Article

Where Am I? SLAM for Mobile Machines on a Smart Working Site

Yusheng Xiang [1,2,3,*], Dianzhao Li [1], Tianqing Su [4], Quan Zhou [5], Christine Brach [2], Samuel S. Mao [3] and Marcus Geimer [1]

[1] Institute of Vehicle System Technology, Karlsruhe Institute of Technology, 76133 Karlsruhe, Germany; dianzhao.li@kit.edu (D.L.); marcus.geimer@kit.edu (M.G.)
[2] Division of Mobile Hydraulics, Robert Bosch GmbH, 89275 Elchingen, Germany; christine.brach@boschrexroth.de
[3] Department of Mechanical Engineering, University of California, Berkeley, CA 94720, USA; ssmao@berkeley.edu
[4] Guanghua School of Management, Peking University, Beijing 100091, China; t.su1991@gmail.com
[5] Vehicle and Engine Research Centre, University of Birmingham, Birmingham B15 2SQ, UK; q.zhou@bham.ac.uk
[*] Correspondence: yusheng.xiang@partner.kit.edu

Abstract: The current optimization approaches of construction machinery are mainly based on internal sensors. However, the decision of a reasonable strategy is not only determined by its intrinsic signals, but also very strongly by environmental information, especially the terrain. Due to the dynamic changing of the construction site and the consequent absence of a high definition map, the Simultaneous Localization and Mapping (SLAM) offering the terrain information for construction machines is still challenging. Current SLAM technologies proposed for mobile machines are strongly dependent on costly or computationally expensive sensors, such as RTK GPS and cameras, so that commercial use is rare. In this study, we proposed an affordable SLAM method to create a multi-layer grid map for the construction site so that the machine can have the environmental information and be optimized accordingly. Concretely, after the machine passes by the grid, we can obtain the local information and record it. Combining with positioning technology, we then create a map of the interesting places of the construction site. As a result of our research gathered from Gazebo, we showed that a suitable layout is the combination of one IMU and two differential GPS antennas using the unscented Kalman filter, which keeps the average distance error lower than 2m and the mapping error lower than 1.3% in the harsh environment. As an outlook, our SLAM technology provides the cornerstone to activate many efficiency improvement approaches.

Keywords: unscented Kalman filter; localization of construction machines; smart working site; SLAM; ROS

1. Introduction

Over the past several decades, the operation strategies aiming to increase mobile machines' holistic efficiency receive much attention. As Osinenko pointed out, the engine power of farm tractors is growing at 1.8 kW per year and has reached about 500 KW for the most powerful traction class [1], indicating the great potential to increase the holistic efficiency of the machines. Current solutions to optimize the system can be concluded as two kinds. To overcome the extreme conditions, the basic idea of the first method is equipping the mobile machines with an energy-saving system [2–4], such as accumulators. The other one is based on a dramatic controller to increase the dispensable power in a short period [5–10]. However, due to safety reasons, most of the operation strategies of mobile machines are designed conservatively. For instance, the energy-saving equipment should always at a relatively high level, and the controller must be designed to have rapid responses. As a result of that, the machines are

still very likely to work in suboptimal areas. In the previous studies, most of them utilize the intrinsic signals to optimize the system [11] since the environmental information is hardware or computationally expensive to be gathered. However, the environment also has an essential influence on the system, i.e., to perform tasks both efficiently and safely, the construction machines shall be optimized by knowing their location and surroundings; thus, we proposed a method that can generate the map information surrounding the mobile machines only with affordable sensors so that provides the possibility to improve the system further. The basic idea of our approach is to generate the map information of the working site based on the vehicle position, rolling resistance, as well as road grade. Concretely, a special recursive least square with forgetting algorithm is used to record the road grade and the rolling resistance in realtime [12,13]. This information will be saved together with the localization information. Consequently, after the machine passes by, it will record the information about that place. Since the mobile machines are driving repeatably for a special task, the method can be expected to work well even when the map information does not cover most of the working site. Figure 1 illustrates the motivation of our approach.
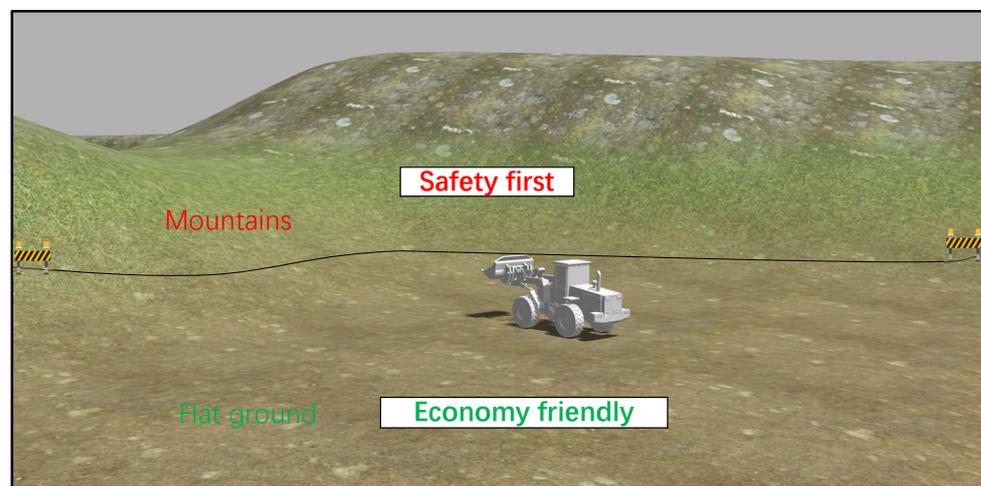

Figure 1. Environment based optimization. Mobile machines perform tasks more efficiently or safer according to their location and surroundings information. The short-term goal of SLAM is to prevent construction machinery from always working in low-efficiency areas for safety reasons, whereas the long-term goal is to increase the productivity of the working site with the help of path planning. The study focuses on affordable SLAM technology for construction machines. This work is also presented in my thesis [14].

2. Problem Statement and Goal of This Research

Although the map information can also be obtained from satellites, it is impossible to get the valuable information, such as a High Definition (HD) map, only depending on remote sensing due to a construction site's fast-changing environment. The sensors can also be quite noisy. They will especially be further exacerbated on a working site. The sensors, such as Global Positioning System (GPS), Inertial Measurement Unit (IMU), and odometry sensor will have higher measurement errors in the case of a harsh environment. In addition, different construction machines have a different drivetrain system, which makes a predefined motion model difficult. For instance, since mobile machines may work outside the coverage of base stations, the GPS signal can only achieve nearly 10 m accuracy [15,16] due to the lack of error correction. Last but not least, for passenger cars, the longitude error might not have such a negative effect as the latitude error since further measurements can be adopted to avoid the collision. In the case of construction machines, both errors shall be treated equally.

The goals of this paper are twofold. The first goal of the paper is to find out the most suitable sensor arrangements for construction machines. For accurate estimation of the

position of machines, rather than only trust the measurement from one sensor, we fused a series of different kinds of sensors with the help of sensor fusion technology, derived from a Kalman filter [17], to achieve better accuracy. Afterwards, the second goal is to create a map with the environment condition by combining the surface resistance, road grade, and position information in realtime. Thanks to this map, further optimization of operation strategy and path planning can be realized. Although we measure the surface resistance and road grade by recursive least square with multi forgetting factors, the map-building approach we proposed can also be combined with other methods with other kinds of sensors, such as using ultrasonic proposed by Jung [18].

3. Related Works

3.1. Sensors

The combination of several sensing systems so that they can compensate the technical shortcoming of each other is well-known in the field of autonomous systems [19–21]. Therefore, there are a series of researches focusing on sensor fusion. Here we first summary the commonly used sensors for simultaneous localization and mapping (SLAM). Although we agree the introduction of the HD map can surely increase the accuracy of the localization, we do not consider this technology for the construction machines due to the dynamically changing construction site, as mentioned in [22].

3.1.1. GPS

Global navigation satellite systems (GNSS) such as GPS, GLONASS, BeiDou, and Galileo rely on at least four satellites to estimate global position at a relatively low cost. Typical standalone GPS average accuracy ranges from a few meters to above 20 m [15] due to ionospheric delay, multipath effects, ephemetrics and clock errors, and Geometric dilution of precision (GDOP). To improve the accuracy, one of the most used techniques is Differential GPS (DGPS), which utilizes measurements from an onboard vehicle GPS unit and a GPS unit on a fixed infrastructure unit with a known location. Here, the known fixed infrastructure unit is called reference station, which calculates the local error in the GPS position measurement periodically. The onboard vehicle GPS units then use this correction to adjust their own GPS estimation. According to [23–27], an average accuracy in the range of 1–2 m can be achieved, mainly depending on the distance between the vehicle and the base station. Another commonly used improvement is realtime kinematic (RTK) GPS, which estimates relative position by means of the phase of the carrier signal and can be expected to achieve centimeter-level accuracy. Notice that, both of them depend on a fixed base station with a known position nearby, though the principle of them is quite different. In Oct 2020, when we write this paper, RTK GPS is still an extraordinary expensive approach and usually be used to define the ground truth position of vehicles. Some low-cost RTK GPS sensors, under 1000 USD, are designed with a much lower receiving frequency [28] and thus cause problems as vehicles driving fast. Thus, in most commercial uses, DGPS is preferable for reducing the cost. Therefore, in our research, we conservatively considered the accuracy of DGPS as 2 m, which is consistent with the normal performance of DGPS. Obviously, as the performance, especially the accuracy, of the GPS increases, our mapping approach will also consequently have a better performance.

3.1.2. IMU

Inertial measurement units (IMUs) are integrated electronic devices that contain accelerometers, magnetometers, and gyroscopes. It can provide raw IMU measurements to calculate attitude, angular rates, linear velocity, and position relative to a global reference frame.

3.1.3. Odometry

Odometry is the most widely used navigation method for positioning; it provides good short-term accuracy, is inexpensive, and allows very high sampling rates. Odometry is based on simple equations, which hold true when wheel revolutions can be translated

accurately into linear displacement relative to the floor. The main advantage of odometry is that all localization information comes from the vehicle itself so that this information is always available. Usually, it is the only localization information when other sensors are not able to provide data. Thus, a good odometry based localization system is always necessary, and it is usually the first step to localization [29].

3.2. Localization Technologies

3.2.1. Mobile Robotics

A series of studies using sensor fusion to achieve a highly accurate localization has been studied worldwide. The technologies about SLAM can be roughly divided into two parts: indoors and outdoors. For indoor localization, such as domestic robots [30], a GPS system cannot be used. Consequently, laser scans are preferable [31]. To enhance the overall performance, some effective methods to deal with outliers are proposed, such as the polynomial time (O(N2) outlier removal method [32]. Since the road is relatively flat, only a two-dimensional map is needed. In contrast, when it comes to offroad navigation in rough terrain, the algorithms must be capable to handle three dimensions of the environment. After the success of Kalman filter [17], extended Kalman filter [33], and finally unscented Kalman filter [34,35], the idea that a mobile robot which executes useful missions should be endowed with navigation ability has become a consensus. However, the selection of combining different sensors is different from case to case. For instance, Bento fused the data from ABS sensors and GPS for outdoor localization, based on extended Kalman filter [36], and Zhang integrated the information from GPS and IMU [37]. Li also used a camera instead of GPS to accomplish mean positioning errors of 75 cm [38]. In addition, Wolcott proposed a Visual Localization method within LIDAR Maps for Automated Urban Driving [39]. For cost purposes, Ward studied the possibility of using radar to localize the vehicle's position and demonstrates that errors go to 27.8 cm laterally and 115.1 cm longitudinally by their approach in worst case [40]. To investigate the use of LiDAR for localization, Hata suggests using LiDAR to detect curbs and road markings to create a feature map of the environment and localize vehicles with the help of RTK-GPS and IMU within the map [41]. Another alternative is using ultrasonic sensors, proposed by Jung [18]. Interestingly, with the development of the internet of things (IoT), more and more researchers are focusing on SLAM by cooperative localization techniques. The basic idea of this approach is to get crucial information even when the perception capability is affected by adverse weather or obstacles from infrastructure or other vehicles. For example, del Peral-Rosado showed the feasibility of 5G based localization technology [42], and Rohani utilized VANET to enhance GPS accuracy to 3.3 m mean level [43]. Further information and applications can be found in the survey paper from Kutti [44] and Ting Goh [45].

