sensors

MDPI

Article

UAPF: A UWB Aided Particle Filter Localization For Scenarios with Few Features

Yang Wang [1], Weimin Zhang [1,2,3,*], Fangxing Li [1,2,3], Yongliang Shi [1], Fuyu Nie [1] and Qiang Huang[1,2,3]

[1] School of Mechatronical Engineering, Beijing Institute of Technology, Beijing 100081, China; 3120190166@bit.edu.cn (Y.W.); wonk2000@bit.edu.cn (F.L.); ylshi@bit.edu.cn (Y.S.); 3120190164@bit.edu.cn (F.N.); qhuang@bit.edu.cn (Q.H.)

[2] Key Laboratory of Biomimetic Robots and Systems, Ministry of Education, Beijing Institute of Technology, Beijing 100081, China

[3] Beijing Advanced Innovation Center for Intelligent Robots and Systems, Beijing 100081, China

[*] Correspondence: zhwm@bit.edu.cn

Received: 30 September 2020; Accepted: 25 November 2020; Published: 28 November 2020

Abstract: Lidar-based localization doesn't have high accuracy in open scenarios with few features, and behaves poorly in robot kidnap recovery. To address this problem, an improved Particle Filter localization is proposed who could achieve robust robot kidnap detection and pose error compensation. UAPF adaptively updates the covariance by Jacobian from Ultra-wide Band information instead of predetermined parameters, and determines whether robot kidnap occurs by a novel criterion called KNP (Kidnap Probability). Besides, pose fusion of ranging-based localization and PF-based localization is conducted to decrease the uncertainty. To achieve more accurate ranging-based localization, linear regression of ranging data adopts values of maximum probability rather than average distances. Experiments show UAPF can achieve robot kidnap recovery in less than 2 s and position error is less than 0.1 m in a hall of 40 by 15 m, when the currently prevalent lidar-based localization costs more than 90 s and converges to wrong position.

Keywords: particle filters; sensor fusion; ultra wideband technology; robot localization; robot kidnap recovery

1. Introduction

Localization technology is subdivided into outdoor and indoor localization according to application scenarios. Global Positioning System (GPS)-based outdoor positioning services has almost matured and is widely used. However, GPS cannot achieve indoor positioning accurately due to severe occlusion. Moreover, indoor localization will bring inevitable errors to the results due to complex environmental structure, uncertain conditions, and numerous obstacles [1].

In order to cope with the state estimation of robots, localization based on probabilistic algorithms is the only effective solution currently known [2]. As the core idea of probabilistic localization, Bayesian filtering algorithm occupies an important role. In the early days, the best technology for implementing Bayesian filtering was Kalman Filter (KF), which could achieve efficient state estimation for linear Gaussian systems, but difficult to depict non-linear systems. Therefore, extended Kalman Filter (EKF) and unscented Kalman Filter (UKF) were proposed to solve the state estimation in nonlinear systems. In general, EKF and UKF perform well except systems highly non-Gaussian distributions. On this basis, PF is applied as a non-parametric filter [2], whose typical implementation is AMCL [3], which performs well in localization efficiency, stability, and accuracy, but poorly in global localization in scenarios with few features.

Indoor localization technology can be subdivided into single sensor localization and multi-sensor localization whose sensors include ultrasonic [4], infrared [5,6], vision [7], lidar, radio frequency identification (RFID) [8,9], Bluetooth [10], Wi-Fi [11] and so on. In addition, UWB [12,13] has also become a research hot-spot in recent years. Due to the limitations of a single sensor, multi-sensor combined localization is generally used in actual applications.

The odometry is a very widely used sensor in wheeled robot localization [11,14]. It has the characteristics of easy data processing, controllable accuracy, and high universality. However, because of accumulated errors, localization accuracy exists during long-term operation will gradually decrease. Thanks to the high ranging accuracy, little influence of light, and easy installation, lidar is popular in various autonomous robots [15–17]. However, the effective measure distance is limited, and the matching-based method has the disadvantages of high cost and low efficiency in achieving global localization. UWB related technology has made remarkable progress since it was approved for civilian application, which has the advantages of wide-ranging and no accumulated error, but with a certain drift in the localization process. At present, the accuracy of Sapphire system of Multi-spectral Solutions is under 10 cm. Salman et al. implemented UWB localization on a mobile robot, CoLORbot, for localization in indoor unknown scenarios [13].

Therefore, there are certain disadvantages when a single sensor acquires information making it difficult to achieve accurate localization, due to which different kinds of sensors are usually combined for localization [18]. White introduced the general model of data fusion in 1998 [19], and Hall et al. introduced the algorithms and strategies of data fusion in detail [20]. At present, the methods employed to multi-sensor fusion localization generally include Bayesian based methods [3,11] and neural network methods [21,22]. There are numerous data fusion methods based on the multi-Bayesian estimation. The Kalman Filter (KF), a kind of Gaussian filter, is a recursive filter for linear systems. For non-linear systems, there are two types, Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF). In general, KF can complete data fusion well, but when it's hard to find out system models, there are cases of low real-time performance and reliability. Numerous non-parametric filters are based on the Monte Carlo Localization proposed by Fox et al, which is a non-parametric filter method based on Bayesian estimation [23]. Valerio Magnago et al. combined odometry and UWB information with UKF [14]. Peng Gang et al. added an additional Gaussian-Newton-based scan-match step on the basic of AMCL, improving the localization accuracy in complex and unstructured environments [24]. In [25], sensors node's movement dynamics and the measurements of it's velocity and the received signal strength (RSS) of the radio signal from above-ground relay nodes are used to achieve localization, using corresponding algorithms based on KF for different scenarios. The idea that one supervisor work as planer and the other supervisor improves the result supports the idea of this article [26]. With the development of machine learning, neural networks have attracted more attention as a new data fusion method. J.Wang et al. used BP neural network to estimate the GPS sampling time and performed subsequent data fusion [27].

In this paper, we focus on achieving robust robot kidnap detection and recovery based on PF localization, where accurate global proposal distribution, provided by ranging-based localization in UAPF, is necessary. In this case, adaptive estimation of the probability of robot kidnap is feasible with the criterion KNP, proposed to measure the probability of robot kidnap. To solve the problem of false identification of robot kidnap, robot kidnap recovery is triggered only if the uncertainty of the particle swarm is high enough, due to which the reliability of robot kidnap detection increases. Besides, for more accurate ranging-based localization, the double-sided two-way (DSTW) is used in ranging-based localization, in which Jacobin matrix is used to get the position error [2]. UAPF could make up for the deficiencies of global localization and robot kidnap recovery of PF and achieve accurate localization in open scenarios with few features. The contributions of this work are as follows:

1. An improved PF-based localization algorithm is proposed, which could achieve robust kidnap detection and pose error compensation.

2. A novel criterion named KNP is proposed to indicate the probability of robot kidnap, based on the inconsistency of two pose distribution.
3. An adaptive covariance matrix ameliorates the reliability of UAPF, which is provided by the improved proposal distribution with UWB information.

The rest of this paper is outlined as follows. In Section 2, we introduce the theoretical basis and the detailed system overview. In Section 3, pre-experiments are conducted to decrease ranging error and improve localization accuracy of UWB, after which experiments are presented to illustrate improvements of this method proposed in this paper. Finally, we highlight some conclusions.

2. Materials and Methods

UAPF is an improved PF-based localization method with adaptive robot kidnap detection and efficient kidnap recovery. This method mainly consists of PF-based localization [23], ranging-based localization and adaptive robot kidnap detection. In PF-based localization, 2D laser-scan is utilized to weight particles sampled around odometry pose. Adaptive robot kidnap detection focus on measuring how similar two poses are, and output an error transform matrix and KNP, which is the criterion to judge whether robot kidnap occurs. Besides, a supplementary adaptive update for particles uncertainty is conducted in this part, decreasing the error caused by fixed lidar measurement. The framework of UAPF is shown in Figure 1 and Algorithm 1.

