

# Định vị robot di động trong nhà dựa trên tín hiệu WiFi

Đỗ Hải Sơn, Trần Đức Mạnh, Trần Thị Thúy Quỳnh

Faculty of Electronics and Telecommunications, VNU University of Engineering and Technology, Hanoi, Vietnam

Email: dohaison1998@gmail.com, quynhttt@vnu.edu.vn

**Tóm tắt nội dung**—Định vị là bài toán quan trọng đối với robot di động. Bài báo trình bày một số kết quả nghiên cứu về việc định vị robot di động trong nhà dựa trên tín hiệu WiFi. Phương pháp định vị sử dụng nhận dạng dấu hiệu cường độ tín hiệu WiFi so với cơ sở dữ liệu ban đầu (Finger Printing). Với không gian định vị là hành lang thẳng dài 20m, hai thuật toán cực tiểu hóa khoảng cách Euclide và học máy SVM với độ chính xác tương ứng đạt được là 6,33cm và 13,21cm với tỷ lệ thành công 27,5%.

**Từ khóa chính**—Định vị trong nhà, robot di động, khoảng cách Euclide, SVM.

## I. GIỚI THIỆU

Robot di động là một kỹ thuật quan trọng trong sự phát triển của cách mạng công nghiệp 4.0. Trong đó, thách thức hàng đầu của robot di động là khả năng xác định vị trí. Hệ thống định vị được dùng rộng rãi nhất là hệ thống GPS (Global Positioning System), nhưng hệ thống này không thể sử dụng được cho các ứng dụng trong nhà do tín hiệu bị suy yếu và không có tia nhìn thẳng [1]. Hầu hết các hệ thống định vị trong nhà IPS (Indoor Positioning System) ngày nay dựa trên tín hiệu RF (Radio Frequency), ánh sáng, và sóng siêu âm [2]. Mỗi loại tín hiệu có những điểm mạnh và điểm yếu nhất định. Đối với robot di động, hệ thống định vị hiện đại nhất hiện nay dựa trên việc nhận dạng các dấu hiệu (đồ vật, ký hiệu,...) và đo khoảng cách tới dấu hiệu đó như LiDAR (Light Detection and Ranging) hoặc hệ thống dựa trên camera với độ chính xác vài cm trên khoảng cách di chuyển vài chục m [2]. Tuy nhiên, các kỹ thuật này có độ phức tạp tính toán lớn và độ nhạy với ánh sáng và các loại dấu hiệu khác nhau cao [2].

Trong khi đó, WiFi là một hệ thống WLAN được phủ sóng rộng rãi tại các tòa nhà (cả công sở và nhà dân). Việc phát triển hệ thống IPS dựa trên tín hiệu WiFi nhận được sự quan tâm của nhiều nhà nghiên cứu với ưu điểm không cần xây dựng hệ thống hạ tầng, các thiết bị cầm tay hầu hết đều tích hợp bộ thu WiFi. Các kỹ thuật định vị dựa trên tín hiệu WiFi gồm: xác định khoảng cách giữa máy thu và điểm truy cập, xác định góc hoặc thời gian tới của tín hiệu, và Finger Printing. Trong ba phương pháp trên, nhóm nghiên cứu lựa chọn phương pháp sau cùng do không đòi hỏi tín hiệu có tia nhìn thẳng, cũng như không cần thay đổi hạ tầng sẵn có của tòa nhà (việc sử dụng kỹ thuật xác định hướng/thời gian sóng tới chỉ thực hiện được với các modem thế hệ mới).

Finger Printing là phương pháp xác định vị trí dựa trên việc so sánh dấu hiệu hiện tại với dấu hiệu có sẵn trong cơ sở dữ liệu. Vì vậy, có hai vấn đề cần quan tâm nghiên cứu đối với

phương pháp này là: 1) Xây dựng bản đồ vô tuyến tốn rất nhiều công sức và cần cập nhật liên tục do môi trường vô tuyến biến động rất lớn; 2) Phương pháp nhận dạng dấu hiệu phải thực hiện được ở thời gian thực.

