

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI  
VIỆN CƠ KHÍ  
BỘ MÔN CƠ SỞ THIẾT KẾ MÁY & ROBOT

\*

LUẬN VĂN  
TỐT NGHIỆP CAO  
HỌC  
NGÀNH CƠ ĐIỆN TỬ

Đề tài: Nghiên cứu và phát triển giải thuật  
điều khiển robot tự hành thông minh

Học viên thực hiện:

Mã số học viên:

Giáo viên hướng dẫn:

Giáo viên phản biện:

Nguyễn Văn Huy

CB180009

Lớp CH2018B

TS. Nguyễn Xuân Hạ

HÀ NỘI 06/2020

# Mục lục

<b>MỤC LỤC</b>	<b>5</b>
<b>DANH SÁCH HÌNH VẼ</b>	<b>6</b>
<b>1 Tổng quan nghiên cứu</b>	<b>1</b>
1.1 Giới thiệu robot tự hành . . . . .	1
1.2 Ứng dụng của robot tự hành thông minh . . . . .	2
1.3 Các bài toán trên robot tự hành thông minh . . . . .	3
1.4 Các nghiên cứu tránh vật cản trong robot tự hành thông minh . .	4
1.4.1 Thuật toán Virtual Force Field (VFF) . . . . .	4
1.4.2 The Vector Field Histogram . . . . .	6
1.4.3 Phương pháp "Bong bóng phản ứng" tránh vật cản . . . .	7
1.5 Nội dung nghiên cứu . . . . .	8
<b>2 Cơ sở lý thuyết</b>	<b>9</b>
2.1 Bài toán về nhiễu trong robot tự hành . . . . .	9
2.2 Bài toán SLAM 2D . . . . .	9
2.3 Bài toán tạo định vị, tạo bản đồ, điều hướng và tránh vật cản . .	9
2.4 Hệ điều hành robot ROS và các ứng dụng . . . . .	9
<b>3 Thực nghiệm</b>	<b>10</b>
3.1 Đặt vấn đề . . . . .	10
3.2 Giới thiệu platform robot của nhóm . . . . .	10
3.3 Giới thiệu platform robot của nhóm . . . . .	10
3.4 Thực nghiệm và đánh giá kết quả . . . . .	10
<b>4 Kết luận và tầm nhìn</b>	<b>11</b>
4.1 Kết luận . . . . .	11
4.2 Tầm nhìn . . . . .	11

# Danh sách hình vẽ

1.1	Robot công nghiệp [Nguồn: Internet]	1
1.3	Mô hình hệ thống robot tự hành	3
1.4	Virtual Force Field [1]	6
1.5	Vector Field Histogram [2]	7
1.6	Bong bóng phản ứng [2]	7
1.7	Phương pháp bong bóng phản ứng động	8

# Chương 1

## Tổng quan nghiên cứu

### 1.1 Giới thiệu robot tự hành

Trước đây, ngành robotics tập trung vào nghiên cứu và chế tạo cánh tay robot công nghiệp. Nó được coi như là robot truyền thống (Hình1.1). Bởi vậy khi nói đến thuật ngữ robot người ta thường nghĩ ngay đến cánh tay robot công nghiệp. Hình1.1 là hình ảnh các cánh tay robot đang làm việc trong khâu sơn khung xe oto. Chúng ta không thấy bóng dáng con người ở trong bức hình này, bởi vì robot công nghiệp truyền thống phải làm việc trong không gian cách ly với con người, có hàng rào bảo vệ vì các lý do an toàn, con người không thể làm việc cùng không gian với robot.



