

Article

Improving Wi-Fi Fingerprint Positioning with a Pose Recognition-Assisted SVM Algorithm

Shuai Zhang ^{1,2}, Jiming Guo ^{1,2,*}, Nianxue Luo ¹, Lei Wang ³, Wei Wang ¹ and Kai Wen ¹

¹ School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China; 2016102140015@whu.edu.cn (S.Z.); nxluo@sgg.whu.edu.cn (N.L.); wanngwei@whu.edu.cn (W.W.); wenkai@whu.edu.cn (K.W.)

² Research Center for High Accuracy Location Awareness, Wuhan University, Wuhan 430079, China

³ State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China; lei.wang@whu.edu.cn

* Correspondence: jmguo@sgg.whu.edu.cn; Tel.: +86-027-6877-8807-801

Received: 31 January 2019; Accepted: 14 March 2019; Published: 17 March 2019



Abstract: The fingerprint method has been widely adopted for Wi-Fi indoor positioning. In the fingerprint matching process, user poses and user body shadowing have serious impact on the received signal strength (RSS) data and degrade matching accuracy; however, this impact has not attracted large attention. In this study, we systematically investigate the impact of user poses and user body shadowing on the collected RSS data and propose a new method called the pose recognition-assisted support vector machine algorithm (PRASVM). It fully exploits the characteristics of different user poses and improves the support vector machine (SVM) positioning performance by introducing a pose recognition procedure. This proposed method firstly establishes a fingerprint database with RSS and sensor data corresponding to different poses in the offline phase, and fingerprints of different poses in the database are extracted to train reference point (RP) classifiers of different poses and a pose classifier using an SVM algorithm. Secondly, in the online phase, the poses of RSS data measured online are recognised by a pose classifier, and RSS data measured online are grouped with different poses. Then online RSS data from each group at an unknown user location are reclassified as corresponding RPs by the RP classifiers of the corresponding poses. Finally, user location is determined by grouped RSS data corresponding to coordinates of the RPs. By considering the user pose and user body shadowing, the observed RSS data matches the fingerprint database better, and the classification accuracy of grouped online RSS data is remarkably improved. To verify performances of the proposed method, experiments are carried out: one in an office setting, and the other in a lecture hall. The experimental results show that the positioning accuracies of the proposed PRASVM algorithm outperform the conventional weighted k-nearest neighbour (WKNN) algorithm by 52.29% and 40.89%, outperform the SVM algorithm by 73.74% and 60.45%, and outperform the pose recognition-assisted WKNN algorithm by 34.76% and 21.86%, respectively. As a result, the PRASVM algorithm noticeably improves positioning accuracy.

Keywords: Wi-Fi fingerprint; pose recognition; PRASVM; indoor positioning

1. Introduction

Location-based services (LBS) have rapidly increased with the growth of mobile intelligent terminals. At present, the global navigation satellite system (GNSS) can provide satisfactory positioning accuracy in outdoor environments. However, GNSS cannot meet the requirements of indoor positioning due to signal fading and the multipath effect in indoor environments [1–3]. Various indoor positioning technologies, such as ultrasound, Wi-Fi, RFID, Bluetooth, ZigBee, geomagnetic

positioning, and ultra-wide band, have been investigated for indoor positioning. Among them, Wi-Fi indoor positioning has attracted considerable attention because it does not require additional equipment [4]. Several methods have been adopted for Wi-Fi indoor positioning: the angle of arrival (AOA), time of arrival (TOA), time difference of arrival (TDOA), and fingerprint method [5–9]. AOA, TOA, and TDOA require point-to-point distance or angle information. These methods have simple calculations, but they are developed under the condition of line of sight (LOS) channels between access points (APs) and mobile users. Conversely, the fingerprint method does not require an LOS assumption between APs and mobile users, and has been widely adopted for indoor positioning.

Mobile devices have become increasingly smart. Numerous sensors, such as accelerometers, gyroscopes, magnetometers, proximity sensors and cameras, are built in mobile devices. Sensors have been applied for motion recognition. Tran et al. [10] used sensor data to recognise actions based on SVM. Six actions, namely, walking, standing, sitting, lying down, going upstairs and going downstairs, were selected in their research. Yang et al. [11] investigated the physical motion of a user by using a built-in accelerometer in smartphones, and their main purpose was to infer a user's daily physical activity. Pei et al. [12] used a wireless positioning method based on motion recognition-assisted hidden Markov model (HMM). In the method, a grid-based filter of HMM is implemented to produce an optimal estimation based on previous motion state, and motion recognition is used to rapidly update the motion state. In our study, the sensor data are used to recognise user poses, which are obtained by using some built-in sensors in smartphones.

Some researchers have done some explorations using the SVM in indoor positioning. Tran et al. [13] investigated a distributed positioning technique based on the SVM for wireless sensor networks. The method defined a set of geographical regions in the sensor field and classified each sensor node into these regions. Then, its location was estimated inside the intersection of the containing regions. The training data were the set of beacons. However, this method needed a lot of beacon and large overheads. Wu et al. [14] tackled the GSM location estimation problems through SVM. Brunato et al. [15] established an SVM-based Wi-Fi strategy for determining the location of mobile user by using RSS intensity. Sang et al. [16] presented Wi-Fi indoor positioning methods based on support vector regression and support vector classification. This method firstly divided the indoor environment into small areas, and then used the classification model to judge the user's sub-regions. Lastly, the target location was estimated by using the regression model of the sub-regions. Yu et al. [17] utilised support vector classification based on fingerprint map for positioning. They only used the maximum RSS value at a test point (TP) as the positioning evaluation standard, which is easily affected by misclassified data due to insufficient RSS data measured online. In our study, a PRASVM algorithm is proposed. Firstly, SVM is used to recognise the poses of the mobile user with sensor data, and RSS data measured online are grouped with different poses. Then online RSS data from each group at an unknown user location are reclassified as corresponding RPs by the RP classifiers of the corresponding poses. Finally, user location is determined by grouped RSS data corresponding the coordinates of the RPs. The positioning performance of the PRASVM algorithm is remarkably improved compared to conventional indoor positioning algorithms [18].

In this paper, we discover that user poses and user body shadowing have serious impact on the collected RSS data and degrade the matching accuracy. The PRASVM algorithm is proposed. In addition, some environmental factors that impact positioning are analysed.

The remainder of this paper is organised as follows: Section 2 analyses the impact of poses on the fingerprint positioning, and illustrates the PRASVM algorithm in detail. Section 3 describes the experimental results of the PRASVM algorithm on positioning. Section 4 discusses the main contents and significance of this paper, as well as our future research directions. Section 5 summarises our conclusions.

2. Materials and Methods

2.1. Impact of Poses on Fingerprint Positioning

In this section, the pose states are described, and 12 poses are defined. Then, the influence of different cardinal orientations and poses on RSS are analysed. Finally, the positioning error resulting from different orientations and poses are compared.

2.1.1. Definition of User Poses

Pose recognition has been investigated by scholars for a long time. Pose recognition has mainly been used for health monitoring of the elderly and children in previous studies [19,20]. At present, pose recognition is used in video classification, human–computer interaction, and security monitoring [21,22].

The impact of different orientations and poses of mobile users on Wi-Fi positioning is investigated in complex indoor environments. In this study, three typical user poses are used: calling, handheld, and pocket [23–25]. Sketches of several typical poses of mobile users are shown in Figure 1.

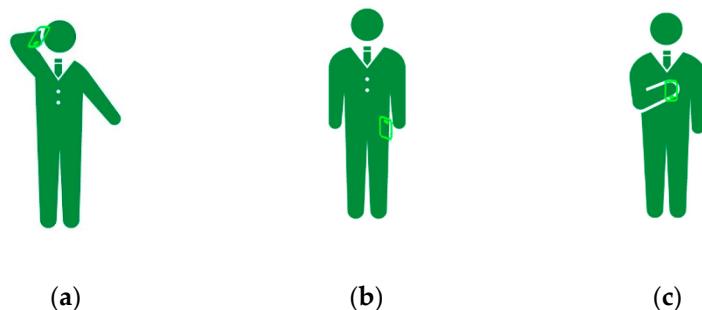


Figure 1. Sketches of three typical user poses: (a) calling pose; (b) pocket pose; (c) handheld pose.

The three basic poses are extended to 12 orientation-related poses. These are useful to investigate the effect of different poses and body shadowing on positioning. The 12 poses are defined as follows.

Calling towards the north: In this state, a standing mobile user is calling without any movement; his (her) hand is to his (her) ear, and his (her) body faces north. There are three similar poses, namely, calling towards the south, calling towards the east, and calling towards the west.

Handheld towards the north: A standing mobile user keeps his (her) eyes on the screen of a smartphone without any movement, and his (her) body faces north. The smartphone is in his (her) hand, held at chest level. There are three similar poses, namely, handheld towards the south, handheld towards the east, and handheld towards the west.

Pocket towards the north: A standing mobile user places the smartphone in his (her) trouser pocket without any movement, and his (her) body faces north. Three similar poses exist, namely, pocket towards the south, pocket towards the east, and pocket towards the west.

2.1.2. Influence of Different Poses on RSS Data

In real life, people hold mobile phones in various ways, creating different body poses. To verify the influence of different poses on RSS, three typical user poses are selected in this study: calling, handheld, and pocket. We conduct this experiment in a conference room. The RSS data are collected by the smartphones for 5 min, and 300 RSS data are obtained in each pose. As shown in Figure 2, the received RSS data have significant differences in across poses. The heights of the smartphones from the ground are different in each pose, and the RSS data received by the smartphones are subject to different environmental factors at different heights. As a result, the received RSS values also are different. In the pocket pose, the mean received RSS is the lowest across the poses. The calling pose has the highest RSS.

