

XÂY DỰNG MÔ HÌNH ĐIỀU KHIỂN THIẾT BỊ ĐIỆN BẰNG CỬ CHỈ TAY ỨNG DỤNG THƯ VIỆN OPENCV-PYTHON

Dương Ngọc Đoàn¹ và Trịnh Hữu Cảnh²

¹ Khoa Điện - Điện tử, Trường Đại học Kỹ thuật - Công nghệ Cần Thơ,

² Sinh viên Khoa Điện - Điện tử, Trường Đại học Kỹ thuật - Công nghệ Cần Thơ,

Email: dndoan@ctu.edu.vn

Thông tin chung

Ngày nhận bài:

22/5/2025

Ngày nhận bài sửa:

12/9/2025

Ngày duyệt đăng:

25/9/2025

Từ khóa: Hand Detector, nhận dạng cử chỉ, OpenCV, Python, xử lý ảnh

TÓM TẮT

Ngày nay, việc ứng dụng trí tuệ nhân tạo, công nghệ xử lý ảnh đã giúp phát triển phương thức giao tiếp giữa con người và thiết bị theo hướng trực quan, sinh động. Nghiên cứu này thực hiện xây dựng mô hình điều khiển thiết bị điện bằng cử chỉ tay bằng cách kết hợp thư viện OpenCV-Python, mediapipe với bo mạch vi điều khiển Arduino để điều khiển các thiết bị ngoại vi. Thông qua camera của máy tính, hệ thống ứng dụng thư viện xử lý hình ảnh OpenCV-Python, mediapipe sẽ theo dõi cử chỉ tay của người dùng. Trong điều kiện ánh sáng đủ tốt, hệ thống được thử nghiệm và có thể nhận dạng tốt các cử chỉ tay trong khoảng cách 100 cm và thực hiện các lệnh đã cài đặt để điều khiển Arduino bật tắt đèn, quạt, đóng mở cửa theo mục tiêu đặt ra.

1. ĐẶT VẤN ĐỀ

Nhiều phương thức tương tác giữa con người và máy tính đã được nghiên cứu, phát triển thay cho phương thức giao tiếp thông qua giao diện dòng lệnh hoặc giao diện đồ họa (GUI). Một trong những phương pháp thú vị là ứng dụng công nghệ xử lý ảnh, thị giác máy tính để máy tính nhận dạng cử chỉ cơ thể của con người giúp máy tính hiểu được ngôn ngữ cơ thể của con người. Từ đó, người dùng có thể đưa ra các lệnh điều khiển cho máy tính thông qua ngôn ngữ cơ thể.

Hiện nay, việc điều khiển các thiết bị điện, điện tử trong gia đình bằng các phương thức tương tác tự nhiên, cụ thể như tương tác bằng cử chỉ tay là giải pháp điều khiển trực quan và được áp dụng ngày càng phổ biến. Công nghệ này mang lại sự tiện nghi, sang trọng cho ngôi nhà và mang đến trải nghiệm thú vị, thân thiện và trực quan cho người dùng.

Những ứng dụng về thị giác máy tính, nhận dạng cử chỉ cơ thể đã được giới thiệu ở các nghiên cứu, ứng dụng cho các mục đích

khác nhau. Ahmad và cộng sự đã ứng dụng thư viện OpenCV để nhận dạng bàn tay và một số cử chỉ tay [1]. Tuy nhiên, nghiên cứu này dừng lại ở việc nhận dạng được cử chỉ tay mà chưa có ứng dụng cụ thể. Akhilchandran và cộng sự sử dụng mô hình mạng nơ-ron tích chập (Convolutional Neural Network) để huấn luyện, nhận dạng 10 cử chỉ tay khác nhau và kết hợp với Arduino để điều khiển của robot đi tới, quay trái, quay phải ... [2]. Jedidiah và cộng sự ứng dụng thư viện OpenCV để nhận dạng bàn tay nắm hoặc mở trong việc điều khiển cánh tay robot thông qua bo mạch Raspberry Pi [3]. Trần và cộng sự thiết kế robot bám theo người sử dụng thuật toán nhận diện tư thế cơ thể Human pose estimation để ứng dụng trong nông nghiệp [4]. Một nghiên cứu khác ứng dụng mô hình YOLO của Ultralytics để nhận dạng đối tượng, phân đoạn đối tượng, theo dõi, ước tính cử động của đối tượng để cải thiện độ chính xác trong việc xác định vật cản cho drone [5].

