MVA '96 IAPR Workshop on Machine Vision Applications, November. 12-14, 1996, Tokyo, Japan

Position Estimation and Correction of Mobile Robot by Model-based Scene Matching and Optimization Method

Yi-Bing Yang
Dept. of Radio Eng., Southeast University
Nanjing, P.R.China, email:ybyang@seu.edu.cn

H.T. Tsui
The Chinese University of Hong Kong, Hong Kong
email:httsui@ee.cuhk.edu.hk

Abstract

This paper presents a method to estimate and correct the positions of an indoor mobile robot. Scene matching is based on geometric hashing[9, 10]. An optimization method[12] of maximizing cross-correlation of image regions is used to estimation correction. We assume that the robot is equipped with two or more video cameras, a 2D laser range finder and an odometry system. The 3D CAD model of indoor environment is known. First, the indoor environments are remodeled in a form convenient for scene matching. Some selected features, which are the model edge positions on the horizontal plane of the 2D laser range finder, are transformed into geometric invariants by system calibration and range data fusion. These geometric invariants of features are modeled off-line in hashing tables according to the proposed basis constraints[13]. They are then used as indices to make scene matching by geometric hashing with weighted voting rule. After scene matching, the self-localization of mobile robot can be completed by coordinate transformation and a least square method. Furthermore, we use a two-step optimized descent method in a number of search directions to make estimation correction. Finally, the next position estimation can be made more accurate and efficient by combining uncertainty analysis[1] with geometric hashing.

1 Introduction

The mobile robot navigation is often problematic. Sensors cannot measure the environment precisely or completely due to the limitation of model representation and the error of real measurements. Robot localization is the prerequisite for correct navigation and is also one of the most difficult problems in robot navigation. The navigation and localization problems are usually divided into two distinct classes: reference landmarks guidance[8, 6] and dead reckoning[1, 2]. The latter methods are often combined with the extended Kalman filter together to navigate and locate robot. They often suffer from the problem of unbounded errors in position estimation due to the errors of the model and odometry system. On the other hand, the former methods depend on the recognition of external reference landmarks and maintain the position estimate errors within bounded limits. However, it is often computationally too complex to allow real-time performance.

For preventing the errors from being accumulated in robot localization, various approaches[11, 3, 7] have been proposed. Most of them employ visual sensors and internal description to represent its environment to perform robot localization. But, in practice, some preliminary conditions about environments are needed for scene matching[5, 4]. An efficient method to cut down the search space for matching is essential. Geometric hashing[10, 9] is an efficient approach for using image features as indices to a database of models for scene matching. The major advantages for geometric hashing in scene matching are its ability to deal with partial occlusion and its efficient search strategy.

In this paper, we presents a new approach for robot localization. It consists of estimating and correcting the positions of an indoor mobile robot by scene matching based on geometric hashing[9, 10] and optimization[12] of maximum cross-correlation of image regions. First, the known indoor environments are remodeled in hashing tables[13] in a form convenient for scene matching after system calibration. Then, scene matching and robot localization are performed by geometric hashing and position transformation. Furthermore, position correction is carried out by an optimal descent method which maximizes cross-correlation of image regions between real and estimated positions. Finally, the next position estimation can be made more accurate and efficient by combining uncertainty analysis[1] with geometric hashing.

2 System Calibration and Scene Modeling

The mobile robot is equipped with two or more video cameras, a 2D laser range finder and an odometry system. The 3D CAD model of indoor environment is known. The

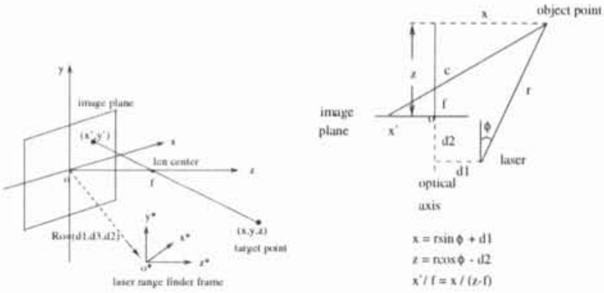


Figure 1. The calibration geometry of the camera and the 2D laser range finder on mobile robot.