3.2.2. Construction Machines

For mobile construction machines, the requirements for localization techniques can be different based on different use cases. Some machines may work in the underground, where the situation is similar to working in a tunnel, whereas others might work on an open-pit mining site. In underground mines, it was proposed to use the laser for extracting the wall positions, and dynamically generate a path from these laser data while considering variable offsets [46,47]. In contrast, for the open-pit site, an autonomous wheel loader introduced by Gu [48] uses a set of sensors, including GPS and IMU for localization, and LIDAR, radar as well as a camera for obstacles capture and identification, ensuring it perceives surroundings accurately. Moreover, Xiang created a dataset for mobile machines detection from the view of a camera fixed on the ground [49], while Bang proposed a method recognizing the machines from a view of the drone [50]. In addition, the visual SLAM is proposed [51]. Besides that, V2X technology was introduced in the field of construction machines [52,53]. In 2020, Xiang proposed to use WIFI to achieve the communication between different vehicles by introducing a realtime estimation method with respect to package loss and delay [54]. Afterwards, the feasibility of using 5G for machines is also investigated by [55].

To avoid additional costs for the vehicles, smartphones show great potential to be utilized as a solution to complement the flaws of onboard ECU [56,57].

In summary, similar to general autonomous vehicles, most automated mobile machines fuse information from onboard sensors such as IMU and GPS by using diverse sensor fusion technologies. Furthermore, camera, LIDAR, and radar are used to detect the environment in the construction site, to avoid obstacles, and to instruct the machines where to go. However, owing to the harsh environment and diversity of working sites, LIDAR and radar can be sensitive. In the recent future, wireless communication can also contribute to better localization of vehicles.

4. Model Building

The wheel loader used in the simulation was modeled in Solidworks and then imported into the Robot Operating System (ROS) to explore the approaches that should be used for mobile machines. Since the first goal of this paper is to find out suitable sensor arrangements to accurately localize wheel loaders, we fused different arrangements of sensor data. Concretely, we used up to three IMUs and three GPSs in the simulation. Based on the characteristic of GPSs, three GPSs were fixed on the cab of the wheel loader. We then installed two IMU sensors under the front axle and other IMUs under the rear axle. Figure 2 illustrates the wheel loader model we used in the Gazebo environment.
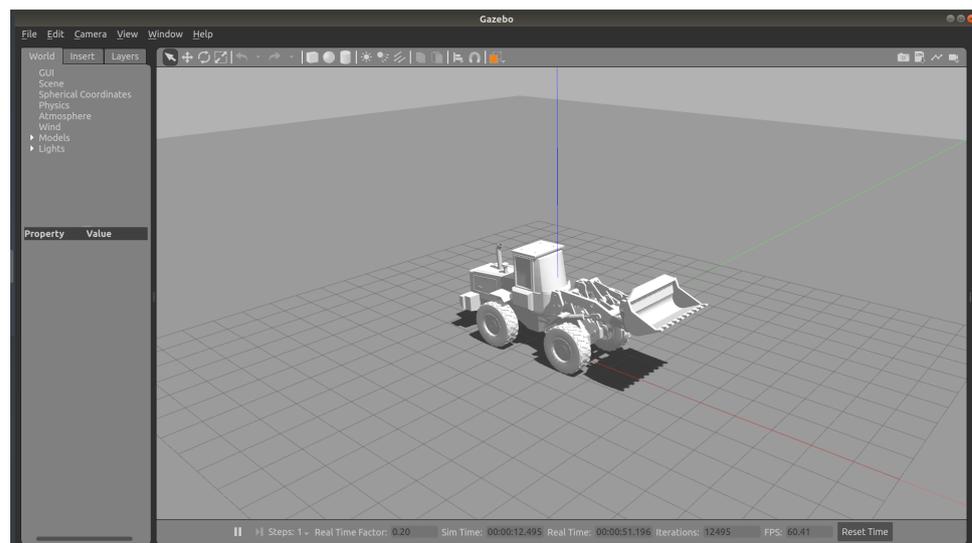

Figure 2. Wheel loader model in Gazebo: once the models had been developed in Solidwork, they were converted to Unified Robotic Description Format (URDF), using a 3rd party URDF conversion tool called "*sw_urdf_exporter*", which allows for conveniently exporting SW Parts and Assemblies into a URDF file. Gazebo enables us to obtain sensors' simulation such as IMUs, GPSs, encoders, cameras, and stereo cameras through *gazebo_plugins*, which can be used to attach into ROS messages and service calling the sensor outputs, i.e., the *gazebo_plugins* create a complete interface (Topic) between ROS and Gazebo.

As we know, the URDF is an XML file format used in ROS to describe all elements of a vehicle. URDF can specify the kinematic and dynamic properties of a single robot in isolation. To make our vehicle works properly in Gazebo, additional simulation-specific tags concerning the vehicle pose, frictions, inertial elements, and other properties have been added. The transform tree is shown in Figure 3.
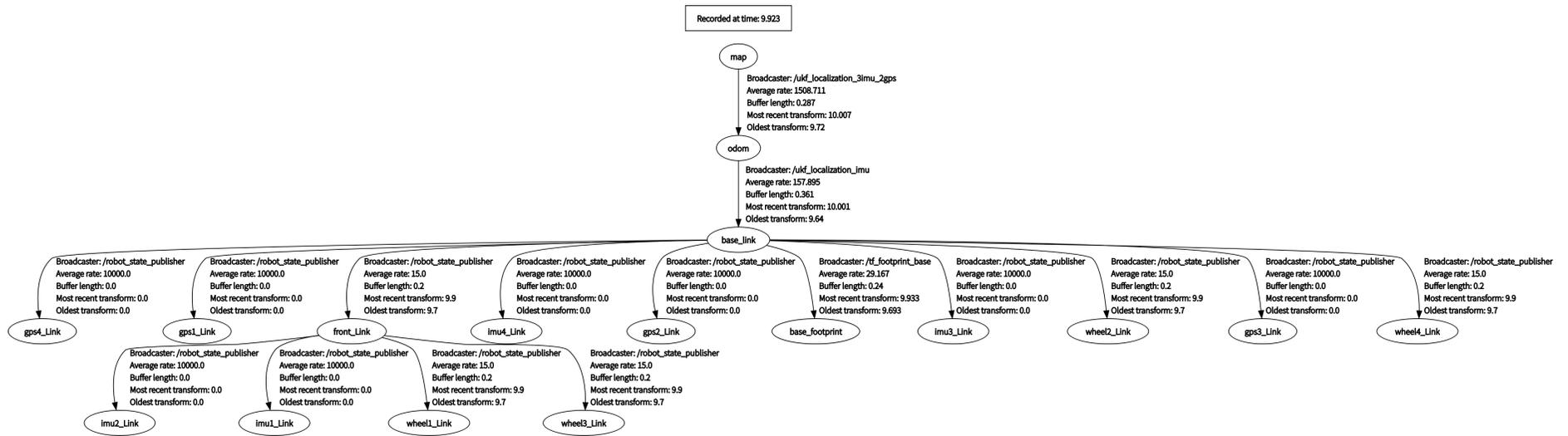
Figure 3. The dynamic system simulated by URDF file on ROS.

Each Link in URDF represents a rigid body. According to the kinematic and dynamic model shown in Figure 3, the wheel loader in the simulation is divided into several parts, including:

1. *base_link* represents the rear part of the wheel loader, which is also the main coordinate frame of the simulated model in ROS, because most of the sensors are attached to the rear part and considered as child links;
2. *front_link* represents the front part of the wheel loader, considered as the child link of the *base_link*, and connected by a revolute joint;
3. *wheel_link* represents the wheels of the wheel loader. Apparently, the two front wheels are attached to the front part, and the two rear wheels are attached to the rear part;
4. *gps_link*: the GPS devices, which were fixed on the roof of the wheel loader;
5. *imu_link*: the IMU devices, two of which were fixed under the front part of the wheel loader and the other two fixed under the rear part of the wheel loader.

In this project, our wheel loader receives GPS data from an onboard GPS sensor plugin with its latitude and longitude; however, the GPS data provided by the GPS plugin cannot be directly applied to the fusion of the sensor data, so coordinate system conversion for GPS data is required. For our simulation, we set a transform for each GPS that converts the vehicle's world frame coordinates, i.e., the frame with its origin at the vehicle's initial position, to the GPS's UTM coordinates, the same as [58], as:

$$
\mathrm{T} = \begin{bmatrix}
c(\theta)c(\psi) & c(\psi)s(\phi)s(\theta) - c(\phi)s(\psi) & c(\phi)c(\psi)s(\theta) + s(\theta)s(\psi) & x_{UTM_0} \\
c(\theta)s(\psi) & c(\phi)c(\psi) + s(\phi)s(\theta)s(\psi) & -c(\psi)s(\phi) + c(\phi)s(\theta)s(\psi) & y_{UTM_0} \\
-s(\theta) & c(\theta)s(\phi) & c(\phi)s(\theta) & z_{UTM_0} \\
0 & 0 & 0 & 1.
\end{bmatrix}
\tag{1}
$$

where ϕ, θ, ψ denote the vehicle's initial UTM-frame roll, pitch, and yaw. c and s denote the cosine and sine functions, respectively. x_{UTM_0}, y_{UTM_0}, and z_{UTM_0} are the UTM coordinates of the first reported GPS position. After that, the GPS signal is then transformed into the vehicle's world coordinate frame, *odom*, by:

$$
\begin{bmatrix}
x_{odom} \\
y_{odom} \\
z_{odom} \\
1
\end{bmatrix}
= \mathrm{T}^{-1}
\begin{bmatrix}
x_{UTM_t} \\
y_{UTM_t} \\
z_{UTM_t} \\
1.
\end{bmatrix}
\tag{2}
$$

In our simulation, the ROS package "*robot_localization*" from Moore [58] was used, including a "*navsat_transform*" node, which provides functions to convert between various coordinate frames and integrate GPS data. It provided a transformation function that allows the conversion between GPS frame, expressed in latitude and longitude, and vehicular coordinates. This process shall be carried out for each GPS independently.

In practical applications, GPS signals can be received infrequently. Yet the localization technology must maintain state estimation even when some of the vehicles' signals are absent. Therefore, we shall evaluate the performance of the filters when GPS signals infrequently arrive in our environment. Taking this problem into consideration, we build a ROS node to filter the collected GPS signals such that GPS data are unavailable for 1 second once every 10 seconds. In case multi GPS sensors are used, the signal failure might not happen at the same time. We aim to observe how the filter and our approach behave with different sensor configurations when some GPS signals go wrong.

4.1. Sensor Fusion for Localization

For localization of the wheel loader in the simulation environment, we used EKF and UKF node in "*robot_localization*" [58]. On the one hand, this package has no limitation on the number of sensor inputs, which is just in line with our construction machines' requirements. On the other hand, a concrete motion model is not needed so that this

method can easily be used on both excavators and wheel loaders with different drivetrain solutions. In "*robot_localization*", the filter's state will be driven forward by a standard 3D kinematic model derived from Newtonian mechanics to calculate the vehicle's motion, including position, velocity, and acceleration in three dimensions.