Các nghiên cứu kết hợp giữa kiến thức lý thuyết, kỹ năng thực hành, kết hợp nội dung

các học phần trong chương trình giảng dạy nhằm thiết kế, chế tạo các sản phẩm phục vụ cho hoạt động giảng dạy, học tập cũng được quan tâm thực hiện [6]. Các công nghệ mới được ứng dụng đã mang lại lợi ích rất lớn cho hoạt động giảng dạy, học tập. Mục tiêu của nghiên cứu này là tìm hiểu, ứng dụng công nghệ xử lý ảnh [7] để nhận dạng cử chỉ tay nhằm xây dựng mô hình đơn giản điều khiển thiết bị điện bằng cử chỉ tay phục vụ cho hoạt động giảng dạy, học tập. Bài báo trình bày cụ thể việc xây dựng hệ thống, thu thập dữ liệu xử lý hình ảnh đến việc nhận dạng cử chỉ của các ngón tay thông qua việc sử dụng thư viện OpenCV-Python, mediapipe, thư viện mã nguồn mở cho xử lý hình ảnh và thị giác máy tính để gửi lệnh điều khiển đến bo mạch vi điều khiển Arduino thực hiện điều khiển các thiết bị ngoại vi như đèn, quạt.

2. PHƯƠNG PHÁP NGHIÊN CỨU

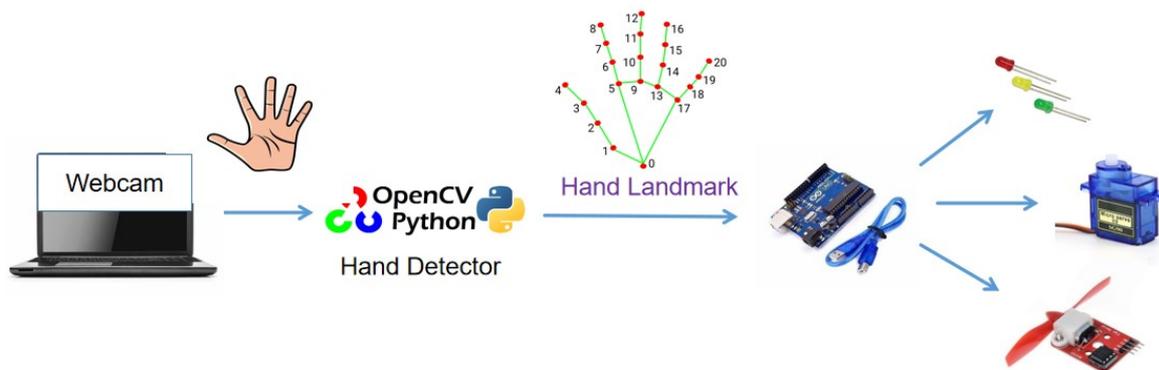
2.1. Tổng quan mô hình nghiên cứu

Mô hình nghiên cứu được xây dựng bao gồm các thành phần chính:

- Máy tính sử dụng thư viện OpenCV, sử dụng camera để thu thập hình ảnh cho quá trình điều khiển và chương trình điều khiển được lập trình bằng ngôn ngữ lập trình Python.

- Dựa trên hình ảnh thu nhận được, máy tính sẽ tiến hành xử lý để nhận dạng cử chỉ tay. Kết quả nhận dạng cử chỉ tay từ máy tính sẽ được gửi đến Arduino dưới dạng các lệnh điều khiển.

Bo mạch vi điều khiển Arduino kết nối với máy tính qua cổng USB thực hiện điều khiển các thiết bị ngoại vi như đèn LED, động cơ Servo SG90, động cơ DC L9110 được minh họa ở Hình 1.



Hình 1. Sơ đồ tổng quát của hệ thống

2.2. Phần mềm, thư viện cần thiết

Chương trình điều khiển các thiết bị điện được lập trình bằng ngôn ngữ Python trên môi trường học tập và phát triển tích hợp (IDLE) Python 3.10.11 và sử dụng các ứng dụng, thư viện gồm OpenCV-Python, Mediapipe, Cvzone, PyFirmata.

Trên máy tính sử dụng hệ điều hành Windows, từ cửa lệnh Command Promt, người dùng có thể cài đặt các thư viện trên

bằng các lệnh dưới đây. Khi thực hiện cài đặt các thư viện này, các thư viện phụ thuộc cũng sẽ được tự động cài đặt.

```
py -m pip install opencv-python
py -m pip install mediapipe
py -m pip install cvzone
py -m pip install pyfirmata
```

OpenCV (Open-Source Computer Vision Library) là thư viện mở về thị giác máy tính

và máy học được thiết kế để cung cấp, hỗ trợ rất nhiều giải thuật cho các ứng dụng về nhận thức máy tính. OpenCV hỗ trợ nhiều ngôn ngữ lập trình khác nhau bao gồm C++, C, Python và Java và hỗ trợ trên các hệ điều hành khác nhau [8].

OpenCV-Python là giao diện lập trình ứng dụng (API) Python dành cho OpenCV, kết hợp chất lượng tối ưu của OpenCV C++ API và ngôn ngữ lập trình Python, dùng để giải quyết các vấn đề về thị giác máy tính [8], [9].

Mediapipe là thư viện mở về máy học cung cấp giải pháp cho việc nhận dạng gương mặt và cử chỉ cho ngôn ngữ lập trình python, js cũng như nhiều ngôn ngữ lập trình khác. MediaPipe Hand là giải pháp giúp theo dõi dấu vết của bàn tay và ngón tay một cách hiệu quả bằng cách ứng dụng máy học để xác định 21 điểm mốc trên tay, bao gồm các khớp ngón tay và cổ tay, xác định tay trái hoặc tay phải để nhận dạng các cử chỉ tay khác nhau [10].

