

NGHIÊN CỨU ĐIỀU KHIỂN ROBOT DELTA TRONG MÔ HÌNH PHÂN LOẠI SẢN PHẨM SỬ DỤNG THỊ GIÁC MÁY

Nguyễn Thị Hiền*, Nguyễn Đức Toàn, Phạm Trọng Hùng

Khoa Cơ - Điện, Học viện Nông nghiệp Việt Nam

**Tác giả liên hệ: nhtien@vnua.edu.vn*

Ngày nhận bài: 21.12.2024

Ngày chấp nhận đăng: 11.07.2025

TÓM TẮT

Nghiên cứu này nhằm xây dựng một thuật toán điều khiển robot delta tích hợp thị giác máy dùng trong dây chuyền phân loại sản phẩm theo màu sắc. Đối tượng được phân loại trong nghiên cứu là bánh KitKat có kích thước $70 \times 30 \times 7$ (mm) và được bao trong các gói có màu sắc khác nhau, trên đó có thông tin sản phẩm dưới dạng ký tự. Bánh được vận chuyển liên tục trên băng tải với vận tốc có thể điều chỉnh được. Một camera được gắn cố định trên khung nhôm, phía đầu băng tải làm nhiệm vụ thu nhận hình ảnh sản phẩm và gửi dữ liệu đến máy tính để phân tích. Thuật toán xử lý ảnh sẽ xác định màu sắc và vị trí của bánh, từ đó thực hiện gán nhãn và gửi dữ liệu đến bộ điều khiển. Dựa trên thông tin này, robot delta sẽ nhận diện, hút và phân loại bánh vào các khay định sẵn. Hệ thống sử dụng vi điều khiển với kit Arduino Mega 2560, lập trình bằng ngôn ngữ C++, Gcode, kết hợp thư viện OpenCV trong xử lý ảnh và phần mềm Qt Designer để thiết kế giao diện điều khiển, giám sát. Thời gian trung bình để hệ thống xử lý và phân loại một sản phẩm bánh là 3,5 giây. Thuật toán xây dựng giúp robot thực hiện tác vụ ngay cả khi băng tải vẫn hoạt động, từ đó nâng cao năng suất làm việc của hệ thống.

Từ khóa: Robot delta, thị giác máy, phân loại.

Study on Control of Delta Robot in Sorting Systems using Machine Vision

ABSTRACT

This study aimed to build a control algorithm for a delta robot integrated with machine vision, used in the sorting systems based on color classification. The classified products here are KitKat cakes with dimensions of $70 \times 30 \times 7$ (mm), packaged in wrappers of various colors containing printed product information. The cakes are continuously transported on an adjustable speed conveyor belt. A fixed camera, mounted on the aluminum frame at the beginning of the conveyor, captures images of cakes, which are sent to a computer for image processing to extract color and position of cakes. This information is used for product labeling and is transmitted to the controller. The delta robot uses this data to identify, grasp and place the cake into predefined tray by color. The system uses Arduino Mega 2560 microcontroller with C++ programming language, Gcode, together with OpenCV library for image processing and Qt Designer software to design a monitoring and control interface. The average time for image processing and classifying a cake is 3.5 seconds. The proposed method helps the delta robot perform tasks even though the conveyor is still operating, thereby improving the productivity of the system.

Keywords: Robot delta, machine vision, sorting system.

1. ĐẶT VẤN ĐỀ

Robot Delta (Hình 1) là một loại robot song song độc đáo, được thiết kế với ba cánh tay điều khiển nối với một bộ di chuyển. Cơ chế chính của loại robot này là sử dụng các hình bình hành để tạo ra chuyển động đầu cuối nhưng vẫn

giữ nguyên hướng, tức là chỉ chuyển động theo các phương X, Y và Z mà không bị thay đổi hướng. Bộ cố định của robot được gắn phía trên không gian làm việc và là nơi đặt tất cả các cơ cấu chấp hành (động cơ điện). Từ bộ cố định, ba cánh tay chủ động được gắn vào và liên kết với những cánh tay bị động. Những cánh tay bị

động được kết nối với một bộ hình tam giác nhỏ (bộ di động). Sự chuyển động của những cánh tay này sẽ di chuyển bộ di động theo hướng X, Y hoặc Z. Việc truyền động có thể được thực hiện với các bộ truyền động tuyến tính hoặc quay, có hoặc không có giảm tốc. Vì các động cơ truyền động đều nằm trên bộ cố định, nên các cánh tay có thể được làm bằng vật liệu nhẹ. Do đó, các bộ phận chuyển động của robot delta có quán tính nhỏ, điều này cho phép robot hoạt động ở tốc độ cao và tăng tốc rất nhanh (Zhang & cs., 2018; Ahmed & cs., 2024).

Trong công nghiệp, đặc biệt là quá trình phân loại, đóng gói các sản phẩm với tải trọng nhỏ trong lĩnh vực chế biến, bảo quản thực phẩm, dược phẩm, robot delta được ưa chuộng do những ưu điểm vượt trội: cấu trúc nhẹ, độ cứng vững cao, hoạt động với tốc độ cao và chính xác. Đặc biệt, khi kết hợp với thị giác máy, robot có khả năng tăng năng suất rất cao và tăng độ chính xác hoạt động (Zhang & cs., 2018; Vo Duy Cong & Le Hoai Phuong, 2023; Ahmed & cs., 2024).

Những năm qua, đã có nhiều nghiên cứu tập trung vào lĩnh vực robot tích hợp thị giác máy. Nghiên cứu của Zhang & cs. (2018) xây dựng một thuật toán điều khiển robot delta dựa trên thị giác máy. Thông tin về hình dáng và vị trí của đối tượng trên băng tải được đưa ra sau quá trình xử lý ảnh, kết hợp với việc sử dụng động cơ servo và điều chỉnh đồng bộ tốc độ băng tải dựa trên lưu lượng đối tượng, hệ thống có thể gấp đôi tượng từ một băng tải đang chạy sang băng tải khác theo thứ tự nhất định với tốc độ rất cao, lên đến 120 lần/phút. Vo Duy Cong & Le Hoai Phuong (2023) đã phát triển một hệ thống phân loại sản phẩm theo màu sử dụng robot delta tích hợp thị giác máy. Hình ảnh đối tượng thu nhận từ camera (không gian màu RGB) được chuyển đổi sang không gian màu HSV, từ đó sử dụng các ngưỡng để phân biệt đối tượng theo màu sắc. Hệ thống có thể phân loại đối tượng theo các màu đỏ vàng với độ chính xác 100% và màu xanh lá với độ chính xác 97,5%. Ở một nghiên cứu khác, Vo Duy Cong & cs. (2023) đã phát triển một thuật toán chuyển đổi hệ tọa độ ảnh

