Image-Based Localization with Depth-Enhanced Image Map

Dana Cobzas, Hong Zhang and Martin Jagersand
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada
{dana, zhang,jag}@cs.ualberta.ca

***Abstract*— In this paper, we present an image-based robot incremental localization algorithm which uses a panoramic image-based map enhanced with depth from a laser range finder. The image-based map (model) contains both intensity information as well as sparse 3D geometric features. By assuming motion continuity, a robot can use the depth information in the image-model to project the relevant 3D model features, specifically vertical lines, of the environment to its camera coordinate frame. To determine its location, the robot first acquires an intensity image and then matches the 2D geometric features in the image with the projected model features. The first contribution of this research is that we avoid the difficult problem of full 3D reconstruction from images by employing a range sensor registered with respect to the intensity image sensor; secondly, we provide an algorithm that performs incremental robot localization using only 2D images. Experimental results in indoor map building and localization demonstrate the feasibility of our approach and evaluate the performance of the algorithm.**

I. Introduction

A challenging and important problem in mobile robotics research is how to represent an unstructured environment in terms of a map, and use the map for robot localization and navigation. A navigation map may contain different levels of detail, varying from a complete CAD model to a graph representing the connectivity among topological elements of interest of the environment. Most of the early vision-based navigation systems rely on spatial geometric models that contain precise metric measurements of the objects in the environment. Unfortunately, without *a priori* knowledge about the environment, it is in general difficult to create a precise metric map that contains the level of detail sufficient for robot localization [8], [1].

Appearance-based models have emerged recently as an alternative. An appearance-based model is created by "memorizing" the navigation environment using images or templates. By comparing the templates in the model with the its current view, a robot can derive control commands to steer itself along a memorized route [6] or to a goal position [4], [7], [10]. One of the major drawbacks of these appearance-based maps is that robot motion is restricted to either a predefined route or positions close to the locations where the images were originally acquired.

In this paper, we propose a new type of navigation map that enhances an appearance map with sparse geometric information. Specifically, it is formed by a panoramic image-based model augmented with a sparse set of 3D vertical line features. This model contains sufficient information about the navigation environment without explicit full 3D reconstruction. Once the room map has been obtained, we subsequently propose an incremental localization algorithm that matches the line features in the robot current view, which is a 2D image, with the line features in the navigation map to estimate the robot s position.

This research represents a significant improvement over our previous work [2] in which the depth information was acquired using a trinocular vision system, and the localization had to be performed by matching 3D features in the robot current view with those in the model. In contrast, our new localization algorithm uses only a 2D camera on the robot, resulting in a more efficient computational process. In addition, the new algorithm uses depth information acquired by a laser range finder, which is much more accurate than stereo vision. Consequently, the localization accuracy is much improved. In order to correlate intensity and range information obtained from separate sensors, however, we must register the two sensors with respect to each other. We propose an image-based algorithm for range and intensity sensor registration.

The rest of the paper will be organized as follows. Section II presents an image-based algorithm for registering range and intensity images. Section III describes the incremental-localization algorithm. Section IV shows experimental results. Conclusions are drawn and future work described in Section V.

II. Model Construction

In this section, we will first briefly describe the data acquisition system, for both range and intensity images, and define their reference frames. We will also summarize an image-based algorithm for registering the intensity and range images, in order to correlate vertical line features that are present in both image types and critical for the localization algorithm.

A. Range and Intensity Image Acquisition

The data acquisition system consists of a *laser range finder* (Acuity Research, Inc.) and a CCD *camera*, mounted on a *pan-tilt unit (PTU)* (see Figure 1). We use the pan axis of the PTU to rotate the camera in order to build a cylindrical or panoramic image model. The same pan axis of the PTU and a rotating scanner attached to the laser range finder produce two degrees of freedom, sufficient for spanning a unit sphere to acquire complete range information. Once two separate images are obtained, they will be registered to generate the final image-based model enhanced with 3D geometric features.

