FULL PAPER

Dependence Maximization Localization: A Novel Approach to 2D Street-map-based Robot Localization

Kiyoshi Irie[a,b*] , Masashi Sugiyama[c,d] and Masahiro Tomono[a]

[a] *Future Robotics Technology Center, Chiba Institute of Technology*; [b] *Department of Computer Science, Tokyo Institute of Technology*; [c] *Center for Advanced Integrated Intelligence Research, RIKEN*; [d] *Department of Complexity Science and Engineering, Graduate School of Frontier Sciences, The University of Tokyo*

(*This is an electronic version of an article published in ADVANCED ROBOTICS, Volume 30, No.20, pp. 1431–1445, 2016. ADVANCED ROBOTICS is available online at:* `http://www.tandfonline.com/doi/abs/10.1080/01691864.2016.1222915`.)

Recently, localization methods based on detailed maps constructed using simultaneous localization and mapping has been widely used for mobile robot navigation. However, the cost of building such maps increases rapidly with expansion of the target environment. Here, we consider the problem of localization of a mobile robot based on existing 2D street maps. Although a large amount of research on this topic has been reported, the majority of the previous studies have focused on car-like vehicles that navigate on roadways; thus, the efficacy of such methods for sidewalks is not yet known. In this paper, we propose a novel localization approach that can be applied to sidewalks. Whereas roadways are typically marked, e.g., by white lines, sidewalks are not and, therefore, road boundary detection is not straightforward. Thus, obtaining exact correspondence between sensor data and a street map is complex. Our approach to overcoming this difficulty is to maximize the statistical dependence between the sensor data and the map, and localization is achieved through maximization of a mutual-information-based criterion. Our method employs a computationally efficient estimator of squared-loss mutual information, through which we achieve near real-time performance. The efficacy of our method is evaluated through localization experiments using real-world datasets.

Keywords: localization; navigation; mutual information

1. Introduction

This paper considers the problem of mobile robot localization. Recently, localization methods based on detailed maps constructed using simultaneous localization and mapping (SLAM) have been widely used with great success [1] [2]. However, the construction of such maps is very costly, because the necessary data collection involves a manual procedure (e.g., manual robot navigation). This cost can be eliminated by reusing existing street maps. Localization using street maps has primarily been studied with regard to roadways [3] [4], but no methods that are effective for sidewalks have been reported. One of the difficulties with regard to sidewalks is that road boundaries are not clearly marked; road boundary detection can be even more difficult than localization. However, localization is possible without road boundary detection. Here, we present a novel localization approach referred to as *dependence maximization localization*, which was inspired by multi-modal image registration.

*Corresponding author. Email: irie@furo.org

The method presented in this paper localizes a robot by maximizing the latent dependence between sensor data and a street map. Squared-loss mutual information (SMI), a variant of mutual information (MI) is employed to measure this dependence.

1.1 *Related Work*

Existing street-map-based localization methods can be classified into two categories. Methods designed to assist human car drivers and those for autonomous navigation. Typical applications within the first category are car navigation systems and advanced driver assistance systems (ADAS). A typical localization approach employed in such cases is to match a vehicle's trajectory with road network information extracted from street maps, assuming that the vehicle is always on the road. Methods using global navigation systems (GPS) for trajectory estimation have been studied for over a decade [3]. Recently, several authors have proposed the use of visual odometry and global localization without GPS has been shown to be possible [5] [6] [7]. However, the limitation of this approach is that the precise position with respect to the road, which is crucial information for autonomous navigation, cannot be estimated.

Although localization using street maps for autonomous navigation has also been studied, the majority of such studies have focused on roadways. As roadway boundaries are typically clearly marked, a large number of methods involve matching of the lane marking to the road boundaries on the map [4] [8]. Highly accurate localization results have been reported in such cases. However, several authors have proposed methods that do not use lane markings. For example, the method proposed by Morales *et al.* [9] estimates the offset between the road center and the robot from laser measurements. In that method, the road center is estimated from laser measurements with the assumption that the road boundaries can be determined from the height information. However, sidewalk environments vary considerably and, in some cases, road boundaries cannot be estimated from height information alone. Therefore, the above method is not directly applicable to sidewalks. In contrast, Hentschel and Wagner [10] have proposed a technique through which building boundaries are detected and correlated with the map. The advantage of using building boundaries is that they can be easily detected from laser measurements. However, this method is only applicable in environments where buildings are sufficiently observable and correctly mapped.

The difficulty of localization in sidewalks has been discussed by Irie and Tomono [11]. They have proposed the application of object recognition to manage varying road boundaries along sidewalks. In this approach, the recognized objects are matched with object prior probabilities embedded in the street map (e.g., the likelihood of curbs is high at the boundary between the sidewalk and roadway). Although this proposed framework is general, object prior probabilities vary from place to place. Therefore, applying their method to *new* environments should involve additional costs to determine the appropriate object prior probabilities.

1.2 *Contribution*

In this paper we propose a novel method that relaxes the limitations imposed on previous techniques. That is, our method does not explicitly detect road boundaries or recognize objects. Instead, sensor observations are directly matched to a map using a variant of MI. Although MI has already been used to register multi-modal sensor data [12] [13] [14], to the best of our knowledge, it has never been applied to street-map-based localization. The advantages of our method compared to existing methods can be summarized as follows:

- Robot trajectories are not limited within the road network.
- Detection of road or building boundaries is not required.
- No prior information e.g., on the colors, textures, or heights of objects in the environment is required.
- No other information besides boundary lines is required from the map. This indicates

that the efficacy of our method is independent of whether or not the regions on the map correspond to roads, buildings, etc.

Our method is the first to possess all the above characteristics.

In addition to outlining the proposed method, we also describe certain implementation details that render our method computationally efficient for practical navigation use. We evaluate the performance of our method via extensive experiments using real-world datasets, and we empirically demonstrate that SMI exhibits superior functionality among several dependence measures. Successful results obtained for long distance (up to 4.5 km) position tracking experiments support the efficacy of the proposed method.

The main contributions of the paper can be summarized as follows:

- Introduction of the dependence maximization approach to street-map-based localization.
- Computationally efficient implementation of the proposed method.
- The finding that SMI has superior functionality among several different dependence measures.
- Empirical demonstrations using real-world datasets showing that our method is both effective and sufficiently computationally efficient for practical use.

Although the essence of our approach has been presented previously in a conference proceedings [15], this paper expands on the evaluation of our system using additional experiments.

2. Proposed Method

As we employ a 2D street map, our method estimates the 2D pose (position and orientation) of the robot $\boldsymbol{w} = (w_x, w_y, w_\theta)$. We are particularly interested in localization for navigation; in this paper we focus on position tracking under the assumption that the initial position of the robot is known (with some errors).

2.1 *Map Representation*

Our method uses 2D street maps consisting of boundary lines as shown in Figure 1 (a). During pre-processing, each closed region and boundary line on the map is given a unique ID. We refer to the IDs as *segment IDs* in this paper. Sample segment IDs are shown in Figure 1 (b). The assumption behind this procedure is that map segments are defined according to the objects in the environment. In other words, we expect that similar objects exist within a segment and different objects exist in different segments (e.g., ID = 1 corresponds to buildings and ID = 7 corresponds to curbs). Because sensor observations naturally depend on the *objects* being observed, sensor data and segment IDs should have a latent dependence. An example of the latent dependence between sensor data, objects, and the map is depicted in Figure 1 (c). The proposed method exploits this dependence for robot localization.

2.2 *Approach*

Our approach was inspired by a method used for image registration problems. One of the well-known image registration methods is maximization of the normalized cross correlation (NCC). Maximizing the NCC can be interpreted as maximizing the dependence between the pixel intensities of two images. It is assumed that the same object exhibits similar intensity values in two images; therefore the pixel intensities of these two images should have *linear* dependence. However, if two images are of different modalities (e.g., with regard to the registration of medical computed tomography (CT) and magnetic resonance (MR) images [16]) the pixel intensities can have *non-linear* dependence. That is, some objects rendered in white in CT images appear

black in MR images, and others do not. Maximizing MI is known to be effective for this kind of multi-modal image registration [12] because MI can measure non-linear dependence. Our method applies this concept to robot localization; the sensor data and map segments may have non-linear dependence and, thus we maximize the MI-based dependence measure to localize a robot.

