A Goal Oriented Navigation System Using Vision

Mehmet Serdar Güzel[1], Panus Nattharith[2], Ahmet S. Duran[1]
[1]*Computer Engineering Department, Ankara University, Turkey.*
[2]*Department of Electrical and Computer Engineering,*
Faculty of Engineering, Naresuan University, Thailand.
panusn@nu.ac.th

***Abstract*—This paper addresses a goal oriented navigation framework in a behavior-based manner for autonomous systems. The framework is mainly designed based on a behavioral architecture and relies on a monocular vision camera to obtain the location of goal. The framework employs a virtual physic based method to steer the robot towards the goal while avoiding unknown obstacles, located along its path. Simulation results validate the performance of the proposed framework.**

***Index Terms*—Goal-Oriented Navigation; Mobile Robots; Monocular Vision; Behavioral Design.**

I. INTRODUCTION

Over recent years, there has been an explosive growth of interest in using vision for goal oriented systems in mobile robot navigation. Most of the goal oriented systems utilize metric or topological maps of the environment to navigate, providing the exact knowledge of the environment [1], [2]. Vision is mainly integrated into goal oriented systems either to recognize landmarks or goals observed in an image. For landmark based systems, localization is essential, which includes four steps, namely image acquisition, landmark detection, matching and calculate position. Details of these steps and can be seen in [3]. Besides, there are methods to be able to detect landmarks on the 2-D image, and track them in the consecutive scenes. Landmarks can be artificial or natural. In both cases, the robot needs to recognize the landmarks in order to be able to track them [4]. A good example of natural landmark tracking-based navigation can be seen in [5].

Vision based robot navigation systems that do not require a metric or topological map are called as mapless navigation systems which are able to navigate by extracting relevant information about the landmarks in the environment. Optical flow is one of the main methodologies used for vision based mapless navigation problem. Optical flow is considered as a 2D vector field describing the apparent motion of each pixel in successive images in a 3-D scene [6]. Several optical flow based navigation strategies using more than one camera have been introduced by researchers; biologically inspired behaviors, based on stereo vision have been adapted for obstacle avoidance. Despite their simplicity, optical flow based navigation systems are mainly designed for vision based obstacle avoidance problem and are sensitive to lighting conditions of the environment. Alternatively, appearance-based methods are employed to navigate robots. The main idea lies behind these methods to store images of the working environment and associate these images with commands to navigate the robot towards a final goal [7], [8]. Feature based methods are also widely used in the corresponding field that the trajectory and motion of the robot is primarily determined by matching the features of reference and the current images during the navigation task [9], [10].

Conventional AI approaches in robotics are limited due to their computational complexity. Alternatively, Behavioral robotics has gathered lots of attention from the researchers, interested in the field. This concept is inspired by the idea that the intelligence has to be studied in terms of a robot interacting with its environment. According to this approach, the navigation task is divided into multiple independent task-achieving modules or behavior [11]. These modules are responsible for a subtask of the whole action. The Subsumption Architecture is a milestone in the control of behavior-based robotics which mainly decomposes complicated behaviors into many primitive behavior modules. Each layer implements a particular goal and subsumes that of the underlying layers [11]. This paper proposes a simple but efficient vision based goal-oriented navigation framework for mobile robot navigation problem.

The framework proposed in this paper is designed based on a behavioral architecture and employs a vision-based goal recognition system. The vision system mainly aims to detect the goal and estimate its distance to the goal. Once the goal is determined, the robot is steered towards the goal using a virtual physic based method in which the goal is associated with attractive forces and the obstacles or other robots are repulsive forces [12]. The main advantage of this method is smoothly converts the entire sensory input space into actuators space without requiring extra rule. This paper is organized as follows. In Section 2, the design of the behavior-based mobile robot system is presented. Section 3 provides the implementation of the behavior-based robot and the experiment results from simulation. The study is concluded in Section 4.

II. GOAL-ORIENTED NAVIGATION FRAMEWORK

This section introduces the proposed goal-oriented framework that the flowchart of the system is illustrated in Figure 1. According to the which, the system primarily searches for the goal with a random walking behavior; once the goal is found the distance estimation module determines the distance and absolute location of the goal. Those position data are passed to the behavioral architecture so as to steer the robot towards a predefined goal safely and efficiently.

A. Goal Estimation

This section addresses the goal estimation methodology used by the proposed framework. A vision based approach, employing an appearance based methodology is adapted, is integrated into the framework. The corresponding algorithm is given in Figure 2.