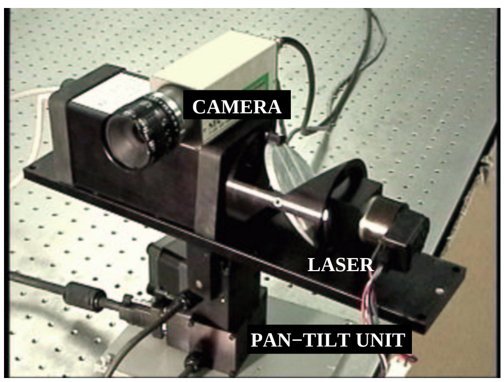

Fig. 1. System configuration: the laser range finder with the camera attached on top of it is mounted on a pan-tilt unit

The data returned for each sample of the laser range finder consist of a range r, an amplitude a, and angular position of the rotating mirror θ of the scanner. The amplitude a corresponds to the strength of the returned signal, and is related to both r and the gray-scale value of the reflecting surface. The latter property will be exploited, and it allows us to extract line features present in an a image. Since the pan angle ϕ of the PTU is also known, each sample can be expressed as a quadruple (r, a, θ, ϕ), and it can be considered as two images sampled on a spherical surface: a range image r and an amplitude image a. We ignore the small translation between the mirror center of rotation and the pan-tilt unit center, and apply a filter to eliminate outliers in the range data. Finally, because the laser range finder often does not generate uniformly sampled data, due to a variety of reasons such as noise and non-reflecting surface, we apply a 3×3 averaging filter over the neighboring samples to fill the missing values and create a uniform grid. The top figure in Figure 2 shows a scaled spherical amplitude range image, a, representing 180° scan of our navigation environment.

A panoramic intensity image mosaic can be constructed by composing planar images taken from a single viewpoint. The camera is rotated by the PTU to acquire intensity images every 10°. These images are projected on a cylinder with radius equal to the focal length of the camera, and then "stitched" or correlated in order to precisely determine the amount of rotation between two consecutive images. In the cylindrical space, a rotation becomes a translation, so we can easily build the cylindrical image by translating each image with respect to the previous one. To reduce discontinuities in intensity between images, we weigh the pixels in each image proportionally to their distance to the edge [9]. The corresponding 180° panoramic mosaic for the range scan is shown in the bottom figure of Figure 2.

B. Range-Intensity Image Registration

The registration of range and intensity data refers to the step of associating each pixel in the range image with a pixel in the intensity image, and it is an important problem, especially in the fields of model building and realistic rendering. We have compared two representative approaches [3], one that is based on recovering the rigid transformation between the two sensors and the other based on locally computing an image warp, and found that the latter is fast and adequate for application that does not require a high-precision alignment. It is therefore chosen for our purpose.

The first step in the registration algorithm is to project the spherical range data into a cylindrical representation with the radius equal with the focal length of the camera. This mapping is given by

$$P(r, \theta, \phi) \mapsto P(r, \theta, f \tan\phi) = P(r, \theta, h) \qquad (1)$$

where r represents the distance from the center of the cylinder to the point, h is the height of the point projected on the cylinder, θ is the azimuth angle and f the focal length of the camera. Again, we sample these data on a cylindrical grid θ, h and represent it as a cylindrical image. The same procedure is applied to the amplitude data to obtain a cylindrical amplitude image.

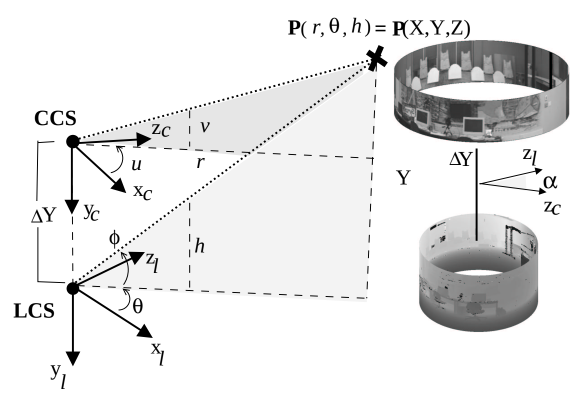

Fig. 3. The projection of a space point P in the cylindrical image (θ, h) and the panoramic mosaic (u, v). We approximate the laser-camera transformation with a translation ΔY and a rotation over y axis.

Fig. 2. (top) Spherical representation of the range data from an 180° scan after filtering (bottom) Corresponding 180° panoramic mosaic.

