DOI: 10.64302/joshusc.v30n1e.1327

ỨNG DỤNG THUẬT TOÁN ĐỊNH VỊ VÀ XÂY DỰNG BẢN ĐỒ ĐỒNG THỜI CHO THIẾT BỊ KHÔNG NGƯỜI LÁI MẶT ĐẤT

Hà Xuân Vinh*, Nguyễn Văn Ân, Hồ Đắc Quý

Khoa Điện, Điện tử và Công nghệ vật liệu, Trường Đại học Khoa học, Đại học Huế

* Email: xuanvinhha@gmail.com

Ngày nhận bài: 22/4/2025; ngày hoàn thành phản biện: 4/5/2025; ngày duyệt đăng: 21/6/2025

TÓM TẮT

Trong xu thế phát triển khoa học của lĩnh vực robotics, thiết bị không người lái mặt đất (Unmanned Ground Vehicle) đã trở thành thiết yếu để sử dụng trong các nhiệm vụ như thăm dò, khám phá, hỗ trợ, giám sát môi trường và những công việc khác trong môi trường không rõ (unknown environment). Các nhiệm vụ này có thể được thực hiện riêng lẻ hoặc cùng với các thiết bị khác trong các nhóm cộng tác với nhiều mức độ thực hiện khác nhau, với khả năng cao về tự lập kế hoạch và ra quyết định. Đối với thiết bị không người lái thông minh, nó phải tự xác định vị trí và xây dựng bản đồ một cách chính xác trong môi trường không rõ. Do đó, bài viết này nói về ứng dụng của thuật toán Sự đồng thời định vị và Xây dựng bản đồ sử dụng bộ lọc Kalman mở rộng (Extended Kalman Filter - Simultaneous Localization and Mapping), được coi như là một tính năng cơ bản, cần thiết cho việc tự chủ động định hướng của thiết bị không người lái.

Từ khóa: Unmanned Ground Vehicle (UGV), Extended Kalman Filter (EKF), Simultaneous Localization and Mapping (SLAM).

1. MỞ ĐẦU

Việc sử dụng các thiết bị không người lái như Unmanned Aerial Vehicles (UAV) hay Unmanned Ground Vehicles (UGV), như Hình 1, trong các ứng dụng quân sự và thương mại đã tăng lên trong những năm qua từ theo dõi mục tiêu, truyền thông, tình báo điện tử và tìm kiếm cứu nạn đến sử dụng trong thương mại, nông nghiệp, giám sát ô nhiễm và theo dõi giao thông hàng hải [1]. Đối với một thiết bị không người lái thông minh, nó phải tự xác định được vị trí của mình để từ đó cho phép nó tự điều hướng trên mọi địa hình. Điều này thường được thực hiện bằng cách sử dụng tín hiệu hệ thống định vị toàn cầu (GPS). Tuy nhiên tín hiệu GPS không đáng tin cậy khi thiết bị di chuyển

trong khoảng cách nhỏ hay trong các tòa nhà hoặc hầm mỏ, nơi mà tín hiệu GPS bị suy hao nhiều. Để định vị chính xác vị trí, thiết bị không người lái phải sử dụng các loại cảm biến khác cũng như bản đồ của môi trường mà nó đang đứng. Tuy nhiên, đối với những môi trường chưa biết rõ, thiết bị không người lái sẽ phải tự động lập/cập nhật bản đồ cho môi trường đó để dùng cho việc định vị và điều hướng của thiết bị. Từ đó kỹ thuật SLAM được phát triển và ứng dụng vào các thiết bị không người lái như hiện nay.

(a)

(b)

Hình 1. a) Unmanned Aerial Vehicles (UAV) b) Unmanned Ground Vehicles (UGV)

Thuật toán SLAM liên quan đến một vật thể chuyển động (ví dụ như một robot, UGV hay UAV), sử dụng ít nhất một bộ cảm biến để thu thập thông tin về môi trường xung quanh của nó, ví dụ như bản đồ, … (một máy ảnh, một máy quét laser, một sonar: chúng được gọi là cảm biến exteroceptive). Đồng thời, các vật thể chuyển động này có thể kết hợp bộ cảm biến khác để xác định các vị trí chuyển động riêng của mình (bộ encoder, cảm biến gia tốc, cảm biến đo vận tốc gốc gyrometers: chúng được gọi là cảm biến proprioceptive).

