

Visual navigation and obstacle avoidance using a steering potential function

Wesley H. Huang^{a,*}, Brett R. Fajen^b, Jonathan R. Fink^a, William H. Warren^c

^a *Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180, United States*

^b *Department of Cognitive Science, Rensselaer Polytechnic Institute, Troy, NY 12180, United States*

^c *Cognitive and Linguistic Sciences, Brown University Providence, RI 02912, United States*

Received 6 January 2005; received in revised form 20 November 2005; accepted 21 November 2005

Available online 3 January 2006

Abstract

Humans have a remarkable ability to navigate using only vision, but mobile robots have not been nearly as successful. We propose a new approach to vision-guided local navigation, based upon a model of human navigation. Our approach uses the relative headings to the goal and to obstacles, the distance to the goal, and the angular width of obstacles, to compute a potential field over the robot heading. This potential field controls the angular acceleration of the robot, steering it towards the goal and away from obstacles. Because the steering is controlled directly, this approach is well suited to local navigation for nonholonomic robots. The resulting paths are smooth and have continuous curvature. This approach is designed to be used with single-camera vision without depth information but can also be used with other kinds of sensors. We have implemented and tested our method on a differential-drive robot and present our experimental results.

© 2005 Elsevier B.V. All rights reserved.

Keywords: Local navigation; Obstacle avoidance; Vision-based navigation; Potential fields

1. Introduction

Local navigation is a fundamental problem for mobile robots operating in real-world environments: a robot must make progress towards a goal location while avoiding unexpected obstacles. Usually, robots make use of active sensors such as SONAR and scanning laser rangefinders. However, we know through our own personal experience that it is possible to navigate locally using only vision. Transferring this capability to robots has been difficult.

Though a rich sensing modality, vision has a number of shortcomings as a sole sensing modality for obstacle avoidance. Three-dimensional scene recovery using stereo vision requires two cameras. Camera calibration is important for getting accurate results, and the computational cost is high. It is possible to recover depth to obstacles using a single camera, but this requires either structure from motion techniques [26] (which are also computationally expensive) or a ground plane assumption (e.g., [11]).

We propose a method for obstacle avoidance that uses a single camera without recovering depth. One solution to this problem is to compute optical flow in order to estimate the “time to impact” which can then be used to steer around fast-approaching obstacles [1,4,19]. In our approach, however, we rely only upon the headings to obstacles and their angular widths: features that can be extracted with relative ease and good accuracy. Our approach to obstacle avoidance requires only this information, but if the distances to obstacles are available (even coarse depth information), through vision or some other sensing modality, it can be easily incorporated.

Our method is based on Fajen and Warren’s model of human navigation [7,8,29]. This model takes the relative heading and distance to the goal and obstacles and computes an angular acceleration to steer the robot towards the goal and away from obstacles. There are many advantages to the general framework of this model. Because it controls angular acceleration, it is directly applicable to steering nonholonomic vehicles: the resulting paths are smooth and have continuous curvature. The paths also have a “natural look”, as one would expect from a model based on human navigation. The method is completely reactive and is simple and fast to compute.

* Corresponding author.

E-mail address: whuang@cs.rpi.edu (W.H. Huang).

However, Fajen and Warren’s model is not directly suitable for vision-based robot navigation for several reasons: (1) obstacles are treated as points with no extent, (2) knowledge of obstacle distance is used, and (3) collisions with obstacles are possible.

We have adapted and extended Fajen and Warren’s model to accommodate obstacles of finite width, to use the angular width of an obstacle instead of its distance, and to use speed control to guarantee that the robot will not collide with an obstacle. The resulting model retains the advantages of being reactive, simple to compute, and directly applicable to controlling nonholonomic robots, while now being suitable for obstacle avoidance on a mobile robot.

In our study of Fajen and Warren’s model, we found that it can be interpreted as following a potential field over the robot heading angles. The goal corresponds to an attractor that pulls the robot heading towards the goal, and obstacles correspond to repellers that push the robot heading away. However, this potential field is quite unlike the traditional potential fields in robotics. Since the heading to the obstacles and the goal changes as the robot moves, the potential landscape also changes; for traditional potential fields, the potential is static and is fixed in the world frame. In addition, local minima in our potential are not the problem that they are for traditional potential fields — any local minimum in our potential will steer the robot around the obstacles (robot and obstacle widths notwithstanding) and will eventually become a global minimum! We have found the potential field interpretation of Fajen and Warren’s model to be more easily understandable and to provide greater insight into the operation of this model. We present our extensions to the model in this paradigm, and one of our extensions depends on the potential values.

After a review of related work in Section 2, we describe Fajen and Warren’s model in detail in Section 3, both in the original acceleration formulation and in our potential field formulation. In Section 4, we describe our extensions to this model. We have implemented and tested our model on a differential drive robot; details of our experiments and results appear in Section 5.

2. Related work

There have been many different approaches to local navigation for mobile robots. Early obstacle avoidance work used SONAR sensors almost exclusively; today, scanning laser rangefinders are often used. Most of the earliest obstacle avoidance methods were based on the potential field approach, so we begin our review of related work there. While potential fields are still used for local navigation, there are a number of other commonly used techniques which we review next. Finally, we review some work from the dynamical systems approach to behavior, which was the inspiration for the Fajen and Warren model.

2.1. Potential fields

The potential fields approach to local navigation was developed by Khatib [14] in the 1980’s to create real-time

systems. With the limited computational resources in the early years of robotics, the prevalent approach of modeling (i.e., map building) and geometric motion planning simply required too much computation to operate in real time. Potential fields provide a simple and easily computed method to reach a goal location while avoiding unknown obstacles.

One shortcoming of this approach is that a robot can get stuck in a local minimum of the potential. A variety of approaches have been proposed for a robot to find its way out of these spots, including active search, backtracking, and random walks [6,18]. Methods have also been proposed that use a navigation function [20] (a local-minimum-free potential) on a local map [3,15].

Another problem with potential fields is that they specify a direction that the robot should move, and this may not respect nonholonomic constraints, e.g., for a car-like or differential-drive robot. There have been several approaches to this problem: deforming the potential field to respect the nonholonomic constraints [23], using control techniques to track the path produced by the potential field [12,16], and using a potential field that respects the nonholonomic constraints [25, 5]. In addition, there are methods for tracking a preplanned path that make use of potential fields for obstacle avoidance, e.g., [17].

One variation of the potential field approach that uses the bearing, distance, and orientation of an obstacle was proposed by Khatib and Chatila [13]. They compute a force from a traditional potential field but then modify this force to control the rotation of the robot, resulting in more direct paths when obstacles do not impede the path to the goal.

2.2. Obstacle avoidance methods

There have been many other approaches to local navigation that have departed from the potential field approach.

Borenstein and Koren’s *vector field histogram* (VFH) method [2] transforms a local map into a one-dimensional discretized “polar obstacle density” function. This density function is explicitly searched for a heading close to the goal heading where there is low (or zero) obstacle density. While this approach can better pick a heading direction due to the explicit search, it does not account for robot dynamics. This method has been extended to incorporate robot dynamics and local search, resulting in the VFH+ and VFH* methods [27,28].

Nonholonomic constraints were incorporated into the *steer angle field approach* of Feiten et al. [9]. In this approach, the robot builds a local map, and projects the robot’s path forward to determine what ranges of steering angles would cause a collision within a certain “hit distance”. If no collision-free steering angle can move the robot towards a goal location, a higher-level planner is consulted.

