IMAGE LICENSED BY INGRAM PUBLISHING

Multiscale Site Matching for Vision-Only Self-Localization of Intelligent Vehicles

Yicheng Li and Zhaozheng Hu
*ITS Research Center, Wuhan University of Technology,
Wuhan 430063, China.
E-mail: ycli@whut.edu.cn; zzhu@whut.edu.cn*

Zhixiong Li
*School of Mechatronic Engineering, China University
of Mining and Technology, Xuzhou 221000, China;
he is also with Department of Mechanical Engineering,
Iowa State University,
Ames 50010, USA, E-mail: zhixiong.li@ieee.org*

Miguel Angel Sotelo
*Department of Computer Engineering. University of Alcalá,
Alcalá de Henares (Madrid), Spain.
E-mail: miguel.sotelo@uah.es*

Yulin Ma
*National Center of ITS Engineering and Technology,
Research Institute of Highway, Ministry of Transport,
Beijing, 100088 China, E-mail: myl@itsc.cn*

*Digital Object Identifier 10.1109/MITS.2018.2842024
Date of publication: 7 June 2018*

Abstract—Self-localization is a challenging issue in intelligent vehicle (IV) systems. Traditional self-localization methods, such as the Global Navigation Satellite System (GNSS), Inertial Navigation System (INS) and vision simultaneous localization and mapping (vSLAM), are subject to low accuracy, high cost or low robustness. To this end, this paper proposes a new multi-scale site matching localization (MS2ML) method for IV systems by using one single monocular camera. The MS2ML consists of a coarse localization, an image-level localization and a metric localization. In coarse localization, the proposed MS2ML calls the Bayesian vision-motion topological localization to obtain a set of nodes from a visual map. Furthermore, the holistic feature is generated for each query image, and hence, the holistic feature matching is implemented to realize image-level localization. A node is then selected from the candidate nodes. In metric localization, the closest node and vehicle pose are calculated through matching local features with three-dimension (3D) data. In order to evaluate the proposed MS2ML, real-world driving tests have been carried out in three different routes, two of which are from an urban roadway and an industrial park in Wuhan, China and the third one is from public KITTI (Karlsruhe Institute of Technology and Toyota Technology Institute) data set. The total lengths of these routes are more than 7 km. The experiment results demonstrate that the average localization errors of the proposed MS2ML method are less than 0.45 frame and the pose errors are less than 0.59 m. As a result, the proposed method remains high accuracy and great robustness in various environments.

I. Introduction

A. Background

The intelligent vehicles (IVs) have achieved considerable progress in the past years. In the field of this mass market, IVs attract more and more attentions all over the world. IVs self-localization, which enables the accomplishment of advanced driver assistance system (ADAS), is a challenging issue in autonomous driving research field [1].

Vehicle localization is often addressed using Global Navigation Satellite System (GNSS), Inertial Navigation System (INS) or both. However, the problems for GNSS mainly lie in low localization accuracy (the error can be over 10 m) and blind area, which can be especially disadvantageous when it is in urban environment [2]. Due to these drawbacks, GNSS is not qualified for intelligent vehicle localization requirement. Compared with using GNSS only, the combination of GNSS and INS improves the localization accuracy significantly [3]. However, the high cost of INS poses a hindrance for installing it on normal vehicles and also INS is quite restricted by blind areas. Other than GNSS and INS, laser scanner can also be employed as intelligent vehicle localization sensor, which can enhance the accuracy to a few centimeters by collecting three-dimensional (3D) Light Detection and Ranging (LIDAR) points. Unfortunately, despite the dramatic drop in price for these sensors, they are still too expensive. Based on the issues discussed above, a low-cost and high-accuracy algorithm is in demand for intelligent vehicle localization. Due to the low cost and easy installation, increasing number of normal vehicles start to use on-vehicle cameras. Therefore, various visual localization methods have been proposed in recent years. Visual localization is based on a visual map, in which every data collection node includes road scene [4]–[6]. As for each node, it contains image features, 3D information and trajectory. When the intelligent vehicle travels somewhere, the goal for vehicle localization is to find the node in the map which is closest to the vehicle's position. The advantage of using visual localization is that it can be independent of GNSS to make localization for intelligent vehicles. Plus, cameras cost less than laser scanner and INS. Therefore, the application of on-vehicle camera is more practical for intelligent vehicle localization.

B. Literature Review

Visual localization for IVs is related to simultaneous localization and mapping (SLAM). SLAM solves the computational problem of constructing or updating a map of an unknown environment while simultaneously keeping track of an agent's location within it. Further details can refer to [7]–[9] and [10]. However, the calculation process of SLAM is quite complicated. It is normally applied in small robots building or indoor vehicles manufacturing. It is normally applied in small robots or vehicles in indoor environments localization. Moreover, due to the unpredictable factors of outdoor environment, robots have to move slowly and also make the wrong way. Therefore, this method is not suitable for driving in reality.

According to the literature, some studies add some other on-vehicle sensors to enhance visual localization accuracy, such as GNSS, INS, speedometer, etc. Generally speaking, GNSS and INS are used together in such a way that they can provide a set of possible positions for visual localization. For example, in [11], a localization system is developed by integrating of GNSS, INS and camera. In this system, GNSS data and INS data are used to provide a set of possible positions and another position set is provided by image feature matching. After these, Bayesian filter integrate these two sets together. This system can achieve the localization accuracy to meter-level. Moreover, Li et al. [12] also use Global Positioning System (GPS, one kind of GNSS) data to determine a possible position range. Then, in image-level localization, lateral localization is provided by vision-based lane detection. Longitudinal localization is provided by vision-based traffic sign detection. Simulation-based experiments show that the accuracy of longitudinal and lateral localization are 0.51 and 0.09 m, respectively. In [13], the authors also first use GPS data to match with digital map. Then they catch images to detect lanes, traffic signs and match them with the map. This localization accuracy achieves sub-meter level. Similarly, Gu et al. [14] achieved vehicle localization in urban areas through combining of GNSS data, image, LIDAR and 3D map. Both lateral positioning error and speed error are evaluated in this study. Their research also enhanced the localization accuracy to lane-level. However, the researches that used additional sensors as discussed above are all restricted to different limitations, such as the high cost of laser scanner and INS or the blind area for GNSS and INS. All these limitations posed a barrier in the spread of intelligent vehicle. In addition, to improve the localization accuracy, some studies also applied speedometers to assist with intelligent vehicle localization.

We propose a multi-scale site matching localization method (MS2ML) by only using a monocular camera. This method does not require sensors like GPS receivers, INSs or any other additional measure instruments.

Speedometers can collect the speed data of vehicle and they compute the travelling distance of vehicle which is a strong constraint for vehicle localization. For example, Hojun et al [15] use in-vehicle sensors to collect vehicle speed and yaw rate. Both of these two data play an important role in vehicle position computation. Then this research is followed by the proposition of an extended Kalman filter method which integrates GPS data, speed, yaw rate and image. This method improves the localization accuracy to sub-meter level. Similarly, Gu et al. [16] mix 3D-GNSS with Inertial Measurement Unit (IMU) and also together with speedometer and this new sensor to enhance the localization accuracy. As discussed above, speedometers improved localization accuracy significantly and simplify the localization process drastically. However, speedometer also has its limitations. Although a speedometer serves as a strong constraint in enhancing the localization accuracy, yet the accuracy of measurement decreases as vehicle speed increases, which are due to factors such as synchronization, timing and registration inaccuracies. Hence, other localization methods that only using vision should be proposed.

In order to get high-level localization accuracy without other sensors, some start-of-the-art visual localization methods are proposed. For example, Wang et al. [17] propose a coarse-to-fine method which divides localization into two steps. Firstly, several candidates are selected in coarse localization. Then, localization result is obtained in fine localization. The use of coarse localization is to replace the use of GNSS. Similarly, Son et al. keep the coarse-to-fine method in their mind and propose a key frame selection method to reduce the matching complexity [18]. They divide the visual map into key frames and non-key frames by checking the number of feature points. In localization process, query image first matches with key frames and finds the closest key frame in the visual map. Next, a set of possible non-key frames which are close to the key frames are selected and then the closest one from visual map is selected. This method reduces the complexity of matching and also enjoys higher accuracy than the one only using image matching. In [19], an autonomous driving system is set up. They utilize global position detection based on visual features for their localization system. The image features are registered with 3D data. The vehicle pose is computed by a 6D rigid-body transformation. Furthermore, Brubaker et al. [20] set up a probabilistic model and utilize two video cameras and road maps to realize localization. Sefati et al. [21] propose a self-localization method which utilizes semantic and distinctive objects. The method is realized via laser scanner or stereo camera or both of them. The localization accuracy is 0.5 m by using stereo camera.