SLAM làm một vấn đề cơ bản và phức tạp trong nghiên cứu về UGV. Nó đã được nghiên cứu hơn hai mươi năm. Giải pháp cho vấn đền SLAM được công bố bởi Hugh Durrant [2] là một trong những thành công đáng chú ý của cộng đồng Robotics. Ngoài ra, các cơ sở lý thuyết thống kê liên quan đến SLAM được mô tả đầu tiên bởi Smith [3] và Durrant – Whyte [4], nó cung cấp các thuật toán ước lượng xác suất cho mối tương quan giữa các tính năng (features) và vị trí của UGV. Hơn nữa, họ là những người đầu tiên đã đề xuất phương pháp tinh chỉnh sự không chắc chắn về mặt hình học của các ước lượng trong các quan sát liên tục. Một các tiếp cận khác, dựa trên nền cơ bản của Extended Kalman Filter – SLAM (EKF-SLAM), sử dụng bộ EKF lớn chứa tất cả các thông số tính năng để thực hiện SLAM [5-7], hay để giảm thời gian tính toán cũng như dung lượng thông tin của các tính năng, bộ FastSLAM được nghiên cứu và ứng dụng [8-9]. Mặc dù việc sử dụng EKF trong SLAM là phổ biến, tuy nhiên rõ ràng là có vấn đề khi làm việc trong môi trường lớn, kích thước bộ EKF lớn nên thời gian tính toán vượt ra ngoài thời gian tính toán thời gian thực, nó không thể nào chứa được toàn bộ thông số bản đồ quy mô lớn và không đáp ứng được thời gian tính toán thực tế [10-11]. Những

nghiên cứu gần đây về SLAM tập trung vào việc cải tiến EKF-SLAM, ví dụ như chia nhỏ các bộ lọc cho các các bản đồ nhỏ thành phần [12-13], sau đó nối các bản đồ phụ này thành một bản đồ của môi trường hoàn chỉnh.

Bài viết này đề cập đến việc áp dụng EKF=SLAM cho thiết bị không người lái. Giải thuật góp phần cung cấp cho các nghiên cứu về lĩnh vực UGV một trong những giải pháp cho bài toán SLAM, một nền tảng cơ bản và rõ ràng về mặt khái niệm để hợp nhất dữ liệu các cảm biến và quản lý các thông số môi trường bất định trong ứng dụng UGV cùng với thực nghiệm thực tế, là nền tảng cho các nghiên cứu thuật toán SLAM sau này.

2. CƠ SỞ LÝ THUYẾT

2.1. Bộ lọc Kalman mở rộng (Extended Kalman Filter - EKF)

R. E. Kalman đã đề xuất một kỹ thuật lọc đệ quy mới vào năm 60. Welch và cộng sự [14] giới thiệu chi tiết về DKF và EKF. Bộ lọc Kalman mở rộng (EKF) mở rộng phạm vi của bộ lọc Kalman sang các bài toán lọc tối ưu phi tuyến tính bằng cách tạo ra một xấp xỉ Gauss cho phân phối kết hợp của trạng thái x và phép đo y bằng phép biến đổi dựa trên chuỗi Taylor. Mô hình không gian trạng thái được sử dụng trong EKF là:

$$x_k = f(x_{k-1}, k-1) + q_{k-1} \tag{1}$$

$$y_k = h(x_k, k) + r_k \tag{2}$$

Giống như bộ lọc Kalman rời rạc, EKF được chia thành hai bước như sau:

- *Bước dự đoán (the prediction step):*

$$m_k^- = f(m_{k-1}, k-1) \tag{3}$$

$$P_k^- = F_X(m_{k-1}, k-1)P_{k-1}F_X^T(m_{k-1}, k-1) + Q_{k-1} \tag{4}$$

- *Bước cập nhật (the update step):*

$$v_k = y_k - h(m_k^-, k) \tag{5}$$

$$S_k = H_X(m_k^-, k)P_k^- H_X^T(m_k^-, k) + R_k \tag{6}$$

$$K_k = P_k^- H_X^T(m_k^-, k)S_k^{-1} \tag{7}$$

$$m_k = m_k^- + K_k v_k \tag{8}$$

$$P_k = P_k^- - K_k S_k K_k^T \tag{9}$$

Trong đó, m_k^- và P_k^-: lần lượt là giá trị trung bình và hiệp phương sai dự đoán của trạng thái tại bước thời gian k trước khi thực hiện phép đo; m_k and P_k: lần lượt là giá trị trung bình và hiệp phương sai ước tính của trạng thái tại bước thời gian k sau khi thực hiện phép đo; v_k: độ đổi mới hoặc phần dư của phép đo tại bước thời gian k; S_k: hiệp phương sai dự đoán phép đo tại bước thời gian k; K_k: độ lợi của bộ lọc, cho biết

mức độ hiệu chỉnh của dự đoán tại bước thời gian k. Các ma trận $F_X(m, k-1)$ và $H_X(m, k)$: là các ma trận Jacobians của các hàm *f* và *h*, với các phần tử như sau:

$$[F_X(m, k-1)]_{j,j'} = \frac{\partial f_j(x,k-1)}{\partial x_{j'}}|_{x=m} \quad (10)$$

$$[H_X(m, k)]_{j,j'} = \frac{\partial h_j(x,k)}{\partial x_{j'}}|_{x=m} \quad (11)$$

