Advanced Robotics, Vol. 21, No. 11, pp. 1281–1304 (2007)

Also available online - www.brill.nl/ar

Full paper

Multi-hypothesis localization with a rough map using multiple visual features for outdoor navigation

JOOSEOP YUN * and JUN MIURA

Department of Mechanical Engineering, Osaka University, Suita, Osaka 565-0871, Japan

Received 1 December 2006; accepted 5 January 2007

Abstract—We describe a method of mobile robot localization based on a rough map using stereo vision, which uses multiple visual features to detect and segment the buildings in the robot's field of view. The rough map is an inaccurate map with large uncertainties in the shapes, dimensions and locations of objects so that it can be built easily. The robot fuses odometry and vision information using extended Kalman filters to update the robot pose and the associated uncertainty based on the recognition of buildings in the map. We use a multi-hypothesis Kalman filter to generate and track Gaussian pose hypotheses. An experimental result shows the feasibility of our localization method in an outdoor environment.

Keywords: Outdoor mobile robot; vision-based localization; rough map; multiple hypotheses.

1. INTRODUCTION

This paper presents an approach to determining the pose (position and orientation) of a mobile robot in an urban area using a set of stereo image pairs. Understanding the surrounding scene and identifying man-made structures are important tasks for the localization of a mobile robot in outdoor environments. For outdoor robot localization, many approaches using odometry, beacons or GPS have been proposed and realized before. Since the pose errors of odometry accumulate without bounds and cannot be corrected without other external sources, odometry is seldom used alone; many other approaches are often accompanied with it, such as GPS-based approaches. While GPS can provide more accurate pose information in open spaces, GPS signals are susceptible to various forms of interference and can be quite unreliable in urban areas [1, 2].

Computer vision can provide both accurate localization and robustness against these environmental influences (see Ref. [3] for a survey). Vision-based approaches

*To whom correspondence should be addressed. E-mail: jsyun@cv.mech.eng.osaka-u.ac.jp

are attractive because they are self-contained in the sense that they require no external infrastructures such as beacons or satellites. The knowledge of having buildings in the environment allows us to exploit their typical characteristics. Horizontal or vertical principal directions and abundant parallel or orthogonal relationships between lines and surfaces are valid assumptions in many man-made environments. The vision system used in this paper attempts to capture these highly structured configurations in the buildings.

1.1. Our approach

In this paper, we will guide the robot with a rough map which represents an environment as a set of two-dimensional (2-D) line segments and can thus be built easily. The map approximates the outlines of buildings except detailed features to be used as landmarks. We propose a method to robustly estimate the robot pose in the map using multiple visual features: low-contrast regions, non-vertical borders, vertical borders and disparity regions. Low-contrast regions include the sidewalls of buildings and the sky in outdoor scenes. Non-vertical and vertical borders are detected from the building structures such as windows, doors, corners, roof, etc. The disparity regions are extracted for matching with the walls of buildings. Multiple visual features are matched to the given map and the results are integrated into the odometry for the estimation of robot pose using extended Kalman filters (EKFs).

In most vision-based localization systems, a key issue is often the data association problem of matching an image taken at arbitrary robot pose relative to a given map. In order to address this problem, we need to first extract a set of features from the sensor readings and identify the corresponding features in the given map usually by some form of constrained search. Once such a correspondence is established, the robot pose can be estimated with reduced uncertainty. There are, however, various errors, such as the noisy sensors and the features found in the image, as well as some uncertainties of roughness on the map in our case. Since they introduce uncertainty in both the landmark pose and the estimation of the robot pose, the association problem is not so easy. The problem is further complicated by unreliable feature extraction and low feature discriminance likely to produce false matches.

These problems motivate the development of a method that can make delayed decision, i.e., a multi-hypothesis approach. The approach allows maintaining when and where to place pose hypotheses, as much as necessary and as few as possible. This property is provided by using a constraint-based search in an interpretation tree. This tree is spanned by all possible local-to-global data associations, given a local map of observed features and a global map of model features [4, 5].

In our work, we explore a localization problem using a rough map in real outdoor environments. To solve this localization problem, we use a novel combination of an efficient map-matching scheme and a multi-hypothesis technique based on multiple visual features. For efficient map matching, we use the ordering and the priority constraints of multiple visual features extracted robustly using stereo and low-contrast region. In addition, hypothesis generation is combined with the

EKF framework to perform the multi-hypothesis localization, followed by heuristic hypothesis management techniques. As far as we know, there are no works that previously explored this kind of localization problem.

1.2. Related work

The problem of vision-based pose estimation has been studied before. Optical flow and feature tracking have both been used for ego-motion estimation, but unavoidably exhibited drift [6]. Yagi *et al.* estimated the azimuth of edges extracted from a conic image sensor to refine robot odometry estimates [7].

Recent research in vision-based localization in urban environments focuses on the recognition and matching of building facades. Georgiev and Allen [8] have used vision-based techniques to supplement GPS and odometry. Their system requires detailed geometric models and they have only localized views in the vicinity of only a single building. Johansson and Cipolla [9] determined the relative pose of a camera by computing the transformation required to match a rectified view of the facade from a single image. Reitmayr and Drummond [10] presented a model-based hybrid tracking system for outdoor augmented reality. The system employs a three-dimensional (3-D) model capturing the overall shape of buildings as large planar surfaces with highly detailed textures. In these systems, creating such detailed models for large outdoor environments becomes a troublesome task.

In addition, since the moving range of outdoor mobile robots is usually much wider and more complicated than that of indoor ones, a promising approach is thus the two-phase method [11, 12]. In the learning phase, the robot first selects natural features of points, lines or regions from observed images and registers them as landmarks on the map. In the execution phase, the robot pose is estimated by matching the observed features to the map. The navigation, however, may be unstable when the robot's viewpoints and the illumination conditions are changed from the learning phase, specifically in the view-based approach. It also needs much time and effort to guide the robot in outdoor environments for extensive training, and requires closely following previously traversed paths.

Since it is hard to build the exact map of outdoor environments, using inaccurate maps is another easy method of giving the environmental information to the robot. Inaccurate maps may include hand-drawn or topological–geometrical maps, where the relative poses among object models can be uncertain. A hand-drawn map is an interface for sketch-based navigation [13] and a topological–geometrical map is a hybrid map for navigation in large-scale environments [14]. A hand-drawn map is, however, hard to use for navigation tasks, because it has no metric information. For a topological–geometrical map, it is costly to build and update the metric local models for object recognition in outdoor environments.

1.3. Paper outline

The rest of this paper is organized as follows. Section 2 shows how to extract the multiple visual features from a pair of stereo images. Section 3 presents a method for representing the rough map. In Section 4, we describe matching processes of the multiple visual features extracted in Section 2. In Section 5, we depict localization algorithms using EKF for combining the matching results. Next, multi-hypothesis localization is represented in Section 6 using the multi-feature EKF-based localization method depicted in Section 5. Then, we present an experiment conducted in an outdoor environment of our university campus in Section 7. From the experimental result, we show the feasibility of the proposed method. Finally, in Section 8, we conclude this study and address some future works.

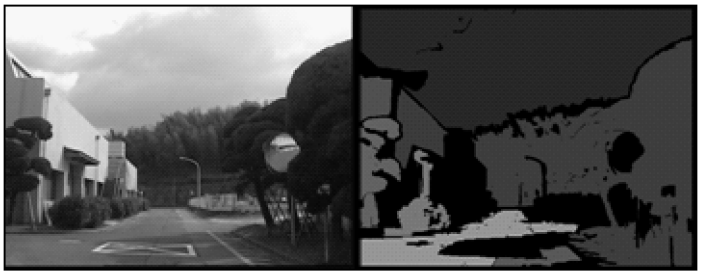
2. FEATURE DETECTION

In this paper, we shall consider self-localization by means of only vision and odometry. We are interested in navigation around urban environments such as our university campus. Since views of trees, cars and bicycles, however, differ from time to time, we use multiple visual features observed from buildings by using stereo vision with an angle of elevation of more than 10°. Their multiple visual features are relatively large and static as landmarks to be used for localization as follows: low-contrast regions for identifying non-vertical and vertical borders, non-vertical borders for the vanishing points to calculate the wall directions of buildings, vertical borders corresponding to the corners of buildings and disparity regions for matching with the walls of buildings.

