

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
KHOA ĐIỆN – ĐIỆN TỬ



ĐỒ ÁN TỐT NGHIỆP
ĐẠI HỌC

NGÀNH: CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ - VIỄN THÔNG

ĐỀ TÀI:

ỨNG DỤNG XỬ LÝ ẢNH TRONG HỆ THỐNG
PHÂN LOẠI TRÁI CÂY SỬ DỤNG
RASBERRY VÀ ARDUINO

Người hướng dẫn : **TS. NGUYỄN LINH NAM**

Sinh viên thực hiện: Phan Hoàng Bửu 1811505410103 18DT1

Phan Văn Thành 1811505410233 18DT2

Nguyễn Gia Quốc Triều 1811505410236 18DT2

Đà Nẵng, 4/2022

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
KHOA ĐIỆN - ĐIỆN TỬ



ĐỒ ÁN TỐT NGHIỆP

ĐẠI HỌC

NGÀNH: CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ - VIỄN THÔNG

ĐỀ TÀI:

**ỨNG DỤNG XỬ LÝ ẢNH TRONG HỆ THỐNG
PHÂN LOẠI TRÁI CÂY SỬ DỤNG
RASBERRY VÀ ARDUINO**

Người hướng dẫn : TS. NGUYỄN LINH NAM

Sinh viên thực hiện: Phan Hoàng Bửu 1811505410103 18DT1

Phan Văn Thành 1811505410233 18DT2

Nguyễn Gia Quốc Triền 1811505410236 18DT2

Đà Nẵng, 4/2022

TÓM TẮT

Tên đề tài: Ứng dụng xử lý ảnh trong hệ thống phân loại trái cây sử dụng Raspberry và Arduino

Sinh viên thực hiện :

Phan Hoàng Bửu

Mã SV : 1811505410103

Phan Văn Thành

Mã SV : 1811505410233

Nguyễn Gia Quốc Triển

Mã SV : 1811505410236

Thiết kế và thi công mô hình sản phẩm “Ứng dụng xử lý ảnh trong hệ thống phân loại trái cây sử dụng Raspberry và Arduino” là đề tài mà chúng em đã chọn làm đồ án tốt nghiệp ngành Điện tử - Viễn thông tại trường Đại Học Sư Phạm Kỹ Thuật – Đại Học Đà Nẵng.

Đồ án này sẽ ứng dụng công nghệ xử lý ảnh, áp dụng nó vào trong hệ thống nhận diện hình dạng và màu sắc của từng loại trái cây từ đó có thể phân loại chúng theo từng hình dạng cũng như màu sắc khác nhau.

Để hoàn thành được đồ án này, nhóm em đã sử dụng Raspberry Pi 3 và Camera Pi để nhận diện được hình ảnh, cùng với đó kết nối với Module Arduino và những cảm biến, động cơ để phát hiện, điều khiển và phân loại sản phẩm.

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Giáo viên hướng dẫn : **TS. Nguyễn Linh Nam**

Sinh viên thực hiện : Phan Hoàng Bửu

Mã SV : 1811505410103

Phan Văn Thành

Mã SV : 1811505410233

Nguyễn Gia Quốc Triền

Mã SV : 1811505410236

1. Tên đề tài

ỨNG DỤNG XỬ LÝ ẢNH TRONG HỆ THỐNG PHÂN LOẠI TRÁI CÂY SỬ DỤNG RASPBERRY VÀ ARDUINO

Thời gian thực hiện: từ ngày 21/02/2022 đến 06/06/2022.

2. Các số liệu, tài liệu ban đầu

- Tài liệu tổng quan về đề tài: kỹ thuật xử lý ảnh, lập trình Raspberry Pi, lập trình Arduino.
- Tài liệu về thông số kỹ thuật và lập trình Raspberry Pi, lập trình Arduino, cảm biến camera Pi, cảm biến hồng ngoại, động cơ.
- Tài liệu về kỹ thuật xử lý ảnh (thu nhận ảnh, tiền xử lý ảnh...), tìm hiểu phương pháp nhận dạng và phân loại sản phẩm.

3. Nội dung chính đồ án

- Thiết kế mô hình băng tải phân loại các trái cây với hình dạng và màu sắc khác nhau. Hệ thống xử lý ảnh sử dụng camera Pi để quan sát rồi gửi dữ liệu về kit Raspberry Pi, Arduino Uno để xử lý đưa ra kết quả phân loại. Hệ thống băng chuyền có cảm biến và cơ cấu chấp hành để phân loại đúng loại trái cây vào các vị trí tương ứng.
- Hoàn thành báo cáo đồ án đảm bảo đúng quy trình và tiến độ.

4. Các sản phẩm dự kiến

- Báo cáo tổng kết đồ án tốt nghiệp.
- Mô hình thực tế Hệ thống phân loại trái cây.