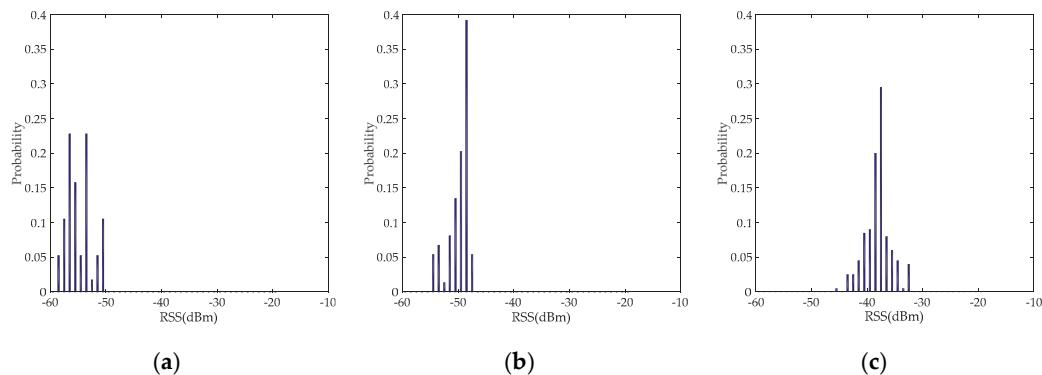


Figure 2. Probability histograms of RSS values with different user poses: (a) pocket pose; (b) handheld pose; (c) calling pose.

2.1.3. Influence of Body Shadowing on RSS Data

To verify the influence of body shadowing on RSS data we conduct this experiment in a conference room. There is an AP in the room. The RSS data are received in four different orientations: facing the AP, with one's back to the AP, and lateral to the AP. During this period of data acquisition, the users are static in each orientation. RSS data are first collected when the user is facing the AP; data collection proceeds with 90 degrees clockwise turns to acquire data under all orientation conditions. The RSS data is collected in each orientation for 5 min (300 RSS data). The results show that the RSS data characteristics are different with the different orientations of the body. As shown in Figure 3, the received RSS data have the most serious attenuation when the user's back is to the AP, whereas minimum attenuation is seen when facing the AP.

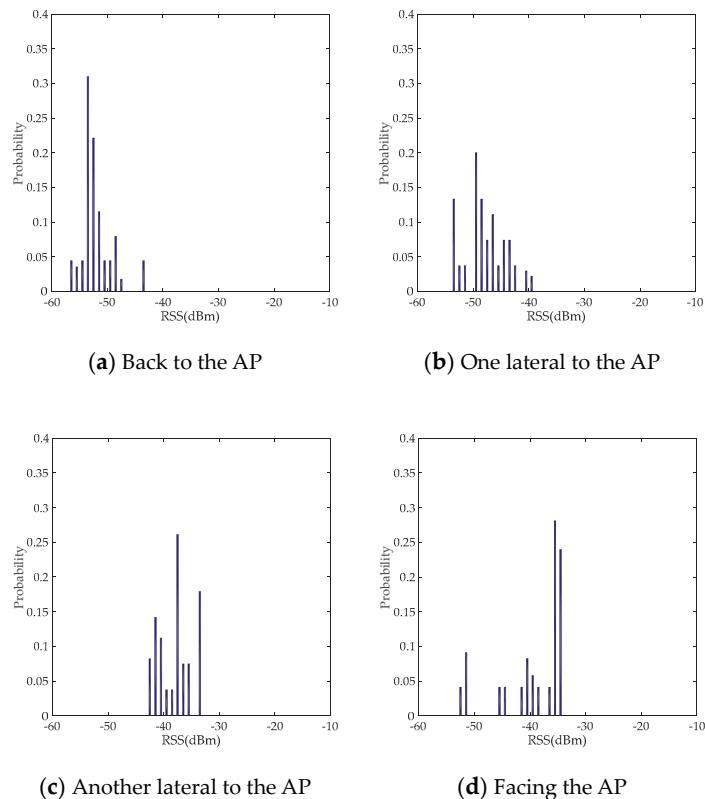


Figure 3. Probability histograms of received RSS values in four different orientations: (a) back to the AP; (b) one lateral to the AP; (c) another lateral to the AP; (d) facing the AP.

2.1.4. Comparison of Positioning Error with Different Cardinal Orientations and Poses

To analyse positioning performances with different orientations and poses, we design a test scheme in a complex office environment. Some TPs and RPs are laid out in the test. To only consider positioning performances in different poses, we collect RSS data with a pose (e.g., the calling pose) and four orientations over a period of time at each RP, and collect RSS data with three poses and four orientations over a period of time at each TP. In order to eliminate the effect of orientation on positioning, data collection in each orientation takes the same time at each RP and TP. To analyse positioning performances in different poses, data of the same pose and different poses are used for fingerprint matching in the offline and online phases. Similarly, to consider only positioning performances in different orientations, we collect RSS data with an orientation (e.g., east) and three poses over a period of time at each RP, and collect RSS data with three poses and four orientations over a period of time at each TP. In order to eliminate the effect of the pose on positioning, data collection in each pose takes the same time at each RP and TP. To analyse positioning performances in different orientations, data of the same orientation and different orientations are used for fingerprint matching in the offline and online phases. The different orientations include east, south, north, west, and mixed orientations. The different poses include calling, handheld, pocket, and mixed poses. The WKNN algorithm is adopted to estimate the location.

The test results show that the mean error of the same calling pose is remarkably less than that of other poses in the offline and online phases. Furthermore, the test results also show that the mean error of the same eastward orientation is remarkably less than that of other orientations in the offline and online phases. As shown via error comparisons in Figure 4, positioning results of WKNN algorithm are optimal when the pose can be completely matched in the offline and online phases. The positioning accuracy is better than 1 m with the same pose.

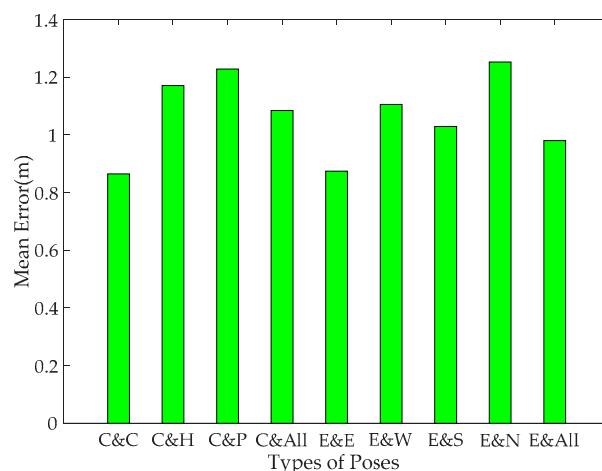


Figure 4. Positioning accuracy of the WKNN algorithm when users take different poses. Each pose is represented by the first letter of the word in the figure. ‘C’ represents the calling pose and ‘All’ represents a mixed pose. ‘C&C’ means that fingerprint data with the calling pose in the offline phase are matched with RSS data with the calling pose in the online phase.

Through the analysis of different orientations and poses, we find that different poses have a serious influence on received RSS data, and we consider the positioning error in different orientations and poses. We also discover that different poses have a serious influence on positioning results. As a result, user poses and user body shadowing have negative impacts on RSS, and degrades matching performance. The current fingerprint approach considers only the spatial distribution of RSS data, and this is the key factor hindering positioning accuracy improvement; thus, this should be seriously considered in the fingerprint matching procedure.

2.2. PRASVM Algorithm for Fingerprint Positioning

In order to incorporate the impact of user poses and user body shadowing in RSS collection, we proposed a PRASVM algorithm. This algorithm firstly recognises the user's pose, and then matches the RP classifier corresponding to the right user pose. In this way, the negative effect of user pose can be eliminated, which improves the performance of the SVM on positioning, and thus allows us to obtain better positioning results. In this section, the SVM and positioning algorithm are described in detail. Finally, the implementation of the PRASVM algorithm is depicted.

2.2.1. Principle of PRASVM Algorithm

Fingerprint positioning is a widely used method for Wi-Fi indoor positioning [26–30] because it does not require LOS conditions. In this paper, the PRASVM algorithm is proposed on the basis of fingerprint positioning. The fingerprint method with poses consists of two phases, namely, offline and online phases. In the offline phase, the location area is divided into grids. The RPs are deployed in the grids. The location accuracy depends on the density of RPs. The RSS values at each RP are collected at pre-set time intervals when the smartphone user takes different poses. The collected RSS data are used as RP training samples. The built-in sensors in the smartphone are used for collecting sensor data during fingerprint data collection. The sensor data are used as pose training samples for pose recognition. A pose classifier is trained by using the pose training samples with different poses, and RP classifiers in different poses are trained by using the RP training samples in the offline phase. The MAC address, RSS data, sensor data, pose label, service set identifiers of Aps, and the location information of RPs are recorded in the fingerprint database. The fingerprint database is expressed as follows:

$$RP_L = \{RSS_{L1}, RSS_{L2}, RSS_{L3}, \dots, RSS_{Ln}, sensordata, poselabel, MAC, SSID, x_L, y_L\}, \quad (1)$$

where RP_L is the vector of the fingerprint database at the RP. n is the number of APs, and (x_L, y_L) are the coordinates of the L th RP. In the online phase, in order to determine user location, firstly the poses of RSS data measured online at unknown user location are recognised by the trained pose classifier, and RSS data measured online are grouped with different poses. Then online RSS data from each group at an unknown user location are reclassified as corresponding RPs by the RP classifiers of the corresponding poses. Finally, user location is estimated using grouped RSS data corresponding the coordinates of the RPs. Figure 5 illustrates the principle of the PRASVM algorithm.