In the correction step, the measurement model integrates sensor data to update the predicted state. GPS provides position information, and wheel encoders provide velocity information for the correction. Moreover, the orientation, velocity, and acceleration information are updated from IMUs via gyroscopes and accelerometers. Furthermore, the process model and measurement model also need to add a noise covariance matrix, Q and R, respectively. The noise matrices can generate uncertainty in the system. The process matrix contributes to the overall uncertainty in the algorithm, which adds to the process model. Intuitively, a large value in the Q matrix means a considerable uncertainty in the process model, causing the system to have greater confidence in the measurement data. In the current implementation, the process noise matrix was set as a diagonal matrix. The state variables, which are directly measured by the sensors, such as x y position by GPS and orientation by IMUs, were set relatively small. The variables, which were not directly measured, could be updated from the measured data. The measurement covariance matrix R corresponding to the confidence in the sensor data. Similarly, the greater the noise in the elements of this matrix, the less confidence in the measurement data. The measurement covariances are derived by the sensors noisy.

In addition to the inaccuracy of filters, outliers are also an important source of error. In our simulation, we assume that the measurements have Gaussian distributions. Although sensors follow the normal distribution as setting, improbable, and extremely noisy measurements can appear due to the high fidelity of ROS. To counteract this problem, we used Mahalanobis distance [59] to detect outliers and thus overcome the consequent adverse effect. After this process, the filtered data were used for state correction. Concretely, the Mahalanobis distance is calculated as a product of the from filter processed vector to find out the outliers:

$$D_M(\vec{x}) = \sqrt{(Z_t - \overline{Z}_t)^{\mathrm{T}} A^{-1} (Z_t - \overline{Z}_t)}, \tag{3}$$

where A is the covariance matrix. Notice that we also recommend to use the adaptive M-estimators [60] in case the percentage of outliers is higher than 50%.

4.2. Sensor Fusion Methods

State estimation is one of the most critical issues in many autonomous applications. Having an accurate state estimation, we can effectively navigate the machines in the environment and thereby make optimal decisions for specific purposes. For instance, to reach a target destination, it needs to know its current state, which consists of position, velocity, acceleration, and heading to execute the right maneuvers correctly. Since sensors are susceptible to noise and imperfections introducing uncertainty to the measurements, the filter's goal is to fuse all the available sensor data, as well as the vehicle's own dynamics to obtain a more precise estimation of the vehicle's state. As mentioned, we present two necessary extensions of the Kalman filter, notably the EKF and UKF.

The filters are modeled to improve the positioning's accuracy by compensating for the disadvantages of the different sensors. As we know, GPS provides relatively accurate positioning, but the signal's availability remains a problem, especially in urban and mountainous environments. This determines that the results of using GPS alone are usually not satisfactory. IMUs also use a combination of accelerometers and gyroscopes to measure linear accelerations and angular velocities, respectively. By estimating the position relative to its initial position, the trajectory can be calculated by the information of the wheel loader as the vehicle travels. For sure, there is also a common problem with IMUs, namely the accumulated errors. To avoid accumulated drift and provide global positioning, the estimated position shall be corrected by using other sensors. Without a doubt, a GPS/IMU system is successful in increasing the accuracy beyond standalone GPS or IMU capabilities.

4.2.1. Extended Kalman Filter

Although linear Gaussian systems are abundant, most systems, in reality, are non-linear. They also often do have Gaussian noise. Wrong assumptions about the system can lead the Kalman filter to diverge and provide estimation with very high errors. Consequently, multiple extensions have been developed to deal with various scenarios encountered in practice. One of the famous variations is the EKF, where it deals with non-linearity by approximating a linear equivalent before performing the required filtering sequence. The idea of the EKF is that if the system is close to linear for short periods, using its linear approximation will then not yield large errors.

Through linearizing the basic equations from Welch [35], we get the following equations:

$$x_{k+1} \approx \tilde{x}_{k+1} + A(x_k - \hat{x}_k) + Ww_k \tag{4}$$

$$z_k \approx \tilde{z}_k + H(x_k - \hat{x}_k) + Vv_k, \tag{5}$$

where x_{k+1} and z_k are the actual state and measurement vectors, $\tilde{x}_{k+1}$ and $\tilde{z}_k$ are the approximate state and measurement vectors. $\hat{x}_k$ is an *a posteriori* estimate of the state at step k, random variables w_k and v_k represent the process and measurement noise. A is the Jacobian matrix of partial derivatives of $f(\bullet)$ with respect to x, W is the Jacobian matrix of partial derivatives of $f(\bullet)$ with respect to w, H is the Jacobian matrix of partial derivatives of $h(\bullet)$ with respect to x, V is the Jacobian matrix of partial derivatives of $h(\bullet)$ with respect to v.

An essential feature of the EKF is that the Jacobian H_k in the equation for the Kalman gain K_k serves to correctly propagate only the relevant component of the measurement information [35]; the linearization error always exists as the function is nonlinear. Because increasing the sampling time and reducing the nonlinearity of function are not always viable, error-state EKF is proposed to counteract the adverse effect. The basic idea of error-state EKF is to reduce the distance of the linear approximation from the operating point; instead of linearization of the nominal state, it handles the error state.

4.2.2. Unscented Kalman Filter

When the state transition and observation models, that is, the predict and update functions f and h are highly nonlinear, the EKF can give particularly poor performance. This is because the covariance is propagated through the linearization of the underlying nonlinear model. By contrast, the UKF uses unscented transform instead of linearization in the prediction and correction steps to make the estimation. As the first step, we shall compute the decomposition of the covariance matrix and carefully select the sample points, for instance, here we selected 2L + 1 points, described as:

$$
\begin{aligned}
\mathcal{X}^0 &= \bar{x} \\
\mathcal{X}^i &= \bar{x} + (\sqrt{(L+\lambda)P_x})_i & i = 1, \cdots, L \\
\mathcal{X}^{i+L} &= \bar{x} - (\sqrt{(L+\lambda)P_x})_i & i = 1, \cdots, L
\end{aligned} \tag{6}
$$

where $\bar{x}$ is the selected mean sample point and λ is set as $\lambda = 3 - N$ for Gaussian probability density function. After that, the sigma points will be propagated through the nonlinear function as:

$$\check{X}^i = f(\mathcal{X}^i) \qquad i = 1, \cdots, 2L, \tag{7}$$

where $f(\bullet)$ is the motion model function and $\check{X}^i$ is the predicted position based on the motion model. As the final step of predicted process, the predicted mean and covariance should be calculated.

$$
a^i = \begin{cases} \lambda/L + \lambda, i = 0 \\ \lambda/2(L+\lambda), otherwise \end{cases} \tag{8}
$$

$$\check{X} = \sum_{i=0}^{2N} a^i \check{X}^i \tag{9}$$

$$\check{P} = \sum_{i=0}^{2N} a^i (\check{X}^i - \check{X})(\check{X}^i - \check{X})^T + Q. \tag{10}$$

Here, Q denote the noisy of this process. In the correction step, we firstly calculate the predict measurement with the sigma points,

$$\hat{y}^{(i)} = h(\check{X}^i, 0), i = 0, \ldots, 2N, \tag{11}$$

where $h(\bullet)$ is the predicted measurement model, considering the process noise: $\check{L}\check{L}^T = \check{P}$. Then, we get the mean and covariance of predicted measurements,

$$\mathcal{Y} = \sum_{i=0}^{2N} a^i \mathcal{Y}^i \tag{12}$$

$$\check{P}_y = \sum_{i=0}^{2N} a^i (\mathcal{Y}^i - \mathcal{Y})(\mathcal{Y}^i - \mathcal{Y})^T + R, \tag{13}$$

where R is the noisy. Based on the previous result, we can compute the cross-covariance and Kalman gain, and then get the corrected covariance and mean.

$$P_{xy} = \sum_{i=0}^{2N} a^i (\check{X}^i - \check{X})(\mathcal{Y}^i - \mathcal{Y})^T \tag{14}$$

$$K = P_{xy} P_y^{-1} \tag{15}$$

$$\hat{P} = \check{P} - K P_y K^T \tag{16}$$

$$\mathcal{X}^i = \check{X} - K(y^0 - \mathcal{Y}^i), \tag{17}$$

where y^0 is the measurement result with respect to $\check{X}$, $\mathcal{X}^i$ is the final results from UKF. As we can see, UKF does not need Jocobian matrix so that it can achieve a better performance in estimation. Usually, the computation effort can be slightly higher than the other methods, making carefully select sensors configurations meaningful.

4.3. Realtime Map Plotter

Inspired by the research from Fankhauser [61], a map can include many layers to store different types of data information. Thus, to develop a realtime plotter of the construction site according to ground condition, we use a multi-layer grid-based map, which divides the environment into uniform cells. Figure 4 illustrates the multilayered grid map concept, where each cell's data are stored on the congruent layers. In this project, since resistance and grade of the road are the most important information for the construction machines, we adopt a two layers grid map. Concretely in our study, the map is divided into small cells, whose resolution is 1 meter per cell. As we discussed in the previous section, we use GPS/IMU fusion Kalman filter algorithms to locate the mobile machine in the construction site.

As can be seen in Figure 4, to describe the ground condition of construction sites, we use the values of each cell to represent the information of the ground situation. In Figure 5, a layer that holds the data of a grid-based map is shown. Obviously, although we only demonstrate the approach with two layers, it is relatively easy to extend the third layer in the case that uphill or downhill is vital, as the third layer is responsible for recording the heading of the vehicle.
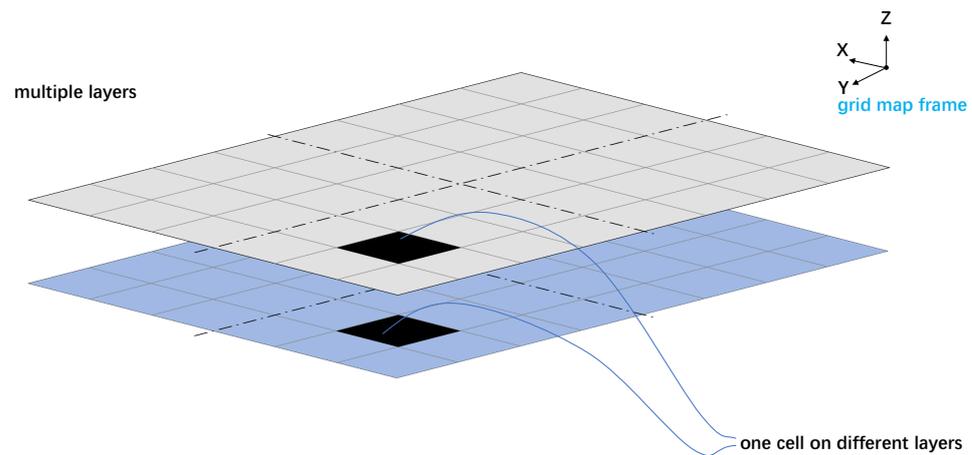
Figure 4. Our approach uses multilayered grid maps to store data for different types of information. Concretely, every grid saves a 1×3 matrix including location information and resistance or grade, depending on which layer it is. A grid with site information will be created after the vehicle passes by. The map is saved as a $2 \times m \times n \times 3$ tensor, where m is the max displacement in the x-direction, while n is the max displacement in the y-direction. In case a grid is not occupied once by the vehicle, it will be marked as NaN to denote the unknown regions.
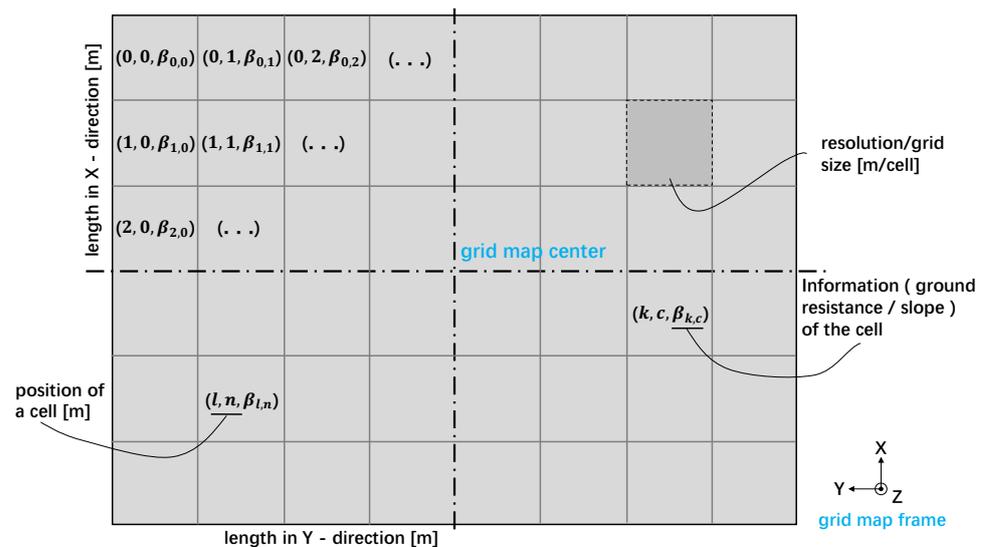
Figure 5. Detail description of a layer in the grid-based map.