2.2. Các mô hình đo lường cảm biến

2.2.1. Mô hình ràng buộc vận tốc từ cảm biến encoder

Trong thiết bị không người lái mặt đất dạng SSMR 4-TW (Four Track Wheel Skid-steered Mobile Robot), Hình 2. Các đại lượng v_L, v_R được định nghĩa là vận tốc tuyến tính trái và phải của UGV, do đó:

$$v_L = v_{1x} = v_{2x} \quad (12) \qquad v_R = v_{3x} = v_{4x} \quad (13)$$

Trong đó, v_{1x},v_{2x},v_{3x} và v_{4x}: là các vận tốc tuyến tính trung tâm các bánh xe tương ứng trước bên trái, sau bên trái, trước bên phải và sau bên phải. Hệ số trượt dọc của bánh xe bên trái và bên phải λ_L, λ_R được định nghĩa là tỷ số giữa vận tốc bánh xe và vận tốc tâm [15], như sau

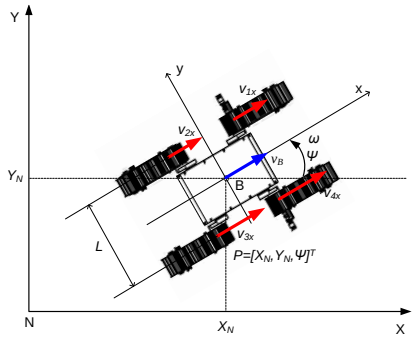
$$\lambda_L = \frac{r\omega_L - v_L}{r\omega_L} \quad (14) \qquad \lambda_R = \frac{r\omega_R - v_R}{r\omega_R} \quad (15)$$

Trong đó, ω_L, ω_R: là tốc độ góc của bánh xe bên trái và bên phải của UGV. Từ các phương trình (14) và (15), có thể thấy độ trượt của bánh xe là $\lambda \in [0,1]$ nếu bánh xe chịu lực kéo và $\lambda \in (-\infty, 0]$ khi bánh xe phanh. Sử dụng định nghĩa về độ trượt trong (14) và (15), ta có:

$$v_L = (1-\lambda_L)r\omega_L \quad (16) \qquad v_R = (1-\lambda_R)r\omega_R \quad (17)$$

Do cấu trúc cơ học đối xứng của UGV, có thể giả định rằng trọng tâm của UGV nằm ở tâm hình học (Centre Of the Geometry - COG) của khung thân. Tọa độ của cảm biến AHRS nằm tại COG của khung thân trong tọa độ B (Body cordinate), ta có vectơ vận tốc AHRS $v_B = [v_{Bx}, v_{By}, v_{Bz}]^T$ trong tọa độ B. Từ các công thức (16) và (17), sử dụng hai phép đo từ hai cảm biến encoder đạt tại hai bánh xe và các hệ số trượt ước lượng, chúng ta thu được vận tốc dọc v_{Bx}:

$$v_{Bx} = \frac{v_L + v_R}{2} = \frac{r}{2}[(1-\lambda_L)\omega_L + (1-\lambda_R)\omega_R] \quad (18)$$



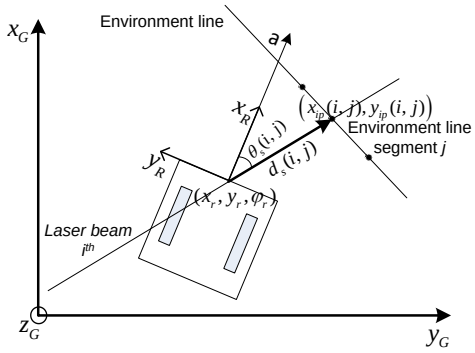
Hình 2. Động học của UGV 4-TW SSMR

Vì bốn bánh xe theo dõi của UGV luôn tiếp xúc với mặt đất và cảm biến AHRS được cố định trên nền khung thân thiết bị, nên các ràng buộc về vận tốc v_{By},v_{Bz} theo hướng trục y và trục z đối với thiết bị AHRS có thể được đơn giản hóa thành bằng 0. Các nhiễu của vận tốc dọc v_{Bx} có thể được xem là nhiễu của tốc độ góc ω_L và ω_R của các bánh xe bên trái và bên phải được đo bằng cảm biến encoder. Giả sử rằng các nhiễu đối với ω_L và ω_R có phân phối chuẩn Gaussian với giá trị trung bình bằng không và các phương sai tương ứng và không có tương quan chéo giữa tiếng ồn của ω_L và ω_R [16], phương sai nhiễu của vận tốc dọc v_{Bx} có thể được biểu thị theo công thức:

$$\sigma_{vBx}^2 = \frac{r^2}{4}((1-\lambda_R)^2(\omega_R) + (1-\lambda_L)^2(\omega_L)) \quad (19)$$

Phương sai σ_{vBy}^2 và σ_{vBz}^2 của các vận tốc v_{By} và v_{Bz} cũng được biểu thị bằng không. Để có được vận tốc dọc v_{Bx}và phương sai nhiễu của nó, chúng ta phải ước lượng các độ trượt $\lambda_L$và λ_R.