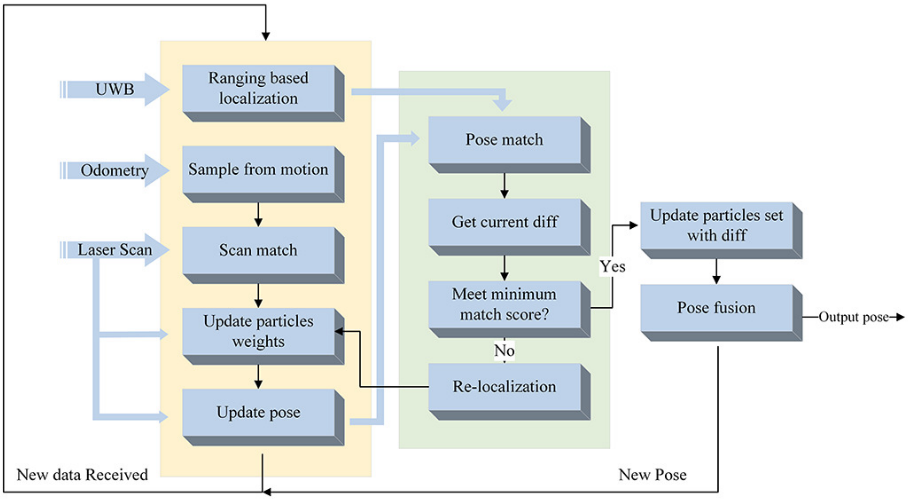


Figure 1. Framewrok of UAPF proposed in this paper.

Algorithm 1 UAPF($x_{t-1}, r_{i,t}, u_t,\ z_t$).

```
$x_{r,t} = Ranging\_Based\_Localization(r_{i,t})$
$x_{p,t} = PF\_based\_Localization(x_{t-1},\ u_t,\ z_t,\ \Sigma_p)$
Get $KNP$ according to distributions of $x_{r,t}$ and $x_{p,t}$
if $KNP > threshold$ then
    $x_t = Pose\_Fusion(x_{p,t},\ x_{r,t},\ \Sigma_r)$
else
    if $\Sigma_p \gg \Sigma_r$ then
        Re-localization:    $x_t \sim \mathcal{N}(\mu_r,\ \Sigma_p + \Sigma_r)$
    else
        $x_t = Pose\_Fusion(x_{p,t},\ x_{r,t},\ \Sigma_r)$
    end if
end if
Update $\Sigma_p$ according to $KNP$ and pose difference
return $x_t$
```

Where x_{t-1} is the fused pose in time $t-1$, $r_{i,t}$ is the distance between the robot and the $Anchor_i$, u_t is the movement of the robot, and z_t is the observing information from ranging-based localization. $x_{p,t}$ and $x_{r,t}$ are from PF-based localization and ranging-based localization separately. Similarly, Σ_p and Σ_r are the covariance describing the degree of pose dispersion. The *threshold* is a constant to determine whether trigger the re-localization process. According to our empirical data, 0.67 is a good choice to achieve good results. $\mathcal{N}(\mu_r,\ \Sigma_p+\Sigma_r)$ is the two-dimensional Gaussian distribution with mean μ_r and covariance $\Sigma_p+\Sigma_r)$. x_t is the fused pose used to describe the accuracy of UAPF.

2.1. PF-Based Localization

Original PF-based localization is improved in this paper, whose main idea is to choose particles with high weight sampled around the pose calculated according to odometry information, as shown in Algorithm 2, which is obtained by substituting last fused pose, the increment of robot movement, environment observations and corrected particles covariance into PF.

At every moment, we get current pose with differential odometry model (line 2), after which particles are sampled following $\mathcal{N}(\mu_p, \Sigma_p)$ (line 5). In this step, the use of Σ_p, corrected in (21) gives a more reasonable proposal distribution, which could adaptively adjust the size of particle swarms according to last pose error, and improve the rationality and reliability of PF-based localization. For every particle, the measurement model is applied in line 6 to weigh the importance according to the matching degree to the current local environment. Candidate particle swarm $\overline{\mathcal{X}}_t$ covers particles' poses and corresponding weights. In the update stage, re-sample is conducted, where particles with high weights are much more possible to be sampled than ones with low weights. Finally, $x_{p,t}$ is got and input into pose fusion method (line 10 in Algorithm 1).

Algorithm 2 Improved PF-based Localization (x_{t-1},u_t,z_t,Σ_p).

```
1:  $\mathcal{X}_t$ , $\overline{\mathcal{X}}_t = \phi$
2:  $x_{odom}$ = $Motion\_model(u_t, x_{t-1})$
3:  Set $m$ as the number of particles should be sampled
4:  for i = 0 to m do
5:      $x_t^i$ = $Sample\_model(x_{odom}, \Sigma_p)$
6:      $w_t^i$ = $Scanmach\_model(z_t, x_t^i, map)$
7:      $\overline{\mathcal{X}_t}$ += $< x_t^i, w_t^i >$
8:  end for
9:  for i = 0 to m do
10:     Draw $x_t^i$ from $\overline{\chi_t}$ with probability $\propto w_t^i$
11:     $\mathcal{X}_t$ += $x_t^i$
12: end for
13: $x_{p,t}$ = $Mean\ (\mathcal{X}_t)$
14: return $x_{p,t}$
```

Where $\mathcal{X}_t$ is the set to store the pose and covariance of particles of whom $\overline{\mathcal{X}}_t$ is the candidate particle swarm. The two are set to the empty set, ϕ. Pose of every particles, x_t^i, is sampled around x_{odom}, calculated according to robot movement and last robot pose, and w_t^i is the corresponding weight.

2.2. Ranging-Based Localization

DSTW ranging method is used in the ranging-based localization method. The schematic diagram is shown in Figure 2. Two axes respectively indicate the time axis of device A and device B.

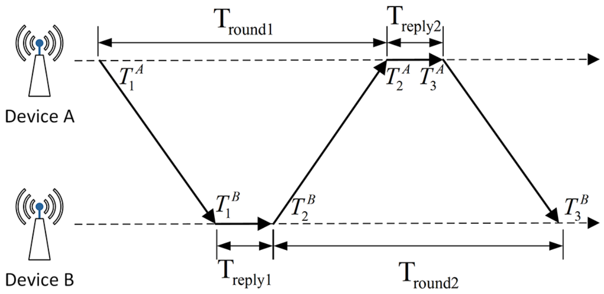


Figure 2. The data packet is sent 3 times in DSTW.

The predicted value of flight time $\hat{T}_{prop}$ can be expressed as

$$\hat{T}_{\text{prop}} = \frac{T_{\text{round1}} \times T_{\text{round2}} - T_{\text{reply1}} \times T_{\text{reply2}}}{T_{\text{round1}} + T_{\text{round2}} + T_{\text{reply1}} + T_{\text{reply2}}} \tag{1}$$

and

$$T_{\text{round1}} = T_2^A - T_1^A \tag{2}$$
$$T_{\text{round2}} = T_3^B - T_2^B \tag{3}$$
$$T_{\text{reply1}} = T_2^B - T_1^B \tag{4}$$
$$T_{\text{reply2}} = T_3^A - T_2^A \tag{5}$$

where T_{round} refers to the time from sending a packet to receiving it, and T_{reply} refers to the time of data processing for a single device. In this way, the error of flight time can be expressed as

$$T_{\text{error}} = \hat{T}_{\text{prop}} \times \left(1 - \frac{e_A + e_B}{2}\right) \tag{6}$$

where e_A and e_B refer to the ratio of actual frequency to the rated value of devices A and B. DSTW can solve the problem of time synchronization in some degree, improving the ranging accuracy, which is returned in millimeters.

