An Intelligent System for Learning Environment Models and Guidance Strategies for Visio -Based Indoor Autonomous Land Vehicle Navigation

Guan-Yu Chen (陳冠宇)
Department of Computer and Information Science
Nati nal Chiao Tung Universit
Hsinchu, Taiwan 300
Republic of China
gis81510@cis.nctu.edu.tw

Tien-Yueh Chiang (江頂岳)
Department of Computer and Information Science
National Chiao Tung Universit
Hsinchu, Taiwan 300
Republic of China

Wen-Hsiang Tsai (蔡文祥)
Department of Computer and Information Science
National Chiao Tung Universit
Hsinchu, Taiwan 300
Republic of China
whtsai@cis.nctu.edu.tw

Abstract

An intelligent system for learning environment models and guidance strategies for vision-based autonomous land vehicle (ALV) navigation in indoor environments is proposed. In the learning process, the ALV is firstly driven manually by an operator through the navigation environment. Then, a navigation model, which consists of the locations of environment features, refined sub-paths, and their corresponding guidance strategies, is generated automatically by the learning system. The selection of th guidance strategy depends on the availability of stable environment features. An intelligent navigation scheme by integrating three guidance strategies is proposed for safe ALV navigation through environments consisting of various conditions. The learned model can be used to guide the ALV through the explored environment by the proposed navigation scheme. When the tested environment changes, the new environment can be re-learned, and the proposed navigation scheme still can work in the new environment without any manual adaptation. The proposed approach has been tested on a prototype ALV and many successful navigation sessions have been per-formed, which confirm the feasibility of the proposed approach.

Keyword: autonomous land vehicle, intelligent learning, line following, model matching, dead reckoning.

1. Introduction

Autonomous land vehicles (ALV's) have attracted intensive research efforts in recent years because of its versatile applications and the fast development of computer techniques. With recent developments of computer vision techniques, ALV systems can perform many tedious or dangerous tasks such as exploration of unknown environments, working in nuclear plants, safety guarding, document delivery, unmanned transportation, house cleaning, etc. ision-based integration of various sensin and guidance techniques for ALV navigation in natural environments is a challenging task because of the variety of environment structures. An intelligent ALV navigation system should be able to integrate all available useful information and adjust its guidance strategies in different environment conditions. Useful environment information includes baselines in straight corridors, pillars in lobby sections, shapes of doors, or vertical lines of buildin structures, etc.

Lines, points, and corners are commonly used visual features. In some navigation environments, for example, a long corridor of a structured building or the scene of a highway, feature lines are parallel to the navigation path. In such environments, line following techniques are widely used for ALV guidance. Model matching is another popular guidance approach. With model matching, the precise position and orientation of the ALV can be obtained. However, it can be employed only when visual features are abundant. Another guidance approach is dead reckoning. It is used when insufficient visual features are available. By combining the above three techniques, the tasks of ALV guidance in general structured indoor environments can be completed in a single guidance scheme. The first goal of our study is to develop such an integrated scheme.

An environment model is required for any model-based navigation system. However, the traditional method of establishing environment models - manual measurement of the navigation environment - is a time-consuming work. It is thus desired to design a system for automatic learning of navigation environments. Several environment learnin systems were developed in recent years [1-7] to meet this requirement. Lebègue and Aggarwal [1,2] developed an integrated system to generate architectural CAD models using a mobile robot. The system consists of a segment detector, a tracker, and a CAD modeler. Ishiguro et al. [3] presented a strategy for establishing the model of an unknown environment by a mobile robot. Panoramic sensing was used to perceive the structure of the environment in their implementation. Kurz [4] introduced an approach to generating environmental maps based on ultrasonic range data. Free-space can be partitioned into situation areas by means of a learning classifier. Then the situation areas are attached to graph nodes b dead-reckoning and finally a map of the free-space in the form of a graph representation is generated. Dean et al. [5] formulated map learning as a problem of inferring the structure of a reduced deterministic finite automaton from noisy observations and provided an exploration algorith to learn the correct structure of the automaton. Pan and Tsai [6] proposed an integrated approach to automatic model learning and path generation for vision-based ALV guidance in building corridors. In Chen and Tsai [7], an incremental environment learning system for ALV navigation was proposed. Rough initial en vironment models are constructed first, and after each navigation session, the proposed system can update the environment model according to the information collected in the previous navigation.