2.2.2. Mô hình cảm biến đo khoản cách laser (Laser Range Finder model - LRF)



Hình 3. Các điểm giao nhau giữa tia laser và các đường của môi trường

Trong môi trường phẳng, cảm biến LRF cung cấp tập hợp các khoảng cách $d_s = [d_{s0^o}, \ldots, d_{s180^o}]$ đến các chướng ngại vật ở các góc $\theta_s = [0^o, \ldots, 180^o]$ trong mỗi bước [3]. Một điểm phản xạ được mô phỏng bằng cách tính toán các điểm giao nhau $\left(x_{ip}(i,j), y_{ip}(i,j)\right)$ giữa đường tia laser thứ i và đoạn thẳng môi trường *j (j = 1,..., N)*, xem Hình 4, với các phương trình sau (20) và (21):

$$d_s(i,j) = \sqrt{\left(x_{ip}(i,j) - x_r\right)^2 + \left(y_{ip}(i,j) - y_r\right)^2} \quad (20)$$

$$\theta_s(i,j) = \left(\frac{y_{ip}(i,j) - y_r}{x_{ip}(i,j) - x_r}\right) - \varphi_r \quad (21)$$

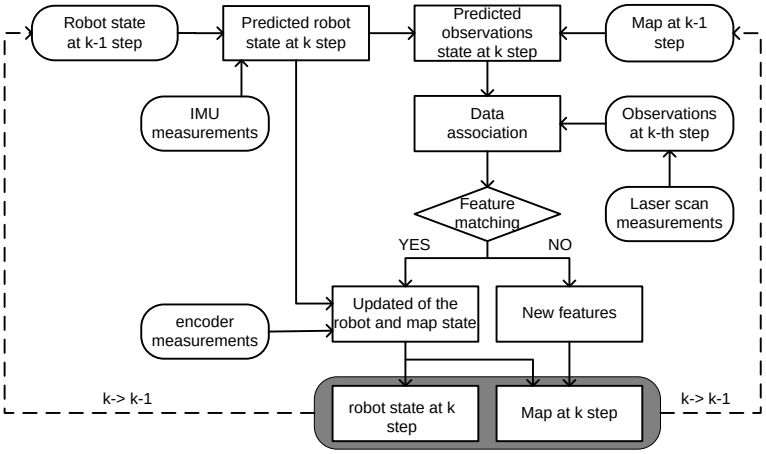
Mô hình nhiễu của LRF được định nghĩa bởi công thức:

$$d_{noise}(i) = d_s(i) + \frac{d_s(i)}{d_{max}(0,\sigma)} \quad (22)$$

Trong đó *N(0, σ)* biểu thị nhiễu phân phối chuẩn Gaussian với giá trị trung bình bằng 0 và phương sai σ. Nếu khoảng cách *d(i)* lớn, ảnh hưởng của nhiễu cũng lớn hơn theo tỷ lệ.

3. SLAM SỬ DỤNG BỘ LỌC KALMAN MỞ RỘNG (EKF-SLAM)

Sơ đồ khối của thuật toán EKF-SLAM được giới thiệu ở Hình 5. Khi cảm biến quán tính (IMU) đo sự thay đổi của UGV đang chuyển động, vị trí mới của UGV được dự đoán. Cùng lúc đó, các vật (landmarks) trong môi trường sẽ được quan sát và trích ra từ cảm biến laser tại vị trí mới của UGV. UGV sẽ kết hợp các landmarks mới này với các landmarks đã thấy trước đó, những landmarks nào chưa thấy ở trước đó sẽ được đưa thêm vào bộ EKF như là các đặc điểm (features) mới của môi trường. Những landmarks nào đã thấy trước đó sẽ được hiệu chỉnh lại cho đúng vị trí hơn. Các thủ tục này sẽ được thực hiện trong mục Data association. Các dữ liệu dự đoán như vị trí UGV, các landmarks cũ và mới được đo từ laser sẽ được kết hợp với dữ liệu đo từ encoder để hiệu chỉnh (hay cập nhật) lại vị trí của UGV, vị trí của các landmarks cũng như trạng thái bản đồ sau bước cập nhật của thuật toán [17].



