Improved grid mapping technology based on Rao-Blackwellized particle filters and the gradient descent algorithm

Tengfei Zhang\textsuperscript{a}, Chuanjiang Wang\textsuperscript{a,b}, Zhen Yuan\textsuperscript{a} and Mingyue Zheng\textsuperscript{a}

\textsuperscript{a}College of Electrical Engineering and Automation, Shandong University of Science and Technology, Qingdao, People's Republic of China; \textsuperscript{b}School of Control Science and Engineering, Shandong University, Jinan, People's Republic of China

\textbf{ABSTRACT}

Recently, the Rao-Blackwellized particle filter (RBPF) has been used to solve the problem of simultaneous localization and mapping (SLAM). Using the odometer information of robot to calculate the proposed distribution requires a number of sampled particles, which increases the calculation complexity in the RBPF operation. In this paper, we integrate the odometer measurement and sensor observation into the proposed distribution, effectively reducing the particle sample scale. To reduce the inconsistency in the map model caused by the cumulative error of the odometer information of robot, we applied a gradient descent algorithm to fuse the sensor data to obtain the real-time attitude angle. This combination method, based on the robot operation system (ROS), runs on a platform of self-built mobile robot equipped with a laser rangefinder. The experimental results show that this method can realize the online real-time high-precision grid map which provides a new approach for robot navigation and SLAM.

\textbf{KEYWORDS}

RBPF; proposed distribution; gradient descent algorithm; ROS

1. Introduction

Building maps is the most basic task on the mobile robot platform. In general, mapping is a process concerning simultaneous localization and mapping (SLAM). The robot requires continuous map upgrading to obtain its best estimated pose in the process of SLAM, which is considered as a tricky issue. The RBPF introduced by Doucet et al. (2013) breaks down the joint probability density function into two parts. In this way, simultaneous localization and mapping problems can be tackled at the same time. However, the main problem with the RBPF approach lies in complexity, measured by the number of particles needed to build an accurate map. Therefore, reducing the number of particles and constructing accurate maps are the major issues of the algorithm. In addition, cumulative errors of robot pose usually result in overlapping and distorting maps. This effect is also known as the odometer measurement error problem (Yang, Yang, & Cai, 2015).

In this article, we improve the performance of the SLAM grid map (Florian, Nico, Jeff, & Darius, 2010; Georg & Dirk, 2017; Matthias, Volker, & Jürgen, 2014) construction by optimizing the odometer attitude measurement information and incorporating high-precision proposed distributions in RBPF.

The odometer attitude measurement information is the source of the data in the SLAM algorithm and affects the performance of the entire system. Intuitively, the RBPF algorithm will have better performance with good attitude measurement information, so that the maps texture will be clearer. As a result, we use the gradient descent filter algorithm (Madgwick, 2010) to achieve the fusion of data poses of inertial measurement units (IMU).

The distribution of proposals is calculated by evaluating the likelihood of the most likely poses that are dependent on the particles obtained by the scan matching procedure (Giorgio, Cyrill, & Wolfram, 2007). This improved method of proposed distribution calculations can be more accurate than just considering the application of odometry models, and effectively reduce the need for sampling particle quantities. In addition, we adopted an adaptive resampling strategy to reduce the risk of particle depletion.

Our method is extensively verified through a series of system experiments. All the experiments conducted have proved that our method can effectively reduce the number of sampled particles and create more accurate metric maps. These experiments were carried out on a mobile robot platform based on the robot operation system, which is released by the Willow garage company.
Experimental data were displayed on RVIZ which is a 3D visualization tool provided by ROS.

2. Odometer attitude data optimization

The data of the odometry model is acquired from IMU and the motor encoder. IMU data can be obtained by integrating the gyroscope and accelerometer. However, the IMU information is easily affected by high noise and other environmental factors. Besides, as the system’s runtime increases, the integration calculation will lead to accumulation errors. Therefore, a new algorithm is needed to fuse data for obtaining accurate attitude data.

