classified are generated from the same distribution independently. CP has been successfully used in many applications such as medical diagnosis and network traffic classification [5, 10].

The CP algorithm can be mathematically generalised as followed. Given a training database  $B = (z_1, z_2, \dots, z_l)$ , with  $z_i = (x_i, y_i)$  where  $x_i \in X$  is the set of attributes and  $y_i \in Y$  is the set of labels. The only requirement of CP is that  $z_i$  is generally from the same distribution. Given a new example  $x_{l+1}$  with an unknown label, CP will predict a label  $\hat{y}_{l+1}$  with a corresponding confidence level  $r$  for this new example.

The most important property of CP is the ‘nonconformity measure’  $\alpha_i$ , which is a real-valued function  $A(B, z_i) = (z_1, \dots, z_{i-1}, z_{i+1}, \dots, z_l, z_i)$  measuring how different a sample  $z_i$  is to the training database  $B$ . Ideally, when classifying a new sample  $z_i$ , we would prefer this sample to be completely different from all training samples in  $B$ . Therefore, the sequence’s randomness can be maintained. This is also the core idea of CP, which can be seen as a test of randomness. Whenever a new sample  $x_{l+1}$  needs to be classified, we exhaustively test every label  $y$  recorded in the training data, and CP tells us whether to accept that label or not. CP uses existing machine learning algorithms such as Nearest neighbours, SVM and Neural network as the underling algorithms to compute a ‘nonconformity measure’. This measurement, however, does not give much information. Thus, we must compare  $\alpha_i$  with all other nonconformity measurements as follows.

$$p(y) = \frac{\#\{i = 1, \dots, l + 1 : \alpha_i \geq \alpha_{l+1}\}}{l + 1}. \quad (1)$$

The label  $y$  is added into the prediction region if  $p(y) > r$ , with  $r$  is the confidence parameter in advance. Regardless of the chosen nonconformity measure, the set of locations predicted by CP is always valid in the online setting. However, the efficiency, in other words, the tightness of the prediction region is affected. Ideally, it is more preferred to have as few samples in the prediction region as possible, without sacrificing the confidence level too much. There does exist a ‘universal conformal predictor’ which can be used for any problems [35], however each problem requires a customised nonconformity measure to fit the purpose. A more detailed treatment of the CP can be found in the book [35]. In the next parts, we apply CP to solve the indoor localisation problem with two approaches: classification and regression.

### 3.2 Solving the indoor localisation as a classification problem

Giving a training database  $B$  mapping physical location co-ordinate to correct Bluetooth signal pattern, and a signal fingerprint at an unknown location, CP precisely predicts a set of locations in the database, which likely matches this new signal fingerprint. This can be formulated as a classification problem, because we divide the labels into grid points, and the label set is finite. The measured signal strengths at these grid points are regarded as the object set  $\mathbf{X}$ , and the physical locations are regarded as the label set  $\mathbf{Y}$ . We will apply CP using both the old examples—the training database  $B = (z_1, z_2, \dots, z_l)$ , and the signal fingerprint of the unknown location (as a new object of  $z_{l+1}$ ). Each example  $z_i$  is a combination of the signal strength  $RSSI_i = (s_1^i, s_2^i, \dots, s_n^i)$  and the co-ordinate  $L_i = (d_i^x, d_i^y, d_i^z)$ . A prediction region of  $K$  examples is  $R^e(L_1, L_2, \dots, L_K) \subset \mathbf{Y}$ .

First, CP demands that the training database and the new real-time sample must be generated from the same distribution. This assumption holds because all signals are generated by the same set of base stations independently. Second, to evaluate the difference amongst the samples, the ‘Weighted K-nearest neighbours’ (W-KNN) [12] is chosen as the underlying algorithm to compute the ‘nonconformity score’  $\alpha$ . The reason we opted for the ‘Weighted version’ of the traditional K-nearest neighbours algorithm is because of the multi-path problem of the indoor signals, which cause many locations further apart to have a similar combination of signal strength. If we treat each of the  $K$  neighbours equally, the final estimated position is shifted towards the wrong neighbours further away. By considering the ‘weight’, corresponding to the inverse ‘signal distance’ between each neighbour and the unknown position, the closer neighbours are prioritised over further away ones. The outline of the Conformal Prediction algorithm with W-KNN is presented in Algorithm 1.

Given a training database  $B$ , and a new example with a known signal strength pattern but unknown co-ordinate, CP assumes the correct position to be each of the recorded locations in  $\mathbf{Y}$ . First, we use the Euclidean approach to calculate the “distance” between two signal strength sequences  $RSSI_P$  and  $RSSI_Q$ .

$$dist(RSSI_P, RSSI_Q) = \sqrt{(s_1^P - s_1^Q)^2 + \dots + (s_n^P - s_n^Q)^2}. \tag{2}$$

Using (1), we can find  $K$  entries  $(z_1, \dots, z_K)$  in the database  $B$  which have the smallest distances  $dist(RSSI_i, RSSI_U)$  to the unknown location  $U$ , as well as having the same co-ordinate label  $L_i = (d_i^x, d_i^y, d_i^z)$  with  $U$  ( $1 \leq i \leq K$ ). Then, a weighted

**Algorithm 1** Conformal predictor with W-KNN

**Input:** Training database  $B = \{z_1, \dots, z_l\}$ , significance level  $\varepsilon$ , new example  $z_{l+1}$  with known RSSI,  $N$  is the number of stations.