Simmons’ *curvature-velocity method* [24] and the *dynamic window approach* of Fox et al. [10] search a space of the robot’s translational and rotational velocities. Obstacles near the robot are transformed into this space by eliminating all commanded velocities that would cause a collision within a certain time period. These methods both take into account the kinematics

and the dynamics of the robot. Commanded velocities are chosen based on an objective function that considers both progress towards the goal and robot safety.

The TerraScout project at Carnegie Mellon University [21] has incorporated Fajen and Warren's original model into their obstacle avoidance system. The robot used in this project travels on outdoor terrain at $3\text{--}7\text{ m s}^{-1}$ along a preplanned path and must avoid obstacles. It creates a discretized local map using scanning laser rangefinders, and places a point obstacle in each occupied cell; the goal is set to a point a certain distance ahead on the path. The steering of the robot is set according to the angular acceleration computed by Fajen and Warren's model.

2.3. Dynamics of behavior

The dynamical systems approach to behavior studies the formulation of behaviors for autonomous agents. Schönner et al. [22] describe a framework for designing behaviors for autonomous robots. As an example, they formulate heading control of a mobile robot using a first order system that places an attractor at the goal heading and repellers at obstacle headings. They make the distinction between the potential field approach, where the system's attractor is a (static) pose of the robot, and their "dynamical approach" in which behavior (e.g. control of the robot heading) is determined by the location of an attractor, and this behavior changes appropriately with time and the robot state.

3. A model of human navigation

Inspired by the work of Schönner et al. [22], Fajen and Warren [7,8,29] devised a model of human navigation based upon experiments with human subjects. These experiments were conducted in a virtual environment where human subjects wore a head-mounted display and were physically free to move about in a 12 m by 12 m space. Subjects were instructed to walk through the virtual environment towards a goal; in some cases, they had to avoid obstacles along the way. A tracking system recorded subjects' position and orientation as they moved.

In the first set of experiments, subjects initially walking along a linear path were presented with a goal at different angles and distances from their initial position and heading. The data revealed that angular acceleration toward the goal increased with goal angle and decreased with goal distance. In the second set of experiments, subjects walking straight towards a goal were presented with an obstacle located at different initial angles and distances. In this case, angular acceleration away from the obstacle decreased with both obstacle angle and distance.

In this section, we describe the details of Fajen and Warren's model, first in the original acceleration framework, and then in our potential field interpretation.

3.1. Acceleration model

The inputs to Fajen and Warren's model, illustrated in Fig. 1, are the agent's heading (ϕ) and the bearing and distance to the

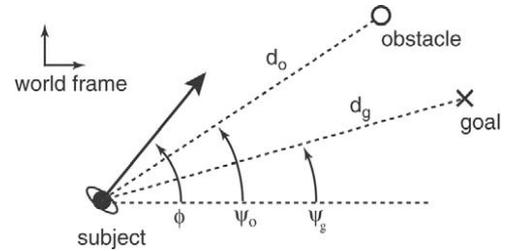


Fig. 1. Subject orientation ϕ and the angles to the goal ψ_g and an obstacle ψ_o are measured from the world x axis. Distance to the goal d_g and obstacle d_o are also shown.

goal (ψ_g and d_g) and to the obstacles (ψ_{o_i} and d_{o_i}). The angular acceleration is computed according to the following equation:

$$\ddot{\phi} = -b\dot{\phi} - k_g(\phi - \psi_g)(e^{-c_1 d_g} + c_2) + \sum_i k_{o_i}(\phi - \psi_{o_i})(e^{-c_3|\phi - \psi_{o_i}|})(e^{-c_4 d_{o_i}}). \quad (1)$$

This equation consists of three terms: a damping term, a goal term, and an obstacle term:

- The damping term opposes acceleration of the heading in proportion to turning rate $\dot{\phi}$ according to the damping constant b .
- The goal term pulls the heading towards the goal. Its strength increases proportionally with goal angle $(\phi - \psi_g)$, and decreases exponentially with goal distance. Note that the goal component never decreases to zero with goal distance, ensuring that the agent will turn toward distant goals. The "stiffness" parameter k_g modulates the strength of this component, c_1 determines the rate of decay with goal distance, and c_2 adjusts the minimum angular acceleration toward distant goals. A graph of this component appears in Fig. 2(a).
- The obstacle component pushes the heading away from obstacles. Its influence increases proportionally with obstacle angle for small angles but decreases exponentially with obstacle angle for larger angles. This means that as the agent turns toward an obstacle, the obstacle's repulsion increases, but only up to a certain point so that the agent can cut in front of the obstacle. In addition, the obstacle component decreases exponentially to zero as obstacle distance increases. Thus, the agent will not turn away from distant obstacles, even those located at small bearing angles. The parameter k_o determines the strength of the obstacle component, c_3 determines the rate of decay with obstacle angle, and c_4 determines the rate of decay with obstacle distance. A graph of this component appears in Fig. 2(b).

The model parameters were fit to the human data using a least-squares technique; the resulting parameters produced goal and obstacle angle profiles that matched the mean human paths with a Pearson correlation coefficient close to 1.0. Using these parameters, the model was simulated with more complex configurations of obstacles, including pairs of obstacles, cul-de-sacs, and arrays of randomly positioned obstacles. With few exceptions, the model successfully reached the goal while avoiding the obstacles. Additional experiments with human

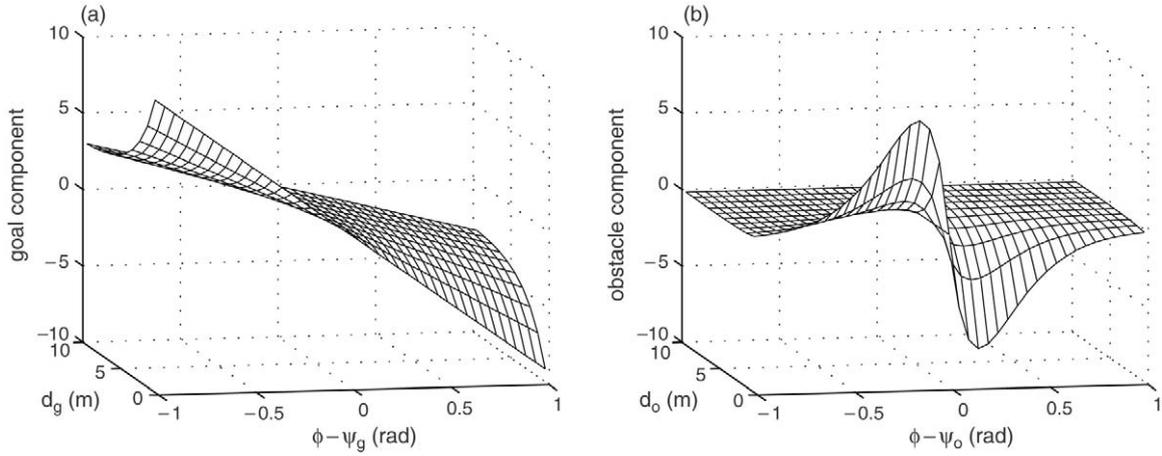


Fig. 2. Components of acceleration: (a) the goal acceleration in terms of the relative heading to the goal ($\phi - \psi_g$) and distance to the goal d_g , (b) the obstacle acceleration in terms of the relative heading to the obstacle ($\phi - \psi_o$) and the distance to the obstacle d_{o_i} .