The Kalman filtering (Arif & Ali, 2015), the Gradient descent algorithm (Predrag & Marko, 2010) and the Complementary filtering (Fourati, Mamanni, Afilal, & Handrich, 2013) algorithm is used to fuse attitude data. The Kalman filter algorithm has better accuracy and effectiveness. But this algorithm is severely limited owing to its complexity and large computation. On the other hand, the complementary filtering attitude fusion algorithm features a relatively less calculation and comparatively simpler algorithm structure. It can generate good results in low-speed motion, but there will be obvious lags in the experimental platform where the motion becomes faster. In this experimental platform, a data fusion algorithm based on quaternion gradient descent algorithm (Madgwick, Harrison, & Vaidyanathan, 2011) refers to looking for the minimum value of the objective function along the gradient descent direction of the motion experiment platform.

The measured values of gravity acceleration $\mathbf{g}$ and the measured value $\mathbf{a}$ of the accelerometer in the coordinate system. The quaternion $\hat{\mathbf{q}}$ is described by the equation

$$\hat{\mathbf{q}} = [q_1 \ q_2 \ q_3 \ q_4]$$

The measured values of gravity acceleration and unit normalized accelerometer are assumed by the following equations $\hat{\mathbf{g}} = [0 \ 0 \ 1], \mathbf{a}_0 = [a_x \ a_y \ a_z]^T$. Then the error function $f_g$ is followed by the equation:

$$f_g(\hat{\mathbf{q}}, \mathbf{a}_0) = \hat{\mathbf{q}}^T \hat{\mathbf{g}} - \mathbf{a}_0 = 
\begin{bmatrix}
2(q_2 q_4 - q_1 q_3) - a_x \\
2(q_1 q_2 + q_3 q_4) - a_y \\
2(1 - q_2^2 - q_3^2) - a_z
\end{bmatrix}$$

For the above equation, $\otimes$ represents the compound operators (Madgwick, 2010). The conjugate of $\hat{\mathbf{q}}$ is defined by the equation:

$$\hat{\mathbf{q}}^* = [q_1 \ -q_2 \ -q_3 \ -q_4]$$

The Jacobian matrix of the error function can be obtained by the derivation of the error function, such as:

$$J_g(\hat{\mathbf{q}}) = \begin{bmatrix}
-2q_3 & 2q_4 & -2q_1 & 2q_2 \\
2q_2 & 2q_1 & 2q_4 & 2q_3 \\
0 & -4q_2 & -4q_3 & 0
\end{bmatrix}$$

Therefore, the gradient value of the error function is defined by the equation:

$$\nabla f(\hat{\mathbf{q}}, \mathbf{a}) = (J_g(\hat{\mathbf{q}}))^T \hat{\mathbf{g}}$$

The precondition that the gradient descent algorithm (Sebastian, Andrew, & Ravi, 2011) can track the timely attitude is that the convergence speed is larger than the movement speed of the robot. The convergence speed of the gradient descent method is closely related to the control step size $\mu_t$. The larger the step size, the faster the convergence rate. In the conventional gradient algorithm, $\mu_t$ is a fixed value. When the value of $\mu_t$ is relatively small, the static performance of the attitude solution is better. When the value of $\mu_t$ is relatively large, the dynamic performance of the attitude solution is better. Therefore, to realize a better performance of the gradient descent algorithm in the application of the high-speed and low-speed mobile platform, it is necessary to dynamically adjust the $\mu_t$ in real time. The step length $\mu_t$ of the gradient descent method is positively correlated with the angular velocity $\hat{\mathbf{q}}_{\omega,t}$ and the sampling period...
\( \Delta t \), equated as follows:

\[
\mu_t \propto \alpha \| \dot{E}q_{w,t} \| \Delta t, \quad \alpha > 1
\]

(7)

\( \alpha \) is the increasingly value of \( \mu_t \), \( w_x, w_y \) and \( w_z \) respectively indicate the angular velocity of the \( x, y \) and \( z \) axes. The module of angular velocity is defined by the following equation:

\[
\| \dot{E}q_{w,t} \| = \sqrt{(w_x)^2 + (w_y)^2 + (w_z)^2}
\]

(8)

