Precise Localization of an Autonomous Car Based on Probabilistic Noise Models of Road Surface Marker Features Using Multiple Cameras

Kichun Jo, *Member, IEEE*, Yongwoo Jo, Jae Kyu Suhr, *Member, IEEE*, Ho Gi Jung, *Senior Member, IEEE*, and Myoungho Sunwoo, *Member, IEEE*

***Abstract*—This paper presents a Monte Carlo localization algorithm for an autonomous car based on an integration of multiple sensors data. The sensor system is composed of onboard motion sensors, a low-cost GPS receiver, a precise digital map, and multiple cameras. Data from the onboard motion sensors, such as yaw rate and wheel speeds, are used to predict the vehicle motion, and the GPS receiver is applied to establish the validation boundary of the ego–vehicle position. The digital map contains location information at the centimeter level about road surface markers (RSMs), such as lane markers, stop lines, and traffic sign markers. The multiple images from the front and rear mono-cameras and the around-view monitoring system are used to detect the RSM features. The localization algorithm updates the measurements by matching the RSM features from the cameras to the digital map based on a particle filter. Because the particle filter updates the measurements based on a probabilistic sensor model, the exact probabilistic modeling of sensor noise is a key factor to enhance the localization performance. To design the probabilistic noise model of the RSM features more explicitly, we analyze the results of the RSM feature detection for various real driving conditions. The proposed localization algorithm is verified and evaluated through experiments under various test scenarios and configurations. From the experimental results, we conclude that the presented localization algorithm based on the probabilistic noise model of RSM features provides sufficient accuracy and reliability for autonomous driving system applications.**

***Index Terms*—Precise localization, multiple cameras, road surface marker, probabilistic noise modeling, probabilistic noise model of road surface marker (RSM) features, particle filtering, autonomous car, autonomous driving.**

I. Introduction

VEHICLE localization is an important component of an autonomous driving system. Many autonomous driving algorithms (such as behavior decision, motion planning, and vehicle motion control) operate based on ego–vehicle position information estimated by a localization system. In early versions of autonomous cars, high-cost GPS and dead reckoning (DR) fusion solutions were usually applied for the localization systems [1]–[4]. The high-cost GPS/DR system provides very accurate position information; however, it is difficult to use in long GPS outage conditions, which occur frequently in urban canyon environments due to the DR sensor noise and the integration error of the DR. To overcome the GPS/DR quality degradation problem, a precise localization algorithm based on precise map and perception data should be used in the autonomous driving system.

This paper presents a precise localization algorithm based on an *information fusion of automotive on-board motion sensors, a low-cost GPS, cameras, and a precise digital map*. The automotive on-board sensors (yaw rate sensors and wheel speed sensors) are used to measure the vehicle dynamic motion. A low-cost GPS receiver that is widely used for commercial navigation systems is employed to set up the initial position of the vehicle and the position validation boundary. Three types of cameras are installed to detect the road surface marker (RSM) features. The digital map database contains roadway geometry and RSM features information.

This precise localization algorithm integrates the information from the multiple sensors using a particle filter. Because the particle filter integrates the sensor measurements in a probabilistic manner, the probabilistic noise modeling of the sensor measurement is key to improving filtering performance. In this paper, we use the RSM features detected by the multiple cameras for measurement updates. The RSM features contain several noise sources, such as projection error, due to unexpected vehicle motion, RSMs occupied by unexpected objects, and false positive RSM features detection. To consider the noise effects more explicitly, we analyze the noise characteristics of the RSM features using a camera geometry model. Based on the results of the analysis, we design a probabilistic noise model of RSM features and apply it into the localization algorithm.

In addition to the probabilistic noise model of the RSM features, we integrate an around-view monitoring (AVM) system into the localization algorithm. An AVM is a parking-assist system that provides information about the nearby surroundings of the ego–vehicle from a virtual bird's-eye view image. To generate the bird's-eye view image, four wide-angle cameras and a separate computing module are installed on the vehicle platform. By integrating the nearby bird's-eye view image into

Manuscript received January 25, 2015; revised April 30, 2015; accepted June 20, 2015. Date of publication July 16, 2015; date of current version November 23, 2015. This work was supported in part by an NRF grant funded by the Korean government (MEST) (No. 2011-0017495), by the Industrial Strategy Technology Development Program of MKE (No. 10039673), and by the BK21 plus program (22A20130000045) under the Ministry of Education, Republic of Korea. The Associate Editor for this paper was N. Zheng.

K. Jo, Y. Jo, and M. Sunwoo are with the Automotive Control and Electronics Laboratory (ACE Lab), Department of Automotive Engineering, Hanyang University, Seoul 133-791, Korea (e-mail: msunwoo@hanyang.ac.kr).

J. K. Suhr is with the Research Institute of Automotive Control and Electronics, Hanyang University, Seoul 133-791, Korea.

H. G. Jung is with the Department of Automotive Engineering, Hanyang University, Seoul 133-791, Korea.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2015.2450738

the localization system, we obtain many benefits to improve the localization performance.

The benefits of the AVM and the validity of the probabilistic noise model of the RSM features are evaluated through experimental studies performed under various driving conditions, such as urban and motorways. The experimental results show that the accuracy and reliability of the proposed localization system are appropriate for application to an autonomous driving system under real driving conditions.

The rest of this paper is organized as follows. Section II summarizes the related research for vehicle localization. Section III presents a system overview of the test vehicle platform and mapping system. Section IV describes the fundamental flow of the particle-filter-based localization algorithm. Section V presents the measurement update process of the particle filter based on the probabilistic noise model of the RSM features. Section VI presents the probabilistic noise modeling of the RSM features using multiple cameras. Section VII describes the experimental results for the proposed algorithm, and the final section provides conclusions.

II. Related Works

Localization systems based on integration of GPS and vehicle motion information (DR) are widely applied for ego–vehicle positioning due to the complementary features of the various sensor [5]–[10]. Low update frequency and short time outage of the GPS can be aided by DR updates, while the long-term integration error of the DR can be corrected by GPS. However, during long time GPS outage, such as in an urban canyon environment, the GPS/DR integration cannot overcome the integration error of the DR. Furthermore, the systematic noise property of the GPS that changes slowly with time is hard to correct by only using DR assistance [11]–[13].

To overcome the limitation of the GPS/DR system, localization systems based on matching of a digital map with perception data have been widely researched. As in [12], [13], the map-perception-aided localization system can be classified into three groups: 1) Kalman-filter-based localization at the feature level, 2) signal-level Monte-Carlo localization, and 3) feature-level Monte Carlo localization, as shown in Fig. 1.

A. Kalman-Filter-Based Localization at the Feature Level

Kalman filtering (extended Kalman filtering for nonlinear systems) is a typical solution for a map-perception-aided localization algorithm. The process of Kalman-filter-based localization at the feature-level is shown in Fig. 1(a). The features extracted from sensor measurements (such as cameras, radars, and LIDARs) are associated with landmarks saved in the digital map. Based on the association results, the Kalman filter updates the state mean and covariance to estimate the position and its uncertainty. Because Kalman-filter-based localization has the advantages of a simple calculation framework and a low computational burden, many previous studies have applied this method.

Object features from LIDARs with highly accurate landmark maps were used for vehicle localization based on extended Kalman filtering [14]. In [15], grid map features of LIDARs with a lane-level map were used for localization. The canny edge features of road markings with satellite images maps were applied for localization based on two data association methods: iterative closest point algorithm (ICP) and iterative recursive least squares (IRLS) [16]. Several types of road surface markers (lanes, stop lines, and traffic signs) and features (lines, corners, intensities, and edges) are based on the Kalman-filter-based localization algorithm [17]–[22]. Road structural features (RSFs) detected from line segments and points of images were used for ego–vehicle position estimation in [23]. The Mercedes Benz S500 INTELLIGENT DRIVE achieved fully autonomous driving on the Bertha-Benz-Memorial-Route based on a Kalman-filter-based localization system using lane and point features from multiple camera images [24]–[27].

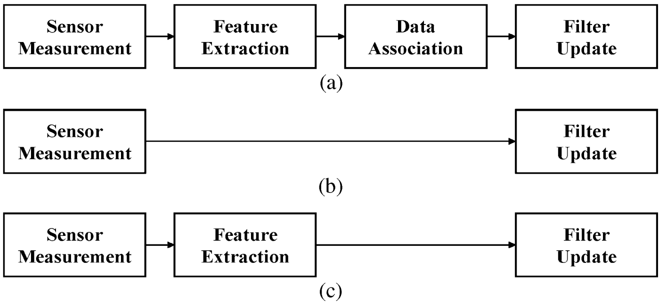


Fig. 1. The three groups for map-perception-aided localization: feature level Kalman filter, signal level particle filter, and feature level particle filter. (a) Kalman filter based localization on feature level. (b) Signal level Monte Carlo localization. (c) Feature level Monte Carlo localization.

Although the Kalman-filter-based localization at the feature-level for the advantage of a simple calculation process, there are limitations when it is applied in cluttered environments and nonlinear system models. In cluttered environments, data association is not easy because the associations of sensor measurements with landmarks are confused by the clutter. Because the standard Kalman filter only provides an estimation framework for linear system models, it cannot be applied to nonlinear system models. Extended Kalman filters and Unscented Kalman filters can approximate the probability distributions of nonlinear systems; however, these filters cannot cover the multi-modal probability distribution of the systems.

B. Signal-Level Monte-Carlo Localization

To deal with system nonlinearity and reduce the data association effort, a particle filter based on Monte-Carlo sampling is used in the localization system. There are two types of Monte-Carlo localization methods: signal-level and feature-level localization.

Signal-level Monte-Carlo localization directly updates the raw sensor data without feature extractions, as shown in Fig. 1(b). Because the raw sensor measurements are directly updated for position estimation, there is no loss of sensor information from data abstraction of the features. In [28], the reflectivity data from LIDARs are directly applied to the measurement updates of the localization system. Because all the reflectivity data for roads should be stored in the map database, the localization system requires huge memory storage. In [12], [13], the camera image was directly updated for the localization algorithm using a coherency value derived from the structure tensor. This algorithm does not require a feature extraction process for image data; however, it requires huge computational power to update all the raw image data. Therefore, this method may be useful when the computational resources are adequate and parallel computing is available.

C. Feature-Level Monte-Carlo Localization

A second type of Monte-Carlo localization is feature-level localization. Unlike signal-level Monte-Carlo localization, feature-level Monte-Carlo localization performs the feature extraction process before the filter update. Because the filter is updated after abstracting the measurements into features, the computational burden may be smaller than for signal-level Monte-Carlo localization. The performance of the localization is determined by which features are used and how the noise characteristics are modeled.

Curb features extracted from dense point clouds of a 3D LIDAR were used for filter updates [29]. Because the raw point cloud data is abstracted into curb features, it can reduce the required computational resources; however, the algorithm may be useless in a curb-less region. The most widely used features for localization are road surface markers (RSMs) such as lanes, traffic signs, stop lines, and crosswalks. A maximally stable extremal region (MSER) that represents bright areas from a camera and LIDAR grid map was used for localization updates [30], [31]. A combination of cameras and range sensors (radar and LIDAR) was used to detect the RSMs and landmarks for localization [32], [33]. A vision-only localization system used RSM features (lanes, crosswalks, traffic signs, and stop lines) to update the localization filter.