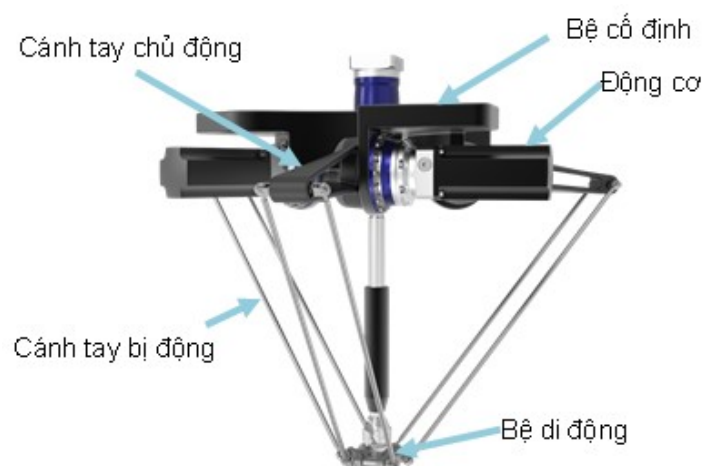
sang hệ tọa độ 3D của robot SCADA dựa trên phương pháp bình phương cực tiểu, giúp robot có thể gấp và đặt các hộp lên pallet. Sai số chuyển đổi là 0,5112mm và 0,6838mm tương ứng với các trục X và Y. Ahmed & cs. (2024) cũng đã phát triển một robot delta giá rẻ tích hợp với thị giác máy để phân loại đối tượng theo màu sắc.

Trên cơ sở kế thừa và phát triển các nghiên cứu trước đó, trong nghiên cứu này, chúng tôi xây dựng một thuật toán điều khiển robot delta kết hợp với hệ thống thị giác máy, ứng dụng trong mô hình phân loại sản phẩm bánh KitKat theo màu sắc. Bánh được đóng gói với bao bì có màu sắc khác nhau, trên đó có các thông tin sản phẩm dạng ký tự. Quá trình vận chuyển sản phẩm diễn ra liên tục trên băng tải có thể điều chỉnh tốc độ. Hệ thống sử dụng một camera gắn cố định trên khung nhôm, phía đầu băng tải để thu nhận hình ảnh của sản phẩm trong thời gian thực. Dữ liệu hình ảnh sau đó được hệ thống xử lý nhằm phân tích màu sắc, gán nhãn và xác định vị trí sản phẩm. Kết quả phân tích được gửi đến bộ điều khiển trung tâm để điều khiển robot delta thực hiện hút và phân loại bánh vào các khay chứa được bố trí ở hai bên băng tải. Hệ thống đạt thời gian phân tích, xử lý dữ liệu và thực hiện phân loại trung bình 3,5s cho mỗi sản phẩm và cho phép thực hiện thao tác phân loại ngay cả khi băng tải đang hoạt động, từ đó góp phần nâng cao hiệu quả và năng suất của hệ thống.

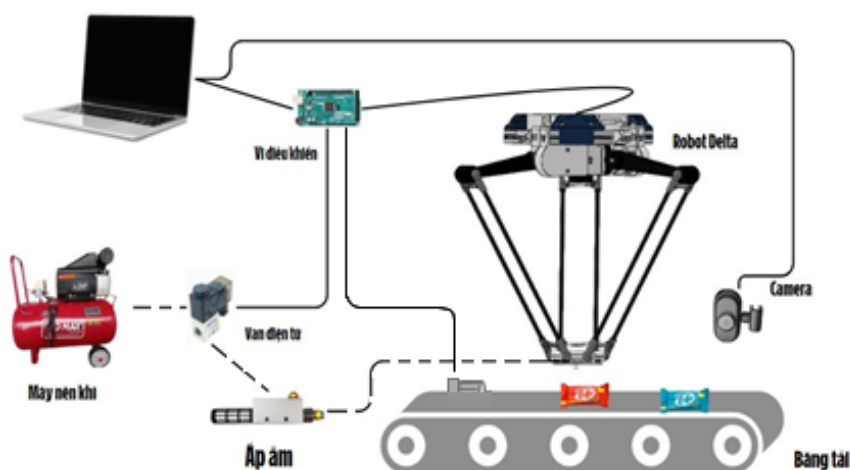
2. PHƯƠNG PHÁP NGHIÊN CỨU

2.1. Vật liệu, thiết bị

Sơ đồ cấu trúc của hệ thống phân loại bánh KitKat sử dụng robot delta kết hợp thị giác máy được thể hiện trên hình 2. Mô hình hệ thống gồm có 4 module chính: (i) Robot delta với giác hút chân không gắn ở bộ di động làm nhiệm vụ hút và đưa bánh đến các khay chứa; (ii) Băng tải vận chuyển bánh; (iii) Hệ thống thị giác máy bao gồm camera và máy tính xử lý; (iv) Bộ điều khiển dùng vi điều khiển với kit Arduino Mega 2560.



Hình 1. Robot delta



Ghi chú: —: Tín hiệu điện; - - -: Đường cấp khí nén.

Hình 2. Cấu trúc hệ thống phân loại bánh KitKat sử dụng robot delta

Trong nghiên cứu này, robot delta trong mô hình hệ thống phân loại sản phẩm được xây dựng với kích thước hình học và vùng làm việc được thể hiện trong bảng 1 (để xác định kích thước này, phần mềm Matlab được sử dụng để mô phỏng, xác định vùng làm việc, từ đó lựa chọn kích thước tối ưu cho mô hình robot (Laribi & cs., 2007; Shehata & cs., 2021)). Các cánh tay của robot delta được truyền động bằng các động cơ bước NEMA 17 kết hợp với bộ truyền đai có tỉ số truyền 1:3, đặt trên bộ cố định. Trên bộ di động có gắn giác hút chân không ZP3P-20PTSF, được cấp nguồn khí nén qua van áp âm để hút sản phẩm phân loại. Để vận chuyển bánh KitKat, hệ thống sử dụng băng tải làm bằng nhựa PVC có kích thước 600 × 60mm (dài × rộng), được truyền

