Investigating the Performance of Corridor and Door Detection Algorithms in Different Environments

Wenxia Shi and Jagath Samarabandu
Department of Electrical and Computer Engineering
The University of Western Ontario
LONDON, ONTARIO, CANADA N6A 5B9
Email: wshi7@uwo.ca, jagath@uwo.ca

Abstract—**The capability of identifying physical structures in an unknown environment is important for autonomous mobile robot navigation and scene understanding. A methodology for detecting corridor and door structures in an indoor environment is proposed, and the performances of the corridor detection algorithm and door detection algorithm applied in different environments are evaluated. In the proposed algorithms, we utilize a feedback mechanism based hypothesis generation and verification (HGV) method to detect corridor and door structures using low level line features in video images. The proposed method consists of low, intermediate, and high level processing stages which correspond to the extraction of low-level features, the formation of hypotheses, and the verification of hypotheses using a feedback mechanism, respectively. The system has been tested on a large number of real corridor images captured by a moving robot in a corridor. The experimental results validated the effectiveness and robustness of the proposed methods with respect to different viewpoints, different robot moving speed, under different illumination conditions and reflection variations.**

Keywords—**vision based robot navigation, vanishing point, corridor line detection, door detection, hypothesis generation and verification, feedback mechanism**

I. INTRODUCTION

THE challenge of a computer vision system for a mobile robot is to be capable of recognizing natural structures and providing the necessary semantic interpretations of their environment. Typical structures in an indoor environment are visually salient features such as corridors, doors, walls, windows, and pathways etc. Corridors and doors are important visual landmarks which establish connections among different places or regions in an indoor environment. Driven by these needs, we propose two algorithms to identify corridor lines and doors in an indoor environment respectively.

Detection of corridor lines and the vanishing point provide important information for robot navigation. A corridor line is the intersection line between a wall and the floor. Vanishing point is a special perspective projection effect. It is well known that with a pinhole perspective projection model, lines parallel to each other in the 3-D space will converge to a common endpoint, called the vanishing point [1], in the 2-D projection plane. The absolute corridor line locations not only determine the robot's transversal moving range in a corridor, but also provide clues for understanding the image. The knowledge of the vanishing point can be used for many purposes on robot navigation, such as inferring the possible directions of advancement, enhancing further analysis of the scene, localizing the robot's position, or building 3-D structure model of the corridor environment.

Doors are one of the most common landmarks for robot navigation since they show the topological structure in indoor environment and mark the entrance/exit of rooms in many office and laboratory environments. Moreover, it is essential to detect doors in indoor scenes to build a map for the environment.

The problems of corridor line detection and vanishing point estimation have been studied by many researchers. Most methods employed generic preprocessing procedures such as Canny edge detection and Hough Transform algorithm for extracting straight line segments. These methods try to find groups of line segments intersecting at a common point of the projective plane and can be further classified with respect to the clustering approach used by these algorithms [1], [2]. However these approaches were designed for obtaining the vanishing point rather than specifically for detecting accurate corridor line locations. An approach was proposed by Yang and Tsai [3] who described a procedure for deriving the orientation and lateral position of a vehicle by viewing corridors as a combination of right parallelepipeds. The information source was the corridor ceiling. Vassallo *et al.* [4] proposed a simple and fast method for detecting corridor line and calculating the vanishing point. This method works well based on the assumption that there is a good contrast between the floor and walls and there is only one such corridor edge. However, in a typical indoor scene, the corridor environments are often complex with many linear features having similar contrast which are parallel to and close to corridor lines due to edging, patterned tiles and reflective surfaces etc. No solution so far has been reported to tackle such situations. With this motivation we developed a procedure for detecting true corridor edge locations in the presence of many spurious line features.