Hình 1.1: Robot công nghiệp [Nguồn: Internet]

Bên cạnh đó, robot di động (mobile robot) truyền thống thực hiện các nhiệm vụ di chuyển trên quỹ đạo xác định trước. Robot di động cũng hoạt động dựa trên các khối chương trình được lập trình sẵn. Các phương pháp điều khiển như điều khiển bằng tay thông qua bảng điều khiển, qua sóng RF, wifi... hay di chuyển bám đường chỉ dẫn gắn ở dưới sàn, đọc mã QR, bar. Robot dạng này



(a)



(b)



(c)



(d)

Hình 1.2: Một số loại robot mới

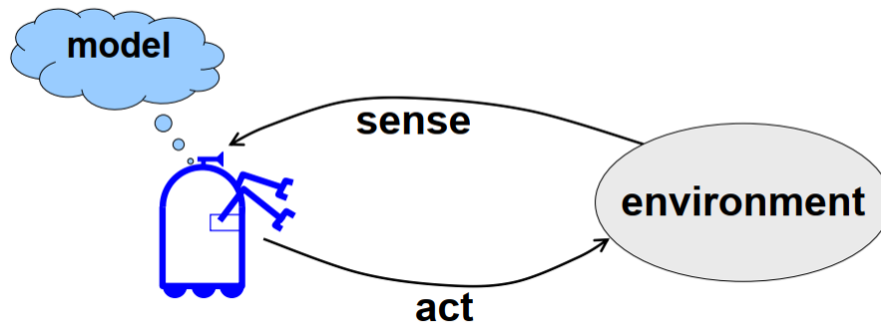
bị hạn chế về không gian hoạt động. Việc thiết lập, cấu hình nhà máy, không gian làm việc cho robot hoạt động hết sức tốn kém về chi phí và thời gian mà lại kém linh hoạt.

Ngày nay, robot có xu hướng đi ra khỏi không gian nhà máy, xuất hiện nhiều loại robot như: robot giải trí (Hình1.2a), robot dịch vụ cá nhân (Hình1.2b) (như máy tính cá nhân), robot trong y tế (Hình1.2c); các loại robot tự động trong công nghiệp như robot hái quả (Hình1.2d), phun thuốc trong nông nghiệp, robot thông minh tác hợp trong công nghiệp; robot đi tới các môi trường mà con người không tới được như trong lòng đất dưới nước, trên không, trong vũ trụ...

Robot tự hành thông minh là loại robot di động, có thể cảm nhận, mô hình hóa môi trường xung quanh nó. Thực hiện các hành động di chuyển mà không cần sự giám sát cũng như điều khiển trực tiếp từ con người (Hình1.3).

## 1.2 Ứng dụng của robot tự hành thông minh

Với sự thông minh và cực kì linh hoạt của nó, robot tự hành thông minh có rất nhiều ứng dụng. Về cơ bản, nó là nền tảng di chuyển cho tất cả các loại robot di động thông minh ngày nay. Một số sản phẩm ứng dụng trong nhà như:



Hình 1.3: Mô hình hệ thống robot tự hành

- Các ứng dụng trong nhà như robot hút bụi thông minh, robot dịch vụ, robot vận chuyển trong các kho hàng. . .
- Các ứng dụng ngoài trời như robot cắt cỏ, chăm sóc cây trồng...
- Ở các không gian mà con người không tới được như robot thám hiểm dưới nước, trong lòng đất, trên không trung trên các hành tinh khác...
- Tại các khu vực nguy hiểm như khu vực nhiễm chất phóng xạ, chữa cháy...
- Và đặc biệt, robot giao hàng tự động, xe tự lái đang phát triển rất nhanh trong những năm gần đây

### 1.3 Các bài toán trên robot tự hành thông minh

Một số bài toán chính trong robot tự hành như sau ([3]):

**Cảm biến:** Robot tự hành thông minh cần một số loại cảm biến để có thể hiểu được môi trường, định vị và di chuyển tránh vật cản.