Moreover, topological model is also used to enhance the localization accuracy. Badino et al. [5], [22] set up a vehicle localization system and keep the topological model in mind. One previous localization result is used to predict the current position. In this study, Bayesian formula is used to convert the prediction into probability computation. In addition, Lategahn et al. [4] proposed a two-step approach. First of all, they build the topological localization model and matched the query image feature with visual map. Second, a dynamic programming procedure is used to find the node that is closest to the query image. The computation is very complicated as it demands 30 previous localization results. Similarly, we set up a topological model to enhance the localization accuracy [23]. This study also follows a two-step approach. First, one previous localization results is used to set up a topological model and then this model selects a set of possible positions from visual map. GNSS can be replaced by using this model. In the second step, both holistic feature matching and local feature matching are combined, which outputs the computation result of the closest data collection node. However, the topological model used here is simple and it has limitations in some road scenes. Further studies are still needed. Moreover, some studies set up topological model in visual map creation. For example, Patrick et al. [24] build hybrid metric-topological maps to make localization. Knolige et al. [25] present a 2D map based on sub-maps. These sub-maps include occupancy grid maps and a topological graph. This topological map facilitates to generate near-optimal plans for localization. As discussed above, topological model is an effective model to improve the localization accuracy. Hence, this paper engages further study on this model and improves the method.

C. Contributions

Herein, we propose a multi-scale site matching localization method (MS2ML) by only using a monocular camera. This method does not require sensors like GPS receivers, INSs or any other additional measure instruments. A previously created visual map is applied as the foundation of this method. On top of this, there are three steps operated to complete vehicle self-localization. First of all, we set up a topological model to compute a set of possible data collection nodes from the visual map. Thereafter, we generate holistic feature and match the feature with a set of possible nodes where one node is selected in image-level localization. Last but not least, 3D data from this node is matched with the local features of query image. As a result, vehicle pose and the node closest to query image are calculated. Contributions of this paper can be summarized as follows:

The visual map is composed of a set of data collection nodes. Each node is collected with a constant distance which is 2 m. For each node, a binocular camera collects pairs of images. To ensure the map accuracy, DGPS (differential GPS) data are collected in each node.

1) A novel localization prediction model called vision-motion model is built. In this model, we propose vision-speed and vision-acceleration to replace the actual speed and acceleration. The vision-speed is not the real speed and it is calculated by travelling nodes from a visual map in a short time. Similarly, vision-acceleration means changes of vision-speed in a short time. Compared with actual speed and actual acceleration, these two parameters are measured by images and they resolve the synchronization problems and registration inaccuracy issues between camera and speedometer. Vision-motion model is used to help vehicle make localization prediction.
2) To compute the vision-motion model, we propose Bayesian vision-motion topological localization method. In this method, we fuse Bayesian model and vision-motion model to convert localization prediction into probability computation. The use of this method is to replace the use of GNSS and then a set of candidates are selected. There are no restrictions such as GNSS blind area problem. This method simplifies the localization prediction procedures and also promotes the accuracy.
3) To further improve the localization accuracy, a novel result refining approach is proposed. In metric localization, we match 3D data with local features and compute the vehicle pose. The pose is utilized to refine the closest node. This extra step enables the further improvement of localization accuracy.

II. Visual Map Creation

The visual map is composed of a set of data collection nodes. Each node is collected with a constant distance which is 2 m. For each node, a binocular camera collects pairs of images. To ensure the map accuracy, DGPS (differential GPS) data are collected in each node. Please note that we only adopt DGPS data for visual map creation.

As the images are too huge to store, features are used to replace images. Actually, the created visual map only contains the following components: (1) Image features. Holistic features and local features are both extracted from left images. (2) 3D data. 3D data are computed from binocular images. The data are corresponding to local features. (3) Trajectory. Trajectory represents the positional relationship between two adjacent nodes. DGPS data are used to correct the trajectory. The created visual map is shown in Fig. 1.

III. The MS2ML Method

The work presented in this paper follows a three-step approach. In coarse localization, we propose a novel method called Bayesian vision-motion topological method to compute a set of possible nodes from visual map. The node closest to query image is selected in this candidate set. The second step is image-level localization. Holistic feature is extracted and matched with candidates in this step. One node is selected in the candidate set. Then local features from the query image are matched with 3D data from the node in metric localization. In this step, we refine the closest node and then get the vehicle pose in visual map. The illustration of our method is shown in Fig. 2.

A. Coarse Localization by Bayesian Vision-Motion Topological Localization

The goal of visual localization is to find the node in the visual map which is closest to the query image. However, simply taking feature matching from the huge data source of visual map is error prone and susceptible to visual aliasing and ambiguities. Some studies use GNSS to solve this problem. Herein, we propose a novel visual localization method instead of GNSS. In [23], we used topological model to predict the localization. In this paper, we make further study and propose Bayesian vision-motion topological localization method for localization prediction.

1) Vision-Motion Model

Vehicle speed can be used as a strong constraint for vehicle localization. For example, Hojun et al. [15] used vehicle speed as constraint to predict the vehicle position. The use of vehicle speed can improve the localization accuracy dramatically. However, it is hard to get the accurate real-time speed from ordinary vehicle. Even though some refitted vehicles can collect their speed by wheel odometer or speedometer, the accuracy of the measurement is low due to synchronization or registration inaccuracies between

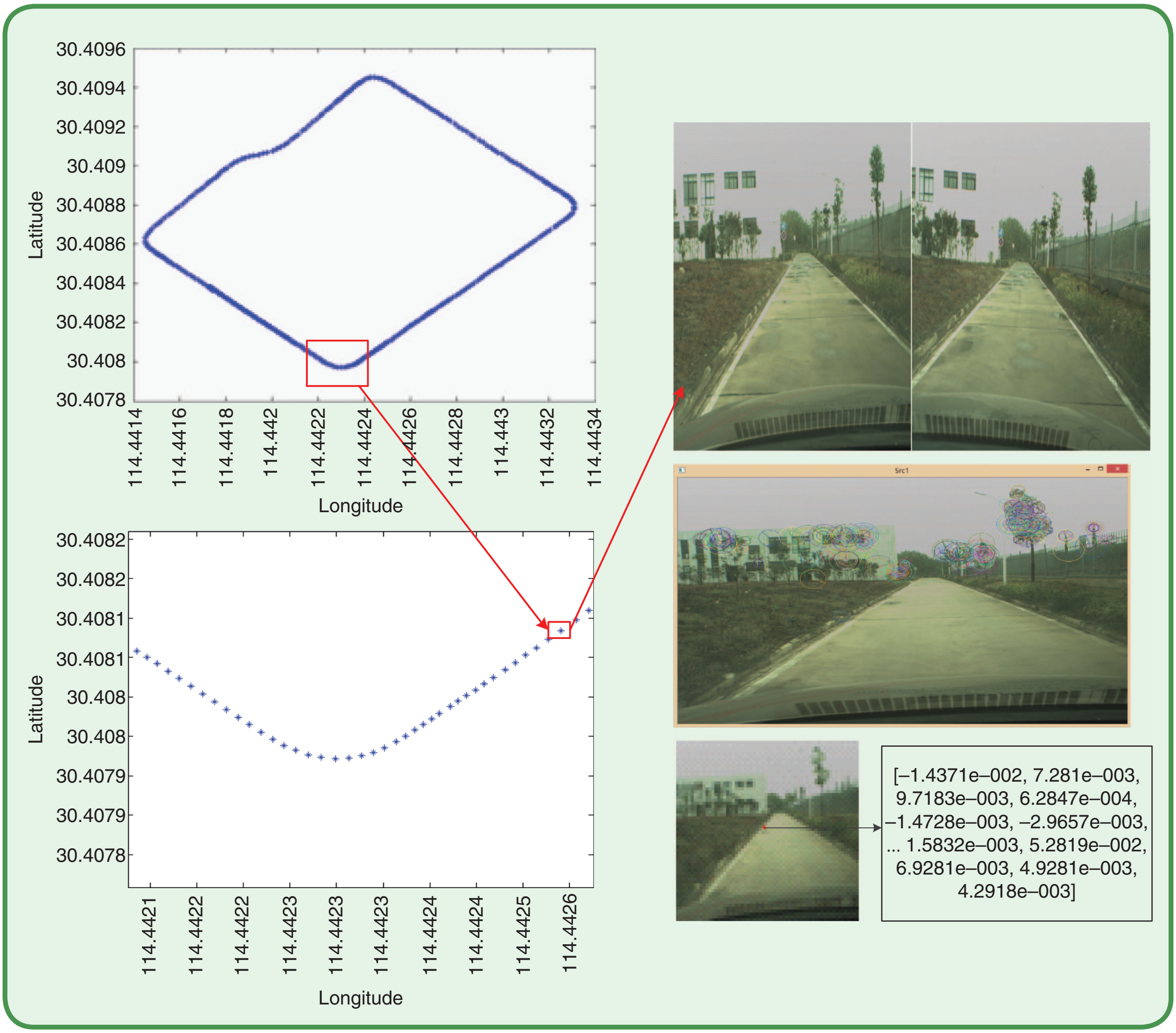


FIG 1 Illustration of visual map.