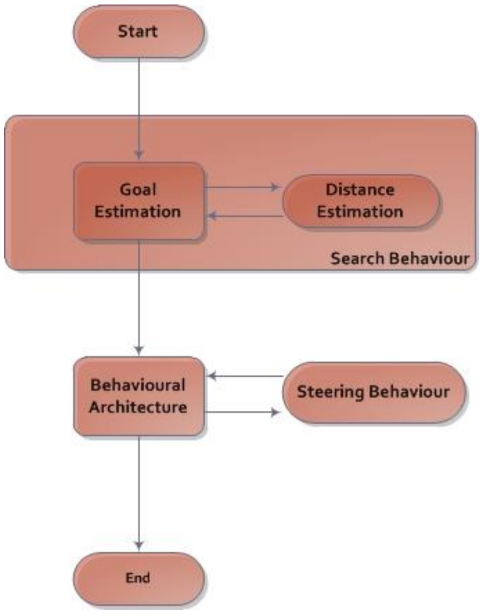

Figure 1: Flowchart of the proposed framework

Distance Estimation Algorithm:
Require: *RGB scale image, physical scale of the object, pixels number and focal length*
Ensure: *Physical distance*

Procedure *Compute Distance*
Initial assignment for the parameters
Load image and convert RGB to HSV color space
Apply low pass filter for noise reduction
Apply morphological operator (dilate and erode)
Until the end of the iterations
Read image pixel by pixel
Compute contours in image
Select the maximum contour
Estimate the rectangular area having the maximum contour
End_until
Compute the distance considering the object size, camera parameters and contour size
EndProcedure

Figure 2: The Algorithm

B. *Behavior Architecture*

The framework is designed based on a behavioural architecture, where the robot is steered towards a predifined position in a smooth and safe maner. The fundemental appraoch lies behind the proposed behavioual architecture is to adapt the conventional potential fields methods to handle the goal-oriented navigation task. The method consists of different behaviours and is able to arrange speed and orientation of a moving robot towards a goal position [12]. Each behavior generates a desired output vector. For instance, consider 'GoTo' behaviour that is assigned the task to steer the robot to an prefinied goal. The output of this is a vector that leads the robot toward the goal. Alternatively, 'AvoidObstacle' behavior that is assigned the task of avoiding the robot from obstacles located in its path. The output of this vector, on the other hand, prevents the robot from colliding obstacles placed on its path. The main idea is to combine attractive 'GoTo' and repulsive 'AvoidObstacle' behaviors to lead the robot towards a goal smoothly while avoiding obstacles. The details of the potential field algorithm used in this paper can be seen in [13]. Finally, multiple potential fields are combined by adding them together, show as follows:

$$\Delta_x = \Delta_x O + \Delta_x G \quad (1)$$

$$\Delta_y = \Delta_y O + \Delta_y G \quad (2)$$

The potential field method is generated by first calculating the attractive forces ΔxG, ΔyG provided by the goal and then finding, ΔxO and ΔyO, the vector generated by the repulsive forces. Finally, these vectors are added together using the aforementioned equations.

Figure 2: Behavioral design of the system

The interaction diagram of the corresponding behaviors is illustrated in Figure 2, inspired from Brooks' subsumption architecture that there exists a priority between behaviors and higher levels are able to subsume lower levels [11]. For instance, Urgency Maneuver behavior has the highest priority and employs an intelligent low level range finder based avoidance algorithm to avoid collision [14]. This behavior is designed to compensate the shortage of the potential field method as local minima and dynamic obstacles. Goal Seek behavior, defined in Section 2.1, randomly navigates robots while searching the goal via vision based algorithm. Once the goal is found, MoveTo behavior is enabled, which utilizes the potential field algorithm and navigates towards the goal while considering obstacles located on its path.

III. EXPERIMENTAL RESULTS

The system has been implemented using ROS and Gazebo framework. Gazebo is a simulator originally designed to accommodate robot simulations in 3-D outdoor and indoor environments. Accordingly, OpenCV, computer vision toolbox, has been integrated into this simulator in order to overcome camera based navigation problem. The primary system environment of the simulator for the given problem is illustrated in Figure 3.