All of the above environment learning systems provide schemes to build environment models of different feature types. Environment features are recorded in the learned models, but no information for guidance strategies is included. In this study, a new system, which not only can learn environment features but also guidance strategies, is proposed. That is, in the learning process, not only the information of environment features is collected, but als the guidance strategy for each navigation section is decided Navigation information, including sub -paths and their corresponding guidance strategies, are (with environment features) recorded in a model, called *navigation model.* With the learned model and the proposed integrated navigation scheme, the ALV can navigate automaticall along the planned path in the learned en vironment by three different guidance strategies.

To achieve this goal, the learning system was designed t handle the following tasks. Firstly, it collects desired environment features; Secondly, it decides the guidance strategy by using the collected inf ormation, including the locations and the numbers, of the local environment features; Thirdly, it divides the navigation path into several sub-paths by the policy of "one sub -path, one guidance strategy"; and finally it adjusts the sub-paths to fit the requirement of its corresponding guidance strategy.

The remainder of this paper is organized as follows. In Section 2, the proposed learning procedures for intelligent navigation are described. In Section 3, the three guidance strategies and the techniques fo r integration of consecutive guidance strategies are presented. In Section 4, several experimental results are presented. Finally, some conclusions are given in Section 5.

2. Proposed Learning System

In this section, the proposed system for learnin environment models and guidance strategies is described in detail.

2.1 Principles of Proposed Learning Algorithm

In the proposed system, computer vision techniques are employed to locate environment features. The selected environment features are the baselines o n the building wall. On the abstract level, the visual features can be categorized into three classes, straight lines, corners and end points of line segments, as shown in Figure 1. The visual features are first found by image processi ng. Then by computer vision techniques, the location of the features are calculated. model matching algorithm for line segments and corners is proposed to find the correspondence between the sensed local model and the learned global model (see Section 2. 2 for the detail). The matching results then are used to locate the ALV and construct environment models.

Unstable environment features, which usually result from image processing error and image projection, cause bad matching results and misguide the ALV. Thus, the learnin system should be able to distinguish stable and unstable features. In the proposed learning system, an environment feature is said to be a *stable feature* when it is detected in two or more learning cycles. The work of checking the multi-occurrence of each environment feature is accomplished in the process for merging the local model t the learned one.

To learn in a navigation process, the ALV is firstly driven manually by the operator along a pre -selected path meanwhile, the grabbed i mage of the environment scene and the vehicle control data (i.e., the moving distance and the turn angle of the front wheels) are recorded. Then, an off-line learning procedure is performed, which consists of three phases. In the first phase, the environme nt model is established by the procedures mentioned previously, and stable and unstable features are separated. In the second phase, for each learning cycle, a planned sub-path is initialized to be identical to the path the ALV traveled in the corresponding learning cycle of the first phase. Then the local model corresponding to the current cycle is matched to the learned global model to determine the numbers, types and positions of stable features in the local model. According to the condition of the environment features, one of the three guidance strategies is selected for guiding the ALV through the sub-path for navigation session in the future. The criteria for selecting the guidance strategy will be described later. In the third phase, successive sub -paths with the line-following guidance strategy, if existing, are merged into a single sub-path. The safe distance between the refined line -following sub-paths and the line segment feature is assigned to be the distance between the last sub-path of those bei ng merged and the corresponding line segment feature.

2.2 Guidance Strategy Selection and Learning Algorithms

The adoption of appropriate guidance strategies in different environment sections depends on the types of the stable features that can be extract ed from environment images during ALV navigation. For example, baselines are common stable features in straight corridor sections, and corners and end points of the baselines are general stable features in turning sections and lobby space inside structured buildings. Sometimes, no stable feature can be found during navigation. In such cases, a guidance strategy not depending on visual features is required. In our study, the dead reckoning approach is used for blind navigation.

More specifically, the criteria for selecting the guidance strategy for sub-paths are summarized as follows. If there are two or more stable environment features in the local model, the model matching strategy is selected as the guidance strategy of the sub -path. If there are only line segments nearly parallel to the sub -path in the local model, the line following strategy is selected for sub -path. And if no stable feature is found in the local model, the dead reckoning strategy is selected for the sub -path.

The entire learning algorithm can be summarized as follows.

Algorithm 1 Intelligent learning of ALV navigation.

Step 1. Perform camera calibration.

Step 2. Drive the ALV manually along the pre-selected path and grab the image of the current environment scene.

Step 3. Record the environment image and the control data.

