

## ỨNG DỤNG XỬ LÝ ẢNH KỸ THUẬT SỐ TRONG VIỆC THEO DÕI SỰ CHUYỂN ĐỘNG CỦA CÁC ROBOT DI ĐỘNG

Phạm Trần Lam Hải<sup>1</sup>, Trần Thanh Hùng<sup>1</sup>, Nguyễn Văn Khanh<sup>1</sup> và Lưu Trọng Hiếu<sup>1</sup>

<sup>1</sup> Khoa Công nghệ, Trường Đại học Cần Thơ

### Thông tin chung:

Ngày nhận: 22/03/2015

Ngày chấp nhận: 17/08/2015

### Title:

Applying digital image processing to track mobile robots

### Từ khóa:

Phép toán hình thái học, phân đoạn ảnh, robot bầy đàn, tính toán vị trí và vận tốc, hướng đứng và hướng di chuyển

### Keywords:

Morphological operation, image segmentation, swarm robots, position and velocity calculation, heading direction and movement direction

### ABSTRACT

The original aim of this article is to develop MATLAB program that can be used to connect with a wireless camera to observe the movement of each robot agent in real-time. However, it is not possible to completely simulate the behaviour of mobile robots in real-time due to the connection speed limitation of the wireless camera. Therefore, this research methodology is based on offline processing for three mobile robots represented by three circular objects, each of which has one of the three secondary colours (cyan, magenta, and yellow). These circular models simulate three mobile robots that are used for the surveillance purpose. Image segmentation and morphological operations were performed in the program for identifying each mobile robot and for calculating its position and velocity. Trigonometric was also applied to identify the heading direction and movement direction of each robot. Further research in mobile robot observation and management can be carried out in real-time when the limitation of wireless cameras in connection speed is overcome.

### TÓM TẮT

Mục tiêu chủ yếu của bài báo là phát triển một chương trình MATLAB có thể kết nối với máy ảnh qua mạng không dây để quan sát từng con trong nhóm robot với quá trình xử lý thời gian thực. Tuy nhiên, không thể mô phỏng hành vi của robot di động theo thời gian thực do sự hạn chế của tốc độ kết nối không dây. Do đó, phương pháp nghiên cứu này chỉ dựa trên xử lý ngoại tuyến (offline) cho một nhóm gồm ba robot đại diện bởi ba vòng tròn với ba màu sắc khác nhau (xanh lơ, hồng sẫm và màu vàng). Các mô hình tròn này mô phỏng ba robot được sử dụng cho mục đích giám sát. Chương trình đã sử dụng phương pháp phân đoạn ảnh (image segmentation) cùng với phép toán hình thái học (morphology) để phân biệt, xác định vị trí và tính vận tốc của từng robot. Bên cạnh đó, chương trình cũng ứng dụng các hàm lượng giác nhằm biết được hướng đứng và hướng di chuyển của mỗi robot. Nghiên cứu sẽ phát triển chương trình này để có thể xử lý thời gian thực khi những hạn chế của máy ảnh kết nối không dây được cải thiện.

## 1 GIỚI THIỆU

Cùng với sự phát triển của kỹ thuật robot, các robot trở nên thông minh hơn và được ứng dụng

trong nhiều lĩnh vực khác nhau nhưng với chi phí cao. Vì thế một câu hỏi được đặt ra: một số lượng lớn các robot đơn giản với giá thành rẻ có thể thực

hiện công việc của những robot đất tiền không? Do đó, robot bày đàn là đối tượng nghiên cứu được nhiều nhà khoa học trên thế giới quan tâm. Sự thành công của dạng robot này sẽ là một câu trả lời tốt nhất cho câu hỏi trên. Các nhà nghiên cứu đã đề nghị nhiều giải pháp phần cứng lẫn giải thuật phần mềm để tạo ra các robot di động phục vụ cho nhiều ứng dụng khác nhau và hướng tới tạo ra các thế hệ robot bày đàn (Ogren, P., et al., 2004; Kerr, W., et al., 2005).

Một trong những giải pháp được các nhà nghiên cứu quan tâm là sử dụng máy ảnh treo để đọc các thông số chuyển động của robot thông qua các thuật toán xử lý ảnh chụp được. Theo Shirai, Y. & Inoue, H. (1973), các nhà nghiên cứu đã bắt đầu nghiên cứu những ưu điểm của việc sử dụng máy ảnh để quản lý các robot di động. Hiện nay, các nghiên cứu về quản lý robot di động bằng hình ảnh đang được phát triển mạnh và đạt được một số kết quả rất khả quan. Cụ thể là một số nhà nghiên cứu đã sử dụng các thuật toán trong việc xử lý thông tin về màu sắc để quản lý một nhóm các robot di động. Ví dụ: điều khiển robot trong môi trường văn phòng bằng máy ảnh treo phía trên (Kruse, E. và Wahl, F.M. 1998), điều khiển robot di động có chức năng nhặt bóng trong sân quần vợt (Webster, R.J. và Brannon, A.S. 2002).