objective of system calibration is to relate the range data measured by a laser range finder to the coordinates relative to each video camera. If $\mathbf{r_{TV}} = (x, y, z)^T$ is the position for point $\mathbf{p}$ measured in a camera coordinate system and $\mathbf{r_L}$ is the position of the same point measured in the laser range finder coordinate frame, then, $\mathbf{r_{TV}} = \mathbf{R}\mathbf{r_L} + \mathbf{r_0}$, where $\mathbf{R}$ is a 3x3 orthonormal matrix representing the rotation and $\mathbf{r_0} = (d_1, d_3, d_2)^T$ is a translation vector. In our system, the range finder is parallel to the optical axis of the lens of the cameras, thus, $\mathbf{r_{TV}} = \mathbf{r_L} + \mathbf{r_0}$. We need only to find the translation vector between the camera frame and the laser range finder frame when the focal length f of the camera is calibrated. Assuming a perspective transform model of the camera, the $\mathbf{r_0}$ can be easily found by using known calibration points by least square fitting. After system calibration, the range data from range finder can be mapped onto each gray image as shown in Fig. 1 when the field of view of the video camera is covered by that of the laser range finder. Hence, for the overlapping part of the two views, the correspondence between the sampling points of the laser range finder and the pixels of the intensity image of the camera are established. Thus, we can easily transform image feature positions into the coordinates in robot-centered frame using the corresponding range data.

In scene modeling, our objective is to transform the vertical edge positions of the environments on the horizontal plane corresponding to the 2D laser range finder height into similarity invariants[13, 10]. Then, these similarity invariants will act as the index to the hashing tables[13]. First, we select a pair of ordered feature points $\mathbf{X_0}$ and $\mathbf{X_1}$ as a basis by assigning the coordinates $(\mathbf{0}, \mathbf{0})$ and $(\mathbf{1}, \mathbf{0})$ to them. $(\mathbf{X_1} - \mathbf{X_0})$ is normalized as an unit vector. The third point $\mathbf{X_2}$ with $(\mathbf{X_2} - \mathbf{X_0})$ as the second orthogonal unit vector in the counter-clockwise direction is uniquely defined. Any other point $\mathbf{X}$ in the same plane can be represented by this basis as: $\mathbf{X} = \mathbf{X_0} + \alpha(\mathbf{X_1} - \mathbf{X_0}) + \beta(\mathbf{X_2} - \mathbf{X_0})$, where, (α, β) is the similarity coordinates for point $\mathbf{X}$, and they are the similarity invariants. That is, $\mathbf{TX} = \mathbf{TX_0} + \alpha(\mathbf{TX_1} - \mathbf{TX_0}) + \beta(\mathbf{TX_2} - \mathbf{TX_0})$, when $\mathbf{T}$ is a similarity transformation. Scene modeling is executed off-line. Each (α, β) is used as an index to a hash table cell where the basis and model identities are recorded as $(model, basis - pair)$. In practice, the selection of basis is guided by a set of rules[13] to improve matching efficiency.

3 Scene Matching and Initial Localization

As in the scene modeling, we consider only those scene edge points corresponding to the intersections of detected vertical edges and the horizontal scanning plane of the 2D laser range finder as scene features. Only those vertical edges whose lengths over a certain threshold in order to reduce the effect of edge detection error. First, these scene features selected are transformed into the coordinates in the robot-centered frame as shown in Fig. 1. Then, for each selected scene basis, we calculate the geometric invariants (α, β) for all other edge points. Each invariant set (α, β) will be used as a pointer to cast weighted votes on choosing the $(model, basis - pair)$. This process is repeated for all (α, β) sets corresponding to a basis. The neighborhood bins of the indexed bin are also checked to account for the effect of noise and uncertainty in the computation. Each hash table entry $(model, basis - pair)$ should get a weighted vote when their hash values are within a certain range of the indexing hash values (α, β). The weight of the vote depends on the difference between two hash values and selected distance function. The weight is set to zero when the distance is over a certain threshold.