Step 4. Establish the environment model (see Algorithm 2 for the detail).

Step 5. For each learning cycle, in addition t recording the positions of the start and the end of the sub-path, select a guidance strategy for use in the future to guide the ALV through the corresponding sub -path according to the numbers, types, and positions of stable features detected in the current cycle, as described previously.

Step 6. Adjust the learned sub -paths. Successive sub-paths with the line-following guidance strategy are merged into a single sub -path.

The algorithm for establishing the environment model is as follows

Algorithm 2 Learning of environment models.

Step 1. Set the initial global model as empty.

Step 2. Extract environment features from the captured image.

Step 3. Calculate the estimated position and orientation of the ALV, then calculate location of the extracted environment features, and set up a local model by collecting the extracted local features.

Step 4. If the global model is non-empty, match the local model with the global model with the proposed matching scheme (described in Section 2.4) and recalculate the accurate position of the local features by the matchin result.

Step 5. Attach the local model to the global model.

Step 6. For each learning cycle, repeat Steps 2 through 5.

2.3 Estimation of ALV Location

The estimated new position and orientation of the ALV are calculated as follows. When the ALV moves away from a known position, the new position of the ALV c an be estimated with the moving distance and the turn angle of the front wheels. The derivations of the equations t calculate the estimated ALV location can be found in [6] and are reviewed in the following. As shown in Fig ure 3, assume that the vehicle is located at A. After moving a distance S forward, the vehicle will be at a new location B, which is the desired estimated ALV location. Let the relative location of B with respect to A be denoted by a vector **T**. The rotation radius R can be written as

$$R = \frac{d}{\sin\delta}, \tag{1}$$

where d is the distance between the front wheels and the rear wheels, and δ is the turn angle of the front wheels. And the angle γ can be determined as

$$\gamma = \frac{S}{R}. \tag{2}$$

So, the length of vector **T** can be solved to be

$$\|\mathbf{T}\| = R\sqrt{2(1-\cos\gamma)}, \tag{3}$$

and the direction of vector **T** is

$$u = \frac{\pi}{2} - \delta - \frac{\gamma}{2}. \tag{4}$$

The coordinates of location B in the vehicle coordinate system with respect to location A can thus be computed b

$$x_B = \|\mathbf{T}\|\cos u$$
$$y_B = \|\mathbf{T}\|\sin u. \tag{5}$$

After the front wheel location of the ALV is determined, the rear wheel location $(\tilde{x}_B, \tilde{y}_B)$ of the ALV can also be determined to be

$$\tilde{x}_B = x_B + d\sin\gamma$$
$$\tilde{y}_B = y_B - d\cos\gamma. \tag{6}$$

Since the global coordinates of location A are known, and since the vehicle coordinates of location B with respect t location A can be obtained from Eq. (5), the global coordinates of location B can be calculated by co ordinate system transformations. Thus the desired estimated ALV location is obtained.

2.4 Matching Algorithms for Corners and Line Segments

The proposed matching algorithm is designed to find the translation and the rotation between the input model and the global model. The proposed algorithm is a two-phase one. The rotation between the two models is found in the first phase. The information used for the matching algorithm in this phase is the slope angles of the line segments. Then the input model is transformed with the obtained rotation angle. In the second phase of matching, the translation between the two models can be obtained. The information used for the matching algorithm in this phase is the position of the corners and end points.

The matching algorithm to find the rotation between the two models is based on the following idea. To find the rotation between models, each of which consists of only one line segment, the most trivial approach is to compare the slope angles of the line segments. For model s consistin of more than one corner or line segment, it might no work to find a matched pair of line segments or corners to obtain the rotation angle by simply comparing the slope angles of them. In the proposed algorithm, a statistical approach is used to find the rotation. The distributions of the slope angles of the line segments (or the branches of the corners) are firstly generated as a histogram. The magnitude of each the histogram value is proportional to the sum of the lengths of the line segments with a certain slope angle. Then, by comparing the two histograms, the rotation between the two models can be obtained. In practice, the work to compare the two histograms is implemented in a way of finding the rotation angle as a shift angle, which results from the maximum correlation values. Furthermore, for error tolerance, a low-pass filter is applied to both histograms before the correlation values are calculated. An illustration for this phase of the proposed matching algorithms is shown in Figure 2. Two environment models are shown in the top-left corner and their correspondin slope-angle histograms are shown in the bottom of the figure. In the upper slope -angle histogram, there are two peaks around 90 and 180 (or 0) degrees. The se peaks correspond to the slope angles of the line segments of the left model. Similarly, in the lower histogram, there are two peaks around 95 and 5 degrees, corresponding to the slope angles of the line segments of the right model. The chart of the correlation value versus the rotation angle is shown in the top-right corner of the figure. The rotation angle corresponding to the maximum of the correlation value is 5 degrees. It is concluded that the rotation between the two models is 5 degrees.