2.1. Low-contrast regions

Many real-world scenes contain regions of low contrast. Typical regions include the sidewalls of buildings and the sky in outdoor scenes. Such low-contrast regions are very hard to estimate in terms of intensity-based depth because they lack any distinctive texture. We can, however, exploit the existence of low-contrast regions instead of their limited texture for detecting multiple visual features. In this case, the multiple visual features should line up in the intermediate regions of different low-contrast regions. The low-contrast region processing involves segmenting each image into no overlapping regions based on intensity [15]. A simple linking algorithm starts at every pixel in the image and recursively grows out regions of similar intensity. An input image at the first frame and its result of low-contrast region processing are shown in Fig. 1.

We recognize the sky regions (the top region shown in Fig. 1b) using an algorithm developed in our laboratory from the extracted low-contrast regions; from *a priori* knowledge of position, color and shape of the sky in the image, we define the following conditions that a region is recognized as the sky:



(a) (b)

Figure 1. An input image (a) and its low-contrast regions (b).

- It touches the top boundary of an image.
- Its average intensity is larger than a certain threshold.
- Its size is larger than a certain threshold.
- The width of its upper part is larger than that of its lower part.

2.2. Border features

We extract couples of non-vertical and vertical line segments fitted to the edge pixels, and attain coupled borders from the line segments near enough to each other both in the image and the disparity space. When identifying the borders, we consider the height of their end-points above ground using a segment-based stereo algorithm [16]. A non-vertical border can be coupled with up to two vertical borders which may be on its left and right sides, respectively. A vertical border can also be coupled with up to two non-vertical borders which may be on its left and right sides, respectively. Each end of a non-vertical border should be coupled with the upside end of its coupled vertical border. Isolated borders are then detected when they are adjoining to the sky regions recognized above.

2.2.1. Non-vertical borders. Non-vertical borders can be extracted from the building structures, and can provide the relative orientation between the robot and the building. What is necessary for estimating the relative orientation in this case is a vanishing point (VP). The VPs of non-vertical borders exist on the horizon in the image. We can then estimate the angles between the image plane and the lines from the camera center to VPs (VP1 or VP2 in Fig. 2). The lines are parallel to the respective directions of visible walls with respect to X_w, x-axis of the world reference coordinate system.

From Fig. 2, we can deduce that there exists the following relation among the robot's orientation, θ_p with a range of $[-\pi, \pi]$, the inward direction of a wall, θ_b with a range of $[-\pi, \pi]$, and the angle from a vanishing point, θ_{vp} with a range of $[-\pi/2, \pi/2]$:

$$\theta_p - \theta_b + \theta_{vp} = 0. \tag{1}$$

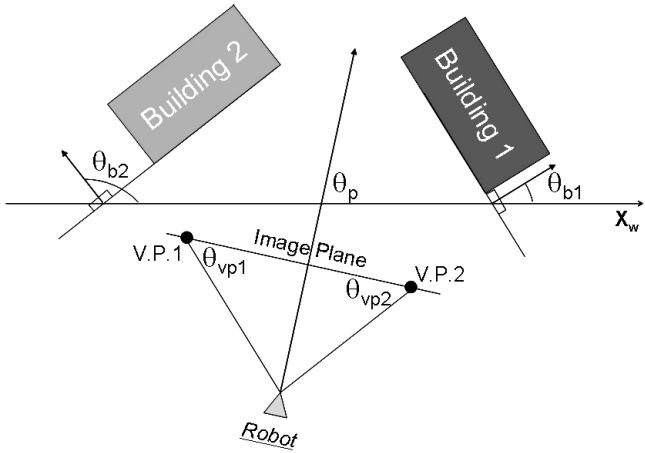


Figure 2. Estimation of a robot orientation from VPs.

2.2.2. Vertical borders. Vertical borders correspond to the corners of buildings. Coupled vertical borders are detected as couples with non-vertical borders described in the previous section. Isolated vertical borders are detected also considering their heights when they are adjoining to the sky regions. Figure 3a shows the detection results of non-vertical and vertical borders in the right image at the first stereo frame. The selected borders must be in the intermediate regions of different low-contrast regions as shown in Fig. 3b because there are many tall trees and streetlights resulting in similar features to the borders of building in outdoor environments. The black regions are not low-contrast regions.

2.3. Disparity regions

We use area-based stereo matching in order to extract a disparity image [16]. The depth is less sensitive to changes of illumination than the previous visual features provided by using a single-intensity image [17]. In this study, disparity regions are detected as connected regions with a disparity in the disparity image.

One of the advantages of stereo vision is that it provides a more informative 2-D depth map. With *a priori* knowledge about the minimum height of buildings in an urban environment, we can extract the regions of buildings from the disparity regions using the height calculated from the disparity image. The histograms of gradient orientations of edge pixels in the regions weighted by their gradient magnitudes seem to be well-suited for discrimination between urban structures and natural environments. Intuitively, the histograms for the building regions tend to be unimodal or bimodal and most of whose peaks tend to be separated by approximately right angles to each other [11, 18]. However, the histograms at the tree regions, for example, tend to be more uniformly distributed and the peak values have lower maximum values than those of the building regions. The textured boxes in Fig. 4a show the resultant building regions detected from the right image of a

Figure 3. Detected borders (a) and their relationships to the low-contrast regions (b).

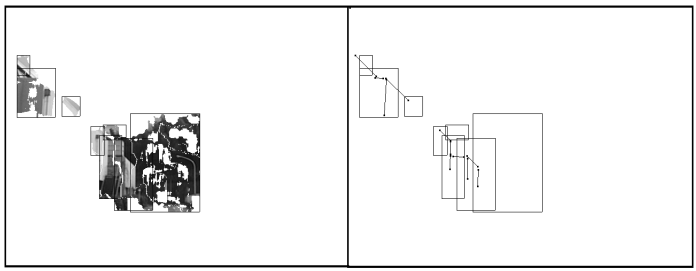
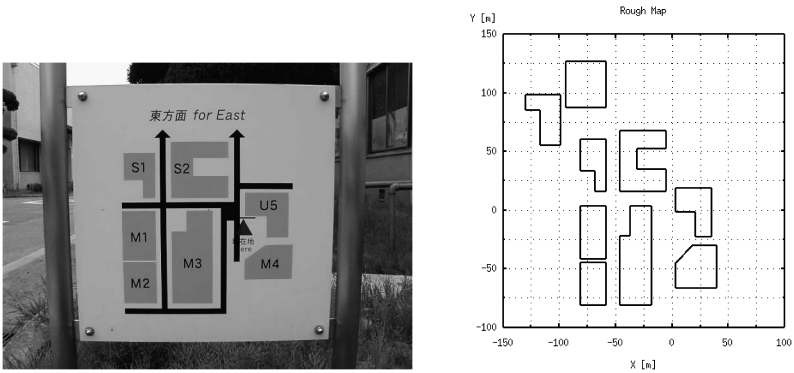


Figure 4. Detected building regions (a) and all multiple visual features (b).

pair of input stereo images shown in Fig. 1. The result of all multiple visual features detected so far is shown in Fig. 4b, where the rectangles represent the recognized building regions.

3. ROUGH MAP

Much of the research efforts in robot navigation have been directed towards object representation on the map and object recognition using the map. Although an accurate map provides accurate and efficient localization, it needs a lot of cost to build and update [8, 14]. A solution to this problem would be to allow a map to be defined roughly since a rough map is much easier to build [13]. A rough map in this paper consists of two parts—one is the objects for buildings and the other is the paths for the roads among them. The rough map is defined as a 2-D segment-based map that contains rough metric information about the poses and dimensions of buildings themselves, and also their relative distances and directions in the environment. We assume that the buildings have planar walls, and that these planes have both horizontal and vertical edges. This is often the case for buildings as they have windows and doors. We also assume a flat polygon on the top of a building as a roof since roof details on a tall building cannot be seen from the ground level.