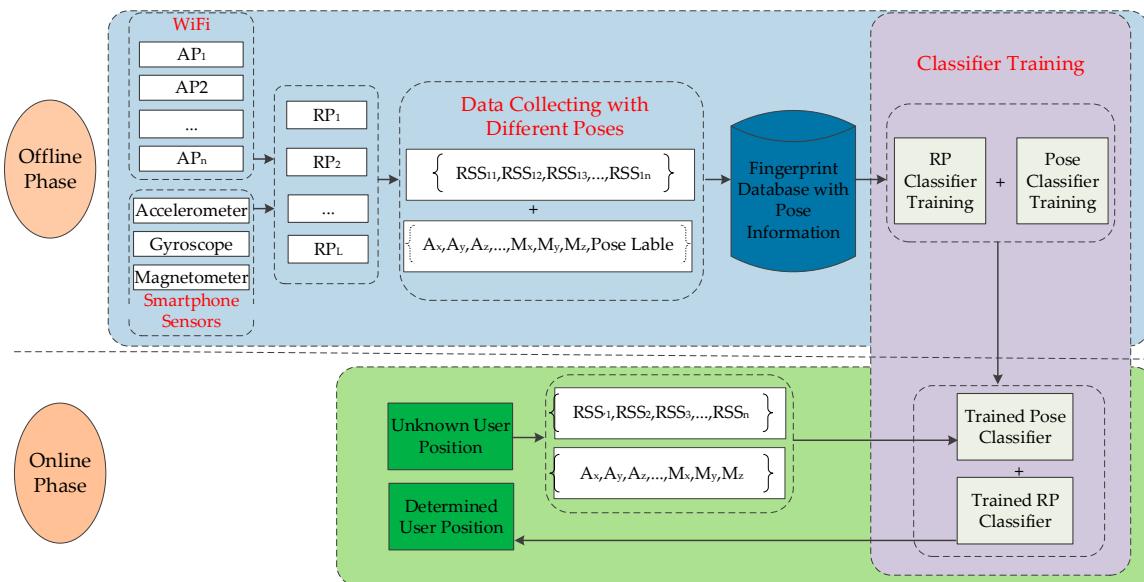


Figure 5. Schematic diagram of fingerprint positioning of the PRASVM algorithm.

2.2.2. Pose Recognition with the SVM Algorithm

There are various machine learning algorithms, such as the Naïve Bayes [31], k-nearest neighbour [32], decision tree [33], neural network [34], and SVM [35–38]. The SVM algorithm was originally developed for binary classification problems, but is now widely used to solve multi-classification problems. It is also used for pose recognition and fingerprint positioning. This algorithm is suitable for linear and nonlinear classifications. To solve pose recognition problems, a kernel function is used to map the pose training samples onto a high-dimensional space, and the best pose classification hyperplane is obtained in the high-dimensional space. The best hyperplane ensures that the margin between different poses is at a maximum. The kernel function can be a linear kernel function, polynomial kernel function, radial basis function (RBF), or sigmoid kernel function. The linear kernel function is used for pose recognition, and the RBF is used for the classification of RPs. When RSS data are used for RP classification, classification results of the linear kernel function is not ideal, and classification results of the RBF are better. Moreover, the RP training data are imperfect for linear separation. Classification data contain training and testing samples. Each pose training sample contains a class label and some classification features. A series of pose training data are used as the training samples set $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$. x_i is equal to $(x_{11}, x_{12}, \dots, x_{1L})$, which is a vector that refers to the used classification features. A 9D feature vector is used for pose recognition in this paper, which includes three-axis accelerometer, three-axis gyroscope and three-axis magnetometer data. L is the dimensionality of features. $y_i (i = 1, 2, \dots, n)$ denotes the label of training samples. SVM penalises the samples of violation constraints to obtain an optimised result. The specific principle is expressed as follows:

$$\min_{\omega, \xi, b} J(\omega, \xi) = \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^n \xi_i, \quad (2)$$

$$\text{subject to : } \begin{cases} y_i (\omega^T x_i + b) \geq 1 - \xi_i \\ \xi_i \geq 0, i = 1, \dots, n \\ C > 0 \end{cases},$$

where J is the cost function. ω is the weight vector of the separating hyperplane. b is a constant. Constant C is a cost factor that is used to balance the relation between a maximum separation region and misclassified samples. The larger the C value, the fewer misclassified samples. ξ are slack variables that allow some samples for error classification.

A classifier is trained by using the following classification function:

$$f(x) = \text{sign} \left(\sum_{i=1}^n \alpha_i y_i K(x, x_i) + b \right), \quad (3)$$

where α_i is Lagrange multiplier. $K(x, x_i)$ is the kernel function.

This study selects a linear kernel function (4) for pose recognition [39].

$$K(x, x_i) = x^T x_i, \quad (4)$$

As a result, the trained pose classifier is used to recognise the poses of online RSS data at each TP.

2.2.3. Positioning Algorithm with Pose Information

Conventional deterministic and probabilistic methods are adopted to predict the location of mobile users. Deterministic methods include the nearest neighbour (NN), k-nearest neighbour (KNN), and WKNN [40–42]. The several nearest RPs to the TP are found by comparing the Euclidean distance between the TP and each RP. We add pose information to improve the conventional positioning method. In the pose recognition-assisted positioning algorithm, Euclidean distance D_i between the i th RP and TP is defined as

$$D_i = \sqrt{\sum_{r=1}^m \sum_{j=1}^n (\bar{rs}_{i,r}^j - \bar{ts}_r^j)^2}, \quad (5)$$

where $\bar{rs}_{i,r}^j$ denotes the average RSS values of the r th pose of the j th AP over a time period at the i th RP ($i = 1, 2, 3, \dots, L$). \bar{ts}_r^j indicates the average RSS values of the r th pose of the j th AP over a time period at the TP. L is the number of RPs, m is the number of pose types, and n is the number of APs.

The pose recognition-assisted deterministic positioning algorithms aim to obtain the several closest RPs to the TP in each pose. However, some small distinctions are observed between these methods. The pose recognition-assisted NN algorithm finds the nearest RP through the Euclidean distance between the TP and each RP in each pose. The coordinates of the nearest RP are regarded as the coordinates of the TP in Equation (6), where $K = 1$. Meanwhile, the pose recognition-assisted KNN algorithm finds the several nearest RPs through the Euclidean distance between the TP and each RP in each pose. The average coordinates of the nearest RPs are regarded as the coordinates of the TP in (6), where $K > 1$. Finally, the pose recognition-assisted WKNN algorithm finds the several nearest RPs through the Euclidean distance between the TP and each RP in each pose, calculates the weight of each RP coordinate in Equation (7), and obtains the weighted coordinates of the TP in (8), where $K > 1$.

$$(\bar{x}, \bar{y}) = \frac{1}{K} \sum_{r=1}^m \sum_{i=1}^K (x_{i,r}, y_{i,r}), \quad (6)$$

where $(x_{i,r}, y_{i,r})$ represents the location coordinates of the i th RP in the condition of the r th pose ($i = 1, 2, \dots, K$). K is the number of nearest RPs.

$$w_{i,r} = \frac{\frac{1}{d_{i,r}}}{\sum_{i=1}^K \frac{1}{d_{i,r}}}, \quad (7)$$

where $d_{i,r}$ is the Euclidean distance between the i th RP ($i = 1, 2, \dots, K$) and the TP in the condition of the r th pose.

$$(x, y) = \sum_{r=1}^m \sum_{i=1}^K w_{i,r} (x_{i,r}, y_{i,r}), \quad (8)$$

where $w_{i,r}$ denotes weight of the i th RP ($i = 1, 2, \dots, K$) coordinates in the condition of the r th pose.

The probabilistic fingerprint positioning method mainly uses Bayesian theorem to calculate the posterior probability. The pose information can also be added to probabilistic positioning methods. The probability of each RP location given the RSS value at the TP is used to weigh the corresponding coordinates of the RP. (x, y) is the estimated location of the TP in each pose:

$$(x, y) = \sum_{r=1}^m \sum_{i=1}^n (p(rl_{i,r}|ts_r) \times (x_{i,r}, y_{i,r})), \quad (9)$$

where $p(rl_{i,r}|ts_r)$ is the probability of the i th ($i = 1, 2, \dots, n$) RP location given the RSS value at the TP in the condition of the r th pose. $(x_{i,r}, y_{i,r})$ represents the location coordinates of the i th RP. Probability $p(rl_{i,r}|ts_r)$ is converted into probability $p(ts_r|rl_{i,r})$ by using Equation (10) in the condition of the r th pose:

$$p(rl_{i,r}|ts_r) = \frac{p(ts_r|rl_{i,r}) \times p(rl_{i,r})}{\sum_{i=1}^n (p(ts_r|rl_{i,r}) \times p(rl_{i,r}))}, \quad (10)$$

where $p(rl_{i,r}) = \frac{1}{n}$ is a constant. $p(ts_r|rl_{i,r})$ is the probability that ts is equal to the RSS value at the TP given the location of the i th RP in the condition of the r th pose.

All AP signals in the environment are assumed to be independent, so the overall likelihood of the i th RP location can be calculated by directly multiplying the likelihoods of all AP signals given the location of the i th RP in the condition of the r th pose:

$$p(ts_r|rl_{i,r}) = p(ts_{1,r}|rl_{i,r}) \times p(ts_{2,r}|rl_{i,r}) \times \dots \times p(ts_{m,r}|rl_{i,r}), \quad (11)$$

where m is the number of APs, and $ts_{j,r}$ means the average RSS from the j th AP in the condition of the r th pose ($j = 1, 2, \dots, m$).

