An Integrated Computer Vision and Infrared Sensor Based Approach to Autonomous Robot Navigation in an Indoor Environment

Sambit Bhattacharya, Bogdan Czejdo and Sean Mobley
Department of Mathematics and Computer Science
Fayetteville State University
Fayetteville, NC 28301
{sbhattac, bczejdo, smobley} @ uncfsu.edu

ABSTRACT

We present a novel approach to robot navigation in an indoor environment based on analysis of single lens camera images and infrared sensor readings. Our software design is based on the state diagram methodology where the robot transitions between states when a change in the environment, known as trigger, is detected. These environmental triggers are based on computer vision techniques, namely, (1) the Hough Transform for lines along with K-Means clustering used to compute orientation points for the robot and (2) histogram difference measurement for classification of objects (wall or door) the robot sees through its camera. We further use the infrared sensor to trigger obstacle avoidance behavior. By putting all these elements together within a state diagram framework we have created a platform for indoor robot navigation that is robust and easy to adapt to different topological layouts of indoor environments.

Keywords autonomous robot navigation, computer vision, Hough transform, K-Means clustering, histogram difference measurement, state diagram methodology.

1. INTRODUCTION

The problem of robot navigation is an active area of research which has yielded a variety of experimental approaches which can be broadly classified as indoor and outdoor navigation [2]. Indoor environments like corridors have regular geometric features that can be captured by the robot's camera and effectively analyzed to navigate the robot.

In this paper we present our approach to solving the indoor navigation problem through a state diagram based program design and combination of computer vision algorithms with infrared sensor information.

The organization of the paper is as follows: we first describe the computer vision techniques we use along with experimental data. Next we describe environmental triggers based on these techniques. Finally we use examples to describe the use of these triggers within the state diagram framework to program autonomous robot navigation.

2. IMAGE ANALYSIS FOR INDOOR NAVIGATION

Our approach uses the Hough transform for detecting line segments and histogram based difference calculation methods for discriminating between corridor features.

.

2.1 Hough Transform

The Hough transform is widely used in computer vision for detecting regular geometric features such as line segments and circles in images. Our system uses an efficient version of the Hough transform known as the Progressive Probabilistic Hough Transform [4] for detecting naturally occurring lines in indoor images of corridors and hallways. Before applying the Hough Transform we perform Canny edge detection on the corridor images to obtain binary images where edge pixels are white and all other pixels are black.

The Hough line transform assumes that any point in a binary image could be part of some set of possible lines. As a result of applying the Hough Transform we detect almost all lines in the image along with a few false positives. Our next step is to reduce the set of lines to two major clusters which represent the direction of the corridor. In order to do that we first eliminate the approximately vertical and horizontal lines from the set. Next, we apply the K-Means algorithm [5] to cluster the lines into two groups as shown in the example in Figure 1 (E) and then we take the average representative line for each cluster (as shown in Figure 1 (F)). In the final step the intersection of these two representative lines yields the orientation point for the robot moving in the corridor.

Figure 1: the images are numbered with letters A to F in the left to right, top to bottom ordering. (A) and (B) are examples of corridor images with door on the left; (C) and (D) are examples of corridor images with wall on the left. (E) Shows image (A) processed with our Hough Transform program; the detected lines are colored white and superimposed on the original image. (F) Shows the result of clustering all lines in (E) into two major diagonal groups; the average line representing each group is shown; their point of intersection is the point of orientation for the robot when moving across the corridor.

2.2 Histogram based difference measurement

In the course of indoor robot navigation a significant problem is recognition of major landmark objects [2] such as doors and wall surfaces between them. Landmarks are used in most aspects of navigation. If a robot finds a landmark in the world and that landmark appears on a map, then the robot is localized with respect to the map. If the robot plans a path consisting of segments, landmarks are needed so the robot can tell when it has completed a segment and another should begin.

(A) EMD	Wall Sample 1	Wall Sample 2	Door Sample 1	Door Sample 2
Wall Sample 1	0	23.51	67.20	72.79
Wall Sample 2	23.51	0	90.27	95.86
Door Sample 1	67.20	90.27	0	7.18
Door Sample 2	72.79	95.86	7.18	0

(B) Corr. Coeff.	Wall Sample 1	Wall Sample 2	Door Sample 1	Door Sample 2
Wall Sample 1	1	0.13	-0.06	-0.21
Wall Sample 2	0.13	1	-0.16	-0.20
Door Sample 1	-0.06	-0.16	1	0.72
Door Sample 2	-0.21	-0.20	0.72	1

Table 1: Histogram based differences calculated between wall and door samples using the Earth Movers Distance (EMD) (table 1 (A)) and the correlation coefficient (table 1 (B)). Here the door samples are thin vertical images slices taken from Figure 1 (A) and (B) and wall samples are similarly taken from Figure 1 (C) and (D).