The highest score $(model, basis - pair)$ is used as a candidate match with the scene if it is above a prescribed threshold. The unique transformation between the model basis and matched scene basis can then be computed from the matching result. The match can be verified by checking whether the transformed edge points lie close to the scene feature positions in robot-centered frame. If no high score is counted or the verification fails with the selected basis, the matching will be restarted with a different basis in the scene. From the result of scene matching, the transformation between the observed scene and the known model can be established. Consider a point (x, y, z) in the world frame. Its coordinates is known to be (x_r, y_r, z_r) in the robot-centered frame by scene matching. The location (p_x, p_y) and pose φ of the robot can be easily obtained from the equation below[1].

$$\begin{bmatrix} x_r \\ y_r \\ z_r \\ 1 \end{bmatrix} = \begin{bmatrix} cos\phi & sin\phi & 0 & -p_x cos\phi - p_y sin\phi \\ -sin\phi & cos\phi & 0 & p_x sin\phi - p_y cos\phi \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

4 Estimation Correction

The localization errors may still be significant, even if the scene matching itself is correct. Imaccuracy of the detected image feature position will result in these errors. Hence, in order to improve robot localization accuracy, we propose an optimized descent method to correct the estimation. The objective function is defined as the cross-correlation function between the initial scene image $f(x_i, y_j, \varphi_0)$ and estimated scene image $f(x_i + \Delta x, y_j + \Delta y, \varphi_0 + \Delta\varphi)$ by back-projection of the 3D CAD model from the initial location and pose estimates.

$$R(\Delta x, \Delta y, \Delta\varphi) = \frac{cov(\Delta x, \Delta y, \Delta\varphi)}{\sigma_0 \sigma_k}$$

$$\begin{aligned} cov(\Delta x, \Delta y, \Delta\varphi) &= \sum_{i=1}^{i=N}\sum_{j=1}^{j=N}[f(x_i, y_j, \varphi_0) - \bar{f}_0] \\ &\times \ [f(x_i + \Delta x, y_j + \Delta y, \varphi_0 + \Delta\varphi) - \bar{f}_k] \end{aligned}$$

$$\sigma_0 = \{\sum_{i=1}^{i=N}\sum_{j=1}^{j=N}[f(x_i, y_j, \varphi_0) - \bar{f}_0]^2\}^{1/2}$$

$$\sigma_k = \{\sum_{i=1}^{i=N}\sum_{j=1}^{j=N}[f(x_i + \Delta x, y_j + \Delta y, \varphi_0 + \Delta\varphi) - \bar{f}_k]^2\}^{1/2}$$

$$\bar{f}_0 = \frac{1}{N^2}\sum_{i=1}^{i=N}\sum_{j=1}^{j=N} f(x_i, y_j, \varphi_0)$$

$$\bar{f}_k = \frac{1}{N^2}\sum_{i=1}^{i=N}\sum_{j=1}^{j=N} f(x_i + \Delta x, y_j + \Delta y, \varphi_0 + \Delta\varphi)$$

The goal is to maximize the objective function $R(\Delta x, \Delta y, \Delta\varphi)$. Correction is done in three steps. The first step is to divide the search space around the initial estimation point into 8 directions (see Fig. 2 (a)) in order to find the steepest error decent direction. Here accuracy is sacrificed for speed of computation. We can take the correction step size as $\alpha \times (\Delta x, \Delta y, \Delta\varphi)$ in each correction direction (see Fig. 2 (a)), where, $\alpha \in (0, 1)$ is a constant which can be adjusted according to experiences, $\Delta x, \Delta y, \Delta\varphi$ represent largest estimation errors for x, y, φ respectively. Then, the search space in the second step is limited to 6 neighboring directions around the direction found in the first step correction, in order to get a more accurate steepest decent direction. In Fig. 2 (b), the 6 search directions will be $(0, -1, -1), (1, -1, 0), (0, -1, 0), (1, 0, 0), (0, 0, -1)$ and $(1, 0, -1)$, if the direction found in the first step correction is $(1, -1, -1)$. In step three, we can further correct the position by bidirectional binary search method in the direction found in step two. In general, we can stop the correction after the first step re-estimation, if the initial estimation error is small. In fact, the correction algorithm follows the common stopping rules, that is, it will not be made or be stopped if the objective function for initial estimation or re-estimation is large enough than a prescribed threshold (i.e. when $r(x_i, y_i, \varphi_i)$ is large enough or $\| \bigtriangledown r(x_k)\|$ is small enough, the computation will be ended). It can be proved[12] that this optimized descent method is linearly convergent. The convergent rate is also quite fast.