Hình 5. Sơ đồ khối của thuật toán EKF-SLAM

Thuật toán 1 cho thấy quá trình thực hiện của thuật toán SLAM [18]. Đầu tiên, khởi tạo bản đồ được bắt đầu bằng cách dùng vị rí đầu tiên của UGV như là một tham chiếu cơ bản, với giả thiết là các thông số đã biết rõ ràng và chính xác, lúc đó xác suất

của các thông số xem như bằng 0. Điều này làm tăng tối đa độ chính xác của bản đồ và vị trí của UGV trong quá trình SLAM. Trong quá trình thực hiện, phần dự đoán vị trí chuyển động của UGV được dựa trên cảm biến IMU, phần cập nhật vị trí UGV và bản đồ được dựa trên các cảm biến encoder và laser. Quá trình được lặp lại cho đến lúc kết thúc.

Thuật toán 1: EKF-SLAM

$x_0 = 0; P_0 = 0$ {Map initialization}

$[z_0, R_0]$ = *get_measurements and* $[x_0, P_0] = add_new_features(x_0, P_0, z_0, R_0)$

for *k=1* to steps do

$[x_{Rk/k-1}, Q_k] = get_ahrs$

$[\hat{x}_{k/k-1}, P_{k/k-1}] = robot_motion(\hat{x}_{k-1}, P_{k-1}, x_{Rk/k-1}, Q_k)$ {EKF predict}

$[z_k, R_k] = get_measurements$ {including LRF and encoder measurements}

$$H_k = data_association(\hat{x}_{k/k-1}, P_{k/k-1}, z_k, R_k)$$

$[\hat{x}_k, P_k] = update_map(\hat{x}_{k/k-1}, P_{k/k-1}, z_k, R_k, H_k)$ {EKF update}

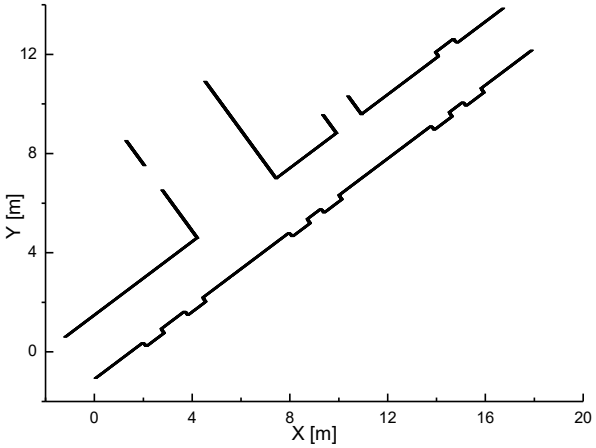
$[\hat{x}_k, P_k] = add_new_features(\hat{x}_k, P_k, z_k, R_k, H_k)$

end for

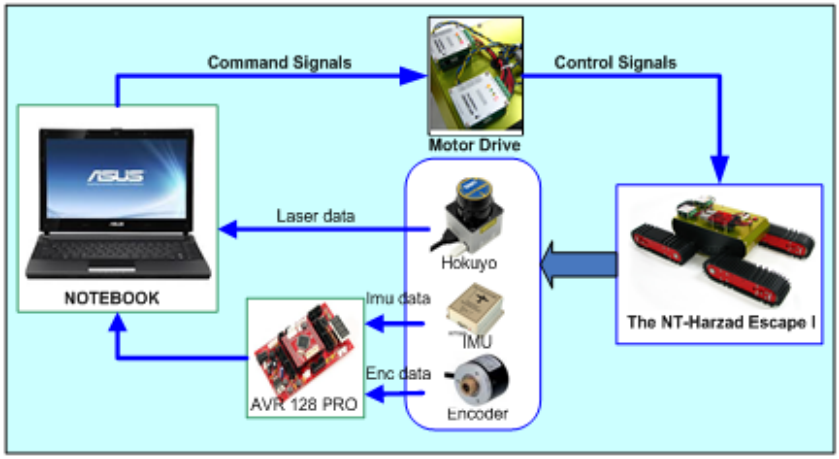
4. KẾT QUẢ VÀ THẢO LUẬN

Thực nghiệm được tiến hành trong hành lang tầng 3, tòa nhà 35 của trường Đại học Ulsan, Hàn Quốc, như Hình 5, với xe UGV hiệu NT-Harzard Escape I. Bản đồ tham chiếu RM (Reference Map) được dùng để đánh giá kết quả của thuật toán như Hình 6. Hình 7 giới thiệu về sơ đồ phần cứng của hệ thống được sử dụng trong thực nghiệm. Tốc độ trung bình của UGV khoảng 25cm/s, tốc độ quét dữ liệu của cảm biến Laser là 1 Hz; của encoder và IMU là 50 Hz. Có hai thí nghiệm được tiến hành.