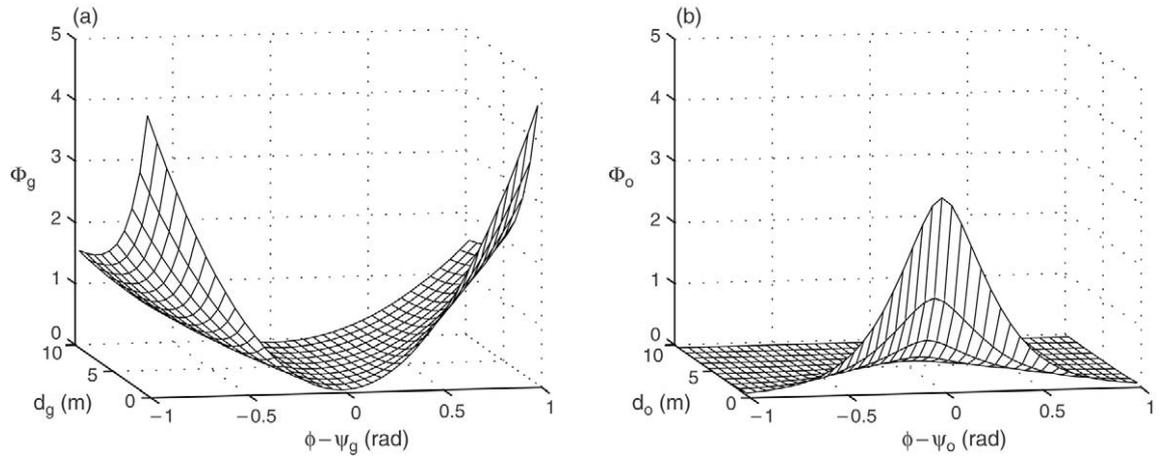


Fig. 3. Components of the potential function: (a) the goal potential Φ_g and (b) the obstacle potential Φ_{o_i} for a single obstacle. These surface plots should be considered a collection of one-dimensional potentials over the agent heading ϕ , relative to the heading to the goal, ψ_g , and to the obstacle, ψ_{o_i} , that are parameterized by the distance to the goal and obstacle, d_g and d_o respectively.

subjects in similar complex configurations revealed patterns of route selection behavior similar to those predicted by the model. The model was demonstrated to be robust through simulations with random noise in angular acceleration at each time step or with 10% error introduced in perceptual variables and parameters.

3.2. Potential field model

We have found that Fajen and Warren’s model can be expressed as a potential field where the goal and obstacles give rise to potential functions over the agent heading. The overall potential can be expressed as:

$$\Phi(\phi) = \Phi_g(\phi) + \sum_i \Phi_{o_i}(\phi) \quad (2)$$

where each obstacle i generates an independent potential Φ_{o_i} . The agent is then controlled according to:

$$\ddot{\phi} = \frac{d\Phi}{d\phi} - b\dot{\phi} \quad (3)$$

which is equivalent to Eq. (1).

This potential field is different than the traditional potentials used for agent navigation because the “potential landscape” keeps changing as the agent moves (which changes the relative headings and distances to the goal and obstacles). We describe this view of Fajen and Warren’s model to provide additional insight into its operation and because one of our extensions depends on the potential values.

3.2.1. Goal potential

The goal component of Eq. (1) can be expressed as a potential function by taking the negative integral with respect to ϕ . The goal potential is:

$$\Phi_g[\psi_g, d_g](\phi) = \frac{1}{2}k_g(\phi - \psi_g)^2(e^{-c_1d_g} + c_2). \quad (4)$$

The potential is a parabolic bowl over ϕ that is centered at the heading to the goal ψ_g . Note that this potential is a function of ψ_g and d_g which change as the agent moves. The distance to the goal d_g scales this potential so that it is steeper when the goal is near. See Fig. 3(a) for a graph of this potential.

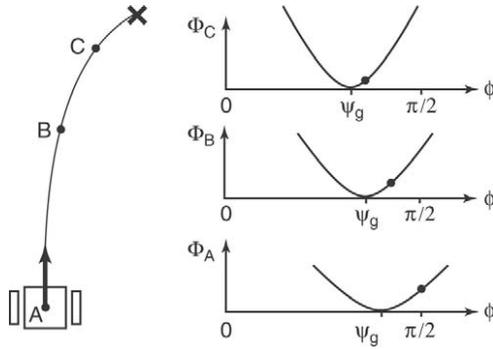


Fig. 4. Evolution of the goal potential along a path. The dot on the potential function indicates the agent's current heading.

When the agent heading is aligned with the goal, there is no angular acceleration from the goal component of Eq. (1); this corresponds to the agent heading being at the minimum of the potential. When the agent is not aligned towards the goal, its heading will accelerate downhill on this potential, i.e., towards the goal. However, as the agent moves closer to the goal, the heading to the goal will change. Damping is therefore required to prevent oscillations around the equilibrium point.

Fig. 4 illustrates the evolution of the goal potential over several points on a path to a goal. Also note the effect of distance to the goal — as the agent approaches the goal, the potential grows steeper.

3.2.2. Obstacle potential

The obstacle component of Eq. (1) can also be expressed as a potential function over the heading by taking the negative integral with respect to ϕ :

$$\begin{aligned} \Phi_{o_i}[\psi_{o_i}, d_{o_i}](\phi) \\ = k_o \frac{c_3|\phi - \psi_{o_i}| + 1}{c_3^2} (e^{-c_3|\phi - \psi_{o_i}|})(e^{-c_4 d_{o_i}}). \end{aligned} \quad (5)$$

The obstacle potential is a “bump” centered at the obstacle heading ψ_{o_i} , dropping sharply to both sides, and then decaying to zero. The height of the obstacle potential also decreases exponentially with increased distance to the obstacle so that distant obstacles have a lesser effect on the agent's steering. See Fig. 3(b) for a graph of this potential. Again note that this potential changes as the agent moves: this potential pushes the agent's heading away from the obstacle which causes the relative heading to the obstacle to increase at the same time.

When the agent's heading is close to an obstacle, this potential will cause the agent heading to accelerate away from the obstacle. As the agent passes by an obstacle, $|\phi - \psi_{o_i}|$ grows, dramatically reducing the influence of the obstacle on the agent heading.

One difference from the traditional potential field method is that the obstacle potential has a finite maximum. This is actually necessary so that the agent heading can cross from one side of an obstacle to the other. For example, if an agent is pointed straight into a cluster of obstacles, the agent heading will have to pass in front of at least one obstacle as it steers around the cluster.

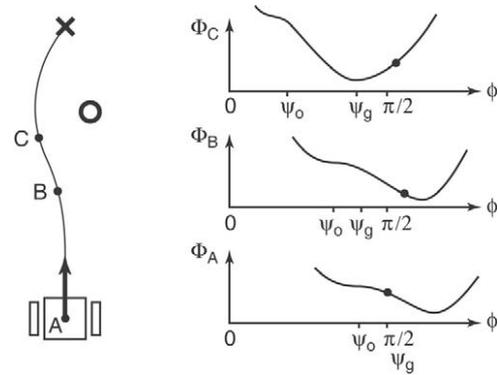


Fig. 5. Evolution of the total potential from a goal and one obstacle along a path. The dot on the potential function indicates the agent's current heading.

Fig. 5 illustrates the evolution of the total potential (obstacle and goal) over several points on a path to the goal.

3.2.3. Local minima and maxima

There will be local minima in the total potential function; however, this is not the problem that it is for traditional potential field methods. In traditional potential field approaches, a local minimum is a location where the agent will stop making progress towards the goal. In Fajen and Warren's model, the potential field controls only the steering direction; the translational velocity is held constant. Local minima in the potential represent possible headings to drive around an obstacle and reach the goal. In general, the agent will follow steering directions determined by local minima, and that a local minimum will become a global minimum once the agent has avoided all the obstacles and has a clear route to the goal.