**Output:** Prediction region  $R$ .

**function** DIST\_RSSI( $P, Q$ )

**for**  $i = 1 \rightarrow N$  **do**

$distance = distance + (s_p^i - s_q^i)^2$

**end for**

**return**  $distance$

**end function**

**function** WEIGHTED\_LOCATION( $K_{set}, z$ )

**for**  $i = 1 \rightarrow K$  **do**

$weight1 = 1/dist\_RSSI(RSSI_i, RSSI_z) * K_{set}(d_x^i)$

$weight2 = 1/dist\_RSSI(RSSI_i, RSSI_z)$

**end for**

**return**  $weight1/weight2$

**end function**

**function** NONCONFORM( $B, z$ )

**for**  $i = 1 \rightarrow N$  **do**

**if** ( $L_i = L_z$ ) &  $dist\_RSSI(B_i, z)$  is the smallest **then**

$K_{same} = K_{same} + z_i$

**end if**

**end for**

**for**  $i = 1 \rightarrow N$  **do**

**if** ( $L_i \neq L_z$ ) &  $dist\_RSSI(B_i, z)$  is the smallest **then**

$K_{diff} = K_{diff} + z_i$

**end if**

**end for**

$e_{same} = weighted\_location(K_{same}, z)$

$e_{diff} = weighted\_location(K_{diff}, z)$

**return**  $sqrt(e_{same} - e_{diff})$

**end function**

**for**  $y \in Y$  **do**

$z_{l+1} = (RSSI_u, y)$  and include it in  $B$

**for**  $z_j$  in  $B$  **do**

$\alpha_j = nonconform(B, z_j)$

**end for**

$p(y) = \frac{\#\{j=1, \dots, l+1: \alpha_j \geq \alpha_{l+1}\}}{l+1}$

**end for**

Predictive set  $R = \{y : p(y) > \varepsilon\}$



average location is calculated from these  $K$  entries, giving an estimated position  $e_{\text{same}} = (d_s^x, d_s^y, d_s^z)$ . The below equation is repeated for each co-ordinate of  $e_{\text{same}}$ , where  $\epsilon$  is a very small constant to prevent division by zero.

$$d_s^x = \frac{\sum_{i=1}^K \frac{1}{\text{dist}(RSSI_i, RSSI_u) + \epsilon} d_i^x}{\sum_{i=1}^K \frac{1}{\text{dist}(RSSI_i, RSSI_u) + \epsilon}}. \tag{3}$$

Similarly, using (1), we find another  $K$  entries  $(z_1, \dots, z_K)$  in the database  $B$  which have smallest distances  $\text{dist}(RSSI_i, RSSI_u)$  to the unknown location  $U$ , but having different co-ordinate label  $L_i = (d_i^x, d_i^y, d_i^z)$  to  $U$  ( $1 \leq i \leq K$ ). Then, another weighted average location is calculated from these  $K$  entries, giving an estimated position  $e_{\text{diff}} = (d_d^x, d_d^y, d_d^z)$ .

Our nonconformity measure is the normal Euclidean distance of the two physical 3-Dimensional locations  $e_{\text{same}}$  and  $e_{\text{diff}}$

$$\alpha = \text{dist}(e_{\text{same}}, e_{\text{diff}}) = \sqrt{(d_s^x - d_d^x)^2 + (d_s^y - d_d^y)^2 + (d_s^z - d_d^z)^2}. \tag{4}$$

Using (3), we calculate the nonconformity score  $\alpha_i$  for every entry in the database  $B$ . Then, we count the number of  $\alpha_i$  which is larger than the nonconformity score  $\alpha_u$  of the new sample location  $U$ , and divide the total number of entries in the database  $B$  to have the p-value number. Given a significance level  $\epsilon$  beforehand (such as  $\epsilon = 0.05$ ), the current assumed co-ordinate label is accepted as the label for the new sample, if and only if the p-value  $> \epsilon$ . All accepted locations form a prediction region, which guarantees to contain the correct position 95 % of the time (when  $\epsilon = 0.05$ ). The size of this prediction region varies significantly, depending on the confidence level, which we will evaluate in the next section.

### 3.3 Solving the indoor localisation as a regression problem

The classification approach becomes impractical when the label set is huge. In our case, when a large tracking zone needs to be covered, and the resolution of the mapping database is small, the indoor localisation can be better seen as a regression problem. Another benefit of the regression approach is the good predicting speed for any sample. We consider three Regression algorithms in the paper—Ridge Regression, Kernel Ridge Regression, and Ridge Regression Conformal Prediction [35].

Ridge regression was proposed in 1960s by Hoerl [11] and is one of the most standard regression algorithms in machine learning. The idea of ridge regression is to fit a linear model into the dataset  $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_l, y_l)$ , and use that model to predict the output  $\hat{y}_{\text{new}}$  of a new sample  $\mathbf{x}_{\text{new}}$ . In the indoor localisation setting,  $\mathbf{x}_i$  represents the signal strength received from the stations and  $y_i$  is the corresponding location. Ridge regression finds the model with the weights  $\mathbf{w}$  which minimise the following expression:  $\sum_{i=1}^l (y_i - \mathbf{w} \cdot \mathbf{x}_i)^2 + \lambda \|\mathbf{w}\|^2 \rightarrow \min$ , where  $\lambda$  is a ridge factor and  $\lambda \geq 0$ . The optimal weights  $\mathbf{w}^*$  can be found by solving the equation.

$$\mathbf{w}^* = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{Y}. \tag{5}$$

The prediction for the new example  $\mathbf{x}_{\text{new}}$  is  $\hat{y}_{\text{new}} = \mathbf{w}^* \cdot \mathbf{x}_{\text{new}}$ . For the indoor localisation problem, we will predict each dimension separately with a different linear model, based on the wireless signal strength. The predicted position  $(\hat{y}_x, \hat{y}_y)$  in 2-Dimensional space is calculated as followed

$$\hat{y}_x = \mathbf{w}_x^* \cdot \mathbf{x}_{\text{new}}. \tag{6}$$

$$\hat{y}_y = \mathbf{w}_y^* \cdot \mathbf{x}_{\text{new}}. \tag{7}$$

The so-called ‘‘kernel trick’’ can be used to map the training data into a higher dimension, so the linear model can easily be fitted [32, 35]. With kernel ridge regression, the new predicted position for x-dimension is calculated as

$$\hat{y}_x = \mathbf{Y}(K + \lambda \mathbf{I})^{-1} \kappa(x), \tag{8}$$

where  $K$  is the kernel matrix and  $\kappa(\mathbf{x}) = K(\mathbf{x}_i, \mathbf{x}_{\text{new}})$ . An example of the popular kernels is radial basis function (RBF) as  $K(\mathbf{u}, \mathbf{v}) = \exp(-\gamma \|\mathbf{u} - \mathbf{v}\|^2)$ ,  $\gamma > 0$  [32].