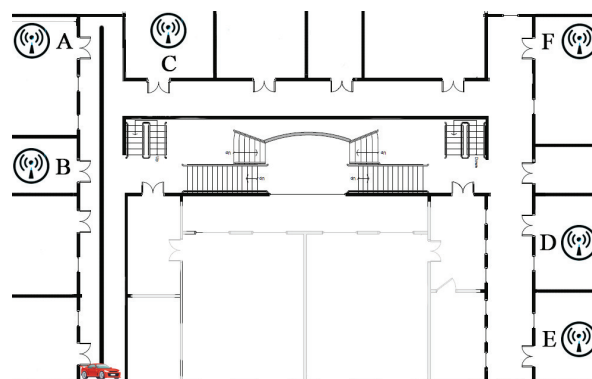
Nghiên cứu này thực hiện việc đánh giá khả năng áp dụng IPS sử dụng phương pháp Finger Printing dựa trên tín hiệu WiFi vào robot di động. Thực nghiệm được tiến hành trên một hành lang dài 20m với thuật toán cực tiểu hóa khoảng cách Euclide truyền thống và học máy SVM hiện đại để nhận dạng vị trí căn xác định so với cơ sở dữ liệu ban đầu.

Cụ thể các nội dung nghiên cứu gồm: phần 2 giới thiệu về việc thu thập dữ liệu để xây dựng bản đồ vô tuyến, phần 3 trình bày về các kỹ thuật nhận dạng vị trí, phần 4 đưa ra các kết quả thực nghiệm và thảo luận, và cuối cùng là phần kết luận.

## II. XÂY DỰNG BẢN ĐỒ VÔ TUYẾN

### 2.1 Mô tả thực nghiệm

Trong robot di động, bên cạnh việc xác định vị trí, vấn đề xây dựng bản đồ đồng thời cũng luôn được đặt ra, hệ thống này được gọi là SLAM (Simultaneous Localization and Mapping). Đây là vấn đề phức tạp do robot vừa phải định vị, vừa phải thu thập bản đồ. Tổng quan về các hệ thống SLAM được chi tiết trong [3]. Tuy nhiên, để đơn giản, bản đồ vô tuyến trong thực nghiệm của bài báo được xây dựng dựa trên việc biết trước quỹ đạo di chuyển trên một đường thẳng dọc hành lang bên trái như trên hình 1.



Hình 1. Hành lang xây dựng bản đồ vô tuyến

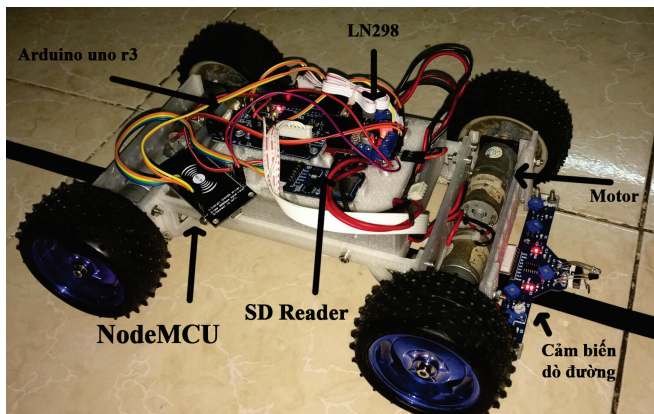
Ngoài ra, cũng trên hình 1, chúng ta có thể quan sát được các điểm truy cập WiFi được đánh dấu từ A đến Z, trong đó,

các điểm A, B, C gần quỹ đạo di chuyển hơn các điểm D, E, F.

Robot di động do nhóm nghiên cứu xây dựng được biểu diễn trên hình 2. Robot sử dụng được thiết kế đơn giản với 4 motor điện một chiều, L298N để điều khiển motor. Module NODEMCU 1.0 chipset ESP8266EX dùng để đo RSSI, gửi nhận dữ liệu lên server, đồng thời tích hợp vi xử lý giúp xử lý số liệu, cuối cùng là cấp xung điều khiển L298N.

Ngoài các bộ phận cơ bản như 04 motor mô tơ một chiều, 01 mô đun điều khiển tốc độ động cơ LN298, 01 cảm biến dò đường và 01 mô đun xử lý tín hiệu Arduino uno R3 để điều khiển việc di chuyển của robot dọc hành lang (đường màu đen), bộ phận quan trọng còn lại là NodeMCU 1.0 chipset ESP8266EX chứa bộ phận thu tín hiệu WiFi hoạt động được trên băng tần 2.4 GHz hỗ trợ chuẩn 802.11 b/g/n và được lập trình để dàng qua Arduino IDE.