động bằng động cơ bước NEMA 17 giúp dễ dàng điều chỉnh tốc độ và đồng bộ tốc độ với robot nhằm xác định chính xác vị trí của bánh trên băng tải. Webcam có độ phân giải 1920 × 1080 (full HD), tốc độ khung hình 30fps được gắn trên khung nhôm ở phía đầu băng tải để thu thập hình ảnh gửi về máy tính. Máy tính MSI GL62M 7RDX với bộ xử lý Intel® Core i5-7300HQ dùng để phân tích, xử lý hình ảnh. Bộ điều khiển trung tâm của hệ thống dùng kit Arduino Mega 2560, kết hợp bo mạch mở rộng Ramps 1.4, driver A4988 và CNC Shield V4 để điều khiển cho các động cơ. Một khung nhôm với kích thước 600 × 500 × 700 (mm) (dài × rộng × cao) được thiết kế để gá lắp tất cả các thiết bị của hệ thống. Hình 3 là hình ảnh của mô hình hệ thống phân

loại sản phẩm sử dụng robot delta kết hợp thị giác máy sau khi được gia công, lắp ráp.

2.2. Phương pháp nghiên cứu

Nghiên cứu tổng quan hệ thống, kế thừa kết quả của các nghiên cứu trước đó để xây dựng thuật toán xử lý ảnh, nhằm phân loại bánh KitKat dựa trên màu sắc của các bao gói sản phẩm. Bên cạnh đó, thuật toán chuyển đổi hệ tọa độ ảnh sang hệ tọa độ thực (hệ tọa độ robot) cũng được nghiên cứu và áp dụng nhằm xác định chính xác vị trí của bánh trên băng tải, giúp hệ thống điều khiển robot hút và phân loại bánh ngay cả khi băng tải đang chạy.

Trong nghiên cứu này, các môi trường phát triển phần mềm như Visual Studio Code, PlatformIO và Visual Studio được sử dụng để lập trình và triển khai thuật toán điều khiển trên các dòng vi điều khiển. Ngôn ngữ lập trình chính bao gồm C++, mã Gcode, kết hợp với thư viện OpenCV để xử lý hình ảnh và phần mềm Qt Designer để thiết kế giao diện giám sát, điều khiển hệ thống. Đồng thời, phương pháp thực nghiệm được áp dụng nhằm kiểm tra, đánh giá hiệu quả hoạt động của hệ thống theo yêu cầu công nghệ đặt ra.

3. KẾT QUẢ NGHIÊN CỨU

3.1. Sơ đồ khối hệ thống điều khiển

Máy tính nhận dữ liệu hình ảnh từ camera, thực hiện phân tích, xử lý ảnh, kết quả nhận được là màu sắc và vị trí của sản phẩm trên băng tải sẽ được truyền đến bộ điều khiển (khối vi điều khiển Arduino Mega 2560). Bộ điều khiển nhận tín hiệu điều khiển từ máy tính, cùng với tín hiệu từ các nút điều khiển, cảm biến (công tắc hành trình - CTHT) sẽ điều khiển các cơ cấu chấp

hành của hệ thống như van khí nén (đóng cắt giác hút chân không lắp ở bộ di động của robot) hay các động cơ của cánh tay robot delta. Động cơ truyền động cho băng tải được điều khiển qua mạch CNC Shield V4, mạch này cũng nhận lệnh điều khiển từ bộ điều khiển Arduino Mega 2560. Các công tắc hành trình giúp giới hạn vị trí ban đầu của robot. Sơ đồ khối điều khiển hệ thống phân loại sản phẩm ứng dụng thị giác máy được thể hiện trong hình 4.

3.2. Thuật toán xử lý ảnh phân loại bánh

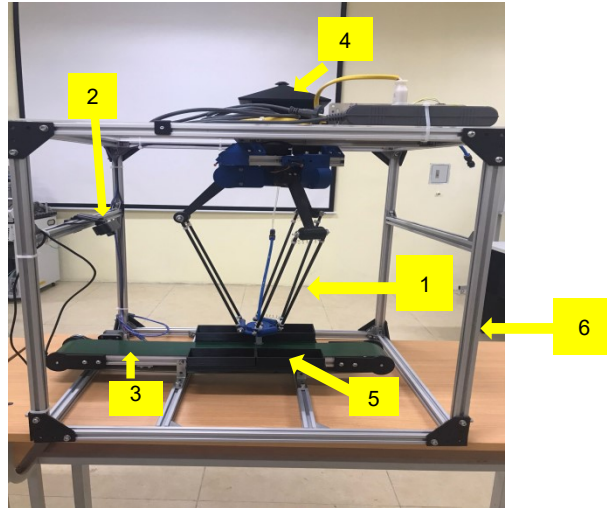
3.2.1. Ngưỡng giá trị HSV

Trong nghiên cứu, chúng tôi thực hiện phân loại bánh KitKat theo các màu sắc bao gói khác nhau, ví dụ bánh KitKat màu hồng và màu cam được thể hiện trên hình 5. Bánh có chiều dài 70mm, chiều rộng 30mm và bề dày là 7mm. Có thể nhận thấy, trên vỏ bánh, ngoài các màu chủ đạo (chiếm phần lớn diện tích), còn có chữ và hình ảnh với màu sắc khác nhau.

Màu sắc của bánh hiển thị ở hình 5 thuộc không gian màu RGB, đây là mô hình không gian màu dựa trên 3 màu sắc cơ bản là đỏ (Red), xanh lá cây (Green) và xanh dương (Blue). Mô hình này được sử dụng rộng rãi trong các thiết bị hiển thị như màn hình máy tính, TV, máy ảnh kỹ thuật số,... Nhược điểm của không gian màu này là khó phân tách màu và khó điều chỉnh độ sáng, độ bão hòa của ảnh, vì thế cần chuyển đổi ảnh RGB sang không gian màu HSV, một mô hình màu được sử dụng phổ biến trong xử lý ảnh và đồ họa máy tính do gần gũi với cách cảm nhận màu sắc của con người (Phạm Văn Nghĩa, 2022). Mô hình HSV bao gồm 3 thông số: H (Hue) mô tả màu sắc, S (Saturation) biểu diễn mức độ bão hòa màu và V (Value) đo lường độ sáng của màu sắc.