Our system recognizes wall surfaces and doors through histogram based techniques. These recognition techniques create the trigger conditions that we describe later in our state diagram framework. Our problem is the following: sample the current camera image so that the program is able to decide whether the robot is currently beside a wall or a door. We solve this problem by analyzing the left and right most vertical slices of the image; then the program classifies the slice as "wall" or "door" based upon histogram difference measures. We have tested two widely used histogram difference measuring techniques, namely the Earth Mover's Distance (EMD) [3] and the statistical technique of calculating the correlation coefficient between two distributions.

Our measurement process compares new samples with stored representative histogram patterns for wall and door. An example test result is summarized for both techniques in Table 1 where images shown in Figure 1 are being used for data. We clearly see the significant differences in EMD values for wall-to-door comparison (high positive) and small differences in EMD values for wall-to-wall and door-to-door comparison (zero or low positive). We also see the significant differences in correlation coefficient values for wall-to-door comparison (negative) versus wall-to-wall and door-to-door comparison (one or positive). Our program therefore uses these techniques to classify whether the robot is beside a door or wall.

3. ENVIRONMENTAL TRIGGERS BASED ON COMPUTER VISION AND INFRARED SENSORS

The Hough transform and histogram analysis can produce triggers reflecting the environment characteristics. Let us first discuss the environmental triggers based on histogram analysis. When the robot moves beside a door and the edge of the camera detects the door pattern then the trigger condition DoorPatternDetected becomes True otherwise the condition is False. Similarly, when the robot moves beside a wall and the edge of the camera detects wall pattern then the trigger condition WallPatternDetected becomes true.

The other triggers e.g. MiddleOfDoorDetected can be derived from these basic triggers. When the trigger WallPatternDetected changes from False to True a robot odometer is initialized and the trigger MiddleOfDoorDetected becomes False. When the value of that odometer is equal or greater than half of the width of the door, then the trigger MiddleOfDoorDetected becomes True.

Let us now discuss the environmental triggers based on Hough transform. The OrientationPointToLeft and OrientationPointToRight triggers are computed based on the algorithm for orientation point described in the previous section. The triggers are listed in Table 2 that includes the Obstacle trigger typical for infrared sensors.

Trigger name	Based on
Obstacle	Infrared Sensor
DoorPatternDetected	Histogram Analysis
WallPatternDetected	Histogram Analysis
MiddleOfDoorDetected	Histogram Analysis and odometer
OrientationPointToLeft	Hough Transform
OrientationPointToRight	Hough Transform

Table 2: List of advanced visual sensor triggers

4. STATE DIAGRAM BASED AUTONOMOUS NAVIGATION

We assume that the robot's environment is known in advance. Under this assumption the robot behavior can be modeled by a state diagram [6, 7, 8, 9] that has different states depending on the location of the robot in the corridor.

Generally, the state diagram in addition to states has transitions consisting of triggers that cause the transition of the robot from one state to another, and actions, that are invoked during a transition. Triggers are expressed by Boolean conditions evaluated continuously to respond to changes in environment. To specify state diagrams we used the notation based on Universal Modeling Language (UML) [10, 11] where a state is indicated by a box and a transition is indicated by an arrow with a label. The first part of the transition label (before the slash) specifies the trigger and the part after the slash specifies the action to be invoked during the transition [1].

The typical simple mobile robot behavior includes states such as Moving Forward, Moving Backward, Stop, Turning Right, and Turning Left. Some of these simple states are also included in the state diagram in Figure 2. The most interesting, however are advanced states that are related to advanced image analysis.

Let us convert a specific corridor map into a state diagram for the movement of our robot until it reaches the goal. The state diagram is shown in Figure 2. In brief, the robot behavior in our specific example can be described as follows:

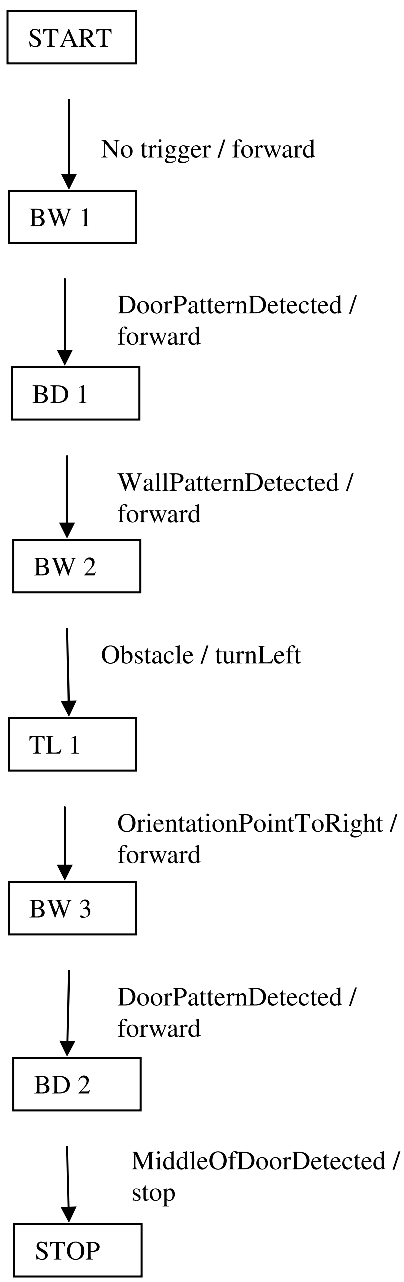

Figure 2: State diagram representation of our specific autonomous navigation example.

1. Initially the robot is in the START state and with no trigger, the transition to the BW 1 (where BW is abbreviation for Beside Wall) state takes place immediately. During the time of transition the forward command is executed. This built-in command engages the motors to move the robot forward.

2. The robot stays in the BW 1 state (and continues moving forward) until it detects a door pattern and the trigger condition DoorPatternDetected becomes True. Detecting the door pattern triggers the transition to the BD 1 (where BD is abbreviation for Beside Door) state while moving forward.

3. The robot stays in the BD 1 state until it detects a wall pattern and the trigger condition WallPatternDetected becomes True. Detecting the wall pattern triggers the transition to the BW 2 state.

5. The robot stays in the BW 2 state (and continues moving forward) until it detects an obstacle (which is the turning point of the corridor) and the trigger condition Obstacle becomes True. Detecting the obstacle initiates the turnLeft operation and transition to the TL 1 (where TL is abbreviation for Turning Left) state. The turnLeft command engages the motors to start turning the robot counter clockwise. Similar to the forward command, the turnRight command causes the motors to be engaged until any new movement command deactivates the older settings.

6. The robot stays in TL 1 state (and continues turning) until the robot is aligned to the orientation point and the trigger condition OrientationPointToRight becomes True. OrientationPointToRight becoming True means that the direction the robot is facing just becomes aligned with the orientation point. This trigger causes the invocation of the forward command and transition to the BW 3 state.

7. The robot stays in the BW 3 state (and continues moving forward) until it detects door pattern and the trigger condition DoorPatternDetected becomes True. Detecting the door pattern triggers the transition to the BD 2 state while moving forward.

8. The robot stays in the BD 2 state (and continues moving forward) until it is in the middle of the door and the trigger condition MiddleOfDoorDetected becomes True. Detecting the middle of the door triggers the stop operation and transition to the STOP state. Arriving at this door was the goal of the robot.

5. GOAL CHANGES IN AUTONOMOUS NAVIGATION

The goal assigned to robots can change depending on situation and new requirements. The existing diagrams can be used for rapid prototyping and implementation of new robot behavior. Let us discuss how to modify diagram from Figure 2 to implement a new robot behavior.

Let us assume that the new goal is to enter the first door. The new corridor map is now significantly simplified and includes only the beginning of the corridor and the first door on the left. This map can be converted into a state diagram for the new movement of our robot.

In order to do it, new environmental triggers need to be defined and developed. The new triggers Door OrientationPointToLeft and Door OrientationPointToRight are computed based on the hybrid analysis combining Hough Transform with histogram analysis. More specifically, the algorithm for DoorOrientationPointToLeft and Door OrientationPointToRight selects vertical lines separating surfaces based on histogram based differences and computes the orientation point as a point in the middle of these lines. The new triggers are listed in Table 3.

Trigger name	**Based on**
Door OrientationPointToLeft	Hough Transform and Histogram Analysis
Door OrientationPointToRight	Hough Transform and Histogram Analysis
MiddleOfDoorDetected	Odometer

Table 3: Additional visual sensor triggers.

Once these new visual sensor triggers are developed, the state diagram can be created as in Figure 3. In brief, the robot behavior in our specific example can be described as follows:

1. Initially the robot is in the START state and with no trigger, the transition to the BW 1 (where BW is abbreviation for Beside Wall) state takes place immediately. During the time of transition the forward command is executed. This built-in command engages the motors to move the robot forward.
2. The robot stays in the BW 1 state (and continues moving forward) until it detects a door pattern and the trigger condition DoorPatternDetected becomes True. Detecting the door pattern triggers the transition to the BD 1 (where BD is abbreviation for Beside Door) state while moving forward.
3. The robot stays in the BD 1 (and continues moving forward) until it is in the middle of the door and the trigger condition MiddleOfDoorDetected becomes True. Detecting the middle of the door triggers the turnLeft operation and transition to the BD 2 state.