Cvzone là thư viện về thị giác máy tính giúp đơn giản hóa công việc xử lý ảnh và chức năng về trí tuệ nhân tạo. Cvzone sử dụng thư viện Open CV và Mediapipe [11].

PyFirmata là thư viện Python dựa trên giao thức Firmata được dùng để giao tiếp giữa máy tính và bo mạch vi điều khiển Arduino, cho phép người dùng điều khiển các ngõ vào, ngõ ra (IO pins) của Arduino từ chương trình trên máy tính [12].

2.3. Thu nhận và xử lý ảnh

Trong nghiên cứu này, hệ thống được lập trình để nhận dạng trạng thái cử chỉ tay và thực hiện điều khiển bật tắt các thiết bị điện. Khi nhận dạng người dùng thực hiện các cử chỉ tay như mô tả ở bảng 1, máy tính sẽ đưa ra các lệnh điều khiển tương ứng và truyền đến Arduino để thực hiện điều khiển các thiết bị được kết nối (Hình 1).

Mỗi ngón tay có thể thực hiện 2 trạng thái cử chỉ: gập lại (tương ứng với mã nhị phân 0) hoặc duỗi ra (tương ứng với mã nhị phân 1). Với 5 ngón tay, chúng ta có thể tổ hợp thành $2^5 = 32$ trạng thái khác nhau. Hệ thống sẽ nhận dạng trạng thái của từng ngón tay và biểu diễn trạng thái của cả 5 ngón tay bằng chuỗi số nhị phân gồm 5 bit.

Bảng 1: Bảng quy ước cử chỉ tay thông qua mã nhị phân

Thứ tự cử chỉ tay	Cử chỉ tay	Nhận lệnh	Mã nhị phân	Ý nghĩa điều khiển
1	Cả bàn tay gập lại	“OFF”	[0,0,0,0,0]	Tắt tất cả thiết bị
2	Ngón tay thứ nhất duỗi ra	“Spin”	[0,1,0,0,0]	Mở động cơ DC L9110 kèm quạt chạy
3	Ngón tay thứ nhất và thứ hai duỗi ra	“Stop Spin”	[0,1,1,0,0]	Tắt động cơ DC L9110 kèm quạt chạy
4	Ngón tay thứ nhất, thứ hai và thứ ba duỗi ra	“LED”	[0,1,1,1,0]	Mở đèn LED
5	Ngón tay thứ nhất, thứ hai, thứ ba và thứ tư duỗi ra	“Open”	[0,1,1,1,1]	Mở cửa
6	Cả năm ngón tay duỗi ra	“Close”	[1,1,1,1,1]	Đóng cửa và tắt đèn LED

Để thực hiện việc nhận dạng cử chỉ tay thu được từ webcam và gửi tín hiệu điều khiển đến Arduino, hệ thống sẽ tiến hành ghi nhận hình ảnh từ webcam của máy tính và tiến hành phân tích theo các bước như sau:

Bước 1: Thu video trực tiếp từ webcam.

```
video=cv2.VideoCapture(0)
```

Bước 2: Lấy các khung ảnh từ video đầu vào để tiến hành xử lý ở bước tiếp theo.

```
frame=video.read()
```

Bước 3: Thư viện HandDetector sẽ tạo ra một đối tượng để phát hiện bàn tay, đối tượng này sẽ sử dụng các thuật toán để nhận dạng và theo dõi bàn tay thông qua hình ảnh thu nhận được ở Bước 1 và Bước 2. Hệ thống được cài đặt các thông số để thực hiện nhận dạng các cử chỉ tay của 1 bàn tay và các cử chỉ tay được nhận dạng với độ chính xác từ 80% trở lên trong từng khung ảnh mới được coi là hợp lệ và đưa ra lệnh để điều khiển thiết bị.

```
detector=HandDetector(detectionCon=0.8,maxHands=1)
```

Bước 4: Tạo khung hình để theo dõi và khoanh vùng bàn tay, xử lý các hình ảnh thu được từ video. Trong mỗi khung hình, nếu phát hiện bàn tay thì hệ thống sẽ đánh dấu các điểm Hand Landmark để nhận dạng cử chỉ.

```
frame=cv2.flip(frame,1)
```