Based on the previous study, both resistance and grade of the road can be gathered in realtime. Thus, we assume that the ground resistance and slope are known after the mobile machine passed by. When the mobile machine passes through each grid cell, ground information will be added to the corresponding grid of the two layer-grid-maps. Concretely, the plotting algorithm combines the localization results from the Kalman filter and the grade and resistance information from the recursive least square algorithm. To implement the plotting algorithm in ROS, we created a *node* in ROS that can subscribe to the localization results from the Kalman filter node and gather the current information from ground truth maps with the assumption that resistance and grade of the road can be estimated or measured well.

5. Vehicle Simulation Scenarios in ROS and Gazebo

To test the feasibility of the map plotter, we first define different road conditions in a construction site. ROS is a mature and flexible framework for robotics programming, providing the required tools to easily access sensors data, process that data, and generate

appropriate responses for the robot's actuators. Due to these characteristics, ROS is a perfect tool for many types of research on modern robotics. After all, a mobile machine can be considered as just another type of robotics, so the same types of programs can be used to develop advanced construction machines. In this paper, we simulated a construction site showed in Figure 6. According to this real construction site, we define five different ground resistances in the simulation environment, according to different ground conditions. Moreover, two areas according to different slopes. Concretely, we distinguish the different regions based on road material.

1. Gravel surface: Gravel is a loose aggregation of small, variously sized fragments of rock. It has a wide range of applications in the construction industry. Therefore, gravel surface road is very common in the construction site. The rolling resistance coefficient of gravel surface is considered to be 0.02;

2. Sand surface: Sand is a type of naturally material that is of a loose, granular, fragmented composition, consisting of particulate things such as rock, coral, shells, and so on. The rolling resistance coefficient between the sand surface and mobile machine tires is 0.250;

3. Dry dirt road: A dirt road is a type of unpaved road made from the native material of the land surface, which is also very normal in the construction site. The typical rolling resistance coefficient of the dry dirt road is 0.040;

4. Wet dirt road: Same as a dry dirt road, the wet dirt road is also a typical road type in the working area. The typical rolling resistance coefficient of the wet dirt road is 0.060;

5. Dry concrete surface: Dry concrete is a normal building material and the typical rolling resistance coefficient of the dry concrete road is 0.008.

Besides, we also define two different ground slopes appropriately. Since we only use a two-layer grid map in this research, we do not discriminate uphill or downhill.

1. Flat area: The slope of the ground is near 0°. In the flat area, we can let the mobile machines move faster to increase working efficiency or reduce the reserved dynamics to let the components work in a more economic region, with only a little concern for safety;

2. 15° slope area: The slope of this area is near 15°. In this area, in contrast, the mobile machines should pay more attention to the safety.

For simulation, we defined the green area as the dry concrete surface in the ground-truth resistance map. The red area represents the gravel surface. The blue area represents the sand surface, the black area and the yellow area represent dry and wet dirt roads, respectively. With this premise, we draw the ground truth map upon this construction site. Same as in the ground-truth resistances map, we also define the ground slope map, where the green area represents the flat area and the red area represents the 15° slope.

In this project, we wrote the *plot_node* with Python and OpenCV library to visualize the plotted map. To test the feasibility of a realtime plotter in simulation, we use a *ground_truth* node in ROS, which provides a ground truth position of the simulated mobile machine in Gazebo. When the mobile machine moves to a certain position, we use the ground truth position to determine the rolling resistance coefficient and the road grade, and then use the Kalman filter filtered position to plot the corresponding information in the grid-based map.

In ROS and Gazebo, the build-in plugins provide many adjustable parameters that can be used to adjust the devices' performance. To get closer to reality, we set the performance parameters of GPSs and IMUs in the simulation according to real GPS and IMU devices. To get the best sensor configuration, we implement different sensor configurations for different algorithms. Each group of sensor configurations is simulated under the same condition by rosbag, and the results are output simultaneously.
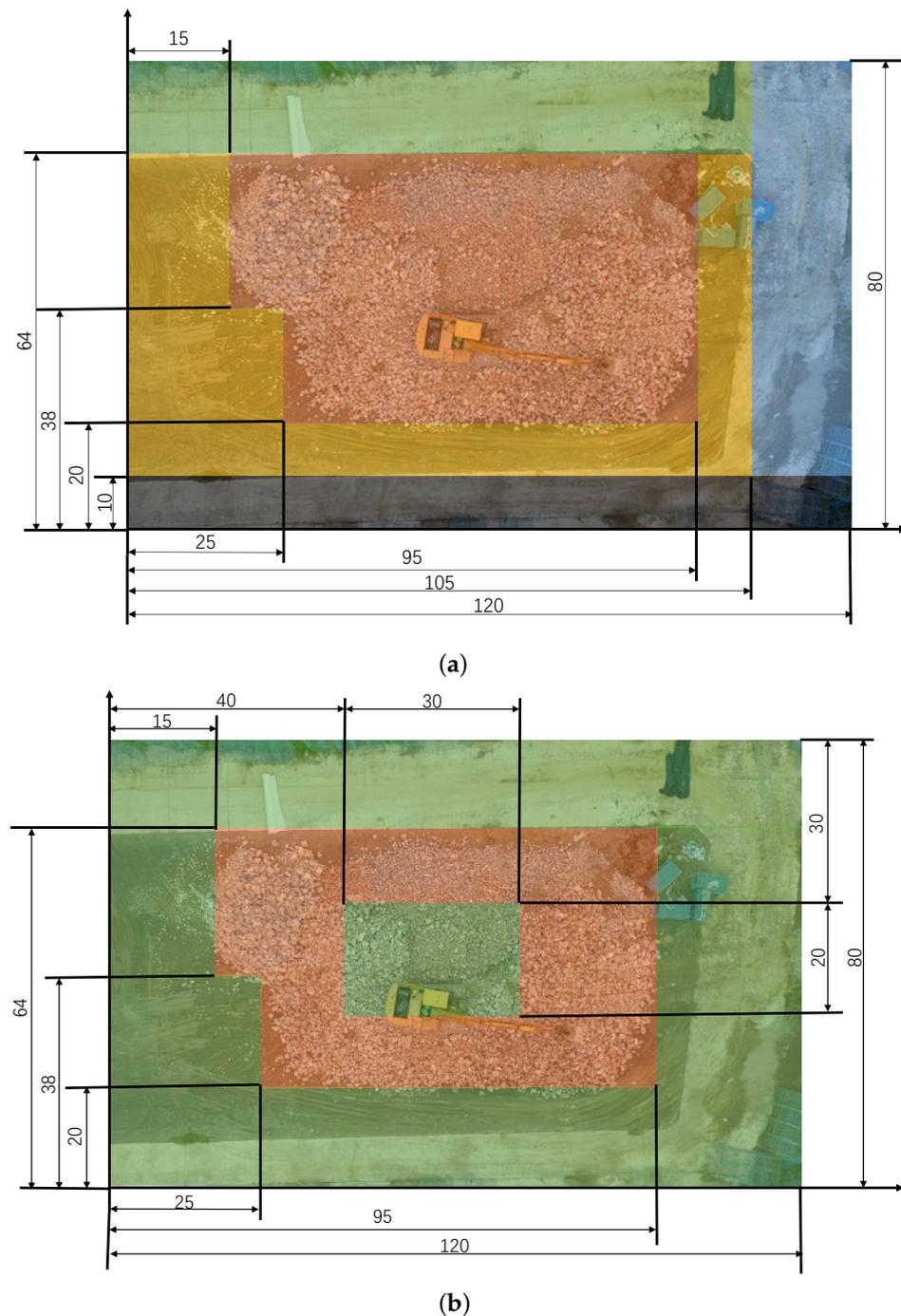
(a)

(b)

Figure 6. The ground truth map with dimensions. The simulation environment we used in Gazebo was modeled based on a real construction site, and the parameters are selected according to material characteristics. Since simulating a small construction site may cause system error and thus lack plausibility, we augmented this real construction site's dimensions in Gazebo. (**a**) Example construction site divided into five areas according to different ground resistances; (**b**) Example construction site divided into two areas according to different slopes.

6. Experiment and Results

6.1. Localization Results

To explore the most suitable sensor arrangements of the Kalman filter for construction machinery, we compared the results from different sensor configurations with different methods. The concrete sensor arrangements in this project are shown in Table 1.

Table 1. Sensor configurations.

Group	Sensor (0 = Deactivated, 1 = Active)							KF
	GPS 1	**GPS 2**	**GPS 3**	**IMU 1**	**IMU 2**	**IMU 3**	**Encoder**	
1	0	0	0	1	0	0	1	EKF
2	1	0	0	1	0	0	1	EKF
3	1	0	0	1	1	0	1	EKF
4	1	0	0	1	1	1	1	EKF
5	1	1	0	1	0	0	1	EKF
6	1	1	0	1	1	0	1	EKF
7	1	1	0	1	1	1	1	EKF
8	1	1	1	1	1	1	1	EKF
9	0	0	0	1	0	0	1	UKF
10	1	0	0	1	0	0	1	UKF
11	1	0	0	1	1	0	1	UKF
12	1	0	0	1	1	1	1	UKF
13	1	1	0	1	0	0	1	UKF
14	1	1	0	1	1	0	1	UKF
15	1	1	0	1	1	1	1	UKF
16	1	1	1	1	1	1	1	UKF

In our simulation, we ignore the mirror difference caused by the slightly different installation position of sensors. Therefore, the various configurations are reduced from 64 to 16. Since odometry is robust and necessary for many applications, we do not consider the case without an encoder.

To evaluate the performance of the different variants of sensors and algorithms concerning accurate positioning, we controlled the wheel loader to drive on the previously defined construction site in Gazebo, and recorded the data from sensors at the same time. Afterward, the localization results of the different methods are compared to the ground truth. Here we use the root-mean-square error (RMSE) as a quantitative indicator of the error to assess the pose estimation results.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} \left(\tilde{S}_i - \hat{\tilde{S}}_i \right)^2}{n}}, \tag{18}$$

where $\hat{\tilde{S}}$ is the vector including estimated x and y position, $\tilde{S}$ denotes the ground truth x and y positions, n is the number of all estimated samples, and the footnote i denotes the ith time step.

