

# Sự kết hợp đa cảm biến dựa trên ROS cho độ chính xác Hệ thống định vị và SLAM

Yi-Xiang, Wang

Đại học Khoa học và Công nghệ Quốc gia Yunlin

Văn Lâm, Đài Loan

M10717011@gmail.yuntech.edu.tw

Ching-Lung, Chang

Đại học Khoa học và Công nghệ Quốc gia Yunlin

Văn Lâm, Đài Loan

chang@yuntech.edu.tw

Trước xu hướng tỷ lệ sinh thấp và xã hội già hóa, làm thế nào để thiết kế robot thông minh được yêu cầu trong các lĩnh vực khác nhau nhằm giảm bớt sự thiếu hụt nguồn nhân lực đã trở thành một trong những chủ đề nghiên cứu quan trọng trong những năm gần đây. Để đáp ứng khả năng di chuyển của robot thông minh, việc định vị chính xác trong nhà là một chức năng quan trọng

Trong bài báo này, chúng tôi sử dụng Hệ thống vận hành robot (ROS) làm nền tảng, kết hợp với các cảm biến và công nghệ định vị khác nhau, chẳng hạn như Phát hiện ánh sáng và phạm vi (LiDAR) Đơn vị đo quán tính (IMU)

máy đo đường và bằng thông siêu rộng (UWB), đồng thời hợp nhất các dữ liệu này bằng Bộ lọc Kalman mở rộng (EKF) để cung cấp khả năng định vị chính xác. Do đó, chúng ta có thể đạt được Bản đồ và Bản đồ hóa Đồng thời (SLAM) tính tế hơn, đáp ứng nhu cầu ứng dụng robot thông minh. Kết quả thực nghiệm cho thấy khoảng cách sai số trung bình của robot di động trong hệ thống của chúng tôi có thể được giới hạn ở mức 10cm.

Từ khóa: Xe dẫn đường tự động (AGV), Robot Hệ điều hành (ROS), Bằng thông siêu rộng (UWB), Ánh sáng Phát hiện và phân loại (LiDAR), Định vị và lập bản đồ đồng thời (SLAM)

## I. GIỚI THIỆU

Khi công nghệ tiến bộ, xã hội già đi và tỷ lệ sinh thấp hơn, để giảm các tác động liên quan, xuất hiện nhu cầu về nhiều loại robot thông minh, thúc đẩy các quốc gia đầu tư nhiều nguồn lực vào ngành công nghiệp robot thông minh.

Đối với robot di động, lập bản đồ hiện trường và định vị chính xác trong nhà là những chức năng quan trọng. Có bản đồ hiện trường tốt và thông tin định vị có thể cung cấp khả năng lập kế hoạch đường đi, điều khiển di động chính xác và khả năng tránh chướng ngại vật.

Về mặt định vị, nó được chia thành định vị ngoài trời và định vị trong nhà. Định vị ngoài trời chủ yếu dựa vào hệ thống GPS để cung cấp chức năng định vị. Đối với định vị trong nhà, dựa trên độ chính xác và chi phí định vị mà có nhiều công nghệ khác nhau để lựa chọn, chẳng hạn như định vị quán tính, dựa trên ước tính cường độ tín hiệu đến định vị Bluetooth/Wi-Fi và định vị UWB (Ultra-wideband) dựa trên thời gian truyền tín hiệu. và định vị LiDAR bằng phạm vi laser.

ROS (Hệ điều hành Robot) là một nền tảng vận hành được thiết kế dành riêng cho việc điều khiển robot, bài viết này dựa trên nền tảng ROS, kết hợp với định vị quán tính, LiDAR, UWB và các cảm biến khác, kết hợp với Bộ lọc Kalman mở rộng (EKF) để đạt được ánh xạ trường chính xác và hệ thống định vị.

## II. CÔNG TRÌNH LIÊN QUAN

Khi robot đang di chuyển điều khiển, nó muốn đi vững chắc theo con đường đã định để đến vị trí được chỉ định thì chức năng định vị rất quan trọng. Trong những năm gần đây, do sự trưởng thành của công nghệ LiDAR và sự đột phá của thuật toán phần mềm, công nghệ SLAM kết hợp với sự phát triển của chức năng quét phạm vi laser LiDAR liên tục được cải tiến, sự ra đời của robot hút bụi và máy bay không người lái khiến SLAM trở thành một chức năng quan trọng của robot tự hành. Tiếp theo, chúng tôi sẽ giới thiệu thuật toán SLAM và các bài viết liên quan.

### A. Bản địa hóa và lập bản đồ đồng thời

Nguồn gốc sớm nhất được phát triển trong lĩnh vực robot. Ban đầu, điều này có nghĩa là robot bắt đầu từ một môi trường không xác định, hoạt động trong môi trường này và có được vị trí cũng như trạng thái của riêng mình thông qua việc quan sát nhiều lần thông tin môi trường, sau đó tìm hiểu môi trường xung quanh dựa trên vị trí của chúng để xây dựng bản đồ. Hiện nay, theo cảm biến cho SLAM có thể chia thành hai loại: SLAM laser và SLAM trực quan.