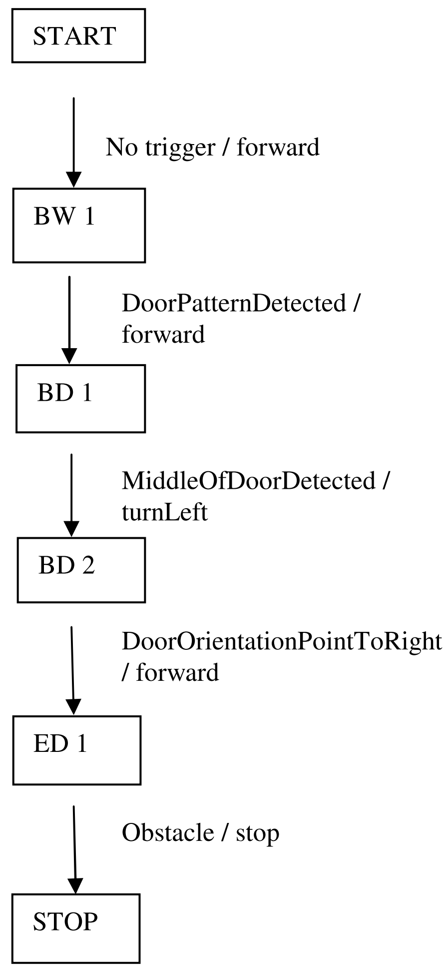

Figure 3: State diagram representation of our modified autonomous navigation example.

4. The robot stays in the BD 2 state (and continues turning left) until it is aligned to the middle of the door point and the trigger condition Door OrientationPointToRight just becomes True. True means that the direction the robot is facing exactly towards the middle of the door. This trigger causes the invocation of the forward command and transition to the ED 1 (where ED is abbreviation for Entering Door) state while moving forward.

5. The robot stays in the ED 1 state (and continues moving forward) until the Obstacle trigger becomes True. Detecting the obstacle triggers the stop operation and transition to the STOP state.

6. CONCLUSION

We have presented a novel approach for autonomous robot navigation in a known indoor environment using computer vision techniques and infrared sensor. Triggers were developed for state diagram based design of our program to detect and respond to changes in the environment. For future directions we will perform research on more advanced image analysis algorithms for better navigation in more complicated environments such as cluttered indoor spaces.

ACKNOWLEDGMENT

We acknowledge the Belk foundation and the Institute for Personal Robotics in Education (IPRE) for supporting this research.

REFERENCES

[1] Czejdo, B. D. and Bhattacharya, S., "Programming robots with state diagrams", Journal of Computing Sciences in Colleges, Volume 24 , Issue 5, 19-26 (2009).

[2] Desouza, G.N. Kak, A.C., "Vision for mobile robot navigation: a survey", Pattern Analysis and Machine Intelligence, IEEE Transactions on, Volume 24, Issue 2, 237-267 (2002).

[3] Rubner, Y., Tomasi, C. and Guibas, J. L., "The Earth Mover's Distance as a Metric for Image Retrieval", International Journal of Computer Vision, Volume 40, Number 2, 99 – 121, 2000.

[4] Galamhos, C. Matas, J. Kittler, J., "Progressive probabilistic Hough transform for line detection", Conference on Computer Vision and Pattern Recognition, 1999, IEEE Computer Society. Volume 1, 560 – 566, 1999.

[5] Bradley, P. S. and Fayyad, U. M., "Refining initial points for k-means clustering", Proc. of 15th International Conf. on Machine Learning, 91-99 (1998).

[6] Baszun, M., Miescicki, J., and Czejdo, B., "Multilevel modeling of iterative engineering design of remote robot system", Proceedings of The Second World Conference on Integrated Design and Process Technology, Austin, 1996.

[7] Czejdo, B. and Messa., K., "Generating Smalltalk code from graphical operations and rules", Proceedings of the IEEE Symposium on Visual Languages, Bergen, Norway, 1993.

[8] Czejdo B., Daszuk W. and Miescicki J., "Concurrent software design based on constraints on state diagrams", Proceedings of The Third World Conference on Integrated Design & Process Technology, Berlin, Germany, 1998.

[9] Embley, D., Kurtz, B., Woodfield, S., Object-Oriented Systems Analysis – a model driven approach, Prentice Hall, New Jersey, 1992.

[10] Harel, D., "On visual formalisms", Communications of the ACM, 31, (5), 514-530, 1988.

[11] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., Lorensen, W., Object-Oriented Modeling and Design, Prentice Hall, New Jersey, 1990.