Having got ranges between some anchor and the tag, triangulation is used to calculate the robot position, (x, y, z), shown in Figure 3 and Equation (7).

$$\begin{cases} (x_0 - x)^2 + (y_0 - y)^2 + (z_0 - z)^2 = \rho_0^2 \\ (x_1 - x)^2 + (y_1 - y)^2 + (z_1 - z)^2 = \rho_1^2 \\ (x_2 - x)^2 + (y_2 - y)^2 + (z_2 - z)^2 = \rho_2^2 \end{cases} \tag{7}$$

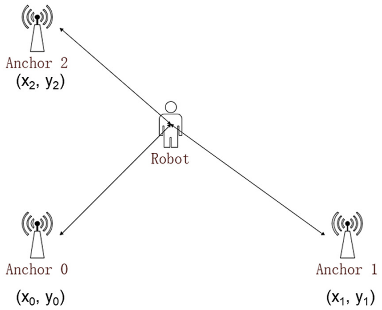


Figure 3. The model to find robot position. Given the positions of anchors and the distances between the some anchor to the robot, the position of robot could be calculated.

To cut down the cost of computation, $Anchor_0$ is set as the original point, with $Anchor_1$ set on the x-axis and $Anchor_2$ set on the y-axis, and all three anchors are at the same height z_h. In Equation (8), ρ_i expresses measured distance between the tag and $Anchor_i$, and (x_i, y_i, z_i) refer to the position of $Anchor_i$.

$$\begin{cases} x = \frac{\rho_0^2-\rho_1^2+x_1^2}{2x_1} \\ y = \frac{\rho_0^2-\rho_2^2-x^2+(x_2-x)^2+y_2^2}{2y_2} \end{cases} \tag{8}$$

Equation (8) finally shows the position of the robot in UWB coordinate system. However, the positioning accuracy cannot be estimated directly, so the detection-correction link is added, for which the error of position (dx, dy, dz) is converted from the distance error using Jacobian.

As a real-time ranging-based positioning technology, positioning with UWB has no cumulative error, but the covariance varies greatly among different hardware. Therefore, the distance errors between the robot and anchors are used to derive the coordinate error to achieve accurate position estimation.

The distance between the robot and a certain anchor $\hat{\rho}_i$ can be expressed as

$$\hat{\rho}_i = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2} \tag{9}$$

Assuming that the robot coordinates (x, y, z) are known, the ranging error of UWB is easily obtained as

$$\Delta\rho_i = \rho_i - \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2} \tag{10}$$

To obtain the coordinate error $\begin{pmatrix} dx & dy & dz \end{pmatrix}^T$, Equation (10) is derived.

$$d\rho_i = \frac{(x_i - x)\,dx + (y_i - y)\,dy + (z_i - z)\,dz}{\sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}} \tag{11}$$

By introduceing Equation (9) into Equation (11). We obtain Equation (12).

$$d\rho_i = \frac{(x_i - x)\,dx + (y_i - y)\,dy + (z_i - z)\,dz}{\rho_i} \tag{12}$$

Therefore, by converting Equation (12) into a matrix representation, a differential matrix from the coordinate error to the distance error can be obtained. Equation (14) are used to get Σ_r, which is used in Algorithms 1 and 3.

$$\begin{bmatrix} d\rho_0 \\ d\rho_1 \\ d\rho_2 \end{bmatrix} = \begin{bmatrix} \frac{x_0-x}{\rho_0} & \frac{y_0-y}{\rho_0} & \frac{z_0-z}{\rho_0} \\ \frac{x_1-x}{\rho_1} & \frac{y_1-y}{\rho_1} & \frac{z_1-z}{\rho_1} \\ \frac{x_2-x}{\rho_2} & \frac{y_2-y}{\rho_2} & \frac{z_2-z}{\rho_2} \end{bmatrix} \begin{bmatrix} dx \\ dy \\ dz \end{bmatrix} = T_{tran} \begin{bmatrix} dx \\ dy \\ dz \end{bmatrix} \tag{13}$$

$$\begin{bmatrix} dx \\ dy \\ dz \end{bmatrix} = T_{tran}^{-1} \begin{bmatrix} d\rho_0 \\ d\rho_1 \\ d\rho_2 \end{bmatrix} \tag{14}$$

2.3. Robot Kidnap Detection and Recovery

To address the problem of global localization and robot kidnap detection in traditional PF-based localization methods [23], we propose a novel criterion, KNP, which is measured according to the distribution of particles, $\mathcal{X}_t$, and the pose of ranging-based localization, x_t, where t represents values at the time t.

In update phase (the green rectangle in Figure 1), a match-based measurement method is conducted. We assume both PF-based poses, x_p, and ranging-based poses, x_r, follow 2D Gaussian distribution.

$$x_p \sim \mathcal{N}(\mu_p, \Sigma_p) \tag{15}$$

$$x_r \sim \mathcal{N}(\mu_r, \Sigma_r) \tag{16}$$

where μ_p means the center pose of PF-based localization and μ_r means the center pose of ranging-based localization. And the variance Σ_p and Σ_r presents how large sizes of particle swarms are. Moreover, Σ_r is assumed to be only related to the distance between anchors and the robot because the ranging results are corrected in the range of 3 m to 20 m, and the system is used in unobstructed scenarios for UWB, without Non-Line-of-Sight, NLOS.

$$\Sigma_r^{'} = \frac{\Sigma_r}{\alpha^n} \tag{17}$$

where α refers to attenuation coefficients, and n is the number that ranging distance over 20 m.

Moreover, to measure the possibility of robot kidnapping, a novel criterion called KNP is introduced into UAPF, expressing the difference between two kinds of localization methods. In Equation (18), expectations and covariance matrices are substituted to get the Wasserstein distance between two localization methods.

$$S = W^2(x_p, x_r) = \left\|\mu_p - \mu_r\right\|^2 + tr\left[\Sigma_p + \Sigma_r^{'} - 2\left(\Sigma_p^{1/2}\Sigma_r^{'}\Sigma_p^{1/2}\right)^{1/2}\right] \tag{18}$$

Then, in Equation (19), KNP is to measure the possibility of robot kidnapping, as shown in line 5 of Algorithm 1 and Equation (21). Generally, the smaller KNP is, the more possible robot kidnap occurs, and it could maintain a relative score of 0.8 with normal operating conditions.

$$KNP = \lambda(p_1S^4 + p_2S^3 + p_3S^2 + p_4S^1 + p_5) \tag{19}$$

2.4. Particles Update for Pose Tracking

In traditional PF-based localization [3], pose error is measured by the variance of particles swarm, but sensor noise of odometry and lidar is regarded as fixed parameters, which ruins the accuracy of localization to some degree.

In general, when the robot moves from x_{t-1} to x_t, with the odometry movement u and the map m, we can obtain the probability distribution of the robot pose. The combined localization of lidar and odometry is robust in most cases. However, lidar can only reduce the accumulated error of the odometry, rather than eliminates it. Therefore, UWB is introduced to eliminate the accumulative error of the system. Therefore, the probability distribution of x_t can be expressed as (20).

$$\begin{aligned} p\left(x_t|x_{t-1},u,m,z\right) &= \eta p\left(z|x_t,m\right) p\left(x_t|x_{t-1},u\right) = \eta \prod_j p\left(z_j|x_t,m\right) p\left(x_t|x_{t-1},u\right) \\ &= \eta\left(z_{uwb}|x_t\right) p\left(z_{lidar}|x_t,m\right) p\left(x_t|x_{t-1},u\right) \end{aligned} \tag{20}$$

where η is the normalization constant, $p(x_t|x_{t-1},u)$ expresses the odometry pose calculated by robot motion, $p\left(z_{lidar}|x_t,m\right)$ expresses lidar likelihood domain model and $p(z_{uwb}|x_t)$ is measured by ranging-based localization sub-process.