(a) (b)

Figure 5. A guide map of our university campus (a) and an example of rough map (b).

The map may include the robot's current pose as an initial pose and the robot's goal pose on the map. The approximate outlines of buildings are represented in the map and thus used for recognizing the buildings present in an environment during the navigation. We can also arrange a path of a robot on the map. Figure 5 shows a guide map for visitors to our university campus and an example of a rough map built from the map. A human can use this kind of map to navigate efficiently, but it is difficult for the robot to use it, because of the deficiency of accurate metric and geometric information.

The characteristics of the rough map can be summarized as follows. The exact model of map uncertainty is unknown. The uncertainty may be not uniform across the map. The geometric details such as exact outlines, exact dimensions and exact poses of buildings are not available. The map also lacks information about exact models of the building structures.

Relative poses between landmarks in a rough map are allowed to be uncertain. The uncertainty of a rough map might cause the robot pose to be inconsistent if it is represented in the global coordinate system of reference. To address this problem, we represent the robot pose in a local coordinate system attached to a landmark which the robot has recognized recently. When the robot finds a new landmark, the robot changes the local coordinate system from the old landmark to the new one with coordinate transformation of its pose based on the relative pose between the old and new landmarks. We refer to the landmark as a local origin. As the robot moves, it changes the local origin. More specifically, we define the robot pose as a pair of a local origin and the pose in a local coordinate system attached to the local origin. Landmarks in the building with the local origin would have smaller positional uncertainty in the local coordinate system than in the global one, thereby becoming easier to recognize.

To handle the uncertainty, the relative pose between two local coordinate systems is defined using a Gaussian random variable. Let $\mathbf{d}_{jk} = (x_{jk}, y_{jk}, \theta_{jk})^{\mathrm{T}}$ be the relative pose of local coordinate system L_k with respect to local coordinate

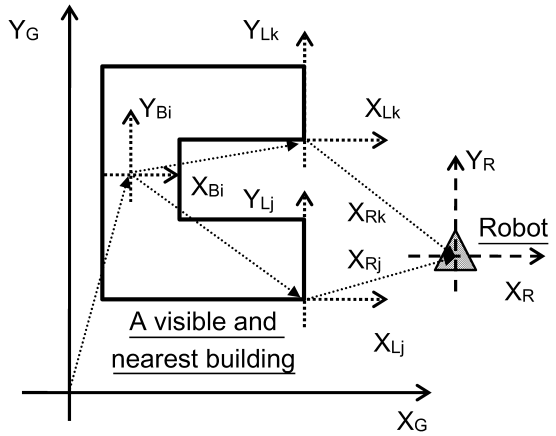


Figure 6. A relationship among global (G), building (B), local (L) and robot (R) coordinate systems.

system L_j. $\mathbf{d}_{jk}$ is a random variable having a Gaussian distribution. Thus, the coordinate transformation of robot pose $\mathbf{X}_{\mathrm{R}} = (x, y, \theta)^{\mathrm{T}}$ from local coordinate system L_j to L_k can be calculated as follows:

$$\mathbf{X}_{\mathrm{R}k} = \mathbf{T}^{-1}(\theta_{jk})(\mathbf{X}_{\mathrm{R}j} - \mathbf{d}_{jk}), \tag{2}$$

$$\mathbf{T}(\theta_{jk}) = \begin{bmatrix} \cos(\theta_{jk}) & -\sin(\theta_{jk}) & 0 \\ \sin(\theta_{jk}) & \cos(\theta_{jk}) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

where $\mathbf{X}_{\mathrm{R}j}$ and $\mathbf{X}_{\mathrm{R}k}$ are the robot poses with respect to the local coordinate systems L_j and L_k, respectively, as shown in Fig. 6.

Figure 6 shows representative transformations among global (X_{G}–Y_{G}), building (X_{B}–Y_{B}), local (X_{L}–Y_{L}) and robot (X_{R}–Y_{R}) coordinate systems. A building coordinate system has its origin at the center of a building and parallel to the principal directions of a building. A local coordinate system is selected such that its origin is at a visible and nearest corner of a certain building from the robot. The local coordinate system is also parallel to the building coordinate system or one of its axes placed on the visible and nearest wall of the building. The uncertainty of the local coordinate system is calculated from the uncertainty of the building coordinate system with respect to the global coordinate system.

We approximate the buildings present in an environment to polygonal objects on the map, and compute the uncertainties of their poses and dimensions for estimating the robot pose. Figure 7 shows examples of modeling the uncertainties of a rough map in the global and local coordinate systems, where the buildings are drawn with their mean poses and mean dimensions. The global map uncertainty is roughly assumed with respect to the world origin of the map, and transformed to the local uncertainty model with respect to its local origin by a coordinate transformation using (2) and (3). The local map has greater uncertainty than the global map except the walls and the corners on the axes of the local coordinate system. The coordinates

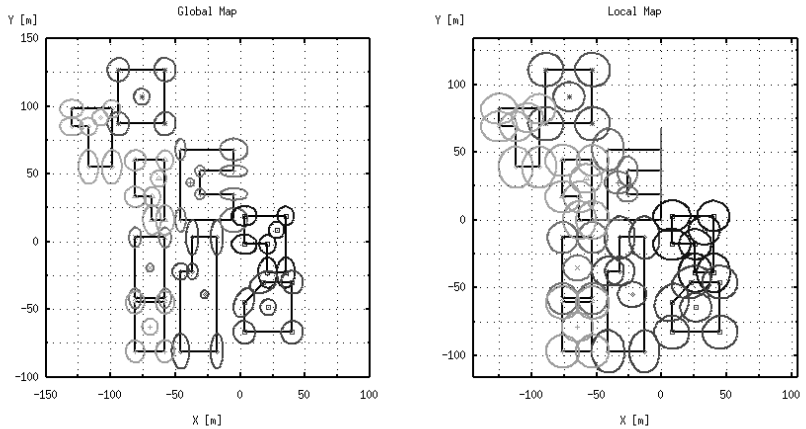


Figure 7. The uncertainty model of the global map (a) and a local map at the first frame (b).

of (0, 0) on both maps are their respective origins and the local origin has no error. Assuming Gaussian error models, the ellipses in each map mark 3σ uncertainty regions of the centers and corners of buildings by their relative pose errors with respect to the respective origin.

4. MAP MATCHING

One of the most important and challenging aspects of map-based localization is map matching. In the matching process, we use the depth information: a segment-based stereo algorithm for the borders and an area-based one for extracting the disparity regions corresponding to the walls of buildings. The multiple visual features should be used in order to match the sensory data to the environment map reliably.

When visual pose estimation is attempted, an approximate estimate of the pose is available from the odometry. This estimate is used to search the map for the most appropriate buildings for visual localization. The buildings within a certain distance range from the robot are selected by scanning through the given map. Only the features of buildings that are viewable under the orientation uncertainty of the robot pose are considered. We also eliminate the features of buildings that are visible at a too low an angle to produce a stable match with the image.

Given the candidate features of buildings that successfully passed this selection process, the robot matches the detected features to those candidates with the Mahalanobis distance criterion using the depth. The resolution of the depth data is, however, not constant; the further from the stereo camera, the larger the error of the depth. In order to simplify the computation, we use the disparity space which keeps the error constant [19].