Ridge regression can be used as the underlying algorithm for conformal prediction. For ridge regression, the nonconformity measure  $\alpha_i$  is normally defined as [35]

$$\alpha_i = |y_i - \hat{y}_i|, \tag{9}$$

where  $\hat{y}_i$  is the prediction of ridge regression for  $\mathbf{x}_i$  based on the examples  $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_l, y_l), (\mathbf{x}_{l+1}, \tilde{y})\}$  and  $\tilde{y}$  is the assumed label for the new example  $\mathbf{x}_{l+1}$ .

The nonconformity measure  $\alpha_i$  of each example can be expressed as a piecewise-linear function of  $\tilde{y}$  [35]

$$\alpha_i = \alpha_i(\tilde{y}) = |a_i + b_i \tilde{y}|, \tag{10}$$

where  $a_i$  and  $b_i$  are the corresponding value of vectors  $A = (y_1, \dots, y_l, 0)' - \mathbf{X}[(\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T (y_1, \dots, y_l, 0)']$  and  $B = (0, \dots, 0, 1)' - \mathbf{X}[(\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T (0, \dots, 0, 1)']$ , respectively.

Instead of having to consider the  $p$ -value of every possible  $\hat{y}$ , we can only compute the set of points  $\tilde{y}$  on the real line that have a  $p$ -value greater than the defined significance level  $\varepsilon$ . It is clear that the  $p$ -value can only change at points where  $\alpha_i(\tilde{y}) - \alpha_{l+1}(\tilde{y})$  changes sign for some  $i = 1, \dots, l$ . An efficient prediction algorithm can be found in [35].

Ridge regression conformal prediction can produce a range (an interval) for each dimension. For a 2-dimensional tracking space, the prediction region is therefore a rectangle, and the predicted position can be considered as the centre of this rectangle. We implemented and applied the ridge regression conformal prediction algorithm to our testbeds.

#### 4 Performance evaluation

Having shown how to solve the Indoor Localisation problem with the conformal prediction, this section evaluates the performance of the algorithms, based on real-life mapping databases on two testbeds using the Bluetooth stations. The enhancement is highlighted by comparing to other state of the art algorithms such as the Bayesian approach [41] and the K-nearest neighbours [2] performed on the same databases.

#### 4.1 Testbed 1

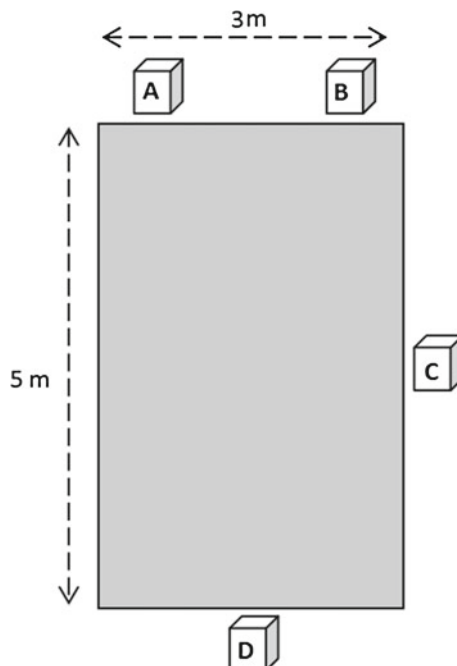
The first testing environment was deployed on the second floor in the Computer Lab, University of Cambridge where we used the Bat system as a reference to provide accurate 3-Dimensional location data [37]. This testbed was, however, limited to just a single room of 15 square metres ( $5\text{ m} \times 3\text{ m}$ ) (Fig. 3). All locations within the tracking zone can be seen by all four Bluetooth stations (A–D). For this small area, four Bluetooth stations would be sufficient to provide distinguishable wireless signal strength at each individual location [26, 27].

We collected the Bluetooth signals at over 200 spots in this area with a 10 cm resolution. At each location, we changed the receiver's orientation to eight different directions corresponding to the North, Northeast, East, Southeast, South, Southwest, West and Northwest. At each orientation, the receiver collected no less than 40 Bluetooth readings, which gives a total of over 64,000 samples for the training database. We collected another 1,000 samples at random locations for the test set.

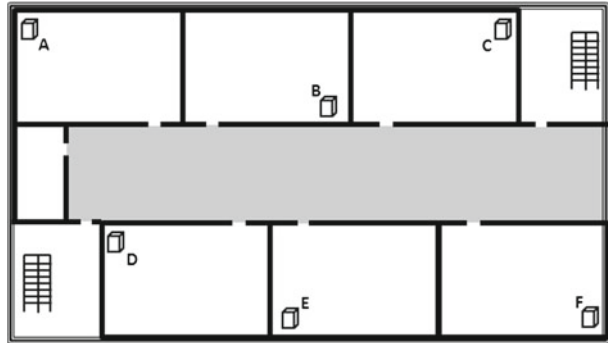
#### 4.2 Testbed 2

The second testbed aimed to deploy the system in a large environment, where unlike Testbed 1, each station alone cannot cover the whole tracking zone. A building's floor of 170 square metres ( $17\text{ m} \times 10\text{ m}$ ) indicated by the gray area was used to deploy the system. There are six base stations (A–F) installed in a tracking zone (Fig. 4).