5. Họ tên cán bộ hướng dẫn

Trưởng bộ môn

Người hướng dẫn

LỜI NÓI ĐẦU

Ngày nay đất nước ngày càng phát triển và hiện đại hơn trong lĩnh vực công nghệ khoa học điện tử. Khi các ngành sản xuất công nghiệp hiện nay ngày càng đòi hỏi tính chính xác và đảm bảo về nguồn cung cấp đáp ứng nhu cầu người tiêu dùng. Nên nhu cầu sử dụng hệ thống xử lý hình ảnh thông qua camera để thay thế con người trong công đoạn phân loại sản phẩm với độ chính xác cao, nhanh và hoạt động không biết mệt mỏi. Sự xuất hiện của camera kỹ thuật số đã giúp con người có những hình ảnh thực hơn và sử dụng camera cho nhiều mục đích khác nhau. Khi tương tác camera với các thiết bị khác để kết hợp thành hệ thống thông minh, có thể phát hiện những điều kiện bất thường tại nơi giám sát. Bên cạnh đó chúng ta kết hợp thêm thư viện OpenCV để xử lý ảnh trên một số lĩnh vực lắp đặt hệ thống nhận dạng biển số xe, hệ thống nhận dạng vân tay, nhận dạng phân loại các thiết bị...

Đồ án tốt nghiệp lần này là một bước đi cần thiết cho chúng em nhằm hệ thống lại những kiến thức đã được học ở nhà trường sau bốn năm đại học. Đồng thời giúp chúng em bắt đầu làm quen với công việc thiết kế mô hình thực tế tạo tiền đề vững chắc cho công việc sau này.

Để hoàn thành tốt đồ án này, chúng em đã nhận sự quan tâm giúp đỡ nhiệt tình của các Thầy, Cô hướng dẫn chỉ đạo thêm những kiến thức cần thiết, góp ý thêm để hoàn thành tốt đồ án, bên cạnh đó cảm ơn sự giúp đỡ tận tình của bạn bè, những tài liệu tham khảo đã phục vụ cho đồ án cũng như thực tế sau này. Chúng em xin chân thành bày tỏ lòng biết ơn sâu sắc của mình đối với sự giúp đỡ quý báu của thầy giáo hướng dẫn **Nguyễn Linh Nam**.

Quá trình thực hiện đồ án tuy đã cố gắng học hỏi, tiếp thu ý kiến, xong chúng em không thể tránh khỏi những thiếu sót do chưa có nhiều kinh nghiệm thực tế, tư duy còn hạn chế, nên mong muốn nhận được sự chỉ bảo thêm của các thầy cô trong khi chấm và bảo vệ đồ án của chúng em để chúng em có thể hoàn thiện hơn đề tài.

Chúng em xin chân thành cảm ơn!

Đà Nẵng, ngày 06 tháng 04 năm 2022

Sinh viên

CAM ĐOAN

Chúng tôi xin cam đoan đề tài đồ án tốt nghiệp “**Ứng dụng xử lý ảnh trong hệ thống phân loại trái sử dụng Raspberry và Arduino**” là kết quả của quá trình nghiên cứu bản thân dưới sự hướng dẫn của các thầy cô trong khoa Điện – Điện tử. Không sao chép bất kỳ kết quả các đồ án tốt nghiệp nào trước đó. Đồ án tốt nghiệp có tham khảo các tài liệu, thông tin theo tài liệu tham khảo của đồ án tốt nghiệp.

Sinh viên thực hiện

MỤC LỤC

NHẬN XÉT ĐỒ ÁN TỐT NGHIỆP
NHẬN XÉT PHẢN BIỆN ĐỒ ÁN TỐT NGHIỆP
TÓM TẮT
NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

LỜI NÓI ĐẦU	i
CAM ĐOAN.....	ii
MỤC LỤC	iii
DANH SÁCH CÁC HÌNH VẼ	vi
DANH SÁCH CÁC BẢNG.....	viii
DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT	ix
MỞ ĐẦU.....	1
Chương 1: Tổng quan	2
1.1. Tổng quan công nghệ chế biến trái cây tại Việt Nam.....	2
1.1.1. Khái quát ngành kinh tế chế biến trái cây ở Việt Nam.	2
1.1.2. Nhu cầu và định hướng phát triển chung ngành chế biến phân loại trái cây ở Việt Nam.	3
1.1.3. Kết luận về công nghệ chế biến trái cây.....	3
1.2. Giải pháp yêu cầu thiết bị phân loại trái cây.....	4
1.2.1. Các trường hợp cần phân loại.....	4
1.2.2. Các phương pháp phân loại trái cây.....	4
1.3. Bài toán công nghệ đặt ra và lựa chọn giải pháp	7
1.3.1. Bài toán công nghệ đặt ra	7
1.4. Đề xuất mô hình phân loại trái cây trên băng tải tự động.....	7
1.5. Tổng quan hệ thống phần mềm.....	8
1.5.1. Hệ điều hành.....	8
1.5.2. Thư viện xử lý ảnh	9
1.5.3. Ngôn ngữ lập trình	9
1.6. Lý do chọn đề tài.....	10
Chương 2: Cơ sở lý thuyết và công nghệ.....	11
2.1. Tổng quan về xử lý ảnh.....	11
2.1.1. Thu nhận ảnh (Image Acquisition)	11
2.1.2. Phân đoạn (Segmentation) hay phân vùng ảnh.....	12