In literature, many approaches can be found to detect doors for robotics applications. A number of approaches use the ultrasound information [5] or combine the sonar data and visual information for detecting doors and robot navigation [6],[7]. Some approaches combine image processing techniques and neural networks [8],[9], and other approaches were usually developed for specific applications[10]. In our application, the quality of the video images acquired by a moving robot is usually poor and often affects low level feature extraction. Proposed approaches seldom mentioned this practical issue or dealt with such situation. Another concern in our application is to try to reduce the processing time as much as possible. Based on the above analysis, we propose a simple and robust door detection algorithm by employing a feedback control strategy to improve the performance of the low level processing.

Both algorithms are developed under the assumption that the knowledge about the vertical direction in the scene is available. This constraint is reasonable since it is similar to human vision in a natural way. This kind of top-down information can benefit the feature extraction by reducing the amount of unwanted features, increasing the sensitivity to good features, and drastically speeding up the computation.

We utilize feedback control strategy and hypothesis generation and verification (HGV) method to detect corridor and door structures using low level linear features.

II. METHODOLOGY FOR INDOOR STRUCTURE DETECTION

A. Overview of the System

The block diagram of the system is shown in Figure 1. The corridor detection algorithm and door detection algorithm are based on the same methodology and have some common steps at preprocessing and low level processing stages, but differ from each other due to the intermediate- and high-level processing stages.

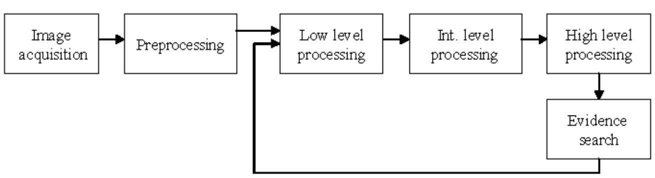


Fig.1. Block diagram of the system

B. Image Acquisition

The video sequences were acquired using a video camera mounted on a mobile robot traversing in typical indoor environments. The video sequences differ from each other due to the environment in which the camera was placed, the moving speed and the field of view of the camera as well as time of the day. The optical axis of the camera is basically parallel to the longitudinal corridor axis to capture video sequences for corridor line detection. The optical axis of the camera is tilted towards left wall or right wall to capture videos for door detection.

C. Preprocessing

The quality of the video images brings additional challenges to the task of low level feature extraction due to vibration of the moving camera, noises introduced by camera when capturing videos as well as the poor resolution and image quality of consumer grade video cameras used in our experiments. To suppress noise while preserving sharp edges, Lee's sigma filter [11] is applied to the input image as the preprocessing step. The sigma filter takes an average of only those neighboring pixels whose values lie within 2δ of the central pixel value, where δ is the sigma parameter found by trial and error for the image.

D. Vertical Line Extraction

This is a common low level processing step for corridor and door detection algorithms. We choose a fast line finder (FLF) algorithm [12], which is a gradient orientation-based and region-based line extraction algorithm first developed by Burns *et al.* [13], to extract vertical lines in the image. This is due to the fact that it is able to extract low contrast long straight lines without post processing. The default eight buckets (a set of ranges) [12] for coarse quantization of gradient direction space are shown in Figure 3. To extract vertical straight lines, only pixels whose gradient direction fall into bucket 1 and bucket 5 need to be processed.

E. Corridor Line Detection Algorithm

In this part we discuss procedures for corridor line detection. The method is implemented based on an assumption that there is relative contrast between the floor and walls.

1. Selection of Points Belonging to Diagonal Lines

Points along corridor lines (diagonal lines) can be selected by choosing pixels having X direction and Y direction gradients above a certain threshold value. This way, edge points lying on vertical or horizontal lines are discarded and only the points belonging to diagonal lines are selected. The relationship between points along diagonal lines and their gradients in x and y directions is shown in Figure 2.

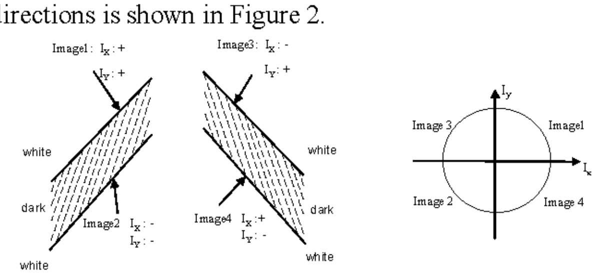


Fig.2. The relationship between points along diagonal lines and their I_x and I_y gradients

The points with positive gradient direction are separated into images 1 and 2, corresponding to the left corridor line; and the points with negative gradient direction are separated into images 3 and 4, corresponding to the right corridor line.