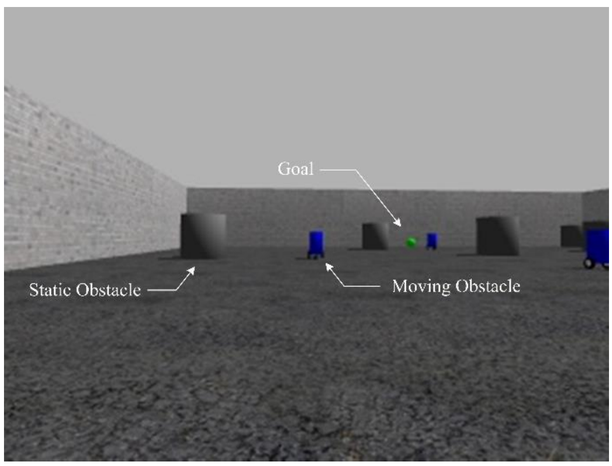

Figure 3: Gazebo simulation environment

The system has been tested under different experimental conditions. Static and/or dynamic obstacles are located into the simulation environment in order to increase the overall challenge of the proposed system. Moreover, a multi-robot based scenario has been designed so as to assess the performance of proposed framework in swarm robotics. Three scenarios are illustrated in this section in order to reveal the overall capacity of the proposed framework. Constants used by potential field method ware shown in Table 1.

Table 1
Constants for potential field used in experiments

Constant Name	Value
Goal Radius	0.5
Goal Spread	10.0
Repellent Radius	0.5
Repellent Spread	0.75
Attaction Alpha	2.0
Repulsion Beta	20.0

The test scenarios are given as follows:

A. Scenario 1

The robot is located at position (x=-6.0, y= -7.0) with 90 degrees' orientation and aims to reach the goal (green ball) while avoiding static obstacles. The scenario is illustrated in Figure 4 and Figure 5 displays the navigation characteristic of the robot using the force applied during the running cycle. According to which, the robot detects the goal at the beginning of the simulation and moves towards the goal within the attractive force until it encounters with obstacles (see 40th cycle) and then rearranges its direction between 50th and 60th cycles. Afterwards, the robot moves toward the goal within a stable force and handles the obstacles again (between 80th and 100th cycles). Finally, the robot moves to the goal with a decreasing attractive force due to the reduction between the goal and the robot and completes the task successfully.

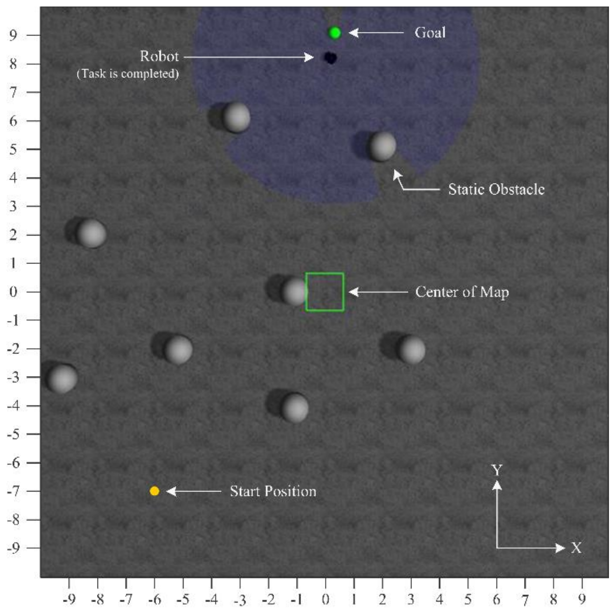

Figure 4: Scenario 1 running on Gazebo

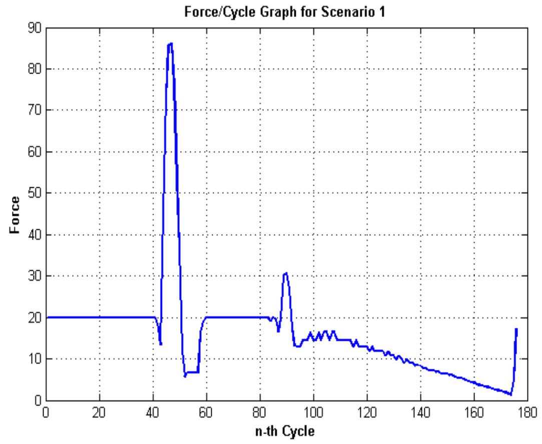

Figure 5: Force-Cycle Graph for Scenario 1

B. Scenario 2

The robot starts running at (x=-6.0, y= -7.0) with 90 degrees' orientation and aims to reach the goal (green ball) while avoiding both static and dynamic obstacles (see Figure 6). Once the goal is detected, the robot moves to the goal with an attractive force. It passes a static obstacle (shown between 75th and 90th cycles) successfully and handles a moving (dynamic) obstacle (encountered at the 160th cycle) and finally completes the task successfully, as shown in Figure 7.