KNP could adaptively measure localization accuracy to some degree, but not enough for real-time pose tracking. Therefore, the euclidean distance between the results of two localization methods is taken into account to update Σ_p.

$$\Sigma_p \leftarrow \Sigma_p + \begin{bmatrix} \frac{\Delta x^2}{KNP} & & \\ & \frac{\Delta y^2}{KNP} & \\ & & \mathbf{0}_{(4,4)} \end{bmatrix} \tag{21}$$

where Σ_p represents the covariance of PF-based localization. Δx^2 and Δy^2 are pose differences between the two localization methods. In (21), only position error is to update because of the low reliability of the yaw in ranging-based localization.

2.5. Pose Fusion

As mentioned above, the UWB position has large uncertainty, which is manifested as positioning results jumping around the real value. Therefore, the fusion of PF-based poses and UWB poses is conducted, to improve the accuracy, as shown in Algorithm 3.

Firstly, fusion starts with UWB pose as initial pose, for solving global localization. In every time t, PF-based localization is regarded as predictive pose, $\hat{x}_t$. In update stage, the result of ranging-based localization, $x_{r,t}$ and Σ_r is used. Due to the high frequency (20Hz) of ranging-based localization, we use sliding window for the average pose value, especially for θ_r.

Algorithm 3 Pose Fusion ($x_{p,t}$,$x_{r,t}$,Σ_r).

```
if t-1 = 0 then
    Initialize $\hat{x}_t$ with $x_{r,t}$
else
    Predict: $\hat{x}_t = x_{p,t} + \delta_p$
    Update: $x_t = Fusion\_Method\,(\hat{x}_t, x_{r,t}, \Sigma_r)$
end if
return $x_t$
```

Where δ_p is the noise compensation which obeys Gaussian distribution. θ_r is the yaw of the ranging-based localization pose.

3. Experiments and Results

3.1. Experimental Scenario and Platform

Figure 4a shows the experimental scenario of this paper. The robot platform used in the experiment is shown in Figure 4b.

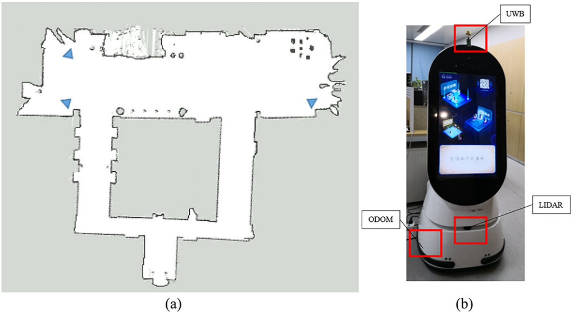


Figure 4. (**a**) The top half of the scenario is a hall of 40 by 15 m. The three triangles are poses of anchors. (**b**)The robot platform is equipped with odometry, lidar, and UWB.

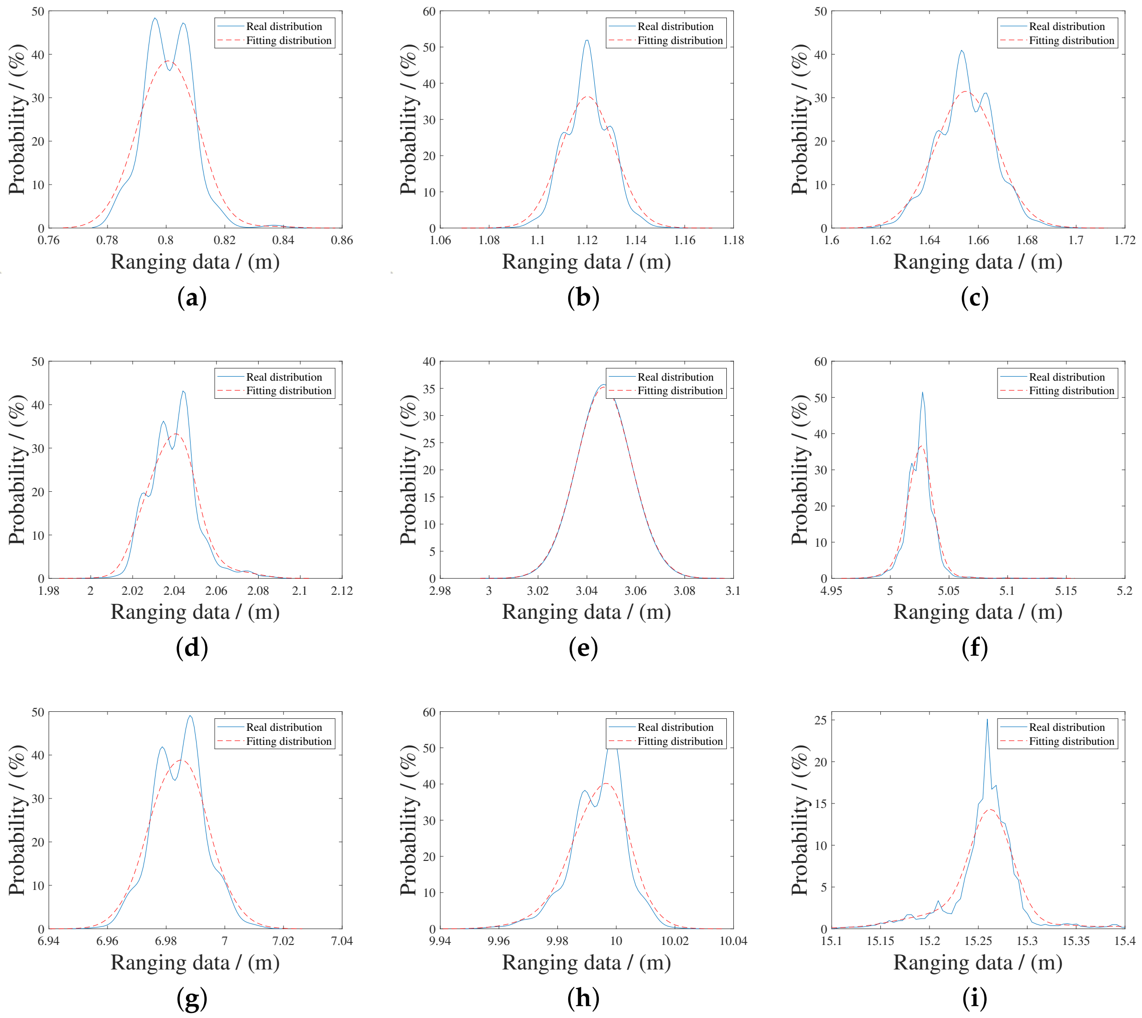
3.2. Ranging Experiments

Experiments on the ranging results of UWB are done, whose purpose is to decrease the ranging error caused by the hardware.

Let the true value of the distance between the tag and the anchor be x_{true} and the measured value be x_m. 1500 ranging experiments were performed at 10 different x_{true} in total. Table 1 shows the results of ranging experiments, got by (1), and Figure 5 shows the probability distribution of various distances, which are approximately Gaussian distributions.