From the intensity and range data in similar cylindrical image representations, we compute a *global mapping* between the two. We approximate the physical configuration of the sensors as in Figure 3 assuming only a vertical translation ΔY and a pan rotation between the two reference coordinate systems LCS (laser coordinate system) and CCS (camera coordinate system). For a point $p_l(\theta,h)$ in the cylindrical laser image its corresponding point in the panoramic mosaic $p_c(u,v)$ is

$$\begin{aligned} u &= a\theta+\alpha \\ v &= f\frac{Y-\Delta Y}{r} = f\frac{Y}{r} - f\frac{\Delta Y}{r} = bh - f\frac{\Delta Y}{r} \end{aligned} \quad (2)$$

where a and b are two warp parameters that will account for the difference in resolution between the two images, α aligns the pan rotation and $Y = rh/\sqrt{f^2+h^2}$ is the height of the 3D point $P(r,\theta,h)$. For our setup we have $f = 1000$ pixels, $\Delta Y = 5$ cm and the range of the points is $r = 5-8$ m, so $f\Delta Y/r = 6-10$ pixels and it can be approximated to a constant $-\beta$. The general warp equations are:

$$u = a\theta+\alpha, \quad v = bh+\beta \quad (3)$$

We compute the warp parameters (a,b,α,β) from two, typically about 20 corresponding points in the two images using a least square approach.

After the global mapping, the two images are only approximately aligned with a local misalignment of $5-7$ pixels. We perform a *local alignment* using a set of 20-30 corresponding control points. The local map "stretches" the range data to fit the intensity data using cubic interpolation based on a 2D Delaunay triangulation of the control points.

C. Model vertical lines

The proposed localization algorithm uses vertical line features. We choose vertical lines because they naturally occur in an indoor environment and when projected on a cylindrical image, they remain vertical and are not transformed into curves as would horizontal or arbitrary lines. Consequently a standard edge detection and linking algorithm can be used to detect vertical line segments.

We manually select a set of vertical line segments in the cylindrical amplitude image. The selected edges represent discontinuities in color and lie inside of a planar surface to avoid errors caused by edges at the boundary between two surfaces. The equation of each 3D line is computed by fitting a vertical line to the selected model points. The segment points are projected on the panoramic intensity image using the above mentioned registration algorithm, and a line segment is then fitted to them. Figure 4 shows the selected lines in the amplitude image (top) and the projected lines on the panorama (bottom). The small misalignment is the error of the registration algorithm. The image-based navigation map consists of the panoramic mosaic with the vertical line segments, and their corresponding 3D coordinates with respect to CCS (model coordinate frame).

III. Robot localization

With the panoramic model constructed in the previous section, the localization problem involves finding the position and orientation of the robot with respect to the model (CCS) using the current robot view in terms of a 2D image. We assume planar motion, which is reasonable for indoor environments where motion takes place on the floor.

An overview of the localization algorithm is presented in Figure 5. We perform an incremental localization where the current position is approximately known either from the previous position - assuming motion continuity - or from other sensors (odometry). An initial position and the height difference between the model location and the robot have to be estimated at the beginning using, for example, manually selected corresponding feature points. The first step (a) is to detect vertical line segments in the current image using standard edge detection and linking algorithms. The next step (b) is the angular calibration and detection of the best match based on the minimum Haus-