Trong khi robot di chuyển, dữ liệu được thu thập liên tục theo lưu đồ hình 3 và được lưu vào thẻ nhớ SD Reader hoặc trên server như trong bảng I. Chúng ta cũng nhận thấy tín hiệu RSSI thu được của các điểm truy cập (APs - Access Points) A, B, và C ở gần vùng dịch chuyển của robot lớn hơn các điểm D, E, F ở xa vùng dịch chuyển.

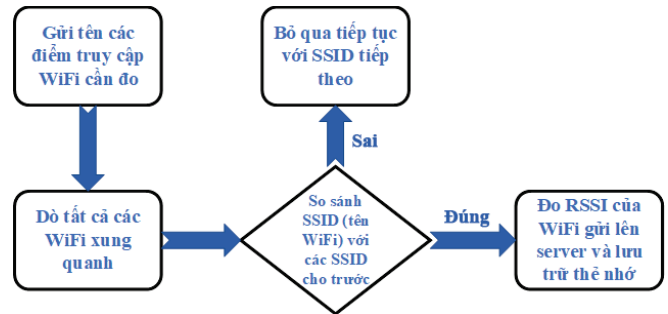


Hình 2. Robot di động thu thập cường độ tín hiệu WiFi

STT	Time	RSSI A	RSSI B	RSSI C	RSSI D	RSSI E	RSSI F
1	15:31:14	-73 dBm	-78 dBm	-64 dBm	-86 dBm	-89 dBm	-82 dBm
2	15:31:16	-72 dBm	-78 dBm	-64 dBm	-86 dBm	-89 dBm	-82 dBm
3	15:31:19	-67 dBm	-79 dBm	-63 dBm	-86 dBm	-89 dBm	-84 dBm
4	15:31:21	-66 dBm	-79 dBm	-62 dBm	-86 dBm	-89 dBm	-85 dBm
5	15:31:24	-66 dBm	-79 dBm	-60 dBm	-86 dBm	-89 dBm	-85 dBm
6	15:31:26	-76 dBm	-79 dBm	-67 dBm	-86 dBm	-89 dBm	-85 dBm
7	15:31:29	-72 dBm	-79 dBm	-61 dBm	-86 dBm	-89 dBm	-85 dBm
8	15:31:31	-67 dBm	-79 dBm	-59 dBm	-86 dBm	-89 dBm	-84 dBm
9	15:31:34	-63 dBm	-79 dBm	-64 dBm	-86 dBm	-89 dBm	-79 dBm
...	...	...	...	...	...	...	...

Bảng I  
BẢNG DỮ LIỆU RSSI

Với quãng đường 20m thời gian robot di chuyển để thu thập dữ liệu là 6 phút. Với tốc độ này, robot có thể thu thập được 120 mẫu trên quãng đường 20m do tốc độ đọc RSSI (Received Signal Strength Indicator) của NodeMCU rất chậm, khoảng 3 giây/mẫu.