Nguồn gốc sớm nhất được phát triển trong lĩnh vực robot. Ban đầu, điều này có nghĩa là robot bắt đầu từ một môi trường không xác định, hoạt động trong môi trường này và có được vị trí cũng như trạng thái của riêng mình thông qua việc quan sát nhiều lần thông tin môi trường, sau đó tìm hiểu môi trường xung quanh dựa trên vị trí của chúng để xây dựng bản đồ. Hiện nay, theo cảm biến cho SLAM có thể chia thành hai loại: SLAM laser và SLAM trực quan.

Hiện nay, thuật toán SLAM phổ biến nhất là Gmapping SLAM được phát triển bởi [1] năm 2007, [2] đã phát triển Hector SLAM được đề xuất vào năm 2010 và [3] đã phát triển Công cụ vẽ bản đồ SLAM được đề xuất vào năm 2016. Gmapping SLAM sử dụng thuật toán lọc hạt RBpf để xử lý môi trường mà robot cảm nhận được, làm việc với máy đo đường bánh xe để tính toán vị trí của robot và xây dựng bản đồ. Hector SLAM không cần sử dụng máy đo đường để tính toán, chỉ dựa vào LiDAR để hoàn thành SLAM mà phải sử dụng LiDAR có độ chính xác cao. Người vẽ bản đồ SLAM trước tiên thực hiện một số điểm laser nhất định để xây dựng bản đồ phụ, sau đó các bản đồ phụ được tập hợp thành toàn bộ bản đồ và đưa ra tính năng phát hiện tránh vòng tròn. Bản đồ phụ được tối ưu hóa theo từng thời điểm nhất định, nó tương tự như thuật toán SLAM trực quan cho các môi trường khác nhau.

### B. Kết hợp đa cảm biến trong SLAM

Để khắc phục các vấn đề về môi trường và giảm thiểu lỗi định vị, người ta đã bắt đầu nghiên cứu

sự hợp nhất của các loại cảm biến khác nhau để cải thiện độ chính xác định vị và áp dụng nó cho SLAM trong những năm gần đây.[4] đề xuất một phương pháp mới kết hợp LiDAR và SLAM trực quan để có được vị trí chính xác hơn.

Dựa trên SLAM laser và dựa trên camera một mắt SLAM có nhược điểm là sai số định vị lớn. Việc tích hợp hai thuật toán SLAM có thể cải thiện vấn đề này.[5] đề xuất một phương pháp dựa trên bình phương tối thiểu và thu được số đo bình phương từ các cảm biến UWB để định vị robot trong môi trường trong nhà. Ngoài ra, một thuật toán khởi tạo dựa trên quét LiDAR được đề xuất, sử dụng lợi thế của vị trí ước tính để tìm hướng ban đầu của robot so với bản đồ thu được trước đó.[6] đề xuất một cảm biến quán tính có hỗ trợ tín hiệu vô tuyến bổ sung cho điều hướng trong nhà.

Thí nghiệm tổng hợp dữ liệu giữa hệ thống định vị dựa trên IMU và hệ thống định vị UWB dựa trên xung. Để đạt được EKF cục bộ, [7] đã hợp nhất các giá trị đo quãng đường và phạm vi Pozyx để cải thiện độ chính xác của thuật toán Pozyx. [8] đề xuất kết hợp đồng hồ đo đường và sử dụng cảm biến quét LiDAR để định vị trong nhà, sử dụng tính năng quét LiDAR để bù lỗi tích lũy của đồng hồ đo đường và áp dụng nó vào Bộ lọc Kalman mở rộng để đạt được khả năng định vị robot

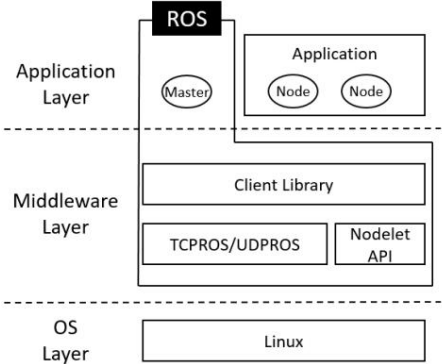
III. LÝ LỊCH

A. Hệ thống vận hành robot

Hệ điều hành Robot còn được gọi là ROS là một phần mềm mã nguồn mở dựa trên môi trường Linux. Nó là một framework được phát triển để viết phần mềm robot. Nó có thể cung cấp các dịch vụ tương tự như hệ điều hành, bao gồm mô tả trừu tượng phần cứng, quản lý trình điều khiển cấp thấp, thực hiện các chức năng chia sẻ, thông báo

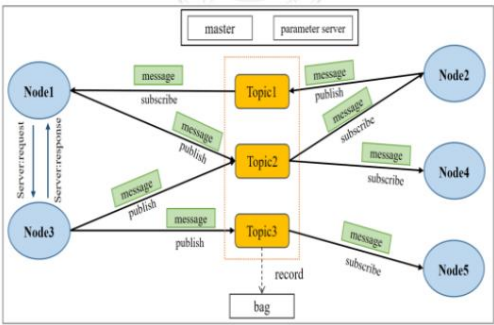
giữa các chương trình, quản lý gói phân phối chương trình, đồng thời cung cấp một số thư viện và công cụ để lấy, tạo, viết và thực thi chương trình giữa các máy tính.

