# Fingerprint-Based Indoor Positioning System Using Visible Light Communication—A Novel Method for Multipath Reflections

**Huy Q. Tran** [ID] **and Cheolkeun Ha** *

Robotics and Mechatronics Lab, University of Ulsan, Ulsan 44610, Korea; quanghuybkdn@gmail.com
* Correspondence: cheolkeun@gmail.com

check for updates

**Abstract:** A highly accurate indoor positioning under the effect of multipath reflections has been a prominent challenge for recent research. This paper proposes a novel indoor visible light communication (VLC) positioning model by connecting k-nearest neighbors (kNN) and random forest (RF) algorithms for reflective environments, namely, kNN-RF. In this fingerprint-based model, we first adopt kNN as a powerful solution to expand the number of input features for RF. Next, the importance rate of these features is ranked and the least effective one(s) may be removed to reduce the computation effort. Next, the training process using the RF algorithm is conducted. Finally, the estimation process is utilized to discover the final estimated position. Our simulation results show that this new approach improved the positioning accuracy, making it nearly five times better than other popular kNN algorithms.

---

## 1. Introduction

In the last few decades, the global positioning system (GPS) has been widely employed in positioning and navigation because of its high reliability and accuracy, and because of its real-time positioning capability [1]. In indoor environments, however, this power substantially declines and it may even be impossible to accurately and continuously locate the coordinates of the followed object, largely due to building structures, building materials, and other obstructions [2].

Recently, with the rapid development of high-intensity, solid-state white LEDs, the idea of using LEDs for both illumination and data transmission has become a promising trend in contemporary applications [3], particularly the growing demand for positioning techniques in indoor environments where people spend approximately 90% of their time during the day [4]. Also, LED lighting technology is gradually replacing conventional indoor lighting methods due to its safety, economy, and environmental friendliness [3,5]. Additionally, visible light communication (VLC) has recently emerged as one of the most promising candidates for fifth-generation (5G) mobile communications [1]. Several other types of wireless signals have been applied in indoor positioning fields, such as WiFi, RFID, Bluetooth, and Zigbee, and each of them has unique strengths, although their positioning accuracy remains somewhat limited [2].

In this paper, we propose a novel kNN-RF model based upon the traditional fingerprint technique. Unlike the existing received signal strengths (RSS)-based algorithms, in this scheme k-nearest neighbors (kNN) is responsible for enlarging the number of inputs for the next random forest (RF) model, which needs enough data to avoid overfitting and to obtain more accurate training results. Not all data generated from kNN are used by the RF model; rather, the data are sent through an evaluation

and collection process which ranks the importance of each feature and eliminates the useless ones. This process reduces the computational time and increases the capacity of the RF training operation. Another noteworthy point is that our proposed model incorporates the strong effects of multipath reflection, ambient light, and other noises. Our approach produced a remarkable reduction in the error range from 94.2 cm by the conventional kNN method to 19.3 cm by the kNN-RF method.

The main contributions of this paper are as follows:

- Most of recent VLC-based research have ignored the impact of reflection (see Section 2). However, this noise exerts a very bad influence on the positioning accuracy, especially in areas outside the room's center. In this paper, the effects of multipath reflection are thoroughly explored at the highest reflective rate.
- We propose a novel machine learning-based indoor positioning solution. This method first uses kNN as a powerful tool to expand the number of features for RF in the next step. Then the position of the mobile object can be located based on the estimation process in the RF algorithm which collects all the features directly from both the LED lights and the kNN output signals.

The remainder of this paper is organized as follows. Section 2 summarizes and analyzes some related works. Section 3 introduces the system model with and without multipath reflections, and then presents our proposed algorithm in detail. Section 4 discusses our simulation results and performance evaluations, then some proposed real applications are presented in Section 5, and Section 6 offers our discussion and conclusion.

## 2. Related Work

To improve accuracy, several VLC-based positioning techniques have been proposed [6], including the angle of arrival (AOA) [7], time of arrival (TOA), time difference of arrival (TDOA) [8,9], phase difference of arrival (PDOA) [10,11], and RSS [12,13]. Each of them has distinct limitations: AOA helps considerably to reduce the positioning errors but requires high computational complexity; TOA and TDOA need to ensure compliance with strict requirements for synchronization; PDOA requires a lower computational time than TDOA but needs a local oscillator; and RSS-based positioning accuracy is considered moderately low because of the effect of ambient light, noises, and the tilt angle of the photo-detector (PD) [10].

In addition to diverse types of signal-based positioning methods, a few positioning algorithms have been developed to optimize the quality of indoor positioning systems (IPS). In Reference [12], the authors proposed an improved kNN model by applying weights to more accurately determine the Euclidean distances.

Another approach, which compares machine learning and other conventional RSS-based solutions, was presented in Reference [13]. However, the general focus of these articles ignores multipath reflections, even though they actually exist and have an enormous impact on the system quality. To elucidate these impacts, the authors in Reference [14] proved that the influence of multipath reflections is much weaker and seems to be almost unaffected in the central region of a given room. In contrast, the reflection in other areas (i.e., the corners and edges) is more powerful than in the case of no reflection. In the given results, the positioning errors in the corner unexpectedly increased and reached roughly 2 m.

In Reference [15] the authors used the trilateration method to show that the positioning errors increased by 76 times, from 2 cm using line-of-sight (LOS) compared to 152 cm with non-line-of sight (NLOS). In Reference [16], the authors proposed a new approach to reduce the positioning error by using calibration methods, including selecting the strongest LED signal and decreasing the distance between the LED bulbs. As a result, there was a considerable improvement in the positioning accuracy, and the errors in the whole room were at 0.3 m and 0.25 m as a result of LED selection and changing the LED distance, respectively. However, the corresponding errors outside the central region were still high, at 0.47 m and 0.36 m. Additionally, a change in the number of LEDs and the distance between

them may lead to changes in the structure of the room and illumination intensity. These analyses demonstrate that interference exerts a very negative influence on the positioning accuracy of LOS and NLOS, especially in corners and near edge areas. To reduce such detrimental effects, the authors of Reference [17] presented a new way to improve the positioning accuracy in a room corner, based on tilting the image sensor. They did not consider the multipath reflections and still achieved low accuracy at some points, and found it difficult to determine the optimal tilt angle.

## 3. System Model and Proposed Positioning Method

In this section, we first investigate the system model used for the proposed solution. This model consists of a directed optical channel and a non-directed optical channel. In each part, we focus on calculating the received optical power, then the total received optical power from both channels are shown. Next, a typical system configuration and our proposed kNN-RF algorithm are introduced in detail.