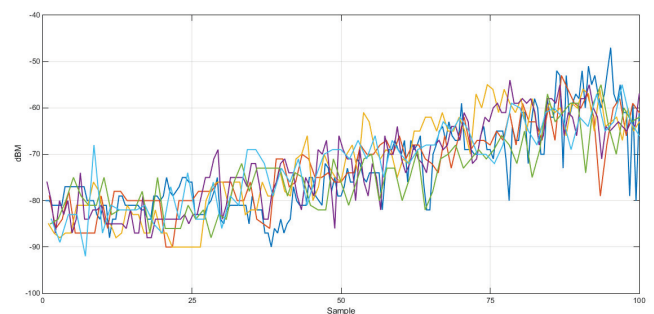


Hình 3. Lưu đồ thu thập dữ liệu RSSI

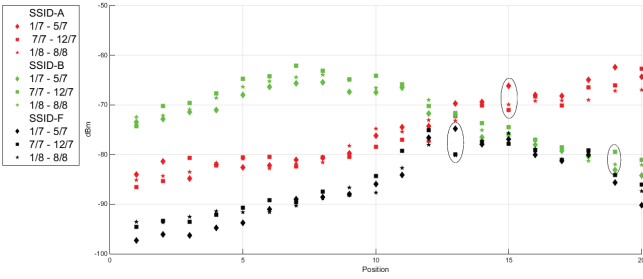
## 2.2 Xây dựng bản đồ vô tuyến

Bản đồ vô tuyến là cơ sở dữ liệu quan trọng trong phương pháp Figure Printing để robot di động có thể xác định vị trí. Việc định vị này càng chính xác khi cường độ RSSI có sai khác đáng kể giữa các điểm trên bản đồ, đồng thời các giá trị tại mỗi vị trí cần ổn định theo thời gian. Nhóm nghiên cứu đã thực hiện đo đạc dữ liệu để xây dựng bản đồ vô tuyến trong 5 ngày với dữ liệu thu được của điểm truy cập A được biểu diễn trên hình 4. Mặc dù thăng giáng ngẫu nhiên, dữ liệu của các ngày khá tập trung. Dữ liệu sau khi thu thập liên tục được tách thành 20 nhóm gần nhau tương ứng với 20 vị trí và được lấy trung bình theo ngày (trong 01 tuần).

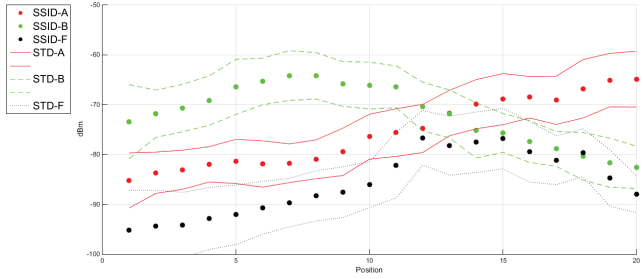
Để kiểm tra tính ổn định của dữ liệu tại mỗi vị trí theo thời gian, nhóm nghiên cứu thực hiện đo đạc và lấy trung bình thêm hai khoảng thời gian (2 tuần) nữa. Kết quả thu thập được biểu diễn trên hình 5 theo các tuần khác nhau. Chúng ta nhận thấy nhóm tín hiệu của mỗi điểm truy cập theo tuần khá gần nhau. Giá trị trung bình và độ lệch chuẩn của RSSI theo vị trí, ứng với các điểm truy cập A, B, và F được biểu diễn trên hình 6. Từ đây, có thể thấy rằng: tại cùng một vị trí, trung bình RSSI của các điểm truy cập khá khác nhau trong khi độ lệch chuẩn thì không khác nhau nhiều. Đây là lý do **trung bình của RSSI sẽ được sử dụng trong bản đồ vô tuyến** để dùng cho mục đích nhận dạng vị trí. Ngoài ra, những điểm truy cập cho kết quả không khác nhau nhiều giữa các điểm cũng không được sử dụng trong nghiên cứu này. Bản đồ vô tuyến hoàn chỉnh với 6 điểm truy cập được biểu diễn trên hình 7 và biểu diễn 3D của một số điểm truy cập như trên hình 8.



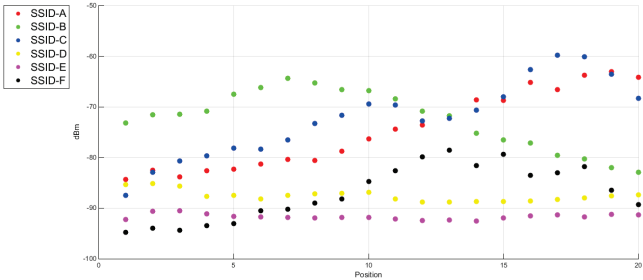
Hình 4. RSSI của một điểm truy cập trong 5 ngày đo