When there are two or more obstacles arranged symmetrically to the left and right of the heading, the obstacle potentials sum together to create a local minimum at a heading between the two innermost obstacles. The local minimum represents a path that the agent may follow even if the obstacles are too close together. It is possible for errors in actuation to dislodge the agent from the shallow local minimum produced in this case — once the agent is sufficiently far off the line of symmetry, one obstacle peak will become larger while the other becomes smaller, and this can cause the local minimum to disappear. Both our extensions presented in Section 4 combine to avoid these situations.

One problem with the original model is a consequence of having a finite obstacle potential: it can produce an unstable local maximum. This is most unfortunate because should the agent not be dislodged from this local maximum, it will run right into the obstacle. While in any real system, we could probably count on noise to dislodge the heading from this point, limited sensor resolution could conspire to prevent this. We have introduced speed control (in Section 4.2) to deal with this problem.

3.3. Agent and obstacle width

Another problem with Fajen and Warren's model is that the agent and obstacles are treated as points, which means that

paths would not take into account robot and obstacle widths. One could tune the parameters to adjust for changes in agent or obstacle size, but collisions are still possible, especially when the sizes of obstacles in a scene vary from small to large. In Section 4, we show how this problem can be solved using angular width together with knowledge of robot size and minimum obstacle size.

4. A model for robot navigation

We made two major modifications to Fajen and Warren’s model. First, we modified the obstacle term to account for non-point obstacles; the revised term generates a potential appropriate to the obstacle size and relative to the robot size. Second, we introduced velocity control to guarantee that the robot won’t collide with an obstacle.

We assume that the goal heading and distance are known, that these quantities can either be sensed directly or may be given to the robot, e.g., a point further ahead on a preplanned path provided by a high-level path planning module. For obstacles, however, we have only what can be sensed by a single camera: the heading to the obstacle and its angular width. We do assume that obstacles are circular or are conservatively approximated by circles.

4.1. Extension to obstacle widths

Several things are required to adapt Fajen and Warren’s model to a robot which must avoid obstacles with nonzero widths. First, wider obstacles should generate a wider obstacle potential so the robot takes a wider path around the obstacle. To guarantee that the robot does not collide with an obstacle, this potential must approach infinity as the robot gets closer and closer to the obstacle. Finally, we wish to remove the dependence on the distance to obstacles so that our method can be implemented using single-camera vision.

We have achieved all these objectives by replacing the term $e^{-c_4 d_o}$ in the obstacle potential by $(\tan(\theta_i + c_5) - \tan c_5)$ where θ_i is the angular width of the i th obstacle and c_5 is a new parameter. The obstacle potential is now:

$$\Phi_{o_i}[\psi_{o_i}, d_o](\phi) = k_o \frac{c_3 |\phi - \psi_{o_i}| + 1}{c_3^2} \times (e^{-c_3 |\phi - \psi_{o_i}|})(\tan(\theta_i + c_5) - \tan c_5). \quad (6)$$

The new term makes use of the tangent function to drive the potential toward infinity as θ_i approaches some maximum value. As the robot gets closer and closer to an obstacle, its angular width will increase. For any given sized obstacle, there will be a maximum angular width that occurs when the robot has made contact with the obstacle. If we knew the obstacle radius, we could set c_5 exactly:

$$c_5 = \frac{\pi}{2} - 2 \tan^{-1} \left(\frac{r_o}{r_o + r_r} \right) \quad (7)$$

where r_o is the obstacle radius and r_r is the robot radius. Since we cannot depend on knowing the size of each obstacle, we conservatively set c_5 using the radius of the smallest obstacle.

This has the effect of making the robot conservative about avoiding distant large obstacles; however, given our aim to avoid using the obstacle distance, there is no other choice. We discuss the effect of different obstacle widths in Section 4.4.

4.2. Velocity control

If the robot is following a local minimum between two symmetrically placed and closely spaced obstacles, the robot may still try to pass between them, even using the new obstacle term above: as the robot gets closer, the potential for each obstacle gets bigger and bigger, but because of the symmetry, there is still a local minimum between them.¹

To keep the robot from attempting to squeeze through an opening that is too narrow, we introduce speed control. Whereas speed was held constant in the original model, we now set it to decrease exponentially as a function of the height of the obstacle potential:

$$v = \max\{v_{\max} e^{-k_v \Phi_o} - \epsilon, 0\} \quad (8)$$

where ϵ is a small positive constant so that the velocity becomes zero for sufficiently large values of Φ_o .

In the absence of obstacles, the obstacle potential is zero, and so speed is equal to v_{\max} . When obstacles are nearby and/or closely aligned with heading, the obstacle potential is greater, which causes speed to decrease.

The addition of speed control also helps with collision avoidance because the robot will make sharper turns. The navigation model controls angular acceleration and velocity independent of how close the obstacles are. For a given rotational velocity $\dot{\phi}$, the robot will travel with a curvature radius of $v/\dot{\phi}$, where v is the translational velocity. If v decreases, then the turning radius will also decrease resulting in a more aggressive obstacle avoidance maneuver.

4.3. Example

Fig. 6 shows a sample path between two obstacles of different sizes to a goal 5 m ahead. The bottom graph of this figure shows the velocity profile, with white squares indicating speed at the four locations along the path. Fig. 7 shows the corresponding potential for these four points: the top row shows the total potential, and the bottom row shows the goal and obstacle components separately.

When the robot is at the starting location (location A), its heading is aligned with the goal and close to the minimum of the total potential function. Although the obstacle potential is weak because both obstacles are still far away, the peak of the obstacle potential is closely aligned with the robot’s heading, which causes the robot to move at less than its maximum speed of 0.7 m s^{-1} .

¹ We note that this situation can occur when the obstacles are not exactly symmetric and when the obstacles appear symmetric to the robot. For sufficient difference in obstacle depth, however, this symmetry will be broken because the angular widths will change at different rates as the robot moves forward.

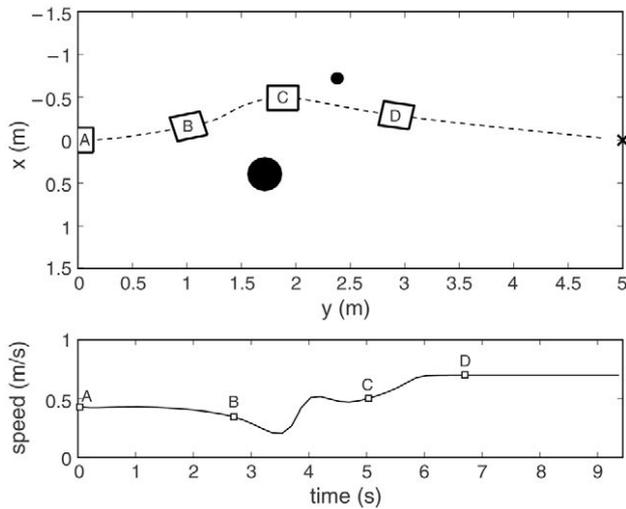


Fig. 6. Simulated example path for our model. The top figure shows the robot path with four locations indicated. The bottom figure shows the velocity profile for this path with squares placed for each of the locations along the path.

By the time the robot reaches location B, the larger obstacle has created a local minimum in the total potential function, which shifts the global minimum to the left of the goal and causes the robot to turn away from this obstacle. The height of the obstacle potential at the robot's heading is also greater than it was at location A, resulting in a further drop in speed.