Furthermore, the initial step \( \mu_0 \) needs to be set for \( \mu_t \), avoiding \( \mu_t = 0 \) in the lower speed movement conditions of the robot platform \( \| \dot{E}q_{w,t} \| = 0 \). The initial size of \( \mu_0 \) is usually determined through experimental data. Therefore, the gradient descent algorithm dynamic step adjustment calculation equation can be given as follows:

\[
\mu_t = \mu_0 + \alpha \sqrt{(w_x)^2 + (w_y)^2 + (w_z)^2 \Delta t}, \quad \alpha > 1
\]

(9)

The attitude quaternion obtained by the differential equation is fused with the attitude quaternion obtained by the gradient descent algorithm. The fusion equation can be given as follows:

\[
\dot{E}q_{est,t} = \gamma_t \dot{E}q_{v,t} + (1 - \gamma_t) \dot{E}q_{w,t}, \quad 0 \leq \gamma_t \leq 1
\]

(10)

The new pose quaternion \( \dot{E}q_{est,t} \) is fused at time \( t \). Gyro’s attitude quaternion \( \dot{E}q_{w,t} \) by differential equations at time \( t \). \( \gamma_t \) and \( (1 - \gamma_t) \) are weights applied at each attitude angle. The optimal value of \( \gamma \) is defined as the case where the weighted divergence of \( \dot{E}q_{w} \) is equal to the weighted convergence of \( \dot{E}q_{v} \). The equation can be deduced as follows:

\[
(1 - \gamma_t) = \frac{\gamma_t u_t}{\Delta t}
\]

(11)

The equations are rearranged into the following equation:

\[
\gamma_t = \frac{\beta}{(u_t / \Delta t) + \beta}
\]

(12)

The \( (u_t / \Delta t) \) is considered as the convergence speed of \( \dot{E}q_{v} \). The \( \beta \) is considered as the divergence speed of \( \dot{E}q_{v} \). \( u_t \) is usually determined by experimental data.

Since the convergence rate of the quaternion differential equation for solving the pose must be much larger than the rotation speed of the motion platform, and the sampling value is generally small, where \( \Delta t \) is about 1–5 ms,

\[
\gamma_t \approx \frac{\beta \Delta t}{u_t}
\]

(13)

Equations (6), (9), and (13) are substituted into Equation (14):

\[
\dot{E}q_{est,t} = \frac{\beta \Delta t}{u_t} \left( \dot{E}q_{v,t} - \mu_t \frac{\nabla f}{\| \nabla f \|} \right)
\]

+ \((1 - \gamma_t) \dot{E}q_{w,t} - \beta \Delta t \frac{\nabla f}{\| \nabla f \|}
\]

(14)

This paper selects the second-order Picasso method (Kallapur, Petersen, & Anavatti, 2009) to solve quaternion differential equations. The equation turns out to be as follows:

\[
\dot{E}q_{w,t} = \left( \left(1 - \frac{\Delta \theta^2}{8} \right) I + \frac{1}{2} [\Delta \theta] \right) \dot{E}q_{w,t-\Delta t}
\]

(15)

The \( \Delta \theta \) angle increment matrix is demonstrated by the following equation:

\[
[\Delta \theta] = \begin{bmatrix}
0 & -\Delta \theta_x & -\Delta \theta_y & -\Delta \theta_z \\
\Delta \theta_x & 0 & -\Delta \theta_z & -\Delta \theta_y \\
\Delta \theta_y & \Delta \theta_z & 0 & -\Delta \theta_x \\
-\Delta \theta_z & -\Delta \theta_y & \Delta \theta_x & 0
\end{bmatrix}
\]

(16)

\( \Delta \theta^2 \) is expressed by the following equation:

\[
\Delta \theta^2 = \Delta \theta^2_x + \Delta \theta^2_y + \Delta \theta^2_z
\]

(17)

\( \Delta \theta_x, \Delta \theta_y, \Delta \theta_z \) are the integrals of the angular velocity \( w_x, w_y, w_z \) of the gyroscope measurement angular velocity within the time range of \( (t - \Delta t, t) \). Therefore, Equation (14) is substituted into (15) to obtain the final formula of the gradient descent method for fused quaternion poses.

\[
\dot{E}q_{est,t} = \left( \left(1 - \frac{\Delta \theta^2}{8} \right) I + \frac{1}{2} [\Delta \theta] \right) \dot{E}q_{w,t-\Delta t} - \beta \Delta t \frac{\nabla f}{\| \nabla f \|}
\]

(18)

The overall flow of the gradient algorithm is shown in Figure 1.