In order to eliminate the spurious edge segments, erosion operation with two structuring elements corresponding to diagonal lines is applied on each group images respectively. In order to improve calculation speed, we only select points belonging to diagonal lines in the bottom half of the image.

2. Corridor Line Hypothesis Generation

Corridor line hypotheses are generated by fitting lines for the points along diagonal lines in the four eroded images. The RANSAC [14] algorithm is used for fitting line models in the presence of many data outliers. In our applications there is one model (inliers set) in some binary images and there may be two models in other images. Therefore, we have modified the RANSAC algorithm for dealing with different kinds of images in a general way. The basic idea is as follows: a) Fit the first model; b) Exit if the percentage of remaining points is less than a ratio; c) Otherwise remove all data points matching the first model; d) Redo RANSAC.

If the RANSAC algorithm fails to fit any lines in the image, the connected component algorithm (CCA) [15] is employed to group adjacent points belonging to diagonal lines into many regions. Then small regions which contain less than a region pixel count threshold are removed from the image. Many outliers can be eliminated by this way.

3. Corridor Line Hypothesis Verification

At this stage, the system is directed back to low level stage to search evidence for verifying the corridor line hypotheses. We choose vertical lines as verification evidence based on the idea that, in a typical corridor scene, vertical lines on the each side of the wall would fall onto corridor lines rather than any other spurious lines. To implement this stage, vertical lines in the image are first extracted at the low level processing stage and then the vertical lines are separated into two groups using the information of vanishing point. Finally, the hypotheses are confirmed or rejected by the number of vertical lines whose endpoints fall onto the corridor lines.

Vanishing point is obtained using the subtractive clustering algorithm (SCA) [16]. The input set of data is the intersection points of each left corridor line hypothesis with each right corridor line hypothesis. The first cluster center is chosen as the estimated vanishing point since it has the highest density of surrounding data points.

F. Door Detection Algorithm

This method is developed to detect any potential door structures on left or right walls when a robot moves forward along a corridor. Thus, we use a sequence of images as input data. Now our algorithm aims to identify visually important doors in the current image frame. The visually important doors are defined as doors which are close to the camera, and can be seen completely (top bars and two boundary lines). In this method, we assume that the door frame (or door panel) has relative contrast with respect to the wall.

We represent door object in terms of its geometric skeleton shape, that is, an upside down U-shape consisting of the top bar, the left boundary line, and the right boundary line. Even through our current work is concerned with door detection, the approach can accommodate a wider range of objects easily by incorporating their geometric skeleton shapes into the system.

Using uniform parameters applied on the whole image for line extraction normally causes two main problems: heavy computational cost and missing information. Thus, a feedback strategy is employed to improve the low level feature extraction using different parameter settings.

The procedure used for detecting door structures can be described as follows:

1. Door Hypothesis Generation

One of the features of the FLF algorithm is its ability to disambiguate lines formed with the different orientations of intensity change. As shown in Figure 4, a line formed due to bright to dark change is put in bucket 1, and a line formed due to dark to bright change is put in bucket 5. Vertical line extraction results labeled with 1 and 5 of two images are illustrated in Figure 5. These two video images are captured from different points of view.