Từ các kết quả nghiên cứu trên ta thấy mặc dù việc đưa máy ảnh treo vào robot di động đạt được nhiều kết quả nhưng chỉ dừng lại ở việc dựa vào màu sắc để phân loại robot. Để cải tiến yếu điểm này, bài báo của chúng tôi giới thiệu các thuật toán xử lý ảnh kết hợp với ứng dụng phương pháp hình học sử dụng phần mềm Matlab để không những có thể xác định được từng robot dựa vào màu sắc mà còn tính toán được vị trí, tốc độ và hướng di chuyển của chúng. Phần còn lại của bài báo trình bày chi tiết nội dung nghiên cứu bao gồm: mô hình thí nghiệm, phương pháp thực hiện, kết quả thực nghiệm trên Matlab, kết luận và định hướng nghiên cứu tiếp theo.

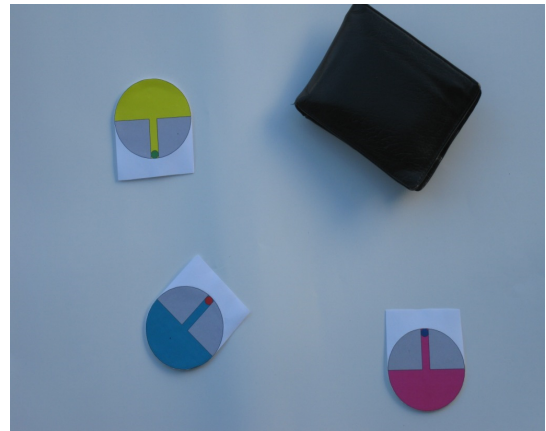
## 2 NỘI DUNG NGHIÊN CỨU

### 2.1 Mô hình thí nghiệm

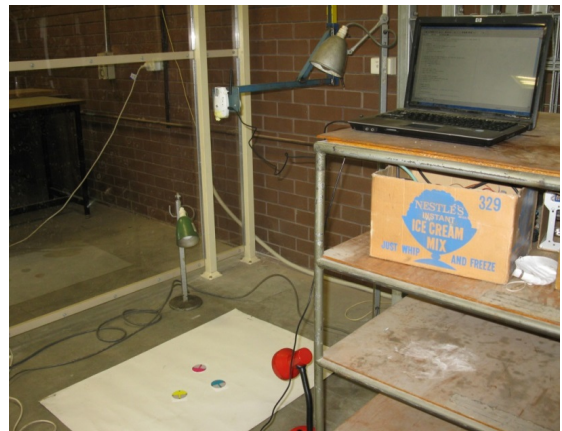
Giá treo máy ảnh được thiết kế để treo Wi-fi Camera Pro. Ba hình tròn được dùng để mô phỏng cho ba robot với đường kính 60 cm. Mỗi robot có các màu khác nhau là màu xanh lơ, hồng sẫm và màu vàng. Trên mỗi robot có hình tròn đường kính 6 cm với các màu tương ứng là đỏ, xanh dương và xanh lá cây được tượng trưng cho phần đầu của chúng (Hình 1). Do sự hạn chế của việc kết nối

webcam với Matlab bằng mạng không dây, nên webcam Wi-fi Camera Pro được sử dụng để kết nối với Matlab qua giao tiếp USB. Máy ảnh này được treo trên giá, các tấm ảnh được chụp để xử lý ngoại tuyến (offline) với chu kỳ là 0.5 giây.

Ba robot được đặt trong khu vực thử nghiệm với diện tích 50x120 cm và có thêm một số bóng đèn để cung cấp đầy đủ ánh sáng. Webcam Wi-fi Camera Pro được treo trên giá, kết nối với laptop HP qua cổng USB và phần mềm Matlab sẽ điều khiển để chụp và xử lý ảnh (Hình 2).



Hình 1: Ba robot và vật cản



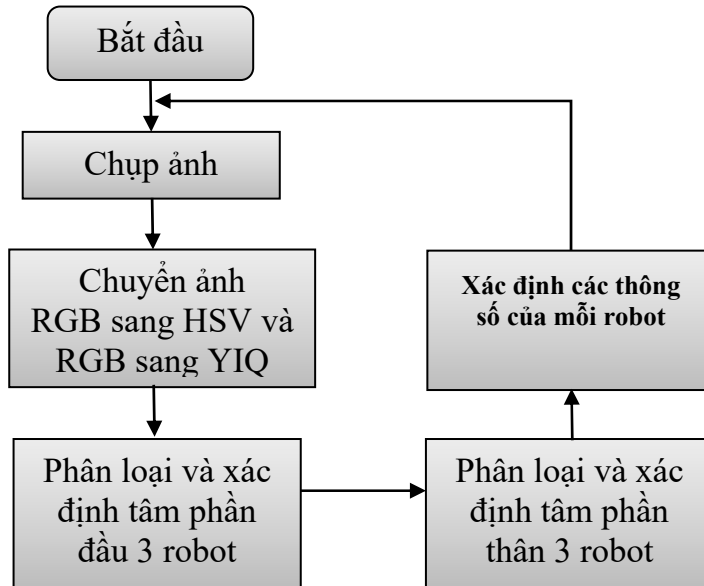
Hình 2: Khu vực thử nghiệm

### 2.2 Phương pháp thực hiện

Để xác định được vị trí, hướng đứng, vận tốc, đồng thời biết được hướng di chuyển của chúng, vấn đề đặt ra là cần nhận dạng các robot trên mỗi tấm ảnh được chụp với tốc độ 2 ảnh/giây. Hình 3 là lưu đồ tổng quát của chương trình. Dựa vào thuật toán này, các thông số của robot được xác định qua 4 bước xử lý. Đầu tiên, ảnh đầu vào được chuyển sang không gian màu HSV và YIQ để phù hợp với