To examine the efficiency of the calibration and fusion of the sensor attitude data by the algorithm of the IMU gesture data fusion described above, the MPU6050, a 16-bit resolution attitude sensor that includes three-axis gyroscopes and accelerometers, is commonly used in the market. Reading of attitude data in the height IIC mode market. Reading of attitude data in the height IIC mode.
The attitude solution block diagram of the gradient descent method.

180° in the opposite direction back to the initial position, as shown in Figure 2(a–d).

The pose calculation data of all the above states are collected and saved. Use MATLAB simulation software to draw waveforms for data. The data waveform is shown in Figure 3.

The four waveforms stage (1)–(4) corresponds to the four states (a)–(d) of the RVIZ test. In the above test process, we can conclude that the algorithm can make attitude tracking conducted in a more steady and effective fashion, although there are minor fluctuations in the output angle.

3. Improved proposed distribution and adaptive resampling

The traditional RBPF algorithm (Lee, Sang, & Wan, 2011) is an effective method to solve the existential problems in simultaneous localization and mapping. The algorithm usually takes the odometry measurement model as the proposed distribution (Giorgio et al., 2007). Therefore, there are problems of large particle demand and complicated calculations. This article applies an improved algorithm. Integrating the odometry measurement information and the observation information of the laser sensor into the calculation of the proposed distribution will effectively reduce the number of required particles and reduce the uncertainty of the robot pose (Lee, Lee, Choi, & Lee, 2012). In addition, an adaptive resampling strategy (Havangi, Taghirad, Nekoui, & Tesnhehlab, 2014) is applied to reduce the number of particle sampling and greatly reduce the risk of particle exhaustion.

To obtain the particles for the next iteration, sampled particles need to be obtained from the proposed distribution $\pi$. Intuitively, the closer the proposed distribution is to the target distribution, the particle filter algorithm will perform better. When the odometer movement model is considered as a proposed distribution, the proposed distribution is expressed by $p(x_t^i | x_{t-1}^i, u_{t-1}, x_t^{(i)})$. $x_t^i$ represents

![Figure 1](image1.png)

**Figure 1.** The attitude solution block diagram of the gradient descent method.

![Figure 2](image2.png)

**Figure 2.** The IMU yaw angle fusion data are displayed on RVIZ. (a) Initial position. (b) The position of rotating 90°. (c) The position of rotating180°. (d) The position of rotating180° counterclockwise.
the waveform of the experimental data.

For the sake of overcoming the above problems, the measurement information of the lidar range sensor is integrated into the calculation of the proposed distribution \( p(x_t|m_{t-1}, x_{t-1}, z_t, u_{t-1}) \). \( z_t \) represents the observation information of the sensor at time \( t \). Its observation likelihood function has a small coverage. Particles that are close to the meaningful area have a high weight. Its weight is calculated by the following equations:

\[
    w_t^{(i)} = \frac{\eta p(z_t|m_{t-1}^{(i)}, x_t^{(i)}, x_{t-1}^{(i)}, u_{t-1})}{p(x_t^{(i)}|x_{t-1}^{(i)}, u_{t-1})} \cdot w_{t-1}^{(i)} \tag{22}
\]

\[
    w_t^{(i)} \propto w_{t-1}^{(i)} \cdot p(z_t|m_{t-1}^{(i)}, x_t^{(i)}) \tag{23}
\]