At location C, the robot is heading slightly to the left of the obstacle, but the goal and the smaller obstacle push the global minimum to the right, which eventually causes the robot to turn back in that direction. Note that the influence of the smaller obstacle is considerably less than the influence of the larger

obstacle because its angular width is smaller. Also, the robot's speed is greater than it was at location B because the obstacle potential at the robot's heading is now lower.

By the time the robot reaches location D, it has settled into the minimum of the total potential, which is dominated by the goal potential because the obstacles are now behind the robot. The robot's speed at this location is close to maximum speed because the obstacle potential is close to zero.

4.4. Simulations with different obstacle widths

In Section 4.1, we pointed out that the robot will behave conservatively when avoiding a distant large obstacle because it turns as if to avoid a nearby small obstacle with the same angular width. This is a consequence of our aim to develop an approach that does not require distance information. When the sizes of obstacles in the scene are similar, then this conservative bias will be inconsequential. As the ratio of the largest to the smallest obstacle increases, the bias may result in longer, more circuitous routes. This is illustrated in Fig. 8(a), which shows a series of simulations with varying obstacle size ratios. For larger ratios, the robot leaves more room than necessary when avoiding the larger obstacle. However, the robot still avoids the obstacles and reaches the goal. Considering the fact that the robot is able to navigate without distance information, avoiding collisions at the expense of taking a longer route is a reasonable tradeoff.

Although the algorithm was designed to work without distance information, such information can be seamlessly incorporated into the algorithm when it is available. Rather than assume that each obstacle's width is equal to the width of the smallest obstacle, the actual width of each obstacle can

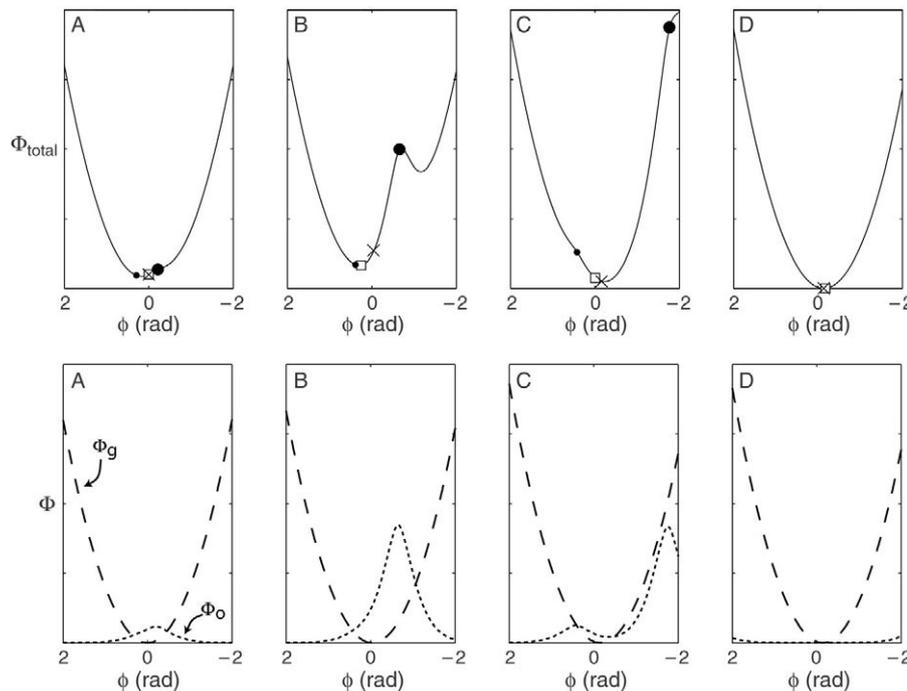


Fig. 7. Potential functions for the four locations on the path in Fig. 6. The top row shows the total potential; obstacle headings are indicated by black circles with the size corresponding to the obstacle size, the goal heading is indicated by the X, and robot heading is indicated by the square. The bottom row shows the goal potential (dashed line) and total obstacle potential (dotted line) separately.

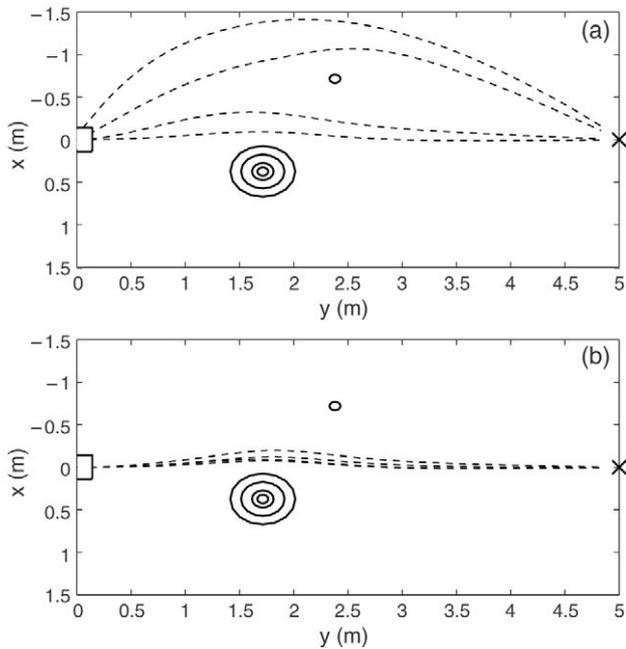


Fig. 8. Simulated paths for four obstacle size ratios (1:1, 2:1, 4:1, and 6:1). Figure (a) uses our original model, while Figure (b) incorporates distance information.

be estimated using distance information together with angular width. That is, distance information can be used to set a different c_5 parameter for each obstacle. This eliminates the conservative bias, and allows the robot to navigate through a field of obstacles whose sizes vary across a wide range. Fig. 8(b) shows the routes followed by the robot using distance information under the same conditions used in Fig. 8(a). Note that the path is affected by the change in obstacle size only to the extent that it is necessary to avoid a collision.

Fig. 9(a) demonstrates that the robot with distance information can effectively navigate through a field of five differently-sized obstacles. When the gap between the last two obstacles is large enough for the robot to fit through, the robot takes the shorter route to the goal (Fig. 9(b)). This illustrates that when distance information is available, it can be used to improve obstacle avoidance and route selection with just a minor modification to the algorithm.

5. Experimental implementation

5.1. Implementation details

We implemented our model on a MagellanPro robot; this is a differential drive robot made by iRobot, with an on-board PC and wireless Ethernet. Although this robot has SONAR and IR sensor rings, the only sensor used in our experiments was an omnidirectional camera mounted atop the robot.

The camera is pointed upwards at a hyperbolic mirror and produces a 640 by 480 pixel image that is then dewarped to a 936 by 240 pixel image with a maximum 0.348 degree/pixel angular resolution. In order to reduce processing time, only half of this image is actually dewarped to provide a 180 degree

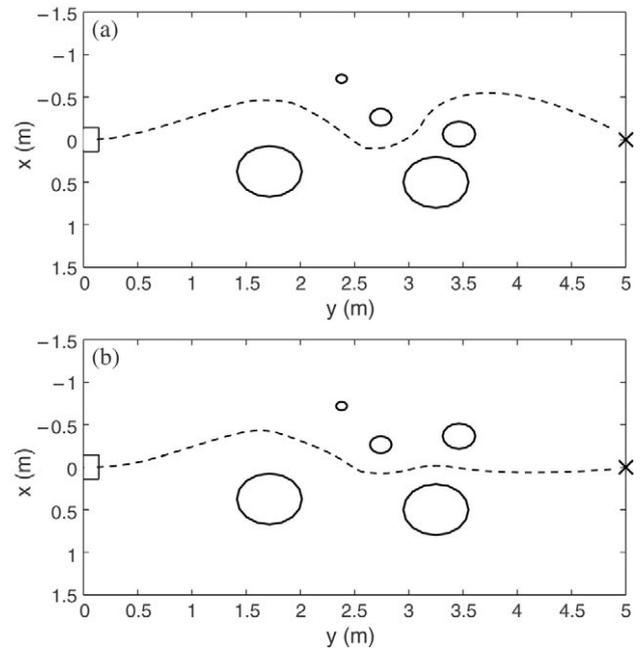


Fig. 9. Simulated navigation of an obstacle field using distance information. In (b), a gap between the last two obstacles has closed, so the robot must go around them.

view; this should not affect navigation since obstacles with large relative headings have a very small effect on the steering.