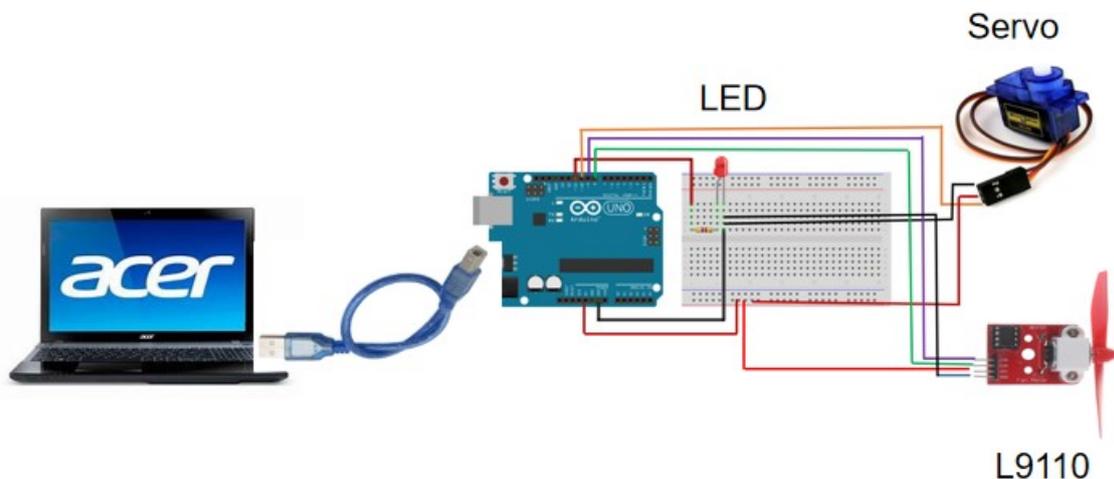
```
hands,img=detector.findHands(frame)
```

Bước 5: Tạo nét vẽ liên kết các điểm Hand Landmark được nhận dạng và thể hiện lên khung hình kèm với câu lệnh điều khiển thiết bị tương ứng.

```
cv2.putText(frame,'OFF:0',(20,460),cv2.FONT_HERSHEY_COMPLEX,1,(255,255,255),1,cv2.LINE_AA)
```

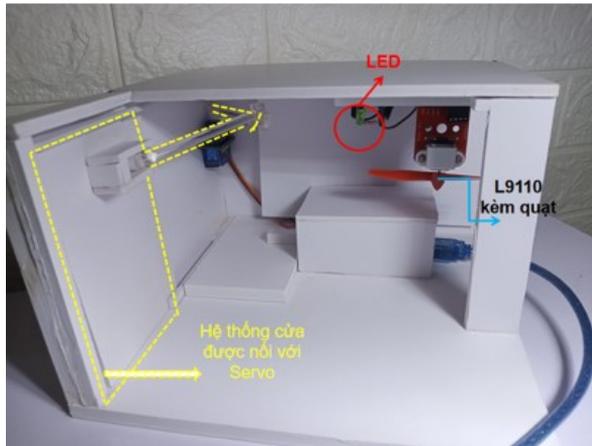
Bước 6: Xuất ra các thông số nhận dạng cử chỉ tay ở Bước 5 như đã cài đặt ở Bảng 1 để thực hiện các lệnh điều khiển thiết bị. Các mã nhị phân [0,0,0,0,0] đại diện cho 5 ngón tay đang gập và [1,1,1,1,1] đại diện cho 5 ngón tay đang duỗi.

Để đánh giá việc ứng dụng kết quả nhận dạng cử chỉ tay từ thư viện OpenCV-Python, kết quả nhận dạng được sử dụng để điều khiển các thiết bị ngoại vi thông qua bo mạch Arduino. Thiết kế phần cứng của hệ thống điều khiển thiết bị điện bằng cử chỉ tay được mô tả ở Hình 2. Phần cứng của hệ thống gồm webcam của máy tính để thu nhận hình ảnh đầu vào. Sau đó, hình ảnh đầu vào được tách thành các khung ảnh để nhận dạng cử chỉ tay và gửi các lệnh điều khiển đến Arduino. Dựa trên các lệnh điều khiển nhận được từ máy tính, Arduino sẽ điều khiển thiết bị theo cử chỉ tay tương ứng đã được lập trình.



Hình 2. Sơ đồ kết nối phần cứng của hệ thống

2.4. Thiết kế mô hình phần cứng



Hình 3. Mô hình thực tế của hệ thống điều khiển

Mô hình thực tế của hệ thống điều khiển được xây dựng như ở Hình 3. Tín hiệu truyền từ máy tính đến Arduino được gửi đi thông qua kết nối USB. Nguồn điện sử dụng cho các thiết bị ngoại vi trong mô hình này được cung cấp từ cổng USB.

Sau khi đã cài đặt các thư viện cần thiết trên máy tính, để thực hiện điều khiển Arduino từ máy tính với ngôn ngữ Python và pyFirmata, chúng ta thực hiện các bước cài đặt, kết nối như sau:

- Kết nối Arduino với máy tính qua cổng USB, nạp chương trình StandardFirmata cho Arduino bằng trình phát triển Arduino IDE.

- Lúc này, pyFirmata sẽ khởi tạo giao tiếp nối tiếp (Serial communication) giữa máy tính và Arduino để truyền các lệnh điều khiển từ máy tính đến Arduino thông qua cổng USB.

- Viết chương trình điều khiển Arduino bằng ngôn ngữ Python với pyFirmata.