At this point, we are ready to match-up the multiple visual features with the map. Instead of generating and testing all of the data associations using the multiple visual

features simultaneously, however, we build the data associations according to the priority of non-vertical borders, vertical borders and disparity regions in this order (see Section 6). Since the multiple visual features can reduce more effectively the uncertainty of robot orientation using this priority constraint, we can, thus, narrow the map candidates for the data association more efficiently. The data association of non-vertical borders with outlines of buildings is possible when satisfying the following two criteria:

(i) The Mahalanobis distance d_{vp} between their VPs should be close enough to each other.

$$d_{\text{vp}} = (x_{\text{vp}} - X_{\text{vp}})(\sigma^2_{x_{\text{vp}}} + \sigma^2_{X_{\text{vp}}})^{-1}(x_{\text{vp}} - X_{\text{vp}}), \tag{4}$$

where x_{vp} and X_{vp} are the VPs of a non-vertical border and a building outline, and $\sigma^2_{x_{\text{vp}}}$ and $\sigma^2_{X_{\text{vp}}}$ are their uncertainties, respectively. If d_{vp} is small, two VPs are considered to be consistent.

(ii) The Mahalanobis distance d_{rt} between their parameters in the disparity space should be small enough, because there are many buildings having parallel walls in urban environments.

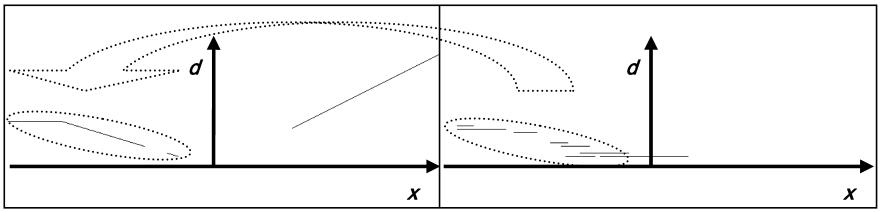
$$d_{rt} = (\mathbf{x}_{rt} - \mathbf{X}_{rt})^{\text{T}}(\mathbf{\Sigma}_{\mathbf{x}_{rt}} + \mathbf{\Sigma}_{\mathbf{X}_{rt}})^{-1}(\mathbf{x}_{rt} - \mathbf{X}_{rt}), \tag{5}$$

where $\mathbf{x}_{rt}$ and $\mathbf{X}_{rt}$ are the Hough parameters (r, t) in the disparity space of a non-vertical border and a building outline, and $\mathbf{\Sigma}_{\mathbf{x}_{rt}}$ and $\mathbf{\Sigma}_{\mathbf{X}_{rt}}$ are their error covariance matrices, respectively. The judgment of data association depends on the value of the threshold d_{thresh} for each distance.

The vertical borders are associated with the corners of buildings using the Mahalanobis distance criterion in the disparity space. The coupled borders are associated with the coupled features of building outlines and corners. We also consider the ordering constraint in their associated building.

In the case of disparity data, the disparity regions recognized as building regions are associated with the walls of buildings using the Mahalanobis distance criterion. Neighboring disparity data that correspond to a wall are grouped from the result of data association. By using a plane fitting procedure, then, a plane is used to fit each group. The results are the plane models as the environment representation. Figure 8 shows the quantized segments (three left segments and a right segment) of four visible walls of two buildings on the map Fig. 8a and a set of points of the observed disparity data Fig. 8b in the disparity space (x, y, d) at the first frame. The observed disparity regions are matched with three visible walls of same building as indicated by an arrow on the x–d plane of the disparity space. The remaining y-coordinate of the disparity space corresponds to the upside height of the disparity region, which was used for recognizing the building regions among the extracted the disparity regions.

Figure 9 shows a building in the map with visible walls and corners pictured as thick segments, and black dots and circles, respectively, on the left side. Coupled



(a) (b)

Figure 8. Predicted map data (a) and observed data (b) in the disparity space at the first frame.

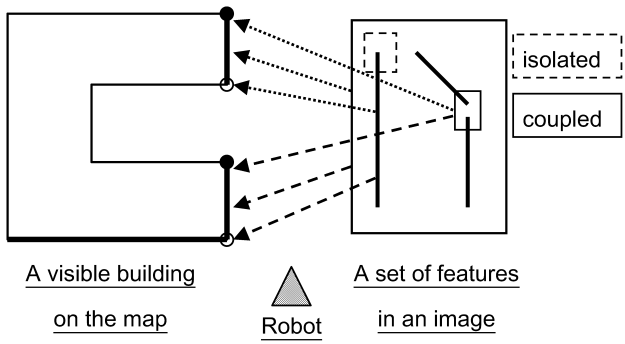


Figure 9. Correspondences between a set of multiple visual features and a visible building.

borders, an isolated vertical border and a rectangular disparity region bounding them are shown on the right side of Fig. 9. A set of arrows of the same style depicts that coupled borders an isolated vertical border and a disparity region are consistently matched to the outline, the corners and the wall of same building, respectively. Considering the coupled borders and their ordering constraints, we can generate consistent sets of data associations of coupled non-vertical and vertical borders, an isolated vertical border and a disparity region.

5. MULTI-FEATURE EKF-BASED LOCALIZATION

At this stage, a set of matched features in the image and the buildings in the map are available, and the task is to estimate the pose of the robot. The map-matching method described in the previous section provides the data associations for a correction of the estimated pose of the robot that must be integrated with odometry. We use the EKFs according to multiple visual features for the estimation of the robot pose from the results of the map matching.

5.1. Kalman filter framework

A localization cycle in this framework mainly consists of three stages: state and measurement prediction, observation, and update according to respective multiple visual features detailed below.

- *Prediction*. The state prediction $X_{(k+1|k)}$ and its associated covariance $\Sigma_{X(k+1|k)}$ is determined from odometry based on the previous state $X_{(k|k)}$ and $\Sigma_{X(k|k)}$. The modeled features in the map, M, get transformed into the observation frame. The measurement prediction $z_{(k+1)} = H(X_{(k+1|k)}, M)$, where H is the nonlinear measurement model. Error propagation is done by a first-order approximation which requires the Jacobian J_X of H with respect to the state prediction $X_{(k+1|k)}$.
- *Observation*. The parameters of features constitute the vector of observation $Z_{(k+1)}$. Their associated covariance estimates constitute the observation covariance matrix $R_{(k+1)}$. Successfully matched observations and predictions yield the innovations:

$$V_{(k+1)} = Z_{(k+1)} - z_{(k+1)}, \tag{6}$$

and their innovation covariance:

$$S_{(k+1)} = J_X \Sigma_{X(k+1|k)} J_X^{\mathrm{T}} + R_{(k+1)}. \tag{7}$$

- *Update*. Finally, with the filter equations:

$$W_{(k+1)} = \Sigma_{X(k+1|k)} J_X^{\mathrm{T}} S_{(k+1)}^{-1}, \tag{8}$$

$$X_{(k+1|k+1)} = X_{(k+1|k)} + W_{(k+1)} V_{(k+1)}, \tag{9}$$

$$\Sigma_{X(k+1|k+1)} = \Sigma_{X(k+1|k)} - W_{(k+1)} S_{(k+1)} W_{(k+1)}^{\mathrm{T}}, \tag{10}$$

the posterior estimates of the robot pose and associated covariance are computed.

5.2. Pose update by the robot motion uncertainty

The state of robot, $X = (x, y, \theta)^{\mathrm{T}}$, consists of the 2-D robot position (x, y) which corresponds to the position of the camera pair, and the orientation of the robot, θ. Figure 10 shows the motion model of the robot controlled by input $U = (l, r)^{\mathrm{T}}$, which is the moving distance of the left and the right wheels.

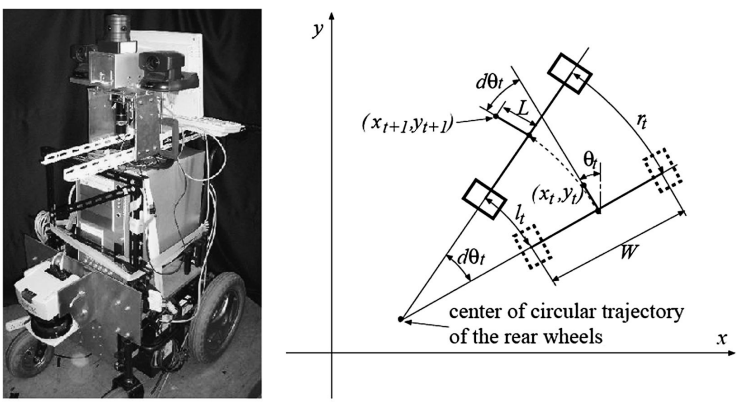


Figure 10. Our four-wheeled mobile robot and its motion model.