Table 1. Results of the ranging experiment.

x_{true} (m)	Average of x_m (m)	Error (m)	Standard Deviation (m)	The Most Probable Value (m)
0.70	0.8007	0.1007	0.0078	0.80045
1.00	1.1204	0.1204	0.0085	1.1205
1.50	1.6548	0.1548	0.0112	1.6544
2.00	2.0394	0.0394	0.0109	2.0403
3.00	3.0471	0.0471	0.0055	3.0468
5.00	5.0254	0.0254	0.0117	5.0266
7.00	6.9843	0.0157	0.0082	6.9856
10.00	9.9938	0.0062	0.0087	9.9961
15.00	15.2645	0.2645	0.0562	15.2606
20.00	20.0607	0.0607	0.0147	20.0580

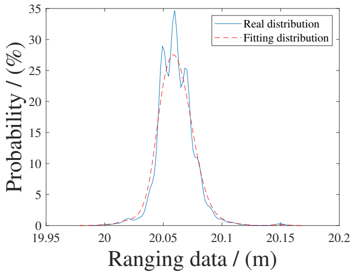


(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

(**g**) (**h**) (**i**)

Figure 5. *Cont.*



(**j**)

Figure 5. The probability of ranging measurements is shown, when distances are (**a**) 0.7 m, (**b**) 1 m, (**c**) 1.5 m, (**d**) 2 m, (**e**) 3 m, (**f**) 5 m, (**g**) 7 m, (**h**) 10 m, (**i**) 15 m, (**j**) 20 m. At every experiment, the probability distribution is approximately Gaussian distribution, but there is a deviation between the maximum probability value and the true value, shown in Table 1.

Due to the geometric relationship and the influence of terrain, the distance between the robot and a certain anchor is mostly in the range of 3 m–20 m. Therefore, we can find the relationship between x_{true} and x_m, shown in (22). Table 2 shows the results and Figure 6 shows fitting results from the distance between 3 m to 20 m. Figure 7 shows the probability distribution of this group of ranging values

$$x_{true} = 1.0172\, x_m - 0.0745 \tag{22}$$

where x_m is the measured value between the tag and Anchor$_i$ and x_{true} is the corresponding true distances.

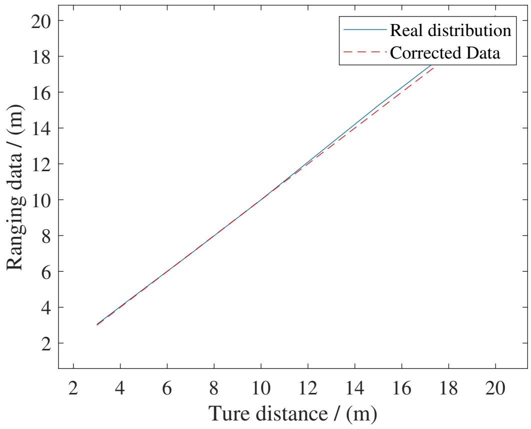


Figure 6. Fitting results from the distance between 3 m to 20 m. Corrected measurements show better linearity and are closer to the true value.

Table 2. Results of experiments after correction

x_{true} (m)	Average of x_m (m)	Error (m)	Standard Deviation (m)	The Most Probable Value (m)
3.00	3.0023	0.0023	0.0082	3.0025
5.00	4.9993	0.0007	0.0121	4.9989
7.00	6.9979	0.0021	0.0074	6.9978
10.00	10.0313	0.0313	0.0108	10.0291
15.00	15.0668	0.0668	0.0158	15.061
20.00	20.0051	0.0051	0.0071	20.0039

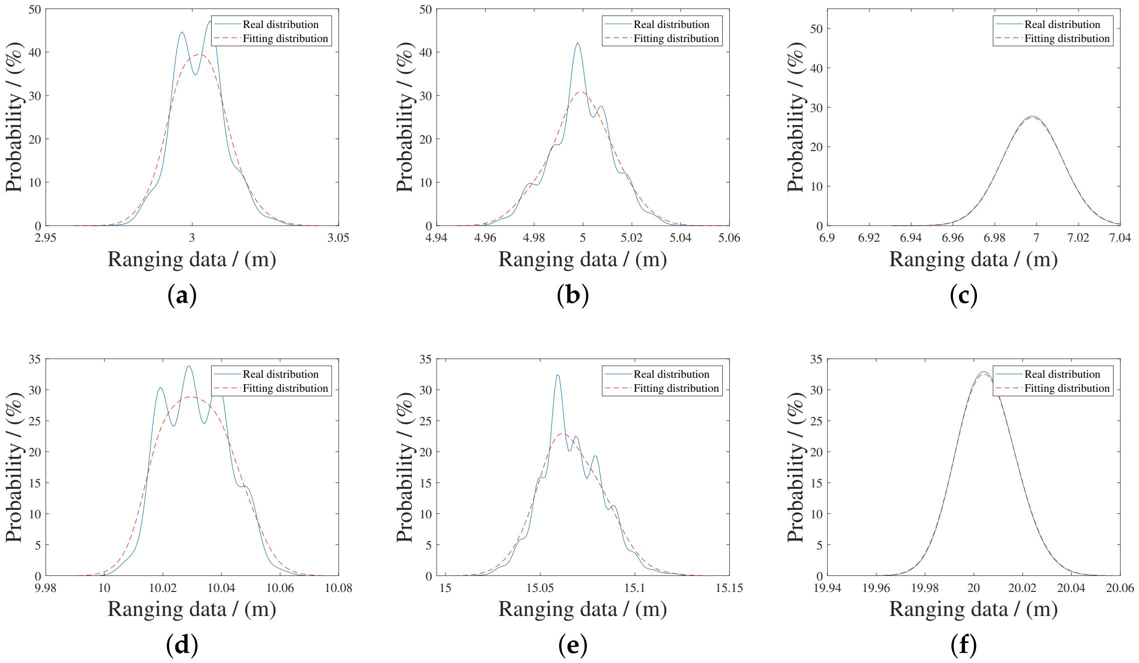


Figure 7. The probability of corrected ranging measurements is shown when distances are (**a**) 3 m, (**b**) 5 m, (**c**) 7 m, (**d**) 10 m, (**e**) 15 m, (**f**) 20 m. At every experiment, the probability distribution is approximately Gaussian distribution, and there is a smaller deviation between the maximum probability value and the true value, shown in Table 2.

Figure 8 shows the comparison results of two experiments. In the vicinity of general working range (3 m–20 m) of the robot, corrected ranging results is more accurate, especially when x_{true} is 15 m (from 0.25 m to about 0.05 m).

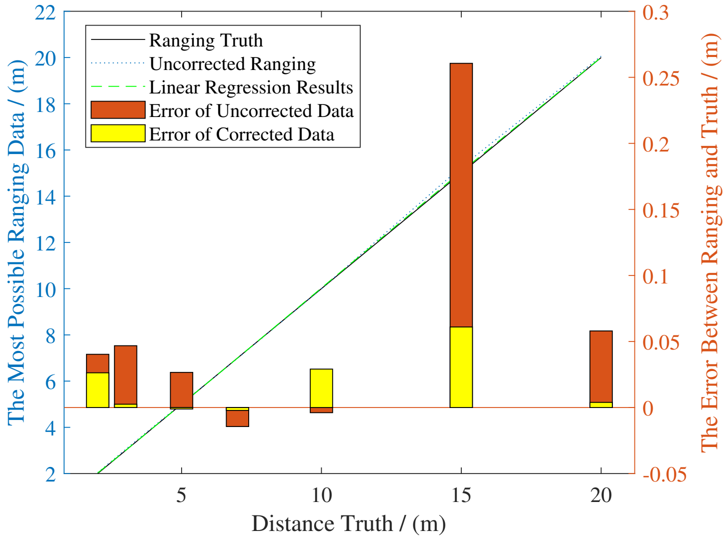


Figure 8. Comparison shows linear regression improves the accuracy of ranging, with error limited in 0.05 m.