3. KẾT QUẢ NGHIÊN CỨU/ THẢO LUẬN

Hệ thống điều khiển thiết bị điện thông qua cử chỉ tay bằng cách ứng dụng công nghệ xử lý ảnh với OpenCV-Python được

xây dựng như minh họa ở Hình 3. Hệ thống thiết kế đã được thử nghiệm để đánh giá độ chính xác trong việc nhận dạng cử chỉ tay bằng máy tính và truyền lệnh điều khiển đến Arduino để thực hiện điều khiển thiết bị điện trong nhiều trường hợp khác nhau.

Mô hình sử dụng nguồn điện được cung cấp trực tiếp từ máy tính và dữ liệu được truyền từ máy tính đến Arduino thông qua dây cáp USB để điều khiển để bật, tắt thiết bị điện với tín hiệu điều khiển là các cử chỉ tay đã được cài đặt. Qua quá trình thử nghiệm và đánh giá nhiều lần, trong điều kiện ánh sáng bình thường với khoảng cách từ bàn tay đến camera của máy tính khoảng 100 cm. Mô hình được thiết lập có khả năng nhận dạng tốt các trạng thái cử chỉ khác nhau của các ngón tay nói chung và 6 cử chỉ tay được xác định ở bảng 1 nói riêng để đưa ra các lệnh điều khiển đến Arduino để điều khiển các thiết bị điện đúng theo yêu cầu thiết kế (Bảng 2).

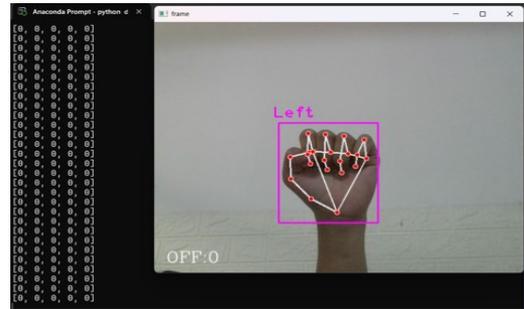
Bảng 2: Thử nghiệm và kết quả nhận dạng cử chỉ tay

Cử chỉ tay thử nghiệm	Số lần thử nghiệm	Số lần nhận dạng đúng	Tỷ lệ nhận dạng đúng (%)
Cả bàn tay gập lại	50	50	100
Ngón tay thứ nhất duỗi ra	50	50	100
Ngón tay thứ nhất và thứ hai duỗi ra	50	50	100
Ngón tay thứ nhất, thứ hai và thứ ba duỗi ra	50	48	96
Ngón tay thứ nhất, thứ hai, thứ ba và thứ tư duỗi ra	50	47	94
Cả năm ngón tay duỗi ra	50	47	94

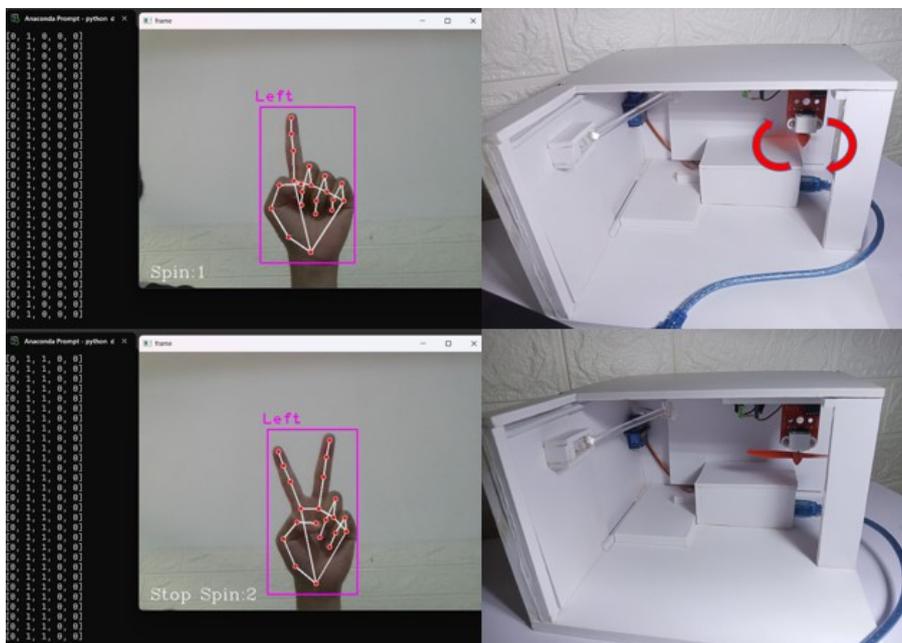
Qua thử nghiệm, khi thực hiện các cử chỉ với cả trường hợp lòng bàn tay hoặc lưng bàn tay hướng trực diện vào webcam thì hệ thống nhận dạng đúng các cử chỉ tay. Khi bàn tay được đưa theo hướng nghiêng so với góc nhìn của webcam thì hệ thống có thể có kết quả nhận dạng sai. Thời gian thực hiện nhận dạng cử chỉ tay được thực hiện nhanh chóng với máy tính có cấu hình phổ biến hiện nay.