Có rất nhiều loại cảm biến, để cảm nhận được đa dạng thông tin của môi trường. Có thể chia làm các nhóm như sau:

- Khoảng cách 1D: Cảm biến khoảng cách hồng ngoại, siêu âm
- Khoảng cách 2D: Lidar
- Cảm biến 3D: Có nhiều loại cảm biến 3D như Intel realsense, Microsoft Kinect, Asus Xtion...
- Ước tính trạng thái robot: GPS, IMU
- Cảm biến lực, momen, cảm biến chạm...
- Âm thanh, giọng nói như microphone, microphone array

- Các loại camera 2D

**Odometry:** là bài toán sử dụng thông tin nhận được từ các cảm biến nhận biết sự di chuyển của robot để ước tính sự thay đổi vị trí của robot qua thời gian. Odometry được sử dụng trong hầu hết các robot tự hành.

**Định vị:** Bài toán định vị giúp trả lời câu hỏi robot đang ở đâu, từ đó có cơ sở để thực hiện các tác vụ khác như tạo bản đồ, xác định hướng di chuyển.

**Xây dựng bản đồ:** Dựa trên dữ liệu từ các loại cảm biến, từ odometry và định vị robot, robot sử dụng các thuật toán để xây dựng bản đồ của môi trường.

**Điều hướng robot:** Từ bài toán định vị và khi có được bản đồ và nhiệm vụ di chuyển từ vị trí hiện tại tới một vị trí đích. Robot sẽ tính toán quỹ đạo và điều khiển di chuyển tới đích.

**Tránh vật cản:** Trong quá trình di chuyển, robot phải phát hiện được các vật cản (bao gồm cả tĩnh và động) và tránh vật cản, sau đó thiết lập lại quỹ đạo di chuyển tới đích.

Vấn đề xuyên suốt trong các bài toán của robot tự hành thông minh đó là các dữ liệu đều không chắc chắn, các bài toán trên đều dựa vào các mô hình xác suất để mô hình hóa được trình bày trong tài liệu [4].

## 1.4 Các nghiên cứu tránh vật cản trong robot tự hành thông minh

Khả năng phát hiện và tránh vật cản theo thời gian thực là rất quan trọng trong robot tự hành. Do đó có rất nhiều nghiên cứu về giải pháp cho vấn đề này. Có nhiều loại cảm biến được sử dụng, nhiều giải thuật được sử dụng. Các loại cảm biến sử dụng như cảm biến khoảng cách hồng ngoại, siêu âm với ứng dụng trên các thiết bị nhúng cấu hình thấp [5, 2]. Các phương pháp sử dụng cảm biến laser radar được trình bày trong [6, 7, 8]. Các thuật toán phổ biến được dùng để phát hiện và tránh vật cản như Virtual Force Field (VFF) [9], Vector Field Histogram (VFH) [10], Dynamic Window Approach (DWA) [11]...

### 1.4.1 Thuật toán Virtual Force Field (VFF)

Được áp dụng cho điều khiển tránh vật cản trình bày trong [12, 9]. Ý tưởng của giải thuật này là tạo một ô lưới quanh robot. Khi có dữ liệu có vật cản từ cảm biến, ô tương ứng sẽ được đặt là bị chiếm dụng bởi vật cản với một tỉ số chiếm dụng (thể hiện cho sự không chắc chắn), nhiệm vụ của thuật toán là tính toán một lực để đưa robot xa khỏi ô bị chiếm dụng đó.

$$F(i, j) = \frac{F_{cr}C(i, j)}{d^2(i, j)} \left[ \frac{x_t - x_0}{d(i, j)}\dot{x} + \frac{y_t - y_0}{d(i, j)}\dot{y} \right] \quad (1.1)$$

Trong đó:

- $F(i, j)$  Lực ảo chống lại vật cản tại ô (i,j)
- $F_{cr}$  Hằng số lực chống lại vật cản
- $d(i, j)$  Khoảng cách giữa ô (i,j) và robot
- $C_{i,j}$  Độ chắc chắn tại ô (i,j)
- $x_0, y_0$  Tọa độ robot
- $x_i, y_j$  Tọa độ của ô (i,j)