### 3.1. System Model

#### 3.1.1. Directed Optical Channel

The directed optical channel, or LOS path, is shown in Figure 1. This channel is of vital importance in almost every indoor VLC positioning system because it receives most of the total optical power. As illustrated in Figure 2, the maximum received power is 0.92 mW and the minimum is 0.34 mW. The power distribution in the central region appears quite uniform, and the illumination intensity in this area is stronger than other areas, especially in the corners.
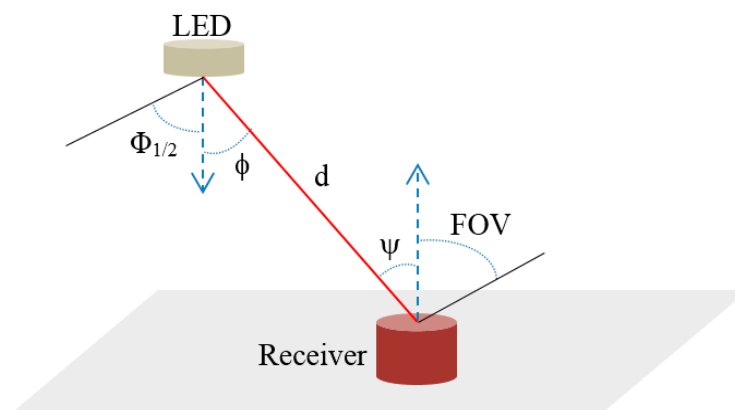


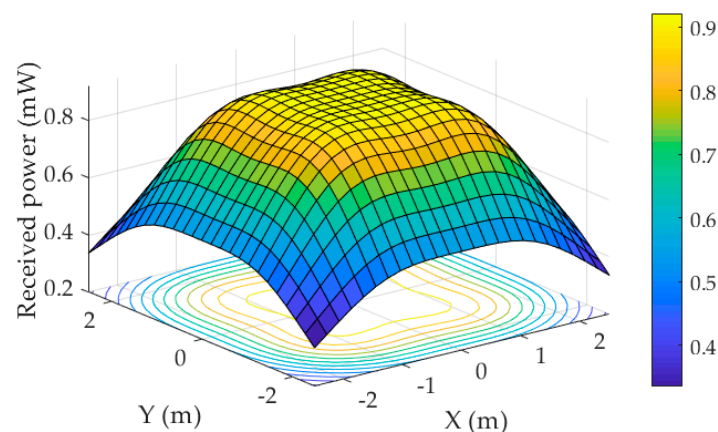**Figure 1.** Directed channel model with one LED.



**Figure 2.** Received power distribution with directed channel.

To compute the LOS power, we assume that the distance and the angle from the transmitter to the receiver are $d$ and $\varphi$, respectively. The irradiance angle $\varphi$ and incidence angle $\Psi$ are of equal value because we accept that both the LED and the photosensitive area of the receiver are parallel to the floor (see Figure 1).

To effectively detect the presence of visible light from the LED, the signal collection area on the optical sensor plays an important role. This area has a great influence over the positioning correctness and can be calculated as [18]:

$$A_{esc} = \begin{cases} A_{PD} T_s(\Psi) g(\Psi) \cos(\Psi), & 0 \leq \Psi \leq \Psi_c, \\ 0, & \Psi > \Psi_c \end{cases} \tag{1}$$

where $A_{PD}$ is the active detector area of the PD; $T_s(\Psi)$ is the gain of the optical filter; $g(\Psi)$ is the gain of the optical concentrator; and $\Psi_c$ is the receiver field of view (FOV).

The irradiance (W/cm$^2$) is given by Reference [18]:

$$I_s(d, \phi) = \frac{P_t R_o(\phi)}{d^2} \tag{2}$$

where $P_t$ is the transmitted power; $d$ is the transmission distance from the LED to the PD; and $R_o(\phi)$ is the Lambertian radiant intensity, written as [19]:

$$R_o(\phi) = \left[ \frac{n+1}{2\pi} \right] \cos^n(\phi) \tag{3}$$

where $n$ is the Lambertian order and is determined from [19]:

$$\Phi_{1/2} = \cos^{-1}\left(\frac{1}{2}\right)^{\frac{1}{n}} \tag{4}$$

Hence,

$$n = \frac{-\ln(2)}{\ln(\cos \Phi_{1/2})} \tag{5}$$

The optical power gathered from the LED lamps can be expressed as [18]:

$$P_{LOS} = I_s(d, \phi) A_{esc} \tag{6}$$

Finally, the total directed optical power at the receiver is as follows [5]:

$$P_{LOS} = \begin{cases} P_t \frac{(n+1) A_{PD}}{2\pi d^2} \cos^n(\phi) T_s(\Psi) g(\Psi) \cos(\Psi), & 0 \leq \Psi \leq \Psi_c \\ 0, & \Psi > \Psi_c \end{cases} \tag{7}$$

### 3.1.2. Non-Directed Optical Channel

To evaluate the impact of multipath noises on the system performance, we conducted a comprehensive study of the effects of the first reflection from the four walls around the room. In practice, the reflection of visible light always exists in an indoor environment, especially in a narrow space with walls, ceiling, floor, and some pieces of furniture. As can be seen in Figure 3, the active area of the receiver is facing upwards and is parallel with the floor. Hence, it is impossible to have a sense of the first signal reflected from the floor. In addition, the direction of the LED bulb is facing downwards, and we also set the half-power semi-angle of each bulb at 60° and the distance from the light source to the ceiling at 0.7 m. This meant that the amount of light which the ceiling directly received from the LED bulb was completely restricted. Furthermore, any additional reflections could be neglected because they would have no significant effect compared to the total noise level power [20]. For these

reasons, we chose the first reflection from the walls and the LOS signal as the major channel model in our work. Figure 4 illustrates that the power distribution in the corner and near the wall was higher than in the central area.
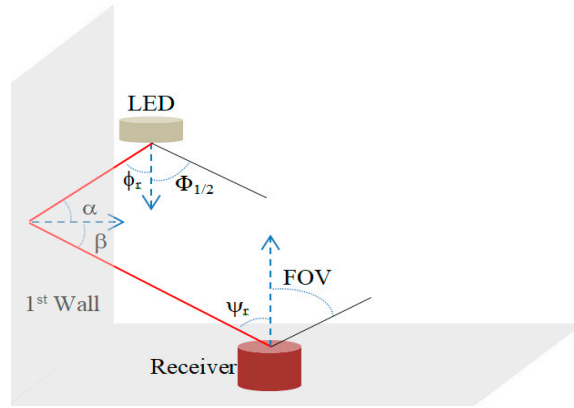


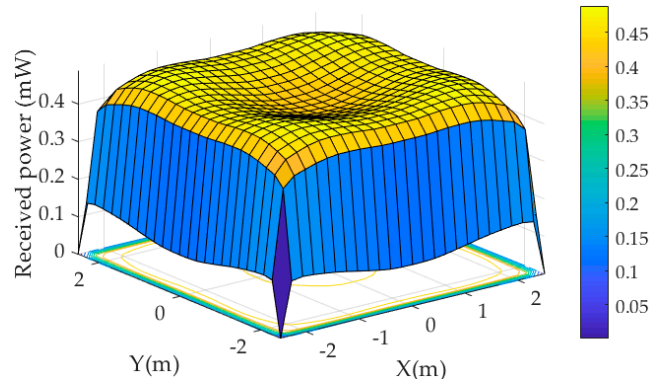**Figure 3.** Non-directed channel model with one LED.



**Figure 4.** Received power distribution with non-directed channel.

To compute the received power of the first reflection link, we compute the received power at a reflective point on the wall as follows:

$$P_{R\_wall} = I_s(d_1, \phi) A_{eff\_wall} \tag{8}$$

where $I_s(d_1, \phi)$ can be obtained from Equation (2); $d_1$ is the distance between an LED and a reflective point; and $A_{eff\_wall}$ is the effective signal collection area on the wall and is given as follows:

$$A_{eff\_wall} = dA_{wall} \cos(\alpha) \tag{9}$$

where $dA_{wall}$ is the surface element on the wall and $\alpha$ is the angle of irradiance from an LED to the reflective point.

Based on the cosine law and the inverse square law, the transmitted power from a reflective point is then given:

$$P_{T\_wall} = \frac{P_{r\_wall}\rho}{d_2^2} \cos(\beta) \tag{10}$$

where $d_2$, $\beta$ are the distance and the angle, respectively, from a reflective point and the receiver, and $\rho$ is the reflectance factor.

