

DOI:10.22144/ctu.jvn.2020.002

ĐIỀU KHIỂN VÀ PHÂN LOẠI VẬT THỂ DỰA TRÊN MÀU SẮC SỬ DỤNG CÁNH TAY ROBOT 3 BẬC TỰ DO CỦA FISCHERTECHNIK

Phạm Thế Thịnh¹ và Đỗ Vinh Quang^{2*}

¹Sinh viên lớp Cơ điện tử, khóa 2015, Trường Đại học Kỹ thuật- Công nghệ Cần Thơ

²Khoa Kỹ thuật cơ khí, Trường Đại học Kỹ thuật-Công nghệ Cần Thơ

*Người chịu trách nhiệm về bài viết: Đỗ Vinh Quang (email: dvquang@ctu.edu.vn)

Thông tin chung:

Ngày nhận bài: 22/08/2019

Ngày nhận bài sửa: 09/11/2019

Ngày duyệt đăng: 28/02/2020

Title:

Control and classification of objects based on colors using 3-axis robot arm of Fischertechnik

Từ khóa:

Cánh tay robot của Fischertechnik, cánh tay 3 bậc tự do, SimMechanics với Inventor, xử lý ảnh LabVIEW

Keywords:

Fischertechnik's robotic arm, LabVIEW Image Processing Techniques, SimMechanics, 3-axis robotic arm.

ABSTRACT

This paper presents a method for building a controller of a 3-axis robotic arm model from Fischertechnik. Autodesk Inventor software was used to draw the 3D model of the arm which was then converted to the model in Matlab (for controller design and simulation) via SimMechanics link toolbox. Feedback control as well as forward and inverse kinematics were applied to control positions of joints and gripper respectively. The arm has been integrated camera to perform color classification based on LabVIEW's image processing library. A Human Machine Interface (HMI) was also designed using LabVIEW that allows operators to control and supervise the model in operation. The experimental results show that the robotic arm model can move to desired positions with low error (<8%) and high stability (standard deviation in percentage < 3%). Therefore, the model meets basic requirements of a robotic arm applied for product classification in warehouse. Moreover, it can serve for studying, learning about robotic arms and control algorithms such as ON-OFF, PID, etc.

TÓM TẮT

Bài báo trình bày phương pháp xây dựng bộ điều khiển cho mô hình cánh tay robot 3 bậc tự do của Fischertechnik. Ứng dụng phần mềm Inventor thiết kế mô hình 3D cho cánh tay, sau đó sử dụng công cụ SimMechanics hỗ trợ liên kết từ Matlab với Cad 3D Inventor để mô hình hóa, mô phỏng. Nguyên lý điều khiển hồi tiếp và bài toán động học thuận, động học nghịch đã được áp dụng vào cánh tay robot 3 bậc tự do để điều khiển vị trí các khớp cũng như khâu chấp hành. Cánh tay được tích hợp thêm camera để thực hiện quá trình xử lý ảnh phân loại màu dựa trên thư viện xử lý ảnh của LabVIEW. Một giao diện được thiết kế (sử dụng LabVIEW), có chức năng cho phép người điều khiển quan sát trực tuyến các chế độ của mô hình đang hoạt động. Kết quả thực nghiệm cho thấy cánh tay gấp vật đến vị trí mong muốn với sai lệch tương đối thấp (<8%) và ổn định cao với độ lệch chuẩn thấp (<3%). Như vậy, mô hình đã xây dựng đáp ứng được các yêu cầu cơ bản của một mô hình cánh tay trong ứng dụng xếp kho hàng phân loại sản phẩm. Hơn thế nữa, nó còn có thể phục vụ cho việc học tập, tìm hiểu về cánh tay robot và một số giải thuật điều khiển như ON-OFF, PID, ...

Trích dẫn: Phạm Thế Thịnh và Đỗ Vinh Quang, 2020. Điều khiển và phân loại vật thể dựa trên màu sắc sử dụng cánh tay robot 3 bậc tự do của Fischertechnik. Tạp chí Khoa học Trường Đại học Cần Thơ. 56(1A): 11-20.

1 GIỚI THIỆU

Ngày nay, tay máy công nghiệp (cánh tay robot) xuất hiện nhiều trong các nhà máy, xí nghiệp ở Việt Nam với các nhiệm vụ: lắp ráp, bốc dỡ hàng hóa, sơn, hàn,...Do vậy, nhu cầu thực tế về nguồn lao động có thể thao tác, lập trình, điều khiển tay máy là rất cần thiết. Nhiều trường đại học trên cả nước đã đưa môn học robot công nghiệp vào giảng dạy và thậm chí ở các nước tiên tiến còn đưa ra một ngành học nghiên cứu về robot (robotics). Tuy nhiên, việc thực hành với tay máy công nghiệp ở nước ta còn hạn chế do chi phí đầu tư cao. Việc tìm hiểu, nghiên cứu một mô hình cánh tay robot giá rẻ để cơ bản hiểu được cách thức hoạt động cũng như điều khiển một tay máy công nghiệp là cần thiết.

Mô hình Fischertechnik được phát minh từ một nhà sáng chế người Đức (Dr. Artur Fischer.1964) đã được sử dụng trong các phòng thí nghiệm điện tử, cơ điện tử ở một số trường Cao đẳng và Đại học trên thế giới (Peng and Huang, 2013; Huang *et al.*, 2018). Nhiều công ty lớn, nổi tiếng trên thế giới như Siemens, BMW đã sử dụng các mô hình của Fischertechnik để trình diễn các dây chuyền sản xuất (Huang *et al.*, 2018). Ngoài ra, các trường cao đẳng và đại học ở Đức và Hoa Kỳ đã sử dụng robot Fischer để thực hiện việc giảng dạy và thực hành ở một số môn (Freund and Matysczok, 2002). Với nhiều quốc gia và các trường đã ứng dụng các mô hình robot Fischertechnik như đã tìm hiểu, chúng hoàn toàn có thể đáp ứng được các yêu cầu trong việc giảng dạy và nghiên cứu ở Việt Nam. Tuy nhiên, chi phí nói riêng cho bộ điều khiển chính hãng (gần 400 USD) vẫn là khá đắt. Để giải quyết vấn đề này, đồng thời giúp vận dụng được những kiến thức đã học, board Arduino Mega cùng với LabVIEW đã được sử dụng để điều khiển mô hình.