Figure 7 shows the wheel loader estimated trajectories given by different approaches and the ground truth trajectory, where the red lines are the ground truth trajectories that the vehicle passes, and the blue lines are the estimated positions of the vehicle. Since only EKF and UKF are capable of handling the nonlinear problem, we compare the results obtained using EKF filtering and the UKF filtering technique with different sensor arrangements. Notice that the odometry sensor is always used though we do not explicitly mention it. Apparently, the UKF performs better than the EKF, which is also in line with the conclusion from most studies. Generally speaking, with GPS fusing in the estimation, the accuracy improves drastically. Since the GPS may lose signal every 10 seconds, an additional GPS sensor can surely increase the position accuracy. In contrast, more IMUs can only slightly improve the accuracy of positioning. As we can see, the IMU + Odometry estimation method yields an inferior performance no matter with EKF or UKF. Once the IMU reports an inaccurate heading, the errors will be accumulated, causing the measured position to drift further away from its true position as the wheel loader travels.
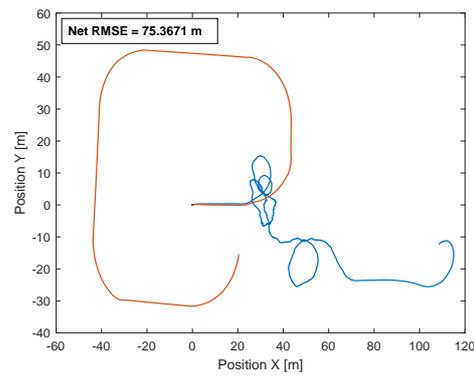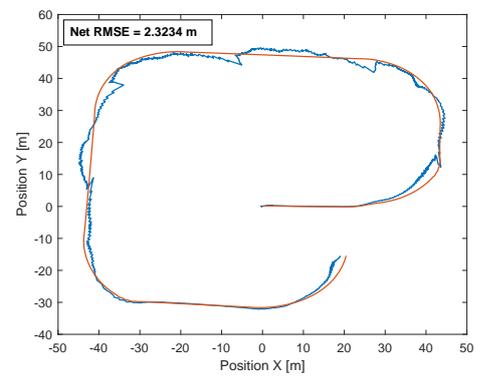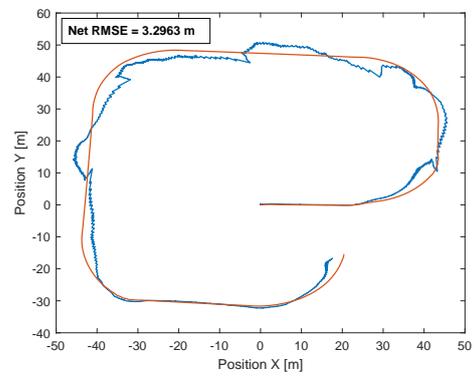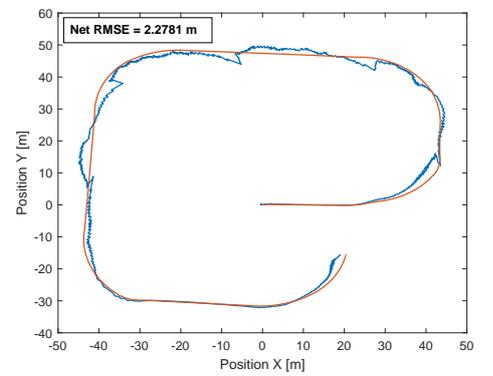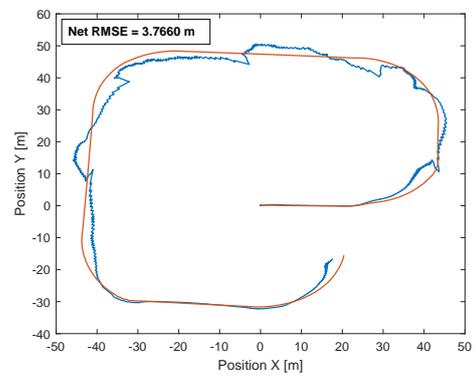
(**a**)
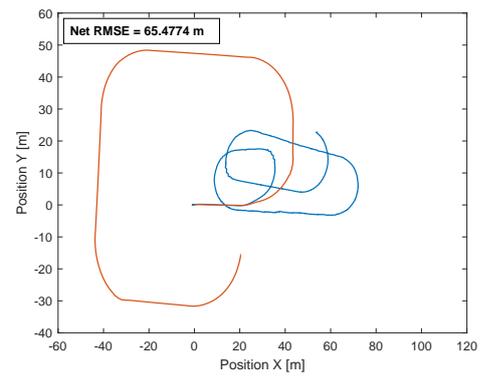
(**b**)
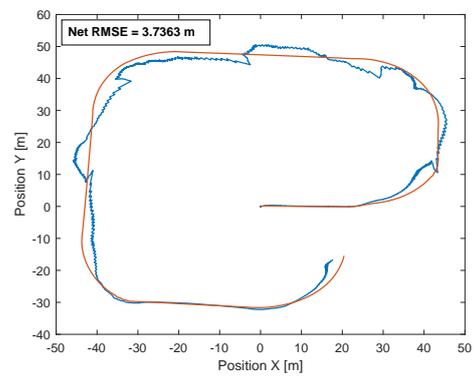
(**c**)

(**d**)

(**e**)

(**f**)

(**g**)

(**h**)

Figure 7. *Cont.*

(**i**)

(**j**)

(**k**)

(**l**)

(**m**)

(**n**)

(**o**)

(**p**)
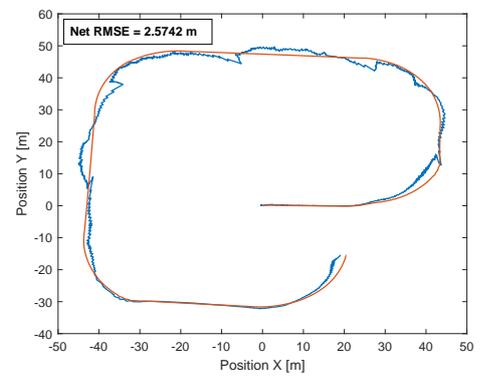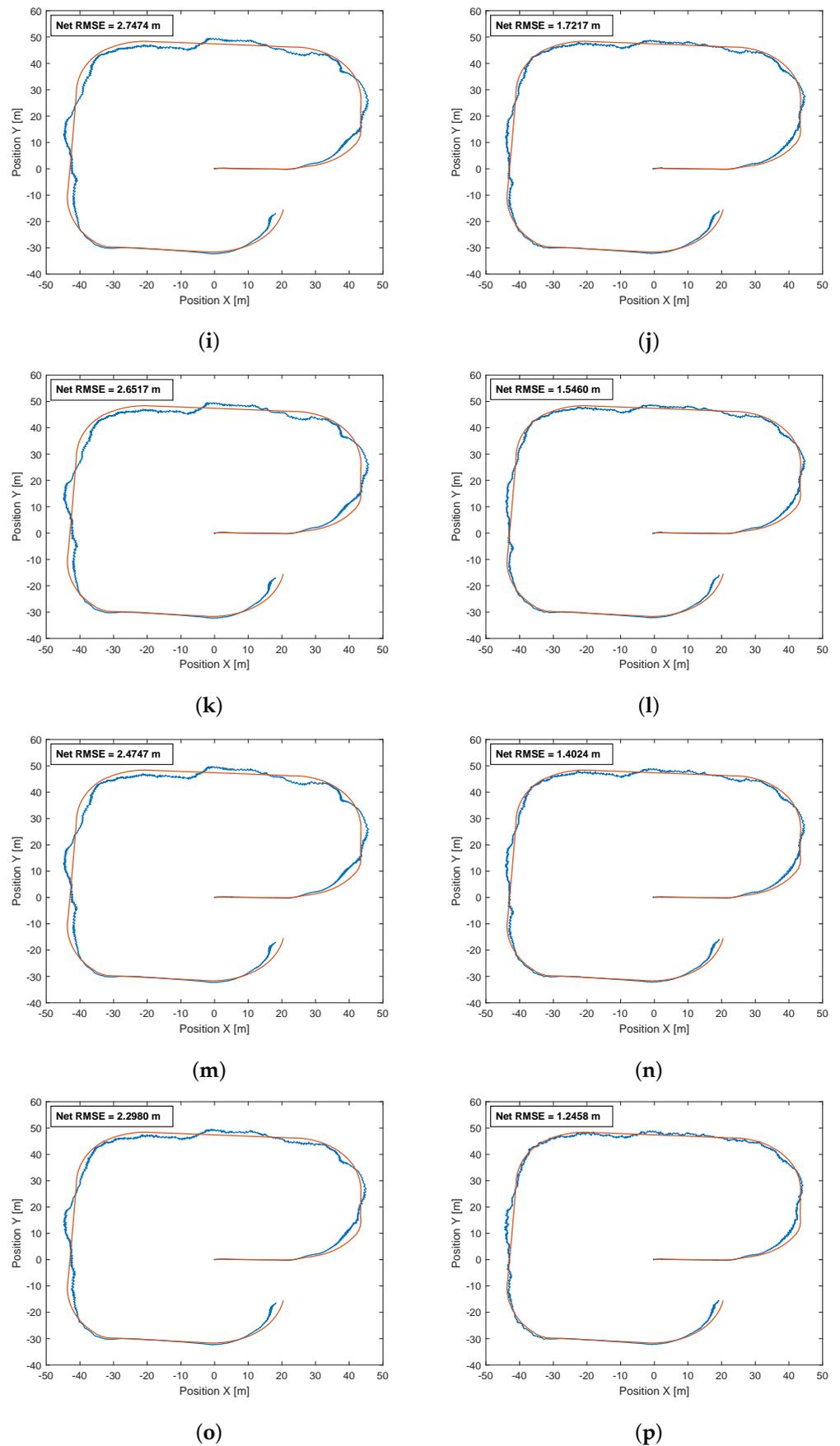
Figure 7. The IMU + Odometry estimation method yields for both EKF and UKF an RMSE over 70 m.

Except for the IMU + Odometry methods, the other results can be divided into four performance levels according to the error scale. The worst level is EKF with one GPS, where the RMSE is about 3~4 m. The second level is EKF with two GPS and EKF with three IMU and three GPS. In these cases, the RMSEs are about 2~3 m. A better level is UKF with 1 GPS and 1 IMU, where the RMSE of the third level can be achieved about 2~2.5 m. Finally, our experiment's best class is UKF with 2 GPS and 1 IMU, and UKF with 3 IMU and 3 GPS, which reduce the RMSE to about 1.2 m. (**a**) EKF with only one IMU. Here the red line is the ground truth and the blue line denotes the estimated position. This definition also applies to later subfigures; (**b**) UKF with only one IMU; (**c**) EKF with 1 IMU and 1 GPS; (**d**) UKF with 1 IMU and 1 GPS; (**e**) EKF with 2 IMU and 1 GPS; (**f**) UKF with 2 IMU and 1 GPS; (**g**) EKF with 3 IMU and 1 GPS; (**h**) UKF with 3 IMU and 1 GPS; (**i**) EKF with 1 IMU and 2 GPS; (**j**) UKF with 1 IMU and 2 GPS; (**k**) EKF with 2 IMU and 2 GPS; (**l**) UKF with 2 IMU and 2 GPS; (**m**) EKF with 3 IMU and 2 GPS; (**n**) UKF with 3 IMU and 2 GPS; (**o**) EKF with 3 IMU and 3 GPS; (**p**) UKF with 3 IMU and 3 GPS.