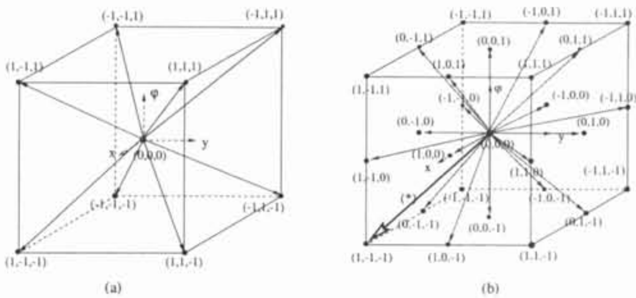


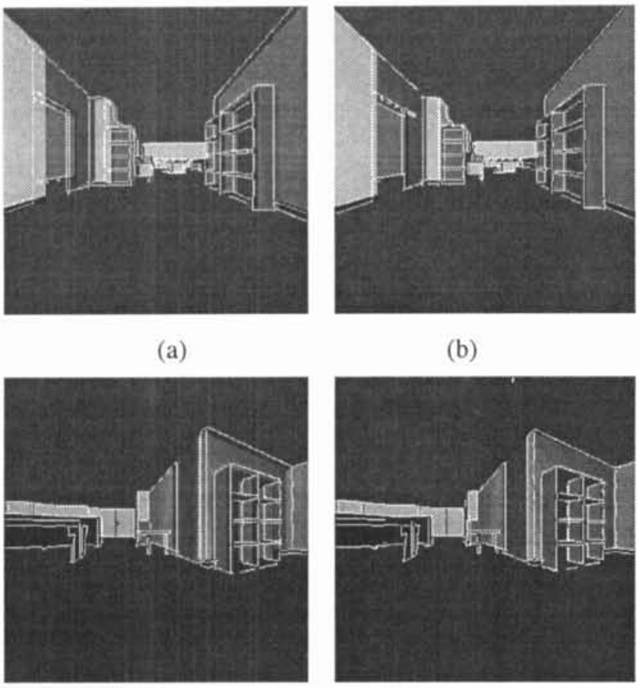
Figure 2. (a). 8 correction directions around initial estimate in the first step correction. (b). The possible correction directions in the second step correction. For example, the 6 search directions will be $(0, -1, -1)$, $(1, -1, 0)$, $(0, -1, 0)$, $(1, 0, 0)$, $(0, 0, -1)$ and $(1, 0, -1)$, if the direction found in the first step correction is $(1, -1, -1)$.

5 Next Position Estimation and Correction

After initial localization and position correction, the robot will move to a next position. We can estimate and correct next position by using the odometry data. It means that the search space for scene matching based on geometric hashing and position correction above can be limited to within the range of the uncertainty[1] of the odometry data and the mataining of the previous location error. We only need to compute the geometric invariants for those scene features located within the range of the uncertainty about the predicted locations.

6 Experimental Results and Conclusions

We had tested the proposed initial localization method in real laboratories and simulated environments. Experimental results show that the initial position estimation method of robot based on geometric hashing and information fusion is feasible and reliable. The initial position and angle estimated errors are within $10cm$ and $1°$ respectively without using correction. The errors can be reduced within $3cm$ and $0.4°$ respectively after using the correction of step one. Fig. 3 (a) and (c) give two initial estimated results and (b) and (d) the result after estimation correction of the first step. Local-



(a) (b)

(c) (d)