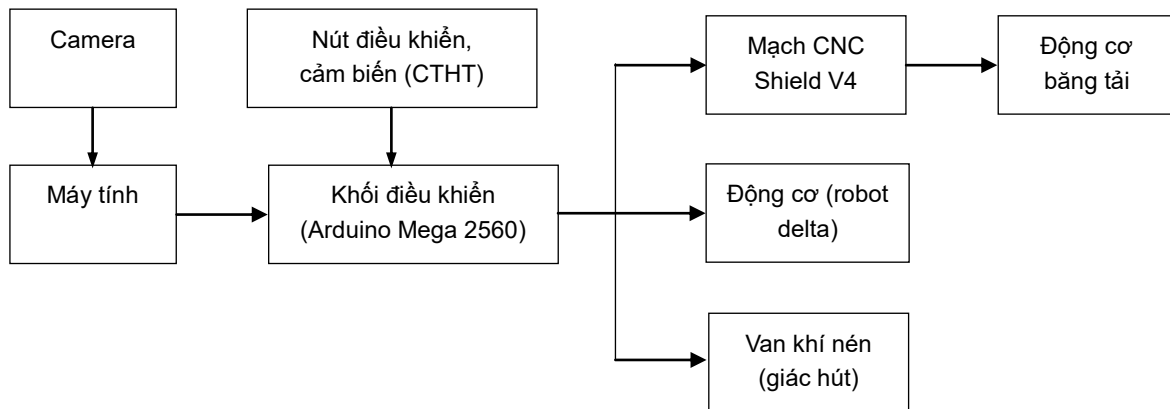
Bảng 1. Kích thước hình học của robot delta

Đặc điểm	Kích thước (mm)
Cánh tay chủ động	130
Cánh tay bị động	315
Tám đế cố định (bán kính)	85
Tám đế di động (bán kính)	45
Vùng làm việc (tối ưu)	200 × 200 × 200



Ghi chú: 1: Robot delta; 2: Camera; 3: Bảng tải; 4: Bộ điều khiển; 5: Khay chứa; 6: Khung nhôm.

Hình 3. Mô hình hệ thống phân loại sản phẩm sau khi lắp ráp



Hình 4. Sơ đồ khối điều khiển mô hình phân loại sản phẩm sử dụng thị giác máy

Để chuyển đổi màu từ không gian RGB sang không gian HSV, trước tiên cần xác định các hệ số M và m (Vo Duy Cong & Le Hoai Phuong, 2023; Vo Duy Cong & cs., 2023):

$$M = \max\{R, G, B\}; m = \min\{R, G, B\} \quad (1)$$

Các thông số trong không gian màu HSV

được xác định theo các công thức (2), (3) và (4).

Trong OpenCV, H có giá trị trong khoảng [0-179], giá trị của S và V tương ứng là [0-255] và [0-255] (Phạm Văn Nghĩa, 2022). Hàm cvtColor dùng để chuyển đổi từ không gian màu RGB sang không gian màu HSV (biểu thức 5).

$$H = \begin{cases} \cos^{-1} \frac{R - \frac{B+G}{2}}{\sqrt{R^2 + G^2 + B^2 - RG - GB - BR}} & \text{khi } G \geq B \\ 360^\circ - \cos^{-1} \frac{R - \frac{B+G}{2}}{\sqrt{R^2 + G^2 + B^2 - RG - GB - BR}} & \text{khi } G < B \end{cases} \quad (2)$$

$$V = \frac{M}{255} \quad (3)$$

$$S = \begin{cases} 1 - \frac{m}{M} & \text{khi } M > 0 \\ 0 & \text{khi } M = 0 \end{cases} \quad (4)$$

$$\text{cv::cvtColor}(Img_Src, Img_Dst, \text{cv::COLOR_BGR2HSV}) \quad (5)$$



(a) Màu hồng



(b) Màu cam

Hình 5. Bánh KitKat

Trong biểu thức (5): *Img_Src* - ảnh đầu vào, *Img_Dst* - ảnh đầu ra và *cv::COLOR_BGR2HSV* là chuyển đổi ảnh từ hệ RGB sang hệ HSV.

Từ ảnh hệ màu HSV thu được, thực hiện lọc ngưỡng ảnh sử dụng hàm *cv::inRange*, kết quả nhận được sẽ là một mặt nạ nhị phân của ảnh đầu vào. Hàm này kiểm tra từng điểm ảnh trong hình ảnh và xác định xem chúng có nằm trong phạm vi màu được chỉ định hay không. Những điểm ảnh nằm trong phạm vi sẽ được đặt giá trị 255 (trắng) trong mặt nạ nhị phân, còn những điểm ảnh nằm ngoài phạm vi sẽ được đặt giá trị 0 (đen).

cv::inRange(src, hsv_min, hsv_max, dst) (6)

Với *src* là ảnh đầu vào, *dst* là ảnh đầu ra, các giá trị *hsv_min* và *hsv_max* là giá trị ngưỡng hệ màu HSV được chỉ định. Kết quả lọc ngưỡng với loại bánh KitKat màu cam được thể hiện trong hình 6.

Trong hình 6b, vùng màu trắng chính là bánh và vùng màu đen là nền. Trên bề mặt bánh sau khi lọc ngưỡng vẫn xuất hiện những vùng đen, nguyên nhân là do trên bao bì bánh còn có nhiều màu sắc khác, cùng với ảnh hưởng độ bóng của bao bì dưới tác động của ánh sáng

môi trường. Kết quả xác định ngưỡng giá trị trong hệ màu HSV để phân biệt bánh KitKat màu cam là H [0-67], S [34-255], V [130-255]. Thực hiện tương tự với các bánh KitKat có màu khác nhau, ta xác định được ngưỡng HSV:

- Màu hồng: H [50-179], S [8-255], V [127-255]

- Màu đỏ: H [80-179], S [138-255], V [137-255]

- Màu xanh: H [13-179], S [108-255], V [144-255]

- Màu trắng: H [0-179], S [0-55], V [0-255].

