Extraction of Preview Elevation Information Based on Terrain Mapping and Trajectory Prediction in Real-Time

LILI WANG1, DINGXUAN ZHAO1,2, TAO NI1, AND SHUANG LIU2

1School of Mechanical and Aerospace Engineering, Jilin University, Changchun 130025, China
2Yanshan University, Qinhuangdao 066004, China

Corresponding author: Dingxuan Zhao (dingxuanz@163.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFC0802900, and in part by the National Natural Science Foundation of China under Grant 5180520.

ABSTRACT Preview elevation values are used as inputs to suspensions to improve a vehicle’s ride. In a previous study, we obtained preview terrain elevation information in post-processing. In this paper, we contributed a method to reconstruct the terrain elevation map in real-time to extract preview elevation information according to the predicted trajectory. We use interdisciplinary knowledge to solve an engineering problem in the field of mechanical engineering, and the innovation of this manuscript belongs to Engineering Application Innovation. First, in the process of data collection, the terrain elevation map is updated with LIDAR measurements. A Kalman filter is used to update the elevation map. Second, as the vehicle moves, the uncertainty in the vehicle’s motion is estimated, and the error variance that propagates to the terrain elevation map is computed. Third, the vehicle trajectory is predicted based on a vehicle kinematics model. According to the predicted trajectory, the elevation information is extracted from the terrain elevation map in advance. The proposed method has been extensively evaluated with the simulations in Gazebo and outdoor environments, and the results demonstrate that the algorithm is effective, with a maximum error of 2.58 cm for the estimated elevation value.

INDEX TERMS Mechanical engineering, laser measurement, measurement uncertainty, robot sensing systems, digital elevation models.

I. INTRODUCTION

While on route to rescue sites in remote areas with rough terrain, the chassis performance of emergency rescue vehicles may decrease due to bumpy roads, potentially damaging the rescue equipment to a certain extent. The ride comfort of vehicles is particularly important for emergency rescue vehicles in mountainous areas. To improve the vehicle’s ride quality, many researchers have made efforts to enact preview control over the vehicle’s suspension [1]–[5]. If preview elevation information is obtained in advance, it can be used as an input for modifying the suspension control, for better ride quality as the vehicle encounters road obstacles.

Preview control has two ways to acquire preview information. One is a wheelbase preview that provides the preview input to rear wheels according to the front wheel disturbance. However, this method does not include a preview input for the front wheels. The second way uses a look-ahead preview, in which three-dimensional (3D) sensors are installed on top of the vehicle to acquire road information in advance. Although some researchers have proposed look-ahead preview elevation method, but they didn’t realize to extract the elevation of changing the terrain in real-time or they only assumed that the elevation profile was already known, and they didn’t describe the research method in detail [6]–[10]. The look-ahead preview elevation method consists of two aspects: terrain elevation map reconstruction in real-time and elevation extraction according to the vehicle’s predicted trajectory. To date, efforts continue in the development of an accurate preview terrain model. This issue is of particular importance in unknown and complex environments, as it greatly affects the precision of the preview elevation information and indirectly influences the vehicle’s ride comfort.

In recent years, there has been a greater focus on terrain mapping similar to that used in planetary exploration, ground robots, unmanned driving, and so on [11]–[17]. Kevin et al.
used a wide-baseline stereo vision system to generate dense terrain maps in real-time. His algorithm achieved a precision of 56cm at an altitude of 40 m [18]. Cappalunga et al. [19] presented an innovative method to reconstruct a 3D elevation map of the terrain in real-time, using a biologically inspired optimization algorithm to segment terrain points as inliers and outliers. Triebel et al. [20] proposed an approach to represent terrain as multi-level surface maps, in which multiple surfaces were stored in each cell of the grid; their experimental results demonstrated the suitability of this method for large-scale outdoor terrain. Cole and Newman [21] used a segmentation algorithm to classify range data into different point cloud according to the vehicle pose, which was based on supervised learning; a delayed extended Kalman filter algorithm was applied to estimate the six degrees of freedom of the vehicle’s pose. Notably, the registration algorithm was particularly effective with regard to wide convergence basins, computation speed, and robustness. Carrio et al. [22] developed a novel adaptive simultaneous localization and mapping scheme to estimate the terrain for rover navigation; an odometry error model using a Gaussian process was used to predict nonsystematic errors to improve the terrain estimation accuracy. Ryde and Hu [23] used voxel lists for efficient neighbor queries to facilitate the construction of 3D maps; the resulting data structure based on occupied voxels efficiently represented 3D scan and map information. Belter et al. [24] developed a method using local grid maps to represent 3D mapping with different resolutions. Herbert et al. [25] introduced elevation maps based on two-dimensional (2D) grid maps, with added height for each cell for a 3D map representation. Pfaff et al. [26] developed an elevation mapping method in which the cumulative error is reduced by loop closures. Frankhauser et al. [27] incorporated the measurement uncertainty of a 3D sensor and motion uncertainty in their elevation mapping method. Droeschel developed an approach involving simultaneous localization and mapping in rough terrain based on a 3D sensor; the resolution of the map in the local vicinity has a high resolution but the resolution decreases with distance. A 6D pose of a robot was obtained by registering local maps to global maps; this was combined with graph optimization to yield a high-precision map [28]. Kleiner and Dornhege [29] utilized ratio-frequency identification technology to associate data; slippage-sensitive odometry was applied for 2D pose tracking, and an efficient method for modeling elevation terrain was introduced that applies an extended Kalman filter for 3D pose tracking. To obtain good light detection and ranging (LIDAR) images of the terrain, Labowski et al. [30] used an inertial navigation system/ global positioning system (INS/GPS) and a modified Kalman filter to align the data; the INS/GPS system reduced the geometric distortions in the mapping. Escobar Villanueva et al. [31] proposed LIDAR-derived control point method to investigate the contribution of LIDAR elevation data in digital elevation model (DEM) generation, based on fixed-wing unmanned aerial vehicle imaging for flood applications; this method achieved an accuracy comparable to other studies. Cremeau and Murray [32] introduced a technique to accommodate the uncertainty in sensor range data for DEM in real-time to create a single 2.5D digital elevation map, which was suitable for high speed and outdoor environments. Fankhauser et al. [33] proposed an elevation mapping approach, based on the drift in the robot pose estimation and distance sensor measurement uncertainty; a map fusion method further improved computation speed.

Although the aforementioned methods have achieved good effects in some applications, an accurate, real-time method for preview terrain elevation extraction is necessary to plan and safely maneuver through uncertain outdoor environments. In this paper, we contribute an approach to reconstruct the terrain elevation map in real-time to extract the elevation values in advance based on the trajectory prediction. The significance of this project is that we use interdisciplinary knowledge to solve an engineering problem in the field of mechanical engineering, and the innovation of this manuscript belongs to Engineering Application Innovation. The remainder of this paper is organized as follows. Section II discusses the system structure, the representation of mathematical symbols, and the definition of coordinate systems. Section III presents the terrain elevation map update with LIDAR measurements, including the uncertainty estimation of the LIDAR model in the process of measurement and the terrain elevation map update with a Kalman filter. Section IV describes the uncertainty in the estimated motion of the vehicle and the error variance that propagates to the terrain elevation map. The vehicle trajectory is predicted based on the kinematics model in Section V. Section VI presents the simulation and experimental results. Concluding remarks are given in Section VII.