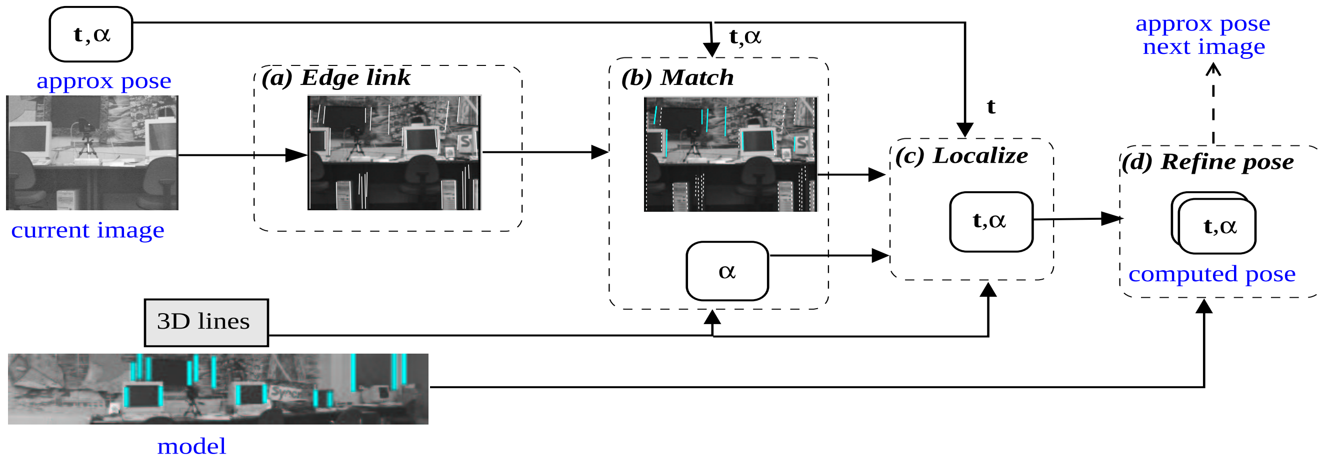

Fig. 5. Overview of the localization algorithm. **t** denotes robot translation and α robot orientation: (a) Edge detection and linking (b) Compute best match and calibrate angle α based on Haudorff distance (c) Localization using model-image correspondence (d) Position refinement suing panoramic model

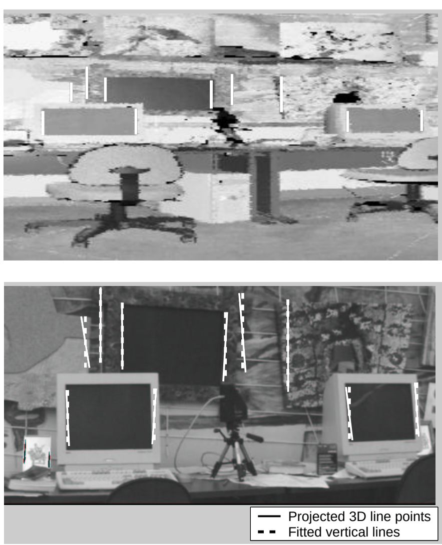

Fig. 4. Selected vertical lines in the laser amplitude image (top) and Projected 3D lines in the panoramic mosaic and the fitted vertical lines (bottom)

dorff distance between the projected model 3D lines and the detected vertical edges. The current position is updated from the corresponding model-image segments in step (c). A final position refinement step (d) uses intensities of the panorama lines to best align the projected image lines. The incremental localization approach assumes that the position is approximately known, limiting the matching search region, relative to a global localization approach. We will first introduce some notations followed by the general theory that is used in the localization algorithm.

A 3D line can be represented in terms of a unit vector $\mathbf{v}$ which indicates the direction of the line and a vector $\mathbf{d}$ that represents a point on the line. For a vertical line $\mathbf{v} = [0,1,0]^T$ and $\mathbf{d}$ can be chosen as the intersection of the line with the horizontal plane $\mathbf{d} = [X,0,Z]^T$ (Note that the vertical axis is perpendicular to the ground plane). Any point on the line can be expressed as $\mathbf{P}_l = [X,k,Z]^T$ where k is a real parameter that is restricted to an interval in case of a line segment. A vertical image line can be characterized by the column coordinate u, and a point on the line has the form $\mathbf{p}_l = (u,q)$, where q is a parameter similar to the 3D case.

We denote the unknown transformation between the model coordinate system (CCS) and the current image position (CIP) by $(R_y, \mathbf{t})$, where R_y is a rotation about Y axis, and $\mathbf{t} = [t_x, H, t_z]^T$ (H is the height difference between the model and the robot). The vertical line points $\mathbf{P}_l = (X,k,Z)$ expressed in model reference system are projected on the image using:

$$\mathbf{p}_l = C(R_y \mathbf{P}_l + \mathbf{t}) \tag{4}$$

where the camera matrix C has the form:

$$C = \begin{bmatrix} a_u & 0 & u_c \\ 0 & a_v & v_c \\ 0 & 0 & 1 \end{bmatrix}$$

Using Equation 4, the column coordinate of the projected line can be derived as:

$$u_{proj} = a_u \frac{X\cos\alpha + Z\sin\alpha + t_x}{-X\sin\alpha + Z\cos\alpha + t_z} + u_c \tag{5}$$

where α is the pan rotation angle.

We assume that N vertical lines from the model are visible in the current view and M vertical edges are detected in the current image ($M > N$).