Hình 5. Giá trị trung bình của RSSI theo tuần



Hình 6. Giá trị trung bình và độ lệch chuẩn của RSSI trong hình 5



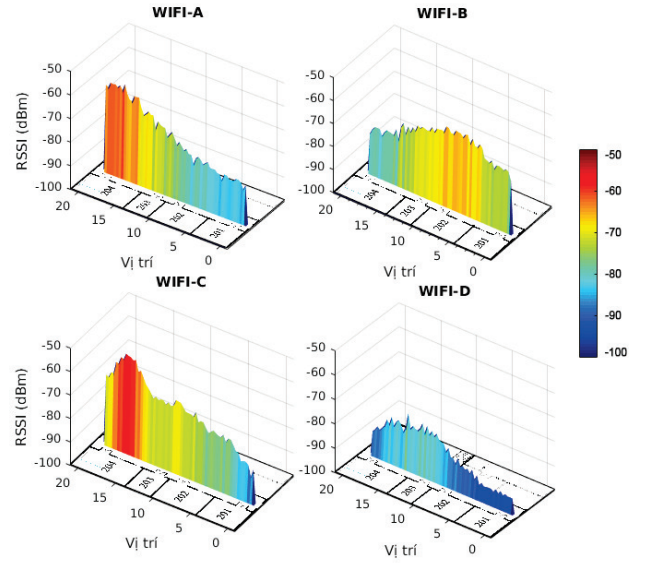
Hình 7. Bản đồ vô tuyến của 6 APs

### III. THUẬT TOÁN XÁC ĐỊNH VỊ TRÍ ROBOT DỰA TRÊN TÍN HIỆU WIFI

Phần này của bài báo thực hiện so sánh khả năng nhận dạng tín hiệu WiFi tại mỗi vị trí của robot. Hai phương pháp nhận dạng đặc trưng cho phương pháp nhận dạng truyền thống và hiện đại thường được dùng phổ biến là: phương pháp cực tiểu hóa khoảng cách Euclide [4] và phương pháp học máy SVM (Support Vector Machine). Trên thực tế, độ chính xác của phương pháp xác định vị trí dựa trên dấu hiệu nhận dạng dựa trên hai yếu tố: 1) Dấu hiệu khác biệt giữa các vị trí khác nhau; 2) Bản đồ vô tuyến ổn định theo thời gian. Tuy nhiên, như đã phân tích ở phần II, tín hiệu WiFi là tín hiệu ngẫu nhiên, chịu ảnh hưởng nhiều của môi trường xung quanh, vì vậy cần tìm ra đặc trưng không (ít) thay đổi theo thời gian của tín hiệu này.

#### 3.1 Phương pháp cực tiểu hóa khoảng cách Euclide

Phương pháp này dựa trên bản đồ vô tuyến có RSSI trung bình khá ổn định theo thời gian. Ngoài ra, để tăng sự khác



Hình 8. Bản đồ vô tuyến 3D

biệt RSSI giữa các vị trí, 6 APs được sử dụng. Công thức để xác định vị trí dựa trên phương pháp cực tiểu hóa khoảng cách Euclide là [4]:

$$\hat{x} = \operatorname{argmin}_{x_i} \sum_{i=1}^6 (r_i - d(x_i))^2 \quad (1)$$

với  $\hat{x}$ ,  $r_i$ , và  $d(x_i)$  lần lượt là vị trí cần ước lượng, tín hiệu RSSI trung bình nhận được, và tín hiệu RSSI trung bình trong cơ sở dữ liệu.

#### 3.2. Phương pháp học máy SVM

Phương pháp học máy SVM (Support Vector Machine) [5] là một trong những thuật toán phân lớp phổ biến. Phương pháp này gồm 2 bước:

- Bước 1: Xây dựng mô hình đặc trưng của tập dữ liệu dựa trên việc tập huấn tập dữ liệu được dán nhãn sẵn.
- Bước 2: Xác định vị trí dựa trên nét tương đồng của dữ liệu kiểm thử với mô hình dữ liệu.

Để phân loại nhiều vị trí, bài báo áp dụng thuật toán học máy nhiều lớp Multi-Class SVM [6]. Thuật toán này kết hợp SVM và mạng Neural để cho bộ phân lớp tốt hơn.

Khác với phương pháp trong phần 3.1 (mỗi vị trí được đặc trưng bởi giá trị RSSI lấy trung bình từ 5 mẫu), SVM sử dụng toàn bộ 5 mẫu của vị trí làm đặc trưng cho vị trí đó.

### IV. KẾT QUẢ THỰC NGHIỆM VÀ THẢO LUẬN

Phần này trình bày kết quả thực nghiệm hệ thống với ứng dụng Android để theo dõi/đặt vị trí đích của robot và lưu đồ

thuật toán điều khiển robot biểu diễn tương ứng trên hình 9 và hình 10.