2.1.3. Biểu diễn ảnh (<i>Image Representation</i>)	12
2.1.4. Nhận dạng và nội suy ảnh (<i>Image Recognition and Interpretation</i>).....	12
2.1.5. Cơ sở tri thức (<i>Knowledge Base</i>)	12
2.1.6. Mô tả.....	13
2.1.7. Những vấn đề trong xử lý ảnh.....	13
2.1.8. Các phép toán hình thái <i>Morphology</i>	16
2.2. Tìm hiểu về Raspberry Pi 3.....	17
2.2.1. Khái niệm.....	17
2.2.2. Thông tin cấu hình Raspberry Pi 3.....	18
2.2.3. Ưu điểm của Raspberry Pi	19
2.2.4. Ứng dụng của Raspberry Pi	19
2.3. Giới thiệu về Arduino	19
2.3.1. Khái niệm.....	19
2.3.2. Sơ đồ chân của Arduino	20
2.4. Giới thiệu về Camera Pi.....	21
2.4.1. Khái niệm.....	21
2.4.2. Ứng dụng	23
2.5. Cảm biến hồng ngoại E18-D80NK.....	23
2.5.1. Khái quát	23
2.5.2. Thông số kỹ thuật.....	24
2.5.3. Sơ đồ chân	24
2.5.4. Ứng dụng	24
2.6. Động cơ DC	24
2.6.1. Khái quát	24
2.6.2. Cấu tạo và nguyên lý hoạt động.....	25
2.6.3. Ứng dụng	26
2.7. Động cơ Servo MG996G	26
2.7.1. Giới thiệu động cơ Servo MG996G.....	26
2.7.2. Ứng dụng	27
2.8. Hệ thống băng tải	27
2.8.1. Giới thiệu.....	27
2.8.2. Cấu tạo.....	28
2.8.3. Ứng dụng	28
2.9. LCD.....	28
2.9.1. Giới thiệu.....	28
2.9.2. Cấu tạo.....	29

2.9.3. Ứng dụng	30
2.10. Hệ điều hành trên Raspberry Pi	30
2.11. Giới thiệu ngôn ngữ Python và thư viện OpenCV	31
2.11.1. Ngôn ngữ Python	31
2.11.2. Thư viện OpenCV	32
2.12. Các chuẩn giao tiếp	33
2.12.1. Chuẩn giao tiếp UART	33
2.12.2. Chuẩn giao tiếp I2C	35
Chương 3: Tính toán và thiết kế	38
3.1. Giới thiệu.....	38
3.2. Tính toán và thiết kế hệ thống.....	38
3.2.1. Thiết kế sơ đồ khối hệ thống.....	38
3.2.2. Sơ đồ kết nối hệ thống	39
3.2.3. Sơ đồ kết nối toàn mạch.....	46
Chương 4: Thi công hệ thống	47
4.1. Giới thiệu.....	47
4.2. Thi công hệ thống	47
4.2.1. Chuẩn bị phần cứng	47
4.2.2. Lắp ráp và kiểm tra	48
4.3. Lập trình hệ thống	51
4.3.1. Lưu đồ giải thuật trên Arduino.....	51
4.3.2. Lưu đồ thuật toán xử lý ảnh trên Raspberry Pi 3.....	51
4.3.3. Hướng dẫn sử dụng và thao tác	52
4.4. Mô hình hoàn thiện	53
4.5. Kết quả tách biên nhận diện từ các quả	54
4.6. Nhận xét và đánh giá.....	57
KẾT LUẬN	58
TÀI LIỆU THAM KHẢO.....	59

DANH SÁCH CÁC HÌNH VẼ

Hình 1.1. Cảm biến màu TCS3200.....	5
Hình 1.2. Robot với cánh tay hái quả sử dụng cảm biến Linder.....	6
Hình 1.3. Phương pháp phân loại quả xoài thủ công.....	7
Hình 1.4. Mô hình mô phỏng quá trình phân loại trái cây.....	8
Hình 1.5. Mô hình tổng quan thiết bị phân loại trái cây.....	8
Hình 2.1. Lân cận 4 và lân cận 8.....	14
Hình 2.2. Phép co.....	17
Hình 2.3. Raspberry Pi 3.....	17
Hình 2.4. Sơ đồ Raspberry Pi 3 và Modul B.....	18
Hình 2.5. Arduino Uno.....	20
Hình 2.6. Arduino Uno.....	20
Hình 2.7. Camera Pi v2.1.....	22
Hình 2.8. Sơ đồ khối Camera Raspberry Pi.....	23
Hình 2.9. Cảm biến hồng ngoại.....	23
Hình 2.10. Động cơ DC.....	24
Hình 2.11. Pha 1 động cơ DC.....	25
Hình 2.12. Pha 2 động cơ DC.....	25
Hình 2.13. Pha 3 động cơ DC.....	26
Hình 2.14. Động cơ servo MG996R.....	27
Hình 2.15. Bảng tải.....	28
Hình 2.16. LCD 16x02.....	29
Hình 2.17. Cấu tạo LCD.....	29
Hình 2.18. Hệ điều hành trên Raspberry.....	31
Hình 2.19. Giao tiếp UART.....	33
Hình 2.20. Truyền thông nối tiếp.....	33
Hình 2.21. UART.....	34
Hình 2.22. Giao tiếp UART.....	35
Hình 2.23. Giao tiếp I2C.....	35
Hình 2.24. Dữ liệu I2C.....	37
Hình 3.1. Sơ đồ khối hệ thống.....	38
Hình 3.2. Sơ đồ kết nối của Raspberry Pi.....	39
Hình 3.3. Sơ đồ các cổng ngoại vi sử dụng.....	40