mắt của con người trong việc cảm nhận màu sắc. Kế đến, các nhóm robot được phân loại dựa trên kỹ thuật phân đoạn ảnh sử dụng ngưỡng cường độ màu với phương pháp tìm và sửa sai. Mỗi robot sẽ được xác định bằng kỹ thuật xử lý hình thái. Tiếp theo, vị trí, hướng đứng và hướng di chuyển của từng robot sẽ được tính toán dựa vào tọa độ pixel

của chúng bằng phương pháp hình học đơn giản. Phần tiếp theo của mục này sẽ trình bày tóm lược lý thuyết được sử dụng để thực hiện nghiên cứu bao gồm xử lý ảnh số trong Matlab, các không gian màu cơ bản được áp dụng, phân đoạn ảnh, xử lý ảnh bằng phép toán hình thái học và hàm lượng giác ngược trong toán học.



Hình 3: Lưu đồ thuật toán của chương trình

2.2.1 Xử lý ảnh số trong Matlab

Matlab là chữ viết tắt của cụm từ “matrix laboratory”, được thiết kế để làm việc chủ yếu với các ma trận và mảng. Ảnh thật được mô tả bằng ảnh số, nó là tập hợp hữu hạn các điểm ảnh với mức xám phù hợp. Điểm ảnh là một phần tử của ảnh số tại tọa độ (x,y) với độ xám hoặc màu nhất định. Kích thước và khoảng cách giữa các điểm ảnh đó được chọn thích hợp sao cho mắt người cảm nhận sự liên tục về không gian và mức xám (hoặc màu) của ảnh số gần như ảnh thật. Mỗi phần tử trong ma trận được gọi là một phần tử ảnh. Các phương pháp được sử dụng để xử lý ảnh trong Matlab sẽ được đề cập trong phần này.

2.2.2 Các không gian màu cơ bản được áp dụng

a. Không gian màu RGB

RGB là không gian màu cơ bản nhất được dùng trong đồ họa máy tính và nhiều thiết bị kỹ thuật số khác. Nó là sự kết hợp của ba màu sắc cơ bản: màu đỏ (R=Red), xanh lục (G=Green) và xanh lơ (B=Blue) để mô tả tất cả các màu sắc khác. Tuy nhiên, không gian màu RGB có hạn chế lớn nhất là không phù hợp với cách con người cảm nhận

về màu sắc. Do vậy, không phù hợp cho việc tìm kiếm ảnh.

b. Không gian màu HSV

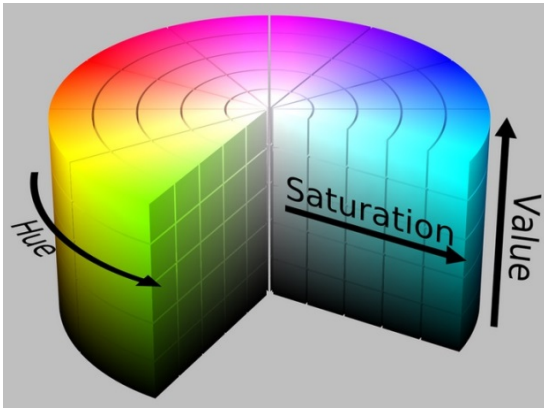
Không gian màu HSV, còn gọi là không gian màu HSB, là một không gian màu dựa trên ba số liệu:

H: (Hue) Vùng màu có giá trị từ  $0^0 - 360^0$

S: (Saturation) Độ bão hòa màu

V (hay B): (Value hay Bright) Độ sáng có giá trị từ 0 – 1

Không gian màu này thường được biểu diễn dưới dạng hình trụ hoặc hình nón và rất hữu ích để xử lý hình ảnh kỹ thuật số. Trong thực tế, HSV là sự thay đổi của không gian màu RGB, chúng mô tả các màu sắc vật lý có liên quan đến màu sắc ban đầu của màu đỏ, xanh lục và xanh lơ của thiết bị hoặc của không gian màu RGB cụ thể. Mô hình HSV giúp tách bạch màu (H, S) và độ sáng (V), phù hợp với cảm nhận của con người về màu sắc hơn so với không gian màu RGB. Hình 4 minh họa không gian màu HSV (Wikipedia, 2014).