2.3 *Dependence Maximization Localization*

The robot observes n different positions using onboard sensors and features are extracted from the sensor data (such as colors and shapes). We denote the extracted features by $\{\boldsymbol{x}_i\}_{i=1}^n$. Using a robot pose hypothesis $\boldsymbol{w}$, we find the segment IDs that correspond to the segments in which the features are located; these are denoted by $\{y_i^{(\boldsymbol{w})}\}_{i=1}^n$ (it is assumed that the relative positions between the robot and the sensor data source are known). Consequently, we obtain pairs of feature and segment ID $\{(\boldsymbol{x}_i, y_i^{(\boldsymbol{w})})\}_{i=1}^n$, which are used as input data.

The features and map segment IDs are denoted by random variables $\boldsymbol{X}$ and $\boldsymbol{Y}^{(\boldsymbol{w})}$. By assuming that the input data is randomly generated from a joint probability distribution $p(\boldsymbol{X}, \boldsymbol{Y}^{(\boldsymbol{w})})$, we search for the hypothesis $\boldsymbol{w}$ that maximizes a dependence measure. The estimated robot pose $\hat{\boldsymbol{w}}$ can be obtained by

$$\hat{\boldsymbol{w}} = \operatorname*{argmax}_{\boldsymbol{w}} f(\boldsymbol{X}, \boldsymbol{Y}^{(\boldsymbol{w})}). \tag{1}$$

We employ SMI [17] as the measure function f. The definition and estimation of SMI are detailed in Section 3.

To facilitate an intuitive understanding of our method, an illustrative example is provided in Figure 2. In this example, the robot obtains five color features $(\boldsymbol{x}_1, ..., \boldsymbol{x}_5)$, where $\boldsymbol{x}_1 = \boldsymbol{x}_2 =$ green, $\boldsymbol{x}_3 =$ yellow and $\boldsymbol{x}_4 = \boldsymbol{x}_5 =$ gray. When the robot position is correct (Figure 2 (a)), the corresponding segment IDs are $y_1 = y_2 = 1, y_3 = 2, y_4 = y_5 = 3$. The conditional entropy of the features given map segment is small, because each segment of the map corresponds to a single color. Therefore, the dependence between $\boldsymbol{X}$ and $\boldsymbol{Y}$ is high. On the other hand, when the robot position is incorrect as shown in Figure 2 (b), segment ID = 3 corresponds to multiple colors, yielding a reduced dependence.

3. Estimation of Squared-loss Mutual Information

We employ a computationally efficient estimator of SMI that was presented by Sakai *et al.* [18]. In this section we briefly review this method in the interest of completeness.

3.1 *Squared-loss Mutual Information*

Mutual information (MI) is a measure of the statistical dependence between random variables and is used to find dependencies in data. Medical image registration is one of the well-known applications of MI [12]. In our method, we employ SMI [17] rather than the original MI to measure the dependence between the sensor data and map. MI and SMI are defined as

$$\mathrm{MI}(\boldsymbol{X}, \boldsymbol{Y}) := \iint p(\boldsymbol{x}, y) \log \frac{p(\boldsymbol{x}, y)}{p(\boldsymbol{x})p(y)} \mathrm{d}\boldsymbol{x}\mathrm{d}y, \tag{2}$$

$$\mathrm{SMI}(\boldsymbol{X}, \boldsymbol{Y}) := \frac{1}{2} \iint p(\boldsymbol{x})p(y) \left(\frac{p(\boldsymbol{x}, y)}{p(\boldsymbol{x})p(y)} - 1 \right)^2 \mathrm{d}\boldsymbol{x}\mathrm{d}y. \tag{3}$$

The random variables $\boldsymbol{X}$ and $\boldsymbol{Y}$ are statistically independent if and only if

$$p(\boldsymbol{x}, y) = p(\boldsymbol{x})p(y). \tag{4}$$

Thus, the dependence can be measured by evaluating the "distance" between the left and right hand sides of the Eq. (4). MI can be regarded as the Kullback-Leibler divergence from $p(\boldsymbol{x}, y)$ to $p(\boldsymbol{x})p(y)$ and therefore it can measure the statistical dependence. Estimating MI is, however, computationally rather expensive. Further, this approach is sensitive to outliers [19] [20] because of the presence of the logarithm term.

On the other hand, SMI can be regarded as the Pearson divergence from $p(\boldsymbol{x}, y)$ to $p(\boldsymbol{x})p(y)$. Because both the Kullback-Leibler divergence and the Pearson divergence belong to the class of f-divergences [21] [22], they share similar properties. Both MI and SMI are always non-negative, and $\boldsymbol{X}$ and $\boldsymbol{Y}$ are statistically independent if and only if $\mathrm{MI}(\boldsymbol{X}, \boldsymbol{Y})$ or $\mathrm{SMI}(\boldsymbol{X}, \boldsymbol{Y})$ is equal to zero. In the proposed method, we employ SMI as the measure of the dependence because SMI has superior robustness against outliers (there is no logarithm term) and a computationally efficient SMI estimator is known [18].

3.2 *Estimating Squared-loss Mutual Information Using Least-squares Mutual Information*

We employ least-squares mutual information (LSMI) [18] to estimate SMI. We briefly review LSMI here. A naive means of estimating SMI is to estimate all components in Eq. (3) ($p(\boldsymbol{x}), p(y)$, and $p(\boldsymbol{x}, y)$) individually and substitute them to the original equation. However, this approach tends to result in large estimation errors, because the estimation errors in each component are combined and magnified. Instead, LSMI estimates the density ratio $r(\boldsymbol{x}, y)$ directly (without estimating each component), where

$$r(\boldsymbol{x}, y) = \frac{p(\boldsymbol{x}, y)}{p(\boldsymbol{x})p(y)}. \tag{5}$$

This improves both the accuracy and robustness. Here, $r(\boldsymbol{x}, y)$ is modeled using a multiplicative kernel model, where

$$r_{\boldsymbol{\Theta}}(\boldsymbol{x}, y) := \sum_{l=1}^{b} \sum_{l'=1}^{b} \boldsymbol{\Theta}_{l,l'} K(\boldsymbol{x}, \tilde{\boldsymbol{x}}_l) L(y, \tilde{y}_{l'}^{(\boldsymbol{w})}). \tag{6}$$

The parameter to estimate $\boldsymbol{\Theta}$ is a matrix of $b \times b$ ($b \leq n$) and each element in it is denoted by $\boldsymbol{\Theta}_{l,l'}$. The kernel centers $\{(\tilde{\boldsymbol{x}}_i, \tilde{y}_i^{(\boldsymbol{w})})\}_{i=1}^{b}$ are randomly chosen from the input data. We use the Gaussian kernel for K and the delta kernel for L as shown below.

$$K(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \exp\left(-\frac{||\boldsymbol{x} - \tilde{\boldsymbol{x}}||^2}{2\sigma^2}\right), \tag{7}$$

$$L(y, \tilde{y}) = \begin{cases} 1, & \text{if} \quad y = \tilde{y}, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

The parameter $\boldsymbol{\Theta}$ is learned so as to minimize the squared error $J(\boldsymbol{\Theta})$ in Eq. (**??**).

$$\min_{\boldsymbol{\Theta}} J(\boldsymbol{\Theta}), \tag{9}$$

$$J(\boldsymbol{\Theta}) := \iint (r_{\boldsymbol{\Theta}}(\boldsymbol{x}, y) - r(\boldsymbol{x}, y))^2 p(\boldsymbol{x})p(y)\mathrm{d}\boldsymbol{x}\mathrm{d}y. \tag{10}$$