Kiến trúc ROS được hiển thị trong Hình 1, được chia thành ba cấp độ: lớp OS, lớp Middleware và lớp Ứng dụng. Lớp OS cần được xây dựng trong hệ thống Linux, thường sử dụng Ubuntu. Lớp Middleware chủ yếu được sử dụng cho giao tiếp TCPROS/UDPROS. Dựa trên mạng TCP/UDP, hệ thống truyền thông sử dụng đăng ký/xuất bản, máy khách/máy chủ và các mô hình khác để thực hiện nhiều cơ chế truyền thông truyền dữ liệu. Ở tầng Ứng dụng, ROS cần có người quản lý “Master”, chịu trách nhiệm quản lý hoạt động của toàn bộ hệ thống. Ngoài ra, một số lượng lớn các gói chức năng ứng dụng robot đã được chia sẻ trong cộng đồng ROS. Các mô hình trong các gói này được vận hành trong các đơn vị Nút. Thông qua đầu vào và đầu ra tiêu chuẩn ROS dưới dạng giao diện, các nhà phát triển không cần hiểu cơ chế thực hiện trong mô hình, chỉ cần hiểu giao diện Các quy định có thể được sử dụng lại, nâng cao đáng kể hiệu quả phát triển robot.



Hình 1. Kiến trúc ROS

Giới thiệu về giao tiếp giữa các nút trong Hình 2, bao gồm Nút, Chính, Tin nhắn, Chủ đề, Dịch vụ, Túi. Các nút đang chạy chương trình. ROS được thiết kế như một hệ thống đa mô-đun. Một hệ thống robot thường chứa nhiều Nút, ví dụ: một Nút điều khiển LiDAR, một Nút khác xử lý việc định vị robot, v.v. Master là người quản lý chính của ROS. Không có Master, Node không thể tìm được Node khác để trao đổi tin nhắn. Việc truyền thông tin qua Tin nhắn giữa Nút và Nút là một cấu trúc dữ liệu đơn giản. Tin nhắn được gửi theo cách đăng ký/xuất bản. Một nút có thể nhận Tin nhắn bằng cách đăng ký/xuất bản chủ đề. Topic là tên dùng để mô tả nội dung của tin nhắn. Có thể có nhiều Nút xuất bản hoặc đăng ký cùng một Chủ đề cùng một lúc; cũng có thể có một Node xuất bản hoặc đăng ký nhiều Chủ đề cùng một lúc. Trong ROS, còn có chế độ truyền đồng bộ-Dịch vụ, dựa trên mô hình máy khách/máy chủ và chứa hai phần của mô hình truyền thông: một phần dành cho yêu cầu và phần còn lại dành cho phản hồi, tương tự như các dịch vụ Web. Túi là một định dạng được sử dụng để lưu trữ và phát tin nhắn ROS. Túi là một cơ chế quan trọng để lưu trữ dữ liệu. Ví dụ: dữ liệu cảm biến khó thu thập nhưng lại cần thiết cho việc phát triển và thử nghiệm.



Hình 2. Giao tiếp giữa các nút

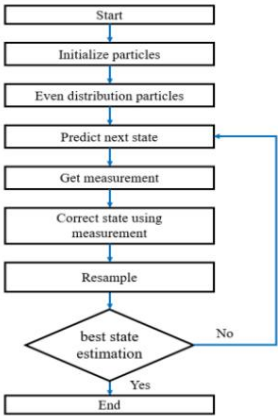
B. SLAM Gmapping

Bản địa hóa và lập bản đồ đồng thời là một vấn đề con gà và quả trứng. Định vị hoàn hảo yêu cầu sử dụng bản đồ không có lỗi, nhưng bản đồ như vậy yêu cầu ước tính vị trí chính xác để xây dựng. Đây là điều kiện khởi đầu cho một chiến lược giải bài toán lặp đi lặp lại.

Gmapping SLAM là một thuật toán SLAM hoàn thiện dựa trên thuật toán lọc hạt RBpf và sau đó được cải tiến, nó khiến chúng tôi phát triển robot tương đối nhanh hơn.

Thuật toán RBpf có nguồn gốc từ bộ lọc hạt. Bộ lọc hạt là bộ lọc đệ quy sử dụng phương pháp Monte Carlo. Nó chuyển một tập hợp các mẫu ngẫu nhiên (gọi các hạt) để biểu thị xác suất hậu nghiệm của các sự kiện ngẫu nhiên, từ chuỗi đo có chứa nhiễu, trạng thái của hệ động được ước tính, so với Bộ lọc Kalman dựa trên không gian trạng thái tuyến tính và phân bố Gaussian về nhiễu, Mô hình không gian trạng thái của bộ lọc hạt có thể phi tuyến tính và phân bố nhiễu có thể thuộc bất kỳ loại nào. Như được hiển thị trong Hình 3, đầu tiên,

khởi tạo các hạt và phân tán đều các hạt, sau đó đưa ra dự đoán và cập nhật chúng bằng so sánh trạng thái đo được, sau đó lấy mẫu lại. Thông qua quá trình đệ quy liên tục như vậy, cuối cùng nó hội tụ thành một nhóm nhỏ các hạt đại diện cho vị trí của robot.