A. Angle calibration and vertical line matching

An error of few degrees in the approximate orientation of the robot results in a big displacement between the projected lines and the corresponding detected image edges making the matching process very difficult. We calibrate the angular orientation using a modified Hausdorff distance [5] between the projected and detected edges. We vary the orientation angle α in an interval of $10°$ around the given approximate position and compute the corresponding Hausdorff distance between the projected model lines $u^i_{proj}(\alpha)$, $i = 1 \dots N$ (Equations 4 and 5) and detected vertical edges u^k, $k = 1 \dots M$.

$$H(\alpha) = \sum_i K^{th}_{i=1\dots N}\{\min_{k=1\dots M} d(u^i_{proj}, u^k)\} \qquad (6)$$

where $K^{th}_{i=1\dots N}$ denotes the K ranked values in the set of distances (one corresponding for each model line). $d(u^i_{proj}, u^k)$ represents the Euclidean distance between the center points of the line segments. We denote the edge segment that gives the minimum distance to a projected model line u^i_{proj} by u^{k_i}. We choose the angle that has the smallest distance $H(\alpha)$. The corresponding set of line-edge pairs $(u^i_{proj}, u^{k_i}), i = 1 \dots K$ from Equation 6 represents the desired matched model-image features. Figure 6 shows the detected vertical edge segments, the projected model lines and the corresponding matched edges, for one example image.

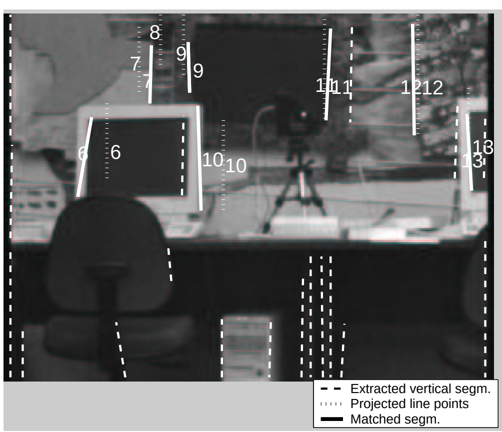

Fig. 6. Illustration of matching algorithm: detected vertical lines, projected lines and detected matches. Numbers indexes lines in the model and show the matched pairs.

B. Localization using vertical line segments

If K pairs of 3D model lines - 2D image edges are available, we compute the motion parameters (α, t_x, t_z) by minimizing the displacement between the corresponding projected and detected lines.

$$(\alpha, t_x, t_z) = \arg\min_{\alpha, t_x, t_z} \sum_{i=1}^{K} (u^i_{proj} - u^{k_i})^2 \qquad (7)$$

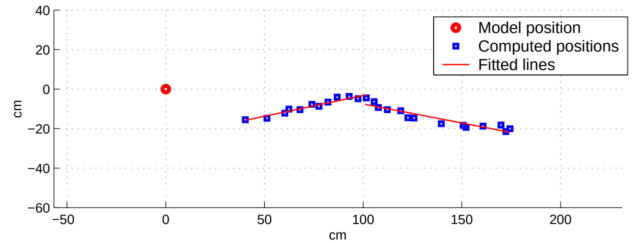

Fig. 7. Recovered positions using localization algorithm where the dimension of the room is in centimeters

We solve this non-linear least square problem using Levenberg-Marquardt non-linear minimization algorithm.

C. Position refinement in image space

The robot position detected using the described algorithm is inaccurate due to errors present in the model lines and edge extraction. We further refine this position using an image-based approach. We consider a small patch along each model line in the panoramic mosaic Υ^i and the corresponding patch in the current image I^i. The patch in the image is calculated by projecting corresponding 3D coordinates of the corners from the model patch Υ^i. This construction gives us the correct size and shape of the image patch. We use a homography to warp the image patch to the dimension of the model patch to obtain I^i_w. The refined motion parameters are calculated by minimizing the difference in intensity between the corresponding model-image patches:

$$(\alpha, t_x, t_z) = \arg\min_{\alpha, t_x, t_z} \sum_{i=1}^{N_s} |\Upsilon^i - I^i_w| \qquad (8)$$

For solving the minimization problem we used Nelder-Mead multidimensional unconstrained nonlinear minimization.

IV. Experimental Results

To evaluate the model accuracy and the performance of the localization algorithm, we took 26 images along a trajectory at positions that are 10 cm apart, and recovered their positions using the localization algorithm described in the previous section. For the first position, we manually selected corresponding point features in the image-based model and the robot view at the position, then applied a point-based localization algorithm described in [2]. This initialization step not only provided a reliable initial position of the robot for studying the incremental localization algorithm along the trajectory, but also recovered the difference in height, H, between the image-based model and the robot camera reference frame, which would remain constant throughout the experiments.