For the sake of overcoming the above problems, the measurement information of the lidar range finder is integrated into the calculation of the proposed distribution \( p(x_t|m_{t-1}, x_{t-1}, z_t, u_{t-1}) \). \( z_t \) represents the observation information of the sensor at time \( t \). Its observation likelihood function has a small coverage. Particles that are close to the meaningful area have a high weight. Its weight is calculated by the following equations:

\[
    w_t^{(i)} = \frac{\eta p(z_t|m_{t-1}^{(i)}, x_t^{(i)}, x_{t-1}^{(i)}, u_{t-1})}{p(x_t^{(i)}|x_{t-1}^{(i)}, u_{t-1})} \cdot w_{t-1}^{(i)} \tag{24}
\]

\[
    w_t^{(i)} \propto w_{t-1}^{(i)} \cdot p(z_t|m_{t-1}^{(i)}, x_t^{(i)}) \tag{25}
\]

Another key aspect of the impact on particle filter performance is resampling. In the resampling process, particles with lower importance weights \( w_t^{(i)} \) will be replaced by those with higher weights. On the one hand, there is a need to use a limited number of particles to assess the target distribution. Resampling, on the other hand, may remove particles that have higher weights. It will lead to
particle exhaustion. Therefore, the sampling time plays a decisive role in the process of resampling. In this experiment, Doucet’s method (Giorgio et al., 2007) was used to calculate this sampling time. $N_{\text{eff}}$ is expressed by the following equation:

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^{N} (\tilde{w}^{(i)})^2}$$  \hspace{1cm} (27)

$\tilde{w}^{(i)}$ refers to the normalized weight of particle $(i)$. Where $N_{\text{eff}}$ represents the degree of dispersion of importance weights, it is considered as an important indicator of the particle set approximation posterior probability function. During the application, a sample is taken each time $N_{\text{eff}}$ drops below the threshold. $N$ represents the number of particles.

4. Experiments

The odometer attitude data fusion algorithm and the improved RBPF algorithm described above were tested on a robot mobile platform based on an ROS system. For the sake of proving the effectiveness of the algorithm described above, the experiment was implemented in a series of different environments. The map generated by the above methods will not appear distorted and distorted. There will not be any obvious inconsistency between the map model and the actual environment. In the environment of a real-world data set, a relatively consistent and accurate map can be established.

To reduce the weight of the robot’s mobile platform and enhance its flexibility, a minicomputer equipped with the ROS system was chosen, as shown in Figure 4. The microcomputer is a Raspberry Pi 3B with a 1.2-core CPU based on the ARM A53 processor. The microcomputer is equipped with wireless WIFI and USB communication interfaces, providing convenience for remote control and data transmission. In the robot mobile platform system, the Raspberry Pi Microcomputer serves as the brain for data reception, processing, and behaviour commands. The mobile platform is equipped with a laser radar range finder to acquire environmental data, as shown in Figure 5. The STM32 microcontroller controls the movement of the geared motor with encoder, the integration of the odometer attitude data, and transfers the data to the upper raspberry function via the USB interface, which plays the role of cerebellum in the entire system.

In this paper, the mobile robot system is divided into a human–machine interaction layer, algorithm implementation layer and motion control layer according to functional modules. The structure of the system is shown in Figure 6.

We use a second ROS-mounted notebook computer to communicate with the microcomputer through the network protocol SSH command. This will facilitate the observation of real-time map data information and manipulation of the microcomputer Raspberry Pi. In other words, the data transmission between the notebook computer and the microcomputer is performed via WIFI. After the notebook computer is connected to the microcomputer, node information is run to start the entire experiment.

```bash
*ssh rikirobot@robot // Connect to the master on the notebook virtual side
*roscore // Open the overall system Master
*rosrun robot_serial_node // Transport protocol execution node
*rosrun slam_gmappingmapping // Mapping node
*rosrun robot_base// Motion control node
*rosrun rviz rviz // Visual tools.
```

The entire system consisting of a notebook computer and a Raspberry Pi only allows the existence of a node manager Master at the same time. Therefore, the master must be fixed on the running system of the Raspberry Pi. Otherwise, the entire system cannot be initiated. The notebook computer only plays the role of display and remote control in this system. The robot_serial_node is
a USB serial data transmission protocol node to ensure correct data transmission. The slam_gampping node is an implementation of the above-described improved RBPF algorithm. The robot_base node is a receiver and motion control node of odometer attitude fusion information. The RVIZ node is a visual graphical tool node.