Việc ứng dụng LabVIEW và Arduino trong điều khiển và tự động hóa là rất phổ biến hiện nay. Aizat *et al.* (2018) đã ứng dụng xử lý ảnh màu để xác định đối tượng từ đó đưa ra quyết định cho cánh tay. Ở nghiên cứu của Ramesh *et al.* (2015) cánh tay được gắn một camera để tiến hành phân loại vật thể dựa vào màu sắc với board điều khiển trung tâm là Arduino UNO kết hợp với phần mềm LabVIEW. Kết quả cho thấy sự ổn định khi thực hiện xử lý ảnh trên LabVIEW. Obinna and Szemes (2017) đã dùng

kỹ thuật xử lý ảnh mà thư viện trong LabVIEW hỗ trợ để cải thiện chất lượng màu.

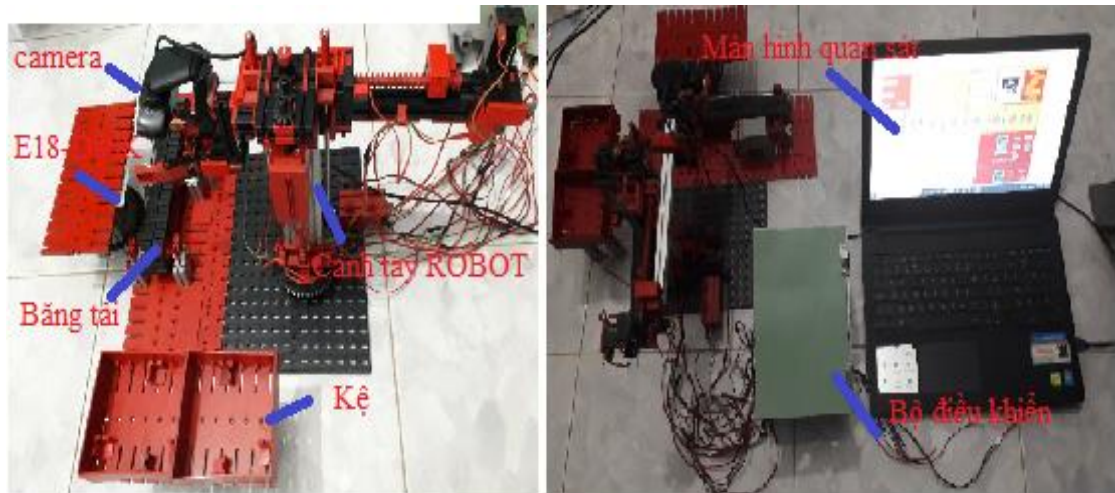
Nhìn chung, các bài báo chỉ đi sâu vào việc nghiên cứu ứng dụng xử lý ảnh trên LabVIEW mà chưa đưa ra các phương pháp điều khiển cánh tay robot. Vì vậy, bài báo này ngoài việc kế thừa các kỹ thuật xử lý ảnh còn đưa ra cách xây dựng bộ điều khiển, mô phỏng cụ thể cho một cánh tay robot. Cụ thể hơn, một bộ điều khiển (board Arduino Mega) cho mô hình cánh tay robot gấp vật từ băng tải (mô hình của hãng Fischertechnik) kết hợp với xử lý ảnh (phần mềm LabVIEW) để phân loại sản phẩm đã được xây dựng. Trong đó, bài toán động học thuận, động học nghịch và nguyên lý điều khiển hồi tiếp đã được áp dụng để điều khiển vị trí gấp và nhả vật. Thêm vào đó, hoạt động và điều khiển của cánh tay cũng được mô phỏng dựa trên công cụ SimMechanics liên kết với Inventor.

2 PHƯƠNG PHÁP THỰC HIỆN

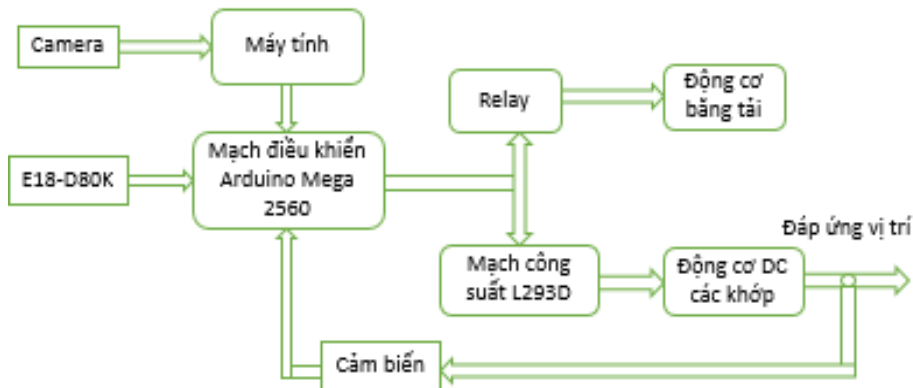
2.1 Tổng quan về mô hình

Mô hình hệ thống sắp xếp hàng hóa được xây dựng như Hình 1. Hệ có các thành phần chính bao gồm: một mô hình cánh tay robot của Fischertechnik tên ROBOTICS TXT Automation Robots (cánh tay có ba bậc tự do, trong robot công nghiệp được gọi là robot tọa độ trụ và có phạm vi làm việc là dạng hình tọa độ trụ trong không gian); một mô hình băng tải và một khay chứa vật đều được lắp ghép từ các khối cơ bản của Fischertechnik; một camera (Webcam Hxsj S20); một cảm biến vật cản (E18-D80); và một mạch điều khiển Arduino Mega.