II. SYSTEM STRUCTURE AND RELATED DEFINITIONS

A. SYSTEM STRUCTURE

If road elevation information could be acquired in advance, the information could be used as inputs to the vehicle to improve the ride comfort, which is particularly important for emergency rescue vehicles in mountainous areas. In this paper, we propose an approach to reconstruct the terrain map in real-time to extract the preview elevation information according to the trajectory prediction. First, in the process
of data collection, the terrain elevation map is updated with LIDAR measurements; a Kalman filter is used to estimate the height of the elevation map. Second, as the vehicle moves, the vehicle’s motion uncertainty is estimated and the error variance that propagates to the terrain elevation map is computed. Third, the vehicle trajectory is predicted based on the vehicle kinematics model. According to the predicted trajectory, we can extract the elevation information from the terrain elevation map in advance.

In addition, LIDAR and inertial measurement unit (IMU) can be used to obtain high-accuracy point cloud and pose data to ensure the precision of the terrain elevation values. This is the first application of its kind in which the uncertainty of the LIDAR model and vehicle motion are taken into consideration. Some positioning and mapping schemes are based on probabilistic robotics theory. Here, we apply cross-cutting of areas, a well-known method, to improve elevation estimates for better preview elevation projections.

B. REPRESENTATION OF MATHEMATICAL SYMBOL

In this paper, many symbols and subscripts are used to define the various coordinate systems, as explained in the following. The left subscript indicates the coordinate system, and the right subscript indicates the start and end points. $L_{LP}$ denotes vector from point $L$ to point $P$ in the Euclidean space $E^3$, and the subscript $L$ to the left of the vector symbol represents the coordinate system $L$. Lie groups and Lie algebra are used to represent the 3D orientations in this paper. The term rotation $\Phi_{VL} \in SO(3)$ is used for the relative orientations from coordinate system $L$ to coordinate system $V$. The equation $V_{LP} = \Phi_{VL}(L_{LP})$ represents a vector with rotation transformation from coordinate system $L$ to coordinate system $V$. The operator $\circ$ represents multiple rotations, e.g., $\Phi_{CA} = \Phi_{CB} \circ \Phi_{BA}$. In addition, the mapping $SO(3) \rightarrow R^{3\times3}$ such that $\Phi(\gamma) = C(\Phi)\gamma$ indicates the corresponding rotation matrix [34].

C. COORDINATE SYSTEM DEFINITIONS

In this study, four coordinate systems are introduced as shown in Fig. 2, the inertial coordinate system $I$, the vehicle coordinate system $V$, the terrain coordinate system $T$, and the LIDAR coordinate system $L$. The inertial coordinate system is stationary with respect to the environment. The real terrain is fixed to the inertial coordinate system. The terrain coordinate system is defined relative to the vehicle coordinate system, and the vehicle coordinate system is defined at the center of the vehicle body. In fact, we are only concerned with local terrain map. The LIDAR coordinate system is fixed at the center of the LIDAR. The relative position relationship between the LIDAR coordinate system and the vehicle coordinate system is invariant. The yaw angles of the LIDAR coordinate system and the terrain coordinate system are the same. In other words, we only care about the terrain map of vehicle direction in order to extract the elevation information based on the predicted trajectory. It is also what we particularly set. Multiple calibrations are required to transform coordinate systems. As such, a transformation matrix is used to represent the relationship between the two-coordinate systems, which is defined as $(r_{VL}, \Phi_{VL})$. $r_{VL}$ represents a 3D translation and $\Phi_{VL}$ represents a 3D rotation. Similarly, the transformation between the inertial coordinate system and vehicle coordinate system is defined by translation $r_{IV}$ and rotation $\Phi_{IV}$ as $(r_{IV}, \Phi_{IV})$. The transformation can also be obtained through the pose estimation of the vehicle while in motion. A covariance matrix is used to represent the uncertainty in the vehicle’s pose at different times, as follows:

$$\Sigma_{IV}^{6\times6} = \text{cov}(r_{IV}, \Phi_{IV})$$  \hspace{1cm} (1)

In general, we can describe 3D rotation by using yaw, pitch, and roll, as given in (2):

$$\Phi_{IV} = \Phi_{IV}(\psi) \circ \Phi_{IV}(\theta, \varphi)$$  \hspace{1cm} (2)

where $\Phi_{IV}(\psi)$ represents the rotation between the vehicle coordinate system and the inertial coordinate system around the z-axis. In other words, the $V$ coordinate system only represents the yaw of the vehicle coordinate system $V$, and the remaining rotation with pitch and roll between the $V$ coordinate system and vehicle coordinate system $V$ are described by $\Phi_{IV}(\theta, \varphi)$. Finally, the transformation between the terrain coordinate system $T$ and vehicle coordinate system $V$ is defined as $(r_{VT}, \Phi_{VT})$ with translation $r_{VT}$ and rotation $\Phi_{VT}$.

In our study, to simplify subsequent derivations and calculations, we assumed that z-axis of the terrain coordinate system $T$ remains always aligned with the z-axis of the vehicle coordinate system $V$. In other words, the yaw is the only different degree of freedom between the two coordinate systems. The $V$ coordinate system represents the yaw of the $V$ vehicle coordinate system. There are two degrees of freedom between the two coordinate systems $V$ and $V$: pitch and roll. Therefore the goal of dimensionality reduction is realized.

III. TERRAIN ELEVATION MAP UPDATE WITH LIDAR MEASUREMENT

In this paper, we use $P_i = (x_i, y_i, \hat{h}_i)$ to indicate the terrain elevation map at cell $i$. $\hat{h}_i$ is used to indicate the height estimate
value in the terrain coordinate system at \( P_i \). To simplify the formula, a point \( P \) is chosen from the measurement point cloud for the analyses, as shown in Fig. 2. The Gaussian probability distribution as \( Z \sim \mathcal{N}(h_p, \sigma^2_{hp}) \) represents the estimated height of point \( P \). \( h_p \) represents the mean, and \( \sigma^2_{hp} \) represents the variance of the elevation map.

### A. THE UNCERTAINTY ESTIMATION OF LIDAR MODEL IN
### THE PROCESS OF MEASUREMENT

Based on Gaussian noise, we can derive the LIDAR error model in the process of rotation and translation, as Eqs. (3)-(9), which is important for the precision of the terrain elevation values.

\[ \{\rho, \alpha, \beta\} \text{ are defined as the range, azimuth and elevation angle in the LIDAR coordinate system; the corresponding variances are denoted by } \{\epsilon_{\rho}, \epsilon_{\alpha}, \epsilon_{\beta}\}. \]

The measurement value of point \( P \) in the LIDAR coordinate system is given by:

\[ M_L = \begin{bmatrix} \rho \cos \alpha \cos \beta \\ \rho \cos \alpha \sin \beta \\ \rho \sin \beta \end{bmatrix} \] (3)

The noise is assumed to be Gaussian noise, which is applied as follows:

\[ \rho \rightarrow \rho_0 + \epsilon_{\rho}, \quad \alpha \rightarrow \alpha_0 + \epsilon_{\alpha}, \quad \beta \rightarrow \beta_0 + \epsilon_{\beta} \] (4)

where 0 represents a nominal measurement and \( \epsilon \) indicates the first-order statistic. Applying the small angle approximation, the noisy measurements in the LIDAR coordinate system are described as (5), shown at the bottom of this page. The covariance of LIDAR in the measurement process is given by:

\[ \Sigma_L = E(\epsilon \cdot M_L \in M_L^T) \] (6)

We also assume that the noise in the rotation and translation processes between different coordinate systems is Gaussian, which is applied as follows:

\[ R_{VL} \rightarrow R_{VL0} + \epsilon_{R_{VL}}, \quad R_{IV} \rightarrow R_{IV0} + \epsilon_{R_{IV}}, \quad T_L \rightarrow T_{L0} + \epsilon_{T_L}, \quad T_V \rightarrow T_{V0} + \epsilon_{T_V} \] (7)

where \( R_{VL} \) indicates the rotation from the LIDAR to the vehicle coordinate system, and \( T \) represents the translation.