Để đơn giản, robot được giới hạn chạy trên một đường thẳng cho trước (vạch màu đen) dọc hành lang 20m và cần dừng lại tại điểm được cắm cờ. Hai phương pháp định vị mô tả trong phần 3.1 và 3.2 được áp dụng, cũng như cờ đích được thay đổi vị trí để phân tích hiệu năng của hệ thống.

Dữ liệu RSSI của các APs được thu thập trong vòng 01 tháng tại các ngày khác nhau, mỗi ngày thu thập 20 lần, mỗi lần đo khoảng 120 mẫu. Dữ liệu này được sử dụng để xây dựng bản đồ vô tuyến sử dụng trong thuật toán cực tiểu khoảng cách Euclide và tập huấn để xác định mô hình trong thuật toán SVM.

Với phương pháp cực tiểu hóa khoảng cách Euclide: robot liên tục thu thập RSSI, tính giá trị trung bình của 5 mẫu và so sánh với dữ liệu trên bản đồ vô tuyến theo công thức (1). Trong thực nghiệm, do dữ liệu bản đồ vô tuyến nhỏ nên không cần sử dụng máy chủ để lưu trữ và xử lý mà chỉ với một module tích hợp chip xử lý là đủ để thực hiện cả việc định vị và điều khiển xe theo đích đến cho trước.

Với phương pháp SVM: được thực hiện trên Visual Studio Code, bằng ngôn ngữ Python. Dữ liệu trong pha xây dựng cơ sở dữ liệu được đưa vào huấn luyện trên máy tính để thu được mô hình dữ liệu. Trong pha xác định vị trí, máy tính cập nhật dữ liệu RSSI hiện tại từ máy chủ (do NodeMCU gửi lên) và đưa qua mô hình dữ liệu để xác định vị trí, sau đó gửi vị trí này lên máy chủ, robot sẽ lấy thông tin vị trí từ máy chủ để quyết định là vị trí đích hay không (robot sẽ dừng lại nếu đúng là đích cần đến). Máy chủ lưu trữ dữ liệu được chọn là Firebase của Google, là loại dễ sử dụng và thuận tiện đối với nhiều ngôn ngữ và nền tảng khác nhau như: Java, Python, C#,... ứng dụng cho Android, iOS, Windows, Web.

Ngoài ra, robot cũng được lập trình dừng lại khi đi đến cuối hành lang (trường hợp robot không xác định được đích cần đến).

#### 4.1. Kết quả thực nghiệm

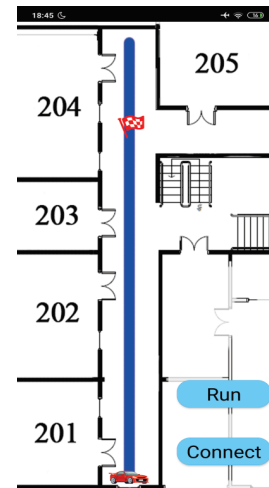
Kết quả thực nghiệm được biểu diễn trên bảng II được đánh giá bởi hai tham số:

- Số lần robot không xác định được vị trí (đi thẳng - không dừng).
- Sai số trung bình trong trường hợp robot xác định được vị trí (robot dừng).

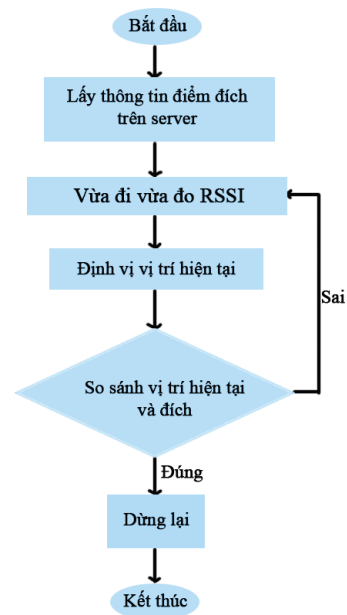
#### 4.2. Thảo luận

Từ số liệu biểu diễn trên bảng II và bảng III ta có một số nhận xét sau:

- Đối với phương pháp Euclide, vị trí 1, 3, và 14 là vị trí



Hình 9. Giao diện ứng dụng



Hình 10. Lưu đồ điều khiển xe robot

đặc biệt có số lần robot không xác định được vị trí lớn nhất (7 lần) trong khi phương pháp SVM là 3 vị trí 3, 4, và 14 (10 lần). Điều này có thể do việc đo kiểm cơ sở dữ liệu chưa chính xác tại các điểm này.