Hình 4 minh họa cử chỉ cả 5 ngón tay gấp lại. Hệ thống nhận dạng chính xác trạng thái này và đưa ra lệnh điều khiển “OFF” tương ứng với mã nhị phân [0,0,0,0,0] gửi đến Arduino để tắt tất cả thiết bị.

Hình 5 minh họa cử chỉ ngón tay thứ hai duỗi ra tương ứng với lệnh điều khiển “Spin” được gửi đến Arduino mở động cơ DC và 2 ngón tay thứ hai và thứ ba cùng duỗi ra tương ứng với lệnh điều khiển “Stop Spin” được gửi đến Arduino để tắt động cơ DC.



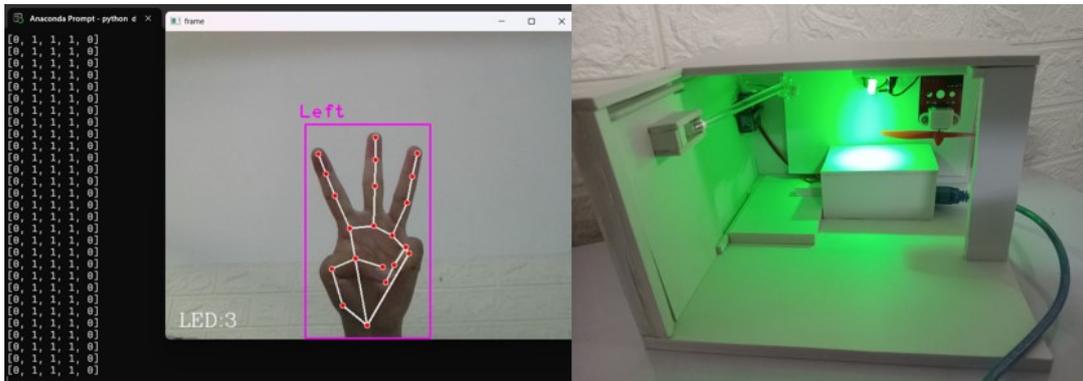
Hình 4. Nhận dạng cử chỉ cả bàn tay gấp lại tương ứng với lệnh “off”



Hình 5. Nhận dạng cử chỉ tay thứ hai tương ứng với lệnh “Spin” và cử chỉ tay thứ ba tương ứng với lệnh “Stop Spin”

Hình 6 minh họa hệ thống thực hiện bật đèn LED khi nhận dạng được cử chỉ tay tương ứng với lệnh điều khiển bật đèn led và giá trị trả về [0,1,1,1,0] như xác định ở bảng 1. Các

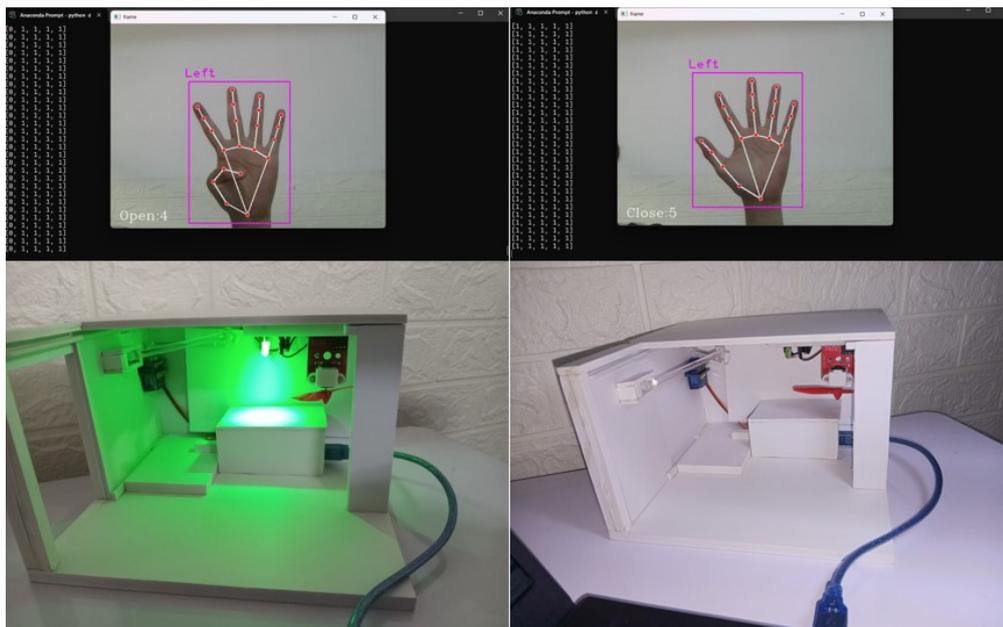
thiết bị có thể hoạt động độc lập với nhau mà không bị ảnh hưởng bởi các cử chỉ khác nhau giúp tối ưu phần điều khiển mà không gây bất tiện gì.