There are many studies for RSM based localization; however, many of these studies did not explicitly analyze the noise characteristics of the RSM features for measurement updates. A measurement model that applies exact sensor noise characteristics can highly improve the accuracy and reliability of localization [34]. Therefore, in this paper, we present an analysis of noise characteristics of RSM features detected from cameras, and design a probabilistic noise model of RSM features. The designed probabilistic noise model of multiple cameras is the basis of the proposed feature-level Monte-Carlo localization algorithm.

III. System Overview

A. Test Car Configuration

A test vehicle (Fig. 2) equipped with electronic stability control (ESC) was used to evaluate the localization algorithm. The ESC system contains several types of vehicle motion sensors (a steering angle sensor, wheel speed sensors, and a yaw rate sensor) to detect the vehicle's dynamic motion. This ESC sensor information is shared through the in-vehicle network (controller area network, CAN); therefore, we can access the motion sensor information of the ESC by connecting to the CAN. Two types of vehicle motion sensors (wheel speed sensors and a yaw rate sensor) were used for the localization algorithm, and the specifications of the motion sensors are listed in Table I.

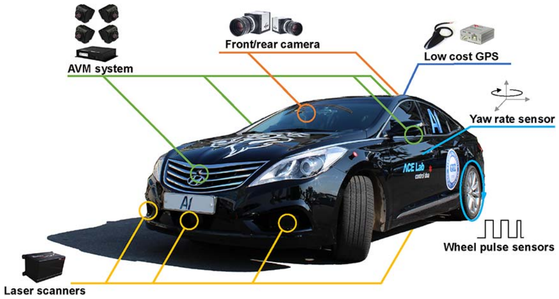


Fig. 2. Sensors of test vehicle used to evaluate the localization system are composed of multiple cameras (AVM, front and rear camera), low-cost GPS receiver, vehicle motion sensors (yaw rate and wheel pulse sensors), and laser scanners.

TABLE I
Specifications of On-Vehicle Sensors to Measure the Vehicle Motion

Sensor	Range	Frequency	Resolution	Noise (RMS)	Unit
Yaw rate sensor	± 120	50 Hz	0.0625	0.5	deg/sec
Wheel speed sensor	0–130	50 Hz	0.035	0.3	m/s

TABLE II
Specifications of Low-Cost GPS Receiver

Accuracy	Position	2.5 m circular error probability (CEP)
	Satellite-based augmentation system	2.0 m CEP
	Precision timing	30 ns root mean square (RMS)
Data rate		5 Hz

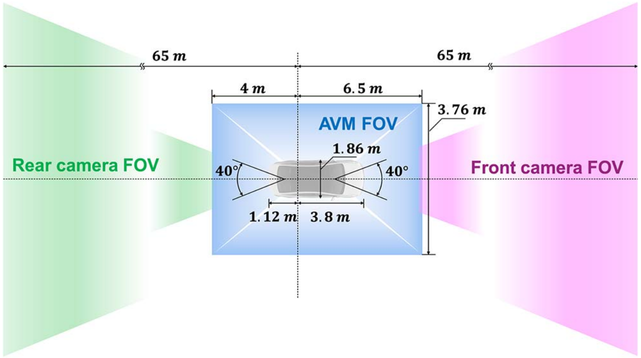


Fig. 3. Field of view (FOV) for multiple cameras (AVM, front and rear camera) used for the proposed localization algorithm.

Positioning information obtained from a low-cost GPS was used for initialization and the GPS measurement updates. We installed Ublox EVK-6T and used a shark antenna, which was basic equipment on the vehicle. The specifications of the GPS receiver are given in Table II.

The test vehicle was equipped with three types of cameras: a front camera, a rear camera, and an AVM system. As shown in Fig. 3, the front and rear camera respectively provide images of a wide field of view (FOV) in the front and rear of the vehicle, whereas the AVM system provides an accurate image of a narrow FOV near the vehicle. These images are used to

TABLE III
SPECIFICATIONS OF MULTIPLE CAMERAS (AVM, FRONT AND REAR CAMERA)

Sensor	Resolution	Area of interest	Pixel size	Focal length	Intensity resolution	Frame rate
Front camera	1280×1024	1280×320	1/1.8"	8 mm	8 bits/color	15 fps
Rear camera	768×576	760×140	1/1.8"	12 mm	8 bits/gray	10 fps
AVM system	640×480	200×280	-	-	8 bits/color	10 fps

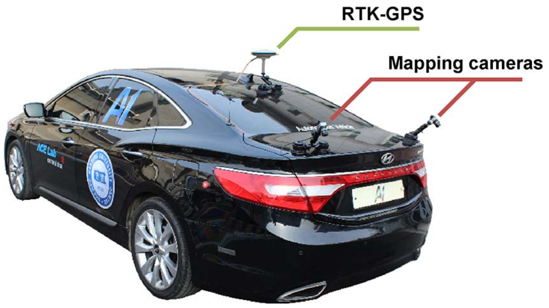


Fig. 4. Mapping system equipped with centimeter-level accuracy RTK-GPS and two down-looking camera is used to capture the road surface image.

detect the RSM features on the roadways. The specifications of the cameras are listed in Table III.

Laser scanners used to detect surrounding objects were applied to remove undesired RSM features from the camera images. Four multi-layer laser scanners were installed on the test vehicle: three were installed inside the front bumper, and the other inside the rear bumper, as shown in Fig. 2.

On the test vehicle was mounted an industrial embedded computer that could endure harsh environments, with shock protection (50 G, IEC 60068-2-27) and vibration protection (5 Grms, IEC 60068-2-64). The computer included an Intel core i7 processor and 16 gigabyte RAM to run the localization algorithm.

B. Map Database

The precise digital map contained the RSM landmark features, such as solid lines, dashed lines, stop lines, and traffic signs. A probe vehicle equipped with a GPS, vehicle motion sensors, and two down-looking cameras was used to generate the RSM feature map, as shown in Fig. 4. A real-time kinematic (RTK) GPS that provided centimeter-level accuracy was used to measure the global position of the vehicle, and vehicle motion sensors (gyro sensors and wheel speed encoder) were applied to correct the position in case of GPS outage. The two down-looking cameras mounted on the rear of the vehicle were used to capture the road surface images.

Images obtained from the two down-looking cameras are converted to bird's-eye view images, and they are stitched into a single global image based on the corresponding vehicle position from the RTK-GPS-based positioning system. RSM features are extracted from the single global image using a matched filter based on the top-hat kernel [35]. Then, the extracted RSM features are abstracted as polylines by using sequential processing of thinning, clustering, and line fitting algorithms. Fault features and undetected RSM features are precisely corrected manually. The abstracted RSM polylines are converted into the Open Street Map file format, which is a widely used standard structure for storing map information [36]. Fig. 5 shows an example map based on the Open Street Map file format. A set of nodes constructs a way that represents RSMs on the map, and a set of ways are clustered as a relation.

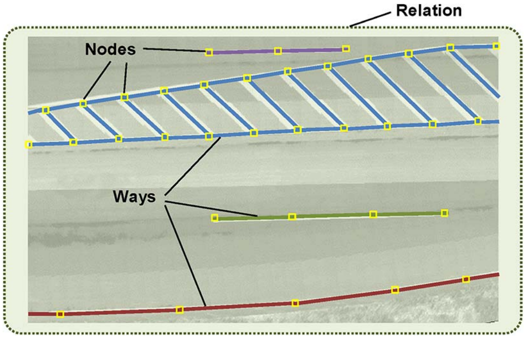


Fig. 5. Open street map database for perception-map-aided localization consists of nodes, ways, and relation in order to represents the RSMs.

IV. LOCALIZATION BASED ON PARTICLE FILTER

Fig. 6 represents the overall architecture of the localization algorithm. The process of the localization system is composed of the time update and measurement update. At the time update step, the probabilistic distribution of the vehicle state is predicted based on the vehicle motion model, as shown in Fig. 6(a). The shape of the probabilistic distribution of the vehicle motion is described as crescent similar nonlinear distribution. At the measurement update step, the GPS and map-perception measurements are used to update the vehicle position distribution. The probabilistic density function of the GPS can be described as the uniform distribution, as shown in Fig. 6(b). The probabilistic distribution of the RSM feature measurement can be formed as a multimodal because an RSM feature can be matched with multiple landmarks on the map, as shown in Fig. 6(c). Because many of the probabilistic distributions of the measurements are non-Gaussian distributions (nonlinear transformations of Gaussian distributions, uniform distributions, and multimodal distributions), the particle filter is used to integrate the various types of probabilistic distributions.

The principle of the particle filter is to approximate the complete non-Gaussian probability density of the state vector conditioned on measurements using Monte-Carlo sampling. The particle filter process consists of the following steps.

A. Initialization

It is assumed that the probability density function of the initial state $p(\hat{x}_{0|0})$ is known. The initial N particles described in $[x^i_{0|0}]^N_{i=1}$ are randomly generated based on $p(\hat{x}_{0|0})$. The parameter N is a design factor that is a trade-off between the computational load and the accuracy of estimation.

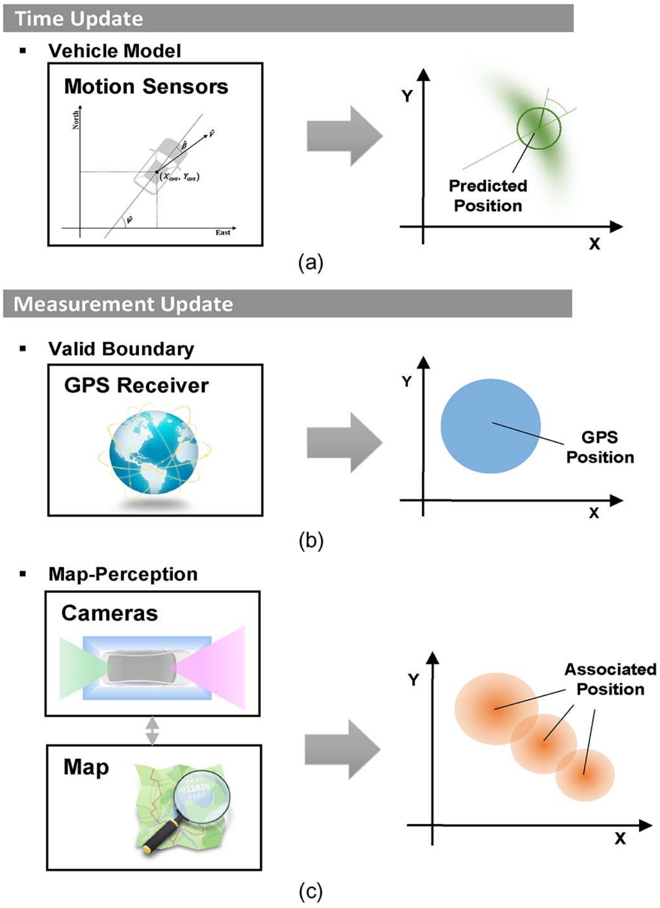


Fig. 6. Overall architecture of the localization algorithm can be described using probability density distribution of time and measurement updates.