By setting the derivative of $J(\boldsymbol{\Theta})$ to zero and approximating the expectations using empirical averages, and by adding an ℓ_2 regularization term to avoid overfitting, we obtain the following discrete Sylvester equation.

$$\frac{1}{n^2}\boldsymbol{K}^T\boldsymbol{K}\hat{\boldsymbol{\Theta}}\boldsymbol{L}^T\boldsymbol{L} + \lambda\hat{\boldsymbol{\Theta}} = \frac{1}{n}\boldsymbol{K}^T\boldsymbol{L}. \tag{11}$$

By solving this equation, we obtain the minimizer of Eq. (**??**). The dimensions of matrices $\boldsymbol{K}$ and $\boldsymbol{L}$ are $n \times b$, and their elements are given by $\boldsymbol{K}_{i,l} = K(\boldsymbol{x}_i, \tilde{\boldsymbol{x}}_l)$ and $\boldsymbol{L}_{i,l} = L(y_i, \tilde{y}_l^{(\boldsymbol{w})})$. The Gaussian kernel width σ and regularization parameter λ are hyper parameters, which can be determined through cross-validation with respect to Eq. (**??**). Finally, the estimated SMI is expressed as

$$\mathrm{LSMI} = \frac{1}{2n}\mathrm{tr}(\boldsymbol{K}\hat{\boldsymbol{\Theta}}\boldsymbol{L}^T) - \frac{1}{2}. \tag{12}$$

We use the K-fold cross-validation ($K = 5$ in our experiments) to determine σ and λ. The input data is split into K disjoint sets. Each set is used as test data in turn, and the remainder of the data is used to estimate $\boldsymbol{\Theta}$. $J(\boldsymbol{\Theta})$ (Eq. (**??**)) is evaluated for each estimation. Hyper parameters that minimize the average error are chosen from several candidates.

4. Position Tracking Using a Particle Filter

Above, we have discussed robot localization from one-shot observation. Now, we move to time-series localization of a moving robot. This problem is also known as position tracking and has been intensively studied for decades [23] [24]. At present, methods fusing a robot's motion estimation and observations using a particle filter are widely used [2] [4] [10] . In this section, we describe the integration of dependence maximization localization with a particle filter.

4.1 *SMI-based Particle Filter Localization*

In each particle-filter prediction step, the robot predicts its pose using the motion $\boldsymbol{u}_t$, which denotes the difference in the estimated pose between the previous time step $t-1$ and the current step t; this information can be obtained through odometry, for example. The robot pose for the j-th particle is denoted by $\boldsymbol{w}_j$ and the particles are drawn from a proposal distribution

$$\boldsymbol{w}_j \sim p(\boldsymbol{w}^t | \boldsymbol{w}_j^{t-1}, \boldsymbol{u}_t). \tag{13}$$

During the particle-filter update steps, the robot extracts features $\{\boldsymbol{x}_i\}_{i=1}^n$ from the collected sensor data. The particles are then resampled according to the weight (W) proportional to the observation likelihood as

$$W_j \propto p(\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^n | \boldsymbol{w}_j, \boldsymbol{M}). \tag{14}$$

Here, the map is denoted by $\boldsymbol{M}$. We estimate the SMI for each particle and interpret the result as the observation likelihood

$$p(\boldsymbol{X} | \boldsymbol{w}_j, \boldsymbol{M}) = \mathrm{SMI}(\boldsymbol{X}, \boldsymbol{Y}^{(\boldsymbol{w}_j)}). \tag{15}$$

4.2 *Computationally Efficient Implementation*

In the update steps, the number of particles greatly affects the processing time. Although the LSMI core calculation is computationally efficient, the cross-validation process consumes a considerable amount of time. This can be a hindrance to the implementation of a real-time system. In our implementation, we reduce the number of cross-validation procedures under the assumption that the changes in the hyper parameters between successive update steps are small. More specifically, we augment the state vector $\boldsymbol{s}$ by appending σ and λ such that

$$\boldsymbol{s}_j = (\boldsymbol{w}_j, \sigma_j, \lambda_j). \tag{16}$$

In the very first update step, we perform cross-validation for all particles and store the chosen parameters in $\boldsymbol{s}$. In the subsequent update, we define an *omission rate* and randomly choose particles for which the cross-validation step is omitted. For example, if the omission rate is set to 70%, we perform cross-validation for only 30% of the particles and reuse the previously chosen hyper parameters for the remaining particles. In this manner, we can reduce the processing time by increasing the omission rate.

5. Experiments

In this section we demonstrate the performance of our method through two types of experiments: localization from a single image and position tracking. We employ four datasets comprised of sensor data collected in an urban area of Narashino, Chiba, Japan (Table 1).

5.1 *Embedding Segment IDs on Street Map*

In the following experiments, we employed a street map generated by extracting segments in Google Maps$^{\mathrm{TM}}$ [1]. A Google Maps$^{\mathrm{TM}}$ screen capture was saved as an image and closed regions were extracted manually. Then, boundary lines were extracted using the Sobel filter. Finally, unique IDs were assigned to the segments using the flood-fill algorithm [26]. Figure 3 illustrates the procedure[2].

5.2 *Single Image 2-degrees-of-freedom Localization*

First, we evaluated the efficacy of SMI as a localization measure.

5.2.1 Setup

In the following experiments, 2-degree-of-freedom (DoF) robot poses (lateral position and orientation with respect to road) were estimated from single images. Note that 3-DoF localization was not performed, because 2D street maps are often non-discriminative in terms of the longitudinal direction and, also, the longitudinal position can be easily estimated from the robot's motion estimation (e.g., using odometry).

Colors and gradients were extracted from the images and used as features (summarized in Table 2). The set of features for the i-th pixel is denoted by $\boldsymbol{x}_i$. The correspondence between segment IDs and pixels was obtained by projecting the street map onto the image plane (Figure 4). The relative pose between the robot base and the camera was assumed to be known. The segment ID projected on the i-th pixel for $\boldsymbol{w}$ is denoted by $y_i^{(\boldsymbol{w})}$.

[1] Although other maps such as OpenStreetMap [25] can also be used, geometrically accurate maps are preferred. We employed Google Maps$^{\mathrm{TM}}$ in this paper because, for our target area, it provided significantly more accurate sidewalk information than OpenStreetMap.

[2] This process is unnecessary if we have vector data for the map.

After the input data pairs $\{(\boldsymbol{x}_i, y_i^{(\boldsymbol{w})})\}_{i=1}^n$ were obtained, a grid search was used to find the position that maximizes Eq. (1). The search ranges were -2 m to $+2$ m with 0.2 m intervals for the lateral offset and $-20°$ to $+20°$ with $2°$ intervals for orientation.

5.2.2 Comparative Method Based on Road Boundary Matching

Although no existing localization method using the same setup is known, we implemented a comparative method based on road boundary matching. Road boundaries were detected using the road detection method proposed by Kong *et al.* [1] [27], which uses vanishing point detection and does not require any prior information on the road color or texture. The detected road boundaries were matched with lines on the map. The robot poses at which the boundary lines match best were found using the same grid search mentioned above.

5.2.3 Results

We conducted single image localization experiments using two datasets: A) 37 images (32 from sidewalks and 5 from roadways) with manually given ground truth positions and, B) 575 images with GPS reference positions. We evaluated the accuracy, robustness, and processing time of our method. We also performed comparison of different dependence measures.

5.2.3.1 Localization Accuracy. The localization results obtained for the two datasets are summarized in Figure 5. Our method outperformed the road matching method in terms of localization accuracy. Further, the average position and orientation errors were 0.45 m and 4.3° for dataset A and 0.64 m and 6.5° for dataset B. The map resolution was approximately 23.5 cm per pixel and, thus, a position error of 0.64 m corresponded to 2.7 pixels in the map image. We are of the view that the localization accuracy is reasonable, because the localization error includes errors of the map itself and the street map used here was not pixel-wise accurate.