cameras and these sensors. Herein, we propose a vision-motion model which includes vision-speed and vision-acceleration instead of vehicle speed and acceleration. Vision-speed and vision-acceleration are measured by vision information, which reflects the vehicle speed and acceleration in visual map. From this model, we assume that vehicles do uniform motion or uniformly accelerated motion in a very short time. Vision-speed is not the real speed of vehicle. It means the vehicle's traveling distance between two adjacent positions in the visual map. The unit of vision-speed is frame/s. Similarly, vision-acceleration means the vision-speed change between two adjacent positions in the visual map. The unit of vision-acceleration is frame/s^2. The advantage of the proposed model is that we can obtain such a strong constraint for localization in a simple way. Next, we illustrate the vision-motion model.

First of all, we assume that X_j is the localization result at time *j*. Then the vision-speed v_j at time j can be formulated as follow:

$$v_j = (X_j - X_{j-1})/\Delta t \tag{1}$$

where Δt is the time difference from j to $j-1$. Since we have got the vision-speed, it can be used instead of the vehicle speed as a constraint for localization prediction. To make localization prediction, we add topological localization model. Since we have got the previous localization results, the current localization result is obtained topologically. When the vehicle wants to localize itself at time i, the result can be predicted by the previous localization result as follows:

$$X_i = X_{i-1} + v_i \cdot \Delta t = X_{i-1} + d_i \tag{2}$$

where $d_i = X_i - X_{i-1}$, it denotes the frame difference from $i-1$ to i. From this formula, we can predict the localization by frame difference evaluation. However, topological localization suffers from low prediction accuracy due to vehicle speed variations. We need to compute the vision-acceleration to solve this problem. Similarly, the vision-acceleration a_j at time j can be formulated as follow:

$$a_j = (v_j - v_{j-1})/\Delta t = (X_j - 2X_{j-1} + X_{j-2})/\Delta t^2 \qquad (3)$$

Similarly, we can use vision-acceleration to compute the current localization as follow:

$$X_i = X_{i-1} + v_{i-1}\cdot \Delta t + 0.5\cdot a_i \cdot \Delta t^2 = X_{i-1} + 0.5\cdot D_i \qquad (4)$$

where $D_i = X_i - X_{i-2}$. Note that the formula (1) and (3) are subject to time delay of $dt/2$ for vision-speed and dt for vision –acceleration. However, the vision-speed and vision-acceleration are computed in a very short time, so the time delay is very short, too. Furthermore, the vision-model is used to make qualitative localization. A localization range is computed in this step. Hence, this time delay has little effects for localization.

2) Bayesian Vision-Motion Topological Localization

Both vision-speed and vision-acceleration are used as constraints for topological localization. To enhance the prediction accuracy, we use the probability which is computed by Bayesian formula to make coarse localization. The illustration of Bayesian vision-motion topological localization is shown in Fig. 3. The thought of topological localization is that we use the previous result X_{i-1} to predict the current result X_i. The prediction is based on vision-speed and vision-acceleration. Therefore, the conditional probability for each possible position X_i^k at time t can be expressed as $P(X_i^k\,|\,vs,va)$, where vs means the vision-speed, va means vision-acceleration. Hence, we can use Bayesian formula to compute this conditional probability as follow:

$$P(X_i^k\,|\,vs,va) = \frac{P(X_i^k)\cdot P(vs,va\,|\,X_i^k)}{P(vs)\cdot P(va)} \qquad (5)$$

where we assume va and vs are independent events. We also assume that $P(X_i^k)$, $P(vs)$ and $P(va)$ are constant value in X_i^k.

Considering va and vs are independent events, the formula can be simplified as follows:

$$P(X_i^k\,|\,vs,va) \cong P(vs\,|\,X_i^k)\cdot P(va\,|\,X_i^k) \qquad (6)$$

where the symbol '≅' denotes that $P(X_i^k\,|\,vs,va)$ is proportional to $P(vs\,|\,X_i^k)$, $P(va\,|\,X_i^k)$. Furthermore, considering that $P(vs\,|\,X_i^k) = P(X_i^k\,|\,vs)\cdot P(vs)/P(X_i^k)$ and $P(va\,|\,X_i^k) = P(X_i^k\,|\,va)\cdot P(va)/P(X_i^k)$, we can simplify the equation as follow:

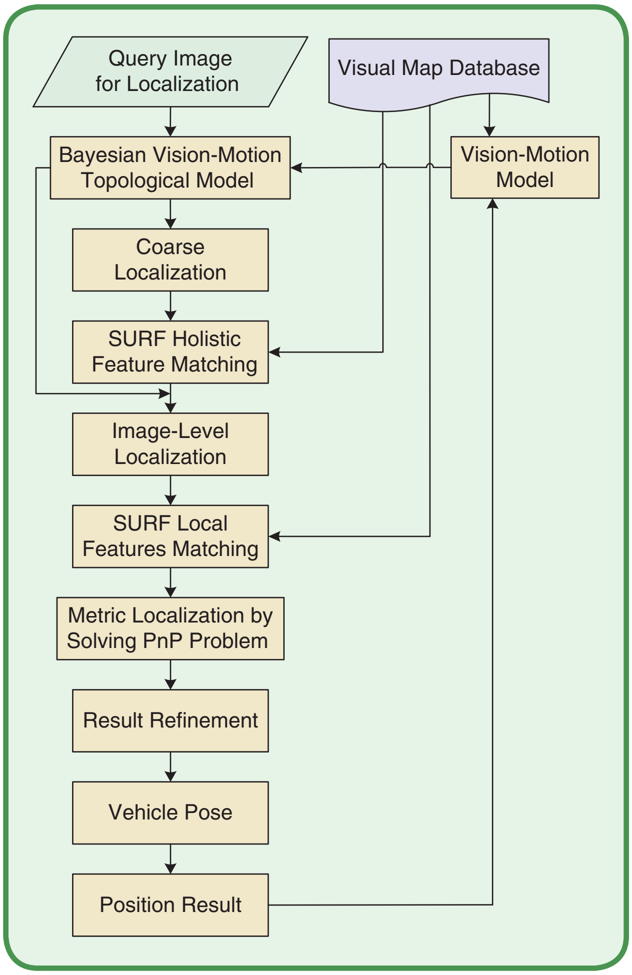

Query Image for Localization
Visual Map Database
Bayesian Vision-Motion Topological Model
Vision-Motion Model
Coarse Localization
SURF Holistic Feature Matching
Image-Level Localization
SURF Local Features Matching
Metric Localization by Solving PnP Problem
Result Refinement
Vehicle Pose
Position Result

FIG 2 The proposed methodology for multi-scale site matching self-localization.

$$P(X_i^k\,|\,vs,va) \cong P(X_i^k\,|\,vs)\cdot P(X_i^k\,|\,va) \qquad (7)$$

Then, we use vision-speed model and vision-acceleration model to compute this formula. In vision-speed model, the formula for prediction localization is shown in Eq. (2). As the previous localization result X_{i-1} is known, the computation of $P(X_i^k\,|\,vs)$ is based on the computation of d_i. We assume the vision-distance d_i variation obeys Gaussian distribution in a short time. Then we can compute the conditional probability $P(X_i^k\,|\,vs)$ as follow:

$$P(X_i^k\,|\,vs) = \frac{1}{\sqrt{2\pi}\,\sigma_s}\exp\left(-\frac{(x-\mu_s)^2}{2\sigma_s^{\,2}}\right) \qquad (8)$$

where $\mu_s = \sum_{m=1}^{n} d_{i-m}/n$, $\sigma_s = \sum_{m=1}^{n}(\mu_s - d_{i-m})^2/n$. In this equation, we select n previous localization results to compute the conditional probability. It means that we compute the average of vision-speed at time $i-1$.