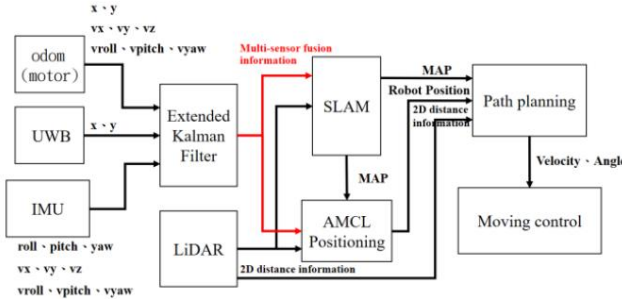
Hình 3. Quá trình lọc hạt

Vấn đề chính của thuật toán RBpf là độ phức tạp cao vì nó đòi hỏi nhiều hạt hơn để xây dựng bản đồ và thường xuyên thực hiện việc lấy mẫu lại. Kể từ khi Số lượng hạt có liên quan chặt chẽ với lượng tính toán, số lượng hạt càng lớn thì thuật toán sẽ càng phức tạp. Do đó, việc giảm số lượng hạt và giảm số lần lấy mẫu lại để ngăn chặn sự suy giảm hạt là một cải tiến của thuật toán RBpf. Tuy nhiên, Gmapping SLAM đề xuất hai điểm trên cơ sở RBpf: cải tiến phân phối đề xuất và lấy mẫu lại có chọn lọc để cải thiện bài toán của thuật toán RBpf. Việc phân phối đề xuất được cải tiến không chỉ xem xét thông tin của mô hình chuyển động (thông tin đo đường) mà còn xem xét thông tin của mô hình đo cuối cùng (thông tin LiDAR) để phân phối đề xuất có thể chính xác hơn và gần với mục tiêu hơn. Việc lấy mẫu lại có chọn lọc tương đối đơn giản. Bằng cách đặt ngưỡng, hành động lấy mẫu lại chỉ được thực hiện khi trọng lượng của các hạt thay đổi vượt quá ngưỡng, điều này làm giảm đáng kể số lần lấy mẫu lại, giúp việc tính toán ít phức tạp hơn và hiệu suất tính toán cũng giảm.

IV. KIẾN TRÚC HỆ THỐNG

Kiến trúc hệ thống được hiển thị trong Hình 4. Có ba loại thông tin cảm biến: máy đo đường (động cơ), bộ đo quán tính (IMU) và băng thông siêu rộng (UWB). Việc tổng hợp dữ liệu được thực hiện thông qua bộ lọc Kalman mở rộng và thông tin đa cảm biến kết hợp và thông tin LiDAR đã được gửi tới SLAM để ánh xạ. Khi robot muốn di chuyển đến mục tiêu,

nó phải biết nó ở đâu trên bản đồ. Tại thời điểm này, Định vị AMCL cần được sử dụng để xác định vị trí của robot và sau đó robot sẽ bắt đầu lập kế hoạch đường đi sau khi biết vị trí của nó. , và thực hiện các hành động điều khiển chuyển động.



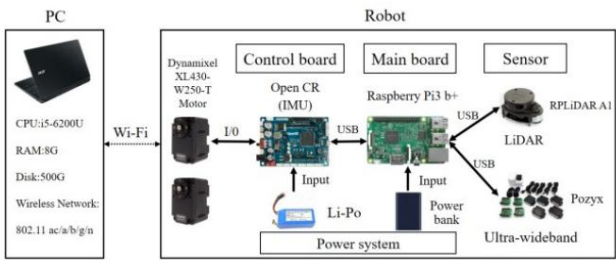
Hình 4. Kiến trúc hệ thống

A. Kiến trúc phần cứng

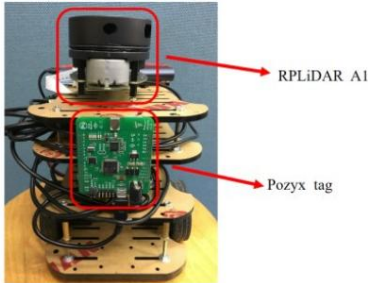
Kiến trúc phần cứng được hiển thị trong Hình 5 được chia thành PC và robot. CPU của PC là i5-6200, RAM có 8G, dung lượng ổ cứng là 500G và hỗ trợ mạng không dây 802.11 ac/a/b/g/n. Robot có 4 bộ phận: bảng điều khiển, bo mạch chính, hệ thống điện và các cảm biến.

Có hai loại thiết bị trong bộ điều khiển, bảng điều khiển OpenCR và động cơ thông minh XL430-W250-T. Cả hai giao tiếp thông qua I/O. Bo mạch chính Raspberry Pi 3 b+ đóng vai trò là lõi của toàn bộ robot và giao tiếp với các bộ phận khác thông qua cổng USB. Có hai loại hệ thống điện, một loại là nguồn sạc dự phòng cho Raspberry Pi 3 b+, loại còn lại là pin Li-Po cho bảng điều khiển OpenCR. Có ba loại phần tử cảm biến, đơn vị đo quán tính (IMU) trên OpenCR, RPLiDAR A1 và Pozyx(UWB), IMU cung cấp thông tin chín trục, RPLiDAR A1 cung cấp thông tin khoảng cách 2D và Pozyx (UWB) cung cấp vị trí tuyệt đối.