The compositional environment for this experiment was carried out in the indoor environment of an office of the Robot Research Center of the university, covering an area of approximately 8 × 8 m. The mapping map of the environment is shown in Figure 7. The size of the grid selected is 1 × 1 m in RVIZ. Figure 8 shows the map constructed by the traditional RBPF algorithm. Because only the odometer model is used as the distribution of the proposal, the error is relatively large as the system running time increases. As a result, the constructed map appears to be significantly distorted (for example, \( q_1, q_2 \) are marked). The movement path of the robot is: \( 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \). However, after applying the above-mentioned odometer attitude data fusion algorithm and improved RBPF algorithm, the map constructed by the mobile platform of the robot obviously improves the distortion phenomenon in the traditional method environment mapping map. And a clearer texture map is built (as shown in Figure 9, \( s_1, s_2 \) marked. The robot path is \( a \rightarrow b \rightarrow c \rightarrow d \)).

Table 1 shows the number of particles and the average time required for the mapping to be constructed on the mobile robot platform after using the traditional particle filter algorithm and the improved algorithms. It can be seen from the comparison results that after optimizing the odometer attitude data and the improved RBPF, the number of particles required by the robot mobile platform in time of constructing a map is significantly

---

**Figure 6.** The block diagram of the robot mobile platform system structure.

**Figure 7.** The combination picture of the office environment.

**Figure 8.** The office map constructed using odometer information without the gradient descent algorithm and conventional particle filtering was used.
Figure 9. The office map constructed using the gradient-descent algorithm-optimized odometer information and particle filtering that improves the proposed distribution.

Table 1. Parameters list of building a consistency office map.

| Office experiment   | Number of particles | Average time /s |
|---------------------|---------------------|-----------------|
| Tradition approach  | 30                  | 460             |
| our approach        | 20                  | 260             |

Figure 10. The laboratory of the Robot Research Center.

Figure 11. The map of the Robot Research Center Laboratory.

Figure 12. The laboratory map constructed using odometer information without the gradient descent algorithm and conventional particle filtering was used.

Figure 13. The office map constructed using the gradient-descent algorithm-optimized odometer information and particle filtering that improves the proposed distribution.

As can be seen from Figure 12, the map shows significant distortion (q3 are marked). However, the problems were significantly improved by using the optimization algorithm from both the odometer and the particle filter (such as s3 in Figure 13). Comparing s4 and q4 (such as in Figures 12 and 13), Figure 13 has a clearer texture, after applying the gradient descent method and the improved RBPF algorithm.
Table 2 shows the number of particles and the average time required for the mapping in a complex lab environment. It can be seen from the comparison results that after increasing the area of environment, the number of particles and the average time required for the mapping are significantly increased. However, the proportion of growth of our method is significantly smaller than that of traditional particle filter that without using the improved proposed distribution and the gradient descent algorithm in mapping.

Through the above experiment and particle number analysis, we can conclude that after the traditional RBPF algorithm incorporates an observation model with relatively high odometer accuracy, a highly accurate proposed distribution that is closer to the target distribution can be calculated. This can greatly reduce the number of particles used. In addition, an adaptive re-sample strategy was applied to reduce the number of particle sampling and improve the precision of map construction as well as the efficiency of map construction. Finally, based on the open source code Grid mapping, the optimized algorithm is integrated into the code. The test and experimental result analysis on the self-built ROS robot mobile platform were also completed.

5. Conclusion

In this paper, we systematically optimize odometers and particle filters for grid map construction. On the one hand, we use the gradient descent algorithm to fuse the odometer attitude data. On the other hand, we use the sensor measurement information and odometry information to calculate the more accurate proposed distribution in the improved RBPF algorithm. This allows us to absorb particles in a more accurate way, which can greatly reduce the number of required particle samples. In addition, an adaptive resampling strategy is also applied, which can greatly reduce unnecessary sampling operations and avoid the risk of particle exhaustion. Finally, the above analysis method was tested and evaluated on the data acquired by a mobile robot platform equipped with an ROS system equipped with a laser range finder. The algorithm was transplanted to mobile platform with ROS for a series of experiments. The higher the precision map, the fewer the number of particles and the shorter average time required exhibit the effectiveness of the algorithm.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by the first batch of Shandong Province Key Research and Development Plan (No. 2016 GSF201197), Shandong Province Science and Technology Program of Higher Education (No. J16LB11) and National Natural Science Foundation of China Youth Fund (No. 61803235).