Lực  $F_r$  đưa robot tránh khỏi các vật cản xuất hiện xung quanh robot là tổng của các lực tại mỗi ô lưới.

$$F_r = \sum_{i,j} F(i, j) \quad (1.2)$$

Trong khi đó, lực  $F_t$  kéo robot đi từ điểm hiện tại tới điểm đích như sau:

$$F_t = F_{ct} \left[ \frac{x_t - x_0}{d(t)}\dot{x} + \frac{y_t - y_0}{d(t)}\dot{y} \right] \quad (1.3)$$

Trong đó:

- $F_{ct}$  Hằng số lực kéo robot tới đích
- $d(t)$  Khoảng cách giữa robot và điểm đích
- $x_t, y_t$  Tọa độ của điểm đích

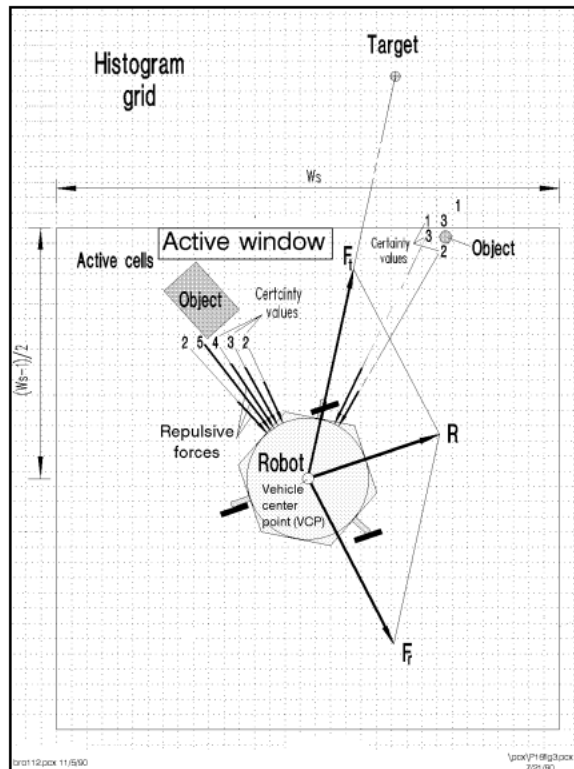
Hợp lực của 2 loại lực này là lực chính kéo robot di chuyển.

$$R = F_t + F_r \quad (1.4)$$

Từ đó tính được hướng di chuyển của robot để điều khiển bánh xe di chuyển, đưa robot tới đích và tránh được vật cản. Theo [12], phương pháp này có một số ưu điểm như sau:

- Phương pháp này không xác định được đường viền cạnh của vật cản, nhưng có thể xác định được cụm các vị trí có vật cản.
- Phương pháp này không yêu cầu robot phải dừng lại để thực hiện lấy dữ liệu và tính toán. Trong điều kiện lý tưởng, phương pháp này giúp robot có thể tránh được tất cả vật cản trong khi vẫn di chuyển với vận tốc tối đa.
- Việc cập nhật bản đồ lưới và điều hướng sử dụng bản đồ lưới là hai nhiệm vụ hoàn toàn không phụ thuộc vào nhau, nhưng có thể đồng bộ với nhau để tối ưu tính toán.





Hình 1.4: Virtual Force Field [1]

- Phương pháp này có thể dễ dàng tích hợp nhiều loại cảm biến để bổ sung thông tin vào bản đồ ô lưới.