**Hình 4: Không gian màu HSV**

*c. Không gian màu YIQ*

Mắt con người thực sự có độ phân giải màu khá thấp. Chính vì vậy, các ảnh màu có độ phân giải cao đều được xử lý bởi hệ thống hình ảnh (visual system), bằng cách kết hợp ảnh đen và trắng có độ phân giải cao và ảnh màu với độ phân giải thấp,

nhằm giúp con người nhìn thấy được. Lợi dụng điểm này, không gian màu YIQ làm giảm lượng thông tin trong phần màu (chrominance) một cách đáng kể, để cho mắt người tự kết hợp chúng lại. Do đó, chúng tôi sẽ dựa trên hai không gian màu HSV và YIQ để tìm ra ngưỡng tối ưu cho mỗi loại robot trong mô hình.

Ảnh gốc là một ảnh thực được biểu diễn bởi một ma trận ba chiều có kích thước 768x1024x3, với 768x1024 là kích thước ảnh theo pixels, biểu diễn ảnh này dưới dạng số uint8 không thích hợp để xử lý, cho nên dữ liệu được chuyển sang dạng double, mỗi thành phần màu có giá trị giữa 0 và 1, với cú pháp “I12=im2double(I1);”. Ngoài ra, ảnh RGB sẽ được chuyển sang ảnh double HSV và ảnh double YIQ cho việc phân loại và các bước xử lý khác. Để xem xét các không gian màu RGB, HSV và YIQ thuận tiện hơn, khối lệnh sau đây (Hình 5) sẽ tách ra các mặt phẳng màu (đỏ - R, xanh lá cây - G, xanh dương - B, vùng màu - Hue, độ bão hòa màu - Saturation, độ sáng - Value, ánh sáng - Y, vùng màu - I, độ bão hòa - Q) riêng biệt của các ảnh trên.

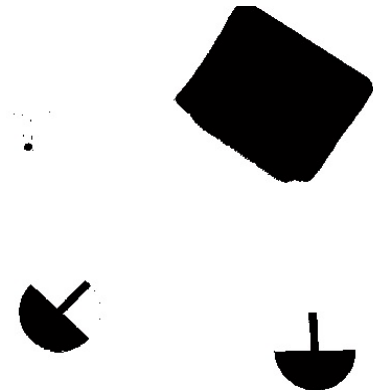
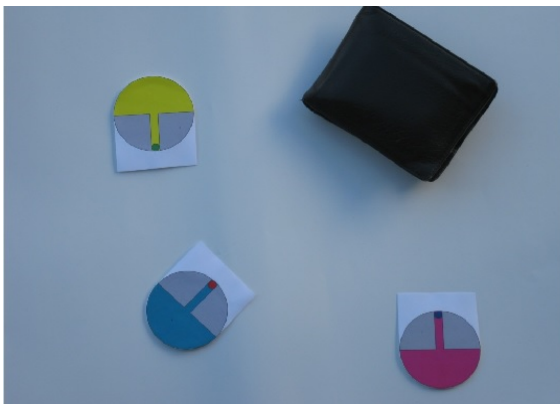
```
I12=im2double(I1);      I21=rgb2hsv(I1);      I22=rgb2ntsc(I1);
R = I12(:,:,1);        H = I21(:,:,1);      Y = I22(:,:,1);
G = I12(:,:,2);        S = I21(:,:,2);     I = I22(:,:,2);
B = I12(:,:,3);        V = I21(:,:,3);     Q = I22(:,:,3);
```

**Hình 5: Tách các mặt phẳng màu**

**2.2.3 Phân đoạn ảnh (image segmentation)**

Kỹ thuật phân đoạn ảnh là một công đoạn rất quan trọng trong các bài toán xử lý ảnh. Với mục đích là chia ảnh đầu vào thành các vùng đồng nhất. Do đó, mỗi vùng đồng nhất là tập các điểm ảnh có chung một số tính chất như màu sắc, kiểu tô. Phân ngưỡng (thresholding) được áp dụng trong bài báo là một trong những thuật giải phân đoạn ảnh hiện có. Chia ngưỡng cường độ và gán điểm ảnh vào

từng lớp khác nhau. Cách giải quyết của chúng tôi dựa trên phương pháp thử và sửa sai để chọn ra cường độ tối ưu trong tất cả các mặt phẳng đã được tách phía trên. Ưu điểm của phương pháp này là đơn giản và được dùng cho phân đoạn sơ bộ. Tuy nhiên, phương pháp này lại không thích hợp cho ảnh màu, ảnh xám, nhiễu. Hình 6 minh họa ảnh gốc và ảnh kết quả mong muốn sau khi phân đoạn theo ngưỡng tối ưu.



**Hình 6: Phân đoạn ảnh theo ngưỡng**

2.2.4 Xử lý ảnh bằng phép toán hình thái học (Morphology)

Các kỹ thuật xử lý ảnh số trên máy tính đang được rất nhiều nhà nghiên cứu quan tâm và phát triển. Phép toán hình thái học được phát triển nhờ sự kết hợp giữa lý thuyết, ứng dụng, phương pháp và các thuật toán (Pierre Soille, 2002). Các phép toán xử lý hình thái học được thực hiện chủ yếu trên ảnh nhị phân và ảnh xám. Bài báo sử dụng phép toán hình thái học trên ảnh nhị phân hay còn được gọi là ảnh đen trắng tương ứng với hai giá trị 0 (màu trắng) và 1 (màu đen). Hầu hết các phép toán hình thái học được định nghĩa từ hai phép toán cơ bản là phép co ảnh (Erosion) và giãn ảnh (Dilation). Yếu tố quan trọng trong các phép toán này là lựa chọn một phần tử cấu trúc có hình dáng phù hợp. Phần còn lại sẽ giới thiệu phép toán cơ bản được áp dụng trong bài báo là phép mở ảnh (Opening).