Exemplary results for dataset A are shown in Figure 6, with LSMI plots against the position and orientation errors (the rightmost column). Successful examples for both methods are shown in Figure 6 (a) and (b). The LSMI plot for (a) exhibits a clear peak around the ground-truth position. Although the road detection method failed on Figure 6 (c-e), which was because of the confusing salient edges in the images, our method exhibited successful localization. Braille blocks caused several localization failures. Two peaks can be seen in the plot of Figure 6 (f); the highest peak appeared at the ground-truth and the second highest peak corresponded to the Braille blocks. Further, some of the localization errors were caused by map errors. An example of a road width error can be seen in Figure 6 (g). Both the road detection method and our method failed in the case shown in Figure 6 (h), where the road boundaries were unclear and the feature differences between the roadway, the sidewalk and the parking lot were very small. A challenging example of map errors is shown in Figure 6 (i). The left road boundary on the map appears curved, but this is inaccurate. Another challenge is coping with occlusions such as the example shown in Figure 6 (j); our method failed in this case because the blue tarp covering the road was of a similar color to the vehicle that appears on the right hand side of the image.

5.2.3.2 Comparison of Dependence Measures. We evaluated our proposed method using different dependence measures: ordinary MI, SMI, quadratic mutual information (QMI) [28], and squared-loss conditional entropy (SCE) [29]. The results are summarized in Table 3 and exemplary score distributions are shown in Figure 7. SMI performed best in terms of both accuracy and processing time. All dependence measures except MI are considered to be robust measures (MI's sensitivity can be seen from the spikes in the MI plot). QMI, which measures the dependence based on the ℓ_2 distance between $p(\boldsymbol{x}, y)$ and $p(\boldsymbol{x})p(y)$, exhibited the second best performance. SCE can also measure dependence robustly, and its score plot appears smooth; however,

[1] We used the code downloaded from `http://web.mit.edu/huikong/www/code.html` .

SCE($\boldsymbol{Y}|\boldsymbol{X}$) had high values where all observations indicated a single segment ID. Therefore the SCE performance was poor.

5.2.3.3 Comparison of Image Size. In the experiments using dataset A, images of 320×240 pixels were used to extract features. They were then downsampled to 80×60 pixels. We compared the performance of our system for different downsampling ratios and the results are summarized in Figure **??**. It is apparent that the localization accuracy was not largely affected even when downsampled by a factor of 5 (both in the height and width directions, resulting in 64 × 48 pixels).

5.3 *Position Tracking*

We integrated our method with a particle filter and conducted position tracking experiments.

5.3.1 Setup

Position tracking experiments were executed using datasets collected by manually navigating a wheel-chair robot (Figure 8). The datasets include LiDAR (Velodyne HDL-32e) measurements, images from an omnidirectional camera (Pointgrey Ladybug3), gyro-assisted odometry, and real time kinematic (RTK) GPS data for reference. Localization experiments were conducted off-line using the collected data.

Gyro-assisted odometry was used for the particle-filter prediction. The update step was executed every 2 s (0.5-Hz frequency) using the local-grid-map measurement model depicted in Figure 9. In each update step, a local grid map was generated using the point cloud and camera images accumulated since the previous update step. The six-dimensional features listed in Table 4 were extracted for each cell in the grid map. They are denoted by $\{\boldsymbol{x}_i\}_{i=1}^n$ (n denotes the number of cells in the local grid map). After extracting the features, the particles were weighted using SMI and resampled. The input data pairs $\{(\boldsymbol{x}_i, y_i^{(\boldsymbol{w})})\}_{i=1}^n$ were obtained by matching the local grid map with the street map using the robot pose hypothesis $\boldsymbol{w}$.

The details of the feature extraction procedure are as follows. Each point in the cloud was assigned a color (RGB) and gradient magnitude (the response of the Sobel filter) feature by projecting the point onto the image plane. As geometric features, we employed the elevation of the point and laser scan discontinuity as described in [31]. The laser scan discontinuity was calculated for each point p_i in the point cloud, using

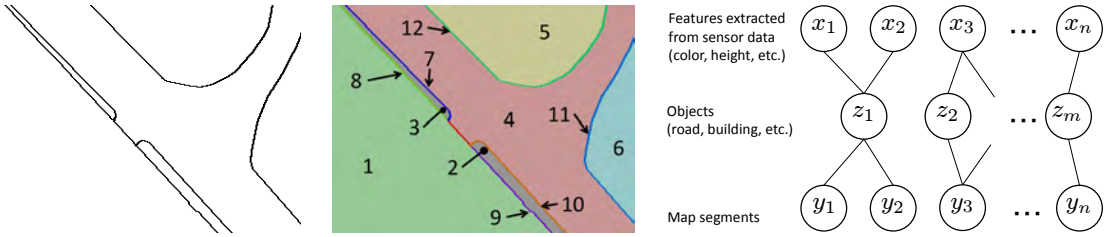
$$\max(R_{p_{i-1}} - R_{p_i}, R_{p_{i+1}} - R_{p_i}, 0)^{0.5}, \tag{17}$$

where the R_{p_i} denotes the LiDAR range measurement for the point p_i, and p_{i-1} and p_{i+1} are the neighboring points. After the abovementioned point-wise feature extraction, the points were projected onto the ground plane and stored in the grid cells. The cell-wise features $\boldsymbol{x}$ were obtained by averaging the point-wise features and scaling them to the range between 0 to 1.

Some other details of the evaluation setup are as follows. The size of the local grid maps were 10 m×10 m and the cell size was 10 cm. To reduce the processing time, the grid map was downsampled by a factor of 4 before SMI estimation. The number of particles was 100, and the number of kernel centers used to estimate the SMI was 40. Unless otherwise noted, the initial particle distribution was given by a normal distribution with a standard deviation of 1 m and 5°, and the cross-validation omission rate was set to 90%. Our implementation was written in MATLAB and a laptop computer with a Core i7-5500U CPU was used to measure the processing time.

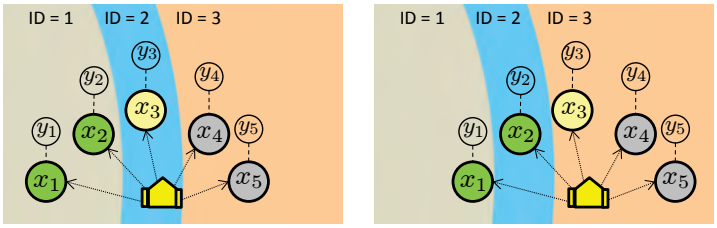
5.3.2 Comparative Method

Because no existing method can be directly compared to the method presented in this paper, we implemented a comparative method based on road boundary detection and matching. We



(a) (b) (c)

Figure 1. (a) Sample map used in proposed method. (b) Map interpretation. Different colors indicate different segment IDs. (c) Illustrative model of the world. The map and sensor data have latent dependence.

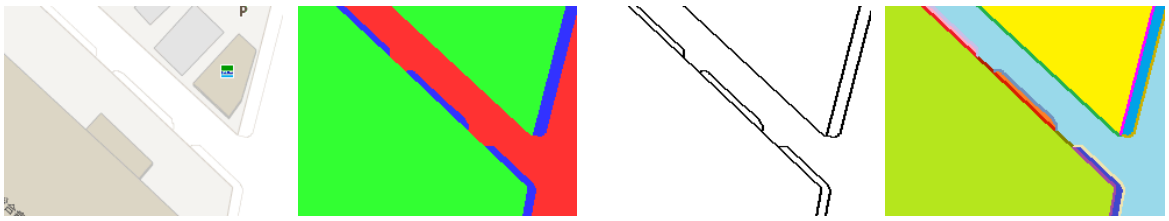


(a) correct localization (high dependence) (b) incorrect localization (low dependence)

Figure 2. Illustration of our localization approach. In this example, the robot obtains five color features ($\boldsymbol{x}_1, ..., \boldsymbol{x}_5$) from different positions. Corresponding map segment IDs ($y_1, ..., y_5$) are determined using the current robot pose hypothesis. (a) The dependence between features $\boldsymbol{x}$ and segment IDs y is high when the robot is correctly localized because y can be uniquely determined from $\boldsymbol{x}$. (b) The dependence decreases when the robot is incorrectly localized.