3.2.2. Phân loại bánh theo màu sắc

Sau khi lọc ngưỡng với các khoảng giá trị màu HSV, ảnh được chuyển sang dạng nhị phân với 2 giá trị 0 (đen) và trắng (255) (Hình 6). Để xác định xem tại thời điểm *t* nào đó, bánh có đang xuất hiện trong ảnh không, ta cần xác định đường viền ảnh. Trong kỹ thuật xử lý ảnh số, quá trình này được thực hiện dựa trên tính không liên tục và tính đồng nhất của điểm ảnh. Thư viện OpenCV cung cấp hàm *cv::findContours* để phát hiện đường viền của vật trong ảnh, với cú pháp:

`cv::findContours(input, contoursContainer, mode, method)` (7)

Giá trị *input* là ảnh đầu vào, *contoursContainer* là biến chứa các đường viền được phát hiện, *mode* là chế độ truy xuất đường viền (có thể chọn: `RETR_EXTERNAL/RETR_LIST`, `RETR_CCOMP/RETR_TREE/RETR_FLOODFILL`) và *method* là phương pháp xấp xỉ đường viền (`CHAIN_APPROX_NONE` hoặc `CHAIN_APPROX_SIMPLE`) (Phạm Văn Nghĩa, 2022). Kết quả nhận được là đường viền của vật thể được lưu trong biến *contoursContainer*, từ đó có thể sử dụng biến này để phát hiện vật cũng như xác định vị trí của vật trên ảnh.

Hàm `cv::minAreaRect` sẽ trả về thông tin về chiều dài, chiều rộng và tọa độ tâm của một hình chữ nhật có diện tích nhỏ nhất bao quanh đối tượng. Hình 7 thể hiện bánh KitKat được vẽ đường viền chữ nhật và phân loại, gắn nhãn với KitKat 1 là bánh màu hồng và KitKat 2 là bánh màu cam.

3.3. Chuyển đổi hệ tọa độ ảnh - hệ tọa độ robot

Như đã trình bày ở mục 3.2.2, kết quả trả về của hàm `cv::minAreaRect` trong OpenCV là thông tin về kích thước dài, rộng và tọa độ tâm của bánh. Tuy nhiên, để điều khiển robot hút và

phân loại bánh, cần phải chuyển đổi hệ quy chiếu của bánh trên ảnh về cùng với hệ quy chiếu của robot (hệ tọa độ thực).

Để thực hiện việc chuyển đổi giữa hệ tọa độ ảnh với hệ tọa độ robot phải trải qua 2 bước (Justinas, 2016; Automatic, 2020; Ahmed & cs., 2024):

+ *Bước 1*: Trong hệ tọa độ ảnh, chuyển đổi từ đơn vị pixel (px) sang đơn vị dài (cm hoặc mm, ở đây chúng tôi sử dụng đơn vị mm).

Trong hệ tọa độ ảnh, lấy 2 điểm bất kỳ, xác định khoảng cách giữa chúng là *a* (px), khoảng cách ấy tương ứng trong thực tế là *b* (mm), tính hệ số tỉ lệ:

$$k = \frac{b}{a} \quad (8)$$

Tọa độ điểm ảnh ($x_{\text{ima_m}}$, $y_{\text{ima_m}}$) tính theo đơn vị millimeter được xác định:

$$\begin{cases} x_{\text{ima_m}} = x_{\text{ima_px}} \cdot k \\ y_{\text{ima_m}} = y_{\text{ima_px}} \cdot k \end{cases} \quad (9)$$

Trong đó: $x_{\text{ima_px}}$ và $y_{\text{ima_px}}$ là tọa độ của điểm trong hệ tọa độ ảnh, tính theo đơn vị pixel.

Hay có thể viết lại biểu thức (9) dưới dạng ma trận:

$$\begin{bmatrix} x_{\text{ima_m}} \\ y_{\text{ima_m}} \end{bmatrix} = k \begin{bmatrix} x_{\text{ima_px}} \\ y_{\text{ima_px}} \end{bmatrix} \quad (10)$$



(a) Ảnh đầu vào



(b) Ảnh đầu ra

Hình 6. Kết quả lọc ngưỡng bánh KitKat màu cam



(a) Ảnh đầu vào



(b) Ảnh đầu ra

Hình 7. Kết quả phân loại bánh với các đường viền được xác định và bánh được gắn nhãn

+ *Bước 2:* Chuyển đổi điểm ảnh (x_{ima_m}, y_{ima_m}) với đơn vị tính millimeter trong hệ tọa độ ảnh sang tương đương trong hệ tọa độ của robot bằng cách sử dụng ma trận biến đổi thuận nhất T (kích thước 4×4):

$$T = \begin{pmatrix} R_{ima}^{real} & d_{ima}^{real} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (11)$$

Trong đó: ma trận quay R_{ima}^{real} có kích thước 3×3 mô tả hướng của hệ tọa độ ảnh so với hệ tọa độ robot và d_{ima}^{real} là vector tịnh tiến giữa hai hệ trục tọa độ.

Để xác định ma trận quay R_{ima}^{real} , chú ý rằng hệ tọa độ cơ sở của robot đặt tại bệ cố định, trục Y_0 hướng dọc theo băng tải, trục X_0 hướng sang phải và trục Z_0 hướng lên trên. Tuy nhiên, để thuận tiện, ta tịnh tiến hệ tọa độ này dọc theo trục Z_0 , hướng theo chiều ngược lại, sao cho gốc tọa độ nằm trên băng tải (khi đó, sản phẩm phân loại nằm trên băng tải sẽ chỉ phụ thuộc vào tọa độ (x_0, y_0) , còn z_0 đã được xác định, có thể coi bằng 0). Hệ trục tọa độ ảnh (trong OpenCV) với các trục X_i, Y_i, Z_i được thể hiện trong mối tương quan với hệ tọa độ của robot như hình 8.