The likelihood of the RSS value of each RP is assumed to be a Gaussian distribution in Equation (12). The means and standard deviations of each RP can be calculated from all RSS data at each RP.

$$p(ts_{j,r}|rl_{i,r}) = \frac{1}{\sqrt{2\pi} \times \delta} \exp \left[-\frac{(ts_{j,r} - \mu)^2}{2\delta^2} \right], \quad (12)$$

In this paper, the PRASVM algorithm is proposed on the basis of pose recognition. The poses of mobile users are recognised by using the SVM algorithm, and grouped online RSS data are classified by RP classifiers after pose recognition. During positioning, the RP classifier is trained by using the following classification function in the condition of the r th pose:

$$f(x_r) = \text{sign} \left(\sum_{i=1}^n \alpha_i y_{i,r} K(x, x_{i,r}) + b \right), \quad (13)$$

where α_i is a Lagrange multiplier. $K(x, x_{i,r})$ is the kernel function in the condition of the r th pose.

This study uses RBF [39,43] for the classification of grouped online RSS data:

$$K(x, x_{i,r}) = \exp \left(-\gamma \| x - x_{i,r} \|^2 \right), \quad (14)$$

where γ is the kernel parameter in the RBF.

Through pose recognition and RP classification, all online RSS data at a TP are classified as corresponding RPs, and the coordinates of the TP is estimated using online RSS data corresponding to the coordinates of the RPs:

$$(\bar{x}, \bar{y}) = \frac{1}{K} \sum_{i=1}^m \sum_{r=1}^n (x_{i,r}, y_{i,r}), \quad (15)$$

where $(x_{i,r}, y_{i,r})$ represents the classified online RSS data corresponding to the coordinates of the i th RP in the condition of the r th pose ($i = 1, 2, \dots, m$). m is the number of RPs, n is the number of poses, and K is the number of collected online RSS data at a TP.

2.2.4. Implementation of the PRASVM Algorithm

During pose recognition, the RSS data of different poses at each RP are used as RSS fingerprint data. The built-in sensors in the smartphone are used to record the sensor data of different poses during fingerprint data collection. The sensor data are used as pose training samples for pose recognition. The RSS fingerprint data correspond to the sensor data. RSS and sensor data of different poses in the database are extracted to train a pose classifier and RP classifiers of different poses. A pose classifier is trained by using the pose training samples with different poses in the offline phase.

In this PRASVM algorithm, firstly, the poses of sensor data at a TP are recognised by the pose classifier in the online phase. The poses of online RSS data at a TP are recognised according to the correspondence between the RSS and sensor data in time, and RSS data measured online are grouped with different poses. Then, the online RSS data from each group at a TP are reclassified as corresponding RPs by the RP classifiers of the corresponding poses. Finally, the coordinates of the TP are estimated using online RSS data from each group corresponding to the coordinates of the RPs, and the positioning results of different groups are combined to obtain the final TP location. The entire process of the PRASVM algorithm is shown in Figure 6.

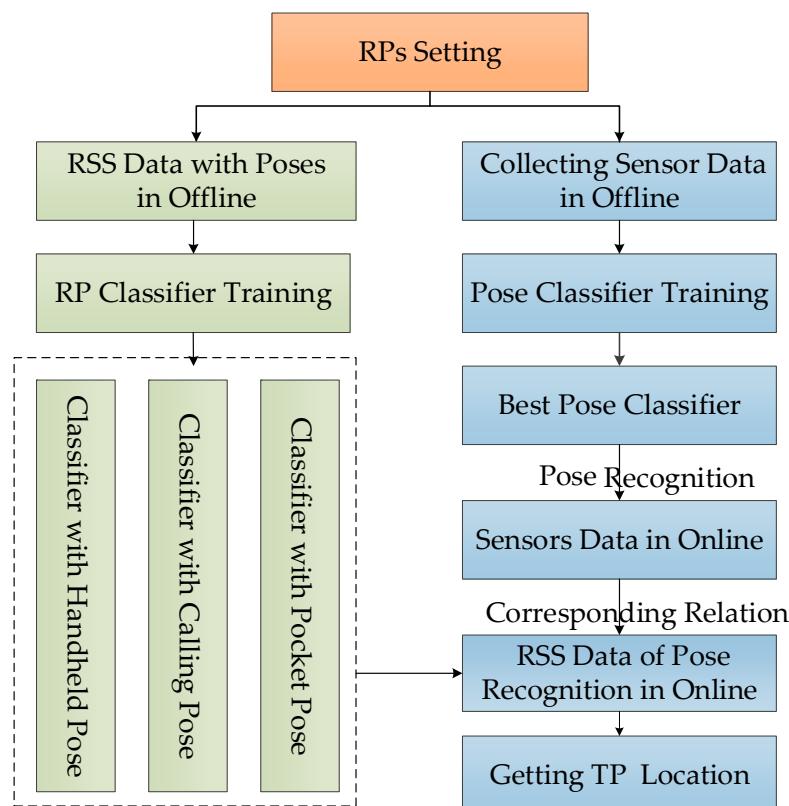


Figure 6. Implementation process of the PRASVM algorithm.

3. Results

In this study, two experiments are conducted to evaluate the performance of the PRASVM algorithm. The performance of pose recognition is evaluated in Section 3.2. Some influencing environmental factors, such as pillars and frequency bands, are considered in Section 3.3. As seen by

the analysis of different positioning methods in Section 3.4, the positioning accuracy of the PRASVM algorithm is remarkably improved.

3.1. Experimental Setup

Two experiments are carried out in the indoor environment to test the proposed algorithms. The first experiment is conducted in an office building of the School of Geodesy and Geomatics, Wuhan University on 15 October 2018. The office area is $6.4\text{ m} \times 12.8\text{ m}$. Many office separators and computers are found in the office. The locations of APs and RPs are shown in Figure 7. Nine APs are distributed in the office as transmitters, and their locations are unknown. There are two models of transmitters. The first is a TL-WR842N wireless router, which has two transmitting antennas and only transmits 2.4 GHz band signals. The second model is a TL-WDR5620 wireless router, which can transmit two kinds of frequency band signals, namely, 2.4 GHz and 5 GHz. The TL-WDR5620 wireless router has three transmitting antennas. Three APs are found on the sides of the pillars in the office. The green circles represent the RPs. The location of each RP is a known position and is represented by its own fingerprint. The blue symbol indicates the TL-WR842N wireless router. The distance between the two consecutive RPs is 1.6 m. The red diamond symbols represent the TPs. A total of 28 RPs and 15 TPs are found in the office.

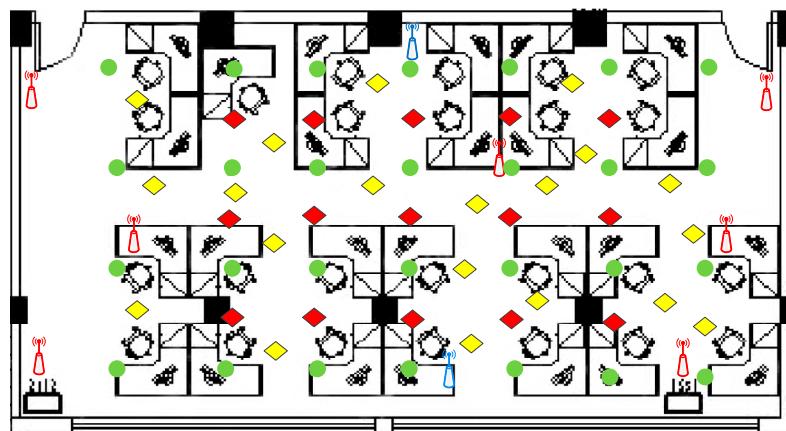


Figure 7. Experiment settings for indoor positioning. Circle symbols represent RPs and red diamond symbols represent TPs. The yellow diamond symbols represent the redeployed TPs in the office.

On the software side, the RSS and sensor data are collected by a smartphone application developed by our research team. A MI4 smartphone (Android 4.4-based MIUI6 operation system) is used to collect RSS data at a rate of 1 Hz. The RSS data are collected at each RP with 12 poses, and each pose data are collected with an MI4 smartphone at each RP for 25 s (25 RSS data). Therefore, the 300 RSS data are collected at each RP in total. At the same time, the corresponding sensor data are collected by the smartphone in different poses for each RP, and the sampling rate of the sensors is set to 10 Hz. In the online phase, the RSS data measured online and sensor data are collected at each TP for 2 min (120 RSS data) in different poses (calling, handheld, and pocket) or in the same pose. To distinguish this TP data from the data of added TPs (yellow diamond symbols in Figure 7) in the office, data of these TPs are called test data 1. The data collector don't have location change at each TP, and other people can move normally (working or talking) in this realistic experiment environment. The collected RSS data in a TP are not grouped, and the pose can arbitrarily change during data collection. To ensure consistency, all the data are collected using the same smartphone. For convenience, an independent coordinate system is established in the office for positioning.

In order to assess the positioning performance of the proposed algorithm in a time-varying environment, we reset 20 TPs in the office and conducted experiment on 22 February 2019. The yellow diamond symbols represent the new TPs in Figure 7. The distribution of TPs is irregular. The used

smartphone, RSS data collector, time interval, and AP and RP layout are the same as in the 15 October 2018 experiment. The data collection scheme and fingerprint data are also the same as in the previous office experiment. Data of new TPs are called test data 2. Without specifying that the experimental results are obtained by test data 2, the experimental results are obtained by test data 1 in the office experiment.

The selected environment is relatively complex, and several pillars are found throughout office. Figure 8 shows part of the experimental scene and some representative poses of a user.



Figure 8. Some photos of the experimental scene: (a) sides of pillars; (b) most of the room; (c) other sides of the pillars, and some representative poses of the mobile user: (d) handheld pose; (e) pocket pose; (f) calling pose.