The state transition of the robot is expressed by the following nonlinear equation:

$$X_{t+1} = \begin{pmatrix} x_t + \frac{W}{2}\frac{l_t+r_t}{l_t-r_t}\left(\cos\theta_t - \cos\left(\theta_t - \frac{l_t-r_t}{W}\right)\right) + L\left(\sin\theta_t - \sin\left(\theta_t - \frac{l_t-r_t}{W}\right)\right) \\ y_t + \frac{W}{2}\frac{l_t+r_t}{l_t-r_t}\left(\sin\theta_t - \sin\left(\theta_t - \frac{l_t-r_t}{W}\right)\right) - L\left(\cos\theta_t - \cos\left(\theta_t - \frac{l_t-r_t}{W}\right)\right) \\ \theta_t - \frac{l_t-r_t}{W} \end{pmatrix} = F(X_t, U_t), \tag{11}$$

where W is the distance between the two rear wheels, and L is the distance between the robot position and the midpoint of the rear wheels.

Linearizing (11) by the first-order Taylor series expansion around the mean value, $\hat{X}_t$ and $\hat{U}_t$, the covariance matrix of the predicted state error $\Sigma_{X_{t+1}}$, can be obtained by:

$$\begin{aligned} \Sigma_{X_{t+1}} &= E\left[(X_{t+1} - \hat{X}_{t+1})(X_{t+1} - \hat{X}_{t+1})^{\mathrm{T}}\right] \\ &= \frac{\partial F}{\partial X_t}\Sigma_{X_t}\frac{\partial F^{\mathrm{T}}}{\partial X_t} + \frac{\partial F}{\partial U_t}\Sigma_{U_t}\frac{\partial F^{\mathrm{T}}}{\partial U_t}, \end{aligned} \tag{12}$$

where Σ_{U_t} is the covariance matrix of the input U_t. We assume that the error Σ_{U_t} is caused only by the slippage of wheels. We also assume that the error of the left and the right wheels, $\sigma_{l_t}^2$ and $\sigma_{r_t}^2$, are Gaussian and independent of each other. Thus, Σ_{U_t} is expressed by the following diagonal matrix:

$$\Sigma_{U_t} = \begin{pmatrix} \sigma_{l_t}^2 & 0 \\ 0 & \sigma_{r_t}^2 \end{pmatrix}. \tag{13}$$

$\sigma_{l_t}^2$ and $\sigma_{r_t}^2$ are considered to be proportional to the moving distance, l_t and r_t; we determine the proportional coefficients experimentally.

In this paper, we define the uncertainty region as the so-called 3σ ellipsoid obtained from Σ_{X_t}. The positional uncertainty is represented as an ellipse generated by projecting the ellipsoid on the X–Y plane. The uncertainty of robot orientation is calculated as the marginal distribution of θ.

5.3. Pose update by the non-vertical border

The VP of a non-vertical border matched with the outline of a building allows a robot pose to be partially estimated since it provides information about the robot's orientation only. From (1) in Section 2.2.1, we can directly observe the robot orientation using the inward direction of a wall and the angle from a VP. Thus, the observation is $Z = \theta_{\mathrm{b}} - \theta_{\mathrm{vp}}$ and the prediction $z = \theta_{\mathrm{p}}$ is the robot orientation of last step. The filter setup for this feature as referred to in (7) and (8) is as follows:

$$S = [\,0 \quad 0 \quad 1\,]\Sigma_X[\,0 \quad 0 \quad 1\,]^{\mathrm{T}}, \tag{7}'$$

$$W = \Sigma_X[\,0 \quad 0 \quad 1\,]^{\mathrm{T}}S^{-1}, \tag{8}'$$

where Σ_X is the uncertainty covariance matrix of a robot pose X. The robot pose is generated using these equations by the update stage of the Kalman filter framework depicted in Section 5.1.

5.4. Pose update by the vertical border

The corner of a building also allows the robot pose to be partially estimated since it provides information about the robot's relative position with respect to the corner only. After we found the corner of a building corresponding to the vertical border $Z = (x, d)^{\mathrm{T}}$ in the disparity space, the observation equation can be described like the following equation:

$$\begin{aligned} Z &= H(X, M) + v \\ &= \begin{pmatrix} x \\ d \end{pmatrix} = \begin{pmatrix} f\frac{(m_x - x_{\mathrm{p}})\sin\theta_{\mathrm{p}} - (m_y - y_{\mathrm{p}})\cos\theta_{\mathrm{p}}}{(m_x - x_{\mathrm{p}})\cos\theta_{\mathrm{p}} + (m_y - y_{\mathrm{p}})\sin\theta_{\mathrm{p}}} \\ \frac{fl}{(m_x - x_{\mathrm{p}})\cos\theta_{\mathrm{p}} + (m_y - y_{\mathrm{p}})\sin\theta_{\mathrm{p}}} \end{pmatrix} + v, \end{aligned} \tag{14}$$

where $X = (x_{\mathrm{p}}, y_{\mathrm{p}}, \theta_{\mathrm{p}})^{\mathrm{T}}$ is the robot pose, $M = (m_x, m_y)^{\mathrm{T}}$ is the coordinates of the building corner on the map, f is the focal length of a stereo camera, l is the base length of the stereo vision system and v is the random observation error. The filter setup for this feature as referred to in (7) and (8) is as follows:

$$S = J_X \Sigma_X J_X^{\mathrm{T}} + J_M \Sigma_M J_M^{\mathrm{T}} + \Sigma_v, \tag{7}''$$

$$W = \Sigma_X J_X^{\mathrm{T}} S^{-1}, \tag{8}''$$

where J_X and J_M are the respective Jacobians of H with respect to X and M, and Σ_X, Σ_M and Σ_v are the uncertainty covariance matrices of X, M and v, respectively. The robot pose is generated using these equations by the update stage of the Kalman filter framework depicted in Section 5.1.

5.5. Pose update by the disparity region

The disparity regions corresponding to the wall of a building allow the robot's partial pose to be estimated since a visible wall of a building gives pose information for the robot to be a certain distance from the wall and with a certain orientation alongside the wall. We formulate the wall of a building by $y = A + Bx$ in the map and that transformed into the disparity space by $d = \alpha + \beta x$ with $z = (\alpha, \beta)^{\mathrm{T}}$. The observation equation $Z = (\hat{\alpha}, \hat{\beta})^{\mathrm{T}}$ of the disparity region corresponding to the wall of a building is described as follows:

$$\begin{aligned} Z &= F(X, L) + v \\ &= \begin{pmatrix} \hat{\alpha} \\ \hat{\beta} \end{pmatrix} = \begin{pmatrix} fl\frac{\sin\theta_{\mathrm{p}} - B\cos\theta_{\mathrm{p}}}{A + Bx_{\mathrm{p}} - y_{\mathrm{p}}} \\ -l\frac{\cos\theta_{\mathrm{p}} + B\sin\theta_{\mathrm{p}}}{A + Bx_{\mathrm{p}} - y_{\mathrm{p}}} \end{pmatrix} + v, \end{aligned} \tag{15}$$

where $L = (A, B)^{\mathrm{T}}$ is the map parameter and v the random observation error.

The filter setup for this feature as referred to in (7) and (8) is as follows:

$$S = J_X \Sigma_X J_X^{\mathrm{T}} + J_L \Sigma_L J_L^{\mathrm{T}} + \Sigma_v, \tag{7}'''$$

$$W = \Sigma_X J_X^{\mathrm{T}} S^{-1}, \tag{8}'''$$

where J_X and J_L are the respective Jacobians of F with respect to X and L, and Σ_X, Σ_L and Σ_v are the uncertainty covariance matrices of X, L and v, respectively. The robot pose is refined using these equations by the update stage of the Kalman filter framework depicted in Section 5.1.