Có thể nhận thấy R_{ima}^{real} là ma trận quay

quanh trục X với góc quay $\alpha = 180^\circ$:

$$R_{ima}^{real} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix} \quad (12)$$

$$\text{Vector } d_{ima}^{real} = \begin{bmatrix} d_x \\ d_y \\ 0 \end{bmatrix} \text{ thể hiện độ dịch chuyển}$$

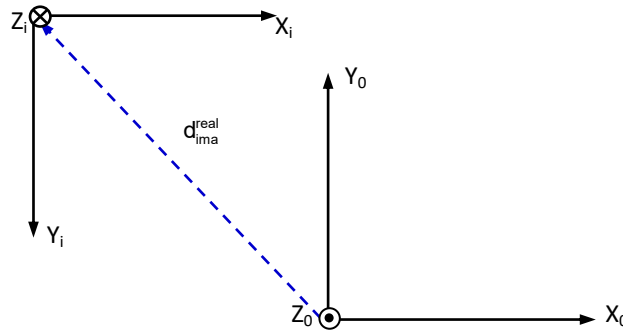
giữa hai hệ trục tọa độ được xác định trực tiếp bằng thực nghiệm với tọa độ điểm ảnh (x_{ima_m}, y_{ima_m}) và tọa độ tương ứng (x_{real}, y_{real}) của điểm đó trong hệ tọa độ robot:

$$\begin{cases} d_x = x_{real} - x_{ima_m} \\ d_y = y_{real} - y_{ima_m} \end{cases} \quad (13)$$

Kết quả tổng hợp từ biểu thức (12) và (13) có được:

$$T = \begin{pmatrix} 1 & 0 & 0 & d_x \\ 0 & \cos\alpha & -\sin\alpha & d_y \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (14)$$

Một cách tổng quát, nếu điểm A có tọa độ $(x_{ima_px}, y_{ima_px}, z_{ima_px})$ (px) trong hệ tọa độ ảnh, tọa độ tương ứng $(x_{real}, y_{real}, z_{real})$ (mm) của điểm đó trong hệ tọa độ robot:



Hình 8. Vị trí tương đối của hệ tọa độ ảnh và hệ tọa độ robot

$$\begin{bmatrix} x_{\text{real}} \\ y_{\text{real}} \\ z_{\text{real}} \\ 1 \end{bmatrix} = k \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & \cos \alpha & -\sin \alpha & d_y \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{\text{ima_px}} \\ y_{\text{ima_py}} \\ z_{\text{ima_pz}} \\ 1 \end{bmatrix} \quad (15)$$

3.4. Lưu đồ thuật toán điều khiển hệ thống

Kết quả của quá trình xử lý ảnh là thông tin về màu sắc và vị trí của bánh trên băng tải được gửi về bộ điều khiển, giúp cho robot delta nhận diện, thực hiện hút và phân loại bánh về đúng khay theo yêu cầu. Sơ đồ các bước của quá trình xử lý ảnh được thể hiện trên hình 9. Ảnh của bánh sau khi thu thập từ camera sẽ được chuyển đổi sang không gian ảnh HSV, sau đó chuyển sang ảnh nhị phân nhờ sử dụng ngưỡng giá trị HSV đã được xác định trước đó. Bằng việc xác định đường viền bánh, bánh được gán nhãn và thông tin về vị trí tọa độ tâm bánh sẽ được chuyển đổi sang hệ tọa độ robot giúp hệ thống thực hiện công đoạn phân loại bánh tiếp theo.

Khi bánh nằm trong vùng hoạt động của robot và số lượng bánh nằm trong giới hạn n_{max} (tổng số lượng bánh khảo nghiệm), bộ điều khiển sẽ gửi tín hiệu điều khiển để đồng bộ tốc độ của robot delta và băng tải, giúp robot chạy theo, hút và đưa bánh vào các khay. Lưu đồ thuật toán chương trình điều khiển hệ thống phân loại bánh KitKat theo màu hồng (KitKat 1) và màu cam (KitKat 2), với số lượng KitKat 1 ở khay chứa (khay 1 và khay 4) là $n_{1\text{max}}$, số lượng KitKat 2 ở khay chứa 2 và 3 là $n_{2\text{max}}$ được thể hiện ở hình

10 (chú ý rằng: $2n_{1\text{max}} + 2n_{2\text{max}} = n_{\text{max}}$). Để đơn giản, giả thiết số bánh mỗi loại không vượt quá khả năng chứa của các khay. Chương trình điều khiển được lập trình với Visual Studio Code. Các lệnh điều khiển được gửi từ giao diện phần mềm trên máy tính qua cổng serial kết nối với vi điều khiển Arduino Mega 2560.

3.5. Kết quả khảo nghiệm

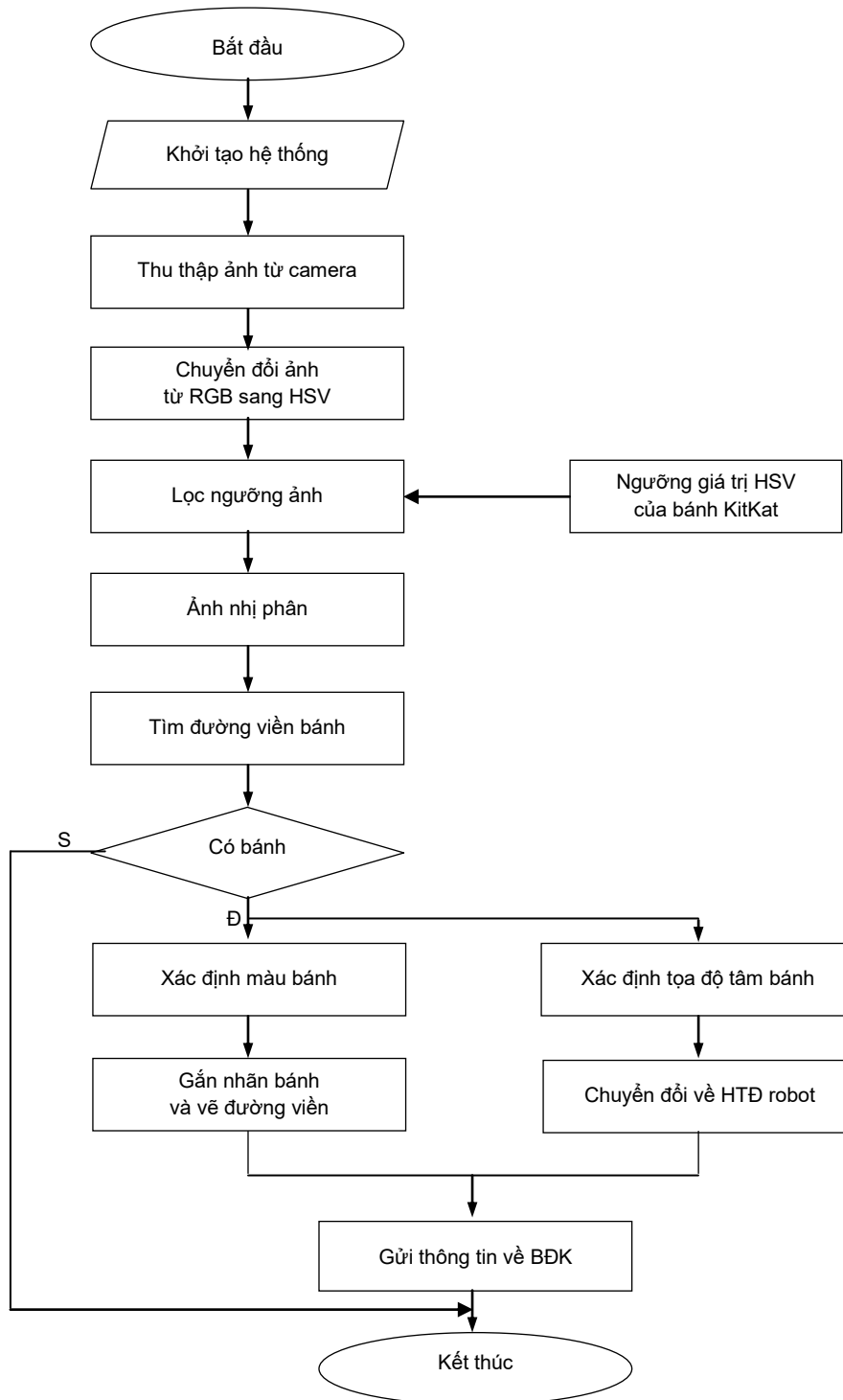
Thực hiện khảo nghiệm hệ thống với thuật toán đã xây dựng ở các tốc độ băng tải khác nhau 60 mm/s, 70 mm/s và 80 mm/s. Ở mỗi mức tốc độ của băng tải, bánh KitKat có màu ngẫu nhiên (hồng, cam) được đặt thủ công lên băng tải. Trên giao diện giám sát được thiết kế bằng phần mềm Qt Designer, tiến hành kết nối robot với máy tính và cấu hình camera nhằm đảm bảo hệ thống sẵn sàng hoạt động. Người dùng cũng có thể lựa chọn và điều chỉnh chương trình điều khiển thông qua nút GCode View trên giao diện.