Nguyên lý điều khiển của hệ thống mô hình được thực hiện như Hình 2. Băng tải vận chuyển vật đến vị trí cảm biến vật cản E18-D80, cảm biến phát hiện vật thể đưa tín hiệu về bộ điều khiển trung tâm Arduino Mega 2560 làm dừng hoạt động của băng tải qua relay. Sau đó, bộ điều khiển trung tâm ra lệnh điều khiển cánh tay đến vị trí của vật trên băng tải và thực hiện gấp vật qua mạch công suất L293D, đồng thời quá trình xử lý ảnh phân loại màu cũng được thực hiện để đưa tín hiệu quyết định vị trí thả vật cho cánh tay (tọa độ mong muốn cho từng vật được nhập trên giao diện). Việc thực hiện nguyên lý điều khiển trên được chia làm hai hướng: (1) Điều khiển vị trí cánh tay và (2) Xây dựng xử lý phân loại màu trên LabVIEW.



Hình 1: Mô hình thí nghiệm

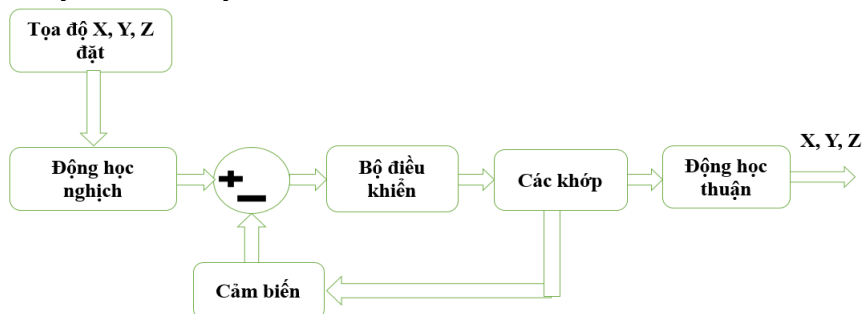


Hình 2: Sơ đồ điều khiển hệ thống

2.2 Xây dựng điều khiển vị trí cánh tay 3 bậc tự do

Bài toán điều khiển cánh tay robot là đưa khâu chấp hành đến vị trí làm việc mong muốn. Để có thể điều khiển vị trí cánh tay robot trước hết phải tìm được hai phương trình động học thuận và động học ngược. Xây dựng giải thuật điều khiển dựa trên phương trình động lực học. Nguyên lý điều khiển vị trí cánh tay được trình bày như Hình 3. Từ tọa độ

mong muốn (X, Y, Z) của khâu chấp hành, các thông số khớp được tìm thông qua phương trình động học ngược. Sau đó, chúng được so sánh với các giá trị thông khớp hiện tại được cảm biến encoder phản hồi về, sai lệch sẽ qua bộ điều khiển tác động đến các động cơ của cánh tay. Từ các thông số khớp hiện tại được lấy liên tục từ encoder, qua phương trình động học thuận, vị trí hiện tại của khâu chấp hành sẽ tính được theo thời gian thực.



Hình 3: Sơ đồ điều khiển vị trí cánh tay

Động học thuận của mô hình

Mô hình cánh tay được thiết kế theo hình dạng của robot công nghiệp loại tọa độ trụ có 3 bậc tự do.

Tên các khớp và phạm vi làm việc của chúng được trình bày trong Bảng 1. Các thông số kỹ thuật của động cơ được trình bày trong Bảng 2 (Fischertechnik ROBO TX, 2013).

Bảng 1: Loại khớp và phạm vi làm việc từng khớp

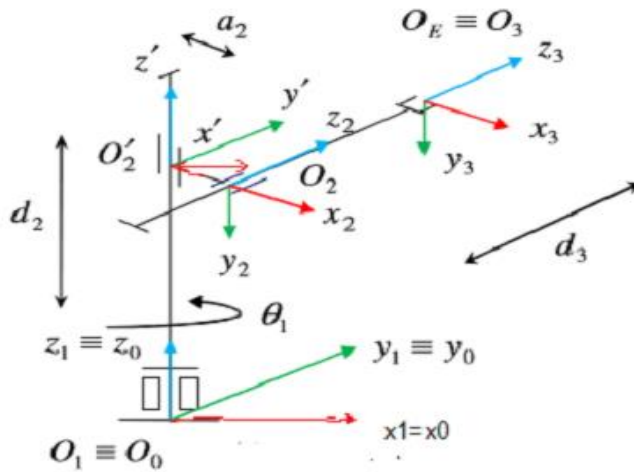
Khớp	Loại khớp	Phạm vi làm việc
1	Khớp quay	Quay ngược chiều kim đồng hồ, 0 ⁰ đến 360 ⁰
2	Khớp tịnh tiến	Tịnh tiến theo chiều cao từ 0cm đến 15cm
3	Khớp tịnh tiến	Tịnh tiến theo phương ngang từ 0cm đến 9cm

Với mô hình đã có, tiến hành tìm động học thuận được trình bày trong sách “Kỹ thuật robot” (Đào Văn Hiệp, 2004). Thực hiện theo quy tắc Denavit

Hartenberg (Đào Văn Hiệp, 2004) nhận được các hệ trục tọa độ robot như Hình 4.