Hình 6. Nhận dạng cử chỉ tay thứ tư tương ứng với lệnh “LED”

Hình 7 (hình bên trái) minh họa hệ thống thực hiện mở cửa thông qua động cơ Servo SG90 khi nhận dạng cử chỉ tay tương ứng với lệnh điều khiển “Open” với tín hiệu nhị phân là

$[0,1,1,1,1]$ tương ứng với 4 ngón tay đang duỗi. Khi nhận dạng 5 ngón tay cùng duỗi tương ứng tín hiệu điều khiển $[1,1, 1,1,1]$ thì hệ thống tiến hành tắt tất cả thiết bị (hình bên phải).

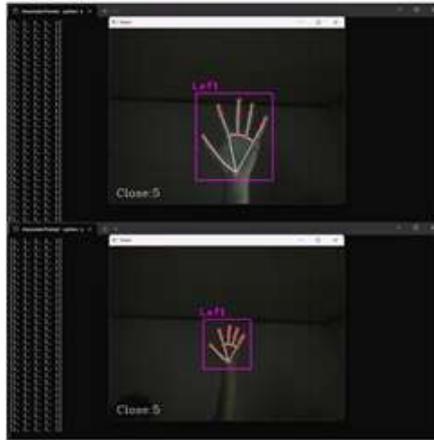


Hình 7. Nhận dạng cử chỉ tay thứ năm tương ứng với lệnh “Open” và cử chỉ tay thứ sáu tương ứng với lệnh “Close”

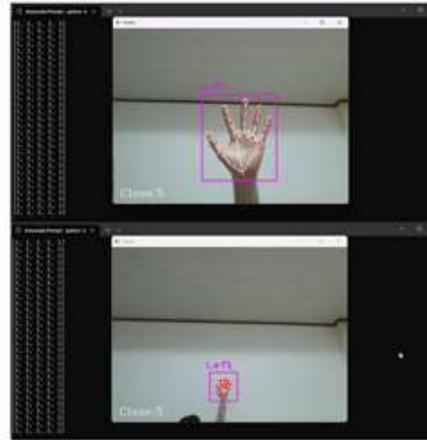
Việc ứng dụng thị giác máy tính, thư viện OpenCV đã được thực hiện trong nhiều nghiên cứu với các mục đích, ứng dụng khác nhau để điều khiển robot chuyển hướng, điều khiển cánh tay robot cầm nắm đồ vật ... Nghiên cứu này chế tạo mô hình đơn giản có thể nhận dạng được trạng thái gấp hoặc duỗi của 5 ngón tay tương ứng với 32 cử chỉ tay

khác nhau và được ứng dụng để điều khiển các thiết bị điện phục vụ việc giảng dạy, học tập, giới thiệu đến sinh viên việc ứng dụng kiến thức, công nghệ mới vào thực tế cuộc sống. Qua thử nghiệm thực tế, mô hình có thể nhận dạng chính xác các cử chỉ tay khác nhau của người dùng và 6 cử chỉ tay được trình bày

ở bảng 1 để điều khiển các thiết bị điện theo thiết kế.



a)



b)

Hình 8. Điều kiện ánh sáng khác nhau và khoảng cách khác nhau có ảnh hưởng đến độ chính xác của việc nhận dạng cử chỉ tay

Điều kiện ánh sáng có ảnh hưởng đến việc nhận dạng cử chỉ tay trong OpenCV-Python. Ánh sáng không đủ sáng hoặc quá sáng có thể làm giảm độ chính xác của việc phát hiện và nhận dạng cử chỉ tay, dẫn đến việc điều khiển thiết bị không chính xác. Đối với hệ thống thiết kế, trong điều kiện ánh sáng kém, hệ thống có thể nhận dạng cử chỉ tay và thực hiện điều khiển thiết bị chính xác trong khoảng cách tiêu chuẩn là 50 cm, khung ảnh và các đường liên kết giữa các điểm vẫn theo dõi tốt cử chỉ tay của người điều khiển. Trong điều kiện ánh sáng như Hình 8a thì hệ thống chỉ nhận dạng khung xương Hand Landmark trong khoảng cách tối đa khoảng 120 cm. Trong điều kiện ánh sáng tốt như ở Hình 8b thì hệ thống nhận dạng các cử chỉ tay trong khoảng cách 300 cm với khoảng 50% số lần thử nghiệm hệ thống thực hiện đúng và không phân biệt được cử chỉ phức tạp và nhỏ.

4. KẾT LUẬN

Nghiên cứu này đã ứng dụng thư viện xử lý ảnh OpenCV-Python, Cvzone, Mediapipe để nhận dạng cử chỉ tay, từ đó xây dựng mô hình sử dụng máy tính kết hợp với Arduino để điều khiển các thiết bị điện thông qua cử chỉ

3.2. Ảnh hưởng của điều kiện ánh sáng đến nhận dạng cử chỉ tay

của bàn tay. Mô hình được xây dựng gồm hệ thống cửa, đèn led và động cơ quạt hoạt động tốt theo từng cử chỉ tay mà hệ thống đã lập trình. Mô hình này có thể được sử dụng trong việc giảng dạy cho sinh viên các nội dung về kỹ thuật lập trình, vi điều khiển, thị giác máy tính, trí tuệ nhân tạo. Bên cạnh đó, nội dung nghiên cứu này có thể ứng dụng cho mô hình nhà thông minh sử dụng công nghệ xử lý ảnh để điều khiển các thiết bị điện có công suất lớn hơn trong gia đình nhằm ứng dụng công nghệ, mang lại sự tiện nghi trong thực tế cuộc sống hoặc có thể ứng dụng trong nhiều lĩnh vực khác nhau.