The goal and obstacles used in our experiments were cylindrical columns ranging from 0.1 to 0.23 m in diameter. The goal was taller than the obstacles (0.6 m high as compared to 0.3 m for the obstacles) in order to avoid goal occlusion. The goal and obstacles were different colors, so color segmentation followed by connected region extraction was used to recover goal and obstacle headings and obstacle angular widths. The distance to the goal was estimated based on its width in the image, taking advantage of its known size of 0.1 m.

The image capture and processing ran at 10 Hz which provided an adequate sample rate to produce smooth paths that avoided obstacles and reached the goal.

5.2. Tuning model parameters

We have developed a procedure for tuning the parameters of our model. In general, the parameters from Fajen and Warren's experiments cannot be used because the robot uses the model at discrete intervals (at 10 Hz) instead of continuously. Table 1 contains a summary of the parameters, their values in our experiments, and an explanation of their effect on the path.

First, the parameters for goal directed behavior must be tuned. This can be done by starting the robot with the goal off to one side and observing the path that the robot takes. The goal potential will accelerate the heading towards the goal; the aim of this phase of the tuning procedure is to achieve a desired response without any oscillation of the robot heading about the goal direction. The c_1 and c_2 parameters should be set to 0.4; this is their value from Fajen and Warren's experiments, and we have found these values to work well on our robot. Initially k_g and b should be set to low values (e.g., 1.0). The damping

Table 1
Model parameters and their effect on paths

Parameter	Value	Effect on path
b	5.5	Controls oscillations when following local minima to goal
k_g	2.0	Sets rate of goal attraction. Increasing will result in paths that more aggressively steer towards the goal.
c_1	0.4	Used to balance the attraction of a distant goal with near obstacles
c_2	0.4	Ensures that a very distant goal still has effect.
k_o	9.0	Controls obstacle repulsion. Increasing will result in paths that avoid obstacles more drastically. Decreasing will allow the robot to approach closer to obstacles before avoiding.
c_3	4.0	Controls “gap shooting” behavior. Increasing will result in paths that try to go between obstacles. Decreasing will result in paths that go around groups of obstacles. Also controls the clearance a path takes around an obstacle.
c_4	0.0	Only used when estimating distance to obstacles. Increasing this parameter will decrease the effect of distant obstacles so that paths can be more aggressive for close obstacles.
c_5	1.16	Increases obstacle potential to infinity as the robot approaches an obstacle to avoid collisions. This parameter is calculated based on the angular width of the minimum-size obstacle.
k_v	0.5	Rate at which translational velocity is decreased. Should be adjusted to stop the robot before a collision.

parameter b should be then increased so that oscillations in the path are eliminated, even for large initial relative headings to the goal. Once this basic response is achieved, k_g can be increased to make the robot turn towards the goal more quickly. The tradeoff for higher values of k_g is that higher values of b are required to prevent oscillations, and this will make the robot less responsive to obstacles.

Next, parameters for the obstacle term are set in experiments with a single obstacle. The robot should start pointing towards the goal, and the obstacle should be about halfway between the robot and goal, slightly off to one side. First, note that the c_5 parameter should be calculated according to Eq. (7) (for the minimum obstacle size). The obstacle gain k_o should be set initially to $2k_g$, the c_3 parameter, to 4.0. The k_o and c_3 parameters should then be adjusted so that the robot smoothly avoids a range of obstacle sizes. Increasing k_o will cause the robot to begin turning away from the obstacle earlier. Increasing c_3 causes the robot to take paths closer to the obstacle because the obstacle potential will be narrower. Since the tangent term of the obstacle function is designed to rapidly approach infinity when the perceived angular width of an obstacle approaches $\frac{\pi}{2} - c_5$, it is important to tune c_3 such that the robot takes a path far enough around the largest obstacle.

If using the distance component of the obstacle term, c_4 can initially be set to zero (effectively eliminating this component) and then increased to better control the distance at which the robot starts to avoid the obstacle.

Tests with multiple obstacles should be done to further tune k_o and c_3 (and c_4 if using the distance component). By decreasing k_o and increasing c_3 , the paths taken are made more aggressive and goal dominated. They will come closer to obstacles and usually try to fit through gaps between obstacles. The opposite adjustment (increasing k_o and decreasing c_3) will result in more conservative paths that tend to go around all the

obstacles to get to the goal. As c_4 is increased, the robot will ignore more distant obstacles and can then be made to respond more aggressively to nearby obstacles.

Finally, the speed parameter k_v should be chosen so that the robot stops before colliding with an obstacle. This should be tested with a goal directly in front of the robot and two obstacles of minimum size placed on either side of the goal heading, and spaced too close together for the robot to pass. This configuration produces the smallest potential that should stop the robot. The k_v parameter should be set high enough that the robot comes to a stop before a collision occurs.

In general, there are many sets of parameters for which the model will correctly avoid obstacles and reach the goal. However, by changing different parameters, the paths can be drastically altered to be more or less conservative or take “inside” or “outside” paths. See Table 1 for a summary description of each parameter’s effect on the path.

5.3. Experiments

We conducted more than fifty experiments to test our model. We ran all of these experiments with a maximum velocity of 0.7 m s^{-1} . There were a number of trials with varying complexity to show that the model works for different sizes and numbers of obstacles.

Fig. 10 shows some trials designed to test basic obstacle avoidance behavior. Note that the paths are all smooth and take reasonably efficient paths through the field of obstacles.

The experiments shown in Fig. 11 illustrate the effect of obstacle width on the robot’s path. A wider obstacle creates a larger and wider potential, steering the robot further away from that obstacle.

Fig. 12 shows experiments to test the behavior of the robot avoiding a cluster of obstacles. In Fig. 12(a), the robot chooses

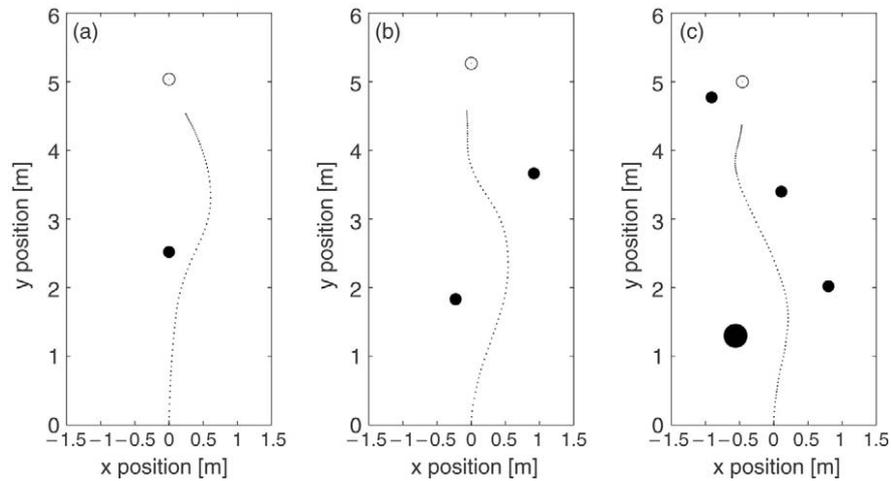


Fig. 10. Experimental results: simple trials to illustrate the paths taken by our model. Note that in these and other experimental data shown, the robot was programmed to slow down and stop short of the goal (the white circle). The black circles are sized proportionately to the corresponding obstacle's size.