The second experiment is conducted in a lecture hall at Wuhan University to further verify the positioning performance of the PRASVM algorithm. The lecture hall's area is $10\text{ m} \times 16\text{ m}$. The locations of APs and RPs are shown in Figure 9. The triangle symbols represent the APs. Six APs are distributed in the room. The pentagram symbols represent the RPs, and the circle symbols present the TPs. All APs can transmit two kinds of frequency band signals. The distance between the RPs is 2 m. A MI4 smartphone is used to collect RSS data at a rate of 1 Hz, and the sampling rate of the sensors is set to 10 Hz. In the online phase, the RSS data are collected at each RP with 12 poses, and each pose is collected at each RP for 25 s (25 RSS data). At the same time, the corresponding sensor data are collected in different poses. In the online phase, the RSS data measured online and sensor data are collected at each TP for 2 min (120 RSS data) in different poses. The pose can arbitrarily change throughout data collection. Figure 10 shows photos of the real experimental environment.

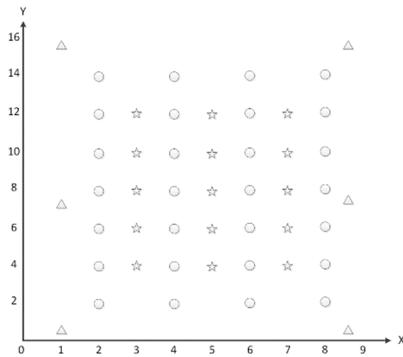


Figure 9. The distribution of indoor APs, RPs, and TPs.



Figure 10. Some photos of the lecture hall: (a) front of the lecture hall; (b) middle of the lecture hall; (c) side of the lecture hall.

3.2. Performance Evaluations of Pose Recognition

In order to evaluate the accuracy of pose recognition, we select several metrics in this paper. The SVM algorithm is adopted to recognise the poses. The accuracy of pose recognition reaches 99.98%.

3.2.1. Evaluation Metrics of Poses Recognition

Several metrics are used in pose recognition to evaluate the classification performance of the SVM. The parameters of these metrics are obtained through a confusion matrix.

As shown in Table 1, TP represents that the target pose is correctly classified as the target pose. FP represents that other poses are mistakenly classified as the target pose. FN represents that the target pose is mistakenly classified as other poses. TN represents that the other poses are correctly classified as other poses. The metrics of SVM performance are precision, recall, accuracy and F_1 score, which are calculated using Equations (16)–(19).

$$Precision = \frac{TP}{TP + FP}, \quad (16)$$

$$Recall = \frac{TP}{TP + FN}, \quad (17)$$

$$Accuracy = \frac{TP + FN}{TP + FN + FP + TN}, \quad (18)$$

$$F_\partial = \frac{(\partial^2 + 1)Precision \times Recall}{\partial^2(Precision + Recall)}, \quad (19)$$

Table 1. Parameter introduction of confusion matrix.

Actual Pose	Estimated Pose	
	Target Pose	Other Pose
Target Pose	True Positive (TP)	False Negative (FN)
Other Pose	False Positive (FP)	True Negative (TN)

Precision is the percentage of correctly classified target pose samples among all samples that are classified as target pose samples. *Recall* is the percentage of correctly classified target pose samples among all target pose samples. *Accuracy* is the percentage of correctly classified total pose samples among all pose samples. F_β is the comprehensive evaluation metric of precision and recall. F_1 ($\beta = 1$) is the harmonic mean of precision and recall.

3.2.2. Pose Recognition Results

The SVM algorithm is adopted to recognise the poses. SVM classification is divided into three steps: data collection, classifier training and testing, and classification prediction. Firstly, all the samples are separated into two parts, namely, training samples (80% of the total training samples) and test samples (20% of the total training samples). These samples are used to train the classifier and validate the performance of the trained classifier, respectively. Then, 80% of RP training samples are selected to train the RP classifier, and 20% of test samples are used to evaluate the trained RP classifier. Finally, the classes of the RSS data measured online are predicted by the RP classifier.

The sensor data are collected at each RP with 12 poses, and each pose data are collected with an MI4 smartphone at each RP for 25 s. 3000 sensor data are collected at each RP in all. A total of 84,000 sensor data are collected as pose training samples at 28 RPs in the office experiment. The sampling rate of the sensors is set to 10 Hz. The best classifier is found by ten-fold cross-validation and evaluation of classifier performance. The recognition results of the 12 poses is shown in Figure 11. The accuracy of pose recognition reaches 99.98%.

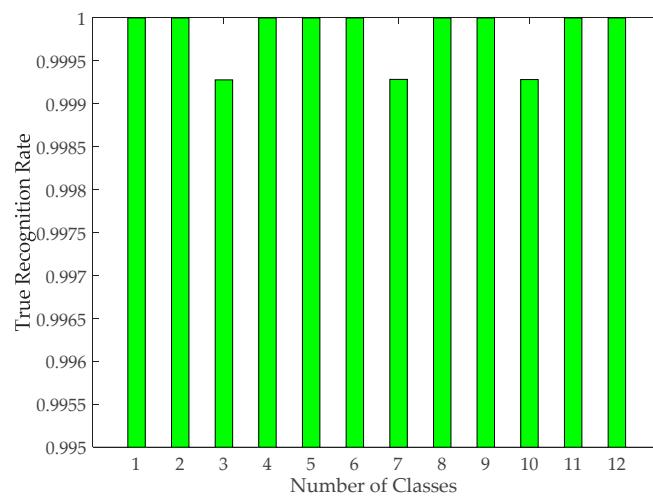


Figure 11. The true recognition rate of different poses. There are 12 kinds of poses in the figure. The true recognition rate represents the percentage of target pose samples of correctly recognised among all target pose samples.

In order to analyse the influence of different poses on RP classification results, 8400 RP training samples are obtained in the office experiment, which contain 12 pose data. A pose data has approximately 700 samples in the fingerprint database. The classification results of 28 RPs without

considering poses are shown with blue bars in Figure 12; the classification accuracy is 74.38%. The classification results of 28 RPs when considering poses are shown with green bars in Figure 12; the classification accuracy is 97.16%. The classification accuracy is the average classification accuracy across poses. The classification accuracy of RP training samples is noticeably improved when considering poses.

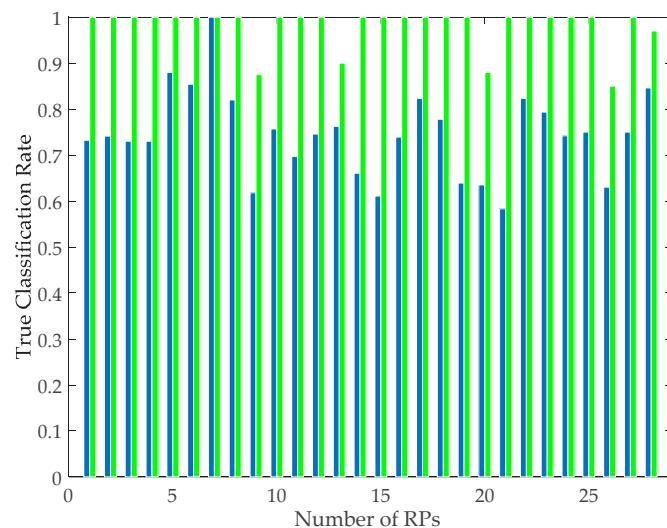


Figure 12. The true classification rate of different RPs. The blue bars represent true classification rates of RP training samples at each RP when not considering poses; the green bars represent the average true classification rates of RP training samples at each RP across poses.

In order to verify the performance of different classification methods, we compare several common classification methods. The RP training samples are obtained from our office experiment, and they cover all poses. Under the condition of considering poses, the classification performance of different methods is of great concern to us. Therefore, classification accuracies of different methods are compared when considering poses. The classification accuracies of all RPs are the average classification accuracy across poses. The classification results of 28 RPs are shown in Figure 13. The classification accuracies of the SVM, decision tree (DT), Gaussian naïve Bayesian (GNB), linear discriminant analysis (LDA), logistic regression (LR), and multi-layer perceptron (MLP), are 97.16%, 24.47%, 90.21%, 55.89%, 93.71%, and 39.18%, respectively. The classification of the SVM algorithm is the best, and the GNB and LR algorithms also have fair accuracies; meanwhile, the other methods have poor classification accuracies.

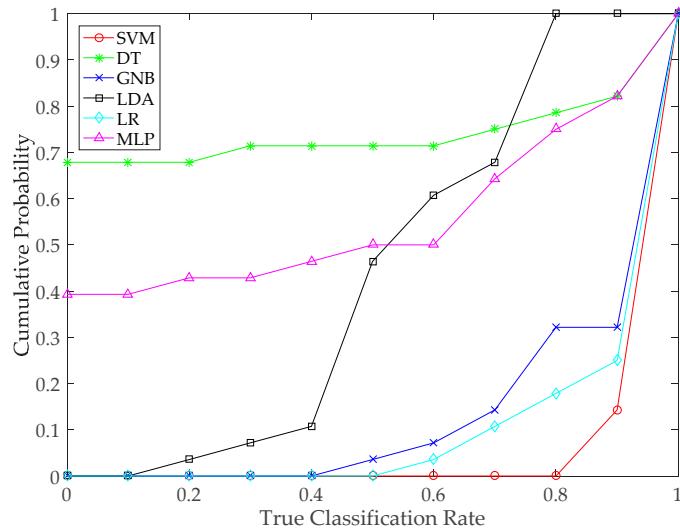


Figure 13. The cumulative probability of mean true classification rate of each RP with different classification methods. The mean true classification rate represents average of true classification rate of each RP across pose.

3.3. Analysis of Positioning Performance When Considering Pillars of the Room and Wi-Fi Frequency Bands

To evaluate the positioning performance, the mean absolute error (MAE), standard deviation (STD), maximum absolute error (MAXAE), minimum absolute error (MINAE), and median absolute error (MEDAE) are selected as accuracy evaluation metrics. The definitions of these evaluation parameters are listed in Table 2.

Table 2. Evaluation and their definitions.