Figure 8 shows a comparison of localization error between the different approaches with respect to time. As shown in Figure 8, the RMSE and the Euclidean distance error of each sensor arrangement can be obtained, indicating that the EKF has a more significant tracking error than UKF. It happens because the linearization through its Jacobian is an approximation, and the kinematic model of the wheel loader is highly nonlinear. To get a more intuitive and accurate description of the error for each sensor configuration and method, we create Table 2 to demonstrate the RMSE in detail.
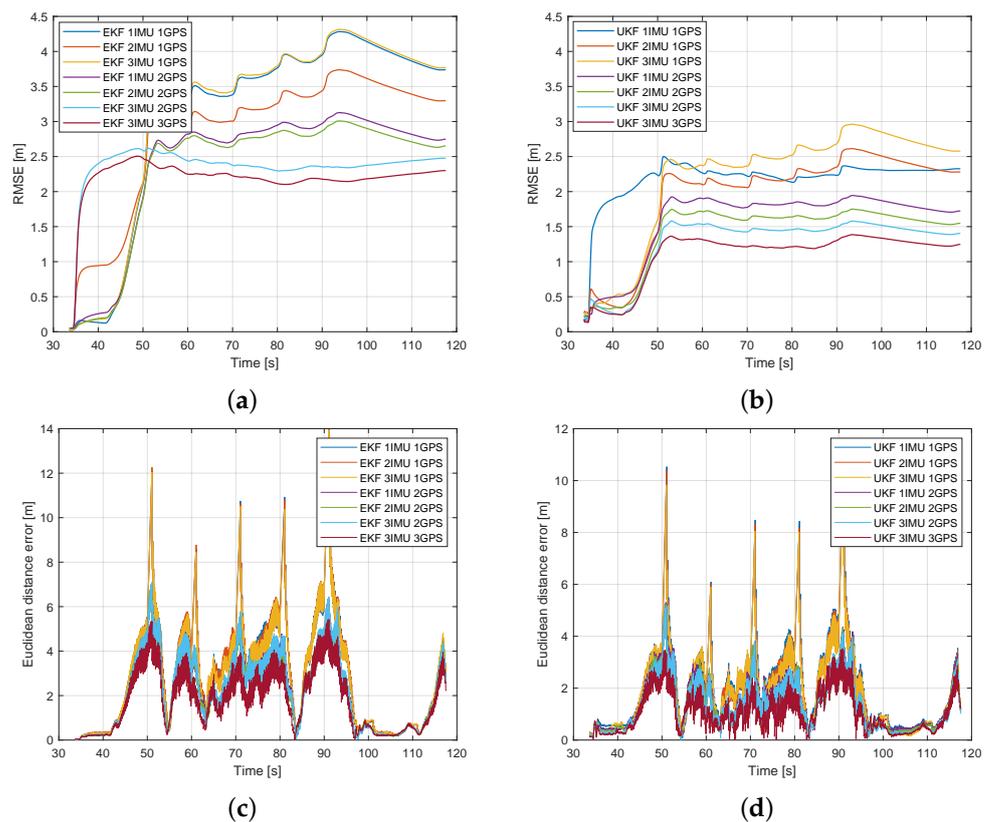

Figure 8. Quantitative evaluation of different methods. Here (**a**) RMSE of EKF and (**b**) RMSE of UKF show the accumulative error, i.e., RMSE, while (**c**) Euclidean distance error of EKF and (**d**) Euclidean distance error of UKF demonstrate the current Euclidean distance error of each sensor arrangement. There are some noticeable instantaneous position changes every ten seconds due to infrequent GPS signal loss. Generally, UKF shows better performance than EKF for both RMSE and Euclidean distance error.

As aforementioned, the GPS signal will be lost for one second every ten seconds. Thus, there are some noticeable instantaneous position changes every ten seconds. The loss of GPS signals clearly causes these jumps. As we can see, for sensor configuration with only one GPS, the leap of the localization error as well as the variance values are more sharp. Obviously, as the number of GPS increases, the jumps are diminished and therefore become more acceptable.

Table 2. Localization errors of each approach.

Group	RMSE (m) (x,y)	Net RMSE (m)
1 (EKF 1 IMU)	(69.122, 30.039)	75.3671
2 (EKF 1 IMU 1 GPS)	(2.472, 2.802)	3.7363
3 (EKF 2 IMU 1 GPS)	(2.172, 2.480)	3.2963
4 (EKF 3 IMU 1 GPS)	(2.474, 2.840)	3.7660
5 (EKF 1 IMU 2 GPS)	(1.830, 2.050)	2.7474
6 (EKF 2 IMU 2 GPS)	(1.778, 1.968)	2.6517
7 (EKF 3 IMU 2 GPS)	(1.894, 1.593)	2.4747
8 (EKF 3 IMU 3 GPS)	(1.794, 1.437)	2.2980
9 (UKF 1 IMU)	(56.586, 32.944)	65.4774
10 (UKF 1 IMU 1 GPS)	(1.654, 1.632)	2.3234
11 (UKF 2 IMU 1 GPS)	(1.374, 1.817)	2.2781
12 (UKF 3 IMU 1 GPS)	(1.611, 2.001)	2.5742
13 (UKF 1 IMU 2 GPS)	(1.093, 1.330)	1.7217
14 (UKF 2 IMU 2 GPS)	(0.975, 1.200)	1.5460
15 (UKF 3 IMU 2 GPS)	(0.873, 1.098)	1.4024
16 (UKF 3 IMU 3 GPS)	(0.826, 0.933)	1.2458

Apparently, the simulation results show that UKF is a better approach using data collected by onboard sensors of the wheel loader in gazebo environments. Intuitively, more sensors represent higher accuracy. However, the results show that an appropriate number of sensors can achieve acceptable accuracy at a lower cost. As we can see in Table 2, the RMSE of UKF with 1 IMU 2 GPS is 1.72 m, and UKF with 3 IMU and 2 GPS is 1.40 m. With additional 2 IMU and 1 GPS, the RMSE is only slightly reduced by 0.3 m. More importantly, the maximal error is strongly diminished by one additional GPS, whereas continually increasing the sensor number does not further reduce the error proportionally. Of course, according to different application scenarios, different sensor configurations shall be chosen. For our application scenario, UKF with one IMU and two GPS has sufficient accuracy and a better economy respecting sensor hardware cost and onboard ECU computational effort. Thus, we suggest using this sensor configuration to locate the mobile machines and then develop the realtime map plotter.

6.2. Plotter Results

As our ultimate goal is to create a map of the current working site in realtime based on localization technology so that corresponding optimization can then be achieved, the ground truth maps and estimated maps with different sensor configurations coupled with various algorithms are shown and compared in Figure 9. Since we use a two-layer grid map, both rolling friction coefficient and road grade were recorded and used to create the estimated maps. In our study, the ground information was saved in a corresponding grid after the wheel loader passed by and the ground information, such as friction and slope, was identified through the specific algorithm, e.g., recursive least square. As aforementioned, to locate the mobile machine's position in a cost-efficient fashion, we adopt the configuration of the UKF with one IMU and two GPS. In order to compare the performance of the selected configuration, we draw the results of the EKF with one IMU and one GPS and the UKF with one IMU and one GPS as the control groups. The mispredicted points are calculated as described in (19):

$$E_{r,s} = \sum_{i=1,j=1}^{m,n} e_{i,j} = \begin{cases} 1, & \text{if } \neg(\hat{G}_{i,j} = G_{i,j}) \cap \neg(\hat{G}_{i,j} = NaN) \\ 0, & \text{if } (\hat{G}_{i,j} = G_{i,j}) \cap \neg(\hat{G}_{i,j} = NaN) \\ 0, & \text{if } \hat{G}_{i,j} = NaN, \end{cases} \tag{19}$$

where $\hat{G}_{i,j}$ denotes the estimated grid map information, $G_{i,j}$ is the ground truth map information, E is the accumulated number of errors, the subscript r and s denotes resistance and slope map, respectively. Obviously, the goal is to minimize the percent of mispredicted points versus total predicted points, described as (20):

$$\min J_{r,s} = \frac{E_{r,s}}{T}, \tag{20}$$

where T is the total estimated number, and J is the quantitative criteria to evaluate the accuracy of the plotted map.

The localization errors of each group are shown in Table 3, where we can see the necessity of the introduction of the second GPS sensors, and the adoption of unscented Kalman filter.

Table 3. Localization errors of each approach during the plotting.

Group	RMSE (m) (x,y)	Net RMSE (m)
1 (EKF 1 IMU 1 GPS)	(2.584, 3.407)	4.4444
2 (UKF 1 IMU 1 GPS)	(2.003, 2.314)	3.0603
3 (UKF 1 IMU 2 GPS)	(1.336, 1.533)	2.0334

To evaluate the plotted maps' accuracy, we used Matlab to implement an algorithm to compare the predefined area and the plotted path the mobile machine traveled. Figure 10 graphically illustrates the mispredicted grid point with white color, where the mispredicted points are generally distributed in the marginal zone. This is because even the localization technology makes some mistakes; the problem is unlikely to cause an error as long as the vehicle is not at the very edge of different zones, indicating the robustness of this mapping idea of the construction site.
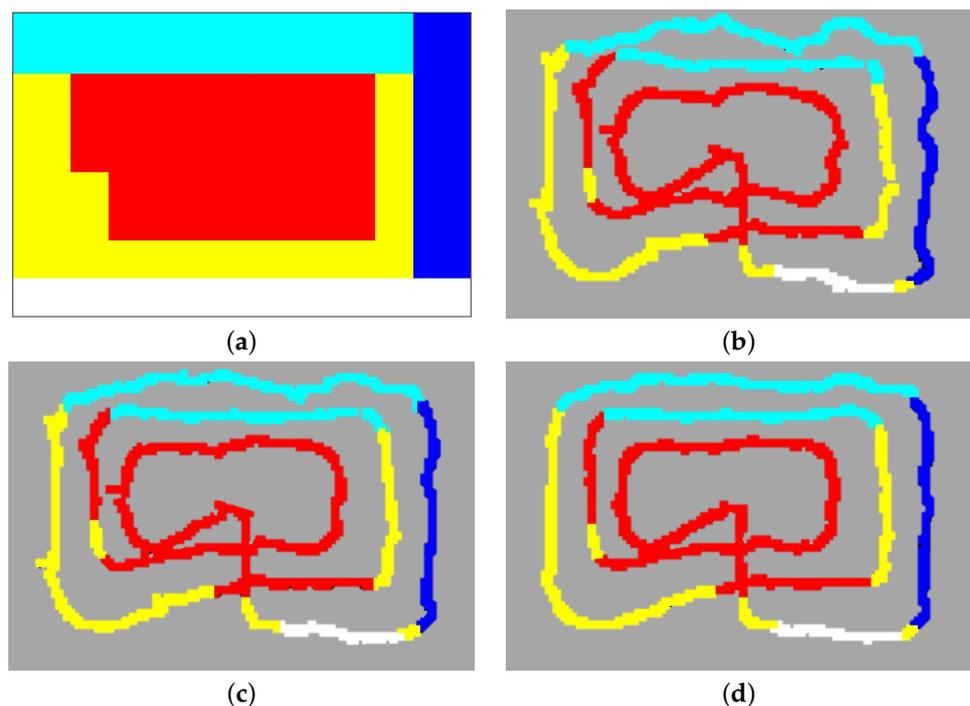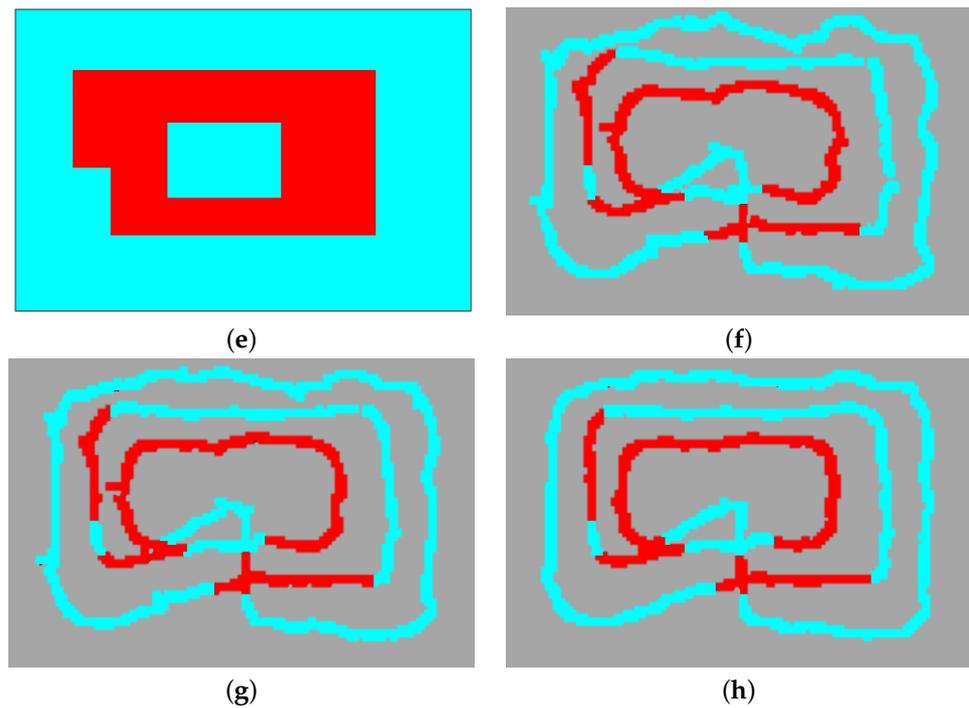