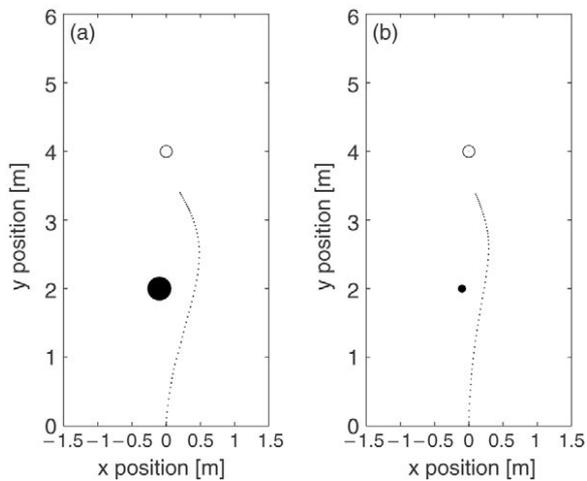


Fig. 11. Experimental results showing the effect of obstacle width.

a path through a gap between two obstacles. When the obstacles are spaced closely together, as in Fig. 12(b), the resulting obstacle potential causes the robot to steer completely around the cluster. There are, however, situations such as in Fig. 12(c), in which the robot is following a local minimum corresponding to a gap that is too narrow for the robot to pass. As the robot approaches this gap, the level of the obstacle potential rises, causing the robot to slow down and stop before a collision occurs. In some cases the robot can rotate until it reaches another local minima in the potential that can lead it out of this situation.

These experiments demonstrate the basic capabilities of our method for obstacle avoidance. Note that this is accomplished without any depth information for the obstacles.

5.4. Comparison with VFH+

We performed a series of tests in simulation to compare the behavior of our model with VFH+ [27], a commonly used obstacle avoidance method. VFH+ (an enhanced version of the Vector Field Histogram method) takes a local occupancy grid map and creates a polar histogram based on a thresholded

“obstacle density”. It then eliminates steering directions that violate kinodynamic constraints of the robot and then chooses a steering direction in an obstacle-free valley of the histogram.

In order to make a fair comparison between our model and VFH+, we incorporated distance information as described in Section 4.4. We implemented our method in the Player/Stage framework using the model for an ActivMedia Pioneer robot (a differential drive robot) with a SICK laser scanner. Depth discontinuities were used to segment the laser scan, and the distance, bearing, and angular width were calculated for each segment. We used the built-in implementation of VFH+ in Player/Stage.

We ran a number of simulations for fields of circular obstacles of varying size; a few of our results are shown in Fig. 13. In general, we found that our method produces much smoother paths than the VFH+ method. This is not surprising given that the VFH+ method starts with a local map and therefore does not consider distant obstacles. In addition, the “obstacle density” is a function of the distance to an obstacle, so an obstacle would not surpass the threshold until it is relatively close. Thus when VFH+ comes upon an obstacle, it tends to have to take sharper turns to avoid it.

6. Conclusions

In this paper, we described and analyzed a model of human navigation, and adapted it for vision-based robot navigation. Both the original and adapted versions of the model can be described in terms of a potential function defined over the robot's heading. The goal potential is a parabolic bowl centered on the goal heading, and the potential for each obstacle is a peak centered on the obstacle heading. Because the potential is formulated in terms of relative headings to the goal and obstacles, the potential function changes as the robot moves. Local minima in the total potential function represent headings that will steer the robot around obstacles and toward the goal. Once the robot has avoided all of the obstacles and has a clear route to the goal, the local minimum will become a global minimum centered on the heading of the goal. In this sense, our

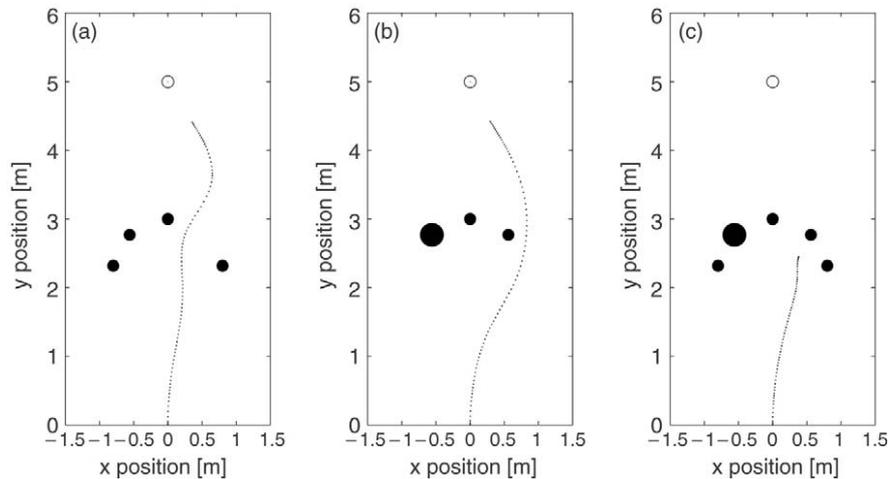


Fig. 12. Experimental results for avoiding a cluster of obstacles.

potential function provides a control law for the robot heading to track a path to the goal.

The original model of human navigation is completely reactive and simple to compute, directly applicable to steering nonholonomic vehicles, and produces smooth, natural looking paths. However, this model is not suitable for vision-based robot navigation because obstacles are treated as points, knowledge of obstacle distance is required, and collisions with

obstacles are possible. Our modification of the original model uses the obstacles' angular width rather than distance, and controls speed based on the obstacle potential to help with collision avoidance. Together, these two modifications address the shortcomings of the original model, while still retaining its advantages. We implemented our model on a differential-drive mobile robot and conducted experiments with multiple obstacles of varying sizes.

Because the angular width of obstacles, not their distance, is used, our method is somewhat conservative — a large distant obstacle can have the same angular width as a closer small obstacle. Despite this limitation, our method has been demonstrated to be effective for real-time navigation and obstacle avoidance.

Acknowledgments

Wes Huang and Jon Fink were supported in part by the NSF through grant IIS-9983642. Bill Warren was supported by the NIH through grant EY10923.

References

- [1] Y. Aloimonos, Is visual reconstruction necessary? Obstacle avoidance without passive ranging, *Journal of Robotic Systems* 9 (6) (1992) 843–858.
- [2] J. Borenstein, Y. Koren, The vector field histogram — fast obstacle avoidance for mobile robots, *IEEE Transactions on Robotics and Automation* 7 (3) (1991) 278–288.
- [3] O. Brock, O. Khatib, High-speed navigation using the global dynamic window approach, in: *IEEE International Conference on Robotics and Automation*, 1999, pp. 341–346.
- [4] T. Camus, D. Coombs, M. Herman, T.-H. Hong, Real-time single-workstation obstacle avoidance using only wide-field flow divergence, in: *International Conference on Pattern Recognition*, vol. 3, 1996, pp. 323–330.
- [5] C.I. Connolly, R.A. Grupen, Nonholonomic path planning using harmonic functions, Technical Report UM-CS-1994-050, University of Massachusetts at Amherst, 1994.
- [6] G. Dudek, M. Jenkin, *Computational Principles of Mobile Robotics*, Cambridge University Press, 2000.
- [7] B.R. Fajen, W.H. Warren, Behavioral dynamics of steering, obstacle avoidance, and route selection, *Journal of Experimental Psychology: Human Perception and Performance* 29 (2) (2003) 343–362.