Bảng 2: Thông số kỹ thuật

Động cơ	Nguồn cấp (V)	Công suất (W)	Vận tốc (vòng/phút)
DC tích hợp encoder	9 V	1,2 W	105 vòng/phút
XS motor DC	9V	1 W	6000 vòng/phút



Hình 4: Các hệ tọa độ được gán trên robot

Bảng 3: Thông số bảng DH

Khâu	θ_i	α_i	d_i	a_i
1	θ_1	0	0	0
2	-90	-90	d_2^*	a_2
3	0	0	d_3^*+18	0

Từ các hệ tọa độ, lập bảng DH (Denavit Hartenberg) như Bảng 3.

Trong đó: $a_2 = 3 \text{ cm}$, $d_2^* = d_2$, $d_3 = d_3^* + 18$, θ_1 , d_2^* , d_3^* là các biến khớp.

Từ các thông số trong bảng DH, tìm các ma trận biến đổi từ ma trận tổng quát (1):

$$A_i^{i-1} = \begin{bmatrix} \cos(\theta_i) & -\cos(\alpha_i) * \sin(\theta_i) & \sin(\theta_i) * \sin(\alpha_i) & a_i * \cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) * \cos(\alpha_i) & -\cos(\theta_i) * \sin(\alpha_i) & a_i * \sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

Từ ma trận (1) thay các thông số từ bảng DH, nhận được:

$$A_1^0 = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$A_2^1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & -a_2 \\ 0 & -1 & 0 & d_2^* \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$A_3^2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3^* + 18 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

Nhân các ma trận (2), (3), (4) với nhau:

$$A_3^0 = A_1^0 * A_2^1 * A_3^2 = H \quad (5)$$

$$= \begin{bmatrix} s\theta_1 & 0 & c\theta_1 & a_2s\theta_1 + (d_3^* + 18)c\theta_1 \\ -c\theta_1 & 0 & s\theta_1 & -a_2c\theta_1 + (d_3^* + 18)s\theta_1 \\ 0 & -1 & 0 & d_2^* \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Trong đó: $s\theta_1 = \sin(\theta_1)$, $c\theta_1 = \cos(\theta_1)$.

Phương trình (6) là ma trận biến đổi thuần nhất của robot. Từ ma trận biến đổi thuần nhất (6), rút ra hệ phương trình như sau:

$$\begin{cases} X = a_2s\theta_1 + (d_3^* + 18)c\theta_1 \\ Y = -a_2c\theta_1 + (d_3^* + 18)s\theta_1 \\ Z = d_2^* \end{cases} \quad (7)$$

Hệ phương trình (7) là hệ phương trình miêu tả động học thuận để xác định vị trí của mô hình cánh tay robot Fischertechnik khi biết trước giá trị các thông số khớp.

Động học ngược của mô hình

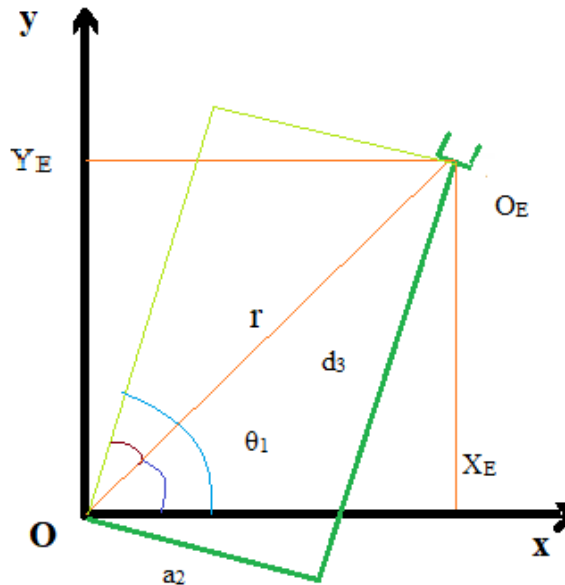
Động học ngược của mô hình được tìm bằng phương pháp hình học và được thực hiện như sau: Chiếu mô hình cánh tay xuống mặt phẳng Oxy như Hình 5 và theo phương đứng.

Có tọa độ: $O_E \equiv O_3$

Với $O_E (X_E, Y_E, Z_E)$ là tọa độ tâm vật. Thu được hệ phương trình mô tả động học ngược:

$$\begin{cases} \theta_1 = a \tan 2(X_E, Y_E) + a \tan 2(r, a_2) \\ d_2^* = Z_E \\ d_3^* = (\sqrt{X_E^2 + Y_E^2} - a_2^2 - 18) \end{cases} \quad (8)$$