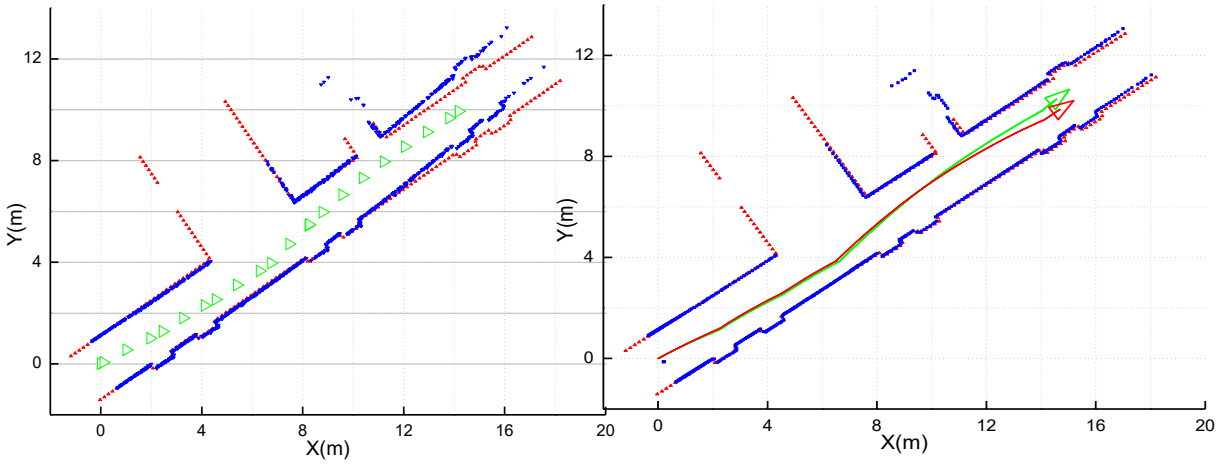
Hình 5. Môi trường hành lang



Hình 6. Bản đồ tham chiếu của hành lang

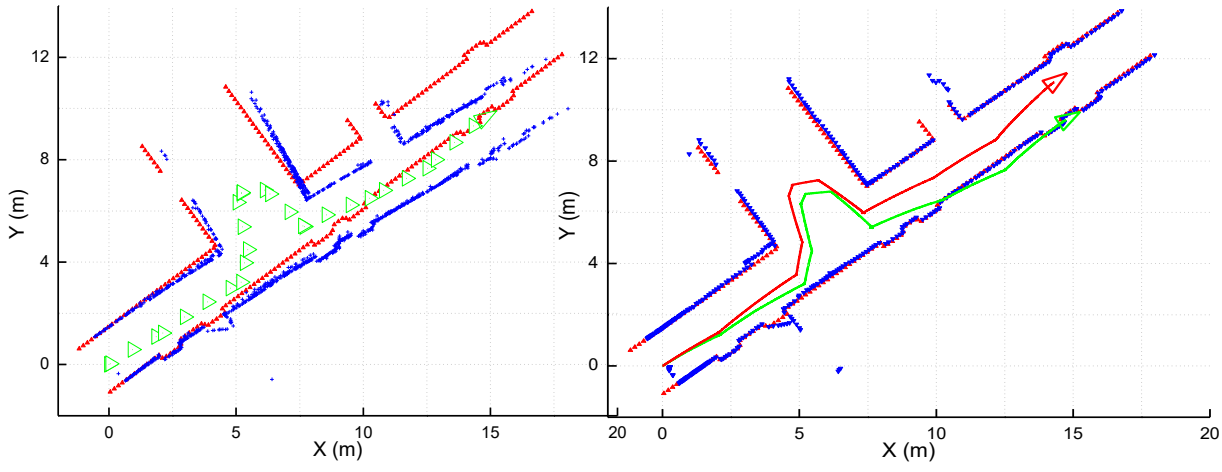


Hình 7. Sơ đồ phần cứng hệ thống



(a) (b)

Hình 8. Bản đồ và Vị trí của UGV trước (a) và sau (b) khi sử dụng EKF-SLAM thí nghiệm 1



(a) (b)

Hình 9. Bản đồ và vị trí của UGV trước (a) và sau (b) khi sử dùng EKF-SLAM thí nghiệm 2

Trong thí nghiệm 1, UGV di chuyển dọc theo hành lang chính mà không rẻ sang lối hành lang bên phải. Trước khi áp dụng EKF-SLAM, một bản đồ đo được (màu xanh lam) từ cảm biến laser, vị trí của UGV được ước lượng từ encoder và IMU (màu xanh lục) như Hình 8a. Qua so sánh bản đồ đo được với bản đồ tham chiếu (đồ thị màu đỏ), ta thấy được độ lệch khá lớn giữa hai bản đồ. Sau khi áp dụng EKF-SLAM, kết quả được biểu diễn trên Hình 8b. Bản đồ ước lượng (màu xanh lam) được định vị rất tốt và gần như chồng lấp lên bản đồ tham chiếu (màu đỏ). Cùng lúc đó quỹ đạo của UGV (đồ thị màu đỏ) cũng được ước lượng lại chính xác hơn và phù hợp với thực tế hơn.