Evaluation Parameters	Definition
Estimated TP Location	TL
True TP Location	TL_{truth}
Absolute Error	$ TL - TL_{truth} $
Maximum Absolute Error	$\max(TL - TL_{truth})$
Minimum Absolute Error	$\min(TL - TL_{truth})$
Median Absolute Error	$\text{median}(TL - TL_{truth})$
Mean Absolute Error	$\frac{1}{N} \sum_{k=1}^N (TL^k - TL_{truth}^k)$
Standard Deviation	$\sqrt{\frac{1}{N} \sum_{k=1}^N (TL^k - \bar{TL})}$

The positioning results with different conditions are analysed to verify the influence of pillars and frequency bands on positioning. We select only all RSS data with a pose (e.g., calling pose) as the fingerprint database at each RP in the office experiment, and re-collect RSS data at each TP with the same pose. Three APs are found on the sides of the pillars in the office. The positioning effect of $K = 4$ is the optimal result in the WKNN algorithm. The WKNN algorithm ($K = 4$) is used for positioning in this office experiment. The positioning performances under different conditions are shown in Table 3. We can see that the positioning result when using all APs outperforms that when using most APs by 0.2186 m with the two frequency bands. The positioning performance of the data containing the two kinds of frequency band outperforms that of the data containing a frequency band. Thus, the positioning performance is optimal when the algorithm uses data of all APs and frequency bands.

Table 3. Positioning performances with the WKNN algorithm when considering pillars and frequency bands.

Selected AP	Frequency Band (GHz)	MAXAE (m)	MINAE (m)	MEDAE (m)	MAE (m)	STD(m)
All APs ¹	5	2.9648	0.3487	0.7782	0.9774	0.6395
	2.4	2.1249	0.1418	0.8161	0.9401	0.5417
	2.4 and 5	1.9084	0.3707	0.7848	0.8743	0.4126
Side APs	2.4 and 5	1.9359	0.2701	1.0719	1.0929	0.4954

¹ The office is divided into two parts by pillars, one part being larger than the other. Side APs mean only APs of the larger part of the office.

3.4. Performance Evaluations of Positioning Algorithms with Pose Recognition

3.4.1. Performance of Pose Recognition-Assisted Conventional Positioning Methods

In this paper, we consider that the pose recognition is used to assist conventional positioning methods. For the office experiment, the positioning results with and without pose recognition are shown in Table 4. MAXAE represents the maximum absolute error among the mean absolute error of each TP, and other errors are also obtained by the mean absolute error of each TP in Table 4. The mean positioning error with pose recognition is remarkably smaller than that without pose recognition: 0.4303 m with KNN, 0.4566 m with WKNN and 0.2543 m with Bayesian algorithm. For the methods with pose recognition, the mean positioning error of the nearest neighbour method outperforms that of the probability method by 0.8938 m when using the WKNN algorithm and by 0.8406 m when using the KNN algorithm. WKNN, KNN, and Bayesian algorithms are used with different methods to assess positioning performance. The evaluation results are reported in Figure 14.

Table 4. Positioning performances with and without pose recognition-assisted conventional positioning methods.

	Method	MAXAE (m)	MINAE (m)	MEDAE (m)	MAE (m)	STD(m)
No Pose Recognition	KNN	3.2274	0.5604	1.5576	1.7270	0.7666
	WKNN	2.8413	0.5148	1.3928	1.7001	0.7399
	Bayesian	4.9406	0.8000	2.0601	2.3916	1.1886
Pose Recognition	KNN	2.6629	0.1600	1.2625	1.2967	0.7912
	WKNN	2.8585	0.0211	1.1054	1.2435	0.8521
	Bayesian	4.3081	0.7999	1.7889	2.1373	1.2308

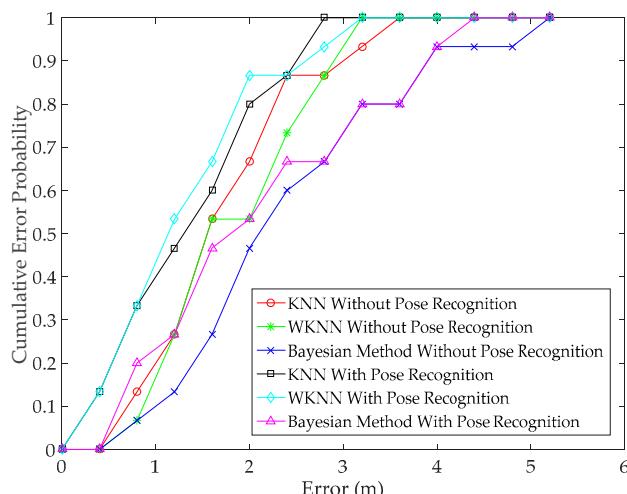


Figure 14. The positioning error with and without pose recognition-assisted conventional positioning methods. The cumulative error probability represents cumulative probability of mean absolute error of each TP. The mean absolute error represents average of absolute error of each TP across pose.

In the case of pose recognition, the error distances of the methods are within 1.2 m, with probabilities of 53.33%, 46.67%, and 26.67%, within 2 m with probabilities of 86.67%, 80%, and 53.33%, and within 3.2 m with probabilities of 100%, 100%, and 80%, according to the error analyses using WKNN, KNN, and Bayesian algorithms, respectively. As shown in Figure 14 and Table 4, the positioning accuracy is remarkably improved when pose recognition is used. In this case, poses of data can be completely matched in the offline and online phases. The positioning accuracy is remarkably reduced in other cases. Moreover, the method utilizing the WKNN algorithm is the most optimal among the pose recognition-assisted conventional positioning methods.

3.4.2. Performance of the PRASVM Algorithm

To compare the positioning performance of the proposed PRASVM algorithm, the positioning results from the office experiment with and without pose recognition are shown in Table 5 and Figure 15a. MAXAE represents the maximum absolute error among the mean absolute error of each TP, and other error are also obtained by the mean absolute error of each TP in Table 5. As shown in Table 5, the mean positioning error with the PRASVM algorithm is remarkably smaller than that of the SVM algorithm (by 2.2774 m). The positioning accuracy of the WKNN method with pose recognition is improved by 26.86% compared to that without pose recognition. Furthermore, the PRASVM algorithm outperforms the WKNN method with pose recognition by 0.4323 m, and the positioning accuracy of the PRASVM algorithm is 34.76% greater than that of the WKNN method with pose recognition. As shown in Figure 15a, the error distance is within 0.4 m with probabilities of 13.33% and 13.33%, within 0.8 m with probabilities of 66.67% and 33.33%, and within 1.6 m with probabilities of 100% and 66.67% according to the error analyses using PRASVM and WKNN with pose recognition. Table 5 and Figure 15a show that the positioning performances of the proposed PRASVM algorithm and the pose recognition-assisted WKNN algorithm are better than that of WKNN algorithm without pose recognition. The proposed PRASVM algorithm performs best among all positioning methods.

Table 5. Positioning performances with and without pose recognition-assisted WKNN and PRASVM algorithm in the office and lecture hall experiments.

		Method	MAXAE (m)	MINAE (m)	MEDAE (m)	MAE (m)	STD (m)
Office Experiment	Test data 1	No Pose Recognition	WKNN	2.8413	0.5148	1.3928	1.7001
			SVM	4.6999	1.6081	2.8000	3.0886
	Test data 2	Pose Recognition	WKNN	2.8585	0.0211	1.1054	1.2435
			PRASVM	1.5807	0.1182	0.5808	0.8112
Lecture hall Experiment	Test data 1	No Pose Recognition	WKNN	2.9385	0.8626	1.8522	1.8538
			SVM	4.8902	1.9408	3.2304	3.2165
	Test data 2	Pose Recognition	WKNN	2.8930	0.4288	1.4322	1.4505
			PRASVM	2.0785	0.2340	1.0304	0.9765
	Test data 1	No Pose Recognition	WKNN	2.7168	0.6887	1.2940	1.4083
			SVM	3.9844	1.3188	2.5343	2.5412
	Test data 2	Pose Recognition	WKNN	2.3537	0.5404	1.1164	1.2861
			PRASVM	1.7888	0.3577	0.9523	1.0049

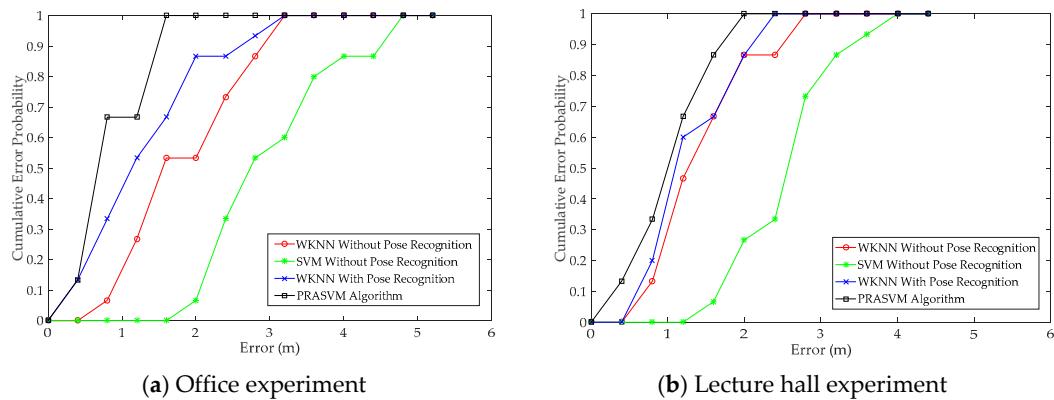


Figure 15. Positioning error with and without pose recognition-assisted WKNN and PRASVM algorithms in the office (a) and lecture hall (b) experiment. The cumulative error probability represents cumulative probability of mean absolute error of each TP. The mean absolute error represents average of absolute error of each TP across pose.