In this paper, $p(\hat{x}_{0|0})$ is derived using the vehicle position and velocity, and the receiver's health status data obtained from the GPS receiver. If the health status of the GPS signal satisfies the pre-defined initialization conditions (number of satellites in use is over than five and horizontal dilution of precision (HDOP) is below two), the state of the particles are initialized by the position and velocity data from the GPS.

B. Time Update

The next step's estimates are created based on the system model. A vehicle kinematic model with on-board sensor inputs is used for the system model of time updates, as shown in (1) at the bottom of the page. The vehicle model contains three states: a vehicle heading ψ and a position in two global coordinates (X, Y).

$$\hat{x}^i_{k|k-1} = \begin{bmatrix} \psi^i_{k|k-1} \\ X^i_{k|k-1} \\ Y^i_{k|k-1} \end{bmatrix} = \begin{bmatrix} \psi^i_{k-1|k-1} + \Delta T \cdot \hat{w}^i_{XY_k} \\ X^i_{k-1|k-1} + \Delta T \cdot \hat{V}^i_{XY_k} \cos\left(\psi^i_{k-1|k-1} + \Delta T \cdot \hat{w}^i_{XY_k}\right) \\ Y^i_{k-1|k-1} + \Delta T \cdot \hat{V}^i_{XY_k} \sin\left(\psi^i_{k-1|k-1} + \Delta T \cdot \hat{w}^i_{XY_k}\right) \end{bmatrix}. \tag{1}$$

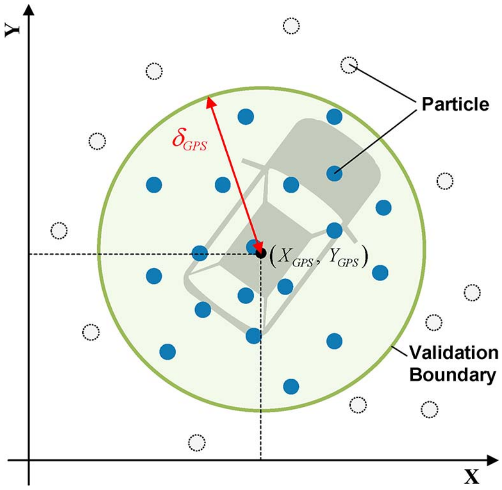


Fig. 7. Validation boundary can be described as uniform probability distribution using GPS measurements in a two-dimensional plane.

The control input of the vehicle model can be obtained from the vehicle motion sensors: a yaw rate sensor w_{gyro} and wheel speed sensors V_{whl}. Because the localization algorithm is based on the particle filter, we can predict the next step's position candidates using the random samples (2), where ε_b is a zero-mean error variable with standard deviation b

$$\begin{aligned} \hat{w}^i_{XY_k} &= w_{gyro_k} + \varepsilon^i_{\sigma_{gyro}} \\ V^i_{XY_k} &= V_{whl_k} + \varepsilon^i_{\sigma_{whl}}. \end{aligned} \tag{2}$$

The probabilistic distribution of a time update based on the vehicle motion model is shown in Fig. 6(a). The shape of the probabilistic distribution can be changed via the configuration of sensor noise $\varepsilon_{\sigma_{gyro}}, \varepsilon_{\sigma_{whl}}$.

C. Measurement Update

The weight of each particle q^i is updated by evaluating the likelihood of sensing measurements $p(z|x, m)$ conditioned on the predefined state and the map. The relationship between weight and likelihood can be expressed as follows:

$$q^i_k \propto q^i_{k-1} p\left(z_k | x^i_{k|k-1}, m\right). \tag{3}$$

There are two types of measurement for the localization: a GPS position measurement and a map-perception matching measurement.

1) GPS Update: The first measurement is obtaining position data from the GPS receiver. Because a low-cost GPS receiver has systematic noise properties due to atmospheric disturbances, it is unsuitable to model the GPS noise as a Gaussian distribution. The systematic noise properties can be regarded as unknown bias from the true position [11]. If we know the boundary of the unknown bias, we can set the noise model as a uniform probability distribution, as shown in Fig. 7. The radius of the validation circle δ_{GPS} can be determined based on the GPS health status, such as number of satellites in use, horizontal dilution of precision (HDOP), and type of differential correction. If a particle is located inside the validation circle, the value of the weight is set to one in order to maintain the previous value; however, if the particle is located outside it, the weight is set to zero. When all particles have been set to zero, the localization system is reinitialized

$$q^i_{GPS_k} = \begin{cases} 1 & \text{for } \left(\hat{X}^i_k - X_{GPS}\right)^2 + \left(\hat{Y}^i_k - Y_{GPS}\right)^2 < {\delta_{GPS}}^2 \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

2) Map-Perception Update: The second measurement is to obtain RSM features from the multiple cameras. A likelihood of the measurement can be obtained from a probabilistic function conditioned on the particle state and the map database. The weight of the RSM feature is updated based on the likelihood value, as shown in

$$q^i_{RSM_k} = p\left(z_k | x^i_{k|k-1}, m\right). \tag{5}$$

The likelihood function for the RSM feature measurement is the most important contribution of this paper. Therefore, we will present more detailed algorithms for the measurement update and likelihood function derivation in Sections V and VI.

3) Integration and Normalization: Because the measurements from the GPS and the cameras are independent of each other, the integrated weight of a particle can be represented by multiplying (4) with (5) based on the relationship represented in (3), as shown in

$$q^i_k = q^i_{k-1} \times q^i_{GPS_k} \times q^i_{RSM_k}. \tag{6}$$

After calculating the weights of all the particles based on the likelihoods of the GPS position and the RSM features, we can obtain the normalized weights using a normalization process, as shown in (7). The obtained weights are called importance weights, and they describe the probability of the particle's being sampled from the correct probability density function

$$q^i_k = \frac{q^i_k}{\sum\limits_{j=1}^{N} q^j_k}. \tag{7}$$

D. Density Approximation

To apply the position estimate to other autonomous driving algorithms, we should find one representative value from the particle's state and weight. If the posterior of the system has a unimodal property, it is suitable to apply the minimum mean square error (MMSE) estimate, which can be approximated by the weighted mean of the particle density distribution, as shown in

$$\hat{x}_{k|k} = \sum_{i=1}^{N} q^i_k \hat{x}^i_{k|k-1}. \tag{8}$$

However, the likelihood distribution of the RSM feature measurement can be formed as a multimodal distribution because an RSM feature can be matched with multiple landmarks on the map, as shown in Fig. 6(c). In a multimodal distribution, the mean value cannot represent the overall probability distribution. In this case, it is better to use a maximum a-posteriori estimate for the representation, as shown in (9). To approximate the maximum a-posteriori estimate, we apply a mean-shift algorithm, which iteratively approximates the maximum a-posteriori [37]

$$\hat{x}_{k|k} = \arg\max_{\hat{x}_{k|k-1}} p\left(\hat{x}^i_{k|k-1} | z_{1:k}, m\right). \tag{9}$$

E. Resampling

Resampling regenerates the new set of randomly generated particles on the basis of their relative likelihoods. Resampling prevents the concentration of the probability mass on only a few particles [38]. The resampling will be executed only if the following condition is met

$$\hat{N}_{eff} = \frac{1}{\sum\limits_{i=1}^{N} \left(q^i_k\right)^2} < \frac{2}{3}N. \tag{10}$$

N_{eff} represents the effective number of samples that indicates the degree of depletion. When all the particles have uniform weight values, the N_{eff} has its maximum value (N). In contrast, when all the weights are devoted to a single particle, the N_{eff} has its minimum value (*one*). We choose the upper threshold of N_{eff} as $2/3N$; however, the value can be modified according to the probabilistic characteristics of the filter.

A low-variance sampling method is used for resampling. This method is based on a systematic sampling technique that selects particles that have lower computational complexity and better spatial coverage of the sample space [34].

V. Measurement Update Using RSM Features

This section presents the measurement update process of RSM features, which was explained in the map-perception likelihood update in the previous section. The update process of the RSM features is composed of a likelihood field generation and a measurement update. In the likelihood field generation step, the likelihood field conditioned on the predicted state and the map is generated by reflecting the noise characteristics of the RSM feature detection. Based on the generated likelihood field, the weight of each particle is updated by applying the likelihood field to the detected RSM features.

A. Likelihood Field Generation

The likelihood field describes the distribution of likelihood for the measurement data in the measurement space. The likelihood distribution in the likelihood field can be obtained from the likelihood function (5). The likelihood function can be obtained by reflecting the measurement noise characteristics to the measurement prediction model $h(x, m)$. The measurement prediction model $h(x, m)$ can predict the sensor measurement by applying the geometric relationships between the sensors' installed locations, the predicted vehicle state, and the map.

We present an example to illustrate the measurement update process in Fig. 8. The polylines that are blue lines with yellow rectangle nodes represent the map of the RSM features. Points in the green series indicate the positions of the predicted states of particles, and the attached line indicates the heading of the

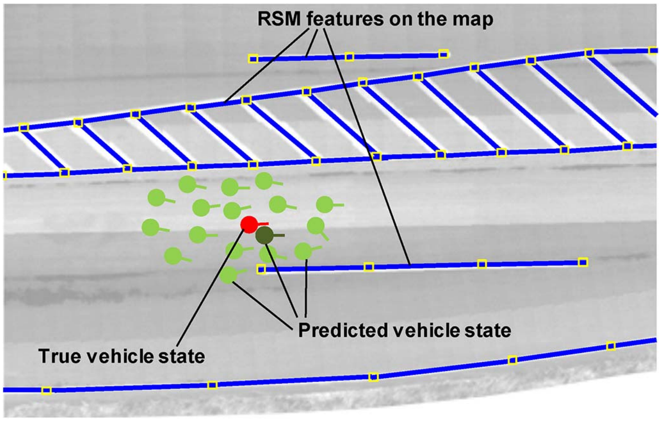
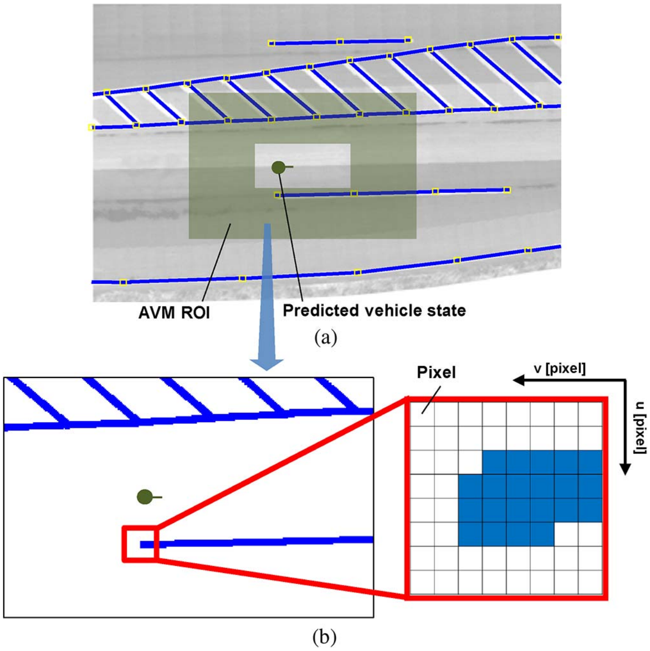


Fig. 8. Example for measurement update using the RSM features of the AVM. The blue lines mean the RSM features on the map, red circle means the true vehicle state, and the greenish circles represents the predicted vehicle state using particle filter.