(a)

(b)

(c)

(d)

Figure 9. *Cont.*

Figure 9. The ground truth and estimated maps. In the ground resistance map and road grade map plotted by EKF with 1 IMU and 1 GPS, the spikes caused by infrequent GPS are quite obvious. With an additional GPS sensor fused in Kalman filter, the spikes improve a lot. (**a**) Predefined five areas of resistance plotted by OpenCV; (**b**) Result of ground resistance map with wheel loader (EKF with 1 IMU and 1 GPS); (**c**) Result of ground resistance map with wheel loader (UKF with 1 IMU and 1 GPS); (**d**) Result of ground resistance map with wheel loader (UKF with 1 IMU and 2 GPS); (**e**) Predefined two areas of road grade plotted by OpenCV; (**f**) Result of road grade map with wheel loader (EKF with 1 IMU and 1 GPS); (**g**) Result of road grade map with wheel loader (UKF with 1 IMU and 1 GPS); (**h**) Result of road grade map with wheel loader (UKF with 1 IMU and 2 GPS).
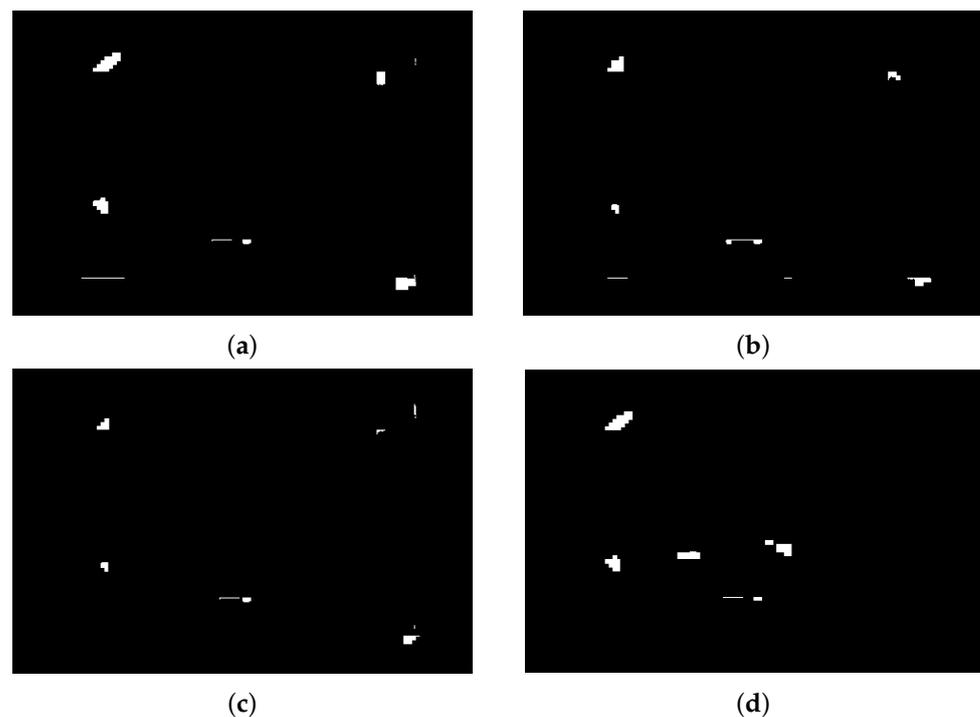

Figure 10. *Cont.*

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

Figure 10. Difference between ground truth and the estimated map. Here we compare the predefined areas and the plotted path, where the white pixels are the wrong plotted grids. (**a**) Result of EKF with 1 IMU and 1 GPS, (**b**) Result of UKF with 1 IMU and 1 GPS, (**c**) Result of UKF with 1 IMU and 2 GPS are the Friction map results, and (**d**) Result of EKF with 1 IMU and 1 GPS, (**e**) Result of UKF with 1 IMU and 1 GPS, (**f**) Result of UKF with 1 IMU and 2 GPS are the road grade map results. UKF shows more accurate positioning capabilities than EKF, and with two GPS fused in the Kalman filter, the wrong located grid is less than with just one GPS signal.

After calculating the wrong located grids and all the plotted grids and according to Equations (19) and (20):

1. For EKF with one IMU and one GPS: an error rate $J_r = 2.389\%$ of the ground resistances map and an error rate $J_s = 2.258\%$ of the slope map can be obtained;
2. For UKF with one IMU and one GPS: an error rate $J_r = 1.482\%$ of the ground resistances map and an error rate $J_s = 1.682\%$ of the slope map can be obtained;
3. For UKF with one IMU and two GPS: an error rate $J_r = 0.997\%$ of the ground resistances map and an error rate $J_s = 1.223\%$ of the slope map can be obtained.

7. Conclusions

In this study, we proposed an approach to creating a multi-layer map of the construction site in realtime so that the environmental information can be taken into account to improve vehicle efficiency and safety further or contribute to the path planning of mobile machines in construction sites. Considering the common phenomenon in reality mentioned by other researchers, such as noisy sensors and infrequent signal loss, we set up our simulation environment in Gazebo with an ROS package based on a real construction site. According to our tests in Gazebo by implementing a series of sensor configurations, we found that the configuration of one IMU and two GPS with an encoder using UKF has the best for overall performance with respect to accuracy and cost. By comparing the estimated maps drawn by the map plotter and the predefined maps, the errors are only 1.0% and 1.2% for road resistant force and grade, separately. Thus, we believe that the developed plotter can be used to save the road condition in realtime within a reasonable error range and encourage the engineers of construction machines to develop novel algorithms to further improve the holistic performance of machines based on our approach.

Outlook

In our research, we present a method to create a map with only one mobile machine. However, in a real working site, many mobile machines work simultaneously on the construction site, indicating the possibility of creating a map even faster if the machines can share the information. Thus, we encourage researchers to enable the cooperative map drawing approach by means of WIFI or 5G.

References

1. Osinenko, P.V.; Geissler, M.; Herlitzius, T. A method of optimal traction control for farm tractors with feedback of drive torque. *Biosyst. Eng.* **2015**, *129*, 20–33. [CrossRef]
2. Sprengel, M.W. Influence of Architecture Design on the Performance and Fuel Efficiency of Hydraulic Hybrid Transmissions. Dissertations, Purdue University, West Lafayette, IN, USA, 2015.
3. Vukovic, M.; Sgro, S.; Murrenhoff, H. STEAM—A Mobile Hydraulic System with Engine Integration. In Proceedings of the ASME/BATH 2013 Symposium on Fluid Power and Motion Control, Sarasota, FL, USA, 6–9 October 2013; p. V001T01A005.
4. Vukovic, M.; Leifeld, R.; Murrenhoff, H. STEAM—A Hydraulic Hybrid Architecture for Excavators. In Proceedings of the 10th International Fluid Power Conference, Dresden, Germany, 8–10 March 2016; pp. 151–162.
5. Xiang, Y.; Li, R.; B.C.; Liu, X.; Geimer, M. A novel Algorithm for Hydrostatic-mechanical Mobile Machines with a Dual-Clutch Transmission. *Energies* **2022**, *15*, 2095. [CrossRef]
6. Ge, L.; Quan, L.; Zhang, X.; Zhao, B.; Yang, J. Efficiency improvement and evaluation of electric hydraulic excavator with speed and displacement variable pump. *Energy Convers. Manag.* **2017**, *150*, 62–71. [CrossRef]
7. Ge, L.; Quan, L.; Zhang, X.; Huang, J.; Zhao, B. High Energy Efficiency Driving of the Hydraulic Excavator Boom with an Asymmetric Pump. In Proceedings of the 11th International Fluid Power Conference, Aachen, Germany, 19–21 March 2018.
8. Joos, S.; Trachte, A.; Bitzer, M.; Graichen, K. Constrained real-time control of hydromechanical powertrains-methodology and practical application. *Mechatronics* **2020**, *71*, 102397. [CrossRef]
9. Xiang, Y.; Geimer, M. Optimization of Operation Startegy for Primary Torque based hydrostatic Drivetrain using Artificial Intelligence. In Proceedings of the 12th International Fluid Power Conference, Dresden, Germany, 9–11 March 2020.
10. Mutschler, S.; Brix, N.; Xiang, Y. Torque Control for Mobile Machines. In Proceedings of the 11th International Fluid Power Conference, Aachen, Germany, 19–21 March 2018; pp. 186–195.
11. Xiang, Y.; Cui, Q.; Su, T.; Brach, C.; Geimer, M. Mass and Grade Estimation of mobile machines. *arXiv* **2020**, arXiv:2011.01830v2.
12. Vahidi, A.; Stefanopoulou, A.; Peng, H. Recursive least squares with forgetting for online estimation of vehicle mass and road grade: Theory and experiments. *Veh. Syst. Dyn.* **2005**, *43*, 31–55. [CrossRef]
13. Xiang, Y.; Xu, B.; Su, T.; Brach, C.; Mao, S.; Geimer, M. 5G meets Construction Machines: Towards a Smart working Site. In Proceedings of the 2021 International Conference on Computing and Communications Applications and Technologies (I3CAT), Ipswich, UK, 15–16 September 2021; pp. 28–35.
14. Xiang, Y. AI and IoT Meet Mobile Machines: Towards a Smart Working Site. Ph.D. Thesis, Institute of Vehicle System Technology, Karlsruhe, Germany, 2021.
15. Tan, H.; Huang, J. DGPS-Based Vehicle-to-Vehicle Cooperative Collision Warning: Engineering Feasibility Viewpoints. *IEEE Trans. Intell. Transp. Syst.* **2006**, *7*, 415–428. [CrossRef]
16. Ross, R.; Hoque, R. Augmenting GPS with geolocated fiducials to improve accuracy for mobile robot applications. *Appl. Sci.* **2019**, *10*, 146. [CrossRef]
17. Kalman, R.E. A new approach to linear filtering and prediction problems. *Trans. ASME J. Basic Eng.* **1960**, *82*, 35–45. [CrossRef]
18. Jung, S.; Kim, J.; Kim, S. Simultaneous localization and mapping of a wheel-based autonomous vehicle with ultrasonic sensors. *Artif. Life Robot.* **2009**, *14*, 186–190. [CrossRef]
19. Wang, S.; Deng, Z.; Yin, G. An Accurate GPS-IMU/DR Data Fusion Method for Driverless Car Based on a Set of Predictive Models and Grid Constraints. *Sensors* **2016**, *16*, 280. [CrossRef] [PubMed]
20. Perea, D.; Hernandez-Aceituno, J.; Morell, A.; Toledo, J.; Hamilton, A.; Acosta, L. MCL with sensor fusion based on a weighting mechanism versus a particle generation approach. In Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, The Netherlands, 6–9 October 2013; pp. 166–171. [CrossRef]
21. Sukkarieh, S.; Nebot, E.M.; Durrant-Whyte, H.F. A high integrity IMU/GPS navigation loop for autonomous land vehicle applications. *IEEE Trans. Robot. Autom.* **1999**, *15*, 572–578. [CrossRef]