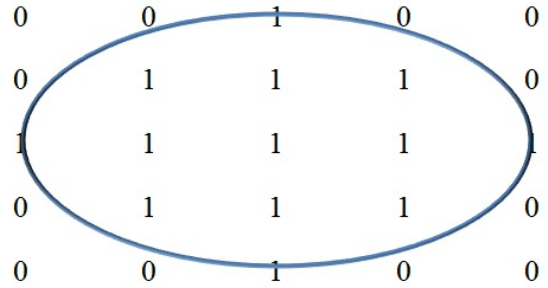
a. Phần tử cấu trúc

Phần tử cấu trúc là một ảnh nhị phân có kích thước nhỏ gồm có hai giá trị 0 và 1, các giá trị bằng 0 được bỏ qua trong quá trình tính toán, gọi S(i,j) là phần tử cấu trúc của ảnh nhị phân và được định nghĩa như sau (theo Wilhelm Burger và Mark J. Burge 2009):

$$S(i,j) \in \{0,1\} \tag{2.1}$$

Một số hình dáng của phần tử cấu trúc thường được sử dụng trên ảnh nhị phân: dạng đường theo chiều ngang và dọc, hình vuông, hình đĩa, hình ellipse (González, R.C. và Woods, R.E. 2008). Tuy

nhiên, bài báo của chúng tôi chỉ chọn hình dáng của phần tử cấu trúc là hình đĩa tròn với các bán kính khác nhau tùy theo từng trường hợp cụ thể. Hình bên dưới thể hiện phần tử cấu trúc hình đĩa với bán kính R=2:



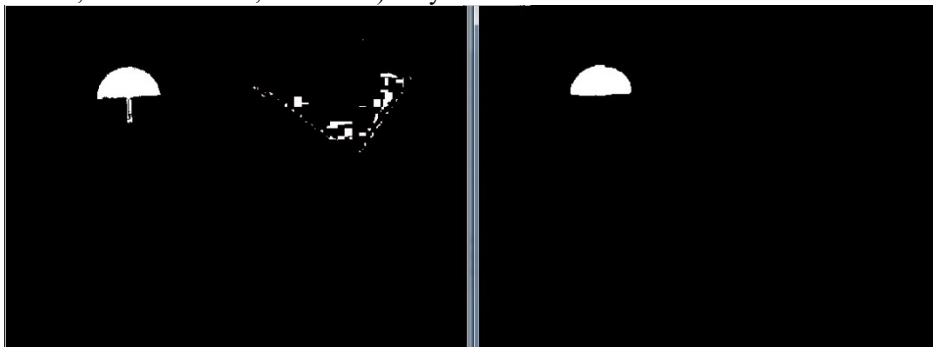
Hình 7: Hình đĩa tròn của phần tử cấu trúc phẳng

b. Phép mở ảnh (opening)

Phép mở ảnh là một trong những phép toán quan trọng nhất của phép xử lý hình thái học. Phép mở ảnh làm mịn đường biên của đối tượng trước và sau đó sẽ loại bỏ các phần lồi ra. Gọi A là hình ảnh gốc và B là phần tử cấu trúc, (⊖) là ký hiệu của phép mở ảnh giữa tập hợp A và phần tử cấu trúc B, phép mở ảnh được xác định bởi công thức (Lotufo, R.A., et al., 2008):

$$A \circ B = (A \ominus B) \oplus B \tag{2.2}$$

Phép mở ảnh được dùng để lấy phần tử mong muốn trong ảnh sau khi được phân đoạn theo ngưỡng (Hình 8).



Hình 8: Phép mở ảnh

2.2.5 Hàm lượng giác ngược trong toán học

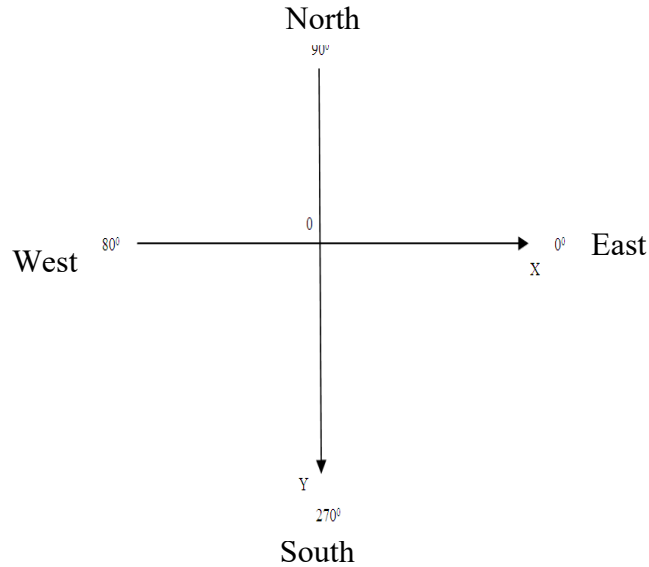
Sau khi xác định được tâm phần đầu và tâm phần thân trên mỗi robot, bài báo của chúng tôi đã áp dụng hàm lượng giác ngược để xác định hướng

đứng và hướng di chuyển của chúng. Hàm arctan sẽ trả về giá trị là góc dựa trên hai tâm điểm đã biết. Kết hợp với hệ tọa độ đã được định nghĩa (Hình 9) để biết được hướng đứng và hướng di chuyển của mỗi robot.