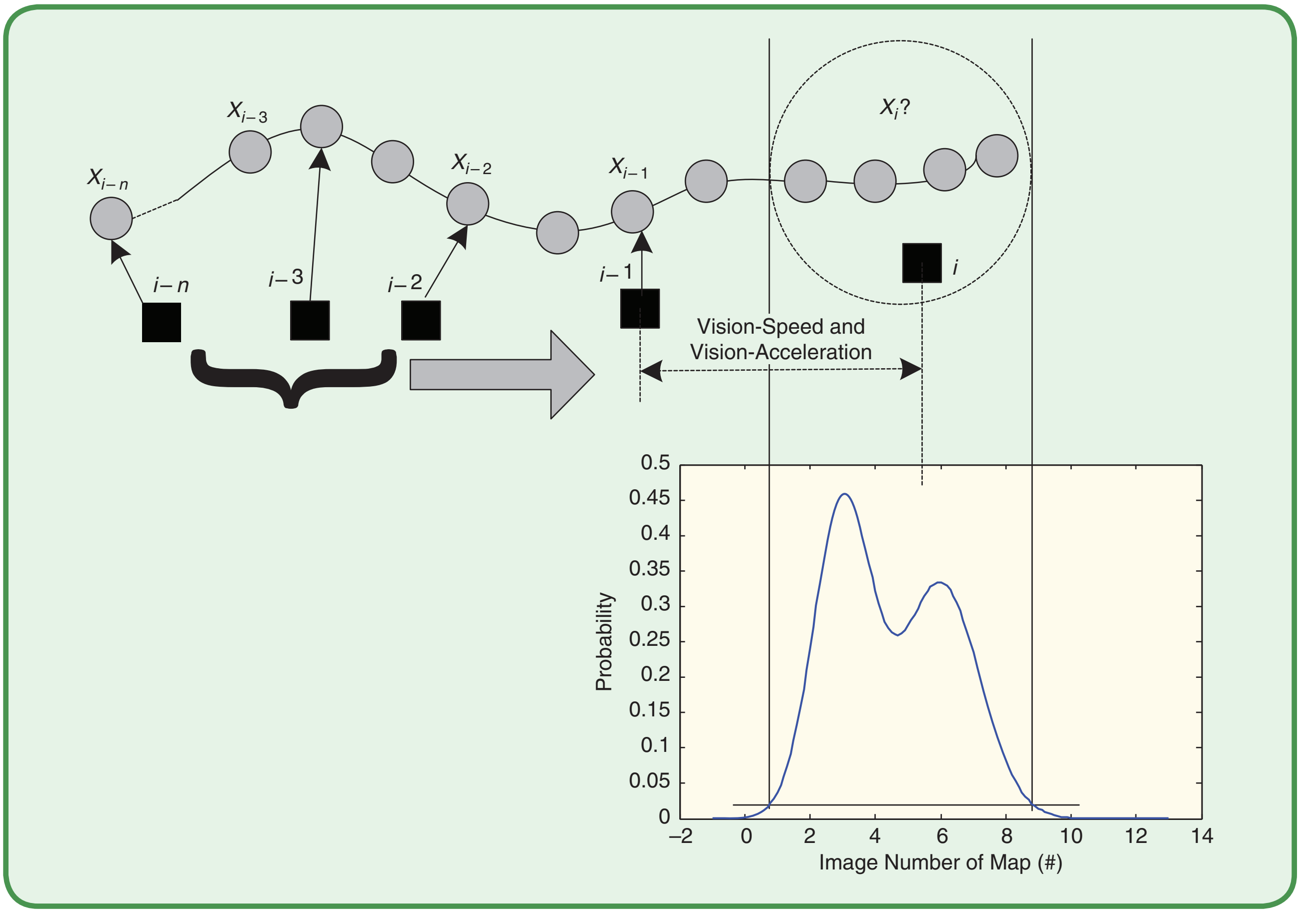


FIG 3 Illustration of Bayesian vision-motion topological localization. The black squares denote vehicle position, the white circles denote node of map. The candidate at the current position i is predicted from the previous localization result, $i-1 \backsim X_{i-1}$.

Actually, there are many solutions on how to choose the parameter n. And we choose $n = 15$ in this paper, as we have tested many times and a higher localization accuracy will be obtained when $n = 15$.

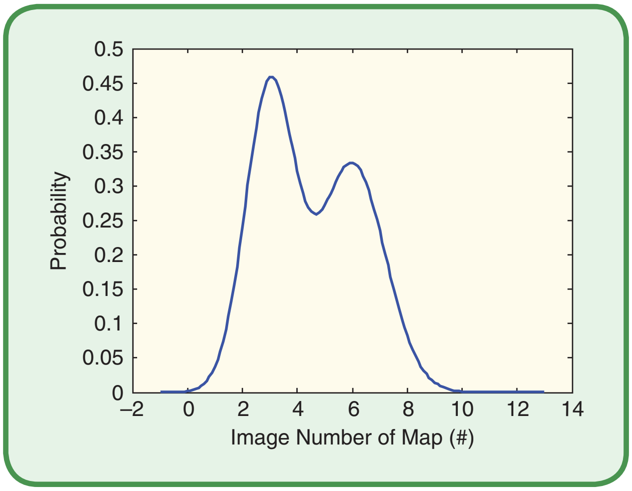


FIG 4 The mix Gaussian distribution generated by 2 Gaussian distributions.

We can also make the similar assumption to compute the conditional probability $P(X_i^k \mid va)$. Since the vision-distance variation obeys Gaussian distribution in a short time, the conditional probability $P(X_i^k \mid va)$ is computed as follow:

$$P(X_i^k \mid va) = \frac{1}{\sqrt{2\pi}\,\sigma_a} \exp\left(-\frac{(x-\mu_a)^2}{2{\sigma_a}^2}\right) \tag{9}$$

where $\mu_a = \sum_{m=1}^{n} D_{i-m}/n$, $\sigma_s = \sum_{k=1}^{n} (\mu_a - D_{i-k})^2/n$. We select n previous localization results to compute the conditional probability. We fuse these two Gaussian distributions to a new distribution which is shown in Fig. 4. This new distribution can be formulated by the following equation:

$$\begin{aligned} P(X_i^k \mid vs, va) &\cong P(X_i^k \mid vs) \cdot P(X_i^k \mid va) \\ &= N(\mu_s, \sigma_s^2) \cdot N(\mu_a, \sigma_a^2) \end{aligned} \tag{10}$$

To get the candidates, we set up a threshold σ for Eq. (10), when the probability exceeds σ, these indexes are output as candidates, denoted by $C = \{c_1, c_2 \ldots c_n\}$.

B. Image-level Localization by Holistic Feature Matching

In the image-level localization step, we extract holistic features from query images and then match them with candidates. By holistic feature we mean that the whole image is considered as a feature. The use of holistic features has higher matching speed and less data storage than the use of local features. In this section, we use SURF descriptor for feature extraction. Compared with Scale-invariant feature transform (SIFT), this method simplifies the computation complexity and has a similar result to SIFT. The matching accuracy is better than the use of ORB descriptor [23]. Further details of SURF descriptor can be found in [26].

By holistic feature we mean that the whole image is considered as a feature. The use of holistic features has higher matching speed and less data storage than he use of local features.

To extract holistic features automatically, all the query images are resized into a standard image with a resolution of 63 × 63 pixels. Then, the holistic features are computed. As a result, each holistic feature is represented with 1 × 64 vectors. Taking a query image as example, its holistic feature is denoted by f_k. Fig. 5 below shows a normalized 63 × 63 (pixel) detection image together with 1 × 64 vectors as holistic feature.

Finally, we match the holistic feature with the candidate set C computed in coarse localization step. It can be computed as follow:

$$D = \{d_j \mid d_j = \mathrm{Euc}(f, c_j)\} \tag{11}$$

where c_j is the jth SURF holistic feature descriptor of candidates; f is the descriptor of the query image. Euclidean distance is computed as follows:

$$\mathrm{Euc}(X, Y) = \sqrt{\sum_{i=1}^{64} (x_i - y_i)^2}, x \in X, y \in Y \tag{12}$$

where X and Y are two SURF holistic features; x_i is the ith vector of X, y_i is the ith vector of Y. To select one node from the candidates, we normalize each distance d_j as follows:

$$\bar{s}_j = \frac{1/d_j}{\sqrt{\sum_{j=1}^{n} 1/d_j^2/n}} \tag{13}$$

Similarly, we normalize each probability p_j which is computed by Eq (10) as follows:

$$\bar{p}_j = \frac{p_j}{\sqrt{\sum_{j=1}^{n} p_j^2/n}} \tag{14}$$

Finally, we add the two scores above for each candidate. The node with the highest score is selected to make metric localization. It can be computed in the following equation:

$$c_i = \arg\max(\bar{p}(c_j) + \bar{s}(c_j)) \tag{15}$$

where $c_i \in C$. Hence, c_i is the image-level localization result.