Substituting Equations (8) and (9) into (10), we have:

$$P_{T\_wall} = P_t \frac{m+1}{2\pi d_1^2 d_2^2} \rho \cos^n(\phi) dA_{wall} \cos(\alpha) \cos(\beta) \tag{11}$$

From here, the PD uses $P_{T\_wall}$ as the input signal from the non-direct link, and the final power $P_{Diff}$ that the sensors collect can be calculated as:

$$P_{Diff} = P_{T\_wall} A \cos(\Psi_r)\, Ts(\Psi_r) g(\Psi_r) \tag{12}$$

where $\Psi_r$ is the incidence angle of the light from the wall.

Finally, we adopt the non-directed optical power as follows [5]:

$$P_{Diff} = P_t \frac{A(n+1)}{2\pi d_1^2 d_2^2} \rho \cos^n(\phi) dA_{wall} \cos(\alpha)\cos(\beta)\cos(\Psi_r)Ts(\Psi_r)g(\Psi_r),\ 0 \le \Psi_r \le \Psi_c \tag{13}$$

### 3.1.3. Overall Optical Channel

In this paper, the overall channel is considered as the connection between the LOS path and the diffuse reflection path. The purpose of this combination was to further investigate the undesirable effects of reflection on the system performance and to make our simulation results more reliable and practical. Furthermore, as previously discussed in Sections 3.1.1 and 3.1.2, we simply focused on the first reflection path from the four walls around the empty room and thus, the total received optical power from the four LED groups is the sum of the directed and non-directed optical power (see Figure 5) given by:

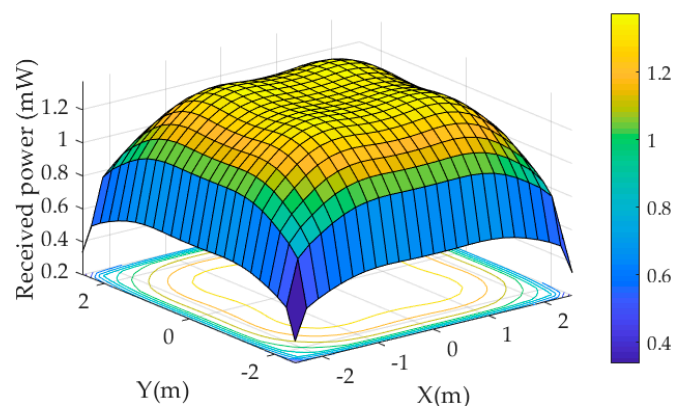$$P_r = P_{Los} + P_{Diff} \tag{14}$$



**Figure 5.** Received power distribution with overall channel.

*3.2. Proposed Positioning Method*

### 3.2.1. System Configuration

An overview of a typical room model is shown in Figure 6 [14], where its length, width, and height are fixed at 5 m, 5 m, and 3 m, respectively. We assume the LED bulbs are installed on an imaginary plane at a height of 2.3 m from the floor. Four LED bulbs are used in this model, and each of them has 10 W of transmitted optical power and a half-power semi-angle of exactly 60°. In our system, the receiver moving around the 25-m$^2$ floor has a PD active area of 1 cm$^2$ and is parallel with the floor. Other technical parameters related to the transmitters, receiver, and multipath reflection noise are summarized in detail in Table 1.
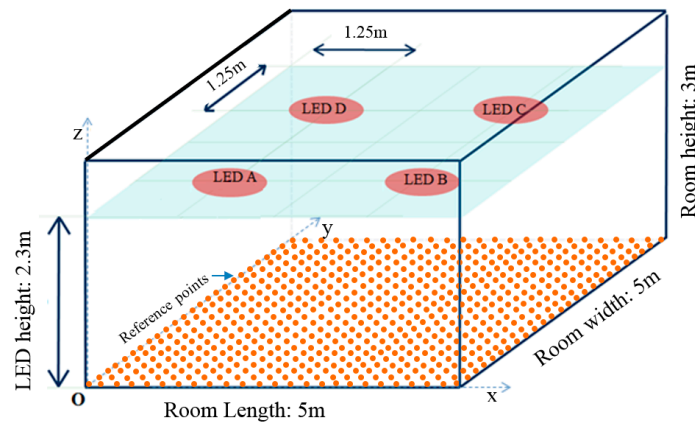
**Figure 6.** System configuration.

**Table 1.** Model technical parameters.

| Parameter | Value |
| --- | --- |
| Room dimension (L × W × H) | 5m × 5m × 3m |
| Reflection coefficient | 0.8 |
| **Transmitters** | |
| LED power per group | 10 W |
| No. of LED groups | 4 |
| Wave length | 420 nm |
| Elevation: | $-90°$ |
| LED position (x, y, z) (m) | A (1.25, 1.25, 2.3) <br> B (3.75, 1.25, 2.3) <br> C (3.75, 3.75, 2.3) <br> D (1.25, 3.75, 2.3) |
| Half power semi-angle | 60° |
| **Receiver** | |
| PD active area | 1 cm$^2$ |
| FOV | 60° |
| Gain of optical filter | 1 |
| Receiver sensitivity | $-30$ dBm |
| $T_s(\Psi)$ | 1 |
| $g(\Psi)$ | 1.5 |

### 3.2.2. kNN-RF Algorithm

In the following section, we present a novel fingerprint location method known as kNN-RF, a combination of kNN and RF. In particular, the training data are expanded by kNN, and this leads to a significant improvement in the positioning accuracy after the RF training process.

The proposed RSS-based technique can be divided into two main phases— the offline phase and the online phase (Figure 7).

The offline phase plays a key role since it takes most of the program execution time. This phase consists of three separable basic steps. First, based on the area of the floor (see Figure 6), we prepared a feasible number of reference points (25 × 25) with the distance between these points of 20.833 cm and the optical receiver on a mobile object then began to collect all the RSS data from the four LED bulbs at each predetermined point. Second, all the sampling data gathered from the previous step were stored in memory.
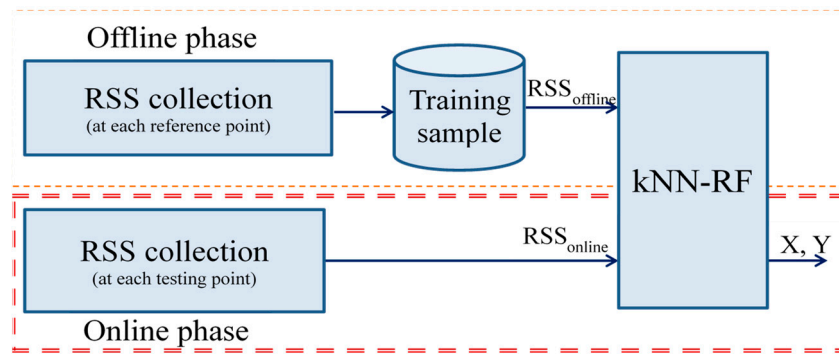
**Figure 7.** Block diagram of the proposed system.