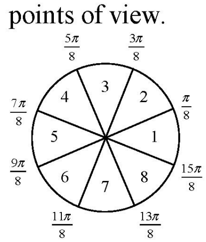


Fig.3. The default 8 buckets for coarse quantization of gradient direction space

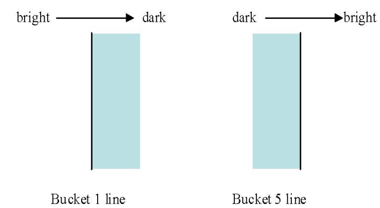
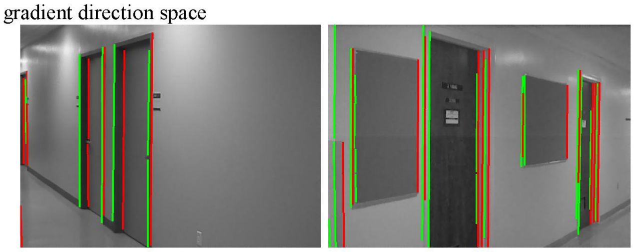


Fig.4. Brightness variation of bucket 1 line and bucket 5 line



(a) (b)

Fig.5. Extracted vertical lines of the image: green lines are labeled with 1; and red lines are labeled with 5 (a) facing to the right wall (b) facing to the left wall.

From line detection images (Fig.5), we found some natures to generate door hypotheses. We establish two rules to group vertical lines into potential boundary lines of doors:

a) Group each 1-labeled line with the 5-labeled lines on its right side to form two boundary lines of a door hypothesis.
b) Reject the line groupings whose distances are less than 2% of the image width or greater than one third of the image width.

2. Information Feedback to Low Level stage to Search the Top Bar

For each hypothesis, a window of interest is generated to search possible top bar edge. The window of interest is a rectangle formed by surrounding the top points of the two boundary lines. The FLF algorithm performs well on extracting straight lines if expected orientations of lines are given. The desired orientation is obtained by the line which is determined by the top points of two boundary lines. Furthermore, parameter setting is suitably adjusted for extracting top bar line segment in each window of interest.

3. Generating U-shape hypotheses

For each available top bar, the distance gap between an end point of the top bar and each vertical boundary line need to be calculated. If distance gap is within a threshold value, the U-shape hypothesis is generated by consisting of this top bar and two boundary lines.

4. Feedback to Detect Corridor Lines

This step is somewhat similar to the corridor location algorithm described above, but contains few modifications due to the different view point of the camera. The process can be summarized as follows: a) Selecting the points that belong to diagonal lines; b) Eroding each image with corresponding structuring element; c) Generating corridor line hypotheses using the RANSAC algorithm; d) Verifying door hypotheses using vertical line evidence.

5. Door Hypothesis Verification

The U-shape hypothesis is verified as a door by checking if the endpoints of the two boundary lines (at least one boundary line) of the U-shape fall onto the corridor line area. In this way, objects similar to door structures such as paintings, posters and boards on the wall can be rejected.

III. EXPERIMENTAL RESULTS

A. Corridor Line Detection

In order to evaluate the performance of the corridor line detection algorithm, we have tested 200 video images acquired from 50 different corridor environments on campus.

1. Illustration of results

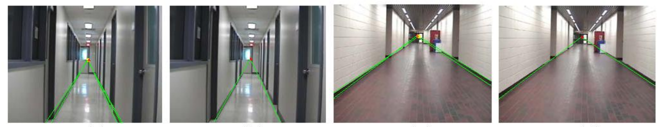
Figure 6 to Figure 9 show the processing results of corridor lines in a variety of corridor environments.

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

Fig.6. Detecting results in a variety of corridor environments (a)(c) Detected vanishing point and generated corridor line hypotheses (b)(d) Detected corridor lines

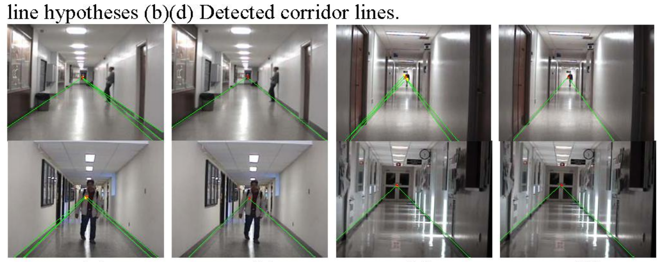
2. Performance Evaluation

Since there have been no attempts reported in the literature to detect true corridor lines in the presence of some spurious lines, there have been no existing methods that quantitatively evaluate this kind of algorithm. Here, we propose two evaluation metrics to compare our results with the ground truth:



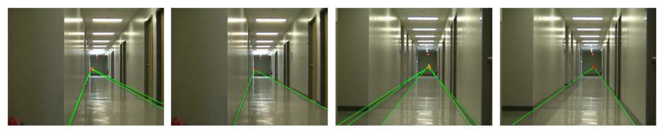
(a) (b) (c) (d)

Fig.7. Detecting results of corridors in wide corridor environment and narrow corridor environment (a)(c) Detected vanishing point and generated corridor line hypotheses (b)(d) Detected corridor lines.



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

Fig.8. Detecting results in the presence of people and strong reflections (the last one) (a)(c) Detected vanishing point and generated corridor line hypotheses (b)(d) Detected corridor lines



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

Fig. 9. Processing results when the robot is positioned in the different places in the corridor: close to the left side of the wall and close to the right side of the wall (a)(c) Detected vanishing point and generated corridor line hypotheses (b)(d) Detected corridor lines.

a) Area deviation: The area deviation is calculated by Eq.1. The area error is calculated as shown in Figure 10.

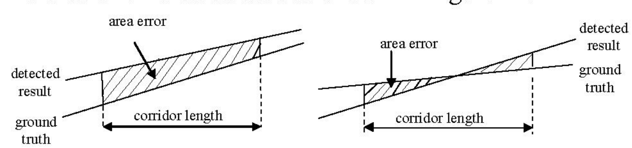


Fig.10. Illustration of the area error between the detected result and the ground truth

$$area\ \ deviation = \frac{area\ \ error}{corridor\ \ length} \quad (1)$$

b) Angular deviation: the angle difference between the detected corridor line and the ground truth.

The ground truth of a corridor line was obtained by manually choosing the corridor line in the image. The numerical results of experiments are illustrated in Table 1. Ideally, the angular deviation should be zero, and the area deviation on each side of the corridor should be zero.

The proposed method has been implemented using C++ with the Intel OpenCV (Open Computer Vision) library [17]. Table 2 reports the processing time for 100 images (with 640×480 pixel resolution) on a laptop with 1.6 GHz Centrino processor and 1 GB memory. The average corridor line detection time for 100 images is about 1 second which meets our real time requirement for robot navigation.

Table 3 summarizes the performance evaluation of vanishing point (VP) for 100 images. The distance error is defined as the Euclidean distance in pixels between the VP ground truth and the processing result. From Table 3, we can see that the detected VP is very close to the true VP. We believe that this error is acceptably small for an indoor robot navigation task and are currently working on utilizing this algorithm for such an application.

Table 1: Comparing detected corridor lines to their ground truth (Processed image size: 640×480 pixels)

Images	Left corridor		Right corridor	
	Angular deviation (degree)	Area deviation (pixels)	Angular deviation (degree)	Area deviation (pixels)
Image 1	0.009	1.859	0	0
Image 2	0.015	1.421	0.023	3.011
Image 3	0.008	1.042	0.006	1.625
Average for 100 images	0.017	5.196	0.021	5.436
Stand deviation	0.026	7.950	0.028	8.191

Table 2: Processing time for corridor line detection (in seconds)

Processing Steps	Image 1	Image 4	Image 5	Average for 100 images	Std. Dev.
Selection of points belonging to diagonal lines	0.01	<0.001	0.01	0.0076	0.0043
Hypothesis generation	0.11	0.291	0.05	0.1340	0.0867
VP estimation	<0.001	<0.0010	<0.001	0.0002	0.0014
Vertical line extraction	1.132	0.6	1.242	0.9361	0.2713
Hypothesis verification	<0.001	<0.001	<0.001	0.0001	0.0010
Total time	1.252	0.891	1.302	1.0782	0.2911

Table 3: Performance evaluation of VP detection for 100 images

	Image 1	Image 3	Image 5	Average for 100 images	Std. Dev.
Distance error(pixels)	1.4569	3.0815	2.0413	3.2146	3.8658

The efficiency, effectiveness and robustness of the corridor line detection method are validated by the above experimental results.