Once the rotation angle between the two models is obtained the input model is then rotated with the obtained rotation angle. After the rotation, a revision of the corner -matching algorithm proposed by Pan and Tsai [6] is performed to find the best m atch pair of corners (or end points). The distance between the two corners (or end points) is then regarded as the translation between the input and the global models.

3. Navigation by Integrating Different Guidance Strategies

After the learning stage, the ALV may conduct navigation session with the learned model and strategies. The navigation techniques using the three strategies are described in detail in this section.

3.1 Navigation by Dead Reckoning

When no desired visual feature can be extracted stabl y from the grabbed images, the ALV is steered to follow a pre-defined navigation path. The approach to follow the pre-defined path by the dead reckoning strategy is described as follows. Firstly, calculate the desired wheel turn angle by an exhaustive sear ch approach, as described later. Secondly, move the ALV with a given moving distance and the obtained wheel turn angle. Thirdly, obtain the real moving distance and the real wheel direction from the odometer and the feedback of the control unit. The real moving distance and the real wheel direction might not be identical to the desired ones owing to the existence of possible control error and imprecision. Fourthly, calculate the new position and orientation of the ALV with the data from the odometer and the feedback of the control unit (see Section 2.1 for the detail). Finally, repeat the above steps t complete another dead reckoning navigation cycle.

The driving wheel direction δ can be calculated by the following wheel adjustment strategy. The basic idea is to search a turn angle of the front wheels to drive the ALV as close to the desired path as possible. As shown in Figure 4, given a path P, either a straight line or a circular segment, define $D_P^F(\delta)$ as the distance from the midpoint between the two ALV front wheels to the given path P after the ALV traverses a certain distance S with the turn angle δ, where S may be assigned to be the average navigation distance during a cycle. Define $D_P^B(\delta)$ as the distance from the midpoint between the two ALV back wheels to the given path P. Also, define Q as the nearest point on P to the estimated ALV position. Define $A_P(\delta)$ as the angl between the head direction of the ALV and the tangent of P on Q. Finally, define measure L_P to be

$$L_P(\delta) = \left(D_P^F(\delta) + D_P^B(\delta)\right)/C_1 + A_P(\delta)/C_2, \qquad (7)$$

where C_1 and C_2 are two pre-selected constants. To find the turn angle of the front wheel to drive the ALV as close to the path as possible, an exhaustive search is performed to find the angle that produces the minimal value of L_P. The obtained angle is used as the turn angle for safe navigation.

3.2 Navigation by Line Following

The line following guidance strategy is used only when stable line features can be extracted from grabbed images. After a line is extracted, the line equation with respect t the vehicle coordinate system is derived by computer vision techniques [8, 9]. With a given safe distance to the extracted line feature, a line following navigation path, which is parallel to the line feature, can be obtained. By the exhaustive search approach proposed in Section 3.1, the desired wheel turn angle can be calculated. The ALV then navigates to the next position with the desired driving wheel direction and starts another navigation cycle.

3.3 Navigation by Model Matching

If corners and the end points of the baselines are available in the grabbed image, the model matching guidance strategy is used to derive a more precise ALV position and orientation. The basic steps of the model matchin guidance strategy are described as follows. Firstly, extract the local environment model from the grabbed image and calculate the location of the environment features with respect to the ALV [7, 8]. Secondly, match the local model to the learned global model and find the translation and rotation between the two models by the matching result. Thirdly, calculate the position and the orientation of the ALV with respect to the learned global model [8]. Finally, calculate the desired wheel turn angle by the approach proposed in Section 3.1 and drive the ALV with the obtained wheel turn angle.

The algorithms of model matching depend on the selection of the environment model. For example, the Generalize Hough Transform (GHT) [10] is a well-known point-pattern-matching scheme. A model-matching scheme for corners and line segment features was proposed by Pan and Tsai [6]. In this study, the environment model consists of corners and end points of line segments. An algorithm t match such a type of models was proposed in Section 2.4. The matching algorithm finds the best match pair, and then calculates the translation and rotation between the two models by the matching result.