Finally, all these data were sent to the kNN-RF model. This process (Figure 8) is described in the following details:

(i)　Feature expanding process using kNN: First using kNN is an important step to increase the number of variables for the RF model, by finding the closest points to each reference point. kNN is a very simple machine learning algorithm and is widely used in the current IPSs. The main objective of this method is to find the Euclidean distances based on all of the reference points collected in the offline phase. For the number of Euclidean distances, it is entirely possible to rely on the value of $k$ ($k > 1$) [12]. Specifically, the Euclidean distances are computed as follows:

$$DE = \sqrt{\sum_{i=1}^{4} \left( RSS_{on\_i} - RSS_{off\_i} \right)^2} \qquad (15)$$

where $i$ is the number of LED bulbs (i.e., $i = 4$); $RSS_{on\_i}$ are the RSS values collected in the online phase; and $RSS_{off\_i}$ are the RSS values stored in the offline phase.
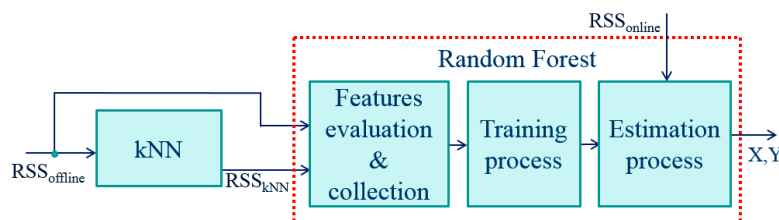


**Figure 8.** kNN-RF model.

After determining the Euclidean distances, the coordinates of each closest point are easily identified. From this location, their RSS values are also inferred. To determine one of the two inputs of the next RF block, the major operation of this step is to identify the nearest RSS of each collected fingerprint based on the Euclidean distance (15). The number of RSS outputs from kNN always depends on how many $k$ closest points are used. In our proposed system, the number of LED bulbs is fixed at four; hence each $k$ value corresponds to four RSS values. In summary, RSS data transferred to the next step from the kNN model are illustrated in Table 2, with $k = 2$ and given as:

$$RSS_{kNN} = LED_{no}kF_{off} \qquad (16)$$

where $RSS_{kNN}$ is the total number of RSS values after processing using kNN; $LED_{no}$ is the number of LEDs (here, $LED_{no} = 4$); $k$ is the number of nearest points; and $F_{off}$ is the number of fingerprints (here, $F_{off} = 625$).

(ii)　Feature evaluation and collection process: The RSS signals from both the kNN output and the sampling data stored in the memory are transferred to the first function block in the RF model,

namely, feature evaluation and collection. As discussed earlier, the number of features from kNN varies depending upon the number of $k$-nearest points, and they obviously exist in direct proportion. In the RF algorithm we can improve the performance of the training process and reduce the computation cost using the feature importance ranking [21,22].

**Table 2.** Received optical power using kNN algorithm with two nearest points (Unit: W).

| No. | RSS (online) | | | | RSS ($k = 1$) | | | | RSS ($k = 2$) | | | |
| | LED A | LED B | LED C | LED D | LED A | LED B | LED C | LED D | LEDA | LED B | LED C | LED D |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | $7.13 \times 10^{-5}$ | $1.16 \times 10^{-5}$ | $1.16 \times 10^{-5}$ | $4.52 \times 10^{-6}$ | $7.13 \times 10^{-5}$ | $1.16 \times 10^{-5}$ | $1.16 \times 10^{-5}$ | $4.52 \times 10^{-6}$ | $10.3 \times 10^{-5}$ | $1.64 \times 10^{-5}$ | $2.34 \times 10^{-5}$ | $9.24 \times 10^{-5}$ |
| 2 | $10.3 \times 10^{-5}$ | $1.64 \times 10^{-5}$ | $2.34 \times 10^{-5}$ | $9.24 \times 10^{-6}$ | $10.3 \times 10^{-5}$ | $1.64 \times 10^{-5}$ | $2.34 \times 10^{-5}$ | $9.24 \times 10^{-6}$ | $10.3 \times 10^{-5}$ | $2.34 \times 10^{-5}$ | $1.64 \times 10^{-5}$ | $9.24 \times 10^{-6}$ |
| 3 | $11.4 \times 10^{-5}$ | $1.70 \times 10^{-5}$ | $2.41 \times 10^{-5}$ | $9.02 \times 10^{-6}$ | $11.4 \times 10^{-5}$ | $1.70 \times 10^{-5}$ | $2.41 \times 10^{-5}$ | $9.02 \times 10^{-6}$ | $12.2 \times 10^{-5}$ | $1.73 \times 10^{-5}$ | $2.54 \times 10^{-5}$ | $8.92 \times 10^{-6}$ |
| 4 | $12.2 \times 10^{-5}$ | $1.73 \times 10^{-5}$ | $2.54 \times 10^{-5}$ | $8.92 \times 10^{-6}$ | $12.2 \times 10^{-5}$ | $1.73 \times 10^{-5}$ | $2.54 \times 10^{-5}$ | $8.92 \times 10^{-6}$ | $12.7 \times 10^{-5}$ | $1.75 \times 10^{-5}$ | $2.78 \times 10^{-5}$ | $9.02 \times 10^{-6}$ |
| 5 | $12.7 \times 10^{-5}$ | $1.75 \times 10^{-5}$ | $2.78 \times 10^{-5}$ | $9.02 \times 10^{-6}$ | $12.7 \times 10^{-5}$ | $1.75 \times 10^{-5}$ | $2.78 \times 10^{-5}$ | $9.02 \times 10^{-6}$ | $13.0 \times 10^{-5}$ | $1.75 \times 10^{-5}$ | $3.12 \times 10^{-5}$ | $9.30 \times 10^{-6}$ |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 621 | $9.02 \times 10^{-6}$ | $2.78 \times 10^{-5}$ | $1.75 \times 10^{-5}$ | $12.7 \times 10^{-5}$ | $9.02 \times 10^{-6}$ | $2.78 \times 10^{-5}$ | $1.75 \times 10^{-5}$ | $12.7 \times 10^{-5}$ | $9.30 \times 10^{-6}$ | $3.12 \times 10^{-5}$ | $1.75 \times 10^{-5}$ | $13.0 \times 10^{-5}$ |
| 622 | $8.92 \times 10^{-6}$ | $2.54 \times 10^{-5}$ | $1.73 \times 10^{-5}$ | $12.2 \times 10^{-5}$ | $8.92 \times 10^{-6}$ | $2.54 \times 10^{-5}$ | $1.73 \times 10^{-5}$ | $12.2 \times 10^{-5}$ | $9.02 \times 10^{-6}$ | $2.78 \times 10^{-5}$ | $1.75 \times 10^{-5}$ | $12.7 \times 10^{-5}$ |
| 623 | $9.02 \times 10^{-6}$ | $2.41 \times 10^{-5}$ | $1.70 \times 10^{-5}$ | $11.4 \times 10^{-5}$ | $9.02 \times 10^{-6}$ | $2.41 \times 10^{-5}$ | $1.70 \times 10^{-5}$ | $11.4 \times 10^{-5}$ | $8.92 \times 10^{-6}$ | $2.54 \times 10^{-5}$ | $1.73 \times 10^{-5}$ | $12.2 \times 10^{-5}$ |
| 624 | $9.24 \times 10^{-6}$ | $2.34 \times 10^{-5}$ | $1.64 \times 10^{-5}$ | $10.3 \times 10^{-5}$ | $9.24 \times 10^{-6}$ | $2.34 \times 10^{-5}$ | $1.64 \times 10^{-5}$ | $10.3 \times 10^{-5}$ | $9.24 \times 10^{-6}$ | $1.64 \times 10^{-5}$ | $2.34 \times 10^{-5}$ | $10.3 \times 10^{-5}$ |
| 625 | $4.52 \times 10^{-6}$ | $1.16 \times 10^{-5}$ | $1.16 \times 10^{-5}$ | $7.13 \times 10^{-5}$ | $4.52 \times 10^{-6}$ | $1.16 \times 10^{-5}$ | $1.16 \times 10^{-5}$ | $7.13 \times 10^{-5}$ | $9.24 \times 10^{-6}$ | $1.64 \times 10^{-5}$ | $2.34 \times 10^{-5}$ | $10.3 \times 10^{-5}$ |