**Fig. 3** A single 15 square metres room (Testbed 1)



**Fig. 4** A single floor with six rooms (Testbed 2)



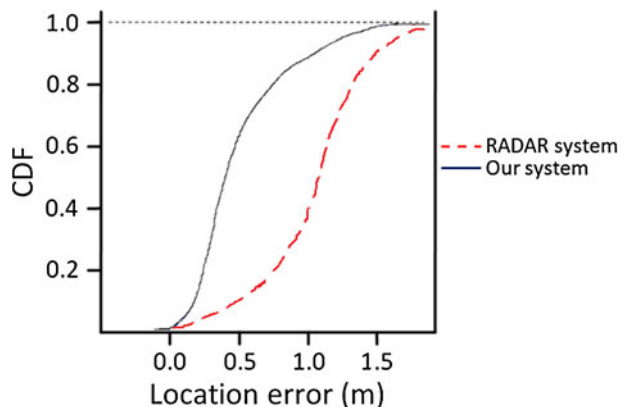
We collected the Bluetooth signal along the offices' corridor at over 150 spots with a larger 30 cm resolution. This is a more realistic scenario. At each spot, we also changed the receiver's orientation to eight different directions, as similar to the first testbed, which gives a total of over 48,000 samples for the second training database. We collected another 500 samples at random locations for the test set.

#### 4.3 Overall prediction accuracy

For both datasets, our algorithms based on W-KNN and ridge regression achieved fewer than 1.5 m location error, in more than 90 % of the time; or it can achieve fewer than 1 m error, 40 % of the time. This prediction accuracy is roughly 10 % better than the performance achieved from the same datasets without conformal prediction algorithms [27]. Compared to RADAR, the first indoor tracking system using a similar Location Fingerprinting technique, but with just a simple K-NN algorithm [2], our algorithm enhanced the prediction accuracy by more than 25 % (Fig. 5), where we use a Cumulative Distribution Function (CDF) plot to compare the performance of the algorithms.

Our algorithms, which bases on W-KNN and Regression, perform equally well with other well-known probabilistic indoor tracking systems, which also use a training database such as the Horus system [41]. Those systems use the Bayesian

**Fig. 5** Prediction accuracy enhancement with RADAR system



**Table 3** Location accuracy at different database sizes

Database size	100 %	70 %	50 %	30 %
CP ridge regression	1.4 m (100 %)	1.66 m (-19 %)	1.7 m (-22 %)	2.3 m (-67 %)
CP classification	1.5 m (100 %)	1.8 m (-20 %)	1.86 m (-24 %)	2.6 m (-74 %)
Bayesian	1.3 m (100 %)	1.6 m (-23 %)	1.8 m (-41 %)	2.5 m (-92 %)

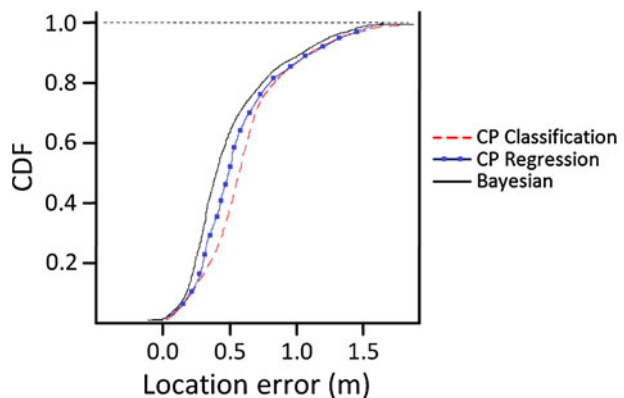
approach to predict the location [41], which perform very well when the training database is dense, and covers enough signal frequency at each tracking spot. In terms of adaptability, our algorithms perform better than the Bayesian approach when the database is sparse, and the signal reading is intermitted. Table 3 shows that the performance of Bayesian approach degrades rapidly in terms of location accuracy, as the training database size is reduced. For example, when we randomly reduced the training database size to only 30 %, the location accuracy of the Bayesian approach went down rapidly from 1.3 m to 2.5 m, while the CP Ridge Regression maintained a 2.3 m accuracy with the same training data. Of the three algorithms, the prediction accuracy of the Bayesian approach suffers the most, when the training database is sparse.

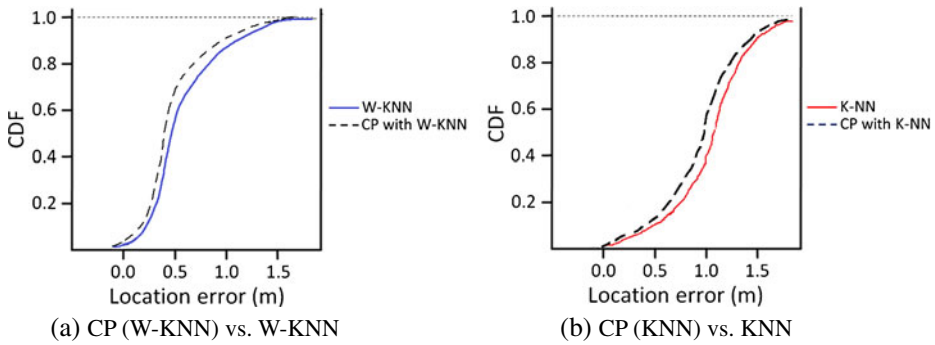
The W-KNN and ridge regression algorithms also have the benefit of lower complexity, which is important for real-time tracking with a large training database. Figure 6 demonstrates the performance of the Bayesian, Regression and Classification approach on Testbed 1. Overall, with a good training database covering many sample locations, the Bayesian approach performs better.