3.3. Global Localization

Global positioning accuracy is measured to figure out whether the correction of ranging useful, in which Probability Density Estimation (PDE) is conducted. Figure 9 shows the probability distribution of ranging-based poses. Most of the measurement points are near the true coordinates. Figure 10 shows that the deviation of ranging-based localization is within 0.2 m in both X and Y directions.

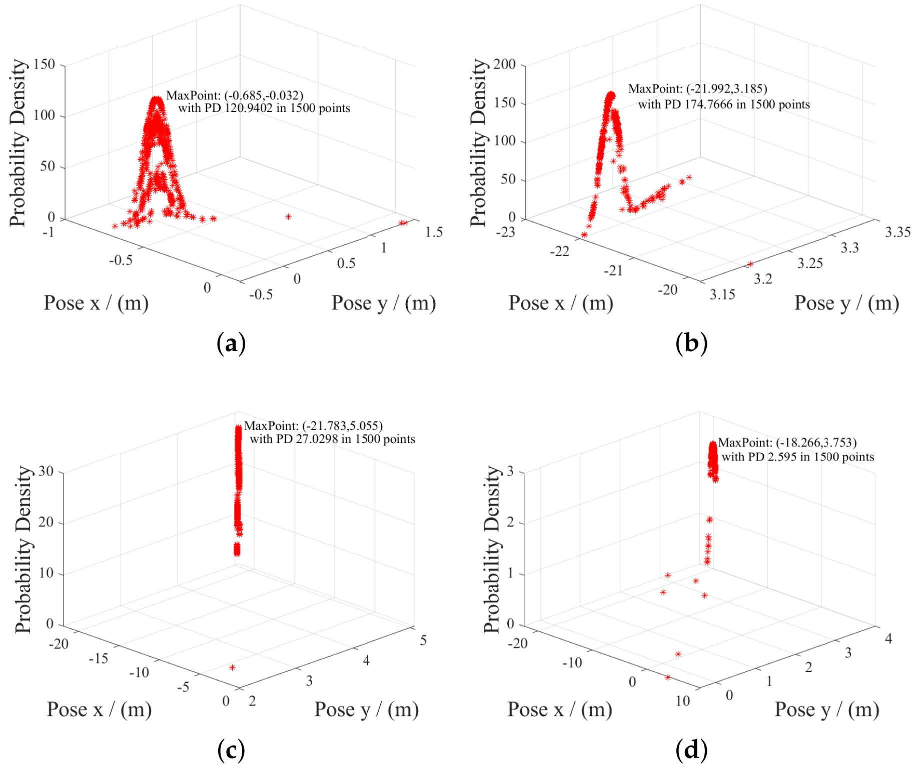


Figure 9. The probability distributions of poses is shown when UAPF conducts global localization in four positions, (**a**) $(x_{true}, y_{true}) = (-0.6,0)$, $(x_{true}, y_{true}) =$ (**b**) $(-22,3)$, $(x_{true}, y_{true}) =$ (**c**) $(-21.7,5)$, (**d**) $(x_{true}, y_{true}) = (-18,3.7)$, which obey Gaussian distributions and is consistent with some theoretical hypotheses.

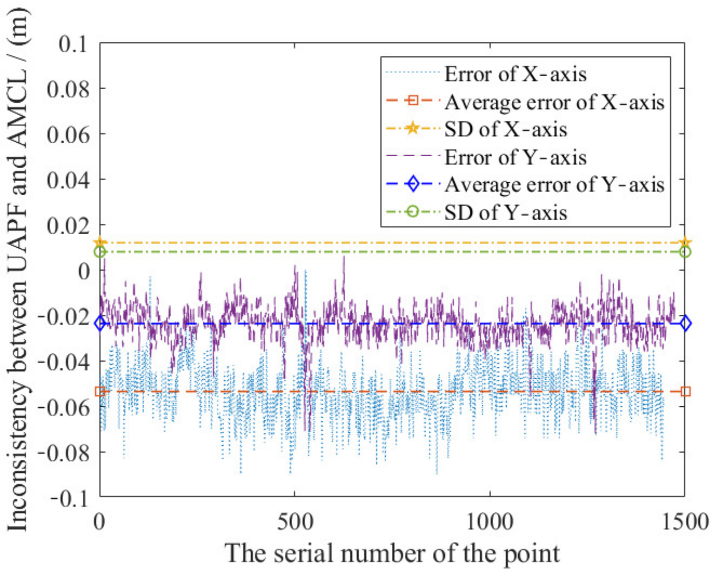


Figure 10. The stability in both X and Y directions. Average errors are both within 0.05 m and the standard deviation (SD) is less than 0.01 m.

For PF-based global localization, better performance generally comes with more particles and bigger covariance. However, in large scenarios, this relation becomes more blurred because of few features for scan matching. Besides, more particles mean more consumption cost. Table 3 shows the time required for UAPF to achieve global localization when the number of particles is less than 1000. In the course of 10 experiments, the average time is 2.1 s, compared to more than 90 s with AMCL shown in Table 4, where false convergence occurs with the number of particles almost 10,000 (shown in Figure 11f).

Table 3. Time to achieve global localization of UAPF.

Experiment number	1	2	3	4	5	6	7	8	9	10
Convergence time (s)	2	2	1	2	1	6	2	2	2	1

Table 4. Time to achieve global localization of AMCL.

Experiment number	1	2	3	4	5	6	7	8	9	10
Convergence time (s)	117	83	74	72	83	66	102	97	132	85

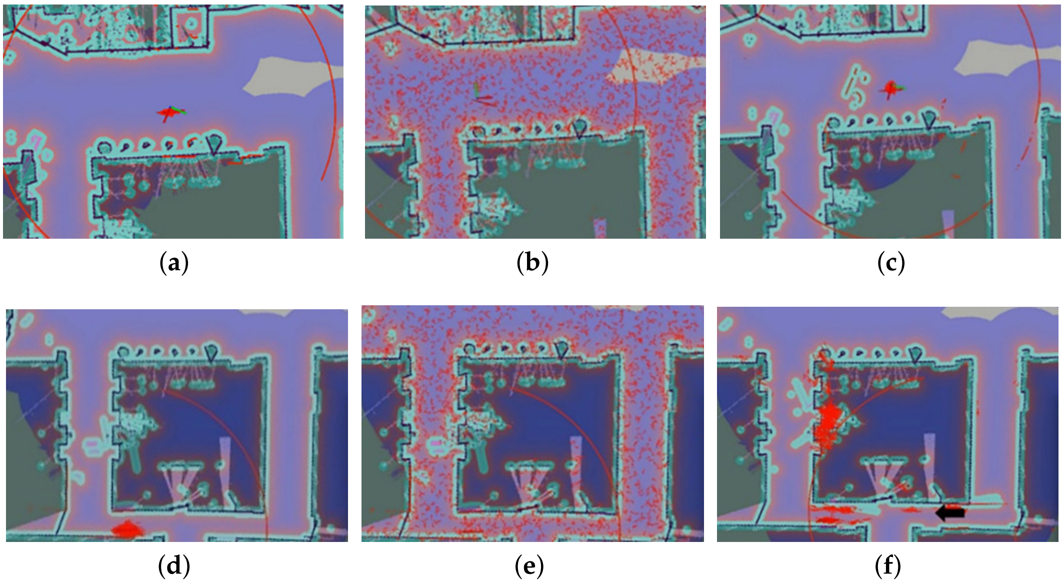


Figure 11. Comparison of global localization between AMCL and UAPF. (**a**) The initial pose of UAPF. (**b**) Global localization with UAPF. (**c**) Result of global localization using UAPF. (**d**) The initial pose of AMCL. (**e**) Global localization with AMCL. (**f**) Result of global localization using AMCL.