C. Metric Localization by Solving the Perspective N-Points (PnP) Problem

We obtain one node from the candidates in image-level localization step. In this section, the vehicle pose and the closest node is computed by metric localization. Our metric localization method follows 3 steps. First, local features of the query image are extracted. We match local features of the query image with local features of the node. Note that visual map also includes 3D data in each node; the 3D data corresponds to local features of the node. Hence, the local features of the query image and 3D data compose a PnP problem. Then, vehicle pose is computed by solving the PnP problem. Finally, result refinement is done and the node closest to the query image is obtained.

Similar to holistic feature extraction, we also use SURF descriptor for local feature extraction. We first resize the query image in standard image. The SURF descriptor is also represented with 1 × 64 vectors. Then, local features are used to match the query image with the node. Local features matching can be shown in Fig. 6. From this figure, we find the 3D points corresponding to the local features of the node. We use the series of corresponding points to compute the vehicle pose by solving PnP problem. The method can be formulated as follows:

$$\begin{bmatrix} u_1 & u_2 & \cdots & u_n \\ v_1 & v_2 & \cdots & v_n \\ 1 & 1 & & 1 \end{bmatrix} \cong K[R \;\; t] \begin{bmatrix} X_1 & X_2 & & X_n \\ Y_1 & Y_2 & \cdots & Y_n \\ Z_1 & Z_2 & & Z_n \\ 1 & 1 & & 1 \end{bmatrix} \tag{16}$$

where $[u_i \;\; v_i \;\; 1]^T$ is the ith local feature of the query image, its unit is pixel. $[X_i \;\; Y_i \;\; Z_i \;\; 1]^T$ is the ith 3D point of the

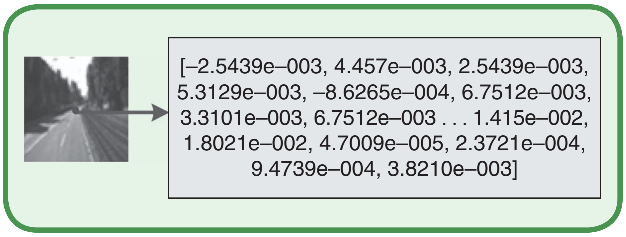

[–2.5439e–003, 4.457e–003, 2.5439e–003, 5.3129e–003, –8.6265e–004, 6.7512e–003, 3.3101e–003, 6.7512e–003 . . . 1.415e–002, 1.8021e–002, 4.7009e–005, 2.3721e–004, 9.4739e–004, 3.8210e–003]

FIG 5 Extraction of SURF holistic feature (1 × 64 Vectors).

node. K is the intrinsic parameter of the in-vehicle camera, which is a 3 × 3 matrix. The intrinsic parameter can be computed by camera calibration [27]. R is the rotation matrix which is a 3 × 3 matrix, while t is the translation vector which is a 3 × 1 vector; both R and t compose the pose of the query image. This formula depends on at least 4 corresponding points. Using R and t, we can compute the position of the vehicle in the visual map as follow:

$$P = -R^t \cdot t \tag{17}$$

The node computed in image-level localization step is sometimes not the closest one to the query image. We can refine the image-level localization result by using the vehicle position P. First of all, we compute the Euclidean distance S from the query image to the node. It can be formulated as follows:

$$S = \| P \| = \sqrt{p_x^2 + p_y^2 + p_z^2} \tag{18}$$

where $P = [p_x, p_y, p_z]^t$. We assume that the data collection frequency of visual map is q m/frame. If $S \leq q$, this node is considered to be the closest node to the query image. However, if $S > q$, the closest node is not this one. We need to adjust the result on the basis of that node. We assume that the change number is y, and y is the integer part of $(S - q)/q$. The change orientation is determined by P. The refinement can be shown in Fig. 7, we compute the distance from query image $i + n$ to candidate X_{k+3} by P. However, the distance shows that it is not the best result. Considering the collection frequency, X_{k+1} is selected as the closest node. In addition, if the distance S is more than 15 m, we treat this node as an outlier. Then, the refinement will not be done in this time. Moreover, if the number of corresponding points in local feature matching is less than a threshold ρ, we also treat this result as an outlier.

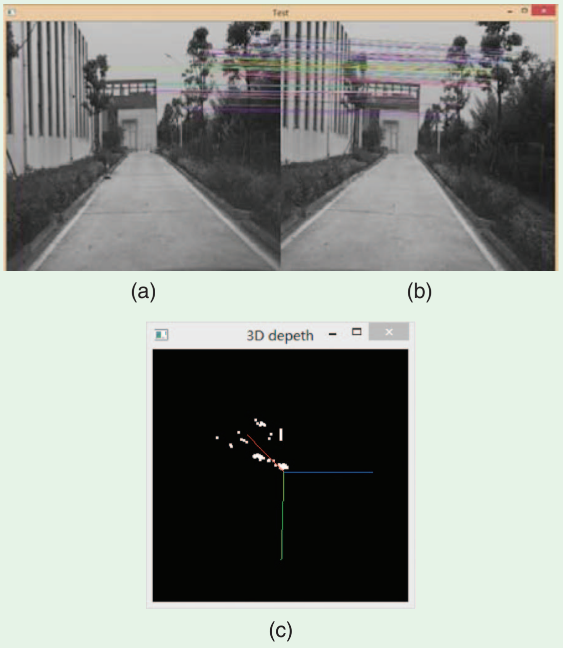


FIG 6 Local feature matching between query image and candidate: (a) the node got from image-level localization; (b) query image; (c) 3D depth of the node.

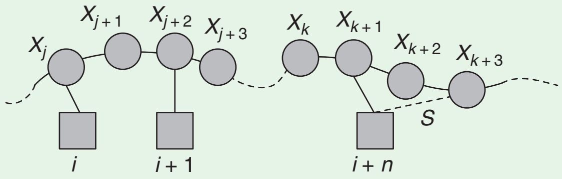


FIG 7 Illustration of result refinement. Squares denote query images at time $i, i+1 \ldots i+n$, respectively. All the circles X_x denote images of map. X_{k+3} is computed as the image-level localization result at time $i+n$. However, the distance computed from the query image to X_{k+3} is farther than data collection frequency of visual map, the closest is refined to X_{k+1}.

D. Outline of the Localization Algorithms

The algorithms for MS2ML can be summarized as follows:

1) We localize the first 15 positions manually. Next, the intelligent vehicle travels somewhere, the on-board monocular camera collects image of road scene. In each image, both holistic feature and local features are extracted by using SURF descriptor.
2) Bayesian vision-motion topological method is used to compute a set of candidates from visual map.
3) Each holistic feature is matched with candidates. Image-level localization is done by fusing feature matching and Bayesian vision-motion topological localization method. One node is selected from candidates.
4) Local features of query image are matched with 3D data of the node. Then vehicle pose is computed by solving a PnP problem. Finally, the node closest to query image is found by localization refinement.
5) Outliers are eliminated by checking the number of corresponding points in the local feature matching. When the number is less than 45, the node is an outlier.

IV. Experimental Results

Next, we present experiments on real-world data to assess and evaluate our method. There are three databases to evaluate our method. To set up the first two data sets, one standard intelligent vehicle was equipped with binocular camera, GPS receiver and INS. The camera was produced by Bumblebee, which was equipped forward in the vehicle. Fig. 8 shows the setup of the data collection system. The red circle shows the camera equipped in the vehicle. We picked two routes to set up data sets in Wuhan city, China. One route was in an industrial park which had a few vehicles. The other route was in an urban roadway. Traffic in this route was more complicated than in the industrial park, given that it has many more vehicles. In addition, one public data set was selected in KITTI (Karlsruhe Institute of Technology and Toyota Technology Institute). The total length of these routes is over 7 km.