Subsequently, we performed the incremental on-line localization algorithm for the remaining 25 points. At each

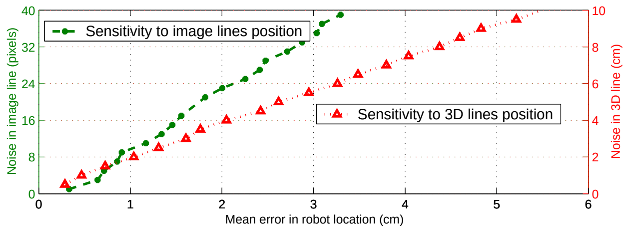

Fig. 8. Sensitivity of the localization algorithm with respect to features position in image vs. 3D.

position, the location computed in the previous step is used as an approximate location for the current step. All the measurements were relative to the reference frame of the image-based model. Figure 7 plots the recovered positions using our algorithm.

We measured the relative accuracy of the localization algorithm, in terms of the position error along the trajectory δ, assumed to be 10 cm apart, and the position error tangent to the trajectory, ρ. For the 26 positions, the two errors are found to 2.24 *cm* and 1.22 *cm*, respectively. Both of those errors are quite satisfactory. It is worth mentioning that errors are less where the 3D line features were more accurate.

To examine the robustness of the algorithm with respect to the errors in the 3D line features of the model and the errors in 2D vertical line detection in the current image, we added different levels of uniform noise to the 3D lines and the manually selected feature points. We compute the average error between the reconstructed positions using the noisy data and the reconstructed position using noise-free data. Figure IV plots the levels of perturbations in the 3D line and image line positions that will produce different errors in the estimated robot location. This result is useful to estimate the performance of the localization algorithm under varying sensor characteristics.

V. Conclusions and Future Work

In this paper, we have presented a new type of robot navigation map that combines the appearance of the environment with sparse geometric information. The map is created by enhancing a panoramic image mosaic with depth data, more specifically, with 3D line features. The range data, acquired using a laser range finder, is registered with respect to the image mosaic using an image-based approach. We have then proposed an incremental localization algorithm that determines the robot location by matching the 2D features in the robot current view with model features projected to robot camera coordinate frame. This new localization algorithm uses only a camera on the robot, and its localization accuracy is superior to the previous approach in [2], with the mean position error reduced by about a half.

In the future, we plan to improve the localization algorithm by embedding a statistical model of the uncertainty in the robot location location. We also want to extend the algorithm to other geometric features.

VI. References

[1] N. Ayache and O. D. Faugeras. Maintaining representation of the environment of a mobile robot. *IEEE Transactions on Robotics and Automation*, 5(6):804–819, 1989.
[2] D. Cobzas and H. Zhang. Cylindrical panoramic image-based model for robot localization. In *Proc. of IROS*, 2001.
[3] D. Cobzas, H. Zhang, and M. Jagersand. A comparative analysis of geometric and image-based volumetric and intensity data registration algorithms. In *Proc. of ICRA*, pages 2506–2511, 2002.
[4] J. Hong, X. Tan, B. Pinette, R. Weiss, and E. Rseman. Image-based homing. *IEEE Control Systems*, pages 38–44, 1992.
[5] D. Huttenlocher, D. Klanderman, and A. Rucklige. Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863, September 1993.
[6] S. Li and S. Tsuji. Qualitative representation of scenes along route. *Image and Vision Computing*, 17:685–700, 1999.
[7] Y. Matsumoto, M. Inaba, and H. Inoue. Visual navigation using view-sequenced route representation. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 83–88, 1996.
[8] H. P. Moravec. The stanford cart and the CMU rover. *Autonomous Robot Vehicles*, pages 407–419, 1990.
[9] R. Szeliski and H.-Y. Shum. Creating full view panoramic image mosaics and environment map s. In *Computer Graphics (SIGGRAPH'97)*, pages 251–258, 1997.
[10] N. Winters and J. Santo-Victor. Mobile robot navigation using omnidirectional vision. In *Proc. 3rd Irish Machine Vision and Image Processing Conference (IMVIP'99)*, 1999.