Hình 3.4. Sơ đồ kết nối của thẻ nhớ Raspberry.....	41
Hình 3.5. Sơ đồ kết nối của Arduino với Raspberry.....	41
Hình 3.6. Giao tiếp Uart giữa Raspberry Pi và Arduino Uno.....	42
Hình 3.7. Sơ đồ kết nối Camera.....	42
Hình 3.8. Giao tiếp I2C giữa Arduino Uno và LCD16x2.....	43
Hình 3.9. Kết nối Arduino Uno và động cơ DC.....	43
Hình 3.10. Kết nối cảm biến hồng ngoại và Arduino Uno.....	44
Hình 3.11. Adapter 5VDC – 2A.....	44
Hình 3.12. Sơ đồ nguyên lý mạch hạ áp LM2596.....	45
Hình 3.13. LM2596.....	45
Hình 3.14. Mạch cấp nguồn cho động cơ DC 24V.....	45
Hình 3.15. Nguồn tổ ong 24VDC – 5A.....	46
Hình 3.16. Sơ đồ nguyên lý mạch.....	46
Hình 3.17. Sơ đồ kết nối toàn mạch.....	46
Hình 4.1. Bảng tải và động cơ DC.....	49
Hình 4.2. Máng khi đưa sản phẩm vào.....	49
Hình 4.3. Máng sau khi phân loại màu.....	49
Hình 4.4. Lưu đồ thuật toán trên Arduino Uno.....	51
Hình 4.5. Lưu đồ thuật xử lý trên Raspberry Pi.....	52
Hình 4.6. Mô hình thực tế.....	53
Hình 4.7. Kết quả nhận dạng của quả chuối xanh.....	54
Hình 4.8. Kết quả nhận dạng của quả chuối vàng.....	54
Hình 4.9. Kết quả nhận dạng của quả cam xanh.....	55
Hình 4.10. Kết quả nhận dạng của quả cam vàng.....	55
Hình 4.11. Kết quả nhận dạng của quả mận xanh.....	56
Hình 4.12. Kết quả nhận dạng của quả mận đỏ.....	56

DANH SÁCH CÁC BẢNG

Bảng 2-1: Cấu tạo LCD 16x02	30
Bảng 3-1: Số liệu của các thiếu bị	41
Bảng 4-1: Thông số chi tiết linh kiện.....	48

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

CHỮ VIẾT TẮT:

OpenCV (Open Source Computer Vision Library): Thư viện nguồn mở hàng đầu cho Computer Vision và Machine Learning, xử lý và hiển thị hình ảnh/video.

CCIR (Comité Consultatif International de Radio et Television): Ủy ban tư vấn Vô tuyến Điện Quốc tế

CPU (Central Processing Unit): Bộ xử lý trung tâm

GPU (Graphics Processing Unit): Bộ xử lý những tác vụ liên quan đến đồ họa cho vi xử lý trung tâm CPU

RAM (Random Access Memory): Bộ nhớ tạm của máy giúp lưu trữ thông tin hiện hành để CPU có thể truy xuất và xử lý.

Wi-fi (Wireless Fidelity): Hệ thống truy cập internet không dây, loại sóng vô tuyến này tương tự như sóng điện thoại, truyền hình và radio.

Vin: Điện áp đầu vào

GND: Nối mass

TX (Transmitter): Chân truyền dữ liệu đến thiết bị khác

RX (Receiver): Chân nhận dữ liệu từ thiết bị khác

PWM (Pulse-width modulation): Điều chế độ rộng xung

SPI (Serial Peripheral Interface): Giao diện ngoại vi nối tiếp, chuẩn đồng bộ nối truyền dữ liệu ở chế độ full – duplex

SCK (Serial Clock): Thiết bị Master tạo xung tín hiệu SCK và cung cấp cho Slave

MISO (Master Input Slave Output): Tín hiệu tạo bởi thiết bị Slave và nhận bởi thiết bị Master. Đường MISO phải được kết nối giữa thiết bị Master và Slave.

MOSI (Master Output Slave Input): Tín hiệu tạo bởi thiết bị Master và nhận bởi thiết bị Slave. Đường MOSI phải được kết nối giữa thiết bị Master và Slave.

SS (Slave Select): Chọn thiết bị Slave cụ thể để giao tiếp. Để chọn Slave giao tiếp thiết bị Master chủ động kéo đường SS tương ứng xuống mức 0 (Low).

I2C (Inter – Integrated Circuit): Giao thức giao tiếp nối tiếp đồng bộ, sử dụng để truyền nhận dữ liệu giữa các IC với nhau chỉ sử dụng hai đường truyền tín hiệu.