B. Door Detection

In order to evaluate the performance of the door detection algorithm, 40 video images acquired from 5 different corridor environments have been tested. These image frames cover a wide range of situations: doors more or less close to each other, double-leaf doors, open and closed doors, partially occluded doors, with respect to different viewpoints and different lighting and reflection variations.

1. Illustration of results

Figure 11 to 12 illustrate the door detection results in different corridor environments. From these images we can see that some doors far from the camera cannot be detected in the current image frame. However, as the robot moves forward, they can be detected in the subsequent image frames.


(a) (b) (c) (d) (e) (f)

Fig.11. Processing results of a sequence of image frames in one corridor. There are open doors, closed doors, and partially occluded doors in the scene.


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

Fig.12. Detecting results of a sequence of images in a corridor environment. There are posters, double-leaf doors and single-leaf doors in the scene.

2. Performance Evaluation and Discussion

In order to evaluate the performance of the door detection algorithm, we propose two evaluation metrics: one is to evaluate the detection rate based on the most salient door in the image frame; and the other is to measure the detection rate using a door-width threshold.

a) The first evaluation metric is to check if the most salient door is detected or not. The detection rate is defined by Eq.2.

$$detection\ rate = \frac{A}{B} \tag{2}$$

where A is the number of images in which the most salient doors are detected. B is the number of testing images. The detection rate for 40 testing images is 100%.

b) The second metric is to evaluate the detection rate using a door-width threshold. A door whose width is greater than the door-width threshold is counted as a door that the algorithm should be able to detect. Since the evaluation metric would depend on the door-width threshold, we choose three values for the door-width threshold: 2%, 4%, and 5% of the whole image width. The detection rate and the false alarm are defined by Eq. 3 and Eq. 4.

$$detection\ \ rate = \frac{correctly\ \ detected\ \ doors}{actual\ \ doors} \times 100\% \tag{3}$$

$$false\ \ alarm = \frac{false\ \ positive}{actual\ \ doors} \times 100\ \% \tag{4}$$

A summary of the experiment results is shown in Table 4. Table 5 shows the performance comparison of our proposed method with several other methods reported in the literature.

From Table 4 and 5 we can see that our proposed method can detect visually important doors in an image at a very high accuracy rate. Compared with other approaches, our method has a high detection rate as well as a high false alarm. This is due to the fact that the presented work acts as the first processing step for detecting doors in real applications. The output of our method provides potential door information for the robot. For accurately recognize if it is the target door, we need to extract doorplate information for further confirmation or rejection. Therefore, at current stage we don't want to miss detection of any possible door structures.

Table 4: Results of door detection in five corridor environments

Door-width threshold	Actual doors	Correctly detected doors	False positives	Detection rate (%)	False alarm(%)
2%	96	90	5	93.8	5.21
4%	70	69	4	98.6	5.71
5%	56	56	3	100	5.36

Tables 5: Performance comparison with existing methods

Author	Method	Number of images	Detection rate (%)	False alarm (%)
Proposed method (4% of the width)	Feedback based HGV	40	98.6	5.71
Cicirelli *et al.* [8]	Neural network	821	92	1.3
Dedeoglu [10]	Color-blob detector	180	92	3
Carinena [5]	Fuzzy temporal rules	Ultrasound information	91	16

IV. Conclusions

In this paper, we present two algorithms for detecting corridor line and door structures for vision based autonomous mobile robot navigation using a single video camera. We summarize our contributions as follows:

1) Propose a feedback mechanism based hypothesis generation and verification (HGV) method to indoor structure detection applications.

2) Implement a corridor line detection method using the proposed HGV method that can detect corridor line and vanishing point efficiently and robustly.

3) Implement a door detection method using the proposed HGV method that can detect visually important doors with a very high accuracy rate.

4) Explore the performance of these algorithms in a large number of different environments.