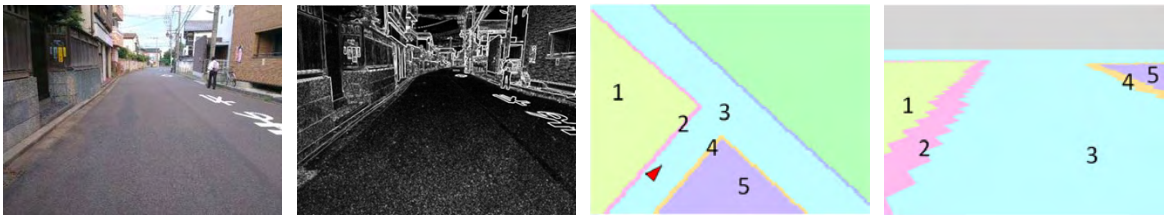
Table 1. List of datasets used in experiments

Dataset	Description
A	37 monocular camera images (320×240) with manually labeled positions
B	575 monocular camera images (320×200) with GPS reference positions
C	3D LiDAR scans and omnidirectional camera images, and 150m odometry with GPS reference trajectory
D	3D LiDAR scans and omnidirectional camera images, and 4.5km odometry with GPS reference trajectory



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

Figure 3. Illustration of map generation procedure. (a) Original map image captured from Google Maps™. (b) Closed regions extracted manually. (c) Boundary lines detected using edge-detection algorithm. (d) Unique IDs assigned using flood-fill algorithm [26]. Different colors indicate different IDs.

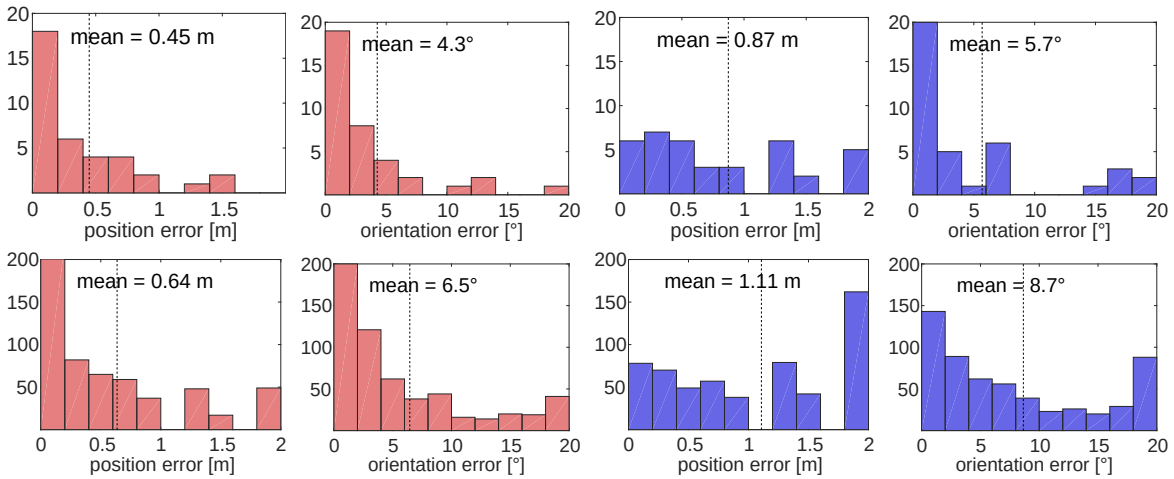


(a) (b) (c) (d)

Figure 4. Single image observation model. (a) Input image. (b) Extracted gradient feature. (c) Map and robot pose (red triangle). (d) Segment IDs projected onto image plane.

Table 2. Features used in single image localization experiments.

Category	Feature	Dimension
Color	R, G, B	3
Gradient	Sobel magnitude, orientation	2



(a) Proposed method (b) Road boundary matching

Figure 5. Histograms of position and orientation errors from single image localization. Top: Dataset A results (37 images). Bottom: Dataset B results (575 images).

Table 3. Comparisons of the four different dependence measures (dataset A).

Measure	Mean position error [m]	Mean orientation error [°]	Run time (per pose hypothesis) [s]	Estimator
SMI	0.45 ±0.42	4.3 ±4.5	0.26 ±0.007	LSMI [18]
MI	1.04 ±0.67	11.9 ±7.0	6.7 ±0.56	MLMI [30]
QMI	0.77 ±0.70	7.0 ±7.0	0.30 ±0.03	LSQMI [28]
SCE	1.64 ±0.56	17.2 ±3.0	1.4 ±0.28	LSCE [29]

Table 4. Features used in position tracking experiments.

Category	Feature	Dimension
Image color	color (R, G, B)	3
Image gradient	Sobel magnitude	1
Height	elevation of the points	1
Height change	laser scan discontinuity	1

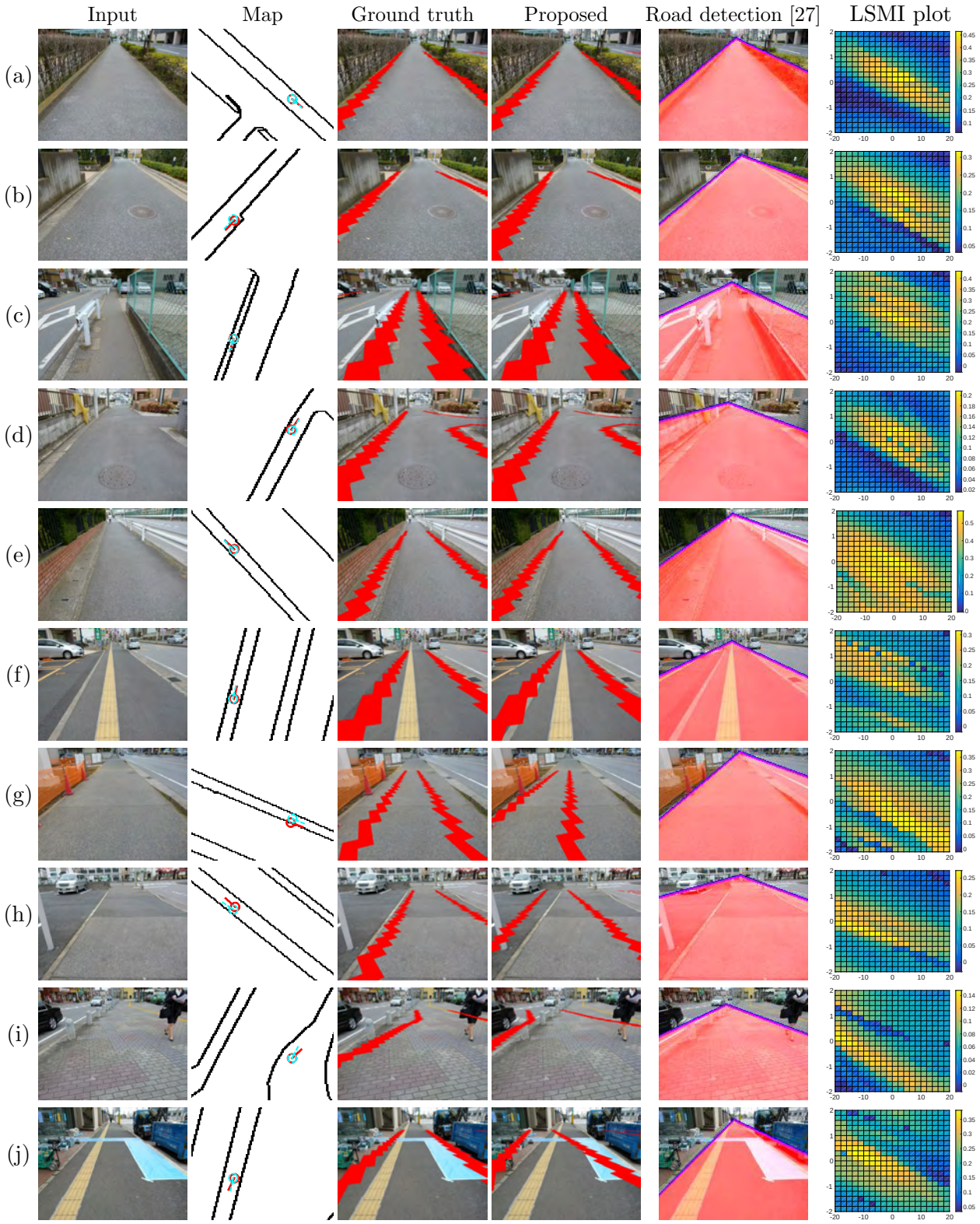


Figure 6. Exemplary results of localization experiments. The localization results are shown by superimposing the road boundaries onto the images. In the maps the estimated pose and ground-truth positions are indicated in red and blue, respectively. The horizontal and vertical axes in the LSMI plots represent the orientation error [°] and the position error [m], respectively. In the comparative method, the detected road boundaries were matched with the map for localization.