### 3 KẾT QUẢ THỰC NGHIỆM

Trong hệ thống những con robot bày đàn, một chương trình cần thiết được lập ra để giúp con người có thể quan sát được việc phân loại, vị trí, hướng đứng, tốc độ và hướng di chuyển của mỗi

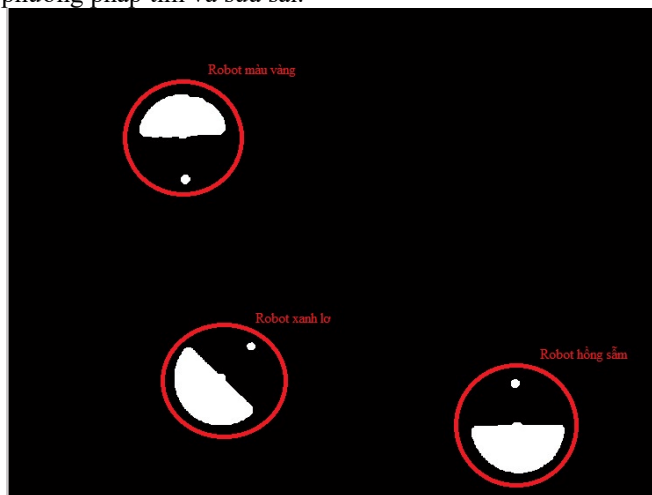
con robot. Từ đó, con người có thể kiểm soát được các robot thành viên trong việc tránh được vật cản và va chạm lẫn nhau. Matlab là một công cụ được chọn trong bài báo này. Và một tọa độ được lập ra như Hình 9 để xác định hướng và hướng di chuyển của mỗi robot.



**Hình 9: Hệ thống tọa độ**

Chương trình sẽ chụp ảnh với chu kỳ 0.5 giây. Những ảnh này được matlab lưu trữ dưới dạng không gian màu RGB. Các ảnh này sẽ được chuyển sang không gian màu HSV và không gian màu YIQ nhằm giúp việc chọn ngưỡng thích hợp trong sáu mặt phẳng màu double (H, S, V, Y, I và Q) cho từng robot được thuận lợi hơn. Sau đó, phép toán phân đoạn ảnh theo ngưỡng được áp dụng trong sáu mặt phẳng này với phương pháp tìm và sửa sai.

Kết hợp với phép toán hình thái học trong xử lý ảnh cho việc phân loại, xác định vị trí, tâm phân thân và tâm phân đầu cho từng robot sẽ dựa trên các ngưỡng đặc trưng này. Kết quả là các thông tin cơ bản của từng robot trong mô hình thí nghiệm đã được xác định (Hình 10). Các thông số khác như: hướng đứng, vận tốc và hướng di chuyển của mỗi robot sẽ được trình bày dưới đây.



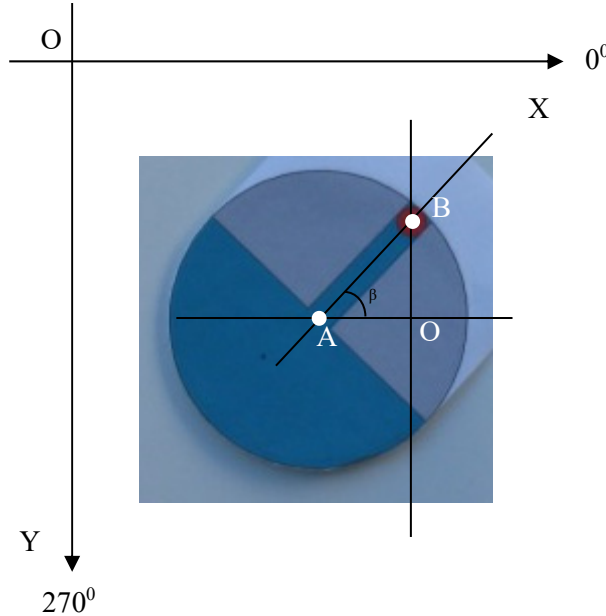
**Hình 10: Các robot trong mô hình thí nghiệm**

**3.1 Xác định hướng đứng**

Hướng đứng của mỗi robot được xác định dựa trên tâm phần đầu và tâm phần thân của cùng một robot. Tọa độ tâm phần thân (điểm A) và tọa độ tâm phần đầu (điểm B) được xác định như Hình 11. Hướng đứng được xác định bởi công thức sau:

$$\text{Actan}(\beta) = \frac{BO}{AO} \text{ (degrees)} \quad (3.1)$$

Dựa trên giá trị của  $\text{actan}(\beta)$  và hệ tọa độ đã được định nghĩa (Hình 9) để xác định hướng đứng của robot.