Trong đó: $r = \sqrt{X_E^2 + Y_E^2}$

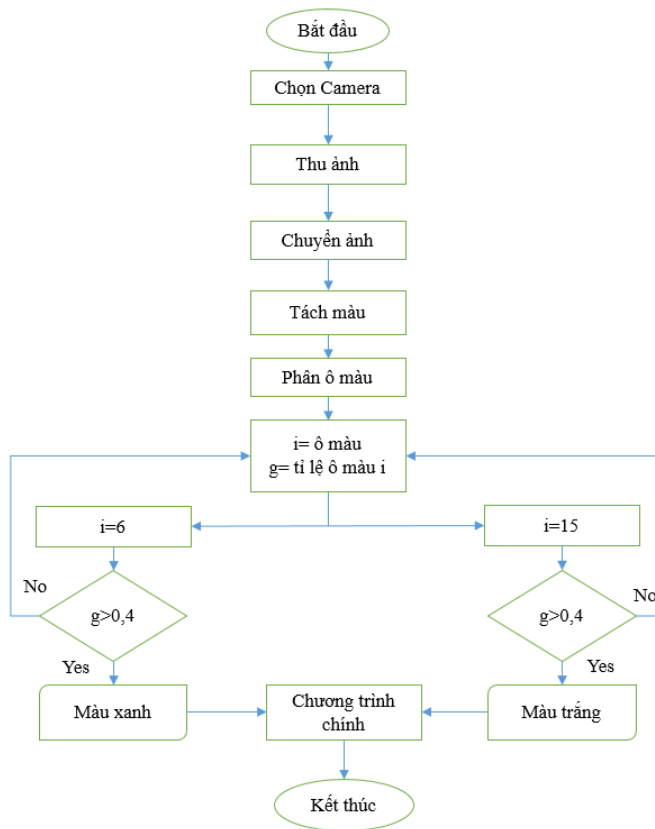


Hình 5: Mô hình chiếu trên mặt phẳng Oxy

2.3 Xử lý ảnh trên LabVIEW

LabVIEW hỗ trợ đầy đủ các chức năng của một quá trình xử lý ảnh và quá trình xử lý được thực hiện như sau: Hình ảnh nhận từ camera được đưa vào một vùng nhớ để tiến hành xử lý ảnh. Ảnh thu được là hệ màu RGB (Red, Green, Blue) được chuyển đổi thành hệ màu HSL (Hue, Saturation, Lightness), sau đó được phân tách màu tương ứng với hệ màu HSL. Tiếp theo, hàm IMAQ Color Learn (thuộc thư viện màu tiện ích) được sử dụng để thực hiện chức năng học màu và đưa ra các kệ ô màu theo tỷ lệ khả năng có thể là màu của vật tại từng kệ màu (với số kệ ô phụ thuộc vào sự chọn lựa độ nhạy màu). Trong bài báo này, phân loại hai màu xanh và trắng, độ nhạy màu được chọn là LOW, nên chia làm 16 kệ từ 0 đến

15 (màu xanh ở kệ 6 và trắng ở 15). Hàm phân loại màu thực hiện chức năng phân loại trả giá trị đúng với màu của ô tương ứng. Ở đây, để có thể chính xác màu, chọn ngưỡng tham chiếu giá trị tại hai ô cần phân loại lớn hơn 0,4. Nếu giá trị trả về tại kệ 6 là lớn hơn 0,4, kết luận là màu xanh, còn ở kệ 15, kết luận là màu trắng. Giá trị ngưỡng này được chọn từ thực nghiệm khi quan sát giá trị tại ô kệ màu xanh (màu trắng) đối với các màu sắc khác nhau. Sau khi được kết quả phân loại màu, khối “chương trình chính” có nhiệm vụ hiển thị số lượng, đèn báo màu được phân loại. Đồng thời đưa các tọa độ mong muốn của màu đó được người dùng đặt sẵn truyền xuống chương trình Arduino thực hiện việc điều khiển cánh tay đến vị trí mong muốn. Lưu đồ giải thuật tổng quát như Hình 6.