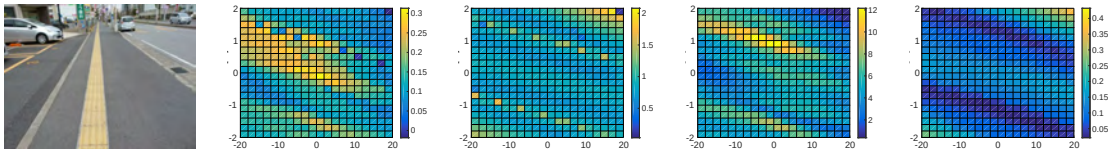


Figure 7. Comparison of score distributions for different dependence measures. The horizontal and vertical axes in the plots represent the orientation error [°] and the position error [m], respectively.

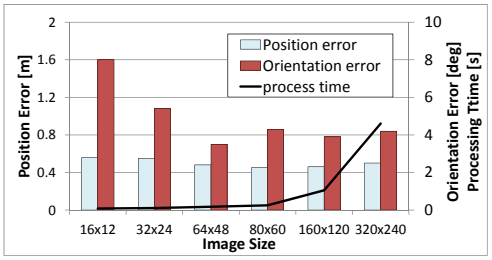


Figure 8. Performance comparison for different image downsampling ratios. The mean position and orientation errors and the processing time per position hypothesis are shown.

Figure 9. Robot used to collect datasets for position tracking experiments. It is equipped with a 3D LiDAR, an omnidirectional camera, a monocular camera, a gyro and a GPS receiver.

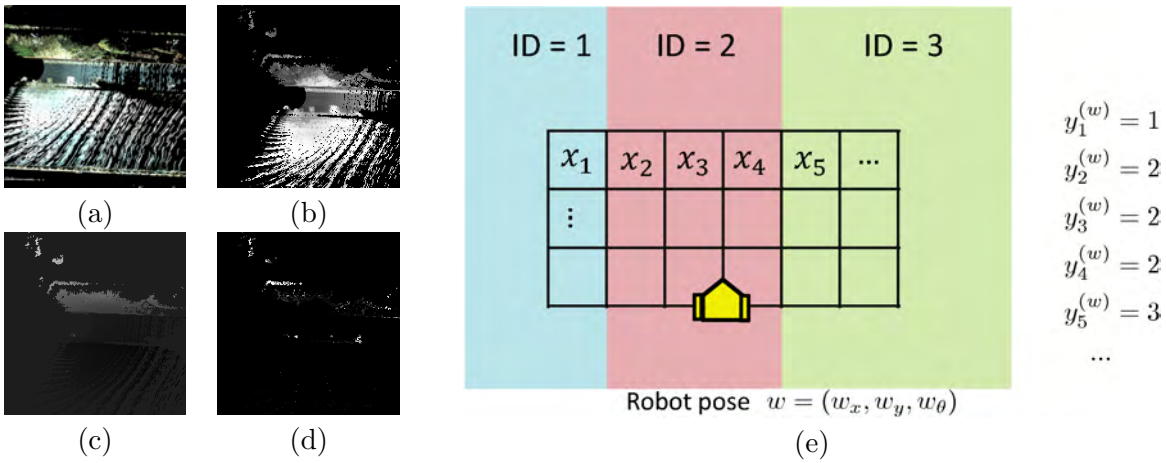


Figure 10. Sample of local grid map generation and measurement model. (a) 3D point cloud obtained by a LiDAR and cameras. A grid map is generated from the point cloud and features stored in the grid map are shown in (b-d). (b) Color feature (red channel). (c) Height feature. (d) Laser discontinuity feature. (e) Illustrative example of data association between a grid map and the map.

incorporated the idea of road edge detection using height changes and image edges [32]. The comparative method employed the same particle filter used in the proposed method, except for the likelihood calculation. The likelihood for each particle was approximated by the correlation between the road boundaries in the street map and the road boundary likeliness of the local grid cells as shown below:

$$p(\{\boldsymbol{x}_i\}_{i=1}^n|\boldsymbol{w},\boldsymbol{M}) = \frac{\sum_{i=1}^n f_{\text{local}}(\boldsymbol{x}_i) f_{\text{map}}(y_i^{(\boldsymbol{w})})}{\sqrt{\sum_{i=1}^n f_{\text{local}}(\boldsymbol{x}_i)^2}\sqrt{\sum_{i=1}^n f_{\text{map}}(y_i^{(\boldsymbol{w})})^2}}, \tag{18}$$

$$f_{\text{local}}(\boldsymbol{x}) = x^{(\text{g})} \cdot x^{(\text{d})}, \tag{19}$$

$$f_{\text{map}}(y) = \begin{cases} 1, & \text{if } y \text{ is road boundary,} \\ 0, & \text{otherwise.} \end{cases} \tag{20}$$

Here, $x^{(\text{g})}$ and $x^{(\text{d})}$ denote the image gradient feature and laser discontinuity feature, respectively.

5.3.3 Results

Here, we show results obtained using two datasets C and D, which were collected in Narashino, Chiba, Japan. The duration and distance of the navigation was 127 s and 150 m for dataset C, and 3,514 s and 4.5 km for dataset D, respectively.

5.3.3.1 Localization Accuracy. We conducted position tracking experiments 100 times for each dataset using different random seeds. Both exemplary results and camera images are shown in Figures 10 and 11. In both figures, it can be seen that our method successfully corrected the odometry errors and tracked the robot position, while the comparative method resulted in large errors for dataset D.

The localization accuracy was evaluated based on the root mean squared (RMS) error using GPS as the ground-truth. Table 5 summarizes the results. The mean RMS error of the proposed method over 100 runs was 0.6 m (approximately 2.6 pixels on the map) for dataset C and 1.33 m for dataset D. The errors were significantly smaller ($p < 0.01$ according to the paired t-test) than those of the comparative method. Note that the estimation errors include both GPS and map errors. Figure 12 shows an example of map errors observed at (D) in Figure 11. The difference between the road position in the satellite image and on the map was approximately 2 m. Nevertheless, our method "correctly" tracked the robot position along the road indicated on the map.

5.3.3.2 Comparison of Cross-Validation Omission. We compared the computational time of the particle-filter update and the localization accuracy for different omission rates. For each omission rate, we conducted position tracking 100 times using dataset C. Figure 13 (a) summarizes the results. By omitting 90% of the cross-validation procedures, the average processing time decreased from 3.2 s to 0.6 s, without significant degradation of the localization accuracy.

5.3.3.3 Robustness Against Odometry Errors. The robustness against odometry errors was evaluated using distorted odometry logs. We generated distorted odometry logs by adding Gaussian noise with different noise levels. For each noise level, 100 distorted odometry logs using different random seeds were generated. We compared the ratio of successful localizations (where the final position error was within 2 m). The results are summarized in Figure 13 (b). As can be seen from that figure, our method showed superior robustness compared to the comparative method. Further, our system maintained good localization accuracy even with relatively large

odometry errors. Accuracy degradation began when the standard deviation of the noise exceeded 20% and 2°/s for the distance and orientation, respectively.