Figure 3. The grey images present the scene from real robot position, the set of white lines superimposed on each image is the edge map derived from the model in the estimated position of the mobile robot. (a) and (c) are two initial estimation positions, (b) and (d) are estimation correction results to (a) and (c) by the first step correction. (a) $err\_x = 7.1cm$, $err\_y = 7.7cm$, $err\_\varphi = 0.95°$, $R(\Delta x, \Delta y, \Delta\varphi) = 0.850630$; **(b)** $err\_x = 2.156cm, err\_y = 2.756cm, err\_\varphi = 0.40°$, $R(\Delta x, \Delta y, \Delta\varphi) = 0.923051$; **(c)** $err\_x = 6.30cm, err\_y = 8.20cm, err\_\varphi = 0.80°$ $r(\Delta x, \Delta y, \Delta\varphi) = 0.856667$; **(d)** $err\_x = 1.356cm, err\_y = 3.254cm, err\_\varphi = 0.306°, r(\Delta x, \Delta y, \Delta\varphi) = 0.930555$.

ization errors are expected to be smaller in subsequent positions when information from odometry data can be used.

The original contributions and their importances for this work are: (1) the initial estimated position can be corrected by steepest descent method, in which, it can be proved that the convergence rate is quite fast and is linear[12] because the first step correction is made only in 8 directions of 3D parametric space. The re-estimation accuracy of robot position can be greatly improved; (2) the search space for scene matching in next position estimation can be greatly cut down by use of the uncertainty analysis of the odometry and geometric hashing with seven constraints; (3) there is no limitation of view point for initial position estimation, and it can work in most cases of occlusion as our scene matching is based on geometric hashing and information fusion.

7. Acknowledgment

This research project is supported by the RGC Earmarked Research Grant RGC Ref. No. CUHK 292/94E of the Hong Kong Government.

References

[1] A.Kosaka and A. Kak. Fast vision-guided mobile robot navigation using model-based reasoning and prediction of uncertainties. *CVGIP(IU)*, 56(3):271–329, 1992.

[2] E. Baumgartner and S. Skaar. An autonomous vision-based mobile robot. *IEEE Trans. Automat. Control*, 39(3):493–502, Mar. 1994.

[3] D.Kang and et. al. Position estimation for mobile robot using sensor fusion. In *Proc. of the 1995 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 1, pages 271–276, Pittsburgh, Pennsylvania, USA, Aug. 1995.

[4] H. S. Dulimarta and A. K. Jain. Mobile robot localization in indoor environment. In *Proc. of the Third International Conference on Automation, Robotics and Computer Vision*, pages 2204–2208, Singapore, Nov. 1994.

[5] E.Krotkov and et al. Mobile robot localization using a single image. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 978–983, Washington, DC, 1989. IEEE Computer Society, Computer Society Press.

[6] E.Yeh and D. Krigman. Toward selecting and recognizing natural landmarks. In *Proc. of the 1995 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 1, pages 47–52, Pittsburgh, Pennsylvania, USA, Aug. 1995.

[7] D. Huttenlocher, M. E. Leventon, and W. J. Rucklidge. Visually-guided navigation by comparing two-dimention edge images. In *Proc. of Computer Vision and Pattern Recognition'94*, pages 842–847, Seattle, Washington, June 1994. IEEE Computer Society Press.

[8] M. Kabuka and A. Arenas. Position verification of a mobile robot using standard pattern. *IEEE Trans. Robotics Automat.*, 3(6):505–516, Dec. 1987.

[9] Y. Lamdan, J. Schwartz, and H. Wolfson. Affine invariant model-based object recognition. *IEEE Trans. on Robots and Automation*, 6(5):578–589, Oct. 1990.

[10] Y. Lamdan and H. Wolfson. Geometric hashing: A general and efficient model-based recognition scheme. In *Proc. of the Int. Conf. on Computer Vision*, pages 238–249, Los Alamitos, CA, 1988. Computer Society Press.

[11] K. Sugihara. Some location problems for robot navigation using a single camera. *CVGIP*, 42:112–129, 1988.

[12] S. Q. Wu. Convergence properties of decent methods for unconstrained minimization. *Optimization*, 26:229–237, 1992.

[13] Y.-B. Yang and H. Tsui. Mobile robot localization by geometric hashing and model-based scene matching. In *Proc. of IEEE 13th Inter. Conf. on Pattern Recognition*, Vienna, Austria, Aug. 1996. IEEE Computer Society Press.