Higher order terms are not considered; thus, the measurement values in the vehicle coordinate system are given by:

\[ M_V = R_{VL}M_L + T_L = R_{VL0}M_{L0} + R_{VL0}M_L + T_{L0} + T_L \] (8)

The measurement values in the inertial coordinate system are as follows:

\[ M_I = R_{IV}M_V + T_V = R_{IV0}M_V + \epsilon_{R_{IV}}(R_{VL0}M_{L0} + T_{L0}) + T_{V0} + \epsilon_{T_V} \] (9)

Now, we can compute \( \Sigma_R \) and \( \Sigma_v \). To improve the computation speed, we use Lie groups and Lie algebra to indicate 3D orientations in this research. The computation process is as follows.

From Fig. 2, the measurement point \( P \) is represented by vector \( L_{LP} \) in the LIDAR coordinate system \( L \). With transformation from the LIDAR coordinate system \( L \) to the terrain coordinate system \( T \), we obtain the measurement elevation information as (10). The Equation (10) is the vector representation of terrain elevation information.

\[ h_p = Z[\Phi_L^{-1}(L_{LP}) - T_{LT}] \] (10)

In this equation, the projection matrix \( Z = [0 0 1] \) is used to map the 3D measurements to extract the height of point \( P \). The error source of the measurement value consists of three aspects: the rotation transformation between different coordinate systems, the LIDAR measurement error model, and the translation transformation between different coordinate systems. The error propagation for variance \( \sigma^2_{hp} \) is described by (11).

\[ \sigma^2_{hp} = J_L \Sigma_L J_L^T + J_{\Phi_R} \Sigma_{\Phi_R} J_{\Phi_R}^T + J_{\Phi_H} \Sigma_{\Phi_H} J_{\Phi_H}^T \] (11)

where \( \Sigma_L \) represents the covariance matrix of the LIDAR model, which is obtained from the LIDAR noise model; \( \Sigma_{IL} \) denotes the covariance matrix of LIDAR rotation; and \( J \) represents the Jacobian matrix.

According to the definition of the terrain coordinate system, the yaw values of the LIDAR and terrain coordinate systems are always the same. In addition, the impact of the yaw uncertainty of the LIDAR is not considered due to the use of the projection matrix \( Z \) and the terrain coordinate system definition. Thus, the translation transformation error is not taken into consideration in this step. The error transmission variance \( \sigma^2_{hp} \) can be rewritten as (12):

\[ \sigma^2_{hp} = J_L \Sigma_L J_L^T + J_{\Phi_H} \Sigma_{\Phi_H} J_{\Phi_H}^T \] (12)

With (10), we can compute the Jacobian matrix of LIDAR measurements and the LIDAR rotation Jacobian matrix, respectively, as (13) and (14):

\[ J_L = \begin{bmatrix} \rho_0 \cos \alpha_0 \cos \beta_0 \\ \rho_0 \cos \alpha_0 \sin \beta_0 \\ \rho_0 \sin \beta_0 \end{bmatrix}, \quad J_{\Phi_R} = \begin{bmatrix} \rho_0 \cos \alpha_0 \cos \beta_0 \\ \rho_0 \cos \alpha_0 \sin \beta_0 \\ \rho_0 \sin \beta_0 \end{bmatrix}, \quad J_{\Phi_H} = \begin{bmatrix} \rho_0 \cos \alpha \cos \beta \\ \rho_0 \cos \alpha \sin \beta \\ \rho_0 \sin \beta \end{bmatrix} \] (13)

\[ J_L = \partial h_p / \partial L_{LP}, \quad \Sigma_L = \begin{bmatrix} \epsilon_{\rho} \\ \epsilon_{\alpha} \\ \epsilon_{\beta} \end{bmatrix} \] (14)
\[ J_\Phi = \frac{\partial h_p}{\partial \Phi_{LT}} = \frac{\partial (Z[\Phi_{LT}^{-1}(I_{L_{LP}}) - T_{L_{LP}}])}{\partial \Phi_{LT}} \]

\[ = -ZC(\Phi_{LT})^{T} L_{LP}^{\gamma_p} \]  

Equations (12), (13) and (14) in section III.I are the basic method to solve Jacobian matrix and variance. According to these equations, we can get the covariance matrix to express the uncertainty of lidar measurement.

**B. TERRAIN ELEVATION MAP UPDATE**

According to terrain continuity, vehicle mobility and measurement uncertainty, two measurements with the same cell may occur in different locations. Each measurement corresponds to different elevation points. According to the value of the Mahalanobis distance, we can use this to judge large drops in the same cell, such as a wall. Thus, the Mahalanobis distance provides information on the distance between two Gaussian distributions. The Mahalanobis distance is defined as follows:

\[ Z_k \sim \begin{pmatrix} Z_k \mid \hat{h}_k \end{pmatrix} \]

\[ Z_{k+1} \sim \begin{pmatrix} Z_{k+1} \mid (k+1)h_p \end{pmatrix} \sigma_{h_p}^2 \]

\[ d(Z_k, Z_{k+1}) = \left((\hat{h}_k - (k+1)h_p)^2 \right)^{1/2} \]

\[ \hat{h}_k = \max\{\hat{h}_k, (k+1)h_p\} \]

The grounds for the claim that the mean of the terrain elevation value is given by (18), which is to choose the bigger value between estimated elevation value and measurement value to represent the terrain elevation value.

The variance of the terrain elevation value at time \(k+1\) is as follows:

\[ \sigma_h^2 = \begin{cases} \sigma_h^2, \quad \hat{h}_k > (k+1)h_p \\ \sigma_{h_p}^2, \quad \hat{h}_k < (k+1)h_p \end{cases} \]

If \(d(Z_k, Z_{k+1}) < d_m\), we can use Kalman filter to fuse the existing elevation terrain estimation values with the new height measurements. The state of the Kalman filter is the scalar height estimate for terrain cells. First, we introduce the Kalman filter algorithm; this algorithm uses measurement values to provide a more accurate estimation of the elevation compared to a single measurement [35]. The Kalman filter includes two steps: prediction and update, as shown in Fig.3 [36].

\[
\begin{align*}
\hat{x}_{k+1/k} &= A_k \hat{x}_k \\
\hat{P}_{k+1/k} &= A_k \hat{P}_k A_k^T + Q_{k+1}
\end{align*}
\]

In the prediction step, we compute the priori values including system condition value and the covariance at time \(k+1\), as follows:

\[
\begin{align*}
\dot{x}_{k+1/k} &= A_k \dot{x}_k \\
\dot{P}_{k+1/k} &= A_k \dot{P}_k A_k^T + Q_{k+1}
\end{align*}
\]

where \(A_k\) is the system transformation matrix at time \(k\). \(Q_{k+1}\) is process noise variance. \(\dot{x}_{k+1/k}\) and \(\dot{P}_{k+1/k}\) represent the system condition priori value and covariance priori value respectively. \(\dot{x}_k\) and \(\dot{P}_k\) are the system condition estimation and covariance estimation at time \(k\), respectively.