A. Mapping Experiments

To localize the intelligent vehicle, we first create a visual map. DGPS (Differential Global Positioning System) station was set up in each route to enhance the accuracy of GPS data. INS was taken as additional sensor to correct position data. The accuracy of these position data is about 20 cm. These additional sensors were only used in map creation. The data collection frequency was about 2 m/node. In each node, high accuracy position data were collected by INS with DGPS. Pairwise images were collected by binocular camera and had a size of 1600 × 1200 (in pixel). Holistic image features were extracted by SURF descriptor. 3D data were computed by triangulation [28], [29].

B. Localization Experiments in Industrial Park

We introduce the performance of our method in industrial park. In vehicle localization, we only use monocular camera as localization sensor without any additional sensors. Considering that intelligent vehicles are usually tested in closed sections with a few vehicles, we also select one route with a few vehicles. The testing ground is shown in Fig. 9. For each query image, both holistic feature and local features were extracted by using SURF descriptor. The first 15 positions were located manually for prior information. The intelligent vehicle speed was in strict compliance with local regulations. We did not deliberately maintain the vehicle speed through the experiments.

The image-level localization results are shown in Fig. 10. The red line denotes the node closest to query image in the visual map. The blue line denotes the holistic feature matching results. From the figure, we can find that there are some outliers in image-level localization. Fortunately, the Bayesian vision-motion topological localization method is robustness against these outliers. There are no cumulative errors in the results. To eliminate these outliers, we use localization refinement. Then the number of local feature matching points is checked. If the number is less than 45, we treat it as an outlier. Although some inliers are eliminated, the localization accuracy is enhanced significantly. The localization results are shown in table 1. As discussed, our core task for vehicle localization is to find the node closest to query image. Hence, we mainly use image error to evaluate the method. Frame is the unit of image error. Furthermore, we compare our method with general GPS localization and the method in [18]. The reason we choose [18] as comparison method is that this is also a vision-only localization method and publishes in a high level journal. From the table, we can see that our method performs better than general GPS localization and the method in [18]. The mean error of our method is only 0.20 frame and the standard deviation is 0.42 frame. The rate of zero mean accuracy is 80.0%.

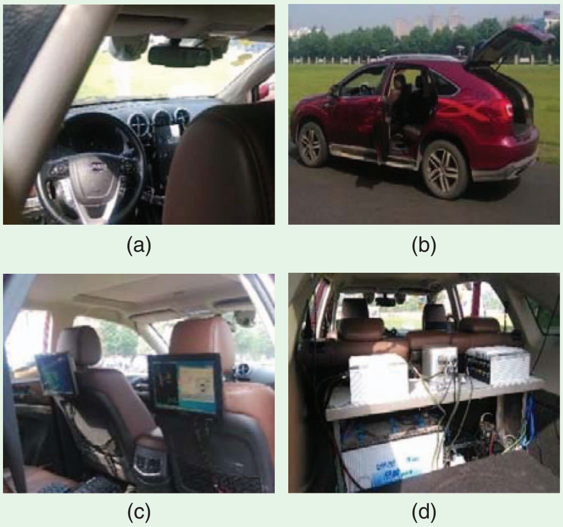


FIG 8 Setup of image collection system.

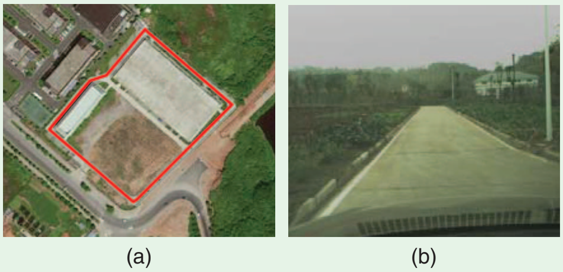


FIG 9 Testing ground: blue line is the urban roadway, red line is the industrial park.

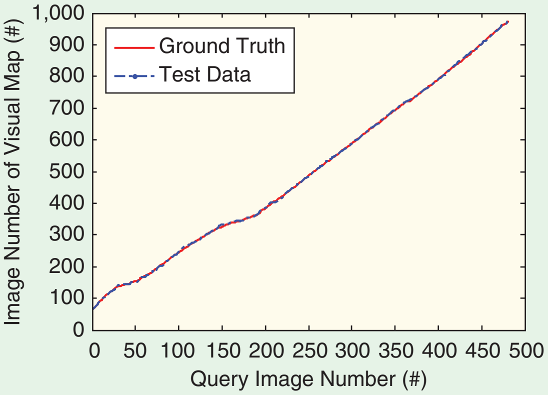


FIG 10 Image-level localization results in industrial park.

Table 1. Comparison of localization errors from the proposed method, general GPS localization, and the method in [18] in industrial park.

Methods	Mean (frame)	Std. Dev (frame)	Max error (frame)	Rate of Zero Frame Accuracy
Proposed method	0.20	0.42	1	80.0%
General GPS localization	1.81	2.29	4	41.3%
Method in [18]	2.75	3.04	10	44.2%

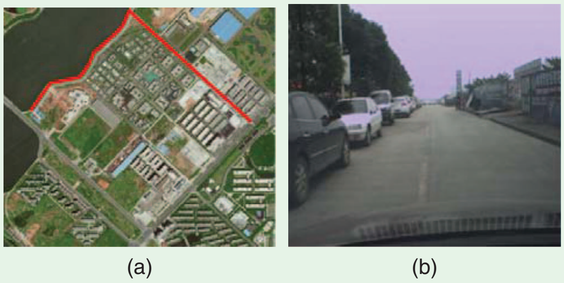


FIG 12 Testing ground: red line is the urban roadway.

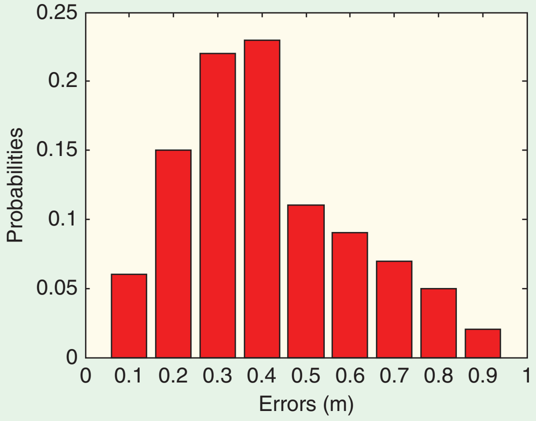


FIG 11 Pose errors for localization in industrial park.

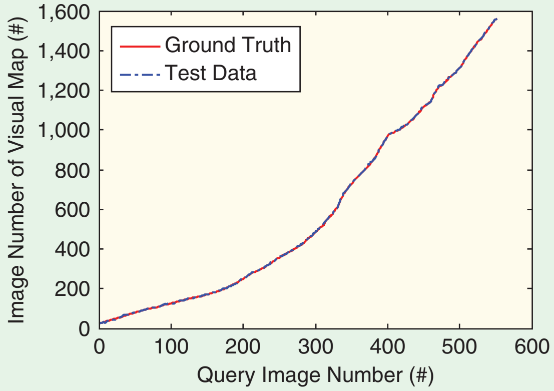


FIG 13 Image-level localization results in urban roadway.

Table 2. Comparison of pose errors from the proposed method, general GPS localization, and the method in [18] in industrial park.

Methods	Mean(m)	Std. Dev(m)	Max error(m)
Proposed method	0.41	0.19	0.92
General GPS localization	3.94	2.38	9.51
Method in [18]	4.62	5.11	13.6

The max error of the result is 1 frame. It means that even though there are 20% positions in which the closest node is not found, the result is only 1 frame from the ground truth. Hence, all the statistics show that the proposed method has high accuracy and great robustness.

Finally, we computed the vehicle pose in each position. The pose errors are shown in Fig. 11. From these results, we can find that all the errors of vehicle pose are less than 1 meter. Majority of the errors are distributed between 0.2 m and 0.4 m. The pose errors are also compared with general GPS localization and the method in [18]. The results are shown in table 2. From the table, we can find that the mean of pose errors is 0.41 m and the standard deviation is 0.19 m. The max error is 0.92 m. They mean that the proposed method also performs better than other methods. It is because if the node closest to query image can be found, the method will have a high accuracy for pose computation. Our method in industrial park can significantly enhance the localization accuracy.

All the experiments above were tested in industrial park. From Fig. 9, we can find that the slope of ground truth changes a little. It means that the vehicle speed is stable when we make localization. What would happen if the experiment was performed when the vehicle speed was changed? Next, we introduce the performance of our method in a different route.

C. Localization Experiments in Urban Roadway

We selected a route in urban roadway. The intelligent vehicle used in this route was the same as the vehicle tested in industrial park. The testing ground is shown in Fig. 12. It can be seen that there are some other vehicles on the road. These vehicles would affect the speed. How about the localization accuracy in this route?