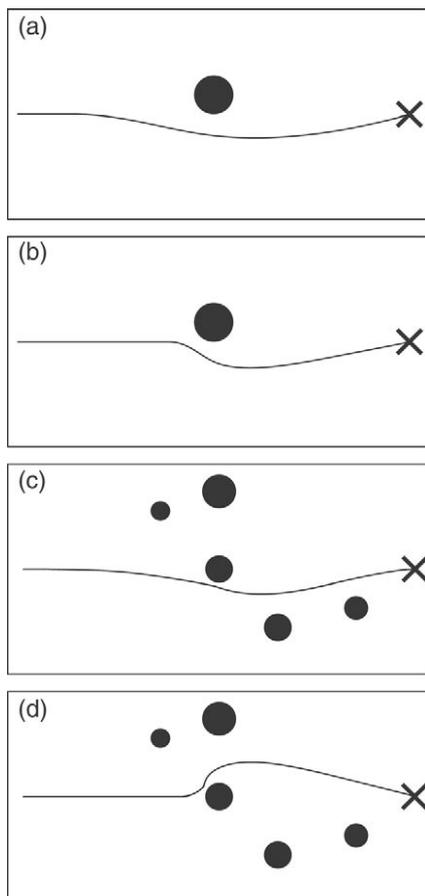


Fig. 13. Simulations comparing our model, in figures (a) and (c), to the VFH+ method, in figures (b) and (d).

- [8] B.R. Fajen, W.H. Warren, S. Termizer, L.P. Kaelbling, A dynamical model of steering, obstacle avoidance, and route selection, *International Journal of Computer Vision* 54 (1–2) (2003) 13–34.
- [9] W. Feiten, R. Bauer, G. Lawitzky, Robust obstacle avoidance in unknown and cramped environments, in: *IEEE International Conference on Robotics and Automation*, 1994, pp. 2412–2417.
- [10] D. Fox, W. Burgard, S. Thrun, The dynamic window approach to collision avoidance, *IEEE Robotics & Automation Magazine* 4 (1) (1997) 23–33.
- [11] G.C. Gini, A. Marchi, Indoor robot navigation with single camera vision, in: *Pattern Recognition in Information Systems*, 2002, pp. 67–76.
- [12] J. Guldner, V.I. Utkin, Tracking the gradient of artificial potential fields: sliding mode control for mobile robots, *International Journal of Control* 63 (3) (1996) 417–432.
- [13] M. Khatib, R. Chatila, An extended potential field approach for mobile robot sensor-based motions, in: *Intelligent Autonomous Systems*, 1995, pp. 490–496.
- [14] O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, *International Journal of Robotics Research* 5 (1) (1986) 90–98.
- [15] K. Konolige, A gradient method for realtime robot control, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems* vol. 1, 2000, pp. 639–646.
- [16] K.J. Kyriakopoulos, P. Kakambouras, N.J. Krikelis, Potential fields for nonholonomic vehicles, in: *IEEE International Symposium on Intelligent Control*, 1995, pp. 461–465.
- [17] F. Lamiroux, D. Bonnafous, Reactive trajectory deformation for nonholonomic systems: Application to mobile robots, in: *IEEE International Conference on Robotics and Automation*, 2002, pp. 3099–3104.
- [18] J.-C. Latombe, *Robot Motion Planning*, Kluwer Academic Publishers, 1991.
- [19] D.N. Lee, A theory of visual control of braking based on information about time-to-collision, *Perception* 5 (1976) 437–459.
- [20] E. Rimon, D. Koditschek, Exact robot navigation using artificial potential functions, *IEEE Transactions on Robotics and Automation* 8 (5) (1992) 501–518.
- [21] S.A. Roth, B. Hamner, S. Singh, M. Hwangbo, Results in combined route traversal and collision avoidance, in: *International Conference on Field and Service Robotics*, July 2005.
- [22] G. Schöner, M. Dose, C. Engels, Dynamics of behavior: theory and applications for autonomous robot architectures, *Robotics and Autonomous Systems* 16 (1995) 213–245.
- [23] S. Sekhavat, M. Chyba, Nonholonomic deformation of a potential field for motion planning, in: *IEEE International Conference on Robotics and Automation*, 1999, pp. 817–822.
- [24] R. Simmons, The curvature-velocity method for local obstacle avoidance, in: *IEEE International Conference on Robotics and Automation*, vol. 4, 1996, pp. 2275–2282.
- [25] H.G. Tanner, S. Loizou, K.J. Kyriakopoulos, Nonholonomic stabilization with collision avoidance for mobile robots, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, pp. 1220–1225.
- [26] C. Tomasi, T. Kanade, Shape and motion without depth, in: *International Conference in Computer Vision*, 1990, pp. 91–95.
- [27] I. Ulrich, J. Borenstein, VFH+: Reliable obstacle avoidance for fast mobile robots, in: *IEEE International Conference on Robotics and Automation*, 1998, pp. 1572–1577.
- [28] I. Ulrich, J. Borenstein, VFH*: Local obstacle avoidance with look-ahead verification, in: *IEEE International Conference on Robotics and Automation*, 2000, pp. 2505–2511.
- [29] W.H. Warren, B.R. Fajen, From optic flow to laws of control, in: L.M. Vaina, S.A. Beardsley, S. Rushton (Eds.), *Optic Flow and Beyond*, Kluwer, 2004, pp. 307–333.



Wesley H. Huang is an Assistant Professor in the Computer Science Department at Rensselaer Polytechnic Institute. He received an S.B. in Electrical Engineering from the Massachusetts Institute of Technology in 1991, and a Ph.D. from the Robotics Institute at Carnegie Mellon University in 1997. Prior to joining Rensselaer, he was a postdoctoral fellow in the Institute for Complex Engineered Systems at Carnegie Mellon. His research spans robotic manipulation, mobile manipulation, and mobile robotics.



Brett Fajen is an Assistant Professor in the Cognitive Science Department at Rensselaer Polytechnic Institute. He received a B.S. in Mathematics from Bucknell University in 1993, and a Ph.D. in Experimental Psychology from University of Connecticut in 1999. Before joining Rensselaer, he was a Postdoctoral Research Associate in the Department of Cognitive and Linguistic Sciences at Brown University. His research focuses on the visual guidance of action and optic flow.



Jonathan Fink received a dual B.S. in Electrical and Computer Systems Engineering at Rensselaer Polytechnic Institute in May 2004. He is currently pursuing a Ph.D. in Electrical & Systems Engineering as a member of the GRASP Lab at the University of Pennsylvania. His research interests include mobile robotics, mobile sensor networks, and computational approaches to design and planning with uncertainty in dynamic simulation.



William H. Warren is Professor and Chair of the Department of Cognitive and Linguistic Sciences at Brown University. He uses virtual reality techniques to study the visual control of human locomotion, navigation, and the dynamics of perceptual-motor behavior. Warren received his Ph.D. in Experimental Psychology from the University of Connecticut (1982), did post-doctoral work at the University of Edinburgh (1983), and has been a professor at Brown ever since. He is the recipient of a National Research Service Award from NIH (1983), a Fulbright Research Fellowship (1989), a Research Career Development Award from NIMH (1997–2002), and Brown's Elizabeth Leduc Teaching Award for Excellence in the Life Sciences.