Trong thí nghiệm 2, UGV di chuyển hướng về phía trước và có rẽ sang phía hành lang khác rồi quay trở về hành lanh ban đầu. Hình 9a hiển thị các kết quả trước khi sử dụng EKF-SLAM, ta thấy bản đồ đo được (xanh lam) từ cảm biến laser bị lỗi, cách xa bản đồ tham chiếu (RM). Lỗi này là do vị trí ước lượng của UGV bị sai lệch khá lớn (xanh lục). Điều này cho thấy vị trí UGV được tính toán bị sai lệch khá lớn, nguyên nhân là do sự trượt của bánh xe ảnh hưởng đến việc ước lượng vị trí của UGV từ các dữ liệu cảm biến encoder và IMU. Tuy nhiên, sau khi áp dụng EKF-SLAM, các vấn đề trên phần lớn được giải quyết. Bản đồ ước lượng (màu xanh lam) phần lớn khớp với bản đồ tham chiếu RM (màu đỏ) và quỹ đạo của UGV (đồ thị màu đỏ) được ước lượng lại sát với quan sát thực tế, như Hình 9b.

Bảng 1. Kết quả trước và sau khi sử dụng EKF-SLAM trong thí nghiệm 1 và 2 (TN 1 và 2)

Thuật toán		Trước khi áp dụng EKF-SLAM TN1	Sau khi áp dụng EKF-SLAM TN1	Trước khi áp dụng EKF-SLAM TN2	Sau khi áp dụng EKF-SLAM TN2
Vị trí UGV (m)	MD	1,7576	0,0921	1,7423	0,1270
	RMSD	1,1149	0,0707	1,1109	0,1022
Hướng góc UGV (rad)	MD	0,3822	0,0177	0,3733	0,0166
	RMSD	0,2995	0,0128	0,3189	0,0124
Vị trí vật cản (m)	MD		0,1191		0,1280
	RMSD		0,0964		0,0908

Dữ liệu thí nghiệm 1 trong Bảng 1 cho thấy Độ lệch tối đa (Maximum deviation - MD) của vị trí và hướng góc của UGV tương ứng là 0,0921m và 0,0177rad nhỏ hơn nhiều so với trước khi áp dụng thuật toán EKF-SLAM là 1,7576m và 0,3822rad. Tương tự đối với thông số Căn bậc hai độ lệch bình phương trung bình (Root mean square deviation - RMSD) trước khi dùng EKF-SLAM là 1,1149m và 0,2995rad, sau khi dùng EKF-SLAM là 0,0707m và 0,0128rad. Đồng thời, việc xây dựng bản đồ sau khi cập nhật bản đồ tốt hơn so với trước khi cập nhật EKF-SLAM, điều này cũng giải thích tính ưu

việt của EKF-SLAM. Việc đánh giá kết quả của thí nghiệm 2 cũng tương tự thí nghiệm 1, tuy nhiên kết quả trong thí nghiệm 2 có phần sai lệch lớn hơn vì quỹ đạo của UGV phức tạp hơn.

4. KẾT LUẬN

Bài báo giới thiệu thuật toán EKF-SLAM áp dụng cho các UGV. Kết quả cho thấy độ sai lệch vị trí và hướng góc của UGV giảm nhiều khi sử dụng thuật toán EKF-SLAM so với trước khi sử dụng. Bên cạnh đó bản đổ môi trường được lập với độ sai lệch nhỏ. Thuạt toán cho thấy sự hiệu quả cao của việc ứng dụng EKF-SLAM trong việc xác định vị trí của UGV và lập bản đồ môi trường không biết rõ thông qua thực nghiệm. Giải thuật góp phần cung cấp nền tảng lý thuyết cùng với thực nghiệm cơ bản để hợp nhất dữ liệu các cảm biến và quản lý các thông số môi trường bất định cho các nghiên cứu về lĩnh vực UGV và SLAM sau này. Tuy nhiên, giải thuật EKF-SLAM đòi hỏi khối lượng tính toán lớn dẫn đến mất nhiều thời gian cũng như bộ nhớ trong quá trình thực hiện. Để khắc phục nhược điểm này, các thuật toán SLAM với tốc độ nhanh như FastSLAM, Semantic SLAM, Multi-robot Collaborative SLAM … sẽ được nghiên cứu và thử nghiệm trong thời gian tới.