(a)

(b)

Fig. 9. Prediction model of the RSM feature measurements. The model can be represented in image coordinates with pixel units.

predicted state. The red point and line represent the true vehicle state. An AVM is installed to provide a bird's-eye view image near the ego–vehicle with limited FOV, as shown in Fig. 3.

The dark green point indicates the representative particle to explain the measurement prediction. Using the geometric relationships among the AVM FOV, the predicted particle state, and the RSM features map, the measurement prediction model can be generated, as shown in Fig. 9(a). Because the AVM provide a bird's-eye view image in image coordinates, the measurement model is also represented in image coordinates with pixel units (Fig. 9(b)).

The probabilistic distribution of measurements (likelihood field) is generated by the likelihood function while applying the *RSM detection noise characteristic* to the measurement prediction (Fig. 9(b)). Fig. 10 shows the two-dimensional (2D) likelihood field for RSM feature detection, which is a 2D Gaussian mixture function. Appropriate noise modeling for the likelihood function can greatly affect the quality of the measurement update and the entire performance of the localization system [34]. In this paper, we design a probabilistic noise model of RSM features based on RSM detection results under real driving conditions. The detailed process of the probabilistic noise modeling of RSM features is described in the next section.

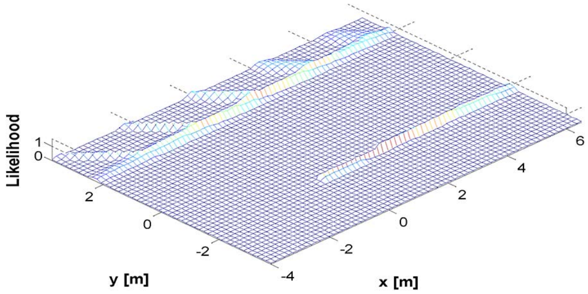


Fig. 10. Likelihood field of RSM features. The likelihood field can be represented using the 2D Gaussian mixture functions.

B. RSM Feature Detection

The AVM system provides a bird's-eye view image of the true vehicle position and heading, as shown in Fig. 11(a) and (b). From the bird's-eye view image, the RSM features can be extracted by applying a matched filter with a top-hat kernel [35]. The filtering algorithm is only intended to detect RSM features; however, there are also undesired features detected from other vehicles, pedestrians, and many objects that belong in a roadway. Laser scanners are used to detect the undesired objects, and the RSM features within the object regions are rejected from the images. Although the matched filter can detect the line-like RSM robustly, and laser scanners reject undesired features effectively, there are still problems such as false-positive RSM detection and undetected RSMs, as shown in Fig. 11(c). To consider the detection noise of the RSM features, we will include the RSM detection noise in the probabilistic noise model for the likelihood update function, which will be introduced in the next section.

C. Measurement Update

The likelihood of the representative particle in the example is updated by intersecting the likelihood field (Fig. 10) with the RSM feature detection results (Fig. 11), as shown in Fig. 12(a). Because there is a difference between predicted state of the particle and true vehicle state, the overlapping area is small, as shown in Fig. 12(b). If the particle locate on the true vehicle state (Fig. 12(c)), the particle will have the most wide overlapping area that means the highest likelihoods.

Multiple cameras (an AVM, a front camera, and a rear camera) are used for the measurement of the localization system (Fig. 3). Because the each camera image is independent to each other, the likelihood of RSM measurement can be obtained as

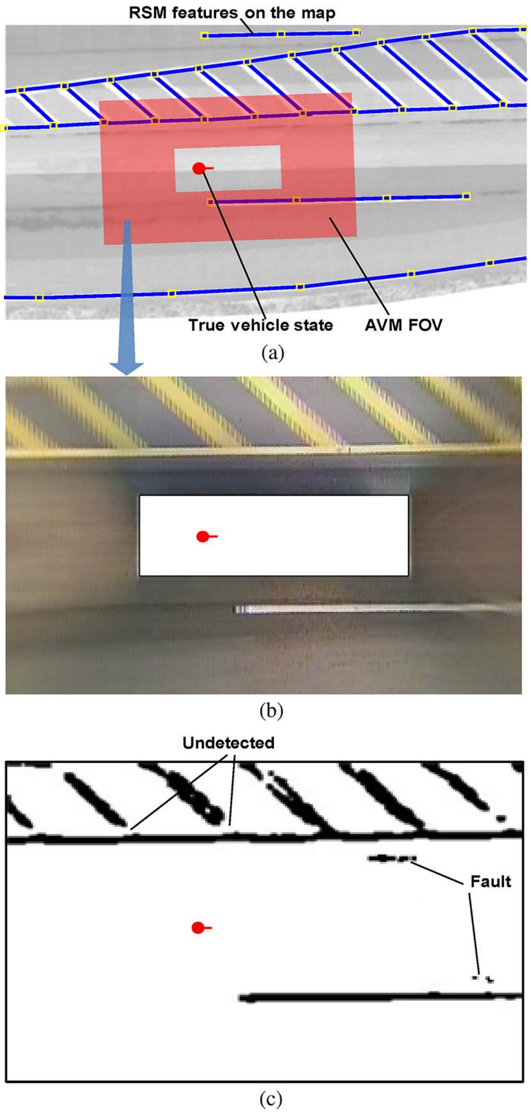


Fig. 11. RSM feature extraction process using a matched filter. (a) Red region describes the FOV of the AVM. (b) Image from the AVM system. (c) Results of the RSM feature extraction.

the production of the individual measurement likelihoods of each camera, as shown in

$$p\left(z_k|x^i_{k|k-1}, m\right) = \prod_{s}^{CAM} p^s\left(z^s_k|x^i_{k|k-1}, m\right) = p^{AVM}\left(z^{AVM}_k|x^i_{k|k-1}, m\right) \cdot p^{FCAM}\left(z^{FCAM}_k|x^i_{k|k-1}, m\right) \cdot p^{RCAM}\left(z^{RCAM}_k|x^i_{k|k-1}, m\right) \tag{11}$$

where CAM describes the types of camera.

VI. Probabilistic Noise Modeling of RSM Features

Proper noise modeling of the measurement for the likelihood function largely influence to accuracy and reliability of the Monte-Carlo localization [34]. The probabilistic noise model of RSM features can be classified into two factors: unexpected ego–vehicle motions and fault detected RSM features.

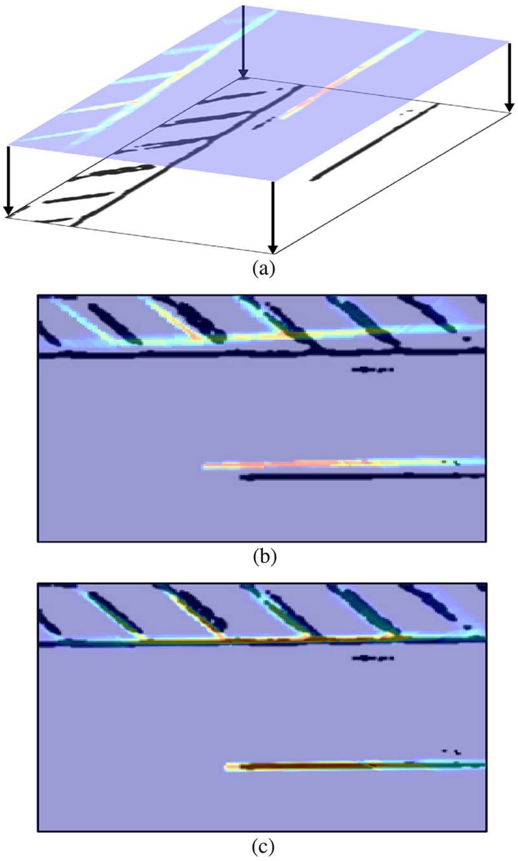
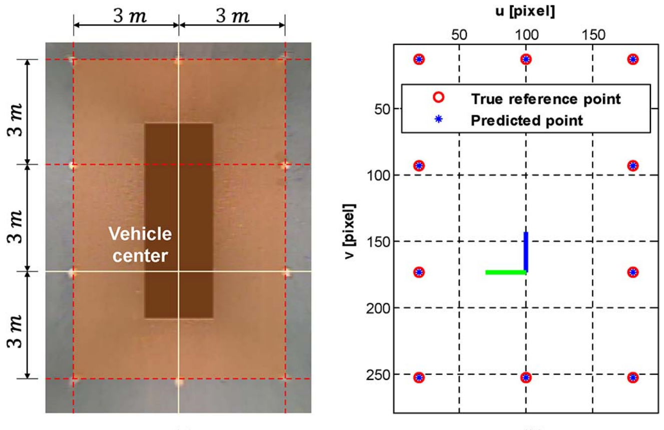


Fig. 12. Measurement update process of the RSM features. (a) Measurement update by matching the likelihood field with RSM features. (b) Example of the poor matching. (c) Example of the correct matching.

A. Noise From Unexpected Vehicle Motion $p_{motion}(z^n_k|x_k, m)$

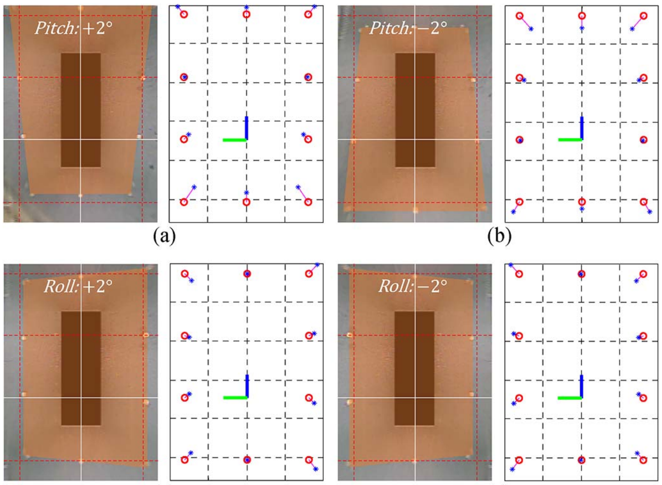
The location of the RSM features in the ego–vehicle body coordinates can be calculated from RSM feature measurements from the image by applying the geometric relationship between the camera pose and the road surface. The geometric relationship can be obtained from the camera's intrinsic and extrinsic calibration parameters. In the AVM example, if the point RSM features are located near the ego–vehicle at regular intervals (Fig. 13(a)), we can predict the AVM image using the geometric relationships (Fig. 13(b)). However, when vehicle motion (pitch and roll) occurs, the geometry relationships will be affected by the motion, which will cause measurement error of RSM feature positions. If we know the amount of vehicle motion, the measurement error of the RSM features also can be predicted using the geometric relationships, as shown in Fig. 14.