5.3.3.4 Robustness Against Initial Position Errors. The robustness against initial position errors was evaluated using inaccurate initial positions. We randomly shifted the mean of the initial particle distribution by 1–6 m and conducted position tracking experiments. The standard deviations of the initial particle distribution were matched with the position error sizes. We evaluated the ratio of successful localizations using 100 differently shifted initial positions for each error size. The results are summarized in Figure 13 (c). The proposed method maintained a 100% success rate for initial position errors of up to 1 m. Although the localization accuracy degraded as the error level increased, the proposed method was superior to the comparative method in terms of the degradation speed.

6. Conclusions

This paper described a robot localization method based on a 2D street map. The key issue addressed in this paper was determining the correct correspondence between the 2D street map and sensor measurements. The proposed method finds the correspondence to localize a robot through maximization of the statistical dependence between segments of the map and features extracted from the sensor data. This dependence is measured using the squared-loss version of a mutual information estimator, which is robust and computationally efficient.

The performance of our system was evaluated through extensive experiments using images and laser data collected in real-world urban environments. The proposed method exhibited reasonable localization accuracy. We also observed that SMI exhibited the best performance among several examined dependence measures in terms of both localization accuracy and processing time. Note that the robustness against occlusions and map error management of this method should be improved through further research.

As dependence maximization localization functions without road boundary detection, this approach should be particularly useful for environments that are not as structured as urban roadways. The successful localization results obtained without using any prior knowledge of the environment allow us to hope that our method can be easily applied to *new* environments. Finally, because our method is general and not restricted to the road network, it should be applicable to localization of other types of robots, such as flying robots and indoor mobile robots.

References

[1] Levinson J, Thrun S. Robust vehicle localization in urban environments using probabilistic maps. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Anchorage, Alaska, USA. 2010. p. 4372–4378.

[2] Kümmerle R, Ruhnke M, Steder B, Stachniss C, Burgard W. A navigation system for robots operating in crowded urban environments. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Karlsruhe, Germany. 2013. p. 3225–3232.

[3] Brakatsoulas S, Pfoser D, Salas R, Wenk C. On map-matching vehicle tracking data. In: Proceedings of the 31st International Conference on Very Large Data Bases (VLDB). Trondheim, Norway. 2005. p. 853–864.

[4] Chausse F, Laneurit J, Chapuis R. Vehicle localization on a digital map using particles filtering. In: Proceedings of the IEEE Intelligent Vehicles Symposium (IV). Las Vegas, Nevada, USA. 2005. p. 243–248.

[5] Parra Alonso I, Fernandez Llorca D, Gavilan M, Alvarez Pardo S, Garcia-Garrido M, Vlacic L, Sotelo M. Accurate global localization using visual odometry and digital maps on urban environments. IEEE Transactions on Intelligent Transportation Systems. 2012;13(4):1535–1545.

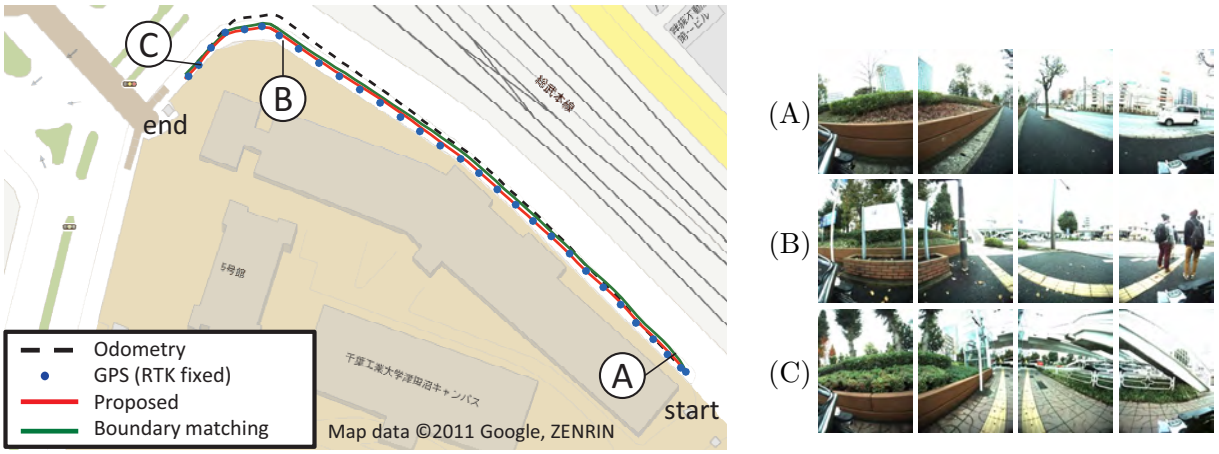


Figure 11. Position tracking result for dataset C (150 m, 127 s).