In order to assess the positioning result of the proposed PRASVM algorithm in a time-varying environment, we obtain test data 2 after relocating some TPs in the office environment. As shown in Table 5, the mean positioning error of the proposed PRASVM algorithm outperform the conventional WKNN algorithm by 0.8773 m, outperform the SVM algorithm by 2.24 m, and outperform the pose recognition-assisted WKNN algorithm by 0.474 m. Whether we use test data 1 to locate or test data 2 to locate, the proposed PRASVM method achieves the best positioning results. Through a comparative analysis of positioning results from test data 1 and test data 2, we can see that the mean positioning error of the PRASVM algorithm with test data 2 increases by 0.1653 m from that of test data 1. Although there is about a four-month time interval between the collection of test data 1 and test data 2, the office environment (e.g., location of office desks, presence of appliances, and office structure) did not change much during this period of time.

In order to further test the performance of the PRASVM algorithm, the lecture hall experiment is designed. The positioning results with and without pose recognition are shown in Table 5 and Figure 15b. As shown in Table 5, the mean positioning error with pose recognition is remarkably smaller than that without pose recognition. The positioning accuracy of the WKNN method with pose recognition is improved by 8.68% compared to that without pose recognition. The mean positioning error of the PRASVM algorithm outperforms the WKNN algorithm with pose recognition by 0.2812 m, and the positioning accuracy of the PRASVM algorithm is 21.86% greater than that of the WKNN method with pose recognition. As shown in Figure 15(b), the error distance is within 0.8 m with probabilities of 33.33% and 20%, within 1.6 m with probabilities of 86.67% and 66.67%, and within 2 m with probabilities of 100% and 86.67%, according to the error analyses using the PRASVM and pose recognition-assisted WKNN. Therefore, the positioning performance of the proposed PRASVM algorithm is optimal.

Through a comparative analysis of the two experiments, we can see that the positioning accuracies of WKNN and SVM algorithms without pose recognition in the lecture hall are better than those using the same algorithms in the office experiment. This is because the desks are low in the lecture hall, signal interference is reduced, and no pillar shadowing occurs. The location accuracies of PRASVM and WKNN algorithms with pose recognition in the lecture hall are lower than those obtained in the office experiment. This is because different poses having a small impact on the positioning results in the lecture hall, and the lecture hall is large, and the fingerprint interval is large in the lecture hall. Therefore, the office environment is more complex than that of the lecture hall, and the positioning accuracy has a more noticeable improvement in the PRASVM algorithm.

4. Discussion

The fingerprint method is a common method for Wi-Fi indoor positioning [26–30,44–49], and various factors affect positioning results [50,51]. In this paper, we discuss the effects of three typical user poses and body shadowing on Wi-Fi signals. Different RSS values are received under different poses and body shadowing. Therefore, we discuss the impact of different poses and body shadowing on the positioning results, and we find that these factors have a serious impact. We use the SVM algorithm [35–38,52] to recognise poses for assisted positioning. We propose a PRASVM algorithm in this paper. The positioning accuracy of the proposed PRASVM algorithm is significantly greater than that of the conventional positioning algorithm. Meanwhile, we also analyse the influence of pillars and Wi-Fi frequency bands on the positioning results. This paper uses the SVM classification method, and makes comparisons to several common classification methods [31,33,34,53]. We find that the classification of the SVM algorithm is the best, and the GNB and LR algorithms also have fair accuracies; meanwhile, other methods have poor classification accuracies.

The main idea of this paper is to improve the accuracy of fingerprint matching and positioning results. We mainly discuss the influence of poses and body shadowing on positioning results, and the improvement of the positioning algorithms. The selected experimental environment is a relatively complex office and lecture hall environment, but the area of the environment is limited. In the future, we will do experiments in more scenarios and within larger experimental areas. Users did not walk much in this experiment. In some studies, dynamic motion modes (walking and going up and down stairs) were considered in indoor positioning [51,54]. In addition, there are many other factors affecting positioning results, such as heterogeneous terminal, user characteristics (e.g., height, weight, behaviour), and indoor temperature [55]. Finding out the relationship between environmental changes and positioning results, and establishing appropriate theoretical models to fundamentally improve positioning accuracy are the focus of future research.

5. Conclusions

In this paper, the impact of user poses and body shadowing on RSS data are explored. The received RSS data experience large changes across poses and orientations. However, the current fingerprint approach only considers the spatial distribution of RSS, which impedes matching performance. This is the key factor hindering positioning accuracy improvement. To solve the impact of user pose and user body shadowing on positioning results, we proposed a PRASVM algorithm. In this method, a fingerprint database with RSS and sensor data corresponding to different poses is established, and fingerprints of different poses in the database are extracted to train a pose classifier and RP classifiers of different poses with an SVM algorithm. The accuracy of pose recognition reaches 99.98%, and the classification accuracy of RP training samples with a pose is 97.16%. The poses of RSS data measured online are recognised by the pose classifier, RSS data measured online are grouped with different poses, and grouped online RSS data match the RP classifier corresponding to the right user pose. In this method, the negative effect of user pose can be eliminated; this improves the performance of the SVM on positioning, thus allowing us to obtain better positioning results.

In this study, we find that the positioning accuracy of the proposed PRASVM algorithm is remarkably better than that of conventional positioning algorithms. The positioning accuracies of the proposed PRASVM algorithm outperform the conventional WKNN algorithm by 52.29% and 40.89%, outperform the SVM algorithm by 73.74% and 60.45%, and outperform the pose recognition-assisted WKNN algorithm by 34.76% and 21.86% in the office and lecture hall experiments, respectively. Moreover, we consider the positioning result of the proposed PRASVM algorithm in a time-varying environment.

In addition, we also find that the positioning accuracy is noticeably improved when pose recognition is used in conventional positioning methods. In the office experiment, the mean positioning error with pose recognition is smaller than that without pose recognition: 0.4303 m smaller with

the KNN algorithm, 0.4566 m smaller with the WKNN algorithm, and 0.2543 m smaller with the Bayesian algorithm.

Author Contributions: This paper is a collaborative work by all the authors. S.Z. proposed the idea, designed the experiments, performed the experiments, analyzed the data, and wrote the manuscript. J.G. and N.L. added the experiments, gave suggestions, and revised the rough draft; L.W., W.W., and K.W. assisted with certain experiments, and all authors proof-read the paper.

Funding: This research was sponsored by the National Natural Science Foundation of China (No.41474004, 41704002). The work was also supported by funds from the Key Laboratory of Precise Engineering and Industry Surveying, NASMG (PF2017-7).

Acknowledgments: We would like to thank the editor and the anonymous reviewers for their valuable comments, which greatly improved the quality of this manuscript. We also would like to acknowledge the School of Geodesy and Geomatics, Wuhan University, and the Research Center for High Accuracy Location Awareness, Wuhan University for the support in our research.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Enge, R.P.; Misra, P. Special issue on global positioning system. *Proc. IEEE* **1999**, *87*, 3–15. [[CrossRef](#)]
2. John, N.B.; Heidemann, J.; Estrin, D. GPS-less low cost outdoor positioning for very small devices. *IEEE Pers. Comm. Mag.* **2000**, *7*, 28–34.
3. Karegar, P.A. Wireless fingerprint indoor positioning using affinity propagation clustering methods. *Wirel. Netw.* **2017**, *3*, 1–9.
4. Bisio, I.; Cerruti, M.; Lavagetto, F.; Marchese, M.; Pastorino, M.; Randazzo, A. A Trainingless Wi-Fi fingerprint positioning method over mobile devices. *IEEE Antennas Wirel. Propag. Lett.* **2014**, *13*, 832–835. [[CrossRef](#)]
5. Niculescu, D.; Nath, B. Ad Hoc Positioning System (APS) Using AOA. In Proceedings of the Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies, San Francisco, CA, USA, 30 March–3 April 2003; pp. 1734–1743.
6. Li, X.; Pahlavan, K. Super-resolution TOA Estimation with diversity for indoor geolocation. *IEEE Trans. Wirel. Commun.* **2004**, *3*, 224–234. [[CrossRef](#)]
7. Amar, A.; Leus, G. A Reference-Free Time Difference of Arrival Source Positioning Using a Passive Sensor Array. In Proceedings of the IEEE Sensor Array Multichannel Signal Process, Workshop (SAM), Jerusalem, Israel, 4–7 October 2010; pp. 157–160.
8. Dakkak, M.; Nakib, A.; Daachi, B. Indoor positioning method based on RTT and AOA using coordinates clustering. *Comput. Netw.* **2011**, *55*, 1794–1803. [[CrossRef](#)]
9. Li, Z.; Liu, J.; Wang, Z.; Chen, R.Z. A Novel Fingerprinting Method of WiFi Indoor Positioning Based on Weibull Signal Model. In Proceedings of the China Satellite Navigation Conference (CSNC) 2018 Proceedings, Harbin, China, 23–25 May 2018; pp. 297–309.
10. Tran, D.N.; Phan, D.D. Human Activities Recognition in Android Smartphone Using Support Vector Machine. In Proceedings of the 2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS), Bangkok, Thailand, 25–27 January 2016; pp. 64–68.
11. Yang, J. Toward Physical Activity Diary: Motion Recognition Using Simple Acceleration Features with Mobile Phones. In Proceedings of the 1st International Workshop on Interactive Multimedia for Consumer Electronics, Beijing, China, 23–23 October 2009; pp. 1–10.
12. Pei, L.; Liu, J.B.; Guinness, R. Using LS-SVM Based Motion recognition for smartphone indoor wireless positioning. *Sensors* **2012**, *12*, 6155–6175. [[CrossRef](#)]
13. Tran, D.A.; Nguyen, T. Positioning in wireless sensor networks based on support vector machines. *IEEE Trans. Parallel Dis. Syst.* **2008**, *19*, 981–994. [[CrossRef](#)]
14. Wu, Z.L.; Li, C.H.; Ng, K.Y. Location estimation via support vector regression. *IEEE Trans. Mob. Comput.* **2007**, *6*, 311–321. [[CrossRef](#)]
15. Brunato, M.; Battiti, R. Statistical learning theory for location fingerprint in wireless LANs. *Comput. Netw.* **2005**, *47*, 825–845. [[CrossRef](#)]
16. Sang, N.; Yuan, X.Z.; Zhou, R. Wi-Fi indoor location based on SVM classification and regression. *Comput. Appl.* **2014**, *31*, 1820–1823.