#### 4.4 Performance gain from the conformal prediction alone

Since CP returns a set of locations, while traditional K-NN returns just one estimated location for each test sample, we averaged the prediction region returned by CP as a single location for comparison. Figure 7a shows that CP using W-KNN as the underlying algorithm outperforms the W-KNN algorithm by 5–10 % more accuracy, at a significance level  $\epsilon = 0.13$  and  $K = 16$ . Comparing to the un-Weighted version of K-NN, CP enhances the performance of the algorithm by a similar 5–10 % (Fig. 7b). A similar improvement in terms of location accuracy was achieved with CP for

**Fig. 6** Bayesian vs. regression vs. classification on Testbed 1





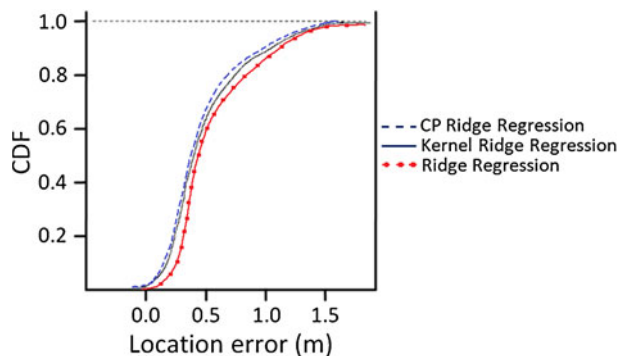
**Fig. 7** Performance of CP with W-KNN and K-NN on Testbed 1

Ridge Regression, in comparison to Ridge Regression and Kernel Ridge Regression RBF (Fig. 8). This accuracy improvement is important for indoor tracking, where centimetre accuracy is desirable. More importantly, by applying CP, we will provide a confidence parameter for our prediction, as we will explain later.

#### 4.5 Confidence level vs. prediction region

The Error Rate of CP is the percentage in which CP does not produce a prediction region containing the exact location. For simplicity, we consider just two dimensions  $X$  and  $Y$  of location, while the  $Z$  co-ordinate is fixed in this paper. To study the credibility of the prediction region, we evaluate different confidence levels (see Tables 4, 5, 6 and 7). Figures 9a and 10a demonstrate the prediction region at  $\varepsilon = 0.05$  with a set of predicted locations concentrated around the actual position. At the same position, by lowering the confidence level, we achieved far less predicted results (Figs. 9b and 10b). The wrongly predicted positions were mainly caused by the indoor multi-path effect. Those positions have very similar signal strength pattern with the correct position, although their physical locations are not similar. In the same example, we saw that Regression produced a smaller prediction region (Figs. 11 and 12).

**Fig. 8** Performance of CP with ridge regression on Testbed 1



**Table 4** System accuracy at different  $\epsilon$  of classification for Testbed 1

Confidence level (%)	Significance level $\epsilon$	Prediction error (average) (m)	Prediction size (average)	CP error rate (average) (%)
90	0.10	>2	62	7.2
87	0.13	1.7	29	11.5
71	0.29	>1.8	8	28.7

**Table 5** System accuracy at different  $\epsilon$  of classification for Testbed 2

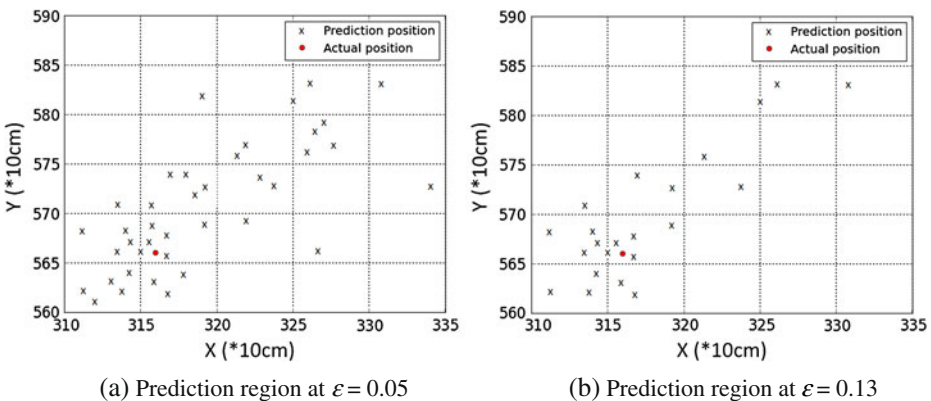
Confidence level (%)	Significance level $\epsilon$	Prediction error (average) (m)	Prediction size (average)	CP error rate (average) (%)
90	0.10	>3.1	44	9.5
87	0.13	2.5	19	12.3
71	0.29	>3	13	27.6

**Table 6** System accuracy at different  $\epsilon$  of regression for Testbed 1

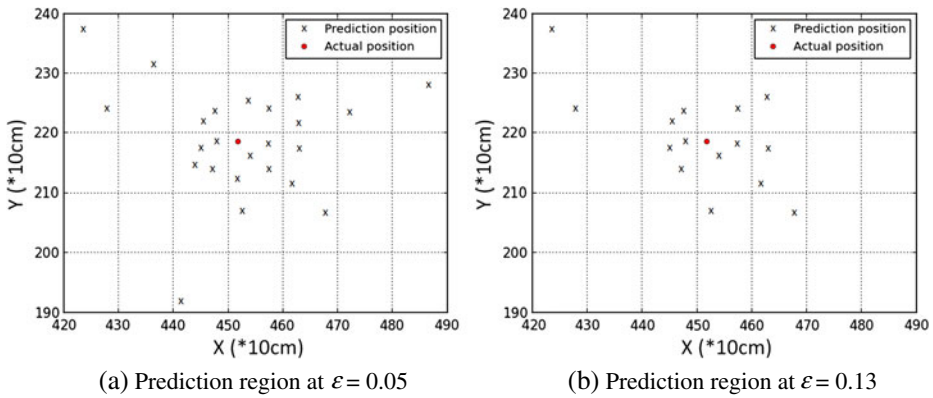
Confidence level (%)	Significance level $\epsilon$	Prediction error (average) (m)	CP error rate (average) (%)
90	0.10	>2	9
87	0.13	1.5	12.2
71	0.29	>1.9	27.9