In Figure 9 the importance level of the variables is illustrated, with $k = 2, 3, 4,$ and 5. In each case there are always differences among the feature groups (e.g., RSS_online, RSS_$k$1, RSS_$k$2, RSS_$k$3, RSS_$k$4, RSS_$k$5), and even among elements in a group. The least important value can be neglected to achieve better or at least the same positioning accuracy using much less computing time.
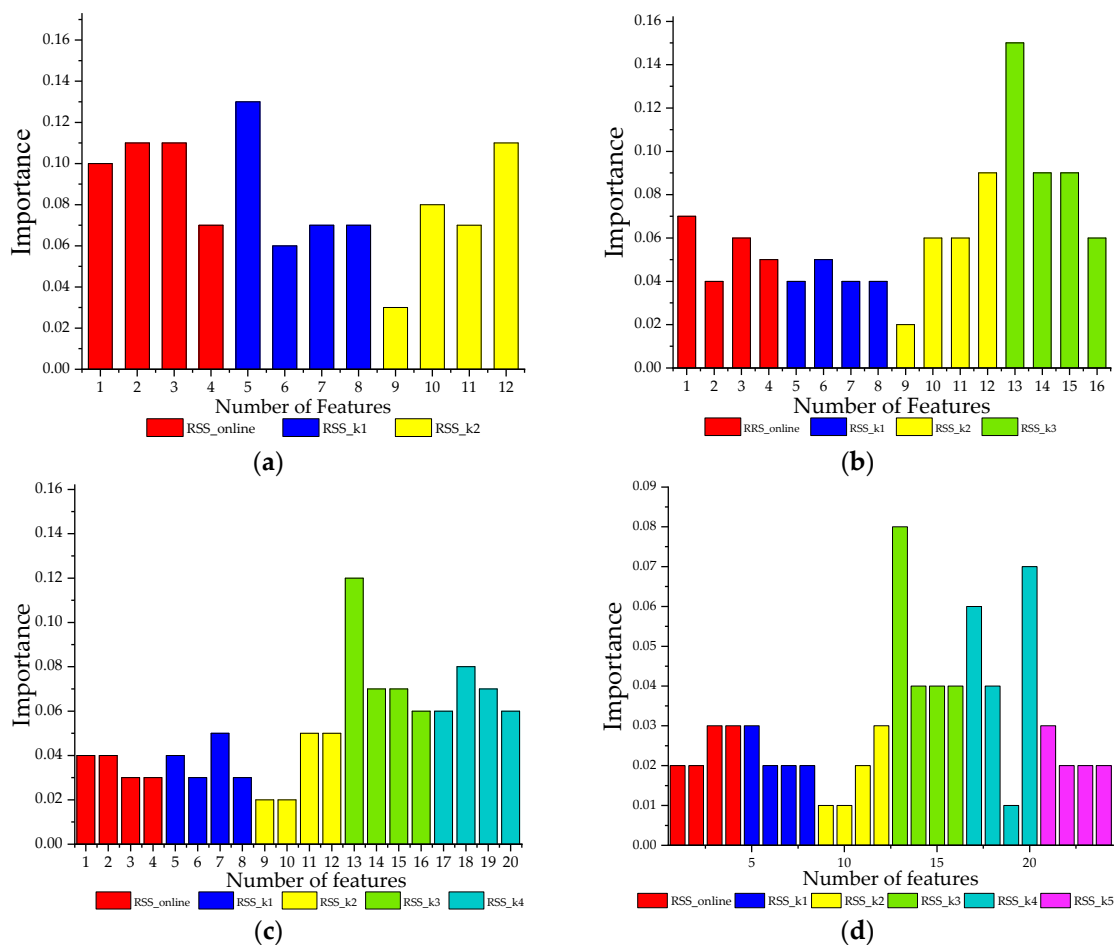


**Figure 9.** Feature importance ranking by (**a**) $k = 2$, (**b**) $k = 3$, (**c**) $k = 4$, (**d**) $k = 5$.

(iii)   The training process using RF: After evaluating and collecting the notable features, the RF training process is executed. We set the number of trees and features to 40 and 12, respectively. Similar to the previous procedure in kNN, the RF also uses the RSS data as the input signals, but the difference here is the combination of RSS data collected from the reference points and the RSS data that came from the kNN. This connection increases the number of variables, which helps reduce the chances of overfitting. However, a tremendous amount of data can lead to a very long run time.

Thus, the analysis results in Figure 9 can be used to choose, as well as remove, the appropriate features. Then, as shown in Figure 10, by neglecting some features that have an importance rate $\leq 4\%$, the CPU time is achievably enhanced, by 5.99% with $k = 2$ and by 18.3% with $k = 3$. With $k = 5$, half of all the features are skipped and the improvement increases up to 31%, while the mean errors rise only slightly by 7.45%. Therefore, the CPU time gradually decreases as more useless features are neglected.
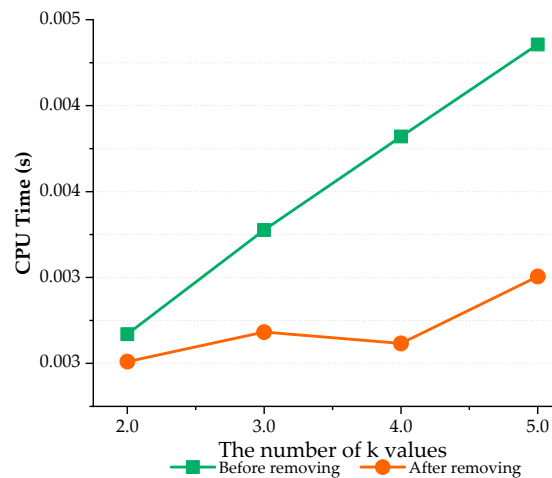


**Figure 10.** CPU time comparison before and after collection process.

In the online phase, the current RSS data from all the LED lights are sent directly to the estimation process block in the kNN-RF model (Figure 8). This block then uses these data and all the trained data in the offline phase to estimate the position of the object being considered. Estimation plays a major role in the online phase and has an effect on the final estimated position.

To evaluate the performance of our proposed algorithm, we shuffled the dataset randomly and divided them into 10 subsets. We then performed cross validations in each of them. By using this method, each subset is given the opportunity to be a testing set 1 time and be a training set 9 times. The results shown in Figure 11 demonstrate that our method is able to predict unseen data with a mean accuracy of 99.33%. More in-depth analyses of the impacts of the number of neighbors in kNN, the number of trees in RF, the receiver angle, and the performance of our method are thoroughly discussed in the next section.

| 10-fold | DATASET | | | | | | | | | | Accuracy (%) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Fold 1 | Test | Train | Train | Train | Train | Train | Train | Train | Train | Train | 99.48 |
| Fold 2 | Train | Test | Train | Train | Train | Train | Train | Train | Train | Train | 99.55 |
| Fold 3 | Train | Train | Test | Train | Train | Train | Train | Train | Train | Train | 99.60 |
| Fold 4 | Train | Train | Train | Test | Train | Train | Train | Train | Train | Train | 99.15 |
| Fold 5 | Train | Train | Train | Train | Test | Train | Train | Train | Train | Train | 99.32 |
| Fold 6 | Train | Train | Train | Train | Train | Test | Train | Train | Train | Train | 99.16 |
| Fold 7 | Train | Train | Train | Train | Train | Train | Test | Train | Train | Train | 99.39 |
| Fold 8 | Train | Train | Train | Train | Train | Train | Train | Test | Train | Train | 99.09 |
| Fold 9 | Train | Train | Train | Train | Train | Train | Train | Train | Test | Train | 99.28 |
| Fold 10 | Train | Train | Train | Train | Train | Train | Train | Train | Train | Test | 99.29 |

**Figure 11.** Cross validations.

## 4. Performance Evaluation

### 4.1. Effects of the Number of Neighbors in kNN

We examined cases with various numbers of *k* nearest points, and from Table 3 it is virtually certain that when *k* gradually increased from 2 to 8, the accuracy score showed only a minor fluctuation. However, some error types, such as the mean absolute error (MAE), mean square error (MSE), and root mean square error (RMSE) were slightly worse, corresponding to the increase in *k* values. As mentioned in Section 3.2, finding an ideal *k* value is considered an excellent way to satisfy the real-time requirements and enhance the quality of the recommended solution. In our proposed method, the number of features sent to the RF block increased four times, corresponding to each *k* value added (see Section 3.2). Therefore, in this paper, *k* is set to 2.