Vehicles are affected by unexpected vehicle motion in real-driving conditions; therefore, the RSM feature noise from the vehicle motion should be considered for the measurement model of localization. Feature noise due to unexpected motion can be modeled using real vehicle motion data from real driving conditions. We obtain real vehicle motion data using



(a) (b)

Fig. 13. AVM image measurement prediction of RSM features using the camera geometry relation.



(a) (b)

(c) (d)

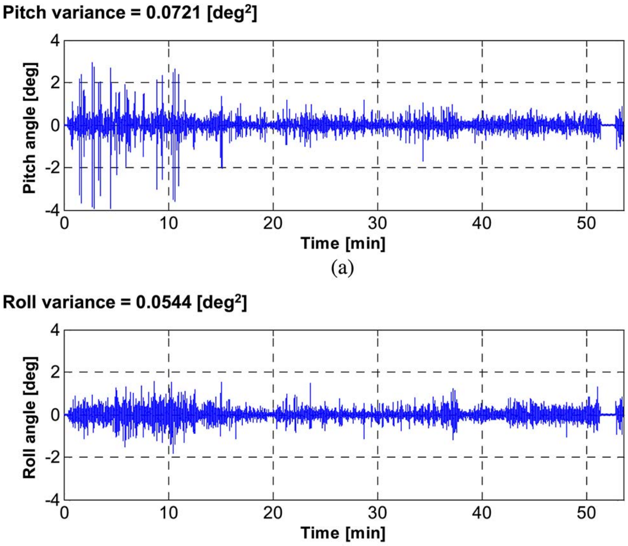
Fig. 14. Image measurement error of RSM features due to the ego–vehicle motions (roll and pitch). When the vehicle motion contains ±2° pitch angles, the upper and lower sides of the AVM image are enlarged or diminished in (a) and (b). When the vehicle motion contains ±2° roll angles, the right and left sides of the AVM image are enlarged or diminished in (c) and (d).

the precision measurement equipment of the vehicle's dynamic states under various driving conditions, such as highway, urban, and rural road conditions, as shown in Fig. 15.

By applying real vehicle motion data to geometric relationships between the camera and the road surface, the noise distribution of the AVM for the RSM feature points can be obtained as shown in Fig. 16. The distribution of each pixel in the AVM image can be modeled as a Gaussian distribution.

The noise modeling process can apply to all the pixels of the AVM image. The variance of the Gaussian noise distribution for all pixels of the AVM image can be represented as shown in Fig. 17. In the measurement noise model of the AVM features, the variances tend to increase with greater distance from the ego–vehicle.

The probabilistic noise model of the RSM features also can apply to the front and rear cameras. The variances of the Gaussian noise distributions for the front and rear cameras in image coordinates are shown in Fig. 18(a) and (b). The variances of the measurement noise models for the front and rear camera features tend to decrease longitudinally away from the ego–vehicle.



(a)

(b)

Fig. 15. Vehicle roll and pitch motion data from precision measurement equipment of the vehicle's dynamic state.

The predicted measurement using the predicted state and the map are composed of several pixels of RSM features, as shown in the AVM example (Fig. 9(b)). By applying the probabilistic noise model of the RSM features to all the pixels in the predicted measurement, the likelihood of unexpected motion noise can be obtained using a Gaussian mixture, as described in

$$p_{motion}^{CAM}\left(z_k^{CAM}|x_k,m\right)=\sum_{j=1}^{M}\frac{1}{M}p_{motion}^{CAM}\left(\left(z^j\right)_k^{CAM}|x_k,m\right) \tag{12}$$

M represents the number of predicted feature pixels in the prediction measurement, and CAM describes the types of cameras (AVM, front camera, and rear camera).

B. Noise From Fault Detection RSM Features $p_{fault}(z_k^n|x_k,m)$

A top-hat-kernel-based matched filter is used for the RSM features detection. However, the matched filter not only detects the RSM features, it also detects the undesired noise features, such as a bush, an unexpected object that is not contained in the map. Therefore, this unexpected feature noise can be regarded as random noise and can be represented using a uniform distribution, as in

$$p_{fault}^{CAM}\left(z_k^{CAM}|x_k,m\right)=\frac{1}{\text{Number of pixels on the image}}. \tag{13}$$

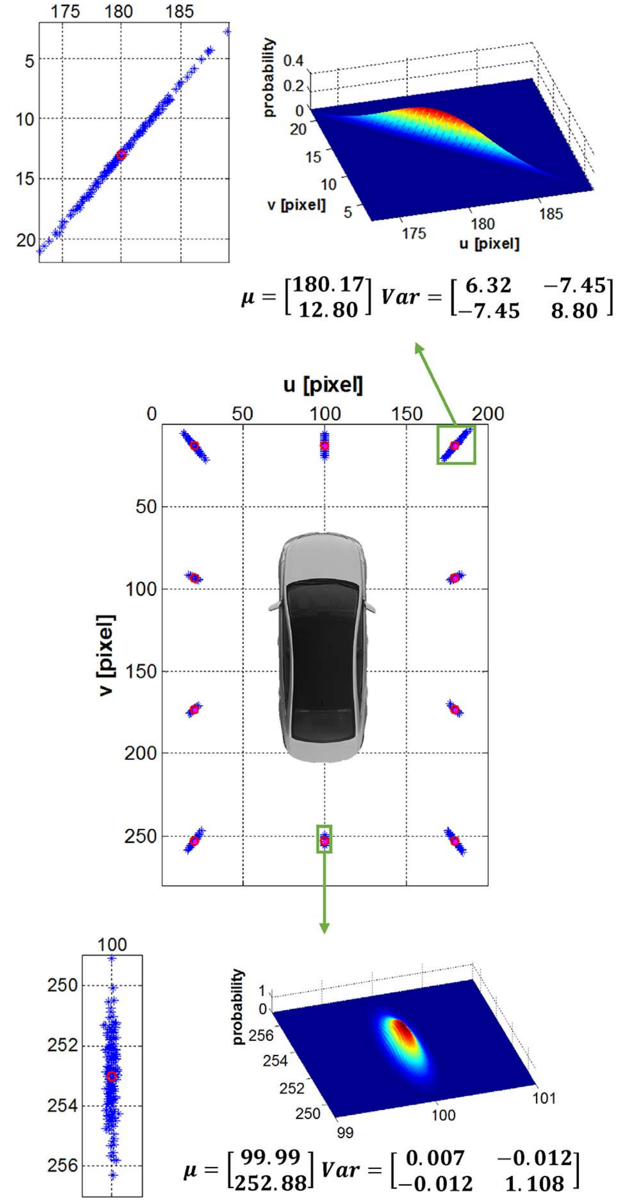


Fig. 16. (a) Noise distribution of the AVM for the RSM features. (b) Modeling of noise distribution from unexpected vehicle motion using a Gaussian distribution.

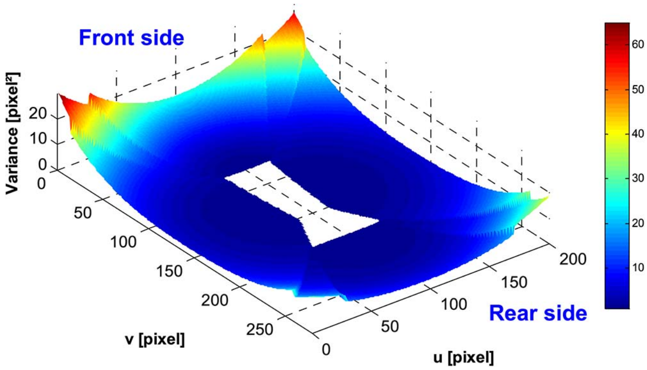


Fig. 17. Variance of the Gaussian noise distribution for the AVM system due to the unexpected vehicle motion.

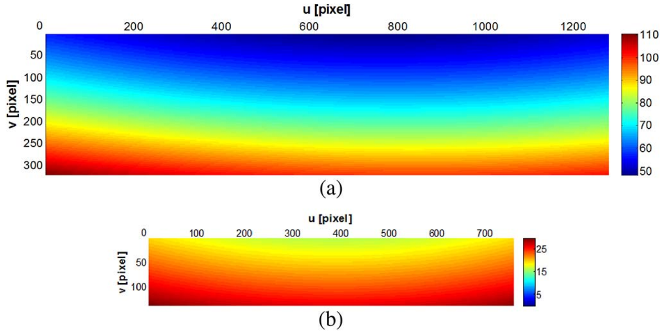


Fig. 18. Variances of Gaussian noise distributions for front camera and rear camera due to the unexpected vehicle motion.

C. Integrated Probabilistic Noise Model of RSM Features

The integrated probabilistic feature model of the RSM features can be described by mixing the two distributions with weighted averages, as represented in

$$p^{CAM}(z_k|x_k,m) = \alpha^{CAM} \cdot p^{CAM}_{motion}(z_k|x_k,m) + \beta^{CAM} \cdot p^{CAM}_{fault}(z_k|x_k,m). \quad (14)$$

The weight factors (α and β) for each probabilistic model can be determined by using the ratio of true-positive detection rate to false-positive detection rate of RSM features, as in

$$\frac{\text{True-positive rate}}{\text{False-positive rate}} = \frac{\alpha}{\beta}. \quad (15)$$

Each rate can be obtained from the manual labeling of RSM detection results under real driving conditions. Because the weight factors are determined by the detection rate of the RSM feature detector, the integrated probabilistic noise model of the RSM can be changed according to the types and performance of the RSM feature detector.

The AVM example for the integrated probabilistic noise model of RSM features is shown in Fig. 19.

VII. Experimental Results

A. Experimental Environments

1) Test Site and Driving Input: The proposed localization system was evaluated at the ITS proving ground of the Korea Automobile Testing and Research Institute (KATRI) located in Hwaseong-si, Gyeonggi-do, Korea. The ITS proving ground is a performance assessment facility for ITS applications including autonomous driving. To evaluate the ITS applications in various real driving scenarios, the proving ground provides many types of test sites that represent real driving conditions, such as a ramp, rural roads, a highway, a roundabout, intersections, and a parking lot, as shown in Fig. 20. The experiments for the localization evaluation were performed by following the red curved path, and the vehicle speed and yaw rate of the experiments are shown in Fig. 21.