**Table 7** System accuracy at different  $\epsilon$  of regression for Testbed 2

Confidence level (%)	Significance level $\epsilon$	Prediction range (average) (m)	CP error rate (average) (%)
90	0.10	>2.7	9.6
87	0.13	2.4	12.7
71	0.29	>2.9	28.2



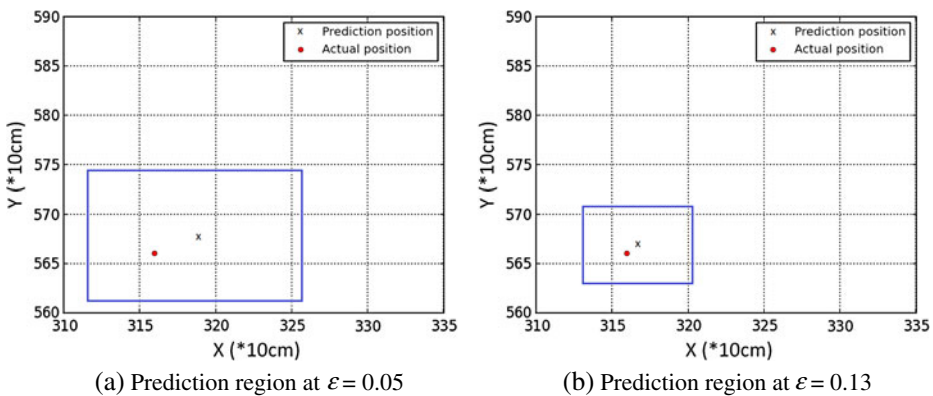
**Fig. 9** Prediction region of a real-time sample using classification on Testbed 1



**Fig. 10** Prediction region of a real-time sample using classification on Testbed 2

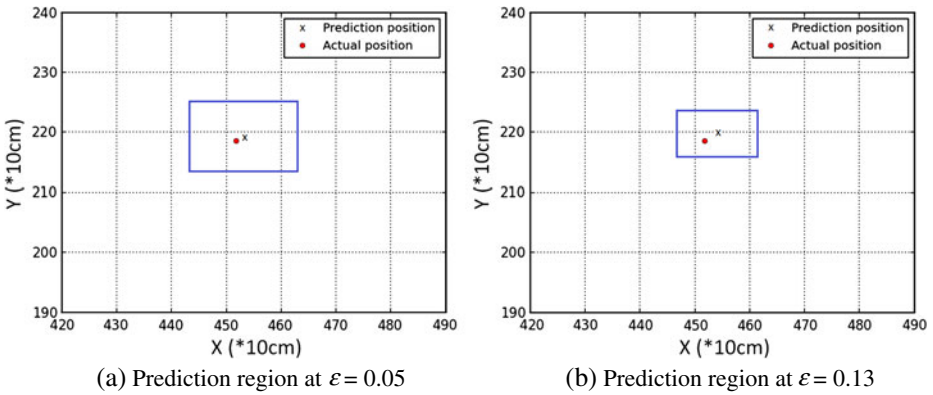
We observed that the size of the prediction region decreases as the significance level  $\epsilon$  increases. By dividing the training database into two equal sets of samples, and used the second set as a test set and the first set as a training set, we worked out that a reasonable confidence level was around 87% with a prediction range of 1.5–1.7 m for Testbed 1, which is acceptable for tracking users at sub-room resolution. There was only 3.5 % of samples with more than 3.5 m error. Increasing the confidence level to above 90 % allows too many samples getting in the prediction region (see Tables 4 and 5). With a small amount of stations in our training datasets, Ridge Regression performed slightly better, with a prediction range of less than 1.5 m for Testbed 1, and less than 2.1 m for Testbed 2 (see Tables 6 and 7).

To verify the validity of the above prediction region, we performed 10-fold cross validation with the training data only. We divided  $L = 64,000$  samples in the first training set into ten partitions, then we used nine partitions for training and the remaining one for testing. The second training set was divided in a similar manner. At the same  $\epsilon = 0.13$ , the error rate was achieved at 10.8 and 8 %, which remains valid



**Fig. 11** Prediction region of a real-time sample using regression on Testbed 1





**Fig. 12** Prediction region of a real-time sample using regression on Testbed 2

**Table 8** 10-fold cross validation classification for Testbed 1

Confidence level (%)	Significance level $\epsilon$	CP error rate (average) (%)
95	0.05	5.3
90	0.10	10.7
87	0.13	14.5
80	0.20	20.4

**Table 9** 10-fold cross validation classification for Testbed 2

Confidence level (%)	Significance level $\epsilon$	CP error rate (average) (%)
95	0.05	6.1
90	0.10	11.3
87	0.13	13.8
80	0.20	21.6

**Table 10** 10-fold cross validation regression for Testbed 1

Confidence level (%)	Significance level $\epsilon$	CP error rate (average) (%)
95	0.05	4.8
90	0.10	9.5
87	0.13	13.6
80	0.20	20.3

**Table 11** 10-fold cross validation regression for Testbed 2

Confidence level (%)	Significance level $\varepsilon$	CP error rate (average) (%)
95	0.05	5.1
90	0.10	9.8
87	0.13	13.4
80	0.20	21.7

up to statistical fluctuations for both Testbed 1 and Testbed 2 (see Tables 8, 9, 10 and 11).

#### 4.6 Calibrating the $K$ and ridge factor parameter

Finding an optimal  $K$  parameter for the W-KNN is challenging. Each environment and each signal property require the  $K$  parameter to be calibrated accordingly. In our system, we experimentally found  $K = 16$  to be an optimal value across all testing points, for our training database. However, some locations had smaller location error with different  $K$  value. Generally, starting from  $K = 1$ , which is equivalent to 1-nearest neighbour only, the accuracy tends to increase as  $K$  increases, up to a certain threshold ( $K = 16$  in our case). Then, it begins to decrease as  $K$  continues to increase. In our experience, from  $K = 16$  onwards, the performance is only 7–10 % affected by increasing  $K$ .