In the update step, we compute the posteriori estimation of the system condition value and covariance at time \(k+1\). In other words, the Kalman filter is used to correct the state and covariance estimation. With \(\dot{x}_{k+1/k}\) and \(\dot{P}_{k+1/k}\), we can compute \(\hat{x}_{k+1/k}\) and \(\hat{P}_{k+1/k}\):

\[
\begin{align*}
K_{k+1} &= \dot{P}_{k+1/k} H_{k+1}^T (H_{k+1} \dot{P}_{k+1/k} H_{k+1}^T + R_{k+1})^{-1} \\
\hat{x}_{k+1/k} &= \dot{x}_{k+1/k} + K_{k+1} (Z_{k+1} - H_{k+1} \dot{x}_{k+1/k}) \\
\hat{P}_{k+1/k} &= (I - K_{k+1} H_{k+1}) \dot{P}_{k+1/k}
\end{align*}
\]

where \(K_{k+1}\) represents the Kalman gain at time \(k+1\). \(R_{k+1}\) is the observation covariance. \(H_{k+1}\) is the transition matrix from state variables to measurements, and \(\hat{x}_{k+1/k}\) and \(\hat{P}_{k+1/k}\) are posteriori estimations of the system condition value and covariance at time \(k + 1\), respectively.

In this study, the system state is the height of the cell. Given that the height measurement estimation from the last moment to the present moment has no independent dynamics association, therefore the system state conversion equation is equal to identity. The height estimation is only updated from measurements for a certain point \((x, y)\). Therefore, Kalman filtering is suitable for our terrain elevation model. With this method, the height value of cells of the terrain map will be updated when a series of measurements are obtained. The \(H\) is \(I\). The error covariance \(P\) corresponds to the variance of the height estimation \(\sigma_h^2\). The observation covariance \(R\) corresponds to the height measurement covariance \(\sigma_{h_p}^2\). The observation value \(x\) is the height estimation \(\hat{h}\), and \(z\) is associated with the height measurement value \(h_p\). Using (22)-(24),
we obtain the following:

\[
K_{k+1} = \frac{(k) \sigma_h^2 + \sigma_{hp}^2}{(k+1) \sigma_h^2 + \sigma_{hp}^2}^{-1}
\]

(25)

\[
(k+1) \hat{h} = (k) \hat{h} + K_{k+1}((k+1)h_p - (k) \hat{h})
\]

(26)

\[
(k+1) \sigma_h^2 = (I - K_{k+1})(k) \sigma_h^2
\]

(27)

With (25) and (26), we obtain the following equation.

\[
(k+1) \hat{h} = (k) \hat{h} + K_{k+1}((k+1)h_p - (k) \hat{h})
\]

\[
= \frac{(k) \hat{h} + K_{k+1}((k+1)h_p - (k) \hat{h})}{(k) \sigma_h^2 + (k+1) \sigma_{hp}^2}
\]

(28)

With (25) and (27), we obtain the following equation.

\[
(k+1) \sigma_h^2 = (I - K_{k+1})(k) \sigma_h^2
\]

\[
= \frac{(k) \sigma_h^2 + (k+1) \sigma_{hp}^2}{(k) \sigma_h^2 + (k+1) \sigma_{hp}^2}
\]

(29)

The resultant equations (28) and (29) are the mean and variance of the terrain elevation calculated according to Kalman filter. In other words, we bring our own terrain mathematical model into the Kalman filtering algorithm to get the estimated terrain elevation values with measurement update.

Therefore, the fusion of the new measurement height value \((h_p, \sigma_{hp}^2)\) with the estimated height value \((\hat{h}, \sigma_h^2)\) through the Kalman filter is completed. The estimation values before an update are denoted by \(k\) on the left-upper superscript, and with \(k + 1\) of the left-upper superscript if the update has already been implemented. This method allows the cells to be updated at high speed.

IV. TERRAIN ELEVATION MAP UPDATE WITH VEHICLE MOTION

A. UNCERTAINTY ESTIMATION OF VEHICLE MOTION

When the vehicle moves, the motion of the vehicle will create noise errors that influence the precision of the height values of the terrain map. Because the elevation terrain coordinate system is defined relative to the vehicle motion pose, the terrain elevation map will be updated when the vehicle moves. To estimate the terrain in the moving terrain coordinate system \(T\), the mean \(\hat{h}\) and covariance \(\sigma_h^2\) should be updated according to the vehicle’s pose. In other words, the mean and covariance of each cell will be updated according to the uncertainty in the vehicle motion. In addition, the estimation for a particular cell is related to the estimations of the surrounding cells. However, this process assumes a huge computational burden. Therefore, in our approach, a spatial covariance matrix \(\Sigma_p \in R^{3x3}\) is used to extend the terrain elevation map structure. Through this method, we can obtain the 3D uncertainty of each point, and the computationally expensive fusion step can be postponed until the elevation information corresponding to the predicted trajectory from the vehicle kinematic model is extracted. Hence, when the grid cell obtains a measurement update, its covariance will be set to [37].

\[
\Sigma_p = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \sigma_p^2 \end{bmatrix}
\]

(30)

where the height estimation variance \(\sigma_p^2\) is computed in Section III.II. If the LIDAR measurement update is not received by the grid cell, the covariance is updated continuously, based on the relative change in the motion pose of the vehicle from the previous moment to the current moment. This step ensures the robustness of the terrain elevation map to motion noise.

Figure 4 shows the derivations of the terrain elevation map update based on vehicle motion from time \(k\) to time \(k + 1\). The map represented in the vehicle coordinate system is transformed to the terrain coordinate system \(T\) by \((v_\hat{T}_{k+1}, \Phi_{v_\hat{T}_{k+1}})\).

The derivation is as follows.

\[
r_{T_{k+1}p} = r_{v_\hat{T}_{k+1}k} + r_{T_{k+1}p}
\]

(31)

\[
r_{T_{k+1}k} = r_{v_\hat{T}_{k+1}k} - r_{v_\hat{T}_{k+1}k+1}
\]

(32)

\[
v_{\hat{T}_{k+1}} = v_{\hat{T}_{k+1}} + v_{\hat{T}_{k+1}k+1}
\]

(33)

\[
r_{T_{k+1}p} = -r_{v_\hat{T}_{k+1}k+1} + r_{v_\hat{T}_{k+1}k} + r_{T_{k+1}p}
\]

(34)

Through (34), we can express \(r_{T_{k+1}p}\) with respect to time \(k\) in the terrain coordinate system. It is transformed to time \(k + 1\) as (35):

\[
T_{k+1}r_{T_{k+1}p} = -T_{k+1}r_{v_\hat{T}_{k+1}k+1} - \Phi_{v_\hat{T}_{k+1}k+1}^{-1}(v_{\hat{T}_{k+1}}r_{v_\hat{T}_{k+1}k+1})
\]

\[
+ \Phi_{v_\hat{T}_{k+1}k+1}^{-1}(v_{\hat{T}_{k+1}}r_{v_\hat{T}_{k+1}k} + T_{k}r_{T_{k+1}p})
\]