References

Arif, T. E., & Ali, O. E. (2015). Fusing inertial sensor data in an extended Kalman filter for 3D camera tracking. *IEEE Transactions on Image Processing*, 24(2), 538–548.

Doucet, A., Freitas, N. D., Murphy, K. P., & Russel, S. J. (2013). Rao-Blackwellised particle filtering for dynamic Bayesian networks. *Statistics*, 176–183.

Florian, H., Nico, K., Jeff, O., & Darius, B. (2010). Efficient occupancy grid computation on the GPU with lidar and radar for road boundary detection. Pages 1006–1013, 21-24 June.

Fourati, H., Manamanni, N., Afifal, L., & Handrich, Y. (2013, July 17-19). Position Estimation approach by complementary filter-aided IMU for indoor environment. 2013 European control conference (pp. 4208–4213).

Georg, T., & Dirk, W. (2017). Evidential grid-based tracking and mapping. *IEEE Transactions on Intelligent Transportation Systems*, 6(18), 1454–1466.

Giorgio, G., Cyrill, S., & Wolfram, B. (2007). Improved techniques for grid mapping with Rao-Blackwellized particle filters. *IEEE Transactions on Robotics*, 23(1), 34–46.

Havanghi, R., Taghirad, H. D., Nekoui, M. A., & Teshnehlab, M. (2014). A square root unscented FastSLAM with improved proposal distribution and resampling. *IEEE Transactions on Industrial Electronics*, 61(5), 2334–2345.

Kallapur, A., Petersen, I., & Anavatti, S. (2009). A robust gyroscope attitude estimation scheme for a small fixed-wing unmanned aerial vehicle. *Asian Control Conference*, 666–671.

Lee, H. C., Lee, S. H., Choi, M. H., & Lee, B. H. (2012). Probabilistic map merging for multi-robot RBPF-SLAM with unknown initial poses. *Robotica*, 30(2), 205–220.

Lee, J. S., Sang, Y. N., & Wan, K. C. (2011). Robust RBPF-SLAM for indoor mobile Robots using sonar sensors in non-static environments. *Advanced Robotics*, 25(9–10), 1227–1248.

Madgwick, S. O. H. (2010). An efficient orientation filter for inertial and inertial/magnetic sensor arrays. *ResearchGate*.

Madgwick, S. O. H., Harrison, A. J. L., & Vaidyanathan, R. (2011). Estimation of IMU and MARG orientation using a gradient descent algorithm. IEEE international conference on rehabilitation robotics 2011, 5975346.

Matthias, S., Volker, W., & Jürgen, A. (2014, May 31–June 7). Grid mapping in dynamic road environments: Classification of dynamic cell hypothesis via tracking. 2014 IEEE International Conference on Robotics and Automation (pp. 3995–4002).

Predrag, S. S., & Marko, B. M. (2010). Accelerated gradient descent methods with line search. *Numerical Algorithms*, 54(4), 503–520.

Sebastian, O. H. M., Andrew, J. L. H., & Ravi, V. (2011, June 29–July 1). Estimation of IMU and MARG orientation using a gradient
descent algorithm. 2011 IEEE International Conference on Rehabilitation Robotics.
Soltero, D. E., Schwager, M., & Rus, D. (2014). Decentralized path planning for coverage tasks using gradient descent adaptive control. The International Journal of Robotics Research, 33(3), 401–425.

Thrun, S. (2001). A probabilistic online mapping algorithm for teams of mobile robots. The International Journal of Robotics Research, 20(20), 335–363.
Yang, J. D., Yang, J. H., & Cai, Z. S. (2015). An efficient approach to pose tracking based on odometric error modelling for mobile robots. Robotica, 33(6), 1231–1249.