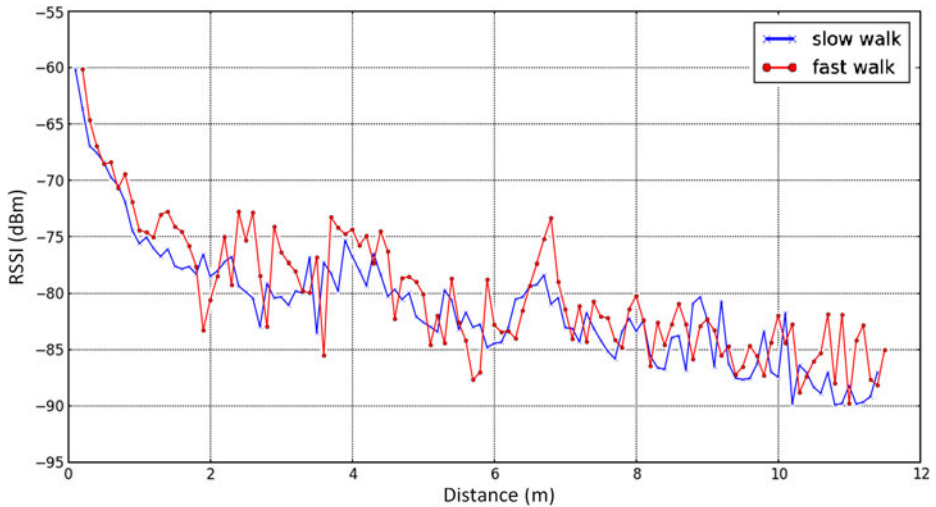
For Ridge factor parameter, in our experience with the Bluetooth signal data, the smaller the factor, the better the prediction accuracy. We performed a grid search with ridge factor ranging from 100 to 0.001, and found the best value to be around 0.01.

## 5 Conclusion and further work

This section outlines the main contributions of this paper, and the ideas to apply CP for tracking a mobile user.

### 5.1 Contributions

We have demonstrated the application of Conformal Prediction into the Indoor Localisation problem. To the best of our knowledge, we are the first to implement Conformal Prediction algorithm into the Localisation problem in general, and the Indoor Localisation in particular. Both classification and regression approaches were considered and evaluated on two testbeds. We also designed a new nonconformity measure with the Weighted K-nearest neighbours as the underlying algorithm. Our initial results showed very promising location accuracy, compared to other traditional algorithms. Our on-going work is to design different nonconformity measures, specifically for the indoor localisation problem where multi-label prediction is needed. We are setting up a new experiment for our Computer Science Department to examine the efficiency of the algorithms in a large scale instance.



**Fig. 13** Wireless signal strength vs. walking speed

## 5.2 Tracking a mobile user

Another advantage of CP is the on-line learning setting, in which CP learns from its previous predictions and updates its training database to improve the accuracy of the next prediction. This idea perfectly fits the purpose of tracking an indoor mobile user, since the user cannot jump a long distance in a short period of time, his next movement is within a certain radius of his current position.

A challenge to apply CP, however, is the latency of many tracking systems, where the environment does not always response immediately. Especially with Bluetooth tracking, the expected delay can be as high as 1.28 s, and the user might have moved away within that time-frame. This is the case of slow teacher, where immediate feedback cannot be guaranteed [35].

Tracking a moving user is also challenging, because the higher the walking speed, the more un-reliable the signal strength becomes. Figure 13 demonstrates the effect of movements on the wireless Bluetooth signal strength. A possible solution is to evaluate the trustworthiness of the received signal at a particular moment. For example, a signal received in less than 0.5 s would have higher reliability than that of 2 s. It is our on-going work to inspect the improvement of CP for tracking a mobile user, considering the user movement speeds.

**Acknowledgements** We would like to thank the anonymous reviewers for their insightful comments on our work. This project is funded by the Computer Science Department, Royal Holloway, University of London and a grant from the National Natural Science Foundation of China (No. 61128003).

## References

1. Anastasi, G., Bandelloni, R., Conti, M., Delmastro, F., Gregori, E., Mainetto, G.: Experimenting an indoor bluetooth-based positioning service. In: 23rd International Conference on Distributed Computing Systems Workshops, pp. 480–483. IEEE (2003)