**Table 3.** Positioning accuracy vs. the number of k.

| The Number of *k* | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|
| **Accuracy Score (%)** | 99.08 | 99.34 | 99.06 | 99.10 | 99.18 | 99.27 | 98.98 |
| **MAE (m)** | 0.132 | 0.146 | 0.150 | 0.151 | 0.156 | 0.159 | 0.163 |
| **MSE (m)** | 0.039 | 0.040 | 0.044 | 0.044 | 0.047 | 0.048 | 0.051 |
| **RMSE (m)** | 0.196 | 0.200 | 0.208 | 0.209 | 0.216 | 0.217 | 0.225 |

### 4.2. Effects of the Number of Trees in RF

The number of trees in RF has a significant impact on not only the computational time, but also the positioning error of the system. To achieve reliable results, we randomly selected 125 testing data points (20% overall) and executed 100 consecutive repetitions to evaluate the influence of the number of trees on the system performance.

As depicted in Figure 12, the more trees that are chosen, the smaller the mean of the positioning errors. When the number of trees is more than 40, the performance continues to slightly improve, but not significantly. With 40 trees, the mean of the positioning errors was 0.193 m, and with 150 trees it was 0.189 m, a difference of only about 2.07%. From 70 trees onwards, the system became more stable, with mean errors in the range of 0.189 m to 0.191 m.
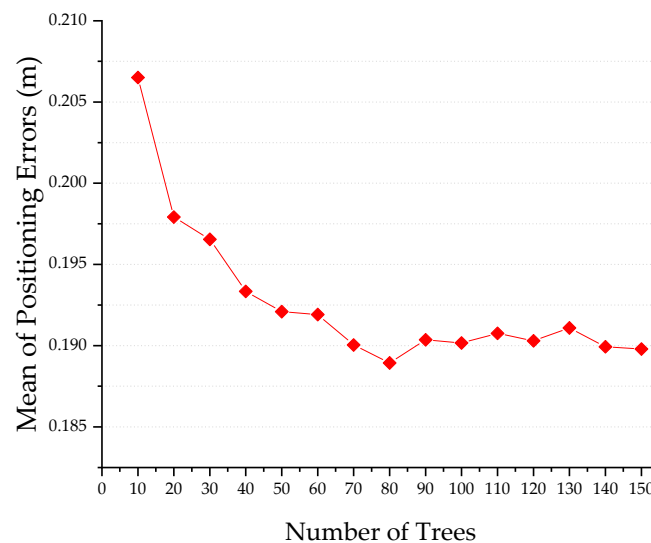


**Figure 12.** The number of trees vs. mean errors.

We also found that there were no considerable differences in the mean errors when the number of trees reached 40 and more. However, as discussed in Section 3.2, the tree numbers are directly proportional to the computational effort required (Figure 13). For instance, if 150 trees are selected

instead of 10, the computational time will increase more than 13 times; thus, our system always suffers from a heavy computation burden. Therefore, to ensure the real-time performance and acceptable positioning quality, we recommend limiting the number of potential trees to the range of 40 to 70; even smaller values can be considered when strict execution time is required.
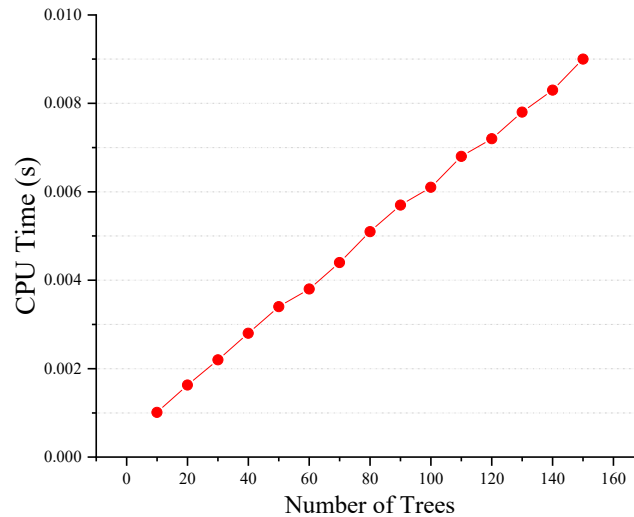


**Figure 13.** The numbers of trees vs. CPU time.

### 4.3. Effects of the Receiver Angle

In practice, the angle of the receiver may change over time and the degree of change depends on either the robustness of the mobile object equipped with a PD, the flatness of the moving surface, or the posture of the person using the PD sensor. In this section, we evaluated the effects of the receiver angle on the positioning accuracy by testing the mean of positioning errors corresponding to three separate cases: $10°$, $20°$, and $30°$. For simplicity we assumed that the receiver angle is formed by the horizontal plane of the PD and the horizontal plane of the floor. This can be seen in Figure 14, compared with the case of a fixed rotation angle ($0°$), the positioning errors in the cases of $10°$, $20°$, and $30°$ inclinations increased 3%, 28%, and 34%, respectively. This means that as we gradually increase the tilt angle of the optical receiver, the positioning accuracy changes in a deteriorating direction. However, the positioning error showed a slight increase in the range from a $0°$ to $10°$ inclination of only 3%. This range of tilt angle is also one of the most ubiquitous situations in reality. We are, therefore, confident that our proposed solution fully satisfies the actual requirements.
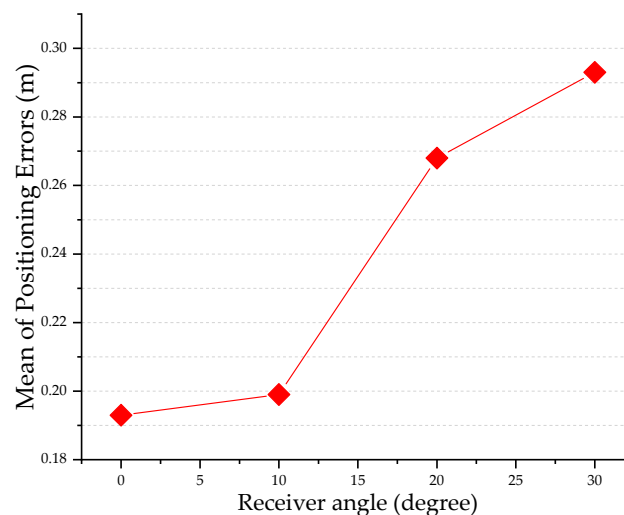