TÀI LIỆU THAM KHẢO

[1]. P. van Blyenburgh (1999). UAVs: An Overview, *Air and Space Europe*, Vol. 1, No. 5-6, pp. 43–47, 1999.

[2]. H. Durrant-Whyte and T. Bailey (2006). Simultaneous localization and mapping (slam): Part i the essential algorithms, *IEEE Robotics and Automation Magazine*, Vol. 13, No. 2, pp. 99 - 110.

[3]. R. Smith and P. Cheeseman (1986). On the representation and estimation of spatial uncertainty, *International Journal of Robotics Research*, Vol. 5, pp. 56-68.

[4]. H.D. Whyte (1988). On uncertain geometry in robotics, *IEEE Journal of Robotics and Automation*, Vol. 4, No. 1, pp. 23 - 31.

[5]. A. Davison (2003). Real-time simultaneous localization and mapping with a single camera, *Proceedings of IEEE International Conference on Computer Vision*, pp. 1403–1410.

[6]. A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse (2007). Monoslam: Real-time single camera slam, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 9, No. 6, pp. 1052 - 1067.

[7]. J. Sol´ a, M. Devy, A. Monin, and T. Lemaire (2005). Undelayed initialization in bearing only slam, *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pp. 2499 – 2504.

[8]. P. Elinas, R. Sim, and J. J. Little (2006). SigmaSLAM: Stereo vision slam using the Rao-BlackWellised particle filter and a novel mixture proposal distribution, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1564 - 1570.

[9]. T. K. Marks, A. Howard, M. Bajracharya, G. W. Cottrell, and L. Matthies (2007). Gamma-SLAM: Stereo visual SLAM in unstructured environments using variance grid maps, *Proceedings of IEEE International Conference on Robotics and Automation*.

[10]. G. Dissanayake, P Newman, and S Clark, et al (2001). A Solution to the Simultaneous Localization and Map Building (SLAM) problem, *IEEE Transactions on Robotics and Automation*, Vol. 17, No. 3, pp. 229-241.

[11]. A.1. Davision and D.W. Murray (2002). Simultaneous localization and map-building using active vision, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 7, pp. 865-880.

[12]. L. Paz, J. Tard´ os, and J. Neira (2008). Divide and conquer: EKF SLAM in o(n), *IEEE Transaction on Robotic*, Vol. 24, No. 5, pp. 1107 - 1120.

[13]. J.D. Tard´os, J. Neira, P.M. Newman, and J.J. Leonard (2002). Robust mapping and localization in indoor environments using sonar data, *International journal robotics research*, Vol. 21, No. 4, pp. 311–330.

[14]. G. Welch and G. Bishop (1995). *An introduction to the Kalman filter, Technical report*, Chapel Hill, NC, USA.

[15]. J. Yi, H. Wang, J. Zhang, D. Song, S. Jayasuriya and J. Liu (2009). Kinematic Modeling and Analysis of Skid_Steered Mobile Robots With Applications to Low cost IMU based Motion Estimation, *IEEE Transaction on Robotics*, Vol. 25, No. 5, pp. 1087-1097.

[16]. T. Teslic, G. Klancar and I. Skrjanc (2008). Kalman-filtering-based localization of mobile robot with a LRF in a simulated 2D environment, *IEEE Mediterranean Electrotechnical Conference*, pp.316-322.

[17]. Joan Sola (2013). *Simultaneous localization and mapping with the extended Kalman filter*, January 17.

[18]. J.A. Castellanos, J. Neira and J.D. Tardós (2005). *Map Building and SLAM algorithms*, Frankl: "dk6033_c009", pp. 335-371.

USING THE SIMULTANEOUS LOCALIZATION AND MAPPING ALGORITHM FOR UNMANNED GROUND VEHICLE

Ha Xuan Vinh*, Nguyen Van An, Ho Dac Quy

[1] Faculty of Electronics, Electrical Engineering and Material Technology, University of Sciences, Hue University

*Email: xuanvinhha@gmail.com

ABSTRACT

In the development of robotics, Unmanned Ground Vehicles have become essential for tasks such as exploration, discovery, support, environmental monitoring, and other work in unknown environments. These tasks can be performed individually or collaboratively with other devices, in groups with varying levels of performance, and by individuals with a high ability to plan and make decisions autonomously. For intelligent uncrewed vehicles, they must accurately locate and map themselves in unknown environments. Therefore, this article discusses the application of the Simultaneous Localization and Mapping algorithm, which is considered a fundamental and necessary feature for autonomous navigation of uncrewed vehicles.

Keywords: Unmanned Ground Vehicle (UGV), Extended Kalman Filter (EKF), Simultaneous Localization and Mapping (SLAM).