The image-level localization results in urban roadway are shown in Fig. 13. From this figure, we can see that this route is longer than the route in industrial park and vehicle speed in this route also changes more. Fortunately, the Bayesian vision-motion topological localization method is robust to these speed variations. Similarly, there are some outliers in image-level localization in urban roadway. The outliers can be eliminated in metric localization. After localization refinement and local features number checking, the localization results are shown in table 3. To evaluate the proposed method, we also use the method in [18] and general GPS localization for comparison. From the results, our method performs better than general GPS localization and the method in [18]. The mean of localization errors is 0.45 frame and the standard deviation is 0.50 frame. The accuracy rate is 55.0%. Compared with the results in industrial park, the localization accuracy in urban roadway is lower than in industrial park. The reason is that traffic is more complicated and more vehicles on urban roadway. This question also appears in the other two methods. Fortunately, the test accuracy is still less than 0.5 frames in urban roadway; our method can also meet the requirement of the intelligent vehicle localization in this route.

Afterwards, the vehicle pose was likewise computed. Fig. 14 shows the pose errors in urban roadway. As the localization accuracy is lower than in industrial park, the pose errors in this route are also larger than the errors in industrial park. A few pose errors are even more than 1 meter. However, most of the errors are less than 1 meter. The majority of them distribute between 0.5 m and 0.8m. The mean error is about 0.59 m, which means that the accuracy also achieves sub-meter localization level. We also compare these results with general GPS method and the method in [18], which is shown in table 4. Although the accuracy in this route is lower than that in industrial park, the proposed method also performs better than general GPS method and the method in [18]. It shows that our method has high accuracy in these two different routes. Furthermore, the standard deviation of our method is only 0.22 m, which means our method has great robustness in different routes.

D. Localization Experiments in KITTI

The public KITTI data set is used to further validate the proposed method. The data set is a set of videos and continuous image frames containing various road scenes in Karlsruhe, Germany. There is a standard station wagon with two high-resolution color and grayscale video cameras, accurate ground truth is provided by GPS localization system and Velodyne laser scanner [30]. The station wagon is shown in Fig. 15. The testing ground is shown in Fig. 16.

The localization result is shown in table 5. As GPS localization system in this data set was accurate, which was not general GPS receiver; we did not use this localization results as comparison. From the results, we can find that our method performs better than the method in [18]. The mean error of the proposed method is 0.36 frames and the standard deviation is 0.48 frames. The localization accuracy is 64%. The test in this route performs better than in urban roadway and worse than in industrial park. The max error for these 3 routes is 1 frame. Hence, we can make a conclusion that our method has high accuracy and great robustness in both public data set and testing routes.

Table 3. Comparison of localization errors from the proposed method, general GPS localization, and the method in [18] in urban roadway.

Methods	Mean (frame)	Std. Dev(frame)	Max error (frame)	Rate of Zero Frame Accuracy
Proposed method	0.45	0.53	1	55.0%
General GPS localization	2.13	2.41	6	36.8%
Method in [18]	4.04	4.18	11	37.1%

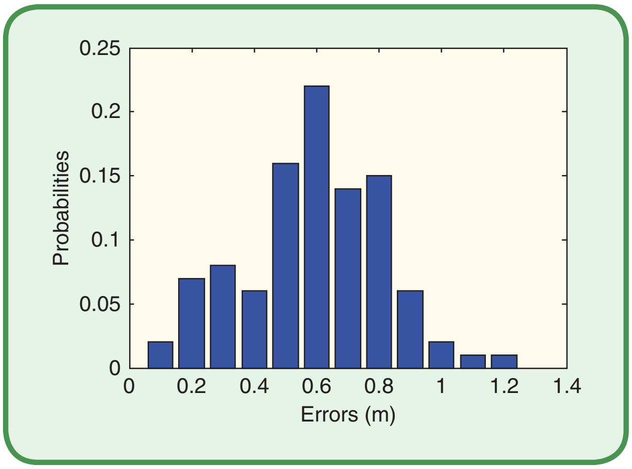


FIG 14 Pose errors for localization in urban roadway.

Table 4. Comparison of pose errors from the proposed method, general GPS localization, and the method in [18] in urban roadway.

Methods	Mean(m)	Std. Dev(m)	Max error(m)
Proposed method	0.59	0.22	1.21
General GPS localization	5.41	3.81	12.8
Method in [18]	7.35	5.89	17.3

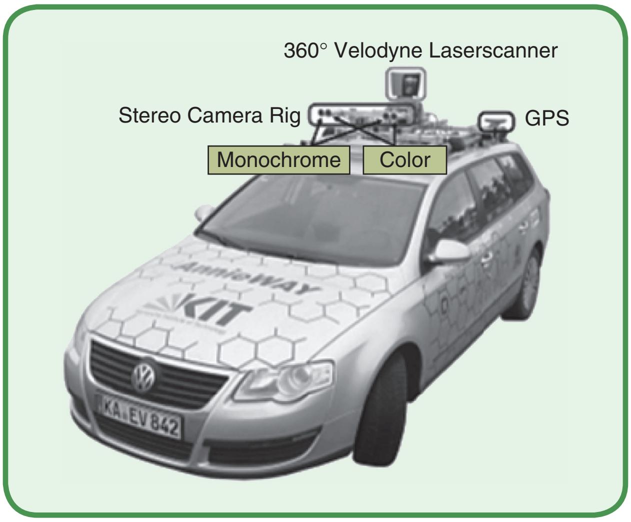


FIG 15 Station wagon in KITTI.

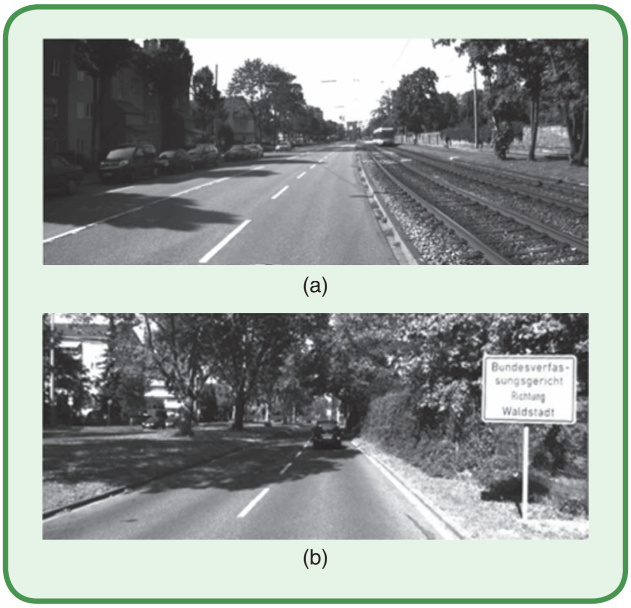


FIG 16 Testing ground in KITTI.

Table 5. Comparison of localization errors from the proposed method and the method in [18] in public data set.

Methods	Mean (frame)	Std. Dev (frame)	Max error (frame)	Rate of Zero Frame Accuracy
Proposed method	0.36	0.48	1	64.0%
Method in [18]	3.01	3.18	8	51.1%

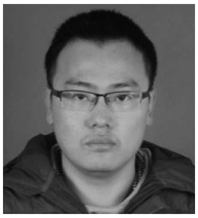
V. Conclusions

This study has proposed an MS2ML method for intelligent vehicles by only using vision and a low-cost monocular camera. This method consists of coarse localization, image-level localization and metric localization. In coarse localization, Bayesian vision-motion topological localization method enables collections of a set of candidates and the proposal of this method can replace GPS localization. In this method, vision-speed model and vision-acceleration model are proposed to enhance localization accuracy. In image-level localization, firstly holistic feature can be generated by using SURF descriptor and then one node from candidates is selected through the combined application of holistic feature matching and Bayesian vision-motion topological localization. In metric localization, result refinement is processed by solving a PnP problem. At last, we get the node that is closest to the query image and the vehicle pose. The proposed method has been proved to be valid on 2 testing routes in Wuhan China and another one in public data set. The total testing distance exceeds 7 km. Experiment results demonstrate that the average localization errors are less than 0.45 frame and the average pose errors are less than 0.59 m. Compared with other methods, this one enjoys great robustness and high accuracy advantages. Therefore, it can be applicable in GPS blind areas. Moreover, there are some other models which can be added for further improvement of the MS2ML method, such as orientation model. In the future work, we will focus on incorporating orientations to the MS2ML method and further improve the localization accuracy.