Quá trình hoạt động của hệ thống được giám sát trên giao diện Qt Designer, với các thông tin hiển thị bao gồm số lượng bánh được camera nhận diện và số bánh mỗi loại theo màu sắc. Dựa trên dữ liệu xử lý ảnh, robot delta sẽ thực hiện hút bánh và phân loại vào các khay tương ứng. Trong mô hình, các khay đựng được đặt ở hai bên của băng tải, mỗi khay chứa được 12 bánh cùng màu (Hình 11). Kết quả khảo nghiệm hệ thống được thể hiện trong bảng 2.

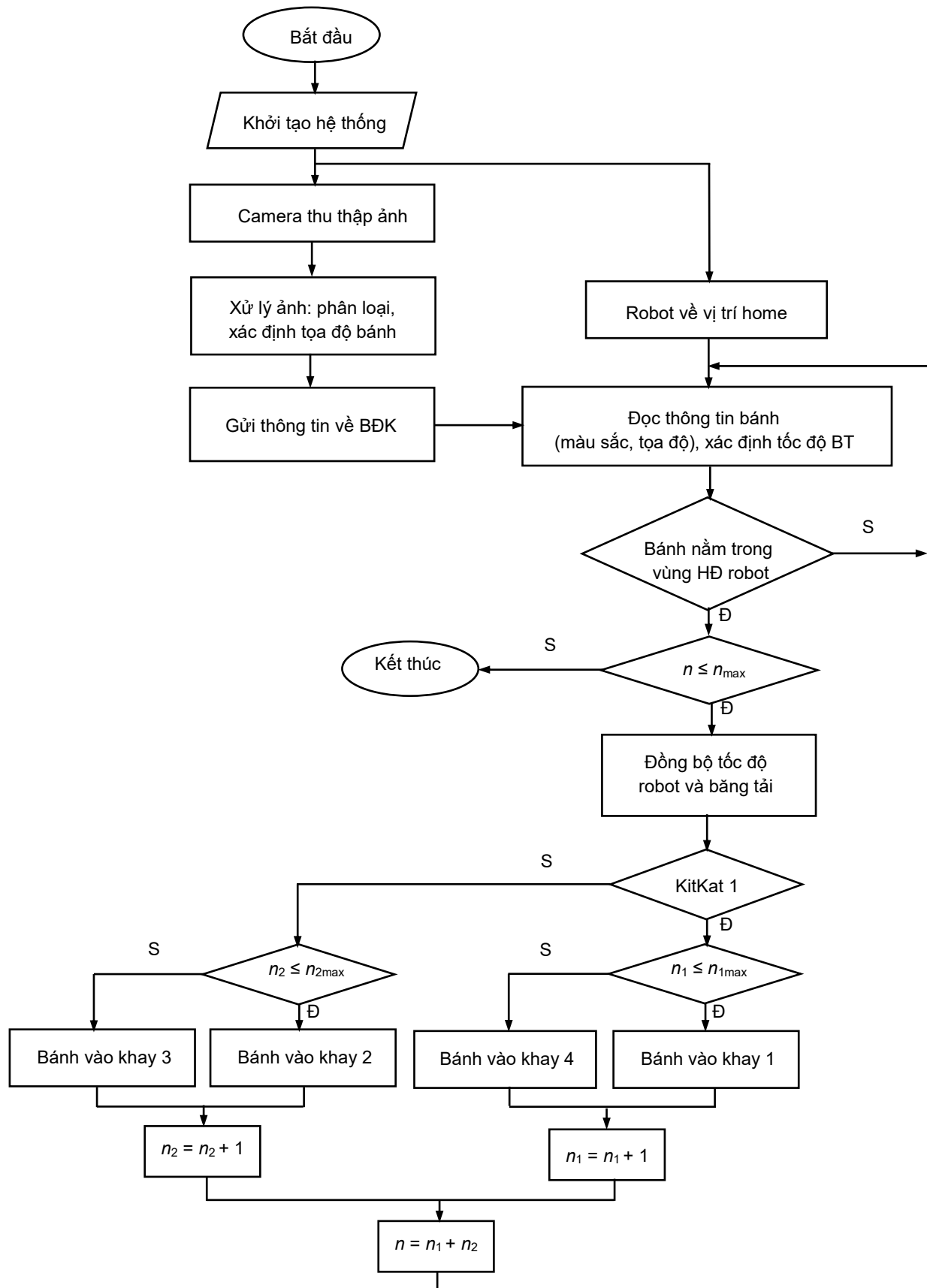
Kết quả khảo nghiệm trong bảng 2 cho thấy khi tốc độ băng tải là 60 mm/s, hệ thống thị giác máy và robot phân loại hoạt động tốt với độ

chính xác là 100%, thời gian trung bình để hệ thống thực hiện phân loại bánh là 3,5 s/sản phẩm. Ở các tốc độ băng tải 70 mm/s và 80 mm/s, hệ thống thị giác vẫn hoạt động chính xác, tuy nhiên khả năng robot delta hút bánh và