(35)

Through Eq.(35), we see that the estimation results at time \(k + 1\) are related to the results at time \(k\). In other words, the errors of previous results will influence the precision of the new estimation results. In this paper, to avoid this impact, we assume that the terrain coordinate system is aligned at time \(k\) and time \(k + 1\), which includes translation and rotation.

\[
r_{T_{k+1}k} = 0
\]

(36)

\[
\Phi_{T_{k+1}k} = \Phi_I
\]

(37)
According to (36) and (38), we can obtain the (38) and (38).

\[ r_{Tk}T_{k+1} = -r_{Tk}r_{k+1} + r_{k+1}r_{Tk+1} = 0 \] (38)

\[ \Phi_{Tk}T_{k+1} = \Phi^{-1}_r \cdot \Phi_{Tk} \cdot \Phi_{k+1} \cdot \Phi_{Tk+1} = \Phi_f \] (39)

\[ r_{Tk}T_{k+1} \] is transformed to time \( k + 1 \) as:

\[ r_{Tk}T_{k+1} = -r_{Tk}r_{k+1} + r_{k+1}r_{Tk} \] (40)

\[ t_{k+1} \cdot r_{Tk}T_{k+1} = = \Phi^{-1}_{Tk} (r_{Tk}T_{k+1}) \] (41)

According to (38), we obtain the following for (42).

\[ \Phi_{k+1}T_{k+1} = (\Phi_{k+1}^{-1} \cdot \Phi_{k+1} T_{k+1})^{-1} = \Phi^{-1}_{k+1} \cdot \Phi_{Tk} \] (42)

Substitution of Eqs.(41) and (42) into (35) gives (43).

\[ t_{k+1} \cdot r_{Tk}T_{k+1} = -t_{k+1} \cdot r_{Tk}T_{k+1} - \Phi_{Tk+1}^{-1} (r_{k+1} r_{Tk+1}) \]

\[ + \Phi^{-1}_{Tk} (r_{Tk+1} r_{k+1}) + r_{Tk} T_{k+1} \]

\[ = \Phi^{-1}_{Tk} (r_{k+1} r_{Tk+1} + t_{k+1} r_{Tk+1}) \]

\[ = \Phi^{-1}_{Tk} (t_{k+1} r_{k+1} + t_{k+1} r_{Tk+1}) \] (43)

We see from (43) that the terrain coordinate systems at different times are aligned, which could reduce the number of calculations for faster computational speed. In this way, we do not have to move efficient data in terrain elevation map. From the above derivations, the uncertainty in vehicle motion from time \( k \) to \( k + 1 \) can be expressed as follows:

\[ r_{Tk} r_{k+1} \sim N(r_{Tk}, \sigma_r) \] (44)

\[ \Phi_{Tk+1} \sim N(\Phi_{Tk}, \sigma_{\Phi}) \] (45)

where \( r_{Tk} r_{k+1} \) and \( \Phi_{Tk+1} \) represent the translation and rotation from time \( k \) to time \( k + 1 \) in the vehicle coordinate system respectively.

According to (30) and (35), we can express the error propagation of covariance from \( k \) to \( k + 1 \) as follows:

\[ r_{Tk+1} \sim N(r_{Tk+1}, \sigma_r) \] (46)

\[ \Sigma_{p,k+1} = J_p \Sigma_p, J_p \Sigma_p - J_f \Sigma_f J_f^T \] (47)

where \( \Sigma_{p,k+1} \) indicates the variance of point \( P \) at time \( k + 1 \). \( \Sigma_r \) represents the variance of point \( P \) at time \( k \). \( \Sigma_{\Phi} \) and \( \Sigma_{\Phi} \) represent the variances of translation and rotation respectively. \( J_r \) and \( J_f \) represent the error Jacobian matrices of translation transformation and rotation transformation of vehicle motion from time \( k \) to time \( k + 1 \), respectively, and \( J_f \) represents the Jacobian matrix of point \( P \) at time \( k \).

The error propagation includes two aspects. One is the vehicle motion’s uncertainty. The other is the variance in the distance estimation of point \( P \) at different times in the terrain coordinate system. \( \Sigma_r \) and \( \Sigma_{\Phi} \) are computed in Section IV. The Jacobian matrices are obtained as (48)-(50), shown at the bottom of this page, according to (35).

**B. COVARIANCE COMPUTATION OF VEHICLE MOTION BASED ON GAUSSIAN RANDOM WALK MODEL**

We can see that when the vehicle is in motion, the full vehicle position covariance matrix \( \Sigma_r \) and the yaw rotation of LIDAR rotation covariance will influence the precision of the terrain elevation map. Therefore, we obtain a reduced variance matrix from the full vehicle pose \( x_{IV} = (r_{IV}, \psi) \) to \( x_{I} = (r_{I}, \psi) \), where the \( V \) coordinate system only represents the yaw of the vehicle coordinate system and satisfies the following equations.

\[ r_{IV} = r_{I} \] \( \psi_{IV} = \psi \) \( \Sigma_{IV} = J \Sigma_{IV} J^T \) \( \Sigma_{IV} = J \Sigma_{IV} J^T \)

The state estimate model of the vehicle during motion is shown as (54)-(56); \( n_v \) and \( n_w \) is Gaussian noise vector. Therefore, the vehicle relative localization based on odometry are given by:

\[ x_{k+1} = \begin{bmatrix} \hat{x}_{k+1} \hat{v}_{k+1} \hat{\psi}_{k+1} \end{bmatrix} \] (54)

\[ \hat{x}_{k+1} = C_{IV}(\psi_k)^{-1}(J_{IV} \hat{I}_{IV} - J_{IV} \hat{I}_{IV}) \] (55)

\[ \psi_{k+1} = \psi_{k+1} = (\omega_k + n_{w,k}) \Delta t \] (56)

\[ J_p = \frac{\partial T_{k+1} \cdot r_{k+1}}{\partial T_k} = \frac{\partial (\Phi^{-1}_{Tk} (r_{Tk} T_{k+1}))}{\partial T_k} = C (\Phi^{-1}_{Tk} T_{k+1}) \] (48)

\[ J_r = \frac{\partial r_{k+1} \cdot r_{k+1}}{\partial r_{k+1}} = \frac{\partial (\Phi^{-1}_{Tk} (r_{Tk} T_{k+1}))}{\partial r_{Tk} T_{k+1}} = -C (\Phi^{-1}_{Tk} T_{k+1}) \] (49)

\[ J_f = \frac{\partial \Phi_{k+1}}{\partial \Phi_{k+1}} = \frac{\partial (\Phi^{-1}_{Tk} T_{k+1})}{\partial \Phi_{k+1}} = -C (\Phi^{-1}_{Tk} T_{k+1}) \] (50)
where \( v_k \) represents the velocity of the vehicle at time \( k \) and \( \omega_k \) represents the rotational angular velocity of the vehicle around the z-axis. The two values are provided by the IMU.