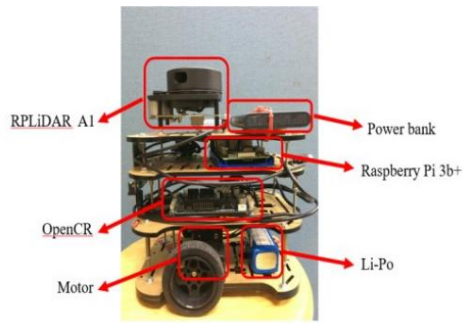
Ngoài ra, Pozyx còn có một mỏ neo và một thẻ, mỏ neo được bố trí ở bốn vị trí trong khu vực. Vị trí của thẻ có được bằng thuật toán. Vì vậy, chúng tôi cài đặt thẻ trên robot để nhận thông tin vị trí. Hình 6 và Hình 7 là biểu đồ tiến và biểu đồ bên của thực thể robot.



Hình 5. Kiến trúc phần cứng



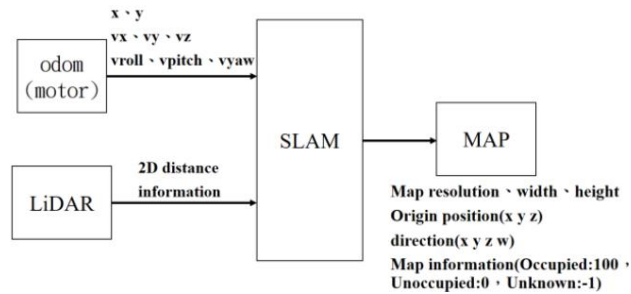
Hình 6. Mặt trước của thực thể robot



Hình 7. Mặt bên của thực thể robot

### B. SLAM Gmapping

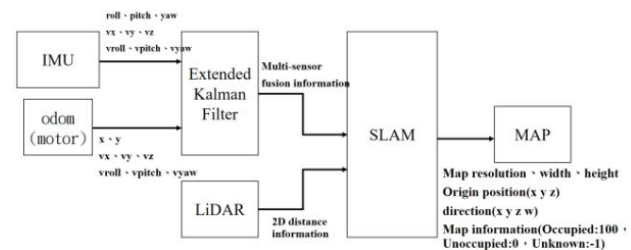
Hình 8 thể hiện kiến trúc Gmapping SLAM. Cấu trúc SLAM yêu cầu thông tin từ đồng hồ đo đường (động cơ) và LiDAR. Đồng hồ đo đường (động cơ) cung cấp  $x, y, vx, vy, vz, vroll, vpitch, vyaw$ . LiDAR cung cấp thông tin khoảng cách 2D. Sau khi xây dựng bản đồ, bản đồ sẽ được tạo ra với một số thông tin liên quan, chẳng hạn như: độ phân giải bản đồ, chiều rộng, chiều cao, vị trí và hướng gốc và thông tin bản đồ (chiếm là 100, không có người ở là 0, không xác định là -1).



Hình 8. Kiến trúc Gmapping SLAM

### C. Sự kết hợp đa cảm biến để định vị và SLAM

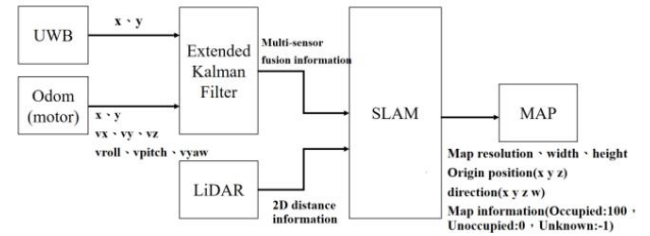
Hình 9 là kiến trúc định vị kết hợp Phương pháp 1odom + IMU. Kết hợp đồng hồ đo đường (động cơ) và đơn vị đo quán tính (IMU) với bộ lọc Kalman mở rộng làm phản ứng tổng hợp dữ liệu cảm biến. Đồng hồ đo đường (động cơ) cung cấp  $x, y, vx, vy, vz, vroll, vpitch, vyaw$ . Đơn vị đo lường quán tính (IMU) cung cấp thông tin cuộn, cao độ, ngáp,  $vx, vy, vz, vroll, vpitch, vyaw$ , sau đó thông tin kết hợp đa cảm biến và thông tin LiDAR được gửi đến SLAM để xây dựng bản đồ.



Hình 9. Phương pháp 1, định vị kết hợp odom+imu

Hình 10 là kiến trúc định vị kết hợp Phương pháp 2odom + UWB. Kết hợp đồng hồ đo đường (động cơ) và băng thông siêu rộng (UWB) với bộ lọc Kalman mở rộng làm phản ứng tổng hợp dữ liệu cảm biến. Đồng hồ đo đường (động cơ) cung cấp  $x, y, vx, vy, vz, vroll, vpitch, vyaw$ . Băng thông siêu rộng (UWB) cung cấp thông tin  $x, y$  và sau đó là sự kết hợp đa cảm biến

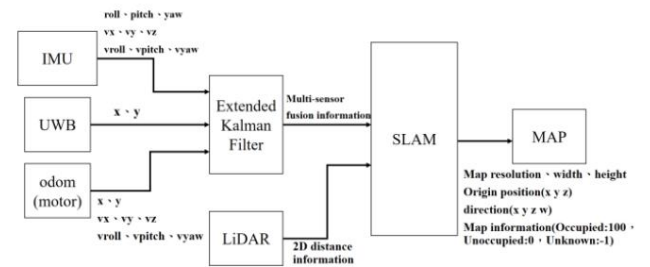
thông tin và thông tin LiDAR được gửi đến SLAM để xây dựng bản đồ.