3.4. Robot Kidnap Recovery

Intuitively, it's easy to achieve recovery from robot kidnapping if the particles swarm has more particles and bigger covariance, which could cost more. Therefore, in this subsection, the number of particles is from 500 to 1000 for UAPF, and from 5000 to 10,000 for AMCL. To simulate a robot kidnap, we move the robot without data of odometry. Figure 12b,c express the situation where no kidnap recovery is performed because KNP is higher than the threshold. Figure 12c shows that when the robot moves, PF-based localization can't achieve robot kidnap detection in real time, making KNP and the uncertainty of PF-based localization increase. Then, in Figure 12d, UAPF achieves a robot kidnap recovery. The odometry is enabled in Figure 12e, when the PF-based localization works normally but there is still some inconsistency between two pose distribution. Figure 12f shows the results of the kidnap recovery. When the odometry, lidar, and UWB work simultaneously, an accurate localization can be achieved.

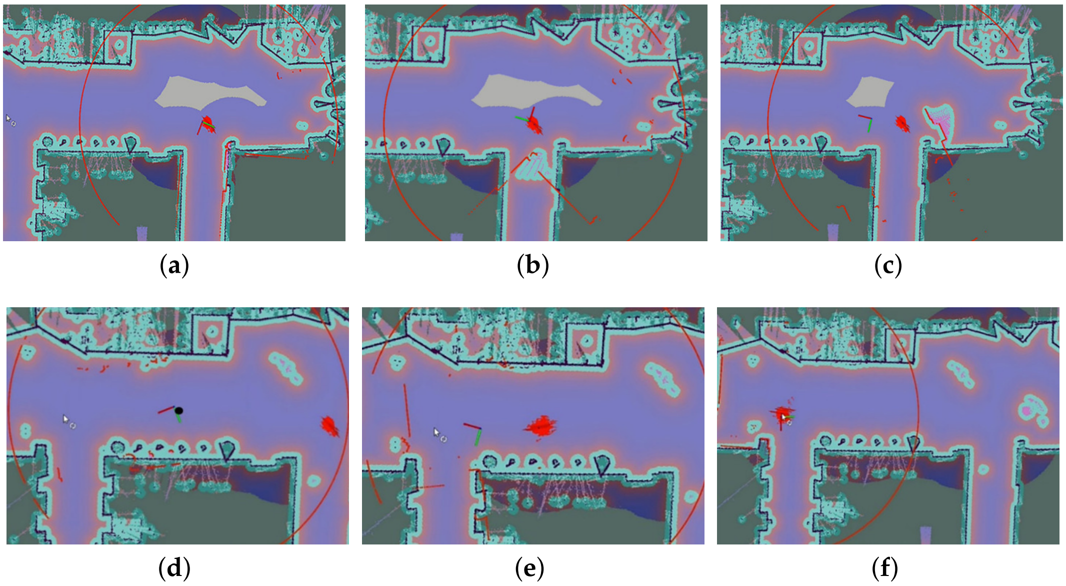


Figure 12. How UAPF achieves robot kidnap recovery. (**a**) Initial pose. (**b**) Rotating without odometry information. (**c**) Moving without odometry information. (**d**) First pose recovery. (**e**) The odometry is activated. (**f**) Second pose recovery is triggered, particles swarm is converged to true pose.

3.5. Pose Tracking

Figure 13 shows the trajectories of single ranging-based localization, Adaptive Mento Carlo Localization and UAPF. The trajectory of individual ranging-based localization is more unstable while the trajectory of UAPF is closer to AMCL (as compared). The two red rectangles show that when a huge bias (about 0.2 m) of ranging-based localization exists, UAPF has the analogous performance to AMCL.

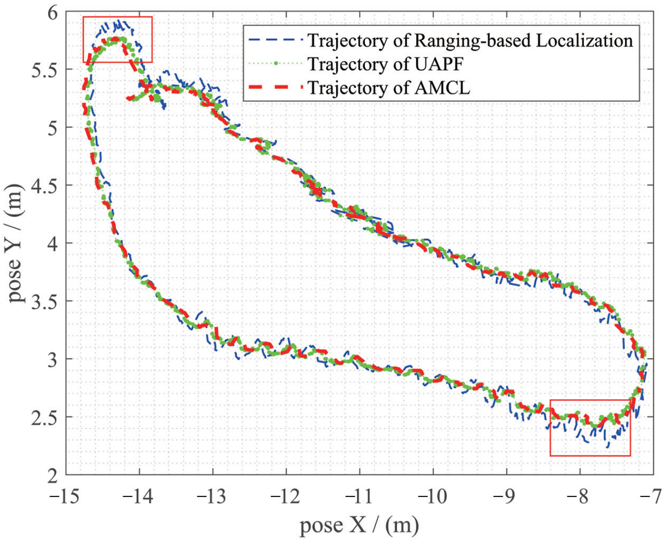


Figure 13. Trajectories of ranging-based localization, UAPF and AMCL. In general, the three have similar performance in accuracy. The two red rectangles show that UAPF could correct the instability of ranging-based localization.

4. Discussion

In this paper, we presented an indoor localization method for open scenarios with few features. Ranging-based localization provided the initial pose for first global localization, and then pose fusion was conducted as the basis of normal pose tracking. Moreover, we used PF-based localization to overcome noise from sensors. A novel criterion called KNP was introduced into UAPF to evaluate the possibility of robot kidnapping and the stability of localization together with the covariance of particles swarm. Experiments in a real-world situation indicated UAPF could achieve robot kidnap recovery in less than 2 s and position error is less than 0.1 m in a hall of 600 m^2.

In section 3, we compared our method with AMCL, because it's state of the art PF-based indoor localization method using lidar and odometry. Tables 1 and 2 indicated that the regression function (22) was suitable for experimental scenario and Figure 8 showed intuitively how linear regression improves the accuracy of ranging.

Tables 3 and 4 expressed time to achieve global localization with UAPF and AMCL separately. Table 5 compared the accuracy and time used to conduct the recovery from robot kidnapping. As mentioned above, the number of particles used in UAPF was from 500 to 1000 and in AMCL was from 5000 to 10,000. In this situation, UAPF could still conduct global localization in less than 3 s on average, much less than AMCL, illustrating the efficiency of UAPF. Figure 11c,f showed results of global localization. Figure 12a–f expressed the process of robot kidnap recovery. Trajectories of different localization methods were shown in Figure 13, illustrating UAPF could achieve analogous performance to AMCL, much stable than single ranging-based localization method, which restricted further improvement of accuracy.

Table 5. Performance of robot kidnap recovery with UAPF and AMCL

Localization Methods	Time (s)	Max Error (m)	Number of Particles	Processor
AMCL	90	2.00	5000–10,000	i3 CPU
UAPF	3	0.15	500–1000	i3 CPU

In the future, the instability of ranging-based localization will be improved, and more sensors such as RGBD will be added to UAPF and make it a universal localization method. Vision-based localization will play an essential role when the robot is in an NLOS environment, lack of ranging information transferred by UWB.

5. Conclusions

In this paper, a UWB aided Particle Filter Localization method is designed to solve the problem of robot kidnap recovery and global localization in open scenarios with few features. Integrating odometry, lidar and UWB, UAPF achieves adaptive pose error compensation, as well as robust robot kidnap detection and recovery. Besides, for reliable pose tracking, pose fusion is utilized to combine PF-based localization and ranging-based localization, returning a relatively accurate pose. The probability of robot kidnap is estimated according to KNP and the uncertainty of particle swarms, and pose recovery is triggered based on the latest ranging-based pose, eliminating accumulated errors of UAPF. To improve localization accuracy, a revised ranging model based on statistical analysis is summarized from extensive experiments. The results show UAPF can achieve robot kidnap recovery in less than 2 s and position error is less than 0.1 m in a hall of 600 m^2, much more efficient than the currently prevalent lidar-based localization.