**Figure 14.** Positioning accuracy in different receiver angles.

### 4.4. Performance Evaluation

In Figure 15, the true positions (tails) and the estimated positions (heads) are used to visually illustrate the positioning accuracy of trilateration, kNN, Weighted-kNN (WkNN), and kNN-RF algorithms. It is clear that the combination of kNN and RF produced the most promising performance.
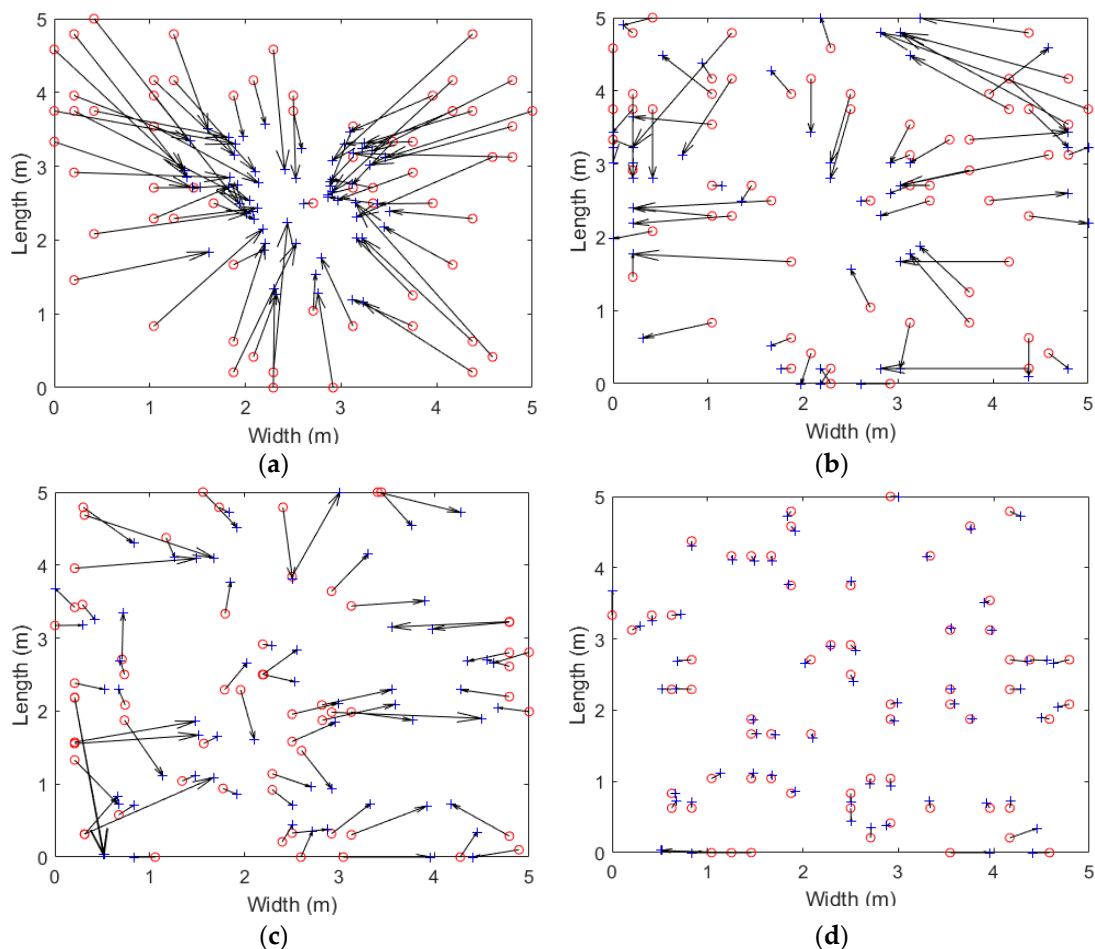


**Figure 15.** Position estimation outcome by (**a**) Trilateration, (**b**) kNN, (**c**) WkNN, and (**d**) kNN-RF.

In Reference [12], the authors demonstrated that a WkNN model in a non-reflective space was approximately 36% to 50% more accurate than the trilateration method with and without ambient light, respectively. In a reflective environment, however, the precision of the positioning system with the WkNN method became much worse, as the mean error rose from 0.031 m to 0.92 m, a 29-fold increase. In contrast to the first three algorithms (i.e., trilateration, kNN, and WkNN), the errors in the kNN-RF method were considerably lower, with an RMSE of 0.193 m (Figure 16).

As shown in Figure 17, the kNN-RF showed positive effects in specific areas of the model. By analyzing the random data with kNN and kNN-RF, we found that the highest error appeared in the corner where the impacts of noise and reflection were greatest. However, after being trained by the kNN-RF algorithm, the error dramatically declined (almost 7-fold) from 1.16 m to 0.166 m. Although the effect of the reflection in the region near the wall was higher than in the central area, and was lower than in the corner area, there was a notable (6-fold) mean error improvement. In contrast, due to the limited impact of the reflection on the central points, that area achieved the best positioning accuracy, 0.069 m, and thus gained only a slight improvement from kNN to kNN-RF.
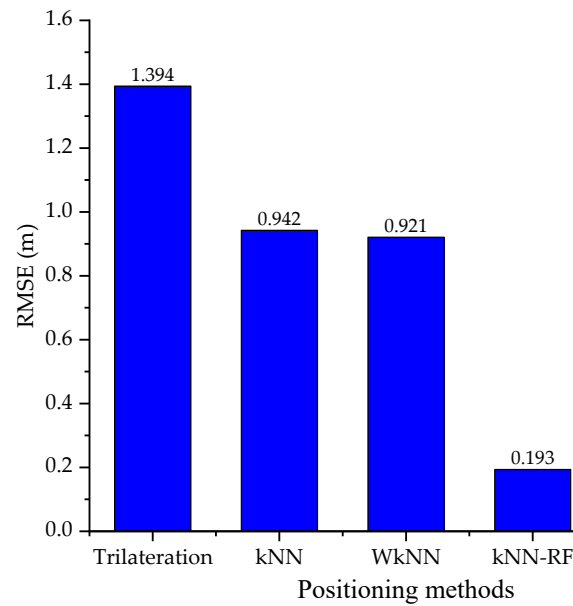
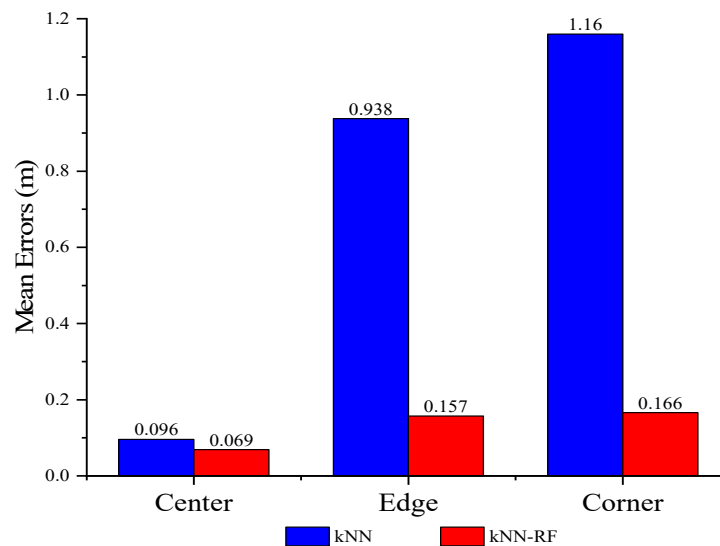**Figure 16.** Performance comparison.



**Figure 17.** Positioning accuracy in different places.

These results demonstrate that our proposed kNN-RF solution produced a significant improvement in the accuracy of the positioning system, even though the interference intensity became stronger due to reflection and other noises.

## 5. Applications

The evolution of indoor positioning system using LED light gives us an ideal opportunity to develop useful products that serve our daily lives. In particular, we can build in-house localization applications at a low cost thanks to the availability of LED bulbs. In this paper, we suggest some applications that we are pursuing.