2) Algorithm Implementation: The proposed Monte Carlo localization used 120 particles for the random sampling. Four computing threads (AVM image processing thread, front

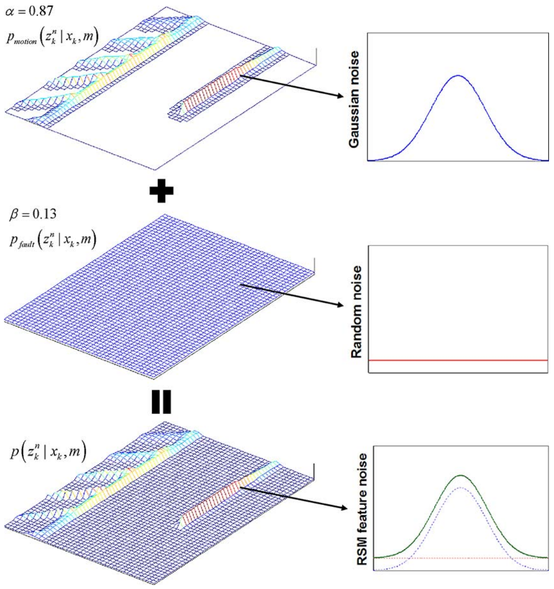


Fig. 19. Integration of two types of measurement noise models (unexpected motion and fault detection) generates the overall RSM feature noise model.

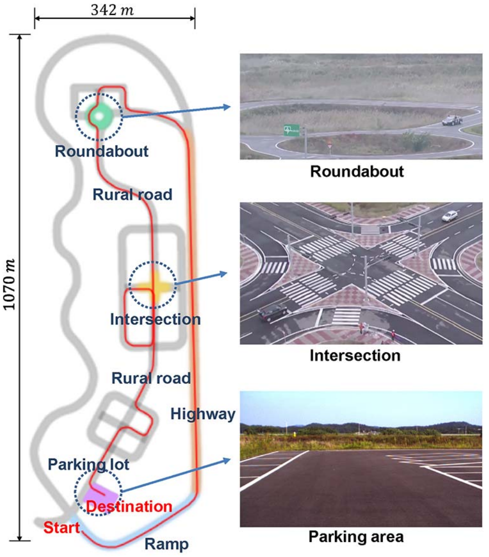


Fig. 20. Test site for evaluation of the localization algorithm (KATRI). The test site imitates the real driving conditions (such ramp, highway, intersection, roundabout, and parking lot).

camera processing thread, rear camera processing thread, and localization thread) are assigned to execute the localization algorithm. The average execution time is about 30 millisecond based on Intel i7-4700EQ CPU 2.4G computing unit. The localization algorithm needs less than ten-megabyte memory for the algorithm execution.

3) Monte-Carlo Experiment: Because the proposed localization algorithm is based on an Monte-Carlo localization that uses random sampling, experimental results under the same test conditions can be different from each other despite having the same control inputs. Therefore, it is not reliable to evaluate the performance of localization by using the results of only one of the experiments. To reliably evaluate the algorithm based on random sampling, the evaluation uses the Monte-Carlo method, which repeats the experiment several times with different random samples and the same control inputs.

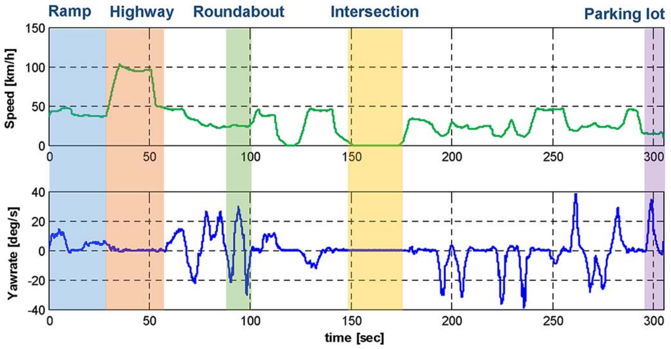


Fig. 21. Input of the vehicle speed and yaw rate to evaluate the proposed localization system.

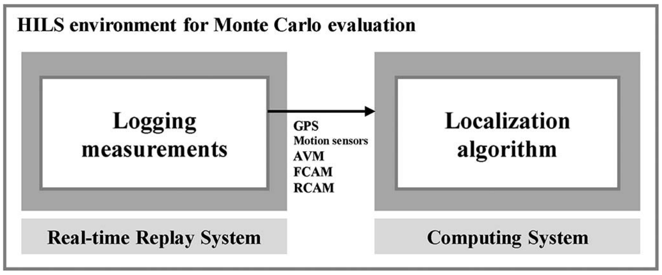


Fig. 22. Hardware-in-the-loop (HILS) environment for Monte-Carlo evaluation. The real-time replay system replay the logging measurements and transfer the logged data to localization computing system to evaluate the localization algorithm.

For the Monte-Carlo experiments, a hardware-in-the-loop simulation (HILS) was set up, as shown in Fig. 22. The HILS consists of a real-time replay system to log measurements (from the GPS, cameras, map, and vehicle motion sensors) and a computing system to execute the algorithm. The same performance computer was installed in the test vehicle as was used for the HILS. The replay system provides the logging measurements to the computing system in real time, and the computer executes the localization algorithm based on the real-time replay measurements. By using the HILS, we can perform a Monte-Carlo evaluation by repeating the experiments several times in the same measurement environment.

B. Localization Accuracy Analysis

A real-time kinematic GPS (RTK-GPS) with centimeter-level accuracy was used to analyze the accuracy of the proposed localization system. The accuracy was evaluated based on the error between the estimated state of the localization system and the true state of the RTK-GPS. Heading error, Euclidian distance error, lateral position error, and longitudinal position error were used for the evaluation, and these errors were obtained by averaging the results of 100 trials of Monte-Carlo experiments. Fig. 23 shows an error comparison result of the Kalman-filter-based GPS/DR system (green) and the proposed localization system based on the integration of a GPS, DR, map, and RSM feature detection (blue). From the RMS error from all of the state error, we conclude that the proposed localization system has superior accuracy to that of the GPS/DR system.

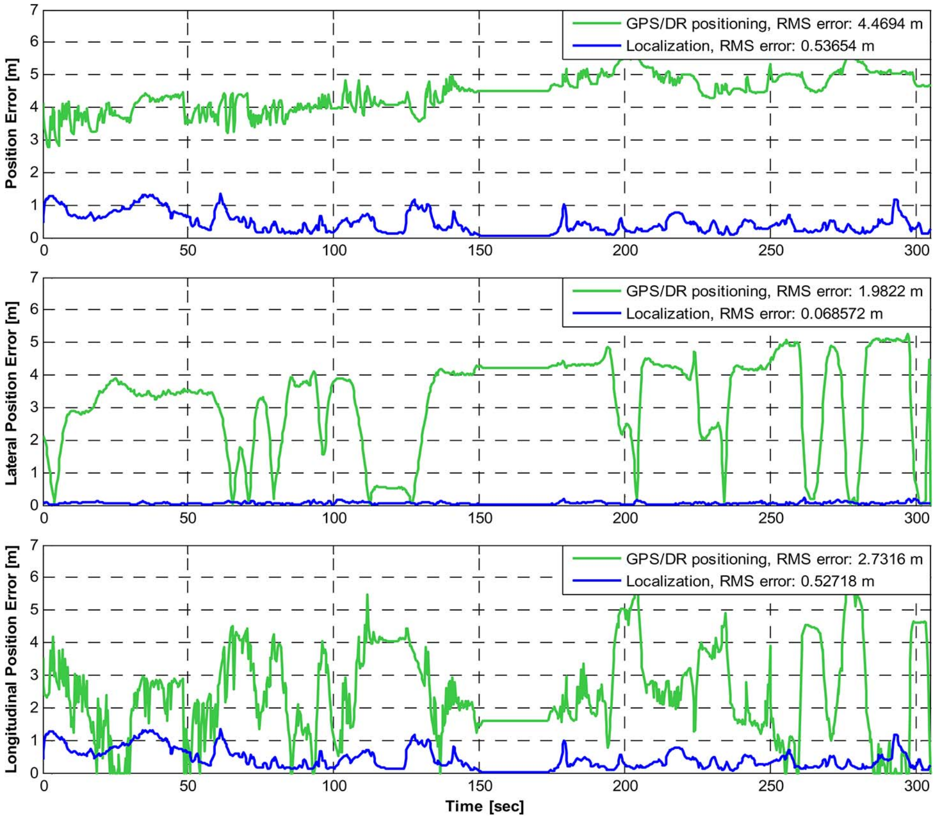


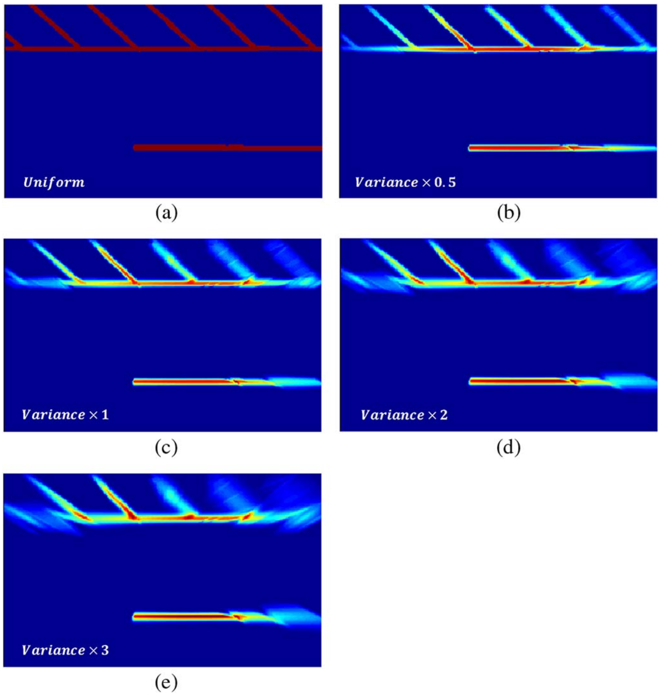
Fig. 23. Error plot of the localization based on information fusion of the GPS/DR and the GPS/DR/map/perception (distance error, lateral position error, and longitudinal position error).

Because there are many RSM features that can be used to correct the lateral position error, such as several types of lanes (dashed lanes, solid lanes, and double lanes), the lateral error (RMS 0.076417 meters) is smaller than the longitudinal error (RMS 0.52718 meters). The longitudinal error can be reduced under conditions where there are many RSM features to correct the longitudinal error, such as stop lines, crosswalks, and parking lot lines.

There were no fails or divergence of the localization using the proposed algorithm for the 100 trials of Monte-Carlo experiments. Furthermore, we verified the reliability and accuracy of the proposed localization algorithm by implementing it into a real autonomous driving system that participated in an autonomous vehicle competition organized by the Hyundai Motor Group [39].

C. Effect of the Probabilistic Modeling of the RSM Features

To analyze the effect of the probabilistic image model of the RSM features on the localization performance, we conducted experiments with various probabilistic image models of the RSM features. The test group of the probabilistic image model consisted of uniform probabilistic distribution and various times of variances for the original probabilistic noise model (1/2, 1, 2, and 3 times). The uniform probabilistic distribution model represents the image model which does not consider the noise effect of the RSM feature detection.



(a) (b) (c) (d) (e)

Fig. 24. Example of variance models: uniform probabilistic distribution model and various times of variances for the original probabilistic noise model (1/2, 1, 2, and 3 times).

Fig. 24 shows an example of uniform distribution and various times of the original probabilistic probability image model shown in Fig. 10. The larger variance model has a smoother and broader distribution.