SCL (Serial Clock Line): Tạo xung nhịp đồng hồ do Master phát đi

SDA (Serial Data Line): Đường truyền nhận dữ liệu

USB (Universal Serial Bus): Chuẩn kết nối và truyền dữ liệu số tuần tự, tốc độ cao, đa năng, đa môi trường.

DC (Direct Current): Dòng điện một chiều.

MỞ ĐẦU

Mục đích thực hiện đề tài:

- Giảm sức lao động, cải thiện được điều kiện làm việc của con người, tạo cho người lao động tiếp cận với sự tiến bộ của khoa học kỹ thuật và được làm việc trong môi trường ngày càng tốt hơn.
- Nâng cao năng suất lao động, tạo tiền đề cho việc giảm giá thành sản phẩm, cũng như thay đổi mẫu mã một cách nhanh chóng và khoa học.
- Giúp cho quản lý và giám sát trở nên rất đơn giản.

Mục tiêu đề tài:

- Phân loại được trái cây theo hình dạng.
- Dựa trên ngôn ngữ lập trình Python với thư viện chính là OpenCV và được thực hiện trên Kit Raspberry Pi 3 và Kit Arduino Uno.
- Thiết kế và thi công mô hình.
- Trau dồi kỹ năng làm việc, hoạt động nhóm cũng như kỹ năng tìm kiếm tài liệu

Phạm vi và đối tượng nghiên cứu

- Áp dụng cho các cơ sở trung thu mua trái cây vừa nhỏ hoặc các nông trại trồng cây ăn quả cần nâng cấp hệ thống xử lý ảnh với các thiết bị đơn giản, lắp đặt dễ dàng và giá thành phù hợp.

Phương pháp và cấu trúc

- Đồ án thực hiện bằng phương pháp xử lý ảnh kết hợp với cảm biến dựa trên nền tảng Raspberry. Bao gồm phần mở đầu và 4 chương. Phần mở đầu nói đến mục đích, mục tiêu, phạm vi đối tượng áp dụng của đề tài.
- Chương 1 nêu lên cái nhìn tổng quan về ngành chế biến trái cây ở Việt Nam và trên thế giới, đặt ra bài toán và các giải pháp liên quan để giải quyết vấn đề.
- Chương 2 là cơ sở lý thuyết các thiết bị trong đề tài.
- Chương 3 tính toán thiết kế mô hình.
- Chương 4 thử nghiệm và đánh giá sản phẩm đưa ra kết luận chung.

Chương 1: Tổng quan

1.1. Tổng quan công nghệ chế biến trái cây tại Việt Nam.

1.1.1. Khái quát ngành kinh tế chế biến trái cây ở Việt Nam.

Với sự tăng trưởng về diện tích đất nông nghiệp, sản lượng và giá trị xuất khẩu trái cây liên tục tăng trưởng cao trong những năm gần đây. Tình hình thị trường xuất khẩu rau quả Việt Nam những tháng đầu năm nay có sự sụt giảm khá mạnh. Bởi thị trường xuất khẩu rau quả lớn nhất của Việt Nam là Trung Quốc gặp nhiều khó khăn do tình hình Covid-19 diễn biến phức tạp các cửa khẩu đóng cửa. Tuy giảm mạnh nhưng nhóm mặt hàng này đang cho thấy sự chuyển dịch khá tích cực sang thị trường có giá trị xuất khẩu cao như EU, Mỹ, Nhật Bản, Hàn Quốc...

Theo tổng cục Hải quan, xuất khẩu ra quả sang thị trường Trung Quốc giảm mạnh trong 2 tháng đầu năm 2022, chỉ đạt được 261 triệu USD và giảm gần 26% so với cùng kỳ 2021. Do xuất khẩu sang Trung Quốc giảm mạnh, kéo theo kim ngạch xuất khẩu các loại quả của cả nước trong 2 tháng qua chỉ đạt 508 triệu USD, giảm 9.6% so với cùng kỳ năm ngoái. Tuy nhiên, điểm đặc biệt là trong 2 tháng đầu năm 2022 xuất khẩu rau quả sang Mỹ lại tăng cao gần 70%; Hàn Quốc tăng khoảng 32%; Nhật Bản tăng 12%; Australia tăng 45,7%; Hà Lan tăng 51,5%; Nga tăng gần 33,9%... Chính sự tăng trưởng của các thị trường này đã góp phần tạo nên doanh thu xuất khẩu ấn tượng của mặt hàng rau quả ở hiện tại và trong tương lai sẽ phát triển hơn [1]