Author Contributions: Conceptualization, Y.W., W.Z. and Y.S.; methodology, Y.W. and F.N.; software, Y.W.; validation, Y.W.; formal analysis, Y.W.; investigation, Y.W.; resources, W.Z.; data curation, Y.W.; writing–original draft preparation, Y.W.; writing–review and editing, Y.W., W.Z., F.L., Y.S. and F.N.; visualization, Y.W.; supervision, W.Z., F.L. and Q.H.; project administration, Q.H.; funding acquisition, W.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China grant number 61973031 and the Preresearch Project grant number 41412040101.

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

References

1. González, J.; Blanco, J.L.; Galindo, C.; Ortiz-De-Galisteo, A.; Fernández-Madrigal, J.; Moreno, F.A.; Martínez, J. Mobile robot localization based on Ultra-Wide-Band ranging: A particle filter approach. *Robot. Auton. Syst.* **2009**, *57*, 496–507. [CrossRef]
2. Thrun, S. Probabilistic robotics. *Commun. ACM* **2002**, *45*, 52–57. [CrossRef]
3. Fox, D. Adapting the Sample Size in Particle Filters Through KLD-Sampling. *Int. J. Robot. Res.* **2003**, *22*. [CrossRef]
4. Haigh, S.; Kulon, J.; Partlow, A.; Rogers, P.; Gibson, C. A Robust Algorithm for Classification and Rejection of NLOS Signals in Narrowband Ultrasonic Localization Systems. *IEEE Trans. Instrum. Meas.* **2019**, *68*, 646–655. [CrossRef]
5. Gorostiza, E.; Lázaro, J.; Meca, F.; Salido-Monzú, D.; Espinosa, F.; Puerto, L. Infrared Sensor System for Mobile-Robot Positioning in Intelligent Spaces. *Sensors* **2011**, *11*, 5416–5438. [CrossRef] [PubMed]
6. Rashid, A.; Ali, A. Performance Analysis of Low-Cost Infrared Sensors for Multi-Robot Localization and Communication. *Int. J. Comput. Appl.* **2018**, *182*, 23–29. [CrossRef]
7. Rémi, B.; Romain, R.; Lei, Q.; Pierre, M.; Xavier, S. A Vision-Based System for Robot Localization in Large Industrial Environments. *J. Intell. Robot. Syst.* **2019**, *99*, 359–370. [CrossRef]
8. Tao, B.; Wu, H.; Gong, Z.; Yin, Z.; Ding, H. An RFID-Based Mobile Robot Localization Method Combining Phase Difference and Readability. *IEEE Trans. Autom. Sci. Eng.* **2020**, 1–11. [CrossRef]
9. Tzitzis, A.; Megalou, S.; Siachalou, S.; Emmanouil, T.G.; Kehagias, A.; Yioultsis, T.V.; Dimitriou, A.G. Localization of RFID Tags by a Moving Robot, via Phase Unwrapping and Non-Linear Optimization. *IEEE J. Radio Freq. Identif.* **2019**, *3*, 216–226. [CrossRef]
10. Perera, C.; Aghaee, S.; Faragher, R.; Harle, R.; Blackwell, A.F. Contextual Location in the Home Using Bluetooth Beacons. *IEEE Syst. J.* **2019**, *13*, 2720–2723. [CrossRef]
11. Shi, Y.; Zhang, W.; Yao, Z.; Li, M.; Liang, Z.; Cao, Z.; Zhang, H.; Huang, Q. Design of a Hybrid Indoor Location System Based on Multi-Sensor Fusion for Robot Navigation. *Sensors* **2018**, *18*, 3581. [CrossRef] [PubMed]
12. Li, X.; He, D.; Jiang, L.; Yu, W.; Chen, X. A method indoor multi-path IR-UWB location based on multi-task compressive sensing. In Proceedings of the 2016 Fourth International Conference on Ubiquitous Positioning, Indoor Navigation and Location Based Services (UPINLBS), Shanghai, China, 2–4 November 2016; pp. 259–263. [CrossRef]
13. Salman, R.; Willms, I. A mobile security robot equipped with UWB-radar for super-resolution indoor positioning and localisation applications. In Proceedings of the 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sydney, Australia, 13–15 November 2012; pp. 1–8. [CrossRef]
14. Magnago, V.; Corbalán, P.; Picco, G.P.; Palopoli, L.; Fontanelli, D. Robot Localization via Odometry-assisted Ultra-wideband Ranging with Stochastic Guarantees. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 1607–1613. [CrossRef]
15. Tamas, L.; Lazea, G.; Popa, M.; Szoke, I.; Majdik, A. Laser Based Localization Techniques for Indoor Mobile Robots. In Proceedings of the 2009 Advanced Technologies for Enhanced Quality of Life, Iasi, Romania, 22–26 July 2009; pp. 169–170. [CrossRef]
16. Chong, Z.J.; Qin, B.; Bandyopadhyay, T.; Ang, M.H.; Frazzoli, E.; Rus, D. Synthetic 2D LIDAR for precise vehicle localization in 3D urban environment. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; pp. 1554–1559. [CrossRef]
17. Wang, S.; Kobayashi, Y.; Ravankar, A.; Ravankar, A.; Emaru, T. A Novel Approach for Lidar-Based Robot Localization in a Scale-Drifted Map Constructed Using Monocular SLAM. *Sensors* **2019**, *19*, 2230. [CrossRef] [PubMed]
18. Waltz, E.L.; Buede, D.M. Data Fusion and Decision Support for Command and Control. *IEEE Trans. Syst. Man Cybern.* **1986**, *16*, 865–879. [CrossRef]
19. White, F. A Model for Data Fusion. In Proceedings of the 1st National Symposium on Sensor Fusion, FL, USA, 5–8 April 1988.
20. Hall, D.; Llinas, J. *Multisensor Data Fusion*; CRC Press: Boca Raton, FL, USA, 2001.

21. Wang, X.; Gao, L.; Mao, S.; Pandey, S. CSI-Based Fingerprinting for Indoor Localization: A Deep Learning Approach. *IEEE Trans. Veh. Technol.* **2017**, *66*, 763–776. [CrossRef]
22. Angeletti, G.; Caputo, B.; Tommasi, T. Adaptive Deep Learning Through Visual Domain Localization. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 7135–7142. [CrossRef]
23. Fox, D.; Burgard, W.; Dellaert, F.; Thrun, S. Monte carlo localization: Efficient position estimation for mobile robots. *AAAI/IAAI* **1999**, *1999*, 2.
24. Peng, G.; Zheng, W.; Lu, Z.; Liao, J.; Hu, L.; Zhang, G.; He, D. An Improved AMCL Algorithm Based on Laser Scanning Match in a Complex and Unstructured Environment. *Complexity* **2018**, *2018*, 1–11. [CrossRef]
25. Wu, D.; Chatzigeorgiou, D.; Youcef-Toumi, K.; Ben-Mansour, R. Node Localization in Robotic Sensor Networks for Pipeline Inspection. *IEEE Trans. Ind. Inform.* **2016**, *12*, 809–819. [CrossRef]
26. Gonzalez, A.G.C.; Alves, M.V.S.; Viana, G.S.; Carvalho, L.K.; Basilio, J.C. Supervisory Control-Based Navigation Architecture: A New Framework for Autonomous Robots in Industry 4.0 Environments. *IEEE Trans. Ind. Inform.* **2018**, *14*, 1732–1743. [CrossRef]
27. Wang, J.; Zhang, Y.; Chong, K.T. Gps/dr navigation data fusion research using neural network. In Proceedings of the 2009 Fifth International Conference on Natural Computation, Tianjin, China, 14–16 August 2009; Volume 2, pp. 58–61. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.