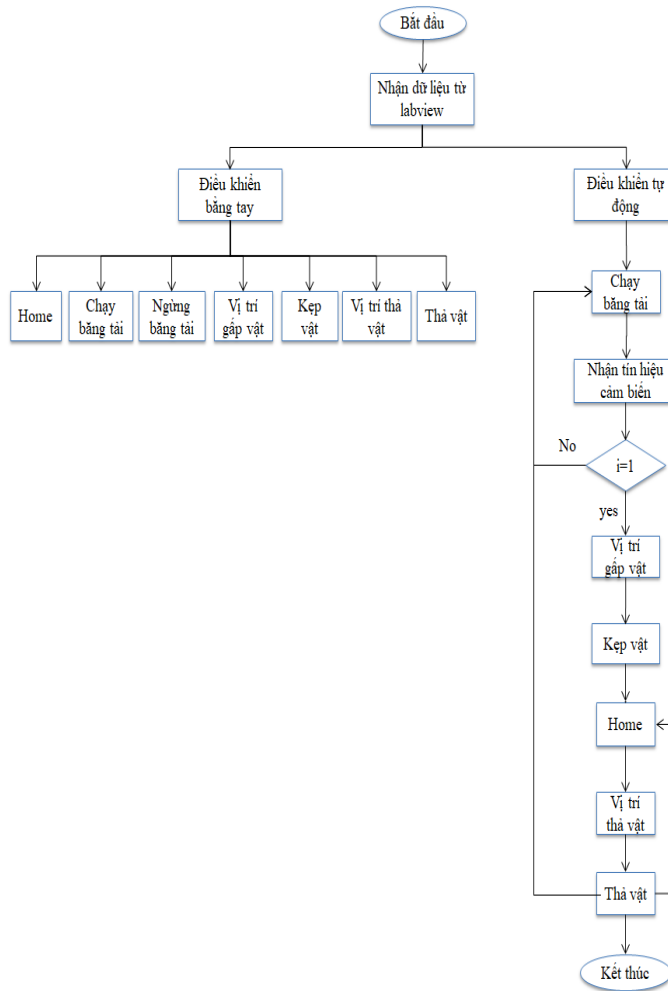


Hình 6: Lưu đồ xử lý ảnh trên LabVIEW

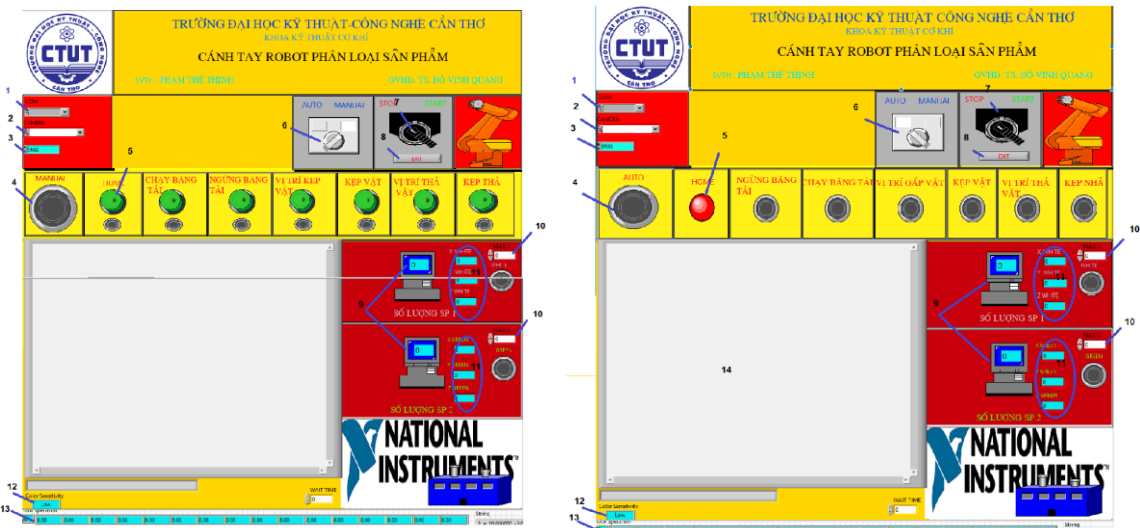
2.4 Giao diện phần mềm

Giao diện mô hình được thiết kế qua phần mềm LabVIEW và có thể tương tác với người dùng qua hai chế độ điều khiển gồm: (1) Điều khiển bằng tay và (2) Điều khiển tự động. Cả hai chế độ đều được kích hoạt trên giao diện đã thiết kế. LabVIEW giao tiếp với mạch Arduino thông qua giao tiếp Serial (giao thức UART). Chương trình viết cho bộ điều khiển này được chia làm nhiều chương trình con để

thực hiện các chức năng khác nhau như thực hiện chế độ điều khiển theo yêu cầu, điều khiển bằng tải, đọc tín hiệu cảm biến, điều khiển các động cơ của cánh tay,... Lưu đồ giải thuật tổng quát của chương trình điều khiển được trình bày như Hình 7. Giao diện điều khiển của toàn bộ hệ thống được thể hiện ở Hình 8. Ngoài các chức năng điều khiển đã được tóm lược, giao diện còn cho phép giám sát trực tuyến và ghi nhận số lượng vật đã gấp, cho phép người dùng can thiệp các lệnh điều khiển trong hệ thống.



Hình 7: Lưu đồ giải thuật điều khiển gắp vật



(a) Điều khiển bằng tay

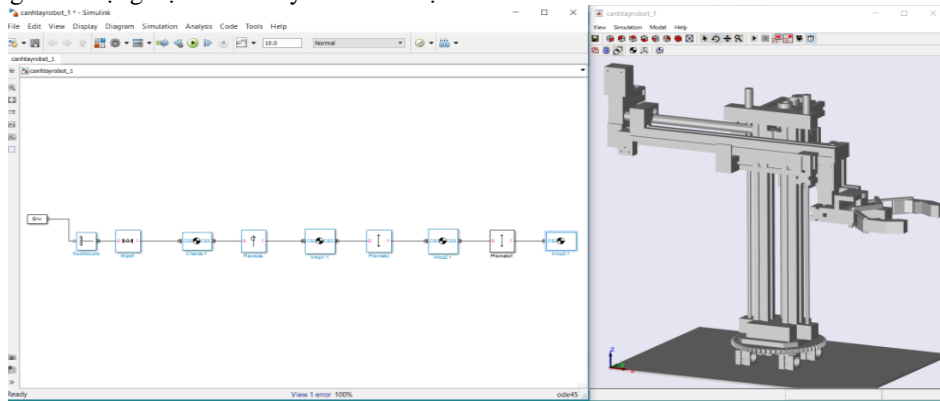
(b) Điều khiển tự động

Hình 8: Giao diện người dùng

2.5 Mô hình mô phỏng

Sim-mechanics là một công cụ trong Matlab cho phép chuyển đổi mô hình Autodesk Inventor từ môi trường Inventor sang môi trường Simulink dưới dạng các khối được kết nối bởi các khớp. Công cụ còn cung cấp cho người dùng các chức năng về xác định các vật thể và tính chất khối lượng của chúng; mô phỏng chuyển động của vật, các ràng buộc động học, hệ tọa độ; đo được chuyển động; hay hỗ trợ việc tìm phương trình động học. Điều này làm cho việc

mô hình hóa và phân tích dễ dàng hơn vì người dùng không cần trải qua quá trình mô hình hóa phức tạp (Gaber *et al.*, 2016). Còn đối với cánh tay robot khi mà việc tìm phương trình động lực học là tương đối khó khăn, việc lựa chọn ứng dụng trên để thiết kế bộ điều khiển sẽ mang lại nhiều thuận lợi. Mô hình được xuất sang Matlab như Hình 9 và sơ đồ mô phỏng sử dụng bộ điều khiển PID được trình bày ở Hình 10 mà sẽ được trình bày kỹ hơn ở phần tiếp theo.