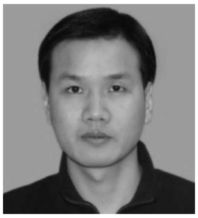
Acknowledgments

The work presented in this paper was funded by the Major Project of Technological Innovation in Hubei Province (No. 2016AAA007), National Natural Science Foundation (No. 51679181), the Science-technology Funds for Overseas Chinese Talents of Hubei Province (No.2016-12), the National Key Research and Development Program (2016YFB0100903) and Yingcai project of CUMT (YC20170001). We also appreciate Wuhan Kotei Informatics Company for the assistance in experiment setup and data collection.

About the Authors



Yicheng Li received the Bachelor degree in electrical engineering and automation from Hebei University of Architecture in 2011, and the Master degree in intelligent transportation engineering from Wuhan University of Technology, in 2014. He is currently a Ph.D. candidate in Wuhan University of Technology. His research interests include intelligent transportation systems, computer vision and image processing, 3D data processing.



Zhaozheng Hu received the Bachelor and PhD degrees from Xi'an Jiaotong University, China, in 2002 and 2007, respectively, both in information and communication engineering. He is now a professor in Wuhan University of Technology, Wuhan, China. His research topics mainly focus on 3D computer vision, intelligent transportation systems, multi-sensor systems, active surveillance system, etc.

Dr. Zhixiong Li (M'16) received his PhD in Transportation Engineering from Wuhan University of Technology, China. Currently he is a Senior Lecture with China University of Mining and Technology, China, and a research associate in Department of Mechanical Engineering, Iowa State University, USA. His research interests include mechanical system modeling and control. He is an associate editor for the Journal of IEEE Access.

Professor Miguel Angel Sotelo received the Dr. Eng. degree in Electrical Engineering in 1996 from the Technical University of Madrid, the Ph.D. degree in Electrical Engineering in 2001 from the University of Alcalá (UAH), Alcalá de Henares, Madrid, Spain, and the Master degree in Business Administration (MBA) from the European Business School in 2008.

Dr. Yulin Ma (M'13) received his PhD in Transportation Engineering from Wuhan University of Technology, China. He was a Post Doctor with Academy of Military Transportation, China. Currently he is an Associate Professor with National Center of ITS Engineering and Technology, Research Institute of Highway, Ministry of Transport, China. His research interests include intelligent vehicles and intelligent transportation systems. He is currently a reviewer for the IEEE Transactions on Intelligent Transportation Systems, International Journal of Intelligent Transportation Systems Research.

References

[1] K. Pahlavan, P. Krishnamurthy, and Y. Geng, "Locali-zation challenges for the emergence of the smart world," *IEEE Access*, vol. 3, pp. 3058–3067, 2015.

[2] B. Gao and B. Coifman, "Vehicle identification and GPS error detection from a LIDAR equipped probe vehicle," in *Proc. IEEE Intelligent Transportation Systems Conf.*, 2006, pp. 1537–1542.

[3] A. Grejner-Brzezinska, R. Da, and C. Toth, "GPS error modeling and OTF ambiguity resolution for high-accuracy GPS/INS integrated system," *J. Geodesy*, vol. 72, no. 11, pp. 626–638, 1998.

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

[5] H. Badino, D. Huber, and T. Kanade, "Real-time topo-metric localization," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2012, pp. 1635–1642.

[6] S. Kammel, J. Ziegler, B. Pitzer et al., "Team Annie-WAY's autonomous system for the 2007 DARPA urban challenge," *J. Field Robot.*, vol. 25, no. 9, pp. 615–639, 2008.

[7] H. Xu, Z. Wei, and Z. Jiang "3D visual SLAM with a Time-of-Flight camera," in *Proc. IEEE Signal Processing Systems Workshop*, 2015, pp. 1–6.

[8] G. Dubbelman and B. Browning, "COP-SLAM: Closed-form online pose-chain optimization for visual SLAM," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1194–1213, 2015.

[9] H. Zhao, J. Sha, Y. Zhao et al., "Detection and tracking of moving objects at intersections using a network of laser scanners," *IEEE Trans. Intell. Transp.*, vol. 13, no. 2, pp. 655–670, 2012.

[10] A. Angeli, D. Filliat, S. Doncieux et al., "A fast and incremental method for loop-closure detection using bags of visual words," in *Proc. Special issue on Visual SLAM, IEEE Trans. Robot.*, 2008, pp. 1–11.

[11] K. Shunsuke, G. Yanlei, and T. Hsu, "GNSS/INS/On-board camera integration for vehicle self-localization in urban canyon," in *Proc. IEEE Intelligent Transportation Systems Conf.*, 2015, pp. 2533–2538.

[12] H. Li, F. Nashashibi, and G. Toulminet, "Localization for intelligent vehicle by fusing mono-camera, low-cost GPS and map data," in *Proc. IEEE Intelligent Transportation Systems Conf.*, 2010, pp. 1657–1662.

[13] S. Nedevschi, V. Popescu, R. Danescu et al., "Accurate ego-vehicle global localization at intersections through alignment of visual data with digital map," *IEEE Trans. Intell. Transp.*, vol. 14, no. 2, pp. 673–687, 2013.

[14] Y. Gu, Y. Wada, L. Hsu et al., "Vehicle self-localization in urban canyon using 3D map based GPS positioning and vehicle sensors," in *Proc. Int. Conf. Connected Vehicles and Expo*, 2014, pp. 792–798.

[15] H. Kim, K. Choi, and I. Lee, "High accurate affordable car navigation using built-in sensory data and images acquired from a front view camera," in *Proc. IEEE Intelligent Vehicles Symp. IV*, 2014, pp. 808–813.

[16] Y. Gu, T. Hsu, and S. Kamijo, "GNSS/Onboard inertial sensor integration with the aid of 3-D building map for lane-level vehicle self-localization in urban canyon," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 4274–4287, 2016.

[17] J. Wang, H. Zha, and R. Cipolla, "Coarse-to-fine vision-based localization by indexing scale-invariant features," *IEEE Trans. Syst. Man, Cybern.*, vol. 36, no. 2, pp. 413–422, 2006.

[18] J. Son, S. Kim, and K. Sohn, "A multi-vision sensor-based fast localization system with image matching for challenging outdoor environments," *Expert Syst. Appl.*, vol. 42, no. 22, pp. 8830–8839, 2015.

[19] J. Ziegler, P. Bender, M. Schreiber et al., "Making Bertha drive: An autonomous journey on a historic route," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 2, pp. 8–20, 2014.

[20] A. Brubaker, A. Geiger, and R. Urtasun, "Map-based probabilistic visual self-localization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 4, pp. 652–665, 2016.

[21] M. Sefati, M. Daum, B. Sondermann et al., "Improving vehicle localization using semantic and pole-like landmarks," in *Proc. IEEE Intelligent Vehicles Symp. IV*, 2017, pp. 13–19.

[22] H. Badino, D. Huber, and T. Kanade. "Visual topometric localization," in *Proc. IEEE Intelligent Vehicles Symp. IV*, 2011, pp. 794–799.

[23] Y. Li, Z. Hu, and Y. Hu, "Vision-based vehicle localization using Bayesian topological model and hybrid k-nearest neighbor," Transportation Research Board, Tech Rep. 17-03371, 2017.

[24] P. Schmuck, A. Scherer, and A. Zell, "Hybrid metric-topological 3D occupancy grid maps for large-scale mapping," *IFAC-Papers OnLine*, vol. 49, no. 15, pp. 230–235, 2016.

[25] K. Konolige, E. Marder-Eppstein, and B. Marthi. "Navigation in hybrid metric topological maps," in *Proc. Int. Conf. Robotics and Automation*, 2011, pp. 3041–3047.

[26] H. Bay, A. Ess, T. Tuytelaars et al., "Speeded-up robust features (SURF)," *Comput. Vis. Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[27] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal.*, vol. 22, no. 11, pp. 1330–1334, 2000.

[28] R. Joseph and U. Perera. "Build-it-an interactive web application for 3D construction, interior and exterior design," in *Proc. IEEE 5th Int. Conf. Intelligent Systems Modeling and Simulation*, 2014, pp. 243–248.

[29] K. Ohno, T. Nomura, and S.Tadokoro. "Real-time robot trajectory estimation and 3D map construction using 3D camera," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2006, pp. 5279–5285.

[30] A. Geiger. [Online]. Available: http://www.cvlibs.net/datasets/kitti/