- Số lần không xác định được vị trí của robot khi sử dụng thuật toán SVM nhiều hơn hẳn so với khi sử dụng thuật toán cực tiểu khoảng cách Euclide (38,5% so với 71%). Điều này có thể do dữ liệu huấn luyện chưa đủ.
- Xét trong những lần robot xác định được vị trí, sai số trung bình của cả hai phương pháp đều nhỏ hơn 0,5m.



Vị trí	Số lần không xác định được vị trí		Sai số với trung tâm ô (cm)	
	Euclide	SVM	Euclide	SVM
1	7	5	50.00	42.00
2	3	4	37.14	48.33
3	7	10	56.67	–
4	3	10	47.14	–
5	2	7	37.50	56.67
6	2	4	47.50	30.00
7	5	9	50.00	–
8	1	4	28.89	41.67
9	–	3	35.00	71.42
10	2	4	38.75	50.00
11	2	6	53.75	30.00
12	2	5	28.75	86.00
13	2	9	32.50	50.00
14	7	10	70.00	–
15	–	6	53.00	50.00
16	4	7	26.67	50.00
17	1	5	47.78	58.00
18	5	3	96.00	37.14
19	2	7	37.50	36.67
20	1	5	48.89	72.00

Bảng II  
KẾT QUẢ THỰC NGHIỆM (SAI SỐ TÍNH TRUNG BÌNH THEO SỐ LẦN XÁC ĐỊNH ĐƯỢC VỊ TRÍ)

	Tỷ lệ thành công		Độ lệch trung bình	
	1x1	2x2	1x1	2x2
Euclide	27.5 %	70 %	6.33 cm	46.79 cm
SVM	21 %	38.5 %	13.21 cm	39.78 cm

Bảng III  
KẾT QUẢ THỰC NGHIỆM (TÍNH TRUNG BÌNH THEO CÁC VỊ TRÍ)

Như vậy, phương pháp định vị robot di động dựa trên tín hiệu WiFi khá khả thi đối với thuật toán cực tiểu hóa khoảng cách Euclide.

## V. KẾT LUẬN

Bài báo này trình bày các nghiên cứu về khả năng sử dụng tín hiệu WiFi để xác định vị trí cho robot di động đơn giản (robot chỉ có phần thu phát WiFi và điều khiển chuyển động). Khác với một số công bố khác, nghiên cứu này thực hiện việc thu thập dữ liệu liên tục trong suốt quá trình robot di chuyển để xây dựng cơ sở dữ liệu cũng như xác định vị trí. Điều này tạo ra khó khăn trong việc tăng độ chính xác của hệ thống nhưng lại là vấn đề thực tiễn cần phải thực hiện. Kết quả nghiên cứu chỉ ra rằng, mặc dù là kỹ thuật hiện đại nhưng SVM cho kết quả định vị không tốt bằng phương pháp truyền thống dựa trên cực tiểu hóa khoảng cách Euclide (khả năng xác định được vị trí của robot đối với phương pháp SVM là 38,5% trong khi phương pháp còn lại là 71%). Với phương pháp Euclide, sai số trong những lần robot xác định được vị trí nhỏ hơn 0,5m.

## TÀI LIỆU

- [1] D. A. Ali Khalajmehrabadi, Nikolaos Gatsis, "Modern WLAN Fingerprinting Indoor Positioning Methods and Deployment Challenges," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1974–2002, 2017.
- [2] D. M. Payam Nazemzadeh, Daniele Fontanelli and L. Palopoli, "Indoor Localization of Mobile Robots through QR Code Detection and Dead Reckoning Data Fusion," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 6, pp. 2588–2599, 2017.
- [3] H. C. Cesar Cadena, Luca Carlone, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, Present, and Future of Simultaneous Localization And Mapping: Towards the Robust-Perception Age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.
- [4] P. Davidson and R. Pich, "A Survey of Selected Indoor Positioning Methods for Smartphones," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 2, pp. 1347–1370, 2017.
- [5] C. M. Bishop, *Pattern recognition and machine learning*. Springer, 2006.
- [6] "CS231n Convolutional Neural Networks for Visual Recognition."