**Hình 11: Xác định hướng đứng**

**3.2 Xác định vận tốc và hướng di chuyển**

Vận tốc của mỗi robot được xác định dựa trên hai khung ảnh kế tiếp nhau. Hình 12 mô tả tọa độ tâm phần thân tại thời điểm  $t_n$  (điểm A) và tọa độ tâm phần thân tại thời điểm  $t_{n+1}$  (điểm B). Vận tốc sẽ được xác định bằng công thức sau:

$$v = \frac{AB \text{ (pixel)}}{0.5 \text{ (giây)}} \quad (3.2)$$

Áp dụng hàm lượng giác ngược để xác định góc đi chuyển của robot bởi công thức:

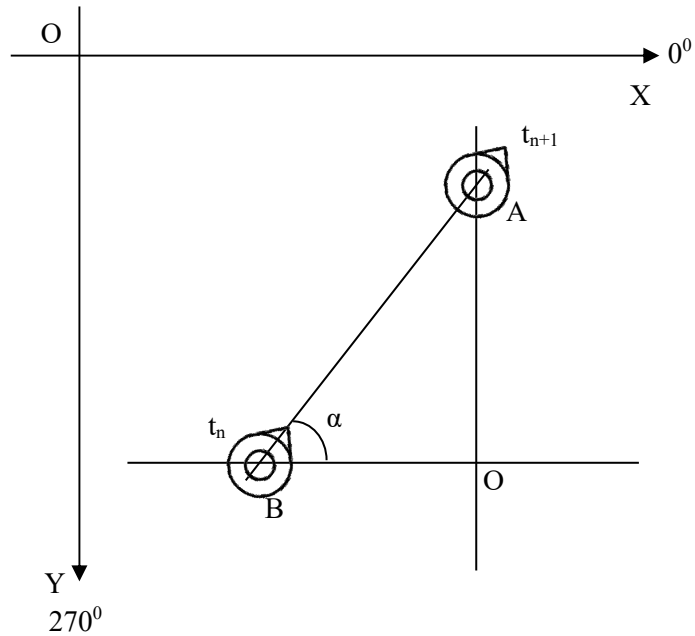
$$\text{Actan}(\alpha) = \frac{AO}{BO} \text{ (degrees)} \quad (3.3)$$

Dựa trên giá trị của  $\text{actan}(\alpha)$  và hệ tọa độ đã được định nghĩa (Hình 8) để xác định hướng di chuyển của robot. Kết quả chương trình (Hình 13) chỉ thể hiện các thông số cơ bản của mỗi robot qua hai lần chụp ảnh liên tiếp nhau. Các thông số cơ

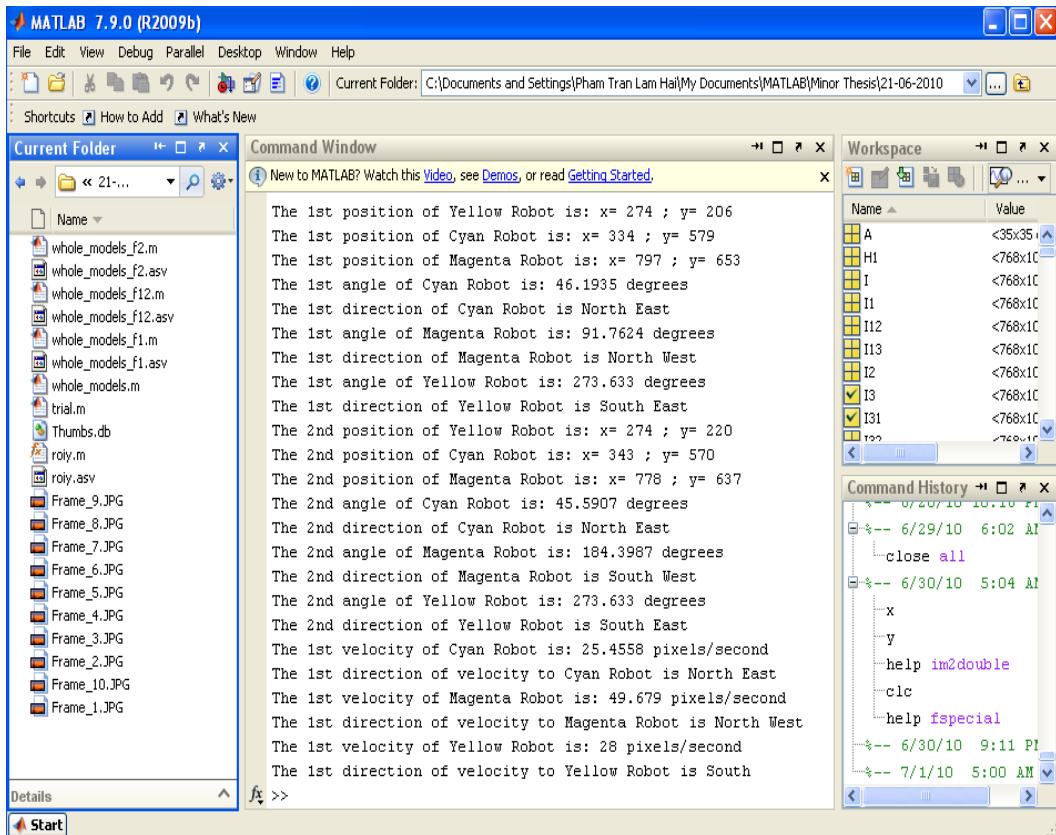
bản của mỗi robot được thể hiện trong Bảng 1. Nhờ đó, hệ thống điều khiển trung tâm sẽ xuất tín hiệu điều khiển các robot bày đàn di chuyển tránh được vật cản và va chạm lẫn nhau.