3.4 Criteria for Switching Guidance Strategies

The major issue arising in the integration of guidance strategies for ALV navigation is the criteria for switching of the guidance strategies. In the proposed approach, the criteria are as follows.

1. If the ALV is in a line -following session: The guidance strategy will be switched to model matchin if the following conditions are satisfied: (1) the end of the line is being approached ; (2) the next sub-path requires the model matching strategy; and (3) one or more stable corners or end points, i.e., features with multi-occurrences, are available in the sensed local model. It is mentioned by the way that the guidance strategy is never s witched to dead reckoning from line-following because the end points of the followed lines, which facilitates model matching, can always be detected before the lines disappear.
2. If the ALV is in a model -matching session: The guidance strategy will be switched to line following if the following three conditions are satisfied: (1) the end position of the sub-path of the model matching session is being approached; (2) the next sub -path is a line-following session; and (3) line features nearly parallel to the next sub-path are available. It will be switched to be dead reckoning if (1) the end position of the sub-path of the model matching session is reached and (2) the next sub -path to be followed is a dead-reckoning session.
3. If the ALV is in a dead reckoni ng session: The guidance strategy will be switched to model matchin if the following conditions are satisfied: (1) the end of the line is being approached; (2) the next sub -path requires the model matching strategy; and (3) one or more stable corners or e nd points are available in the sensed local model. The guidance strategy will be switched to line following if the following three conditions are satisfied: (1) the end position of the sub-path of the model matching session is bein approached; (2) the next sub-path is a line-following session; and (3) line features nearly parallel to the next sub-path are detected in the sensed image.

If insufficient visual features are extracted during the model-matching or line-following navigation sessions, the guidance strategy will automatically switch t dead-reckoning temporarily until sufficient visual features are found in the grabbed images. If no sufficient visual features are found within a certain number of cycles, the ALV will stop automatically to ensure safety.

4. Experimental Results

The external view of the prototype of the ALV is shown in Figure 5(a). The ALV is computer-controlled with a modular architecture, as shown in Figure 5(b), including four major components, namely a vision system, a central processing unit (an Intel Pentium II 450MHz PC), a motor control system, and a DC power system. The vision system consists of a camera, a color monitor, and an image frame grabber. The motor control system consists of a main control box with a controller, a motor driver, and two motors.

The image processing works for extracting line segments and corners in sensed images are accomplished in two phases. In the first phase, two consecutive pixels with gray values smaller than a pre-selected threshold value indicate a possible candidate pixel on a baseline. Then, we check in a downward direction to see if there exist five consecutive pixels with their gray value smaller than the threshold value We also do this in the upward direction to see if there exist five consecutive pixels with their gray value larger than the threshold value. If the two constraints are satisfied, the current pixel is regarded as a baseline candidate point. In the second phase, an algorithm similar to the edge-linking algorithm is performed to form a set of line segments and corners from the candidate set. The equations of the baseline segments are computed by *least-mean-square-error* (LMSE) fitting [11]. An example of image processin results is shown in Figure 6. Detected baselines are shown in white line segments and detected corners and end points are shown in white spots.

The ALV learning and navigation experiments were performed in a building corridor in National Chiao Tun University. By using the proposed approach, man successful navigation sessions have been conducted. An example of the learned navigation models is shown in Figure 7. The environment features are shown as heavy black lines from the top view, and the planned sub-paths are shown as lines with different gray levels, depending on their corresponding guidance strategies. Sub -paths corresponding to dead reckoning, line following, and model matching guidance strategies are shown as light gray, dark gray and black ones, respectively.

5. Conclusion

In this paper, we proposed a system for learning environment models and guidance strategies for ALV navigation. The system not only collects the information of the environment features to build up an environment model but also selects a guidance strategy for each of the navigation sub -paths according to the encountered environment conditions. Furthermore, an intelligent navigation scheme by integrating three guidance strategies is proposed for safe ALV navigation through environments consisting of various conditions. The proposed learning and navigation system has been implemented on a prototype ALV and successful navigation sessions in indoor corridor environments confirmed the feasibility of the approach.

This research is supported under the project NSC-88-2213-E-009-114 of National Science Council, epublic of China.