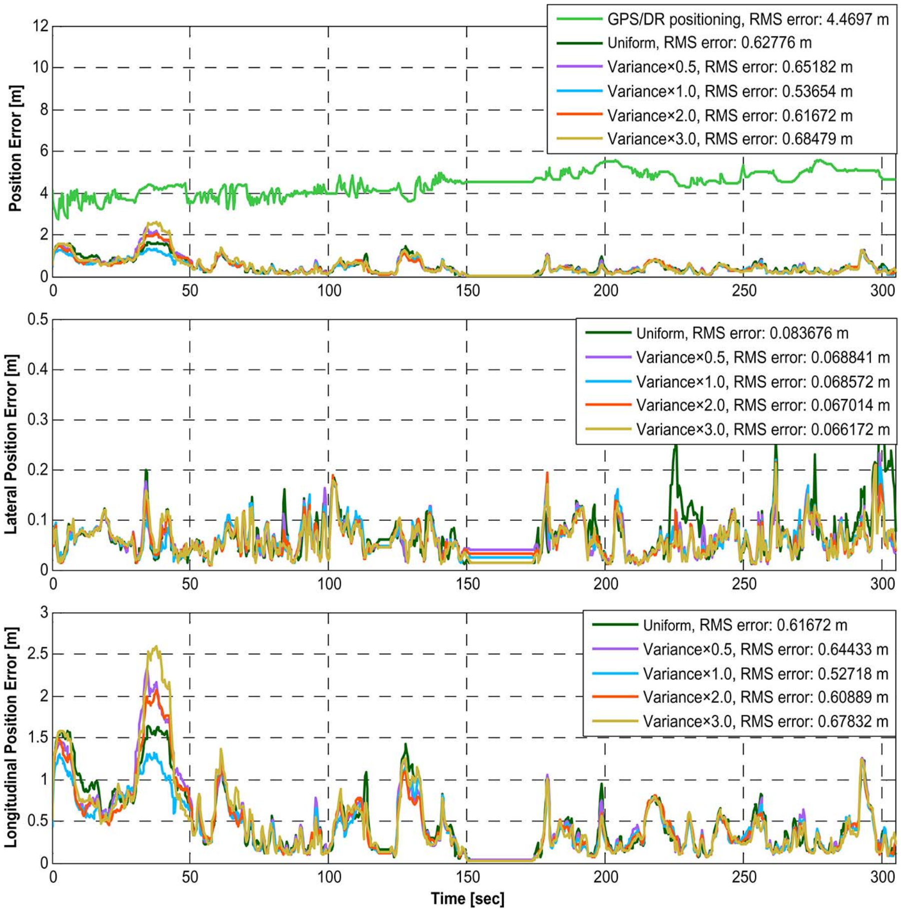


Fig. 25. Error plot for the difference variance probabilistic measurement model of RSM features. The RMS error of the each results represents the performance of the each noise model based localization algorithm.

TABLE IV
FAILURE RATE FOR THE VARIOUS MEASUREMENT MODELS

Variance	Uniform	1/2 times	1 times	2 times	3 times
Failure rate [%]	26	2	0	2	0
RSM error [m] (success set)	0.62776	0.65182	0.52489	0.61672	0.68479

A variance test using the original probabilistic noise model resulted in the minimum error among the test scenarios, as shown in Fig. 25 and Table IV. This indicates that the probabilistic noise model generated by the vehicle motions obtained from the real driving conditions is the most reasonable for precise localization.

Large variance probabilistic noise models resulted in large localization error. That is because an RSM feature model with large variance generates a wide distribution of the likelihood field. The wide likelihood field influences the scoring weights of each particle even though they are not correctly matched.

Small variance probabilistic noise model tests (include the uniform distribution test) also resulted in large localization error, and occasionally, the localization failed. That is because an RSM feature model with small variance generates a narrowly distributed likelihood field. Only if the RSM features are correctly matched in the RSM features model, then the particle is highly weighted; however, the other particles are not weighted even though they are close to the correct matching.

From these results, we conclude that the correct probabilistic measurement model is highly influential on the accuracy and reliability of the localization performance.

D. Advantage of the AVM System for Localization

Most of the cameras for the intelligent vehicle are installed inside the windshield to protect them from rain, snow, and various temperature conditions. The proposed localization system also utilizes front and rear cameras installed inside the front and rear windshields. Because the cameras are mounted at relatively high positions in the vehicle, they provide the advantage of broad FOVs. However, the image plane is almost perpendicular to the ground surface, so the RSM detection is highly vulnerable to the vehicle motion (pitch and roll).

As shown by the above model results, because the AVM system consists of down-looking cameras, it is highly robust to the noise of vehicle motion and can detect the RSM more accurately than the front and rear camera. Furthermore, the down-looking camera makes the AVM system robust to the illumination changes due to the direct sunlight. However, the AVM system only provides a narrow FOV near the ego–vehicle, so it could fail to localize where there are not many RSMs.

The experiments to evaluate the usefulness of the AVM system for localization were conducted under four different

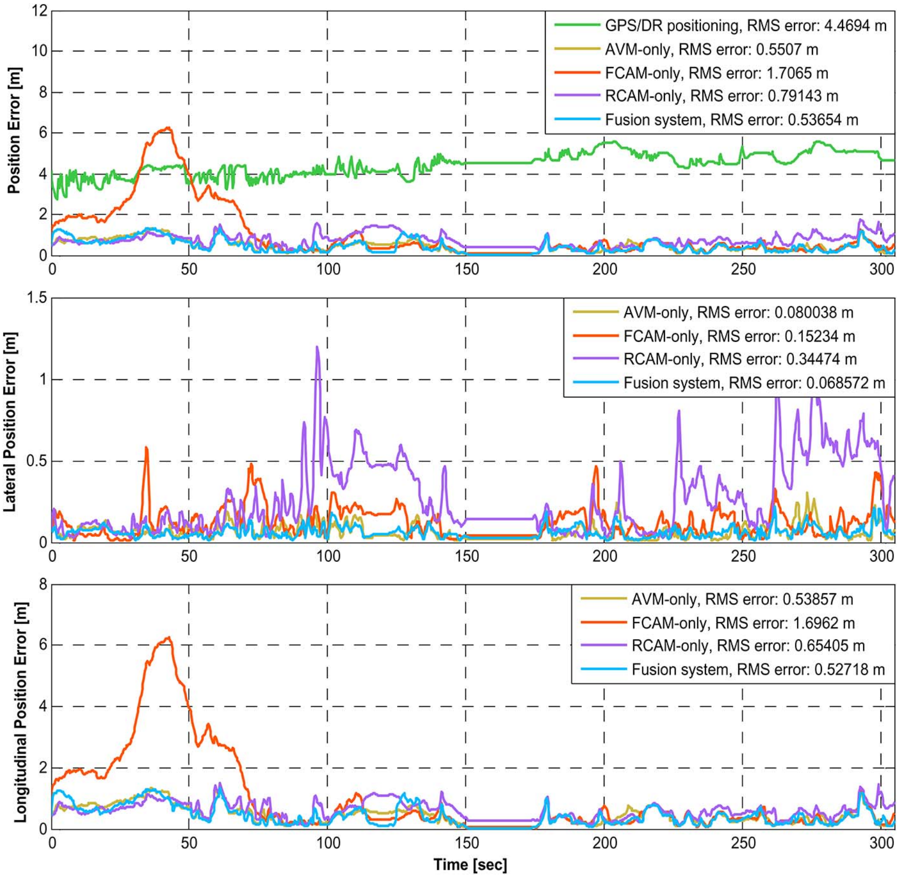


Fig. 26. Error plot of the localizations based on AVM-only, front camera-only, rear camera-only, and integrated cameras. The RMS error of the each results represents the performance of the each sensor components based localization algorithm.

scenarios: AVM-only, front camera-only, rear camera-only, and all three cameras integrated, as shown in Fig. 26.

The AVM-only test results showed that the localization was successfully performed in the test scenarios that contain enough RSMs on the road. However, AVM-only localization has the possibility to fail in a real driving situation in which there may not be sufficient RSMs in the AVM image because of occlusion from nearby vehicles. To overcome this sparse RSMs problem, the localization system must use the front and rear camera.

The localization test using only the front camera resulted in large localization errors. That is because the front camera image is highly influenced by the effects of the vehicle, as shown in Fig. 18(a). When the vehicle accelerates or decelerates, especially in the highway section, the vehicle pitch motion is enlarged. The front camera image is considerably affected by the pitch motion, and then it creates a discordance between the RSM features in the image and the RSM feature model. As shown in Fig. 26, the longitudinal error in the highway section exceeds five meters.

Even though the rear camera configuration is similar to that of the front camera, the localization error of the rear camera-only test was smaller than in the front camera-only test. That is because the rear camera observes a road surface far from the ego–vehicle due to the current camera setting. As shown in Fig. 18(a) and (b), the variance of the noise model for the rear camera is smaller than the error model of the front camera.

TABLE V
FAILURE RATE FOR THE VARIOUS CAMERA CONFIGURATIONS

Variance	AVM-only	Front camera-only	Rear camera-only	Integrated
Failure rate [%]	0	0	16	0
RSM error [m] (success set)	0.5507	1.7065	0.79143	0.53654

Despite the high accuracy of the rear camera-only localization, it has the risk to fail about 16 times over 100 trials as shown in Table V.

The test integrating multiple cameras (AVM, front camera, and rear camera) resulted in the minimum RMS error in longitudinal and lateral directions. The wide FOV images obtained from the front and the rear camera provides enough RSMs such that the localization system roughly estimates the ego–vehicle position, and the AVM image provides accurate RSMs to confirm the precise ego–vehicle position.

VIII. Conclusion

This paper proposed a precise localization algorithm for an autonomous driving system based on an information fusion of automotive on-board motion sensors, a low-cost GPS, multiple cameras, and a precise digital map. The localization algorithm integrates the data from the multiple sensors using a Monte-Carlo localization based on a particle filter method. The main advantages of the proposed localization algorithm can be summarized as follows.

1) The proposed localization algorithm operates based on low-cost automotive specification sensors, such as ESC motion sensors, along with a low-cost GPS and an AVM camera. Therefore, the localization system is cost effective and reliable under tough driving conditions. Furthermore, even though the localization system uses only low-cost automotive sensors, we conclude that the proposed localization algorithm is appropriate to apply to an autonomous car based on our analysis of the accuracy and reliability of the localization system when applied to real autonomous driving.
2) The probabilistic noise modeling of the RSM features is performed using data from real driving conditions and applied to the measurement update of the Monte-Carlo localization. When the RSM features are detected from cameras, there are several noise sources, such as unexpected vehicle motion, false-positive RSM features detection, and RSMs occupied by unexpected objects. Because the RSM features occupied by objects can be extracted by using the range sensors (radars and LIDARs), we focus on a noise model of unexpected vehicle motions and false RSM detections. To construct the probabilistic noise model of the RSM feature detection more explicitly, we analyzed the unexpected motion noise and false-positive noise by using motion and RSM detection results from various real driving conditions. Based on the results of the analysis, we could design a probabilistic noise model of RSM features and apply it to the localization algorithm. From the experimental study using the various variance noise models, we conclude that the exact probabilistic noise model of the sensor measurement greatly affects the localization performance.
3) We applied an around-view monitoring (AVM) system to the localization algorithm. The original purpose of the AVM was parking assistance by providing drivers with a virtual bird's-eye view image of nearby surroundings. By applying the AVM image into the localization algorithm, we enhanced the accuracy and reliability of the localization. Because the down-looking cameras in the AVM system provide accurate and stable road surface images that are robust to vehicle motion and illumination changes, the RSM features in the AVM image are reliable to use for localization. The advantage of using the AVM system for precise localization was evaluated by experimental studies. The results show that, as long as a sufficient number of RSMs are detected in the AVM image, the localization algorithm successfully operates using only the AVM system.