Tuy nhiên, theo [1], phương pháp này có một số điểm hạn chế như: Các trường hợp mắc bẫy do cực tiểu địa phương, không thể di chuyển qua giữa các vật cản gần nhau, lưỡng lự khi có vật cản, lưỡng lự trong lối đi hẹp

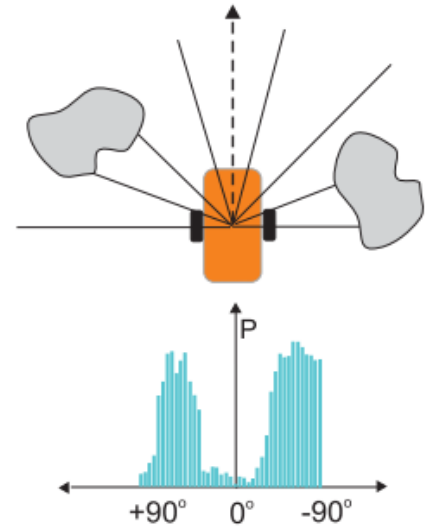
Do đó, phương pháp Vector Field Histogram cải thiện các hạn chế của phương pháp Vector Force Field

### 1.4.2 The Vector Field Histogram

Phương pháp VFH [10] sử dụng kỹ thuật hai giai đoạn giảm dữ liệu và ba mức thể hiện dữ liệu:

- Mức biểu diễn dữ liệu cao nhất giữ mô tả chi tiết của môi trường robot, bản đồ lưới 2 chiều được cập nhật liên tục theo thời gian thực như trong phương pháp VFF.
- Mức ở giữa, một biểu đồ H một chiều được dựng quanh vị trí tức thời của robot. H bao gồm  $n$  góc với độ rộng  $\alpha$ , phép chuyển đổi từ C sang H.
- Mức biểu diễn dữ liệu thấp nhất là đầu ra của thuật toán VFH: các giá trị tham chiếu cho động cơ và bánh xe điều khiển robot.

Phương pháp này có thể phát hiện được lỗi đi đủ cho robot đi qua giữa các vật cản. Phương pháp VFF dễ bị ảnh hưởng bởi sai số của cảm biến, với phương pháp này, sử dụng làm mịn biểu đồ H đã giảm trọng số của các giá trị sai ngẫu nhiên của cảm biến, do đó phương pháp này mạnh hơn phương pháp VFF. Vẫn bị hiện tượng mắc bẫy khi vào các trường hợp "dead-end" (đường cụt) do sử dụng phương pháp cục bộ. Tốc độ di chuyển tối đa khi điều khiển robot bằng VFH bị giới hạn bởi tốc độ lấy mẫu của cảm biến.

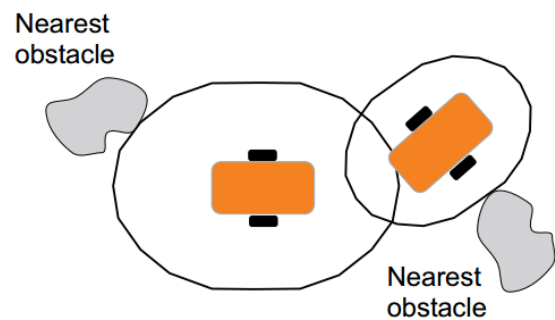


Hình 1.5: Vector Field Histogram [2]

Dựa trên phương pháp này, [13] đã đề xuất phương pháp được gọi là VFH+ thực hiện 4 giai đoạn giảm dữ liệu từ bản đồ lưới chiếm dụng hai chiều xuống thành dữ liệu điều khiển hướng di chuyển của robot. Sử dụng một số cải tiến để giúp robot có thể di chuyển tốt hơn.