2. Bahl, P., Padmanabhan, V.N.: Radar: An in-building rf-based user location and tracking system. In: INFOCOM 2000. Proceedings of Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies., vol. 2, pp. 775–784. IEEE (2000)
3. Bargh, M.S., de Groote, R.: Indoor localization based on response rate of bluetooth inquiries. In: Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments, pp. 49–54. ACM (2008)
4. Battiti, R., Le, N.T., Villani, A.: Location-aware computing: a neural network model for determining location in wireless lans. Tech. Rep. DIT-02-0083 (2002)
5. Bellotti, T., Luo, Z., Gammerman, A., Van Delft, F.W., Saha, V.: Qualified predictions for microarray and proteomics pattern diagnostics with confidence machines. *Int. J. Neural Syst.* **15**(4), 247–258 (2005)
6. Brunato, M., Kiss Kallo, C.: Transparent location fingerprinting for wireless services. Tech. Rep. DIT-02-071 (2002)
7. Bruno, R., Delmastro, F.: Design and analysis of a bluetooth-based indoor localization system. In: *Personal Wireless Communications*, pp. 711–725. Springer (2003)
8. Chen, Y., Lymberopoulos, D., Liu, J., Priyantha, B.: Fm-based indoor localization. In: *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, pp. 169–182. ACM (2012)
9. Cheung, K.C., Intille, S.S., Larson, K.: An inexpensive bluetooth-based indoor positioning hack. In: *Proc. UbiComp06 Extended Abstracts* (2006)
10. Dashevskiy, M., Luo, Z.: Reliable probabilistic classification of internet traffic. *IJIA* **6**(2), 133–146 (2009)
11. Draper, N.R., Smith, H., Pownell, E.: *Applied Regression Analysis*. Wiley, New York (1966)
12. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification*. Wiley-interscience (2012)
13. Fang, S.H., Lin, T.N.: Indoor location system based on discriminant-adaptive neural network in iee 802.11 environments. *IEEE Trans. Neural Netw.* **19**(11), 1973–1978 (2008)
14. Hallberg, J., Nilsson, M., Synnes, K.: Positioning with bluetooth. In: *10th International Conference on Telecommunications, ICT 2003*, vol. 2, pp. 954–958. IEEE (2003)
15. Hay, S., Harle, R.: Bluetooth tracking without discoverability. In: *Location and Context Awareness*, pp. 120–137. Springer (2009)
16. Hightower, J., Borriello, G.: Location systems for ubiquitous computing. *Computer* **34**(8), 57–66 (2001)
17. Huang, A.: The use of bluetooth in linux and location aware computing. Ph.D. thesis, Massachusetts Institute of Technology (2005)
18. Jevring, M., de Groote, R., Hesselman, C.: Dynamic optimization of bluetooth networks for indoor localization. In: *Proceedings of the 5th International Conference on Soft Computing as Transdisciplinary Science and Technology*, pp. 663–668. ACM (2008)
19. Kemper, J., Linde, H.: Challenges of passive infrared indoor localization. In: *5th Workshop on Positioning, Navigation and Communication. WPNC 2008*, pp. 63–70. IEEE (2008)
20. Kothari, N., Kannan, B., Glasgwo, E.D., Dias, M.B.: Robust indoor localization on a commercial smart phone. *Procedia Computer Science* **10**, 1114–1120 (2012)
21. Letchner, J., Fox, D., LaMarca, A.: Large-scale localization from wireless signal strength. In: *Proceedings of the national conference on artificial intelligence*, vol. 20, pp. 15–20. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999 (2005)
22. Lin, T.N., Lin, P.C.: Performance comparison of indoor positioning techniques based on location fingerprinting in wireless networks. In: *2005 International Conference on Wireless Networks, Communications and Mobile Computing*, vol. 2, pp. 1569–1574. IEEE (2005)
23. Link, J.A.B., Smith, P., Viol, N., Wehrle, K.: Footpath: Accurate map-based indoor navigation using smartphones. In: *2011 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–8. IEEE (2011)
24. Madhavapeddy, A., Tse, A.: A study of bluetooth propagation using accurate indoor location mapping. In: *UbiComp 2005: Ubiquitous Computing*, pp. 105–122. Springer (2005)
25. Naya, F., Noma, H., Ohmura, R., Kogure, K.: Bluetooth-based indoor proximity sensing for nursing context awareness. In: *Ninth IEEE International Symposium on Wearable Computers*, pp. 212–213. IEEE (2005)
26. Nguyen, K., Luo, Z.: Evaluation of bluetooth properties for indoor localisation. In: *Progress in Location-Based Services*, pp. 127–149. Springer (2013)
27. Nguyen, K.A.: Robot-based evaluation of bluetooth fingerprinting. Master's thesis, Computer Lab, University of Cambridge (2011)

28. Pandya, D., Jain, R., Lupu, E.: Indoor location estimation using multiple wireless technologies. In: 14th IEEE Proceedings on Personal, Indoor and Mobile Radio Communications. PIMRC 2003, vol. 3, pp. 2208–2212. IEEE (2003)
29. Priyantha, N.B.: The cricket indoor location system. Ph.D. thesis, Massachusetts Institute of Technology (2005)
30. Rizos, C., Dempster, A.G., Li, B., Salter, J.: Indoor positioning techniques based on wireless lan (2007)
31. Robertson, P., Angermann, M., Krach, B.: Simultaneous localization and mapping for pedestrians using only foot-mounted inertial sensors. In: Proceedings of the 11th international conference on Ubiquitous computing, pp. 93–96. ACM (2009)
32. Schölkopf, B., Smola, A.J.: Learning with kernels: support vector machines, regularization, optimization and beyond. MIT Press (2002)
33. Shafer, G., Vovk, V.: A tutorial on conformal prediction. *The Journal of Machine Learning Research* **9**, 371–421 (2008)
34. Taheri, A., Singh, A., Emmanuel, A.: Location fingerprinting on infrastructure 802.11 wireless local area networks (wlans) using locus. In: 29th Annual IEEE International Conference on Local Computer Networks, pp. 676–683. IEEE (2004)
35. Vovk, V., Gammerman, A., Shafer, G.: *Algorithmic learning in a random world*. Springer Science+ Business Media (2005)
36. Want, R., Hopper, A., Falcão, V., Gibbons, J.: The active badge location system. *ACM Transactions on Information Systems (TOIS)* **10**(1), 91–102 (1992)
37. Ward, A., Jones, A., Hopper, A.: A new location technique for the active office. *IEEE Personal Communications* **4**(5), 42–47 (1997)
38. Wölfle, G., Hoppe, R., Zimmermann, D., Landstorfer, F.M.: Enhanced localization technique within urban and indoor environments based on accurate and fast propagation models. In: *European Wireless*, pp. 25–28 (2002)
39. Woodman, O., Harle, R.: Pedestrian localisation for indoor environments. In: Proceedings of the 10th international conference on Ubiquitous computing, pp. 114–123. ACM (2008)
40. Xiang, Z., Song, S., Chen, J., Wang, H., Huang, J., Gao, X.: A wireless lan-based indoor positioning technology. *IBM Journal of Research and Development* **48**(5.6), 617–626 (2004)
41. Youssef, M., Agrawala, A.: The horus location determination system. *Wireless Networks* **14**(3), 357–374 (2008)
42. Youssef, M.A., Agrawala, A.: On the optimality of wlan location determination systems. Tech. Rep. CS-TR-4459 (2003)