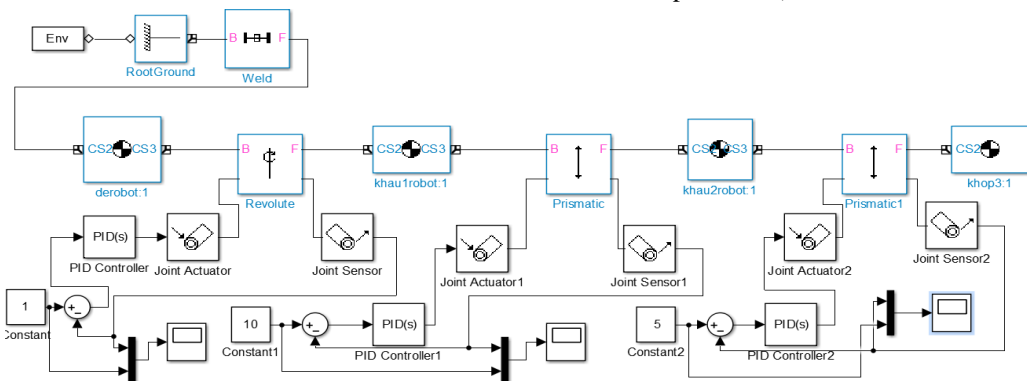


Hình 9: Mô hình trong môi trường simulink Matlab

2.6 Xây dựng thuật toán điều khiển

Bộ điều khiển PID được thiết kế cho từng khớp để kiểm tra khả năng điều khiển, đáp ứng từng khớp của mô hình khi sử dụng bộ điều khiển này (Hình 10). Khi nhận được mô hình từ Inventor liên kết Matlab, thuật toán điều khiển PID được xây dựng như sau: (1) sử dụng khối Joint Actuator để có thể tạo lực, momen làm di chuyển đối với khớp trượt, làm quay đối với khớp quay; (2) sử dụng khối Sensor Joint để phản hồi góc quay và độ di chuyển

của khớp; (3) sử dụng bộ PID trong thư viện, đầu vào bộ điều khiển là sai số giữa giá trị đặt và giá trị phản hồi về từ khối Sensor Joint, đầu ra là tín hiệu điều khiển áp lên khối Joint Actuator tác dụng cho từng khớp, sau đó tiến hành mô phỏng. Giá trị ngõ vào bộ điều khiển là các sai lệch của từng khớp được trình bày ở Hình 3 (để có được giá trị từng khớp thì phải qua chuyển đổi động học nghịch). Do đó giá trị đặt ở Hình 10 là giá trị từng biến khớp (ở đây đơn vị của giá trị đặt là radian đối với khớp quay, centimet đối với khớp tịnh tiến).



Hình 10: Bộ điều khiển PID được dùng để điều khiển cánh tay robot 3 bậc tự do

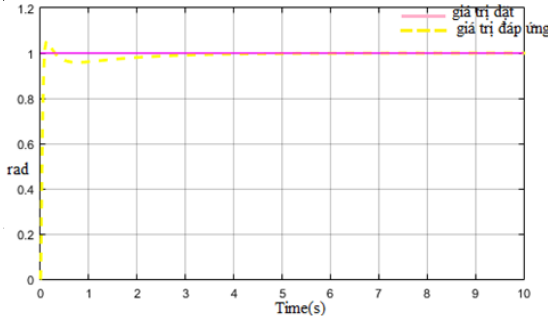
3 KẾT QUẢ VÀ THẢO LUẬN

Việc mô phỏng được tiến hành với $m_1 = 0,6$ kg, $m_2 = 0,3$ kg, $m_3 = 0,5$ kg là khối lượng tương ứng của khâu 1, 2, 3 của cánh tay robot. Các tham số PID

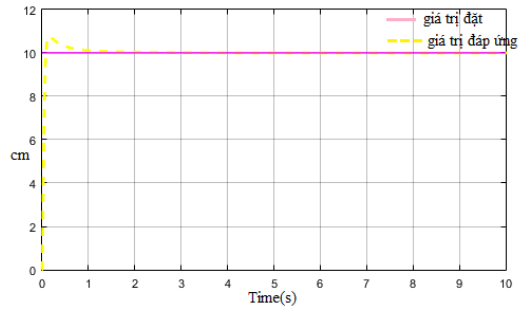
được chọn bằng phương pháp thực nghiệm Ziegler – Nichols (Huỳnh Thế Hiển và *ctv.*, 2018) là: $K_p = 120$, $T_I = 75$, $T_D = 20$. Kết quả mô phỏng của từng khớp không có nhiễu được thể hiện ở Hình 11 với

đường màu hồng là các giá trị đặt (Khớp 1 = 1 rad, khớp 2 = 10 cm, khớp 3 = 5 cm) và đường màu vàng nét đứt là giá trị đáp ứng. Kết quả cho thấy, các khớp đều đến được vị trí mong muốn. Trong đó, khớp 3 có độ vọt lố cao nhất (10%) và thời gian xác lập chậm nhất (3s). Các chỉ tiêu chất lượng cho từng

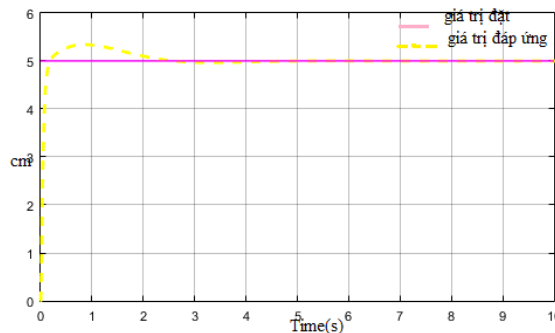
khớp được trình bày như Bảng 4. Lưu ý rằng, bộ điều khiển trong thí nghiệm này đóng vai trò kiểm nghiệm hoạt động của mạch điều khiển với mô hình. Thực tế triển khai, người dùng có thể thiết kế các giải thuật khác để có chất lượng tốt hơn.