The paper basically assumes that the RSM is well detected by using the simple matched filter. However, it is impossible to detect the RSM perfectly in the degraded surface condition. In order to compensate the problem, the authors plan to extend the localization algorithm by integrating other types of sensors such as radars, LIDARs, and stereo visions. The performance of the proposed algorithm is expected to be improved when it integrates additional sensor information into the measurement update. Probabilistic noise modeling of the additional sensors will be required when we apply the new types of sensors for the measurement updates of the localization system.

REFERENCES

[1] C. Urmson *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *J. Field Robot.*, vol. 25, no. 8, pp. 425–466, Aug. 2008.

[2] M. Montemerlo *et al.*, "Junior: The Stanford entry in the urban challenge," *J. Field Robot.*, vol. 25, no. 9, pp. 569–597, Sep. 2008.

[3] C. Urmson *et al.*, "A robust approach to high-speed navigation for unrehearsed desert terrain," *J. Field Robot.*, vol. 23, no. 8, pp. 467–508, Aug. 2006.

[4] S. Thrun *et al.*, "Stanley: The robot that won the DARPA grand challenge," *J. Field Robot.*, vol. 23, no. 9, pp. 661–692, Sep. 2006.

[5] K. Jo, K. Chu, and M. Sunwoo, "Interacting multiple model filter-based sensor fusion of GPS with in-vehicle sensors for real-time vehicle positioning," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 329–343, Mar. 2012.

[6] R. Toledo-Moreo, M. A. Zamora-Izquierdo, B. Ubeda-Minarro, and A. F. Gomez-Skarmeta, "High-integrity IMM-EKF-based road vehicle navigation with low-cost GPS/SBAS/INS," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 491–511, Sep. 2007.

[7] K. Jo, K. Chu, K. Lee, and M. Sunwoo, "Integration of multiple vehicle models with an IMM filter for vehicle localization," in *Proc. IEEE Intell. Veh. Symp.*, La Jolla, CA, USA, 2010, pp. 746–751.

[8] S. Rezaei and R. Sengupta, "Kalman filter-based integration of DGPS and vehicle sensors for localization," *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 6, pp. 1080–1088, Nov. 2007.

[9] J. Huang and H. S. Tan, "A low-order DGPS-based vehicle positioning system under urban environment," *IEEE/ASME Trans. Mechatronics*, vol. 11, no. 5, pp. 567–575, Oct. 2006.

[10] Y. Cui and S. S. Ge, "Autonomous vehicle positioning with GPS in urban canyon environments," *IEEE Trans. Robot. Autom.*, vol. 19, no. 1, pp. 15–25, Feb. 2003.

[11] K. Jo, K. Chu, and M. Sunwoo, "GPS-bias correction for precise localization of autonomous vehicles," in *Proc. IEEE IV Symp.*, 2013, pp. 636–641.

[12] N. Mattern and G. Wanielik, "Camera-based vehicle localization at intersections using detailed digital maps," in *Proc. IEEE/ION PLANS*, 2010, pp. 1100–1107.

[13] N. Mattern, R. Schubert, and G. Wanielik, "High-accurate vehicle localization using digital maps and coherency images," in *Proc. IEEE Intell. Veh. Symp.*, La Jolla, CA, USA, 2010, pp. 462–469.

[14] T. Weiss, N. Kaempchen, and K. Dietmayer, "Precise ego-localization in urban areas using laserscanner and high accuracy feature maps," in *Proc. IEEE Intell. Veh. Symp.*, 2005, pp. 284–289.

[15] R. Matthaei, G. Bagschik, and M. Maurer, "Map-relative localization in lane-level maps for ADAS and autonomous driving," in *Proc. 25th IEEE IV Symp.*, Dearborn, MI, USA, 2014, pp. 49–55.

[16] O. Pink, "Visual map matching and localization using a global feature map," in *Proc. IEEE CVPRW*, 2008, pp. 1–7.

[17] D. Gruyer, R. Belaroussi, and M. Revilloud, "Map-aided localization with lateral perception," in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 674–680.

[18] Z. Tao, P. Bonnifait, V. Fremont, and J. Ibanez-Guzman, "Lane marking aided vehicle localization," in *Proc. 16th IEEE ITSC*, 2013, pp. 1509–1515.

[19] Z. Tao, P. Bonnifait, V. Fremont, and J. Ibanez-Guzman, "Mapping and localization using GPS, lane markings and proprioceptive sensors," in *Proc. IEEE/RSJ Int. Conf. IROS*, 2013, pp. 406–412.

[20] N. Suganuma and T. Uozumi, "Precise position estimation of autonomous vehicle based on map-matching," in *Proc. IV Symp.*, 2011, pp. 296–301.

[21] W. Tao and A. Ranganathan, "Vehicle localization using road markings," in *Proc. IEEE IV Symp.*, 2013, pp. 1185–1190.

[22] A. Ranganathan, D. Ilstrup, and W. Tao, "Light-weight localization for vehicles using road markings," in *Proc. IEEE/RSJ Int. Conf. IROS*, 2013, pp. 921–927.

[23] Y. Yu, H. Zhao, F. Davoine, J. Cui, and H. Zha, "Monocular visual localization using road structural features," in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 693–699.

[24] H. Lategahn and C. Stiller, "Vision-only localization," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 3, pp. 1246–1257, Jun. 2014.

[25] M. Schreiber, C. Knoppel, and U. Franke, "LaneLoc: Lane marking based localization using highly accurate maps," in *Proc. IEEE IV Symp.*, 2013, pp. 449–454.

[26] J. Ziegler *et al.*, "Video based localization for Bertha," in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 1231–1238.

[27] P. Bender, J. Ziegler, and C. Stiller, "Lanelets: Efficient map representation for autonomous driving," in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 420–425.

[28] J. Levinson and S. Thrun, "Robust vehicle localization in urban environments using probabilistic maps," in *Proc. IEEE ICRA*, 2010, pp. 4372–4378.

[29] A. Y. Hata, F. S. Osorio, and D. F. Wolf, "Robust curb detection and vehicle localization in urban environments," in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 1257–1262.

[30] H. Deusch *et al.*, "Multi-sensor self-localization based on maximally stable extremal regions," in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 555–560.

[31] J. Wiest *et al.*, "Localization based on region descriptors in grid maps," in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 793–799.

[32] A. Schindler, "Vehicle self-localization with high-precision digital maps," in *Proc. IEEE IV Symp.*, 2013, pp. 141–146.

[33] M. Lundgren, E. Stenborg, L. Svensson, and L. Hammarstrand, "Vehicle self-localization using off-the-shelf sensors and a detailed map," in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 522–528.

[34] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA, USA: MIT Press, 2005.

[35] T. Veit, J. P. Tarel, P. Nicolle, and P. Charbonnier, "Evaluation of road marking feature extraction," in *Proc. IEEE 11th ITSC*, 2008, pp. 174–181.

[36] M. Hentschel and B. Wagner, "Autonomous robot navigation based on OpenStreetMap geodata," in *Proc. IEEE 13th ITSC*, 2010, pp. 1645–1650.

[37] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, Aug. 1995.

[38] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 25, no. 7, pp. 53–82, Jul. 2010.

[39] ACELab, 2014 Autonomous vehicle A1 from ACE Lab, Korea, 2014. [Online]. Available: http://www.youtube.com/watch?v=2Bj0e1BqC00

Kichun Jo (S'10–M'14) received the B.S. degree in mechanical engineering and the Ph.D. degree in automotive engineering from Hanyang University, Seoul, Korea, in 2008 and 2014, respectively. He is currently with the Automotive Control and Electronics Laboratory (ACE Lab), Department of Automotive Engineering, Hanyang University. His main fields of interest are autonomous driving system, information fusion theories, distributed control systems, real-time embedded systems, and in-vehicle networks. His current research activities include system design and implementation of autonomous cars.

Yongwoo Jo received the B.S. degree in electrical engineering from Pusan University, Pusan, Korea, in 2012 and the M.S. degree in electrical engineering from Pohang University of Science and Technology, Pohang, Korea, in 2014. He is currently working toward the Ph.D. degree with the Automotive Control and Electronics Laboratory (ACE Lab), Department of Automotive Engineering, Hanyang University, Seoul, Korea. His main fields of interest are autonomous driving system, localization, and perception. His current research activities include vehicle localization for an autonomous car.

Jae Kyu Suhr (M'12) received the B.S. degree in electronic engineering from Inha University, Incheon, Korea, in 2005 and the M.S. and Ph.D. degrees in electrical and electronic engineering from Yonsei University, Seoul, Korea, in 2007 and 2011, respectively. He is currently a Research Assistant Professor with the Research Institute of Automotive Control and Electronics, Hanyang University, Seoul. His current research interests include computer vision, image analysis, and pattern recognition for intelligent vehicle and visual surveillance.

Ho Gi Jung (M'05–SM'10) received the B.E., M.E., and Ph.D. degrees in electronic engineering from Yonsei University, Seoul, Korea, in 1995, 1997, and 2008, respectively. He was with MANDO Corporation Global R&D H.Q. from 1997 to April 2009. He developed environmental recognition systems for intelligent parking assist system, collision warning and avoidance system, and active pedestrian protection system. From May 2009 to February 2011, he was with Yonsei University as a Full-Time Researcher and Research Professor. He researched computer vision applications for intelligent surveillance systems and biometric systems. Since March 2011, he has been with Hanyang University, Seoul, as an Assistant Professor. He is researching recognition systems for intelligent vehicles. His interests are recognition system for intelligent vehicles, next-generation vehicles, computer vision applications, and pattern recognition applications.

Myoungho Sunwoo (M'81) received the B.S. degree in electrical engineering from Hanyang University, Seoul, Korea, in 1979, the M.S. degree in electrical engineering from the University of Texas at Austin, TX, USA, in 1983, and the Ph.D. degree in system engineering from Oakland University, Rochester, MI, USA, in 1990.

He joined General Motors Research (GMR) Laboratories, Warren, MI, in 1985 and has worked in the area of automotive electronics and control for 28 years. During his nine-year tenure at GMR, he worked on the design and development of various electronic control systems for powertrains and chassis. Since 1993, he has led research activities as a Professor with the Department of Automotive Engineering, Hanyang University. His work has focused on automotive electronics and controls (such as modeling and control of internal combustion engines, design of automotive distributed real-time control systems, intelligent autonomous vehicles, and automotive education programs).