The covariance of \( x_k \) is then given by

\[
\Sigma_{\dot{x}x_k} = \text{cov}(x_k, \dot{x}k) = Q_{k+1} \Delta t^2
\]

The covariance of vehicle motion is written as

\[
\Sigma_{V}^{k+1} = F \Sigma_{V}^{k} F^T + GQ_{k+1}G^T
\]

From (58), the \( Q_{k+1} \) is expressed as

\[
Q_{k+1} = G^{-1}(\Sigma_{V}^{k+1} - F \Sigma_{V}^{k} F^T)(G^T)^{-1}
\]

Substituting (59) into (57), we obtain (60), as follows:

\[
\Sigma_{\dot{V}V_k} = G^{-1}(\Sigma_{V}^{k+1} - F \Sigma_{V}^{k} F^T)(G^T)^{-1} \Delta t^2
\]

Equation (60) can be simplified to obtain (61).

\[
\Sigma_{\dot{V}V_k} = \tilde{G}(\Sigma_{V}^{k+1} - \bar{F} \Sigma_{V}^{k} \bar{F}^T)\tilde{G}^T
\]

\[
\bar{F} = \begin{bmatrix}
 I^{3 \times 3} & (0) \\
 (0) & C_{I}V(\psi_k) F^{k+1} - j^{k+1} I_{V}V
\end{bmatrix}
\]

\[
\tilde{G} = \begin{bmatrix}
 C_{I}V(\psi_k) & 0^{1 \times 1} \\
 0^{1 \times 3} & 1
\end{bmatrix}
\]

The variance of vehicle motion from time \( k \) to time \( k + 1 \) is determined by the translation and rotation variances, given by

\[
\Sigma_{\dot{V}\dot{V}_k} = \begin{bmatrix}
 \Sigma_{\dot{V}} & 0 \\
 0 & \Sigma_{\dot{\phi}}
\end{bmatrix}
\]

V. TRAJECTORY PREDICTION BASED ON A VEHICLE KINEMATICS MODEL

In this section, the trajectory prediction method based on a vehicle kinematics model is proposed. Unlike the dynamic model, the kinematics model describes the motion of the vehicle according to various parameters of the vehicle motion without considering the force exerted on the vehicle. The kinematics trajectory prediction model is based on the hypothesis of vehicle kinematics theory, which assumes that some motion variables, such as speed and acceleration, are invariable with regard to trajectory predictions. According to [38], the effects of a constant yaw rate and the acceleration model should provide the most accurate outcome and as such, were used in this study. A schematic diagram of the vehicle kinematics model is shown in Fig.5.

The velocities along the x-axis and y-axis in the Cartesian coordinate system are given by:

\[
v_x(t) = v(t) \cdot \cos(\omega t + \theta_0)
\]

\[
v_y(t) = v(t) \cdot \sin(\omega t + \theta_0)
\]

\[
v(t) = a \cdot t + v_0
\]

The predicted trajectory is as follows:

\[
x(t) = \frac{a}{\omega^2} \cos(\omega t + \theta_0) + \frac{v(t)}{\omega} \sin(\omega t + \theta_0) + c_x
\]

\[
y(t) = \frac{a}{\omega^2} \sin(\omega t + \theta_0) - \frac{v(t)}{\omega} \cos(\omega t + \theta_0) + c_y
\]

\[
c_x = x_0 - \frac{v_0}{\omega} \sin(\theta_0) - \frac{a}{\omega^2} \cos(\theta_0)
\]

\[
c_y = y_0 + \frac{v_0}{\omega} \cos(\theta_0) - \frac{a}{\omega^2} \sin(\theta_0)
\]

According to the relative position relationship between the wheels and vehicle center, the predicted trajectory of each wheel can be obtained.

VI. SIMULATION AND EXPERIMENTAL RESULTS

A. SIMULATION IN GAZEBO

In this paper, the simulation experiment was conducted using a Gazebo simulator. Gazebo is a 3D dynamic simulator, that can effectively simulate robot motion in complex indoor and outdoor environments. There are two reasons for using this simulator. Robot control and sensor data acquisition are carried out via communication among robotic operating system (ROS) processes. In addition, Gazebo provides a set of sensor models to account for the GPS, IMU and LIDAR. Therefore, to preliminarily verify the effectiveness of the algorithm, we conducted a simulation model; the parameters of each sensor were set to the same parameters as the actual equipment. Figure 6 shows the simulation terrain environment that we built in Gazebo to simulate the real terrain.

In this section, we discuss the reconstruction of the terrain elevation map in real-time as the vehicle is moving in the simulation terrain environment, in an attempt to extract the elevation information in advance. The sensor model error and the uncertainty of vehicle motion were taken into consideration in the process of terrain mapping, and 2σ confidence intervals were used to judge the rationality of the estimated terrain elevation map. Because in the simulation with gazebo, we can’t get the real values of the terrain, therefore we use the estimated confidence intervals to describe the actual terrain at high fidelity. But in the next section in the outdoor experiments, we can obtain the roadblock’s true values. And we can
compare the estimated values with the true values to verify the effectiveness and the precision of the algorithm. The partial terrain elevation map and a local enlarged drawing are shown in Fig.7 and 8, respectively. The blue terrain map represents the upper bound of the confidence interval, and the green terrain map represents the lower bound of the confidence interval. The red terrain map represents the estimated terrain elevation map. The estimated confidence intervals are used to describe actual terrain at high fidelity. As we can see from the results, the estimated terrain elevation values were located in the confidence intervals, which indicates the rationality of the estimated terrain elevation map. Figures 9 and 10 describe the elevation values corresponding to the specific trajectory in the estimated terrain elevation map of Fig.7. The elevation values corresponding to the specific trajectory in the estimated terrain elevation map of Fig.11 are shown in Fig.12 and 13. The results indicate flat terrain with a low confidence internal and fluctuating terrain with a high confidence interval. In other words, the estimated precision of flat terrain was higher than that of the fluctuating terrain.

Based on the vehicle kinematics model, the predicted trajectories of the wheels are represented by a yellow line, as shown in Fig.14-17. In this paper, we only considered the
left front wheel; thus, the figures show preview elevation values of the left front wheel based on the predicted trajectory. Through this method, we can obtain the preview elevation information of the wheel in advance, which is used as an input to the suspension to improve the vehicle’s ride.

B. EXPERIMENT IN OUTDOOR

1) ACCURACY ANALYSES

Through the above simulation test, the rationality of the algorithm was verified preliminarily. However, the actual outdoor environment contains additional uncertainties that may not have been accounted for in the simulation environment, as well as various interference factors. Therefore, in this section, we discuss the experimental results obtained from tests conducted in an outdoor environment. A roadblock was set to test the precision of the algorithm, as shown in Fig.18. And we also can obtain the roadblock’s true values to compare the estimated values with the true values to get the precision of this algorithm.

We used a 32-line LIDAR of Leishen to obtain the point cloud. Through the fusion of IMU and GPS, odometry data were obtained. The GPS time was used for time
synchronization of the LIDAR and IMU, to ensure point cloud precision. The LIDAR measurement error, the uncertainty of motion estimation, and equipment calibration and fusion error all influence the precision of the estimated terrain elevation map. The experimental results showed that the fluctuation in the estimated terrain value is consistent with the real terrain. In Figs.19-21, the blue terrain map represents the upper bound of the confidence interval, the green terrain map represents the lower bound of the confidence interval, and the red terrain map represents the estimated terrain elevation map. The estimated terrain elevation values are located in the confidence intervals, which indicates the rationality of the estimated terrain elevation map. According to the vehicle kinematics model, the trajectory of the wheels could be predicted, and the terrain elevation values can be extracted in advance. Figure 22 shows preview elevation values of the left front wheel based on the predicted trajectory. From Fig.23, the lower and upper confidence intervals are larger than the simulation results due to the uncertainty in the outdoor environment. In this paper, we can obtained the tree values of roadblock. The maximum error of the estimated elevation value was 2.58cm; i.e., the precision of the algorithm was approximately 2cm.