### 1.4.3 Phương pháp "Bong bóng phản ứng" tránh vật cản

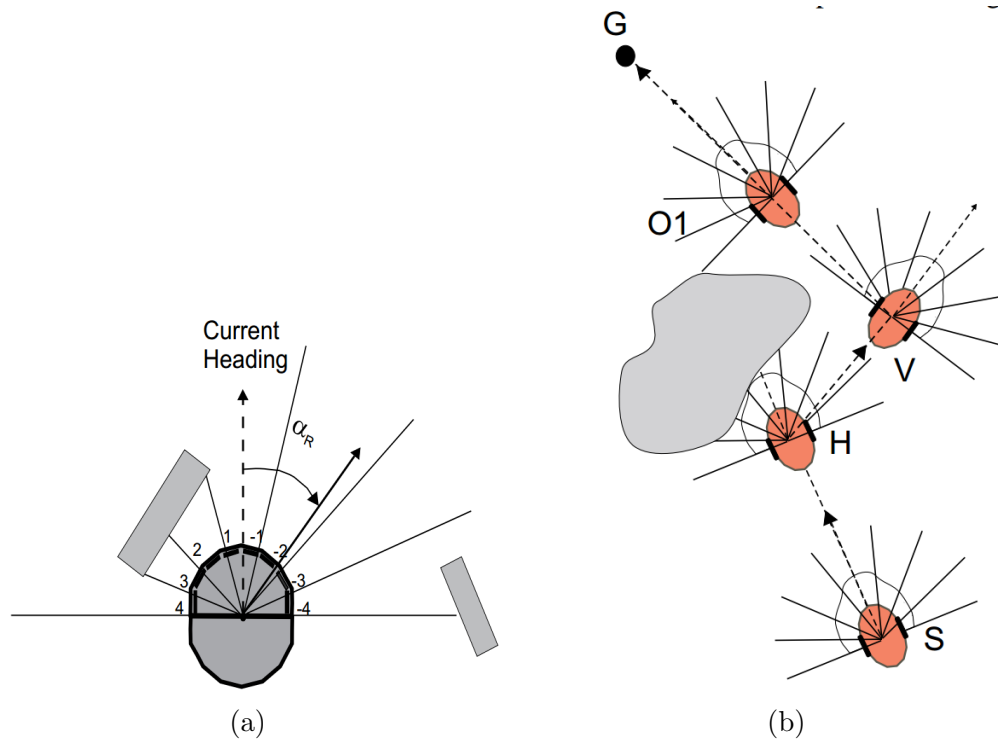
Phương pháp này được đề xuất trong [14], phương pháp này định nghĩa một "bong bóng" bao quanh robot thể hiện không gian lớn nhất có thể di chuyển mà không gặp vật cản. Hình dạng và kích thước của bong bóng được xác định đơn giản dựa trên hình dạng vật lý của đế robot và từ thông tin của cảm biến (như Hình1.6).



Hình 1.6: Bong bóng phản ứng [2]

Một phương pháp cải tiến được đề xuất trong [15]. Tác giả đã cải tiến hình dạng của bong bóng, không phải là hình dạng cố định như trong [14] mà hình dạng và kích thước được điều chỉnh liên tục dựa vào tốc độ di chuyển của robot. Khi các cảm biến phát hiện vật cản nằm trong giới hạn của bong bóng phản ứng, một thuật toán được dùng để xác định hướng của vật cản so với hướng di chuyển của robot Hình1.7a. Một quá trình di chuyển của robot sử dụng phương pháp này được minh họa tại Hình1.7b.

Cũng theo tác giả [15], phương pháp này có ưu điểm là có thể triển khai trên các thiết bị giá thành thấp, chi phí tính toán thấp nhưng có thể hoạt động theo



Hình 1.7: Phương pháp bong bóng phản ứng động

thời gian thực. Tuy nhiên, nó cũng có một số điểm hạn chế như quá trình di chuyển chưa mượt, robot khó xử lý khi gặp nhiều vật cản đồng thời ...

## 1.5 Nội dung nghiên cứu

Với mong muốn tiếp cận với công nghệ thế giới trong lĩnh vực robot tự hành, xe tự lái, luận văn này thực hiện 2 nhiệm vụ chính:

- Cấu hình robot ứng dụng SLAM trên nền tảng hệ điều hành ROS
- Tăng cường phát hiện và tránh vật cản cho robot bằng đa cảm biến.