Khớp 1



(b) Khớp 2



(c) Khớp 3

Hình 11: Đáp ứng bộ điều khiển PID

Bảng 4: So sánh các chỉ tiêu chất lượng của từng khớp với bộ điều khiển PID

Chỉ tiêu	Khớp 1	Khớp 2	Khớp 3
Độ vọt lố trên (%)	5	8	10
Độ vọt lố dưới (%)	6	-	2
Thời gian tăng (s)	0,12	0,1	0,2
Thời gian xác lập (s)	2,6	1,3	3

Sau khi thực hiện quá trình mô phỏng, tiến hành thực nghiệm khả năng điều khiển vị trí cánh tay của bộ điều khiển bằng việc cho cánh tay di chuyển đến

một tọa độ cố định nhiều lần (Ví dụ: X = 5 cm, Y = 20 cm, Z = 10 cm). Kết quả thu được bằng phương pháp đo thủ công và được trình bày ở Bảng 5.

Bảng 5: Kết quả kiểm nghiệm thực tế

Tọa độ mong muốn	Tọa độ mong muốn			Tọa độ đo được		
	X	Y	Z	X	Y	Z
1	5	20	10	5,2	19,8	10,2
2	5	20	10	5	20,4	9,8
3	5	20	10	5,4	20	10
4	5	20	10	5,2	20,2	10
5	5	20	10	5	20,2	9,8
6	5	20	10	5,2	20	10,2
Giá trị trung bình				5,17	20,1	10
Độ lệch chuẩn				0,137	0,191	0,163
Độ lệch chuẩn (%)				2,66	0,95	1,63

Kết quả cho thấy, cánh tay có thể di chuyển tới tọa độ mong muốn với sai lệch tương đối thấp (<8%). Sai lệch điều khiển này là chấp nhận được vì sai số cơ khí của mô hình là tương đối lớn. Các lần lặp lại cho độ lệch chuẩn <3%, đã chứng tỏ sự ổn định của bộ điều khiển. Cuối cùng, khả năng thực hiện gấp và nhả vật dựa trên phân loại màu sắc được kiểm nghiệm. Việc kiểm tra được tiến hành 6 lần với hệ thống hoạt động ở chế độ tự động. Kết quả được trình bày ở Bảng 6.

Bảng 6: Kết quả số lần cánh tay gấp vật thành công

Số lần thực hiện	Gấp vật: Thành công/ Không thành công
1	Thành công
2	Thành công
3	Thành công
4	Không thành công
5	Thành công
6	Thành công

Qua kiểm nghiệm tỷ lệ thành công là cao hơn 80%. Lần thực hiện thứ 4 không thành công là do ánh sáng thay đổi đột ngột, làm cho camera bị nhiễu sáng nên không phân biệt rõ màu của vật.

4 KẾT LUẬN VÀ ĐỀ XUẤT

Một bộ điều khiển cho mô hình cánh tay robot gấp vật từ băng tải kết hợp với xử lý ảnh để phân loại sản phẩm đã được hoàn thành. Bài toán động học thuận, động học nghịch và nguyên lý điều khiển hồi tiếp đã được áp dụng vào cánh tay robot 3 bậc tự do để điều khiển vị trí gấp và thả vật ở mọi vị trí trong không gian làm việc của nó. Việc mô phỏng hoạt động và điều khiển của cánh tay cũng được thực hiện dựa trên phần mềm Matlab liên kết với Inventor. Mặc dù sai lệch vị trí còn tương đối lớn do sai lệch cơ khí của mô hình và việc phân loại màu sắc còn bị ảnh hưởng bởi nhiễu sáng, mô hình hoàn thiện này có thể phục vụ cho việc tìm hiểu về cánh tay robot và một số giải thuật điều khiển như ON-OFF, PID,... Hướng phát triển tiếp theo của bài báo sẽ là sử dụng các bộ điều khiển như mạng Noron

hàm cơ sở xuyên tâm, bộ điều khiển trượt, ... để nâng cao hiệu suất.

TÀI LIỆU THAM KHẢO

Ali, M.H., Aizat, K., Yerkhan, K., Zhandos, T., and Anuar, O., 2018. Vision-based robot manipulator for industrial applications. *Procedia Computer Science*. 133(2018): 205 – 212.

Đào Văn Hiệp, 2004. *KỸ THUẬT ROBOT*. Nhà xuất bản khoa học và kỹ thuật, 246 trang.

Fruend, J., and Matysczok, C., 2002. Designing flexible manufacturing systems with augmented reality. In *The First IEEE International Workshop Augmented Reality Toolkit*. 3 pages.

Gaber, A. N. A. E., Rezekka, S. F., and El Gamal, H. A., 2016. Design and Position Control of Arm Manipulator, Experimentally and in MATLAB SimMechanics. *International Journal of Engineering Research and Technology*. 5(08): 352–359.

Huỳnh Thế Hiển, Huỳnh Minh Vũ và Nguyễn Hoàng Dũng, 2018. Bộ điều khiển PID dựa trên mạng No-ron hàm cơ sở xuyên tâm. *Tạp chí Khoa học Trường Đại học Cần Thơ*. 7A(2018): 9-19.

Huang, Z., Liu, L. and Li, Z., 2018. Study on the reform of innovative experimental teaching mode based on Fischer robot. *Advances in Social Science, Education and Humanities Research*. 194: 406-410.

Obinna, N., and Szemes, P., 2017. LabVIEW Image Processing Techniques for the Industrial Quality Improvement Processes Design of a Color Calibration System Using the NI Vision Acquisition Software and NI Vision Development Module. Accessed on 4 January 2019. Available from: <https://www.researchgate.net/publication/321588731>.

Peng, H., and Huang, Z.C., 2012. Design of a welding robot based on fischertechnik combination model platform. *Advanced Materials Research*. 619(2013): 384–387.

Ramesh, H.R., and Pooja, 2017. Colour Sorting Robot in LabVIEW Using Image Processing. *International Journal of Science and Research*. 6(12): 47–50.