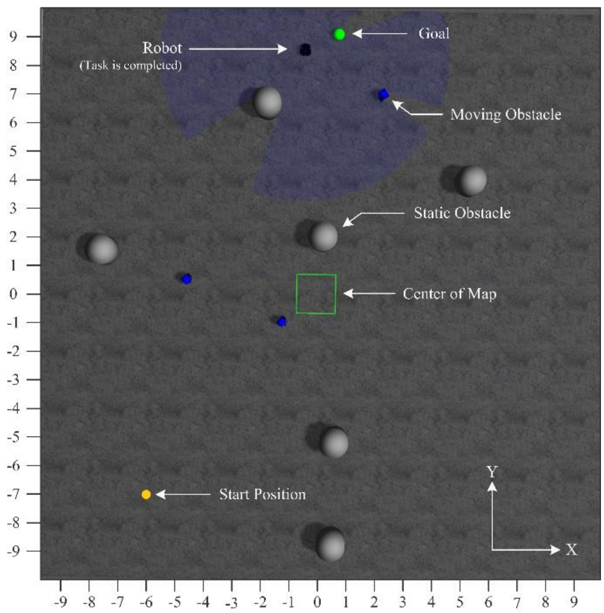

Figure 6: Scenario 2 running on Gazebo

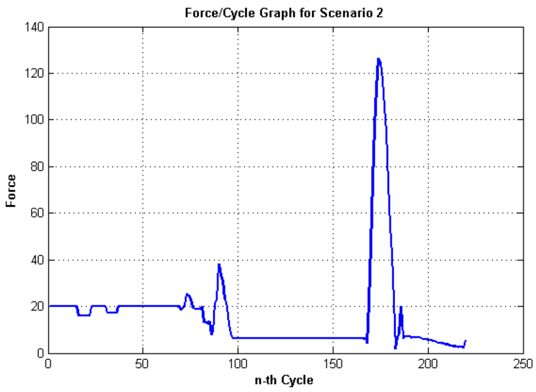

Figure 7: Force-Cycle Graph for Scenario 2

C. Scenario 3

This scenario defines a multi-robot navigation task that the first robot is located at (x=-6.0, y= -7.0) with 90 degrees' orientation and the second one is located at (x=6.0, y= -7.0) with the same orientation. Both of the robots start moving simultaneously to the goal (green ball) while avoiding static obstacles, as illustrated in Figure 8. First robot randomly searches for the goal and detects the goal at the 7th cycles where the attractive force increases from 0 to 20. It passes a static obstacle between the 40th and 60th cycle. The robot moves to the goal with a decreasing attractive force after the 80th cycle due to the reduction of distance between the goal and the robot and completes the task successfully (see Figure 9). The second robot, on the other hand, searches for the goal and detects it at the 5th cycle, the robot encounters with two successive static obstacles and defeats them as shown in peak points at the corresponding Figure. Afterwards it approaches the goal in a stable manner and achieves the task as expected (see Figure 10).