References

[1] X. Lebègue and J. K. Aggarwal, "Extraction and Interpretation of Semantically Significant Line Segments for a Mobile Robot," *Proc. of 1992 IEEE Intl. Conf. on Robotics and Automation* , Nice, France, pp. 1778-1785, May 1992.

[2] X. Lebègue and J. K. Aggarwal, "Generation of Architectural CAD Models Using a Mobile Robot," *Proc. of 1994 IEEE Intl. Conf. o n Robotics and Automation*, San Diego, California, U.S.A. , Vol. 1, pp. 711-717, May 1994.

[3] H. Ishiguro, T. Maeda, T. Miyashita and S. Tsuji, " Strategy for Acquiring an Environmental Model with Panoramic Sensing by a Mobile Robot," *Proc. of 1994 IEEE Intl. Conf. on Robotics and Automation* , San Diego, California, U.S.A., Vol. 1, pp. 724-729, May 1994.

[4] A. Kurz, "Constructing Maps for Mobile Robot Navigation Based on Ultrasonic Range Data," *IEEE Trans. on System, Man, and Cybernetics - Part B: Cybernetics*, Vol. 26, No. 2, pp. 233-242, April 1996.

[5] T. Dean, D. Angluin, K. Basye, S. Engelson, L. Kaelbling, E. Kokkevis, and O. Maron, "Inferrin Finite Automata with Stochastic Output Functions and an Application to Map Learning," *Machine Learning*, Vol. 18, pp. 81-108, 1995.

[6] F. M. Pan and W. H. Tsai, "Automatic environment learning and path generation for indoor autonomous land vehicle guidance using computer vision techniques", *Proc. of 1993 Nationa l Computer Symposium*, Chia-Yi, Taiwan, Republic of China, pp. 311-321, 1993.

[7] G. Y. Chen and W. H. Tsai, "An Incremental-Learnin -by-Navigation Approach t Visio -Based Autonomous Land Vehicle Guidance in Indoor Environments Using Vertical Line Information and Multi-Weighted Generalized Hough Transfor Technique", *IEEE Trans. on System, Man and Cybernetics – Part B: Cybernetics*, Vol 28, No. 5, pp. 740-748, Oct. 1998.

[8] S. D. Cheng and W. H. Tsai, "Model-based guidance of autonomous land vehicle in indoor environments by structured light using vertical line information," *Proc. of 1991 Workshop on Computer Vision, Graphics and Image Processing*, Tainan, Taiwan, Republic of China, pp. 340-345, Aug. 1991.

[9] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision, Volume 2*, Addison-Wesley, Reading, MA, U.S.A., 1993.

[10] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognition*, vol. 13, no. 2, pp. 111-122, 1981.

[11] L. L. Wang and W. H. Tsai, "Car safety driving aided by 3-D image analysis techniques," *MIST Technical Report, TR-MIST-86-004*, National Chiao Tun University, Hsinchu, Taiwan, R.O.C., 1986.

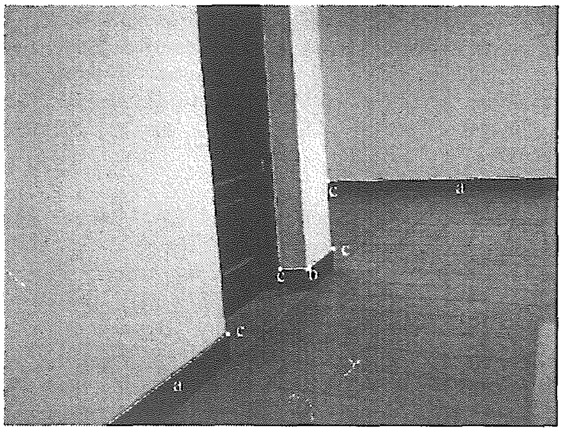


Figure 1. Three types of the environment features: (a) baseline (b) corners (c) end points of the baselines.

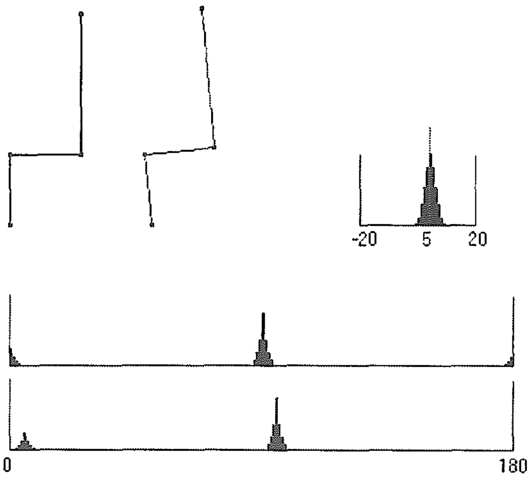


Figure 2: Illustration of the proposed matching algorithm.