17. Yu, F.; Jiang, M.H.; Liang, J. An indoor positioning of Wi-Fi based on support vector machines. *Adv. Mater. Res.* **2014**, *4*, 926–930.
18. Khalajmehrabadi, A.; Gatsis, N.; Akopian, D. Modern WLAN Fingerprinting indoor positioning methods and deployment challenges. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 1974–2002. [[CrossRef](#)]
19. Lu, W.; Teng, J.; Zhou, Q. Stress Prediction for distributed structural health monitoring using existing measurements and pattern recognition. *Sensors* **2018**, *18*, 419. [[CrossRef](#)]
20. Bovio, I.; Monaco, E.; Arnese, M. Damage detection and health monitoring based on vibration measurements and recognition algorithms in real-scale aeronautical structural components. *Key Eng. Mater.* **2003**, *519*–526. [[CrossRef](#)]
21. Barnachon, M.; Bouakaz, S.; Boufama, B. Ongoing human action recognition with motion capture. *Pattern Recognit.* **2014**, *47*, 238–247. [[CrossRef](#)]
22. Zhu, Y.; Xu, G.; Kriegman, D.J. A Real-time approach to the spotting, representation, and recognition of hand gestures for human-computer interaction. *Comput. Vis. Image Underst.* **2002**, *85*, 189–208. [[CrossRef](#)]
23. Wang, G.; Li, Q.; Wang, L. Impact of Sliding window length in indoor human motion modes and pose pattern recognition based on smartphone sensors. *Sensors* **2018**, *18*, 1965. [[CrossRef](#)]
24. Guo, G.; Chen, R.; Ye, F. A Pose awareness solution for estimating pedestrian walking speed. *Sensors* **2019**, *11*, 55. [[CrossRef](#)]
25. Zhang, H.; Yuan, W.; Shen, Q. A handheld inertial pedestrian navigation system with accurate step modes and device poses recognition. *IEEE Sens. J.* **2015**, *15*, 1421–1429. [[CrossRef](#)]
26. Oh, J.; Kim, J. Adaptive K-nearest neighbour algorithm for WiFi fingerprint positioning. *ICT Express* **2018**, *4*, 91–94. [[CrossRef](#)]
27. Liu, H.H. The Quick radio fingerprint collection method for a Wi-Fi-based indoor positioning system. *Mob. Netw. Appl.* **2015**, *22*, 1–11.
28. Caso, G.; Nardis, L.D. Virtual and oriented Wi-Fi Fingerprint indoor positioning based on multi-wall multi-floor propagation models. *Mob. Netw. Appl.* **2017**, *22*, 825–833. [[CrossRef](#)]
29. Ma, R.; Guo, Q.; Hu, C. An Improved Wi-Fi indoor positioning algorithm by weighted fusion. *Sensors* **2015**, *15*, 21824–21843. [[CrossRef](#)] [[PubMed](#)]
30. Nuñomaganda, M.A.; Herrrerarivas, H.; Torreshuitzil, C. On-device learning of indoor location for Wi-Fi fingerprint method. *Sensors* **2018**, *18*, 1–14.
31. Hand, D.J.; Yu, K. Idiot's Bayes—Not so stupid after all? *Int. Stat. Rev.* **2010**, *69*, 385–398.
32. Witten, I.H.; Frank, E. Data Mining: Practical machine learning tools and techniques with java implementations. *ACM Sigmod Rec.* **2011**, *31*, 76–77. [[CrossRef](#)]
33. Sun, L.; Zhang, D.; Li, B.; Guo, B.; Li, S. Activity Recognition on an Accelerometer Embedded Mobile Phone with Varying Positions and Orientations. In Proceedings of the 7th International Conference on Ubiquitous Intelligence and Computing, Xi'an, China, 26–29 October 2010; pp. 548–562.
34. Khan, A.M.; Lee, Y.K.; Lee, S.; Kim, T.S. Human Activity Recognition via an Accelerometer-Enabled-Smartphone Using Kernel Discriminant Analysis. In Proceedings of the 5th International Conference on Future Information Technology, Busan, Korea, 21–23 May 2010; pp. 1–6.
35. Wu, C.L.; Fu, L.C.; Lian, F.L. WLAN Location Determination in E-home via Support Vector Classification. In Proceedings of the IEEE International Conference on Networking, Sensing and Control, Taipei, Taiwan, 21–23 March 2004; pp. 1026–1031.
36. Li, N.; Yang, D.; Jiang, L. Combined use of FSR sensor array and SVM classifier for finger motion recognition based on pressure distribution map. *J. Bionic Eng.* **2012**, *9*, 39–47. [[CrossRef](#)]
37. Li, J.; Hu, G.Q.; Zhou, Y.H.; Zou, C.; Peng, W. A Temperature compensation method for piezo-resistive pressure sensor utilizing chaotic ions motion algorithm optimized hybrid kernel LSSVM. *Sensors* **2016**, *16*, 1707. [[CrossRef](#)]
38. Selvakumari, N.A.S.; Radha, V. A Voice Activity Detector Using SVM and Naïve Bayes Classification Algorithm. In Proceedings of the 2017 International Conference on Signal Processing and Communication (ICSPC), Coimbatore, India, 28–29 July 2017; pp. 1–6.
39. Burges, J.C. A Tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.* **1998**, *2*, 121–167. [[CrossRef](#)]

40. Singh, R.; Macchi, L.; Regazzoni, C.S. A Statistical Modelling Based Location Determination Method Using Fusion Technique in WLAN. In Proceedings of the IEEE International Workshop on Wireless Ad-hoc Networks (IWWAN), London, UK, 23–26 May 2005; pp. 1–5.
41. Ge, X.; Qu, Z. Optimization WI-FI Indoor Positioning KNN Algorithm Location-based Fingerprint. In Proceedings of the 2016 7th IEEE International Conference on Software Engineering and Service Science, Beijing, China, 26–28 August 2016; pp. 135–137.
42. Ma, J.; Li, X.; Tao, X.P.; Lu, J. Cluster filtered KNN: A WLAN-based Indoor Positioning Scheme. In Proceedings of the 2008 International Symposium on a World of Wireless, Mobile and Multimedia Networks, Newport Beach, CA, USA, 23–26 June 2008; pp. 1–8.
43. Liu, Q.; Chen, C.; Zhang, Y. Feature selection for support vector machines with RBF Kernel. *Artif. Intell. Rev.* **2011**, *36*, 99–115. [[CrossRef](#)]
44. Sánchez-Rodríguez, D.; Alonso-González, I.; Ley-Bosch, C. A Simple indoor localization methodology for fast building classification models based on fingerprints. *Electronics* **2019**, *8*, 103. [[CrossRef](#)]
45. Han, C.; Tan, Q.; Sun, L. CSI Frequency domain fingerprint-based passive indoor human detection. *Information* **2018**, *9*, 95. [[CrossRef](#)]
46. Haider, A.; Wei, Y.; Liu, S. Pre-and post-processing algorithms with deep learning classifier for Wi-Fi fingerprint-based indoor positioning. *Electronics* **2019**, *8*, 195. [[CrossRef](#)]
47. Santos, R.; Barandas, M.; Leonardo, R. Fingerprints and floor plans construction for indoor localisation based on crowdsourcing. *Sensors* **2019**, *19*, 919. [[CrossRef](#)] [[PubMed](#)]
48. Tan, J.; Fan, X.; Wang, S. Optimization-based Wi-Fi radio map construction for indoor positioning using only smart phones. *Sensors* **2018**, *18*, 3095. [[CrossRef](#)]
49. Seong, J.H.; Seo, D.H. Real-time recursive fingerprint radio map creation algorithm combining Wi-Fi and geomagnetism. *Sensors* **2018**, *18*, 3390. [[CrossRef](#)]
50. Garcia-Villalonga, S.; Perez-Navarro, A. Influence of Human Absorption of Wi-Fi Signal in Indoor Positioning with Wi-Fi Fingerprinting. In Proceedings of the International Conference on Indoor Positioning & Indoor Navigation, Banff, AB, Canada, 13–16 October 2015; pp. 1–10.
51. He, S.; Chan, S.H.G. Wi-Fi Fingerprint-based indoor positioning: recent advances and comparisons. *IEEE Commun. Surv. Tutor.* **2015**. [[CrossRef](#)]
52. Wei, Y.; Hwang, S.H.; Lee, S.M. IoT-Aided Fingerprint Indoor Positioning Using Support Vector Classification. In Proceedings of the International Conference on Information and Communication Technology Convergence (ICTC), Jeju, South Korea, 17–19 October 2018; pp. 973–975.
53. Alshamaa, D.; Mourad-Chehade, F.; Honeine, P. A Weighted Kernel-based Hierarchical Classification Method for Zoning of Sensors in Indoor Wireless Networks. In Proceedings of the 2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Kalamata, Greece, 25–28 June 2018; pp. 1–5.
54. Chen, Y.C.; Chiang, J.R.; Huang, P.; Tsui, A.W. Sensor assisted Wi-Fi indoor location system for adapting to environmental dynamics. In Proceedings of the 2005 8th ACM international symposium on modeling, analysis and simulation of wireless and mobile systems, Montréal, QC, Canada, 10–13 October 2005; pp. 118–125.
55. Xia, H.; Wang, X.; Qiao, Y. Using multiple barometers to detect the floor location of smart phones with built-in barometric sensors for indoor positioning. *Sensors* **2015**, *15*, 7857–7877. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).