Hình 10. Phương pháp 2, định vị kết hợp odom+uwb

Hình 11 là kiến trúc định vị kết hợp Phương pháp 3odom + UWB + IMU.

Kết hợp máy đo đường (động cơ), đơn vị đo quán tính (IMU) và băng thông siêu rộng (UWB) với bộ lọc Kalman mở rộng làm phản ứng tổng hợp dữ liệu cảm biến. Đồng hồ đo đường (động cơ) cung cấp  $x, y, vx, vy, vz, vroll, vpitch, vyaw$ . Đơn vị đo quán tính (IMU) cung cấp thông tin cuộn, cao độ, ngáp,  $vx, vy, vz, vroll, vpitch, vyaw$ . Băng thông siêu rộng (UWB) cung cấp thông tin  $x, y$ , sau đó thông tin tổng hợp đa cảm biến và thông tin LiDAR được gửi đến SLAM để xây dựng bản đồ.



Hình 11. Phương pháp 3, định vị kết hợp odom+imu+uwb

Bộ lọc Kalman mở rộng hoạt động theo công thức (1) ~ (5).

Đầu tiên, chúng tôi ước tính vị trí của robot với tiếng ồn ước tính, sau đó sử dụng thông tin từ cảm biến làm phép đo và cập nhật nó để có giá trị thực tế hơn và tiếp tục thực hiện các bước như vậy, để quy được sử dụng để nhận được nhiều hơn vị trí robot chính xác.

Ước tính trạng thái dự đoán:

$$\hat{x}_{(k|k-1)} = f(x_{k-1}, u_k, 0) \quad (1)$$

Hiệp phương sai ước tính dự đoán:

$$P_{(k|k-1)} = F_k P_{(k-1|k-1)} F_k^T + Q_k \quad (2)$$

Ước tính trạng thái được cập nhật:

$$\hat{x}_{(k|k)} = \hat{x}_{(k|k-1)} + K_k \tilde{y}_k \quad (3)$$

Mức tăng Kalman tối ưu:

$$K_k = P_{(k|k-1)} H_k^T S_k^{-1} \quad (4)$$

Hiệp phương sai ước tính được cập nhật:

$$P_{(k|k)} = (I - K_k H_k) P_{(k|k-1)} \quad (5)$$

V. KẾT QUẢ THỰC NGHIỆM

A. Môi trường thí nghiệm

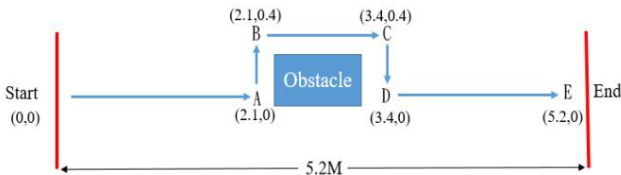
Hình 12 cho thấy khu vực môi trường thực tế được sử dụng cho cuộc thí nghiệm. Chúng tôi đã cố định bốn Neo Pozyx ở bốn vị trí khác nhau.



Hình 12. Diện tích môi trường thực tế

B. Phương pháp thí nghiệm

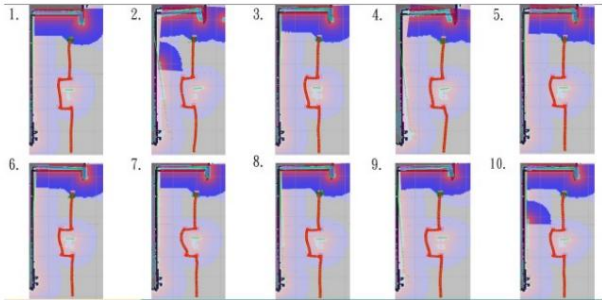
Phương pháp thử nghiệm quy hoạch đường đi được thể hiện trên Hình 13. Chúng tôi quy hoạch một đường đi có tổng chiều dài 5,2 mét, đặt chướng ngại vật trên đường, xuất phát từ điểm xuất phát và vượt qua A (2.1,0), B (2.1, 0,4), C (3,4 0,4), D (3,2, 0) và cuối cùng đến điểm E (điểm cuối) và đo sai số từ điểm cuối.



Hình 13. Lập kế hoạch đường đi có chướng ngại vật

C. Kết quả thí nghiệm

Chúng tôi sử dụng các kiến trúc phương pháp khác nhau, tiến hành tổng cộng mười lần thông qua phương pháp lập kế hoạch đường đi có chướng ngại vật, đồng thời ghi lại đường đi chuyển của robot và đo sai số từ khoảng cách cuối cùng.