6. MULTI-HYPOTHESIS LOCALIZATION

The Kalman filter acts as a pose tracker in this paper. However, a false matching of the observed features to the model features can lead to an irrecoverable lost situation if only a single distribution is maintained. Although the most credible estimation at one time turns out to be totally wrong, a Multiple Hypothesis Localization (MHL) allows alternative pose estimates to be maintained instead of tracking only the most credible hypothesis. The MHL has been widely used to solve global localization problems in which a robot has no knowledge of its initial pose. The global localization, therefore, has to determine its current pose based on past observations of the environment [20]. In our work, however, instead of starting with an empty hypothesis, we start with a highly reliable hypothesis of robot pose. A starting robot pose around the true pose and its uncertainty of random size must be supplied by an operator.

This method can be achieved by explicitly tracking multiple pose hypotheses, via multiple Kalman filters discussed in the previous section, by the priority data associations of multiple visual features. Direct application of all observations to all targets association is, however, not practical as the number of possible hypotheses may be huge with frame steps. This is the reason various heuristics are introduced to keep the algorithm practicable in the method.

A pose hypothesis is represented by a pose estimate with an associated covariance. Given a sensor reading, the data associations which have generated the current recognition are made from the map matching. As previously described, a stereo camera makes it possible to detect multiple visual features: non-vertical borders, vertical borders and disparity regions. The pose hypotheses are generated and updated using the data associations of these features according to the so-called "measurement-update formula" of (8)–(10). They are driven by the odometric information according to the so-called "time-update formula" of (11) and (12). The overall algorithm is summarized in Fig. 11 and the key steps are described below.

6.1. Step 1: hypothesis evolution by robot motion

When the algorithm starts, it takes as input the prior set of pose hypotheses from the previous cycle. Each of the current hypotheses evolves to take into account the uncertainty of robot motion according to the odometry. Then, the local origin of each evolved pose hypothesis is changed when a new one is found. The global map is transformed to a local map with respect to the new local coordinate system and then to the disparity space for map matching. When no features are detected from the

```
Algorithm MHL {

  For each of current hypotheses {
    (1) Hypothesis evolution

    (2) Hypothesis generation using nonvertical borders
          (2)-1 Data association
          (2)-2 Pose hypothesis generation using EKF
          (2)-3 Hypothesis pruning

    (3) Hypothesis generation using vertical borders
          (3)-1 Data association
          (3)-2 Pose hypothesis generation using EKF
          (3)-3 Hypothesis pruning

    (4) Hypothesis refinement using disparity regions
          (4)-1 Data association and clustering
          (4)-2 Pose hypothesis refinement using EKF
          (4)-3 Hypothesis pruning
  }
  (5) Hypothesis merging
}
```

Figure 11. The overall algorithm. Summary of the MHL procedure.

current input image or no detected features are matched with the map, the evolved pose hypotheses become the input set of current pose hypotheses in the next cycle.

6.2. Step 2: hypothesis generation by non-vertical borders

Non-vertical borders are first used for hypothesis generation. For each combination of possible associations between non-vertical borders and building outlines, a new pose hypothesis is generated using EKF. In this hypothesis generation, the robot orientation is mainly adjusted.

When each pose hypothesis violates one of the following four constraints, the hypothesis is considered to be infeasible:

- The pose hypothesis should satisfy the ordering constraint of borders: When visible borders are on a single building, the relative order of the borders is maintained from map predictions.
- Matched borders in each data association must be in the predicted field of view.
- Each association should be possible in the sense of Mahalanobis distance check.
- A generated pose must not be largely far away from the evolved pose in the Mahalanobis distance sense.

Such infeasible pose hypotheses are all pruned.

6.3. Step 3: hypothesis generation by vertical borders

Vertical borders are then used for hypothesis generation. For each pose hypothesis generated in the previous step, a set of combinations of consistent associations

between vertical borders and building corners is generated. A new pose hypothesis is generated for each combination by using EKF. In the case of coupled vertical borders, only the hypotheses generated using the corresponding non-vertical borders are considered. The same pruning process is then applied to the generated pose hypotheses.

6.4. Step 4: hypothesis refinement by disparity regions

For each pose hypothesis generated using non-vertical and/or vertical borders, the disparity data are clustered and matched with the map of the hypothesis (see Fig. 8). The pose hypothesis is then refined using the matching and EKF. The same pruning process is again applied to the refined pose hypotheses.

6.5. Step 5: hypothesis merging

If multiple pose hypotheses have the same local coordinate system and are within a specified range of each other in the Mahalanobis distance sense, they are grouped and merged into a new pose hypothesis. Since the hypotheses in a group are considered to be equally plausible, the pose estimate of the resulting merged hypothesis is the mean of their poses and its estimated covariance is determined to cover all their uncertainty regions. All pose hypotheses after this step constitute the input set of current pose hypotheses in the next cycle.

7. EXPERIMENTAL RESULTS

To study the performance of the localization algorithms described in this paper, we performed a test in an actual outdoor environment of our university campus. In this test, the robot makes a number of moves and ends up near its start position.

Our implementation uses the multiple visual features and the state prediction is made by using only odometry data. The system was tested on about 200×200 m^2 site with nine buildings on variable outlines (refer to Fig. 5). In the test, a sequence of stereo images was obtained by driving the robot using the joystick interface to the steering control system all along the path and back to the starting position. The visual localization routine proposed in this paper was then performed. It used the accumulated error from odometry as an initial guess to determine the visible buildings on the given map and chose their multiple visual features observed for localization. Figure 12 shows a robot path and sampled images of 30 frames counterclockwise used for our experiment, where the numbers in circles indicates the numbers of the frame steps.

Figure 13 shows the results of multi-hypothesis localization on the magnified local map using the multiple visual features at the start position in Fig. 12. The ellipses in Fig. 12 are the estimated 3σ uncertainty regions of the robot positions by matching the respective visual features to the map. A corner of a building linked to the centers

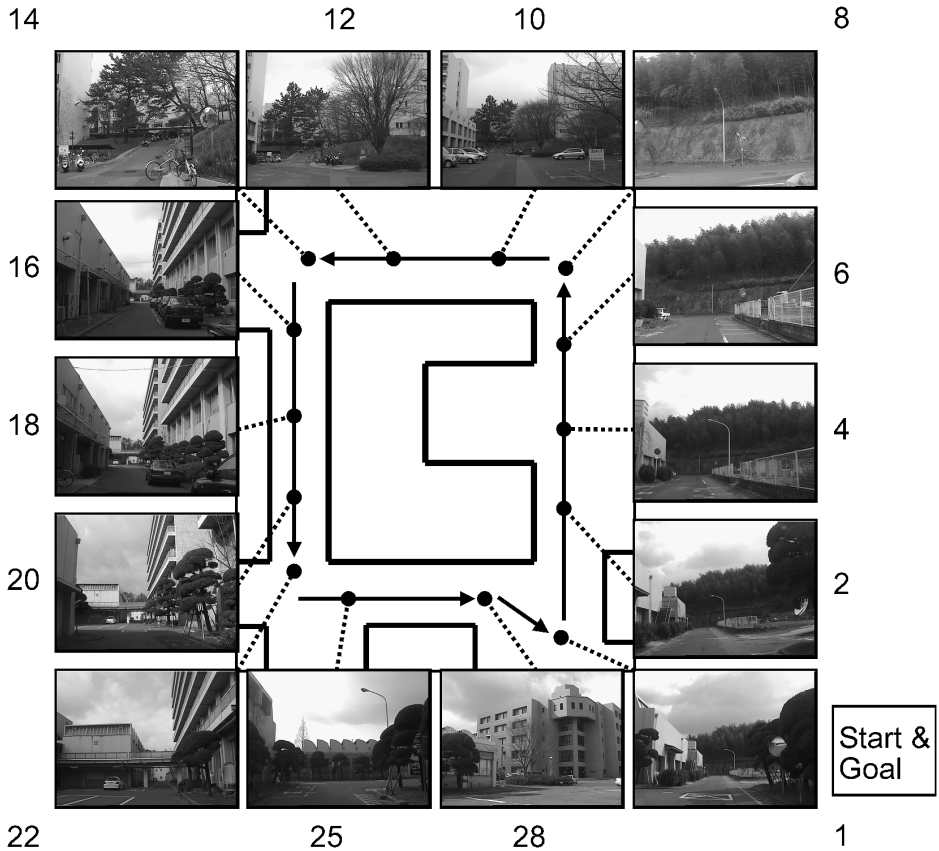


Figure 12. A robot path and sampled images used for an experiment.

of ellipses means the origin in the local map of the pose hypotheses. Multiple pose hypotheses are merged in order to keep the number of hypotheses low.