**Bảng 1: Các thông số của mỗi robot**

Robot	Các thông số
Robot màu vàng	Vị trí: x=274; y=206
	Hướng đứng: Đông Nam
	Vận tốc: 28 pixels/giây
Robot màu xanh lơ	Hướng đi chuyển: Nam
	Vị trí: x=334; y=579
	Hướng đứng: Đông Bắc
Robot màu hồng sẫm	Vận tốc: 25.5 pixels/giây
	Hướng đi chuyển: Đông Bắc
	Vị trí: x=797; y=653
Robot màu hồng sẫm	Hướng đứng: Tây Bắc
	Vận tốc: 49.7 pixels/giây
	Hướng đi chuyển: Tây Bắc



Hình 12: Xác định vận tốc và hướng di chuyển



Hình 13: Kết quả thu được



#### 4 KẾT LUẬN

Bài báo đã hoàn thành phần chương trình xử lý ngoại tuyến ở khu vực làm việc trong nhà thuộc dự án robot bầy đàn. Chương trình này được viết bằng ngôn ngữ lập trình Matlab để phân loại, định vị, xác định hướng đứng, vận tốc và hướng di chuyển của mỗi robot trong bầy đàn. Do đó, chương trình có thể phát hiện từng con robot đang di chuyển từ bất cứ hướng nào trong khu vực làm việc, qua đó giúp cho chúng tránh được vật cản và va chạm lẫn nhau. Chúng tôi đã tìm ra các thông số có ích cho các robot làm nhiệm vụ giám sát an ninh trong nhà dưới sự quản lý của hệ thống điều khiển trung tâm thông qua một camera treo. Tuy nhiên, chương trình vẫn còn các hạn chế:

Tọa độ các tâm của ba robot trong ảnh có sai lệch do các bước xử lý ảnh đã được trình bày ở trên.

Không thích hợp trong môi trường bên ngoài và các đối tượng có hình dạng bất kì.

Việc tính toán chỉ đúng trong không gian 2D (chiều x,y).

Chương trình này có thể được tiếp tục phát triển để xử lý trực tuyến với máy ảnh kết nối không dây trong việc theo dõi các con robot.

#### TÀI LIỆU THAM KHẢO

1. González, RC và Woods, RE, 2008. Digital Image Processing. Pearson Prentice Hall. Upper Saddle River. New Jersey, pp. 954 .
2. Kruse, E và Wahl, F.M., 1998. Camera-based monitoring system for mobile robot guidance. IEEE/RSJ International

Conference on Intelligent Robots and Systems. Vol.2, pp. 248 – 1253.

3. Kerr, W., Spears, D., Spears, W. và Thayer, D., 2005. Two formal gas models for multi-agent sweeping and obstacle avoidance. Lecture Notes in Computer Science. Heidelberg: Springer Berlin. pp. 111-130.
4. Lotufo, R.A., Audigier, R, Sau' de, A.V. & Machado, R.C., 2008. Morphological Image Processing. Microscope Image Processing. Academic Press. pp. 113-157.
5. Ogren, P., Fiorelli, E. và Leonard, N.E., 2004. Cooperative control of mobile sensor networks: adaptive gradient climbing in a distributed environment. IEEE Transactions on Automatic Control. Vol. 49, No. 8, pp. 1292-1302.
6. Pierre Soille (2002), Morphological image analysis: principles and application 2nd, Springer, pp.1-3.
7. Shirai, Y. và Inoue, H., 1973. Guiding a Robot by Visual Feedback in Assembling Tasks. Pattern Recognition. Vol. 5, pp. 99-108.
8. Webster, R.J. và Brannon, A.S. The Electronic Ball Boy: a reactive visually guided mobile robot for the tennis court. IEEE International Conference on Robotics and Automation, 2002, vol. 2, pp. 2054 – 2059.
9. Wikipedia, 2014. HSL and HSV [online]. [http://en.wikipedia.org/wiki/HSL\\_and\\_HSV](http://en.wikipedia.org/wiki/HSL_and_HSV) assessed on 01/11/2014
10. Wilhelm Burger và Mark J. Burge (2009), Principles of Digital Image Processing: Fundamental Techniques, Sprigner, p.160.