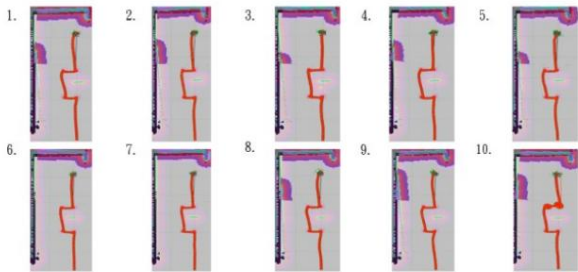


Hình 14. Đường đi của Gmapping SLAM

BẢNG I. Lỗi khoảng cách cuối cùng của Gmapping SLAM

Mục	Khoảng cách cuối cùng ước tính(cm)	Khoảng cách cuối cùng đo (cm)	Lỗi (cm)	Thời gian (S)
1	520	510,8	9,2	52
2	520	508,4	11.6	50
3	520	506.1	13.9	47
4	520	506.2	13,8	47
5	520	509,6	10,4	49
6	520	507.0	13.0	47
7	520	503.0	17,0	53

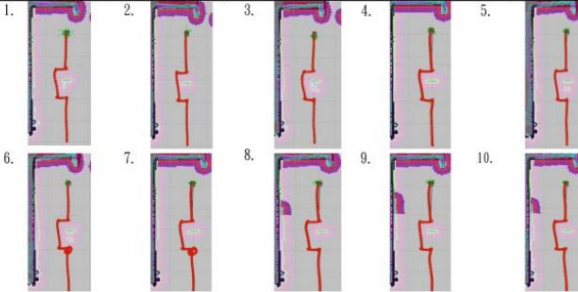
...	520	506,7	13.3	51
9	520	501.7	18.3	50
10	520	507,2	12.8	48



Hình 15. Đường đi chuyển của phương pháp 1, định vị kết hợp odom+imu

BẢNG II. Lỗi khoảng cách cuối cùng của phương pháp 1odom+imu định vị tổng hợp

Mục	Khoảng cách cuối cùng ước tính(cm)	Khoảng cách cuối cùng đo (cm)	Lỗi (cm)	Thời gian (S)
1	520	518,6	1.4	50
2	520	511,5	8,5	52
3	520	516,3	3,7	65
4	520	514.0	6.0	51
5	520	511,9	8.1	50
6	520	515,8	4.2	49
7	520	517,4	2.6	52
...	520	523,5	3,5	50
9	520	518,8	1.2	53
10	520	522,8	2,8	63

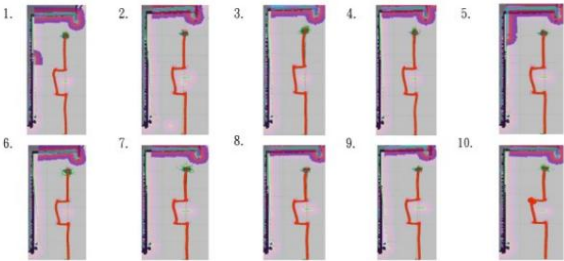


Hình 16. Đường đi chuyển của phương pháp 2, định vị kết hợp odom+uwb

BẢNG III. Lỗi khoảng cách cuối cùng của phương pháp 2, định vị kết hợp odom+uwb

Mục	Khoảng cách cuối cùng ước tính(cm)	Khoảng cách cuối cùng đo (cm)	Lỗi (cm)	Thời gian (S)
1	520	508.3	11.7	52
2	520	514.2	5,8	49
3	520	506.4	13.6	50
4	520	510,6	9,4	50
5	520	506.1	13.9	49
6	520	509.4	10.6	63
7	520	510.3	9,7	60
...	520	509.1	10.9	52
9	520	512,2	7,8	51
10	520	505,6	14.4	53

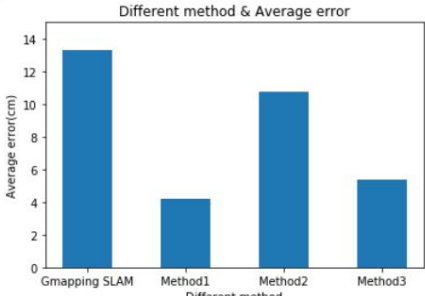




Hình 17. Đường đi chuyển của phương pháp 3, định vị kết hợp odom+imu+uwb

BẢNG IV. Lỗi khoảng cách cuối cùng của phương pháp 3, định vị kết hợp odom+imu+uwb

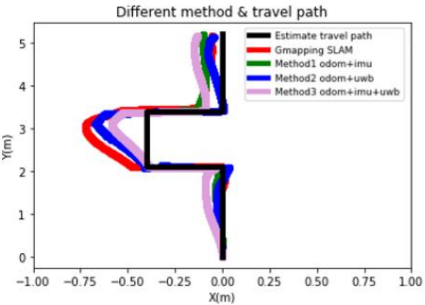
Mục	Khoảng cách cuối cùng ước tính(cm)	Khoảng cách cuối cùng đo (cm)	Lỗi (cm)	Thời gian (S)
1	520	523,6	3.6	50
2	520	511,2	8,8	48
3	520	518,4	1.6	50
4	520	522,7	2.7	49
5	520	510,9	9.1	49
6	520	527,2	7.2	53
7	520	517,9	2.1	51
8	520	512,4	7,6	51
9	520	514,7	5.3	52
10	520	525,8	5,8	53