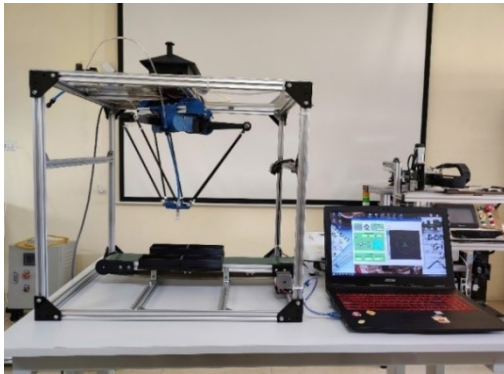
đưa vào đúng khay phân loại giảm, tương ứng với độ chính xác là 83,33% và 72,92%. Có thể lý giải cho sự hạn chế này là do tốc độ, độ chính xác và vùng hoạt động của robot còn hẹp, tốc độ xử lý của vi điều khiển chưa cao.



Hình 9. Lưu đồ các bước xử lý ảnh phân loại bánh theo màu sắc và xác định vị trí bánh trên băng tải



Hình 10. Lưu đồ thuật toán điều khiển hệ thống phân loại bánh



(a) Khảo nghiệm hệ thống



(b) Màn hình giám sát điều khiển

Hình 11. Khảo nghiệm hệ thống phân loại sản phẩm

Bảng 2. Kết quả khảo nghiệm mô hình

Tốc độ băng tải	Tổng số bánh	Xử lý ảnh chính xác	Hút và đưa bánh vào đúng vị trí
60 mm/s	48	48/48 (100%)	48/48 (100%)
70 mm/s	48	48/48 (100%)	40/48 (83,33%)
80 mm/s	48	48/48 (100%)	35/48 (72,92%)

Như vậy, có thể thấy rằng, thuật toán điều khiển robot delta kết hợp thị giác máy trong mô hình phân loại bánh KitKat hoạt động khá tốt ở tốc độ băng tải 60 mm/s. Tuy nhiên, tốc độ xử lý ảnh và thực hiện phân loại sản phẩm còn chưa cao (trung bình 3,5s/sản phẩm), số lượng sản phẩm và các màu sắc cần phân loại còn hạn chế. Trong tương lai, nghiên cứu cần tiếp tục mở rộng, nâng cấp và khảo nghiệm nhiều hơn để có thể có kết luận chính xác về tính hiệu quả của hệ thống.

4. KẾT LUẬN

Trong nghiên cứu này, chúng tôi đã xây dựng một thuật toán điều khiển robot delta tích hợp thị giác máy ứng dụng trong dây chuyền phân loại bánh KitKat theo màu sắc. Hệ thống thu nhận dữ liệu bánh nhờ một camera gắn trên khung nhôm, ở phía đầu băng tải. Quá trình phân tích, xử lý hình ảnh cho thông tin về màu sắc, vị trí của bánh trên băng chuyền, làm cơ sở cho việc điều khiển robot hút chính xác bánh cần phân loại, ngay cả khi băng tải vẫn đang hoạt động. Kết quả khảo nghiệm cho thấy, hệ

thống có thể hoạt động chính xác khi tốc độ băng tải 60 mm/s với thời gian trung bình để thực hiện xử lý, phân loại và hút sản phẩm vào khay là 3,5 s/sản phẩm. Tuy nhiên, để có thể áp dụng vào thực tiễn sản xuất, hệ thống cần tiếp tục nghiên cứu, hoàn thiện để tăng tốc độ hoạt động, mở rộng khả năng cũng như nâng cao độ chính xác phân loại, từ đó tăng năng suất của hệ thống.

TÀI LIỆU THAM KHẢO

- Ahmed Elassal, Mahmoud Abdelaal, Mohmad Osama & Hager Elhmidy (2024). Low-cost parallel delta robot for a pick-and-place application with the support of the vision system. *Advances in Electrical Engineering, Electronics and Energy*. doi.org/10.1016/j.prime.2024.100518.
- Automatic Addison (2020). How to Convert Camera Pixels to Robot Base Frame Coordinates. Retrieved from <https://automaticaddison.com/how-to-convert-camera-pixels-to-robot-base-frame-coordinates/> on Sep 10, 2024.
- Ho Vinh Nguyen, Vo Duy Cong & Phan Xuan Trung (2023). Development of a SCARA robot arm for palletizing applications based on computer vision. *FME Transactions*. 51(4): 541-549.

- Justinas Mišėikis, Kyrre Glette, Ole Jakob Elle & Jim Torresen (2016). Automatic Calibration of a Robot Manipulator and Multi 3D Camera System. Retrieved from <https://www.researchgate.net/publication/289587273> on Sep 10, 2024.
- Laribi M.A., Romdhane L. & Zeghloul S. (2007). Analysis and dimensional synthesis of the DELTA robot for a prescribed workspace. *Mechanism and Machine Theory*. 42(7): 859-870.
- Phạm Văn Nghĩa (2022). Không gian màu - Color space. MMLab UIT. Truy cập từ https://mmlab.uit.edu.vn/tutorials/cv/basic/color_space ngày 30/8/2024.
- Shehata M., Elshami M., Bai Q. & Zhao X. (2021). Parameter estimation for multibody system dynamic model of delta robot from experimental data. *IFAC-PapersOnLine* (3rd IFAC Conference on Modelling, Identification and Control of Nonlinear Systems MICNON). 54(14): 72-77.
- Vo Duy Cong & Le Hoai Phuong (2023). Design and development of a delta robot system to classify objects using image processing. *International Journal of Electrical and Computer Engineering*. 13(3): 2669-2676.
- Zhang H., Su T., Wu S., Zheng J. & Wang Y. (2018). Simultaneous path planning and trajectory optimization for high-speed sorting system. *International Journal of Advanced Robotic Systems*. 15(5): 1-13. doi: 10.1177/1729881418797870.