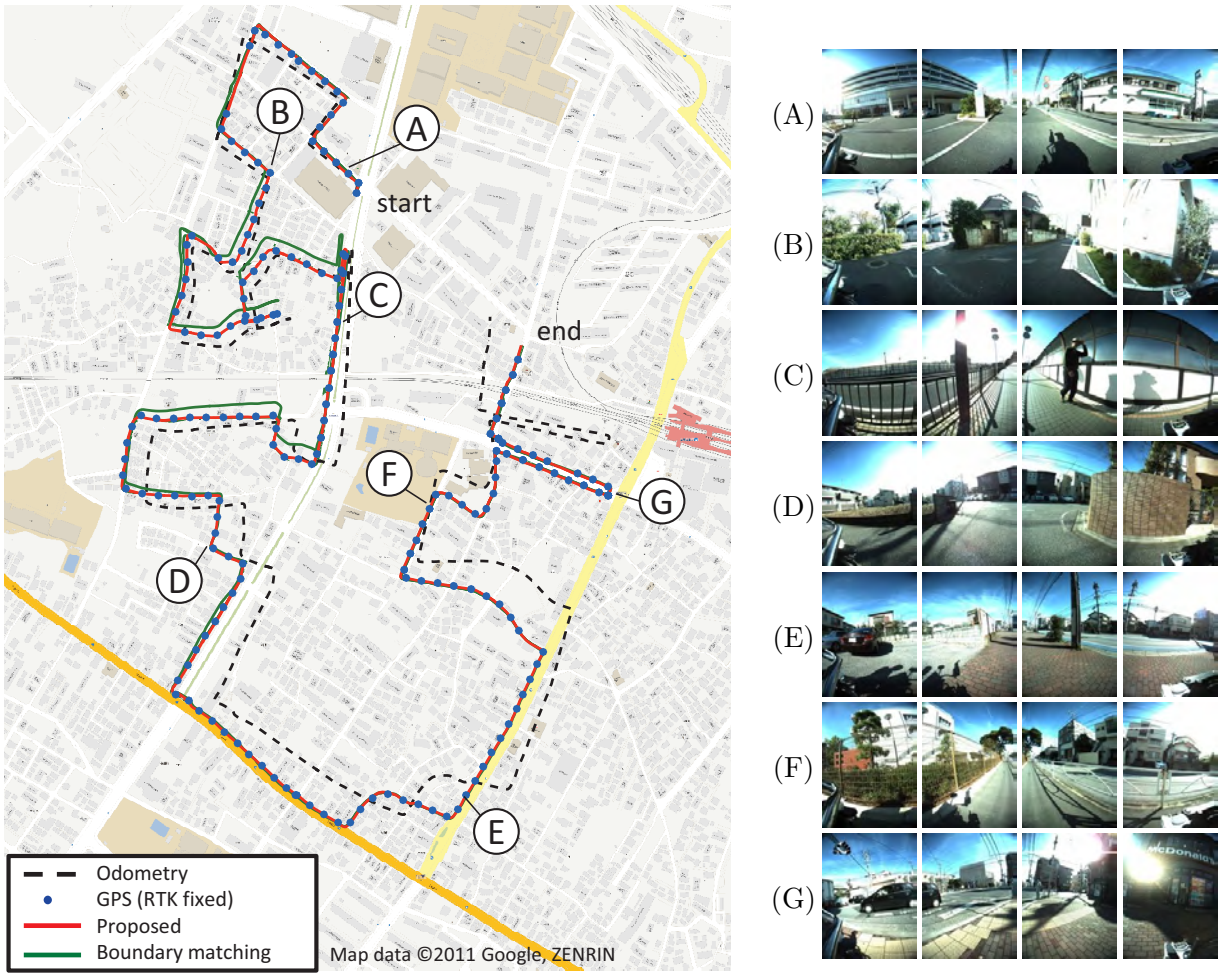


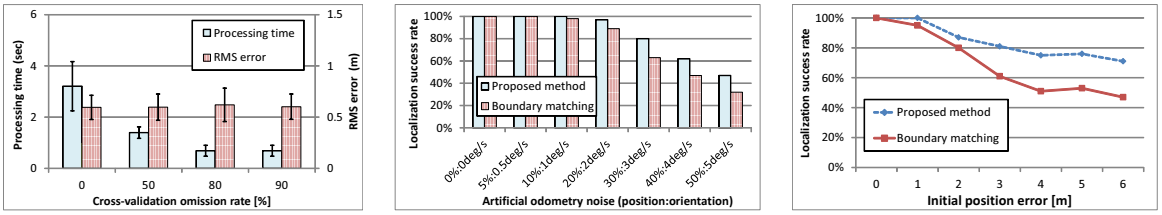
Figure 12. Position tracking result for dataset D (4.5 km, 3,514 s). The proposed method took 871 sec (excluding feature extraction time) to estimate the trajectory.

Table 5. Mean RMS error (in meters) over 100 runs of the position tracking experiments.

	proposed method	boundary matching
dataset C	0.60 ± 0.13	1.2 ± 0.78
dataset D	1.33 ± 0.10	9.58 ± 20.7

(a) (b)

Figure 13. Sample map errors. (a) Satellite image with road boundary line overlay from (b) street map. A notable difference in road shape can be seen at the intersection.



(a) (b) (c)

Figure 14. (a) Comparison of update time and localization errors for different cross-validation omission rates (the time for generating local grid maps is excluded). The error bars indicate standard deviations. (b) Comparison of localization success rate using distorted odometry logs. (c) Comparison of localization success rate with initial position errors.

[6] Brubaker MA, Geiger A, Urtasun R. Lost! leveraging the crowd for probabilistic visual self-localization. In: Proceedings of IEEE Computer Society Conference on Vision & Pattern Recognition (CVPR). Portland, Oregon, USA. 2013. p. 3057–3064.

[7] Floros G, van der Zander B, Leibe B. OpenStreetSLAM: Global vehicle localization using OpenStreetMaps. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Karlsruhe, Germany. 2013. p. 1054–1059.

[8] Tao Z, Bonnifait P, Fremont V, Ibanez-Guzman J. Mapping and localization using gps, lane markings and proprioceptive sensors. In: Proceedings of the IEEE International Conference on Intelligent Robots & Systems (IROS). Tokyo, Japan. 2013. p. 406–412.

[9] Morales Y, Tsubouchi T, Yuta S. Vehicle 3D localization in mountainous woodland environments. In: Proceedings of the IEEE International Conference on Intelligent Robots & Systems (IROS). St. Louis, Missouri, USA. 2009. p. 3588–3594.

[10] Hentschel M, Wagner B. Autonomous robot navigation based on OpenStreetMap geodata. In: Proceedings of the IEEE International Conference on Intelligent Transportation Systems (ITSC). Madeira Island, Portugal. 2010. p. 1645–1650.

[11] Irie K, Tomono M. Localization and road boundary recognition in urban environments using digital street maps. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). St. Paul, Minnesota, USA. 2012. p. 4493–4499.

[12] Pluim JPW, Maintz JBA, Viergever MA. Mutual information based registration of medical images: a survey. IEEE Transactions on Medical Imaging. 2003;22(8):986–1004.

[13] Alempijevic A, Kodagoda S, Dissanayake G. Cross-modal localization through mutual information. In: Proceedings of the IEEE International Conference on Intelligent Robots & Systems (IROS). St. Louis, Missouri, USA. 2009. p. 5597–5602.

[14] Pandey G, McBride JR, Savarese S, Eustice RM. Toward mutual information based place recognition. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Hong Kong, China. 2014. p. 3185–3192.

[15] Irie K, Sugiyama M, Tomono M. A dependence maximization approach towards street map-based localization. In: Proceedings of the IEEE International Conference on Intelligent Robots & Systems (IROS). Hamburg, Germany. 2015. p. 3721–3728.

[16] Collignon A, Maes F, Delaere D, Vandermeulen D, Suetens P, Marchal G. Automated multimodality image registration using information theory. In: Proceedings of 14th International Conference on Information Processing in Medical Imaging (IPMI). Ile de Berder, France. 1995. p. 263–274.

[17] Sugiyama M. Machine learning with squared-loss mutual information. Entropy. 2013;15(1):80–112.

[18] Sakai T, Sugiyama M. Computationally efficient estimation of squared-loss mutual information with multiplicative kernel models. IEICE Transactions on Information and Systems. 2014;E97-D(4):968–971.

[19] Nguyen X, Wainwright MJ, Jordan MI. Estimating divergence functionals and the likelihood ratio by convex risk minimization. IEEE Transactions on Information Theory. 2010;56(11):5847–5861.

[20] Sugiyama M, Suzuki T, Nakajima S, Kashima H, von Bünau P, Kawanabe M. Direct importance estimation for covariate shift adaptation. Annals of the Institute of Statistical Mathematics. 2008; 60(4):699–746.

[21] Ali SM, Silvey SD. A general class of coefficients of divergence of one distribution from another. Journal of the Royal Statistical Society, Series B. 1966;28(1):131–142.

[22] Csiszár I. Information-type measures of difference of probability distributions and indirect observation. Studia Scientiarum Mathematicarum Hungarica. 1967;2:229–318.

[23] Leonard J, Durrant-Whyte H. Mobile robot localization by tracking geometric beacons. IEEE Transactions on Robotics and Automation. 1991;7(3):376–382.

[24] Thrun S, Fox D, Burgard W, Dellaert F. Robust Monte Carlo localization for mobile robots. Artificial Intelligence. 2000;128(1-2):99–141.

[25] Haklay MM, Weber P. OpenStreetMap: User-generated street maps. IEEE Pervasive Computing. 2008;7(4):12–18.

[26] Shaw JR. QuickFill: An efficient flood fill algorithm. 2004 [cited 2016 May 7]. Available from: `http://www.codeproject.com/Articles/6017/QuickFill-An-efficient-flood-fill-algorithm`.

[27] Kong H, Audibert J, Ponce J. General road detection from a single image. IEEE Transactions on Image Processing. 2010;19(8):2211–2220.

[28] Sainui J, Sugiyama M. Direct approximation of quadratic mutual information and its application to dependence-maximization clustering. IEICE Transactions. 2013;96-D(10):2282–2285.

[29] Tangkaratt V, Xie N, Sugiyama M. Conditional density estimation with dimensionality reduction via squared-loss conditional entropy minimization. Neural Computation. 2015;27(1):228–254.
[30] Suzuki T, Sugiyama M, Tanaka T. Mutual information approximation via maximum likelihood estimation of density ratio. In: Proceedings of the IEEE International Symposium on Information Theory (ISIT). Seoul, Korea. 2009. p. 463–467.
[31] Levinson J, Thrun S. Automatic online calibration of cameras and lasers. In: Proceedings of Robotics: Science and Systems. Berlin, Germany. 2013 June.
[32] Matsushita Y, Miura J. On-line road boundary modeling with multiple sensory features, flexible road model, and particle filter. Robotics and Autonomous Systems. 2011;59(5):274–284.