At the first frame (start position), the robot has a rough knowledge on its pose and observes six non-vertical borders, five vertical borders and eight disparity regions (refer to Fig. 4b). This yields 20, 14 and 13 pose hypotheses for each visual feature, respectively (shown in Fig. 13a–c); only position information from the pose hypotheses is displayed. Figure 13a shows a 3σ uncertainty ellipse of superposed 20 pose hypotheses using non-vertical borders. The reason for the superposition is because we assumed no correlation between the position and orientation of the robot at the start position. The VP of a non-vertical border provides information about the robot's orientation only. Thirteen pose hypotheses generated using disparity data in Fig. 13c are merged to four hypotheses in Fig. 13d by the constraint of the same local coordinate system and the threshold of Mahalanobis distance. The large circle drawn in Fig. 13d denotes the uncertainty region of an initial robot position with respect to a local coordinate system. The uncertainty circle consists of the global

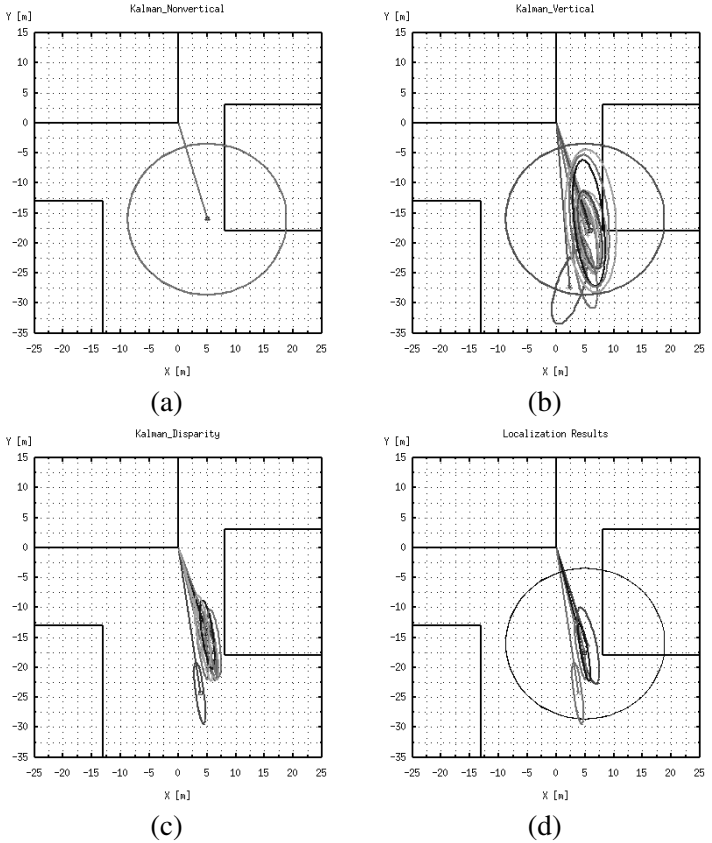


Figure 13. Multiple pose hypotheses set at the first frame step using non-vertical borders (a), vertical borders (b), disparity regions (c) and resulting merged hypotheses (d) of (c).

uncertainty of an initial robot pose and the global uncertainty of a local origin by a coordinate transformation using (2) and (3) in Section 3.

During this test run of 30 frame steps, the sensor data were recorded by the robot stopping at regular intervals to take a pair of stereo images. The average relative displacement between the observations of each frame was less than 10 m and 20° for translation, and less than 90° for rotation. The algorithm always succeeded in generating and tracking the pose hypotheses around the true poses of the robot. The error ellipses in Fig. 14a denote the unmagnified 3σ uncertainty levels of the robot positions in the local coordinate system displayed on the global map. The linked tracks of the ellipse centers between current and descendent pose hypotheses of Fig. 14a are shown in Fig. 14b with line segments of the same style for separating different frame steps. The terminated line segments in represent the pruned hypotheses.

The robot stays localized in the presence of errors and sensing ambiguities where single tracking of a pose hypothesis would fail. This is a dramatic increase in robustness which is made possible with a small computational cost of pose

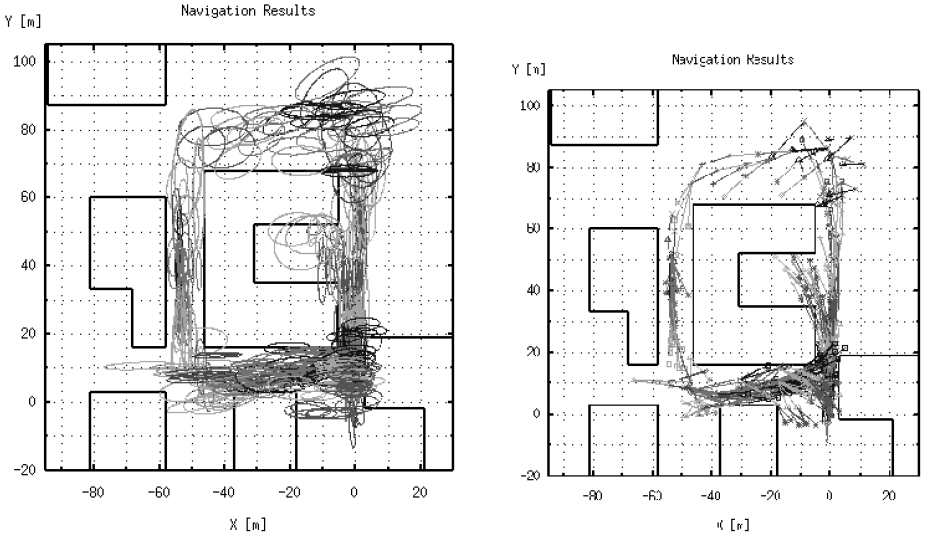


Figure 14. The results of multi-hypothesis localization on the test run.

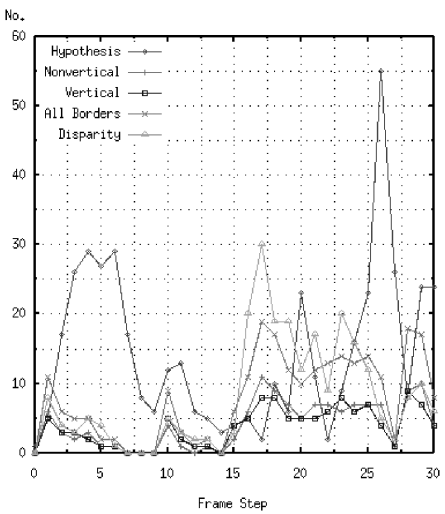


Figure 15. Number of validated pose hypotheses and respective features.

hypotheses as shown in Fig. 15. At each frame step, the average processing time of all hypotheses was lower than 5% of the total execution time including the visual processing. The minimum number of pose hypotheses at each step was usually larger than 1000 when the hypothesis management strategies of validating, pruning and merging were not applied.