*Assistive devices for people with disabilities*: The PD will be put on a smart cane or other assistive walking devices to localize the position of the blind users and help them reach their destination quickly and safely. Additionally, a smart wheel chair with a positioning function enables the elderly and the disabled to travel easily. This can help improve their quality of life and maintain their sense of independence.

*Indoor mobile robots*: In addition to the applications for the disabled, LED-based indoor positioning solution is also particularly useful in the field of mobile robots. We can use a mobile robot as an effective means of transport in a factory or warehouse. The robot is used to pick up, move, and place objects automatically. We can also adopt this kind of robot as a home assistant device which can make our lives easier and more comfortable.

## 6. Discussion and Conclusions

To improve the positioning accuracy in multipath reflective environments, an innovative indoor VLC positioning model is proposed in this paper, namely kNN-RF. Our simulation results show that this combination of kNN and RF algorithms was five times more precise than other kNN methods. As for conventional kNN methods, the quality of the positioning system is satisfactory if and only if there is no existence of reflection. This noise, however, always persists in the indoor environment. In this paper, we consider not only the highest reflection rate, but also the effect of ambient light, thermal noise, and shot noise. The obtained outcomes show a considerable improvement in the positioning accuracy outside of the room center, which faces the highest multipath reflections. For instance, the positioning errors represent a marked improvement from kNN to kNN-RF in the room corners and near the wall by 7-fold and 6-fold, respectively.

The potential results of the kNN-RF method present a chance of improving the performance of the positioning system despite the negative impact of noises.

To further our research, we plan to conduct a real-time, full-scale experiment to demonstrate the performance of our proposed method. We will also focus on improving the positioning accuracy by strengthening the optimization of signal pre-processing before taking the kNN-RF algorithm into account. After performing some improvements to the algorithm, we will try to apply our proposed algorithm to indoor a mobile robot. For our current study, a VLC-based positioning system is one of the main parts in the field of a multi-robot system, including collision avoidance and multi-robot coordination.

**Author Contributions:** All authors proposed the idea as well as contributed equally to writing the paper. H.Q.T. designed the proposed algorithm and performed the simulation. As the corresponding author, C.H. supervised the research and revised the paper.

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

## References

1. Kaplan, E.D.; Hegarty, C.J. *Understanding GPS: Principles and Applications*; Artech House Publishers: Norwood, MA, USA, 2006; ISBN 1-58053-894-0.
2. Zhuang, Y.; Hua, L.; Qi, L.; Yang, J.; Cao, P.; Cao, Y.; Wu, Y.; Thompson, J.; Haas, H. A Survey of Positioning Systems Using Visible LED Lights. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 1963–1988. [CrossRef]
3. Chowdhury, M.Z.; Hossan, M.T.; Islam, A.; Jang, Y.M. A comparative survey of optical wireless technologies: Architectures and applications. *IEEE Access* **2018**, *6*, 9819–9840. [CrossRef]
4. Modern Indoor Living Can Be Bad for Your Health: New YouGov Survey for VELUX Sheds Light on Risks of the "Indoor Generation". Available online: https://www.prnewswire.com/news-releases/modern-indoor-living-can-be-bad-for-your-health-new-yougov-survey-for-velux-sheds-light-on-risks-of-the-indoor-generation-300648499.html (accessed on 27 November 2018).
5. Ghassemlooy, Z.; Popoola, W.; Rajbhandari, S. *Optical Wireless Communications, System and Channel Modeling with MATLAB*; CRC Press: Boca Raton, FL, USA, 2012; ISBN 9781439851883.
6. Liu, H.; Darabi, H.; Banerjee, P.; Liu, J. Survey of Wireless Indoor Positioning Techniques and Systems. *IEEE Trans. Syst. Man. Cybern. Part C (Appl. Rev.)* **2007**, *37*, 1067–1080. [CrossRef]
7. Eroglu, Y.S.; Guvenc, I.; Pala, N.; Yuksel, M. AOA-based localization and tracking in multi-element VLC systems. In Proceedings of the 2015 IEEE 16th Annual Wireless and Microwave Technology Conference (WAMICON), Cocoa Beach, FL, USA, 13–15 April 2015. [CrossRef]

8. Du, P.; Zhang, S.; Chen, C.; Alphones, A.; Zhong, W.D. Demonstration of a Low-Complexity Indoor Visible Light Positioning System Using an Enhanced TDOA Scheme. *IEEE Photonics J.* **2018**, *10*, 1–10. [CrossRef]

9. Jung, S.; Hann, S.; Park, C. TDOA-based optical wireless indoor localization using LED ceiling lamps. *IEEE Trans. Consum. Electron.* **2011**, *57*, 1592–1597. [CrossRef]

10. Zhang, S.; Zhong, W.D.; Du, P.; Chen, C. Experimental Demonstration of Indoor Sub-Decimetre Accuracy VLP System Using Differential PDOA. *IEEE Photonics Technol. Lett.* **2018**, *30*, 1703–1706. [CrossRef]

11. Naz, A.; Asif, H.M.; Umer, T.; Kim, B.S. PDOA Based Indoor Positioning Using Visible Light Communication. *IEEE Access* **2018**, *6*, 7557–7564. [CrossRef]

12. Van, M.T.; Van Tuan, N.; Son, T.T.; Le-Minh, H.; Burton, A. Weighted k-nearest neighbour model for indoor VLC positioning. *IET Commun.* **2017**, *11*, 864–871. [CrossRef]

13. Guo, X.; Shao, S.; Ansari, N.; Khreishah, A. Indoor Localization Using Visible Light via Fusion of Multiple Classifiers. *IEEE Photonics J.* **2017**, *9*, 1–16. [CrossRef]

14. Gu, W.; Kashani, M.A.; Kavehrad, M. Multipath reflections analysis on indoor visible light positioning system. *Comput. Sci.* **2015**, *57*, 13–24. [CrossRef]

15. Mohammed, N.; Elkarim, M. Exploring the effect of diffuse reflection on indoor localization systems based on RSSI-VLC. *Opt. Express* **2015**, *23*, 20297–20313. Available online: https://www.osapublishing.org/oe/abstract.cfm?uri=oe-23-16-20297 (accessed on 27 November 2018). [CrossRef] [PubMed]

16. Gu, W.; Aminikashani, M.; Deng, P.; Kavehrad, M. Impact of Multipath Reflections on the Performance of Indoor Visible Light Positioning Systems. *J. Lightw. Technol.* **2016**, *34*, 2578–2587. [CrossRef]

17. Fu, M.; Zhu, W.; Le, Z.; Manko, D. Improved visible light communication positioning algorithm based on image sensor tilting at room corners. *IET Commun.* **2018**, *12*, 1201–1206. [CrossRef]

18. Kahn, J.M.; Barry, J.R. Wireless infrared communications. *Proc. IEEE* **1997**, *85*, 265–298. [CrossRef]

19. Gfeller, F.R.; Bapst, U. Wireless in-house data communication via diffuse infrared radiation. *Proc. IEEE* **1979**, *67*, 1474–1486. [CrossRef]

20. Fan, K.; Komine, T.; Tanaka, Y.; Nakagawa, M. The effect of reflection on indoor visible-light communication system utilizing white LEDs. In Proceedings of the 5th International Symposium on Wireless Personal Multimedia Communications, Honolulu, HI, USA, 27–30 October 2002; Volume 2, pp. 611–615. [CrossRef]

21. Improving the random Forest in Python Part 1. Available online: https://towardsdatascience.com/improving-random-forest-in-python-part-1-893916666cd (accessed on 27 November 2018).

22. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Vanderplas, J.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.