## Chương 2

# Cơ sở lý thuyết

2.1 Bài toán về nhiều trong robot tự hành

2.2 Bài toán SLAM 2D

2.3 Bài toán tạo định vị, tạo bản đồ, điều hướng và tránh vật cản

2.4 Hệ điều hành robot ROS và các ứng dụng

## Chương 3

# Thực nghiệm

3.1 Đặt vấn đề

3.2 Giới thiệu platform robot của nhóm

3.3 Giới thiệu platform robot của nhóm

3.4 Thực nghiệm và đánh giá kết quả

## Chương 4

# Kết luận và tầm nhìn

### 4.1 Kết luận

### 4.2 Tầm nhìn

# Tài liệu tham khảo

- [1] Y. Koren, J. Borenstein, *et al.*, “Potential field methods and their inherent limitations for mobile robot navigation,” in *ICRA*, vol. 2, pp. 1398–1404, 1991.
- [2] I. Susnea, V. Minzu, and G. Vasiliu, “Simple, real-time obstacle avoidance algorithm for mobile robots,” in *CI 2009*, 2009.
- [3] M. Wise and F. Robotics, “Understanding the Basics of AMR Technology: What You Need to Know,” *autonomous Mobile Robot Conference*.
- [4] T. Sebastian, B. Wolfram, and F. Dieter, *Probabilistic Robotics*. 2005.
- [5] Q. Dongyue, H. Yuanhang, and Z. Yuting, “The investigation of the obstacle avoidance for mobile robot based on the multi sensor information fusion technology,” *Int. J. Mat. Mech. Manuf*, vol. 1, pp. 366–370, 2013.
- [6] M. Gao, J. Tang, Y. Yang, Z. He, and Y. Zeng, “An obstacle detection and avoidance system for mobile robot with a laser radar,” in *2019 IEEE 16th International Conference on Networking, Sensing and Control (ICNSC)*, pp. 63–68, 2019.
- [7] P. Wu, S. Xie, H. Liu, J. Luo, and Q. Li, “A novel algorithm of autonomous obstacle-avoidance for mobile robot based on LIDAR data,” in *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, IEEE, dec 2015.
- [8] Y. Peng, D. Qu, Y. Zhong, S. Xie, J. Luo, and J. Gu, “The obstacle detection and obstacle avoidance algorithm based on 2-D lidar,” in *2015 IEEE International Conference on Information and Automation, ICIA 2015 - In conjunction with 2015 IEEE International Conference on Automation and Logistics*, no. August, pp. 1648–1653, 2015.
- [9] J. Borenstein and Y. Koren, “Real-time obstacle avoidance for fast mobile robots,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, pp. 1179–1187, sep 1989.

- [10] J. Borenstein, Y. Koren, *et al.*, “The vector field histogram - fast obstacle avoidance for mobile robots,” vol. 7, no. 3, pp. 278–288, 1991.
- [11] D. Fox, W. Burgard, and S. Thrun, “The dynamic window approach to collision avoidance,” *IEEE Robotics & Automation Magazine*, vol. 4, pp. 23–33, mar 1997.
- [12] J. Borenstein and Y. Koren, “High-speed obstacle avoidance for mobile robots,” pp. 382 – 384, 09 1988.
- [13] I. Ulrich and J. Borenstein, “Vfh+: reliable obstacle avoidance for fast mobile robots,” in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 2, pp. 1572–1577 vol.2, 1998.
- [14] S. Quinlan and O. Khatib, “Elastic bands: connecting path planning and control,” in *[1993] Proceedings IEEE International Conference on Robotics and Automation*, pp. 802–807 vol.2, 1993.
- [15] I. Susnea, A. Filipescu, G. Vasiliu, G. Coman, and A. Radaschin, “The bubble rebound obstacle avoidance algorithm for mobile robots,” in *IEEE ICCA 2010*, pp. 540–545, 2010.