2) COMPUTATIONAL LOAD ANALYSIS
In this section, we discuss the computational load of the proposed method for the given example. The times taken for each of the first 200 frames of data are listed in Table 1. This data was processed in real-time using an Intel Core i5-4460 CPU running at 3.2GHz with 8GB RAM.

Next, we discuss the relationship between the installation height, installation angle, maximum speed of the vehicle, and point cloud processing speed of the LIDAR, which is shown as Fig.24; the first and last laser beams are shown, and the laser beams in the middle are omitted. We show one frame point cloud. The LIDAR frequency is 10Hz, the installation height was about 1m, and the installation angle was 30°. x represents the horizontal distance between the current frame of the point cloud and the center of LIDAR.

The maximum value of the velocity was 3.085m/s. In addition, if we consider the response time of the suspension, the velocity would be less than 3.085m/s. The installation height and installation angle determine the size of the preview distance. The larger the preview distance, the lower the terrain elevation accuracy. Thus, velocity is a key consideration in terrain elevation precision.
The significance of this manuscript is that we use interdisciplinary knowledge to solve an engineering problem in the field of mechanical engineering, and the innovation of this manuscript belongs to Engineering Application Innovation. In future work, we will continue to improve the precision of the terrain elevation map by reducing calibration and fusion errors. A driver cognitive model will also be taken into consideration in the trajectory prediction process to further enhance the performance of the proposed approach.

### TABLE 1. Time to process a single frame of the point Cloud.

| The number of frames | Average time/s | The number of frames | Average time/s |
|----------------------|----------------|----------------------|----------------|
| 1-10                 | 0.2931         | 101-110              | 0.3512         |
| 11-20                | 0.3449         | 111-120              | 0.3778         |
| 21-30                | 0.3379         | 121-130              | 0.3430         |
| 31-40                | 0.3418         | 131-140              | 0.3688         |
| 41-50                | 0.3379         | 141-150              | 0.3505         |
| 51-60                | 0.3436         | 151-160              | 0.3343         |
| 61-70                | 0.3412         | 161-170              | 0.3773         |
| 71-80                | 0.3500         | 171-180              | 0.3413         |
| 81-90                | 0.3746         | 181-190              | 0.3985         |
| 91-100               | 0.3306         | 191-200              | 0.3112         |
| Average time of 200 frames |           |                      | 0.3475         |

### VII. CONCLUSION

This paper introduces a method to reconstruct the terrain elevation maps in real-time to extract preview elevation information according to the predicted trajectory of a vehicle in advance. The significance of this paper is that we use interdisciplinary knowledge to solve an engineering problem in the field of mechanical engineering, and the innovation of this manuscript belongs to Engineering Application Innovation. LIDAR noise and the uncertainties in the rotation of the pitch and roll are considered in the process of terrain elevation mapping. Based on Gaussian noise, we derived a LIDAR...
error model in the process of rotation and translation. In addition, the vehicle’s motion uncertainty is estimated and the error variance propagated to the terrain elevation map. According to the value of the Mahalanobis distance, we can judge large drops in the same cell. Measurement and motion updates are divided into two sections. When the cell does not receive new LIDAR measurements to update, the terrain elevation map will be updated based on the relative motion of the vehicle from its previous to its current pose, which is the benefit of this treatment. In addition, we use multithreading to process the two parts in code to improve the velocity of computation. As this is our first step in investigating preview elevation inputs, we concentrated on accurately extracting preview terrain elevation values. Here, the contribution of this paper is applying known methods in probabilistic robotics to improve the precision and efficiency of predicting elevation, as verified by a comparison between simulation and experimental results. In the future, we plan to focus on improving the precision of the terrain elevation map and investigating in more detail the algorithm and the trajectory prediction process that uses a dynamic vehicle model and maneuvering recognition to improve its accuracy.