Với sự tăng trưởng hàng năm vào các thị trường có giá trị kinh tế cao như Mỹ, Australia, Hàn Quốc... đang cho thấy việc sản xuất, vùng nguyên liệu đang có sự chuyển dịch đáp ứng yêu cầu thị trường ngày càng tốt hơn. Các doanh nghiệp đã sẵn sàng cung cấp những nguyên liệu đảm bảo cho việc cung cấp sản phẩm trong nước và xuất khẩu sang thị trường nước ngoài. Những sự liên kết giữa các doanh nghiệp và nông dân tạo thêm niềm tin cho người tiêu dùng. Tuy nhiên, vấn đề đáng lo ngại nhất với các doanh nghiệp là xuất khẩu rau quả tươi thường diễn ra tình trạng kẹt hàng ở các cảng biển. Với mặt hàng quả, việc bảo quản chỉ có giới hạn nên nếu bị kẹt hàng lâu sẽ gây thiệt hại lớn. Nên hiện các doanh nghiệp chủ yếu chọn các mặt hàng có thể bảo quản ở thời gian dài để xuất khẩu. Còn với các sản phẩm bảo quản ngắn hơn như thanh long, doanh nghiệp phải chuyển sang đường hàng không, nhưng chi phí cao, khó cạnh tranh với các mặt hàng cùng chủng loại của những quốc gia khác cùng xuất khẩu.

Đi sâu phân tích xuất khẩu rau quả, đặc biệt là mặt hàng quả cà Việt Nam, chủ yếu xuất khẩu dưới dạng trái cây tươi nên giá trị gia tăng thấp và bị tác động bởi các rào cản kỹ thuật như kiểm dịch thực vật, chất lượng trái cây giảm nhanh, thời gian bảo quản

ngăn do chưa có công nghệ bảo quản tiên tiến và các vấn đề vệ sinh an toàn thực phẩm. Năng suất chế biến phân loại trái cây thấp do chưa sử dụng công nghệ kỹ thuật vào việc phân loại, chủ yếu dựa vào thủ công sức lao động con người.

1.1.2. Nhu cầu và định hướng phát triển chung ngành chế biến phân loại trái cây ở Việt Nam.

Bên cạnh những tiềm năng, lợi thế hiện nay ngành sản xuất từ cây ăn quả Việt Nam đứng trước không ít khó khăn, thách thức như: Quy mô sản xuất nhỏ lẻ, phân tán, liên kết sản xuất theo chuỗi còn hạn chế chưa mang lại giá trị cao, năng suất thấp chưa đáp ứng nhu cầu, khâu kiểm tra phân loại chưa đảm bảo, các sản phẩm qua chế biến còn ít...

Một trong những thách thức mà ngành công nghệ chế biến trái cây đang phải đối mặt đó là công nghệ chế biến chưa đáp ứng được nhu cầu thị trường trong và ngoài nước, để nâng cao giá trị năng suất, chất lượng của sản phẩm nhằm tạo ra lợi nhuận bền vững nên khâu phân loại nâng cao năng suất rất quan trọng trong các dây chuyền phân loại này.

Vì vậy, giải pháp hàng đầu được đưa ra chính là sự thay đổi áp dụng công nghệ mới trong sản xuất, đầu tư dây chuyền sản xuất và kiểm định chất lượng tiên tiến nhằm tăng tín hiệu quả và đảm bảo chất lượng sản phẩm.

Việt Nam xác định trái cây là ngành còn nhiều tiềm năng, dư địa để phát triển. Định hướng chung mà ngành công nghệ chế biến đặt ra là: tiếp tục tập trung đẩy mạnh áp dụng lĩnh vực khoa học công nghệ vào quá trình sản xuất cây ăn quả nhằm tăng giá trị kinh tế, năng suất sản lượng và nâng cao chất lượng, hệ thống kiểm định. Bên cạnh đó, cần tăng cường phát triển công nghệ bảo quản sản phẩm, sử dụng nhiều phương pháp chế biến vào các sản phẩm từ cây ăn quả với các sản phẩm chế biến chính gồm các loại quả đông lạnh, đóng hộp, sấy khô, nước tự nhiên, nước quả cô đặc... nhằm gia tăng giá trị sản xuất, mở rộng thị trường, tăng kim ngạch xuất khẩu.

1.1.3. Kết luận về công nghệ chế biến trái cây.

Ngành sản xuất và chế biến trái cây tại Việt Nam cũng như trên thế giới đang ngày càng một tăng trưởng mạnh hơn do nhu cầu từ đời sống sinh hoạt của con người. Khả năng sản xuất của Việt Nam không thua kém các nước xuất khẩu hàng đầu, nhưng các phương pháp, thiết bị tại các cơ sở trung thu, chế biến phân loại trái cây còn thô sơ, lạc hậu, làm giảm mức độ uy tín, năng suất cho ngành xuất khẩu trái cây.

1.2. Giải pháp yêu cầu thiết bị phân loại trái cây

1.2.1. Các trường hợp cần phân loại

Trong việc phân loại và đánh giá chất lượng trái cây, người ta đánh giá thông qua màu sắc, kích thước và mức độ khuyết tật trên bề mặt quả. Với trường hợp màu sắc có thể chia làm 2 loại chính: quả chín và quả chưa chín.