22. Iwataki, S.; Fujii, H.; Moro, A.; Yamashita, A.; Asama, H.; Yoshinada, H. Visualization of the surrounding environment and operational part in a 3DCG model for the teleoperation of construction machines. In Proceedings of the 2015 IEEE/SICE International Symposium on System Integration (SII), Nagoya, Japan, 11–13 December 2015; pp. 81–87. [CrossRef]

23. Skog, I.; Handel, P. In-Car Positioning and Navigation Technologies—A Survey. *IEEE Trans. Intell. Transp. Syst.* **2009**, *10*, 4–21. [CrossRef]

24. Isa, H.L.; Saad, S.A.; Hisham, A.; Ishak, M. Improvement of GPS Accuracy in Positioning by Using DGPS Technique. In *AsiaSim 2017: Modeling, Design and Simulation of Systems*; Mohamed Ali, M., Wahid, H., Mohd Subha, N.A., Sahlan, S., Md. Yunus, M., Wahap, A.R., Eds.; Springer: Singapore, 2017; pp. 3–11.

25. Kim, J.; Song, J.; No, H.; Han, D.; Kim, D.; Park, B.; Kee, C. Accuracy Improvement of DGPS for Low-Cost Single-Frequency Receiver Using Modified Flächen Korrektur Parameter Correction. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 222. doi: 10.3390/ijgi6070222. [CrossRef]

26. Zhang, E.; Masoud, N. Increasing GPS Localization Accuracy With Reinforcement Learning. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 2615–2626. [CrossRef]

27. Soatti, G.; Nicoli, M.; Garcia, N.; Denis, B.; Raulefs, R.; Wymeersch, H. Implicit Cooperative Positioning in Vehicular Networks. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 3964–3980. [CrossRef]

28. Skoglund, M.; Petig, T.; Vedder, B.; Eriksson, H.; Schiller, E.M. Static and dynamic performance evaluation of low-cost RTK GPS receivers. In Proceedings of the 2016 IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, 19–22 June 2016; pp. 16–19. [CrossRef]

29. Toledo, J.; Piñeiro, J.D.; Arnay, R.; Acosta, D.; Acosta, L. Improving Odometric Accuracy for an Autonomous Electric Cart. *Sensors* **2018**, *18*, 200. [CrossRef]

30. Qi, X.; Wang, W.; Yuan, M.; Wang, Y.; Li, M.; Xue, L.; Sun, Y. Building semantic grid maps for domestic robot navigation. *Int. J. Adv. Robot. Syst.* **2020**, *17*, 172988141990006. [CrossRef]

31. Li, J.; Zhong, R.; Hu, Q.; Ai, M. Feature-Based Laser Scan Matching and Its Application for Indoor Mapping. *Sensors* **2016**, *16*, 1265. [CrossRef]

32. Li, J.; Hu, Q.; Ai, M.; Wang, S. A Geometric Estimation Technique Based on Adaptive M-Estimators: Algorithm and Applications. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 5613–5626. [CrossRef]

33. Julier, S.J.; Uhlmann, J.K. New extension of the Kalman filter to nonlinear systems. In Proceedings of the Signal Processing, Sensor Fusion, and Target Recognition VI, Orlando, FL, USA, 21–25 April 1997; p. 182. [CrossRef]

34. Wan, E.A.; van der Merwe, R. The unscented Kalman filter for nonlinear estimation. In Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373), Lake Louise, AB, Canada, 4 October 2000; pp. 153–158. [CrossRef]

35. Welch, G.; Bishop, G. *An Introduction to the Kalman Filter*; University of North Carolina: Chapel Hill, NC, USA, 1995.

36. Bento, L.C.; Nunes, U.; Moita, F.; Surrecio, A. Sensor fusion for precise autonomous vehicle navigation in outdoor semi-structured environments. In Proceedings of the IEEE Conference on Intelligent Transportation Systems, Vienna, Austria, 16 September 2005; pp. 245–250. [CrossRef]

37. Zhang, F.; Stahle, H.; Chen, G.; Chen, C.; Simon, C.; Buckl, C.; Knoll, A. A sensor fusion approach for localization with cumulative error elimination. In Proceedings of the IEEE Conference on Intelligent 2012 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Hamburg, Germany, 13–15 September 2012; pp. 1–6. [CrossRef]

38. Li, C.; Dai, B.; Wu, T. Vision-based precision vehicle localization in urban environments. In Proceedings of the 2013 Chinese Automation Congress, Changsha, China, 7–8 November 2013; pp. 599–604. [CrossRef]

39. Wolcott, R.W.; Eustice, R.M. Visual localization within LIDAR maps for automated urban driving. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 176–183. [CrossRef]

40. Ward, E.; Folkesson, J. Vehicle localization with low cost radar sensors. In Proceedings of the 2016 IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, 19–22 June 2016; pp. 864–870. [CrossRef]

41. Hata, A.Y.; Wolf, D.F. Feature Detection for Vehicle Localization in Urban Environments Using a Multilayer LIDAR. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 420–429. [CrossRef]

42. del Peral-Rosado, J.A.; Lopez-Salcedo, J.A.; Kim, S.; Seco-Granados, G. Feasibility study of 5G-based localization for assisted driving. In Proceedings of the 2016 International Conference on Localization and GNSS (ICL-GNSS), Barcelona, Spain, 28-30 June 2016; pp. 1–6. [CrossRef]

43. Rohani, M.; Gingras, D.; Vigneron, V.; Gruyer, D. A New Decentralized Bayesian Approach for Cooperative Vehicle Localization Based on Fusion of GPS and VANET Based Inter-Vehicle Distance Measurement. *IEEE Intell. Transp. Syst. Mag.* **2015**, *7*, 85–95. [CrossRef]

44. Kuutti, S.; Fallah, S.; Katsaros, K.; Dianati, M.; Mccullough, F.; Mouzakitis, A. A Survey of the State-of-the-Art Localization Techniques and Their Potentials for Autonomous Vehicle Applications. *IEEE Internet Things J.* **2018**, *5*, 829–846. [CrossRef]

45. Ting Goh, S.; Zekavat, S.A.; Abdelkhalik, O. An introduction to Kalman filtering implementation for localization and tracking applications. In *Handbook of Position Location: Theory, Practice, and Advances*, 2nd ed.; Wiley: Hoboken, NJ, USA, 2018; pp. 143–195.

46. Larsson, J.; Broxvall, M.; Saffiotti, A. An evaluation of local autonomy applied to teleoperated vehicles in underground mines. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–7 May 2010; pp. 1745–1752. [CrossRef]

47. Roberts, J.M.; Duff, E.S.; Corke, P.I. Reactive navigation and opportunistic localization for autonomous underground mining vehicles. *Inf. Sci.* **2002**, *145*, 127–146. [CrossRef]

48. Gu, R.; Marinescu, R.; Seceleanu, C.; Lundqvist, K. Formal verification of an autonomous wheel loader by model checking. In Proceedings of the 2018 ACM/IEEE Conference on Formal Methods in Software Engineering, Gothenburg, Sweden, 2 June 2018; pp. 74–83. [CrossRef]

49. Xiang, Y.; Tang, T.; Su, T.; Brach, C.; Liu, L.; Mao, S.; Geimer, M. Fast CRDNN: Towards on Site Training of Mobile Construction Machines. *IEEE Access* **2021**, *9*, 124253–124267. [CrossRef]

50. Bang, S.; Baek, F.; Park, S.; Kim, W.; Kim, H. An Image Augmentation Method for Detecting Construction Resources Using Convolutional Neural Network and UAV Images. In Proceedings of the 36th International Symposium on Automation and Robotics in Construction, International Association for Automation and Robotics in Construction (IAARC), Banff, AB, Canada, 21–24 May 2019. [CrossRef]

51. Shang, Z.; Shen, Z. Real-time 3D reconstruction on construction site using visual SLAM and UAV. *arXiv* **2019**, arXiv:1712.07122. https://doi.org/10.48550/arXiv.1712.07122.

52. Quan, Z.; Li, J.; Shuai, B.; Williams, H.; He, Y.; Li, Z.; Xu, H.; Yan, F. Multi-step reinforcement learning for model-free predictive energy management of an electrified off-highway vehicle. *Appl. Energy* **2019**, *255*, 113755. [CrossRef]

53. Quan, Z.; Zhang, Y.; Li, Z.; Li, J.; Xu, H.; Olatunbosun, O. Cyber-physical energy-saving control for hybrid aircraft-towing tractor based on online swarm intelligent programming. *IEEE Trans. Ind. Inform.* **2017**, *14*, 4149–4158.

54. Xiang, Y.; Mutschler, S.; Brix, N.; Brach, C.; Geimer, M. Optimization of hydrostatic-mechanical transmission control strategy by means of torque control. In Proceedings of the 12th International Fluid Power Conference, Dresden, Germany, 9–11 March 2020.

55. Xiang, Y.; Wang, H.; Su, T.; Li, R.; Brach, C.; Mao, S.; Geimer, M. KIT MOMA: A Mobile Machines Dataset. *arXiv* **2020**, arXiv:2007.04198. [CrossRef]

56. Samie, F.; Bauer, L.; Henkel, J. IoT technologies for embedded computing: A Survey. In Proceedings of the 2016 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ ISSS), Pittsburgh, PA, USA, 1–7 October 2016; pp. 1–10. [CrossRef]

57. Xiang, Y.; Su, T.; Brach, C.; Liu, X.; Geimer, M. Realtime Estimation of IEEE 802.11p for Mobile working Machines Communication respecting Delay and Packet Loss. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020.

58. Moore, T.; Stouch, D. A Generalized Extended Kalman Filter Implementation for the Robot Operating System. In *Intelligent Autonomous Systems 13*; Springer: Cham, Switzerland, 2016; pp. 335–348.

59. De Maesschalck, R.; Jouan-Rimbaud, D.; Massart, D.L. The mahalanobis distance. *Chemom. Intell. Lab. Syst.* **2000**, *50*, 1–18. [CrossRef]

60. Li, J. A practical O (N2) outlier removal method for point cloud registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2021**. [CrossRef] [PubMed]

61. Fankhauser, P.; Hutter, M. A Universal Grid Map Library: Implementation and Use Case for Rough Terrain Navigation. In *Robot Operating System (ROS)*; Springer: Cham, Switzerland, 2016; Volume 625, pp. 99–120. [CrossRef]