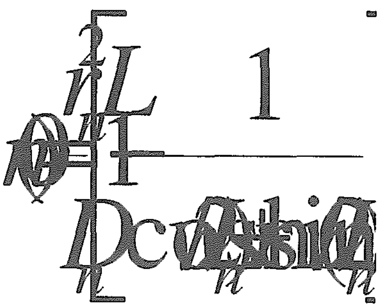


Figure 3: The vehicle location before and after the ALV moves a distance S forward.

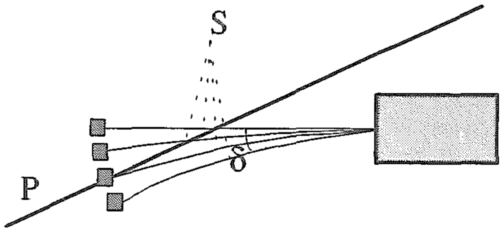
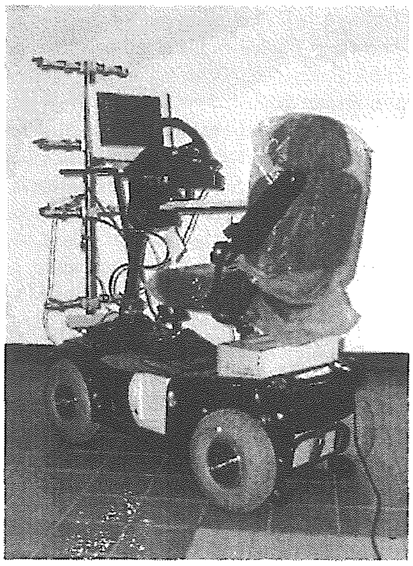
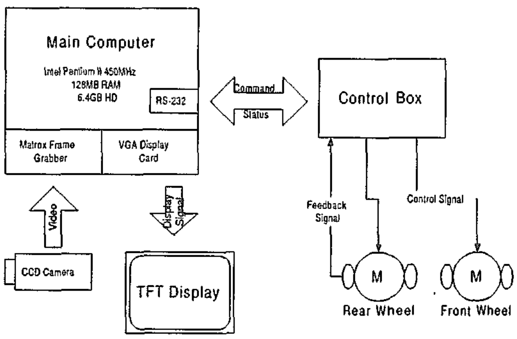


Figure 4: Illustration of adjustment of the front wheels in a path.



(a)



(b)

Figure 5: The prototype ALV used in the experiments. (a) External view. (b) System structure.

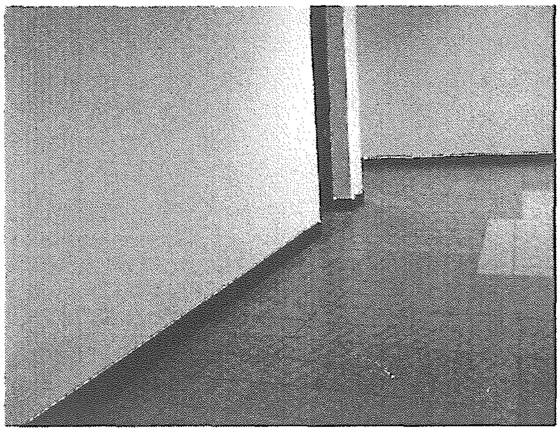


Figure 6. An example of the image processing results. Detected baselines are shown in white line segments and detected corners and end points are shown in white spots.

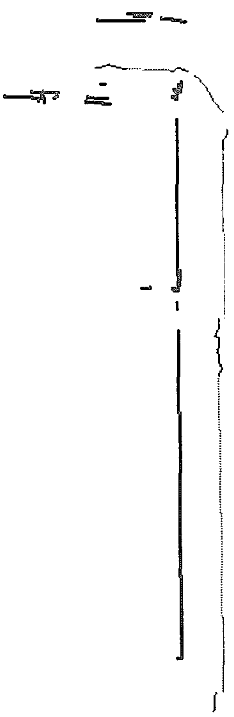


Figure 7. An example of the learned navigation model. Environment features are shown as thick black line segments, and the planned sub -paths are shown as thin lines segments.