Hình 18. So sánh các phương pháp khác nhau về sai số trung bình

BẢNG V. So sánh các phương pháp khác nhau về sai số trung bình

Phương pháp	Sai số trung bình của khoảng cách cuối cùng (cm)
Gmapping SLAM	13:33
Phương pháp 1 định vị kết hợp nhất odom + imu	4.2
Phương pháp 2 định vị kết hợp nhất odom + uwb	10,78
Phương pháp 3 định vị kết hợp nhất odom+imu+uwb	5,38



Hình 19. So sánh khoảng cách di chuyển tối đa của các phương pháp khác nhau

BẢNG VI. So sánh khoảng cách di chuyển tối đa của các phương pháp khác nhau

Phương pháp	Sai số trung bình của khoảng cách cuối cùng (cm)
Gmapping SLAM	28,2
Phương pháp 1 Định vị kết hợp odom+imu Phương pháp 2	25,7
Định vị kết hợp nhất odom+uwb Phương pháp 3	25,3
Định vị kết hợp odom+imu+uwb Trong Hình 14	23,6

đến Hình 17, Bảng 1 đến Bảng 4 hiển thị đường đi và dữ liệu lỗi khoảng cách cuối cùng cho từng phương pháp. Cuối cùng, trong Hình 18 và Bảng 5, chúng ta có thể thấy sự so sánh giữa các phương pháp khác nhau về sai số trung bình. Phương pháp định vị kết hợp 2odom+imu là tốt nhất trong tất cả các phương pháp. Sai số trung bình của khoảng cách cuối cùng chỉ là 4,2cm. Trong Hình 19 và Bảng 6 cho thấy sự so sánh giữa các phương pháp khác nhau về khoảng cách di chuyển tối đa. Phương pháp 3,

Định vị kết hợp odom+imu+uwb là tốt nhất trong tất cả các phương pháp. Điều đó có nghĩa là robot có thể bước đi chính xác hơn và có thể vượt qua thuận lợi hơn khi bị bao quanh bởi nhiều chướng ngại vật.

VI. PHẦN KẾT LUẬN

Chúng tôi sử dụng Bộ lọc Kalman mở rộng để hợp nhất nhiều thông tin cảm biến nhằm có được vị trí chính xác và sử dụng nó trong SLAM và lập kế hoạch đường đi. Kết quả thử nghiệm chứng minh rằng phương pháp tổng hợp đa cảm biến của chúng tôi là khả thi và có thể kiểm soát lỗi định vị trong vòng 10cm.

NGƯỜI GIỚI THIỆU

[1] Giorgio Grisetti; Cyrill Stachniss; Wolfram Burgard, "Các kỹ thuật cải tiến để lập bản đồ lưới bằng bộ lọc hạt Rao-Blackwellized", Giao dịch của IEEE về Robotics, Tập. ngày 23 tháng 1 năm 2007, trang 34-46.

[2] Stefan Kohlbrecher, Johannes Meyer, Thorsten Graber, Karen Petersen, Oskar von Stryk và Uwe Klingauf, "Mô-đun nguồn mở Hector để lập bản đồ và điều hướng tự động với Robot cứu hộ", RoboCup 2013; Robot World Cup XVII, 2014, trang 624-631.

[3] Wolfgang Hess, Damon Kohler, Holger Rapp, Daniel Andor, "Đóng vòng lặp thời gian thực trong 2D LIDAR SLAM", Hội nghị quốc tế của IEEE về Robot và tự động hóa (ICRA), Stockholm, Thụy Điển, 2016, trang 1271-1278.

[4] Shao-Hung Chan, Ping-Tsang Wu,Li-Chen Fu, "Bản địa hóa trong nhà 2D mạnh mẽ thông qua Laser SLAM và Visual SLAM Fusion" Hội nghị quốc tế IEEE 2018 về Hệ thống, Con người và Điều khiển học.

[5] Bekir Bostanci, Sercan Tekkok, Emre Soyunmez, Pinar Oguz-Ekim, Faezeh Yeganli, "Thuật toán khởi tạo và định vị nguồn dựa trên LIDAR và UWB cho hệ thống robot tự động" Hội nghị quốc tế lần thứ 11 về Kỹ thuật điện và điện tử (ELECO), Bursa , Thổ Nhĩ Kỳ.

[6] Lukasz Zwiello, Xuyang Li, Thomas Zwick, Christian Ascher, Sebastian Werling, Gert F. Trommer, "Hợp nhất dữ liệu cảm biến trong các hệ thống định vị quán tính được UWB hỗ trợ để điều hướng trong nhà", Hội nghị quốc tế IEEE 2013 về Robot và Tự động hóa, Karlsruhe, Nước Đức.

[7] Teresa Conceicao, Filipe Neves dos Santos, Paulo Costa, Ant'onio Paulo Moreira, "Hệ thống định vị robot trong môi trường ngoài trời khắc nghiệt", ROBOT 2017: Hội nghị về robot Iberia lần thứ ba, trang 215-227

[8] Espinosa F, Santos C, Marrón-Romera M, Pizarro D, Valdés F, Dongil J, "Sự kết hợp đo đường và máy quét laser dựa trên Bộ lọc Kalman mở rộng riêng biệt để hướng dẫn phân đội robot", Cảm biến, Tập. Ngày 11 tháng 11 năm 2011, trang 8339-8357.