As far as we know, the corridor line detection algorithm reported in this paper represents the first attempt to detect corridor lines in the presence of many spurious line features. An issue we need to consider is the case when the vertical line evidences are not available. Further investigations are in the direction of exploring possible approaches that involve searching for extra information using machine leaning based feature detection methods to verify hypotheses.

Our main future research direction is to incorporate the proposed methods into the flexible image analysis framework developed in our lab to make generic structure detection feasible. Another future work for door detection algorithm is to incorporate tracking algorithm in our proposed method to label the recognized doors and keep tracking them in subsequent image frames.

Acknowledgment

We would like to thank National Sciences and Engineering Research Council (NSERC), IRIS Precarn and the University of Western Ontario for the financial support.

References

[1] S. Segvic and S. Ribaric, "Determining the absolute orientation in a corridor using projective geometry and active vision," *IEEE Trans. Industrial Electronics, vol. 48, pp. 696-710, June, 2001.*

[2] E.Lutton, H.Maitre, and J.Lopez-Krahe, "Contribution to the determination of vanishing points using Hough transform, " *IEEE Trans. Pattern Anal. Machine Intell., vol. 16, pp 430-438, Apr. 1994.*

[3] Z.F.Yang and W.H. Tsai, "Viewing corridors as right parallelepipeds for vision-based vehicle localization*," IEEE Trans. Industrial Electronics, vol. 46, pp. 653-661, June, 1999.*

[4] R.F. Vassallo, H.J. Schneebeli, and J. Santos-Victor, "A purposive strategy for visual-based navigation of a mobile robot," *Circuits and Systems. 1998 Proc., pp. 334-337, August, 1998.*

[5] P. Carinena, C. V. Regueiro, A. Otero, A. J. Bugarin, and S. Barro, "Landmark detection in mobile robotics using fuzzy temporal rules," *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 4, pp. 423-435, Aug.2004.

[6] S.A. Stoeter, F. Le Mauff, and N. P. Papanikolopoulos, "Real-time door detection in cluttered environments," in *Proceedings of the 2000 IEEE International Symposium on Intelligent Control* 2000, pp. 187-192.

[7] D. Anguelov, D.Koller, E. Parker, and S. Thrun, "Detecting and modeling doors with mobile robots," in *IEEE International Conference on Robotics and Automation,* vol. 4, 2004, pp. 3777-3784.

[8] G.Cicirelli, T.D'orazio, and A.Distante, "Target recognition by components for mobile robot navigation," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 15, no. 3, pp. 281-297, 2003.

[9] E. Cokal and A. Erden, "Development of an image processing system for a special purpose mobile robot navigation," in *The Fourth Annual Conference on Mechatronics and Machine Vision in Practice* 1997, pp. 246-252.

[10] G.Dedeoglu andG. S. Sukhatme, "Landmark-based matching algorithm for cooperative mapping by autonomous robots," in *Fifth International Symposium on Distributed Autonomous Robotic Systems (DARS 2000).*

[11] J.S. Lee, "Digital Image Smoothing and the Sigma Filter," *Computer Vision, Graphics, and Image Processing 24, pp. 255-269, 1983.*

[12] P. Kahn, L. Kitchen, and E.M. Riseman, "A fast line finder for vision-guided robot navigation," *IEEE Trans. Pattern Anal. and Machine Intell., vol. 12, pp. 1098-1102, November, 1990.*

[13] J. B. Burns, A. R. Hanson, and E. M. Riseman, "Extracting straight lines," *IEEE Trans. Pattern Anal. Machine Intell., vol. 8, pp. 425-455, July 1986.*

[14] R. C. B. M.A.Fischler, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Communications of ACM,* vol. 24, pp. 381-395, June1981.

[15] C. Ronse, *Connected Components Algorithm for Binary Images.* New York: Research Studies Press, 1984.

[16] Chiu, S., "Fuzzy Model Identification Based on Cluster Estimation," *Journal of Intelligent & Fuzzy Systems,* vol. 2, No. 3, Sept. 1994.

[17] "Intel Open Source Computer Vision Library," available from *http://www.sourceforge.net/projects/opencvlibrary.*