REFERENCES
[1] M. S. Seong, S. B. Choi, M. W. Cho, and H. G. Lee, “Preview control of vehicle suspension system featuring MR shock absorber,” J. Phys., Conf. Ser., vol. 149, Feb. 2009, Art. no. 012079.
[2] I. Youn, M. A. Khan, E. Youn, M. Tomizuka, and N. Uddin, “Road disturbance estimation for the optimal preview control of an active suspension system based on tracked vehicle model,” Int. J. Automot. Technol., vol. 18, no. 2, pp. 307–316, 2017.
[3] J. Marzbanrad, G. Ahmadi, Y. Hojjat, and H. Zohoor, “Optimal active control of vehicle suspension system including time delay and preview for rough roads,” J. Vibrot. Control, vol. 8, no. 7, pp. 967–991, Jul. 2002.
[4] J. Hong and L. F. Li, “Research on preview control of MRF semi-active suspension based on LabVIEW,” Control Eng. China, vol. 17, no. 2, pp. 187–190 2010.
[5] Q. Wang and H. E. Ren, “Multi-objective control and simulation of vehicle active suspension based on wheelbase preview,” Chin. J. Construct. Mach., 2017.
[6] S. Ichikawa, Y. Yokoi, S. Doki, S. Okuma, T. Naitou, T. Shimado, and N. Miki, “Novel energy management system for hybrid electric vehicles utilizing car navigation over a commuting route,” in Proc. IEEE Intell. Vehicles Symp, Parma, Italy, Jun. 2004, pp. 161–166.
[7] C. Gohrle, A. Schindler, and A. Wagner, “Design and vehicle implementation of preview active suspension controllers,” IEEE Trans. Control System Technol., vol. 22, no. 3, pp. 1135–1142, May 201, doi: 10.1109/TCST.2013.2272342.
[8] W. Gang, C. Changzheng, and Y. Shenbo, “Finite frequency multi-objective control of vehicle active suspension with road preview information,” Trans. Chin. Soc. Agricult. Mach., 2015.
[9] J. Marzbanrad, G. Ahmadi, H. Zohoor, and Y. Hojjat, “Stochastic optimal preview control of a vehicle suspension,” J. Sound Vibrot., vol. 275, nos. 3–5, pp. 973–990, Aug. 2004.
[10] A. Ahmed and D. Purdy, “Control designer of active suspension system with terrain preview using evolutionary multi-objective algorithms.”
[11] V. L. Mulder, S. de Bruin, M. E. Schaepman, and T. R. May, “The use of remote sensing in soil and terrain mapping—A review,” Geoderma, vol. 162, nos. 1–2, pp. 1–19, Apr. 2011.
[12] K. Iagnemma, S. Kang, H. Shibly, and S. Dubowsky, “Online terrain parameter estimation for wheeled mobile robots with application to planetary rovers,” IEEE Trans. Robot., vol. 20, no. 5, pp. 921–927, Oct. 2004.
[13] W. Wan, M. Peng, Y. Xing, Y. Wang, Z. Liu, K. Di, B. Teng, X. Mao, Q. Yang, X. Xin, and M. Jia, “A performance comparison of feature detectors for planetary rover mapping and localization,” ISPRS-Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci., vol. 42, pp. 149–154, Jul. 2017.
[14] C. Wang, N. I. Jian-Jun, and H. X. Yin, “Robot simultaneous localization and mapping algorithm based on improved fuzzy adaptive Kalman filter,” Comput. Modernization, vol. 2014, no. 3, pp. 4–7, 2014.
[15] M. Khan, S. Hassan, S. I. Ahmed, and J. Iqbal, “Stereovision-based real-time obstacle detection scheme for unmanned ground vehicle with steering wheel drive mechanism,” in Proc. Int. Conf. Commun., Comput. Digit. Syst. (C-CODE), Mar. 2017, pp. 380–385.
[16] S. R. Sukumar, S. Yu, A. F. Koschan, M. A. Abidi, and D. L. Page, “Multi-sensor integration for unmanned terrain modeling,” Proc. SPIE, vol. 6230, May 2006, Art. no. 623008.
[17] G. Ishigami, M. Ottsuki, and T. Kubota, “Range-dependent terrain mapping and multipath planning using cylindrical coordinates for a planetary exploration rover,” J. Field Robot., vol. 30, no. 4, pp. 536–551, Jul. 2013.
[18] K. V. Stefanik, J. C. Gassaway, K. Kochersberger, and A. L. Abbott, “UAV-based stereo vision for rapid aerial terrain mapping,” GISci. Remote Sens., vol. 48, no. 1, pp. 49–59, Jan. 2011.
[19] A. Cappatanga, S. Cattani, A. Broggi, M. S. McDaniel, and S. Dutta, “Real time 3D terrain elevation mapping using ants optimization algorithm and stereo vision,” in Proc. IEEE Intell. Vehicles Symp., Jun. 2010, pp. 902–909.
[20] R. Triebel, P. Pfaff, and W. Burgard, “Multi-level surface maps for outdoor terrain mapping and loop closing,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., Beijing, China, Oct. 2006, pp. 2276–2282.
[21] D. M. Cole and P. M. Newman, “Using laser range data for 3D SLAM in outdoor environments,” in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), May 2006, pp. 1556–1563.
[22] J. Hidalgo-Carrió, P. Poulakis, and F. Kirchner, “Adaptive localization and mapping with application to planetary rovers,” J. Field Robot., vol. 35, no. 6, pp. 961–987, Sep. 2018.
[23] J. Ryde and H. Hu, 3D Mapping With Multi-Resolution Occupied Voxel Lists, Norwell, MA, USA: Kluwer, 2010.
[24] D. Belter, P. Labcki, and P. Skrzypczynski, “Estimating terrain elevation maps from sparse and uncertain multi-sensor data,” in Proc. IEEE Int. Conf. Robot. Autom., vol. 2, 1999, pp. 997–1002.
[25] N. Miki, “Novel energy management system for hybrid electric vehicles based on tracked vehicle model,” in Proc. IEEE Int. Conf. Robot. Autom., vol. 2, no. 3, pp. 1135–1142, May 20, doi: 10.1109/TCST.2013.2272342.
[26] J. R. E. Villanueva, L. I. Martínez, and J. I. Pérez Montiel, “DEM generation from fixed-wing UAV imaging and LiDAR-derived ground control points for flood estimations,” Sensors, vol. 19, no. 18, p. 3895, 2019.
[27] L. B. Cremen and R. M. Murray, “Uncertainty-based sensor fusion of range data for real time digital elevation mapping (RTDEM),” in Proc. IEEE Int. Conf. Robot. Automat., Barcelona, Spain, Apr. 2005, pp. 18–22.
[28] P. Fankhauser, M. Bloesch, C. Gehring, M. Hutter, and R. Siegwart, “Robot-centric elevation mapping with uncertainty estimates,” in Proc. Int. Conf. Climbing Walking Robots (CLAWAR), Poznan, Poland, 2014, pp. 433–440.
[29] D. Droeschel, M. Schwarz, and S. Behnke, “Continuous mapping and localization for autonomous navigation in rough terrain using a 3D laser scanner,” Robot. Auto. Syst., vol. 88, pp. 104–115, Feb. 2017.
[30] A. Klein and C. Dornhege, “Real-time localization and elevation mapping within urban search and rescue scenarios,” J. Field Robot., vol. 24, nos. 8–9, pp. 723–745, Aug. 2007.
[31] M. Labowski, P. Kaniawski, and P. Serafin, “Motion compensation for radar terrain imaging based on INS/GPS system,” Sensors, vol. 19, no. 18, p. 3895, 2019.
[32] R. Schubert, E. Richter, and G. Wanielik, “Comparison and evaluation of advanced motion models for vehicle tracking,” in Proc. 11th Int. Conf. Inf. Fusion, Jun./Jul. 2008, pp. 1–6.
LILI WANG received the bachelor’s degree in vehicle engineering from the Shandong University of Technology, in 2014, and the master’s degree from the College of Mechanical and Aerospace Engineering, Jilin University, Changchun, China, in 2016, where she is currently pursuing the Ph.D. degree. She has published two SCI articles. She received the National Scholarship, in 2018.

DINGXUAN ZHAO received the Ph.D. degree from the College of Mechanical and Aerospace Engineering, Jilin University, Changchun, China, in 1994.

He was a Visiting Scholar with the Wuteng Research Laboratory of Mechanical Systems Engineering, Ministry of Engineering, Gifu University, Japan. From 2000 to 2007, he was the Vice-Dean, a Professor, and a Ph.D. Tutor with the School of Mechanical Science and Engineering, Jilin University. From 2007 to 2010, he was the Vice-Dean, a Professor, and a Ph.D. Tutor with the Graduate School, Jilin University, where he was the Executive Vice President, a Professor, and a Ph.D. Supervisor, from 2010 to 2015. From 2015 to 2019, he was the Vice President of Yanshan University. He holds over 11 patents and two monographs. He has published more than 260 academic articles. He has undertaken 48 scientific researches at all levels, 45 completed, more than 30 included from the national and provincial and ministerial level projects or from the National Defence Science and Technology Project, and three projects under research, one National Key Research and Development Project and two National Defence Science and Technology Projects. His research interests fields hit a wide range of issues from engineering robot to complex mechanical system dynamics and it’s simulation to virtual reality, driving simulation technology, and so on. He received the five provincial and ministerial level awards. He was rated as the Cheung Kong Scholars Chair Professor, in 2009. He is an Academic Evaluation Committee Member of the Complex Mechanical System Dynamics Subject of the State Council, a Chief Scientist with the National Key Research and Development Plan, the Vice chairman of China Construction Machinery Society, and the Chairman of the China Construction Machinery Society’s Earthmoving Machinery Chapter.

TAO NI received the Ph.D. degree from the College of Mechanical and Aerospace Engineering, Jilin University, Changchun, China, in 2006.

From 2006 to 2008, he was a Postdoctoral Student with the Virtual Reality and Multimedia Systems Laboratory, Gifu University, Japan. He is currently a Ph.D. Supervisor with the College of Mechanical and Aerospace Engineering, Jilin University. He has published more than 20 SCI and EI articles. He holds over four invention patents and more than ten projects have been presented over by the National Natural Science Foundation, military projects, and provincial and ministerial projects. His current research interests include intelligent robot technology and application and complex system modeling and simulation.

SHUANG LIU received the Ph.D. degree from Yanshan University, Qinghuangdao, China, in 2010. He is currently with the School of Electrical Engineering, Yanshan University. His current research interests include nonlinear system modeling, rolling mill, friction, stability control, suspension, and so on.