The number of pose hypotheses generally increases as the number of observed features increases. Considering the large number of features around step 17, the

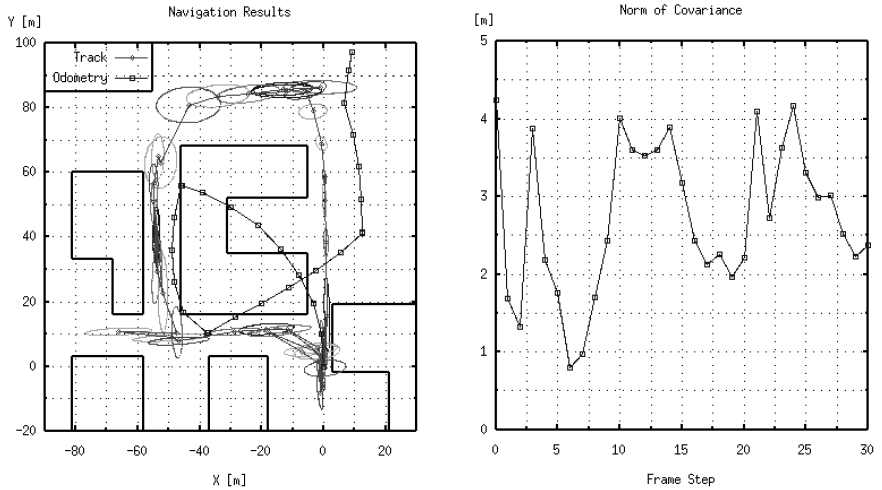


Figure 16. Comparison of a longest track and an odometry-alone result of localization (a) and norm of uncertainty covariance along the track (b).

small number of pose hypotheses at the step is, however, thought mainly due to that completely new buildings were obtained in the map when turning left into the byroad, and most of the observed features were extracted from and matched to the same building. The high number of pose hypotheses for the low number of features around step 26 is thought as being due to the place where many buildings exist.

Figure 16a shows the dead-reckoning estimates and a trajectory estimated by the MHL approach on the global map which is drawn with the mean poses and mean dimensions of the buildings. The total path length is about 250 m and the image sequence consists of 30 frames. The dead-reckoning path is completely wrong after a short frame interval. The longest track of the tracks displayed in Fig. 14b is also plotted with its 3σ uncertainty ellipses at each step. It is backtracked from the end position nearest to the start position to demonstrate the feasibility and good performance of the proposed MHL approach in this paper. Figure 16b reports the change of the norm of position covariance along the track. When no matchable features are in view, the uncertainty of the robot position becomes greater. The uncertainty of the robot position decreases whenever matchable features are in the field of view.

8. CONCLUSION AND FUTURE WORK

This paper presents an approach to determining the robot pose in an urban area where GPS cannot work since the satellite signals are often blocked by the buildings. We tested the method with real data and the obtained results show that the method is potentially applicable even in the presence of errors in feature detection of the visual features and incomplete model description of the rough map.

To make use of the rough map, which is an incomplete description of the environment, we deploy a technique based on multi-hypothesis tracking in localization. The main disadvantage of the multiple hypothesis approach is, however, the very large number of hypotheses that may be generated, although the hypothesis management techniques of validating, pruning and merging appear to constrain the hypothesis trees to manageable sizes.

This method is a part of our ongoing research aiming at autonomous outdoor navigation of a mobile robot to follow a planned path to a user-chosen location on the rough map. Thus, we want to address the integration of our system with an autonomous navigation module in the near future.

REFERENCES

1. R. Thrapp, C. Westbrook and D. Subramanian, Robust localization algorithm for an autonomous campus tour guide, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, Seoul, pp. 2065–2071 (2001).
2. K. Ohno, T. Tsubouchi, B. Shigematsu and S. Yuta, Differential GPS and odometry-based outdoor navigation of a mobile robot, *Adv. Robotics* **18**, 611–635 (2004).
3. G. N. DeSouza and A. C. Kak, Vision for mobile robot navigation: a survey, in: *IEEE Trans. Pattern Anal. Mach. Intell.* **24**, 237–267 (2002).
4. P. Jensfelt and S. Kristensen, Active global localization for a mobile robot using multiple hypothesis tracking, *IEEE Trans. Robotics Automat.* **17**, 748–760 (2001).
5. K. O. Arras, J. A. Castellanos, M. Schilt and R. Siegwart, Feature-based multi-hypothesis localization and tracking using geometric constraints, *Robotics Autonomous Syst.* **44**, 41–53 (2003).
6. B. K. P. Horn and B. G. Schunck, Determining optical flow, *Artif. Intell.* **16**, 185–203 (1981).
7. Y. Yagi, Y. Nishizawa and M. Yachida, Map-based navigation for a mobile robot with omnidirectional image sensor COPIS, *IEEE Trans. Robotics Automat.* **11**, 634–648 (1995).
8. A. Georgiev and P. K. Allen, Vision for mobile robot localization in urban environments, in: *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Lausanne, pp. 472–477 (2002).
9. B. Johansson and R. Cipolla, A system for automatic pose-estimation from a single image in a city scene, in: *Proc. Int. Conf. on Signal Processing, Pattern Recognition, and Applications*, Crete (2002).
10. G. Reitmayr and T. Drummond, Going out: robust model-based tracking for outdoor augmented reality, in: *Proc. IEEE/ACM Int. Symp. on Mixed and Augmented Reality*, Santa Barbara, CA, pp. 109–118 (2006).
11. H. Katsura, J. Miura, M. Hild and Y. Shirai, A view-based outdoor navigation using object recognition robust to changes of weather and seasons, in: *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Las Vegas, NV, pp. 2974–2979 (2003).
12. H. Morita, M. Hild, J. Miura and Y. Shirai, View-based localization in outdoor environments based on support vector learning, in: *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Alberta, pp. 3083–3088 (2005).
13. G. Chronis and M. Skubic, Sketch-based navigation for mobile robots, in: *Proc. IEEE Int. Conf. on Fuzzy Systems*, St Louis, MO, 284–289 (2003).
14. M. Tomono and S. Yuta, Mobile robot localization based on an inaccurate map, in: *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Maui, HI, 399–405 (2001).
15. N. L. Chang and A. Zakhor, Constructing a multivalued representation for view synthesis, *Int. J. Comp. Vis.* **45**, 157–190 (2001).

16. I. Moon, J, Miura and Y. Shirai, On-line extraction of stable visual landmarks for a mobile robot with stereo vision, *Adv. Robotics* **16**, 701–719 (2002).
17. J. M. Porta, J. J. Verbeek and B. J. A. Krose, Active appearance-based robot localization using stereo vision, *Autonomous Robots* **18**, 59–80 (2005).
18. C. Pantofaru, R. Unnikrishnan and M. Hebert, Toward generating labeled maps from color and range data for robot navigation, in: *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Las Vegas, NV, 1314–1321 (2003).
19. A. Okamoto, Y. Shirai and M. Asada, Integration of color and range data for three-dimensional scene description, *IEICE Trans. Inf. Syst.* **E76-D**, 501–506 (1993).
20. D. Fillat and J. A. Meyer, Map-based navigation in mobile robots: I. A review of localization strategies, *Cognitive Syst. Res.* **4**, 243–282 (2003).

ABOUT THE AUTHORS

Jooseop Yun received the BE degree in Mechanical Engineering and the ME degree in Automation and Design Engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea in 1992 and 1997, respectively. He is currently a PhD student at the Department of Mechanical Engineering, Osaka University, Suita, Japan. His current research interests include autonomous mobile robot navigation, robot vision and artificial intelligence.

Jun Miura received the BE degree in Mechanical Engineering in 1984, the ME and DE degrees in Information Engineering in 1986 and 1989, respectively, all from the University of Tokyo, Tokyo, Japan. In 1989, he joined the Department of Mechanical Engineering, Osaka University, Suita, Japan, where he is currently an Associate Professor. From March 1994 to February 1995, he was a Visiting Scientist at Computer Science Department, Carnegie Mellon University, Pittsburgh, PA. He received the Best Paper Award from the Robotics Society of Japan in 1997. His research interests include robotics, artificial intelligence and computer vision.