- Trường hợp 1: Đi trung thu mua trái cây.
 - Kiểm tra thuộc tính quả từ người bán, đưa ra kết luận chung về chất lượng của quả để định ra mức giá tiền.
 - Cả hai bên cùng đánh giá quả qua màu sắc để kiểm chứng định giá.
 - Trái cây có hình dạng chung sẽ được lựa chọn, loại bỏ các quả bị dẹt.
- Trường hợp 2: Trái cây tự trồng và thu hoạch.
 - Khi trung thu từng loại trái cây phải kiểm tra một vài cây, một vài loại quả để xem xét mức độ chín chung của loại trái cây đó trong vườn.
 - Người sử dụng thiết bị cầm tay, lấy mẫu một số lượng nhất định các hình ảnh trên các cây, sau đó đưa ra kết luận có thể thu hoạch hay chưa.
 - Với nhu cầu tìm kiếm từng loại, cũng có thể chụp 3-5 hình ảnh với mỗi loại quả để đưa ra kết luận có thể hái được hay chưa, nhu cầu đòi hỏi cần có một lưới hái quả sử dụng làm cơ cấu chấp hành (tương tự như các thiết bị cắt hái quả tự động hiện nay).
- Trường hợp 3: Trái cây khi ở cơ sở chế biến phân loại.
 - Được tích trữ lại và đưa lên các băng tải để phân loại sản phẩm.
 - Sau khi đi qua hệ thống kiểm tra băng tải tự động đẩy các loại trái cây vào các ngăn tùy theo hình dạng từng loại quả và mức độ màu sắc đặt ra.
 - Phân loại quả theo mức độ chín, loại bỏ các quả hỏng sau khi phát hiện khuyết tật trên mặt hoặc bị thối ngoài vỏ.

1.2.2. Các phương pháp phân loại trái cây

a) Phương pháp sử dụng cảm biến màu và camera

Hiện nay trên thế giới, có rất nhiều loại máy móc dùng để phân loại màu sắc các vật thể sử dụng cảm biến màu sắc, đặc biệt là trong trường hợp phân loại hình dạng và màu sắc như hình tròn, hình vuông hay các loại quả xanh/chín như cà chua, cam, quýt, táo... hay các vật thể nhỏ như hạt cà phê.

Các thiết bị này chỉ cần bộ xử lý có tốc độ cao vừa phải và dễ dàng thiết kế như Arduino, kèm theo cảm biến màu. Tổng chi phí cho một thiết bị như vậy ước tính khoảng

1 triệu đồng (Arduino Nano, camera và cảm biến màu TCS3200). So với các phương pháp khác thì sử dụng thiết bị này giá thành tương đối rẻ phù hợp.



Hình 1.1. Cảm biến màu TCS3200

Tuy nhiên nhược điểm chính là cảm biến màu chỉ nhận biết được 1 màu, tức là muốn phân loại độ xanh/chín của quả thì cần phải sử dụng nhiều cảm biến cho mỗi loại, vì vậy làm chi phí và độ phức tạp của hệ thống tăng.

Hiện nay có một số loại cảm biến sử dụng nhiều kênh màu, tuy nhiên giá thành của nó khác cao như ASTECH CR5-FO (4 kênh màu, tốc độ phản hồi khoảng 5-10ms) có mức giá khoảng 15 triệu đồng.

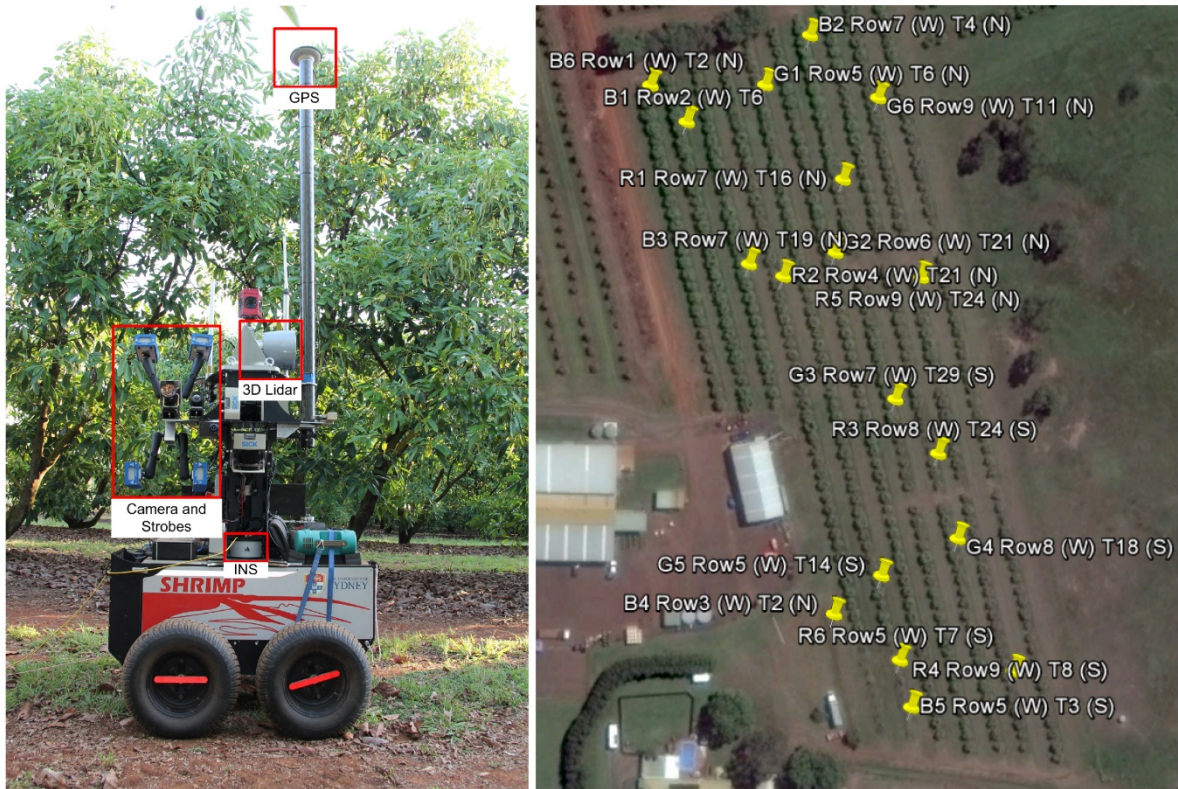
Các thiết bị sử dụng những loại cảm biến như vậy có giá trị rất đắt tiền khi và thường sử dụng ở các nước phát triển hoặc các tập đoàn lớn, các ngành chế biến, xuất khẩu hoa quả chuyên biệt hóa mỗi cơ sở chỉ làm một hoặc hai công đoạn chính.