Để phát triển hệ thống, nâng cao tính tiện lợi, giảm chi phí thiết bị, có thể nghiên cứu sử dụng bo mạch Raspberry Pi có kích thước nhỏ gọn để thực hiện việc nhận dạng hình ảnh, cử chỉ tay để đưa ra lệnh điều khiển. Ngoài ra, việc kết nối điều khiển thiết bị ngoại vi cũng có thể được thực hiện trực tiếp trên bo mạch Raspberry Pi. Lúc này, Raspberry Pi có thể được dùng là máy tính và vi điều khiển để kết nối trực tiếp với các thiết bị ngoại vi mà không cần thêm bo mạch Arduino.

Hệ thống vẫn còn hạn chế khi hoạt động trong điều kiện thiếu ánh sáng, có thể dẫn đến

việc hệ thống nhận dạng không chính xác. Chất lượng của camera cũng là vấn đề cần quan tâm trong các ứng dụng về thị giác máy tính. Sử dụng camera có chất lượng tốt, độ phân giải cao thì hệ thống sẽ nhận dạng tốt hơn, từ đó khoảng cách nhận dạng có thể được tăng lên và nhận dạng chính xác các hành động, cử chỉ.

Tài liệu tham khảo

[1] Ahmad P. I, Farah A. A. A, Nazirah M. K et al. Hand Gesture Recognition on Python and OpenCV. IOP Conference Series: Materials Science and Engineering. 2021; 1045 (1): 012043.

[2] Akhilchandran N., Shejna N.M. Hand Gesture Controlled Robot Using OpenCV. International Journal for Scientific Research & Development. 2020; 8(5): 310-314.

[3] Jedidiah P., and Ahmed A. Gesture-Controlled Robotic Arm Utilizing OpenCV. 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA). 2021; Ankara, Turkey.

[4] Trần Lê Trung Chánh, Nguyễn Quốc Lĩnh, Nguyễn Chí Nguyễn và cộng sự. Điều khiển robot mang chậu cây bonsai mini ứng dụng trong nông nghiệp. Tạp chí Khoa học và Công nghệ Cần Thơ, 2024; 4:12-22.

[5] Nguyễn Phương Nam, Hứa Ngọc Nhi và Nguyễn Đình Tứ. Phát triển phương pháp

xác định vật cản cho drone sử dụng YOLOv8. Tạp chí Khoa học và Công nghệ Cần Thơ. 2024; 2: 31-39.

[6] Đường Khánh Sơn, Huỳnh Minh Vũ, Trần Sĩ Lâm và cộng sự. Thiết kế bộ điều khiển cho mô hình xe máy tự cân bằng phục vụ học tập và giảng dạy. Tạp chí Khoa học và Công nghệ Cần Thơ. 2024; 4: 3-11.

[7] Rafael C. G., Richard E. W. Digital Image Processing. 4th Edition, Pearson, London; 2018.

[8] OpenCV (2025). OpenCV modules. <https://docs.opencv.org/4.11.0/>, truy cập 06/6/2025.

[9] Alexander M., Abid R. K. (2025). Introduction to OpenCV-Python Tutorials. https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html, truy cập 06/6/2025.

[10] Techtutorialsx (2021). Python: Hand Landmark Estimation with MediaPipe. <https://techtutorialsx.com/2021/04/10/python-hand-landmark-estimation/>, truy cập 06/06/2025.

[11] Computer Vision Zone (2025). One Stop Computer Vision. <https://www.computervision.zone/>, truy cập 06/6/2025.

[12] Pipy.org (2025). Project description. <https://pypi.org/project/pyFirmata/>, truy cập 06/6/2025.

BUILDING A MODEL TO CONTROL ELECTRIC DEVICES BY HAND GESTURES USING OPENCV-PYTHON LIBRARY

ABSTRACT

Nowadays, the application of artificial intelligence and image processing technology has helped develop the method of communication between humans and devices in an intuitive and vivid way. This study aims to develop a model for controlling electrical devices through hand gestures by combining the OpenCV-Python library, mediapipe with the Arduino microcontroller board to control peripheral devices. Through the computer's camera, the system employs the image processing library OpenCV-Python, mediapipe to track the user's hand gestures. Under adequate lighting conditions, the system was tested and was able to recognize hand gestures well within a distance of 100 cm and execute the commands to control the Arduino to turn on and off lights, fans, and doors.

Keywords: Hand Detector, Hand Gesture Recognition, Image processing, OpenCV, Python