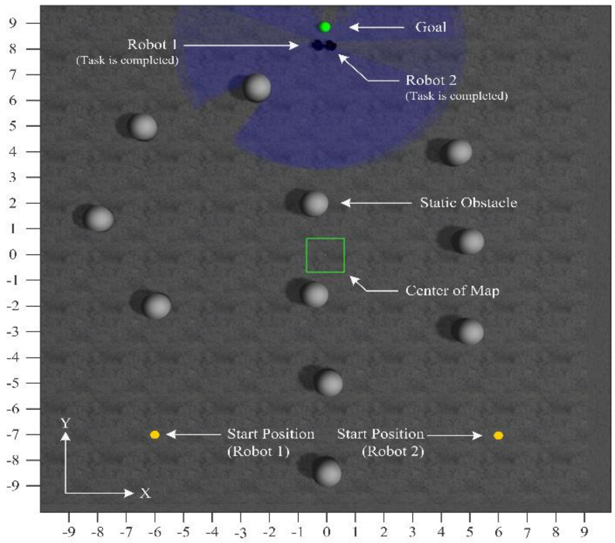

Figure 8: Scenario 3 running on Gazebo

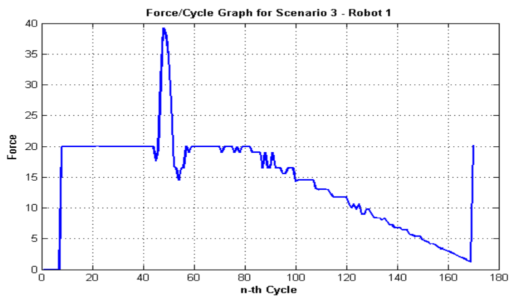

Figure 9: Force-Cycle Graph for Scenario 3 (Robot 1)

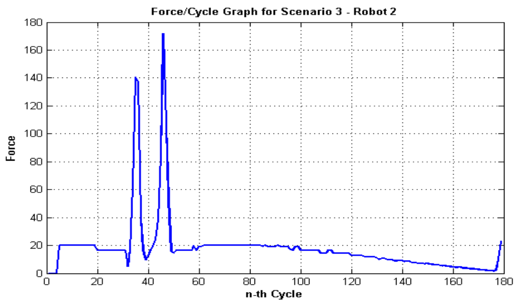

Figure 10: Force-Cycle Graph for Scenario 3 (Robot 2)

IV. Conclusion

This paper introduces an efficient goal-oriented navigation framework for mobile robot navigation. The system employs an appearance based algorithm to detect the goal with a monocular camera. Once the distance of goal is detected, the robot is headed towards the goal direction via a local navigation strategy based on the potential field method. The local navigation strategy is designed based on a behavioral architecture, inspired from the conventional Subsumption architecture that the framework allows a robot to be able to seek a predefined goal, and moves towards this goal while avoiding both static and dynamic obstacles successfully. The framework has been applied to both single robots and extended to multi-robot application, as detailed in the experimental section. The results verify that the proposed framework provides an appropriate and alternative solution to the goal based navigation problem.

References

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

[2] Guzel M. S., 2013. Autonomous vehicle navigation using vision and maples strategies: A survey, *Hindawi Publishing Corporation Advances in Mechanical Engineering*: 1-10.

[3] Hashima M., Hasegawa F., Kanda S., Maruyama T. and Uchiyama T., 1997. Localization and obstacle detection for a robot for carrying food trays, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robot and Systems (IROS '97)*: 345-351.

[4] Sawasaki N. Morita T., and Uchiyama T., 1996. Design and implementation of high-speed visual tracking system for real-time motion analysis, *Proceedings of the IAPR International Conference Pattern Recognition*, 3: 478-483.

[5] Kabuka M. R. and Arenas A. E., Position verification of a mobile robot using standard pattern, *IEEE Journal of Robotics and Automation*, 3(6): 505-516.

[6] Guzel M. S. and Bicker R., 2010. Optical flow based system design for mobile robots, *Proceedings of the IEEE International Conference on Robotics, Automation and Mechatronics (RAM '10)*: 545-550.

[7] Bonin-Font F. Ortiz A., and Oliver G., 2008. Visual navigation for mobile robots: a survey, *Journal of Intelligent and Robotic Systems*, 53(3): 263-296.

[8] Ulrich I. and Nourbakhsh I., 2000. Appearance – based obstacle detection with monocular color vision, *Proceedings of the AAAI National Conference on Artificial Intelligence*.

[9] Szenher D. M., 2008. Visual homing in dynamic indoor environments [Ph.D. thesis], *University of Edinburgh*, Edinburgh, UK.

[10] Harrell R. C., Slaughter D. C., and Adsit P. D., 1989. A fruit – tracking system for robotic harvesting, *Machine Vision and Applications*, 2(2): 69-80.

[11] Brooks, R. A., 1986. Robust Layered Control System for a Mobile Robot, *IEEE Journal of Robotics and Automation*, RA-2: 14-23.

[12] Arkin R. C., 1998. Behavior – Based Robotics, *MIT Press*, Cambridge, Massachusetts

[13] Brooks, R. A., 1999. Cambrian Intelligence: The Early History of the New AI. *MIT Press*, Cambridge, Massachusetts.

[14] Guzel, M. S. and Nattharith, P., 2014. A New Technique for Distance Estimation using SIFT for Mobile Robots, *International Electrical Engineering Congress (iEECON 2014)*.