b) Phương pháp quét laser từ trên không

Hiện nay trên thế giới có một thiết bị từ trường “Linkoping” ở Thụy Điển trong nghiên cứu “Thiết bị dò tìm trái xoài và ước tính thời gian trưởng thành” [2]

Phương pháp này sử dụng thiết bị cảm biến Lidar quét lại hình ảnh 3D của cây cần kiểm tra (phương pháp này đang áp dụng cho cây xoài) từ phía xa, chuyển vào hệ thống xử lý hình ảnh, sau đó đưa ra đánh giá về độ chín của quả trên cây, ở đây sử dụng cơ cấu chấp hành là cánh tay Robot để hái quả ngay trên cây đúng với yêu cầu đặt ra trước đó.

- Ưu điểm: Tính tự động cao, tốc độ kiểm tra cực nhanh khi so sánh với phương pháp thủ công khác.
- Nhược điểm: Giá thành của cảm biến Lindar cũng giống như cảm biến màu có giá thành rất cao và chưa thể đáp ứng được chi phí nếu sử dụng cho các chủ nông trại.



(a) The Shrimp Robotic Platform

(b) Mango Orchard Site

Hình 1.2. Robot với cánh tay hái quả sử dụng cảm biến Lidar

c) Phương pháp phân loại thủ công

Ở phương pháp này sử dụng trực tiếp bằng mắt để kiểm tra thuộc tính và màu sắc của quả có thể loại bỏ những quả bị hư hỏng, tuy nhiên phương pháp này không được sử dụng trong ngành xuất khẩu trái cây do tính chính xác và thiếu vệ sinh khi chạm trực tiếp lên da tay hoặc các đồ vật có bụi bẩn ngay trước khi đóng gói sản phẩm. Điều này có thể gây ra hỏng quả do các vi sinh trên da tay hay các vật tiếp xúc với quả, đặc biệt là trong quá trình phân loại, hay lưu trữ sau khi đóng gói.

- Ưu điểm:
 - Đơn giản, không cần máy móc.
 - Dễ thực hiện nên được áp dụng rộng rãi.
- Nhược điểm:
 - Phân loại độ chín trái cây theo cảm tính và độ chính xác không cao.
 - Tốc độ phân loại chậm, năng suất thấp.

Tại Việt Nam, các nông trại hoa quả sau khi trưng thu (ước tính thời gian quả chín và trưng thu hàng loạt) chủ yếu vận chuyển cho các đầu mối ở chợ, địa phương tiêu thụ, chỉ thông qua quy trình phân loại đánh giá thủ công và trong quá trình này quả xoài nói riêng hay trái cây nói chung đều bị tiếp xúc môi trường ô nhiễm, nơi các vi sinh khuẩn gây bệnh cho trái cây.



Hình 1.3. Phương pháp phân loại quả xoài thủ công

Ngành xuất khẩu hoa quả của Việt Nam chủ yếu là xuất khẩu sang thị trường Trung Quốc, do quy trình kiểm tra đầu vào của trái cây ở đây không cần tính chính xác cao. Tuy nhiên thị trường trái cây nhập khẩu vào Trung Quốc của Việt Nam vẫn xếp sau Thái Lan.

1.3. Bài toán công nghệ đặt ra và lựa chọn giải pháp

1.3.1. Bài toán công nghệ đặt ra

Những phương pháp phân loại trái cây hiện nay hoặc có giá thành cao, hoặc không đảm bảo tính chính xác cao. Công nghệ cao sẽ mất chi phí vô cùng lớn, còn quy trình thủ công thì lại làm giảm tính chính xác, đem lại năng suất thấp và tốn sức lao động. Vậy bài toán đặt ra ở chúng ta nên cần chọn như thế nào để phù hợp cho các cơ sở nhỏ lẻ, đảm bảo được mức độ vừa phải, độ chính xác tương đối cao để đáp ứng được yêu cầu mong muốn.

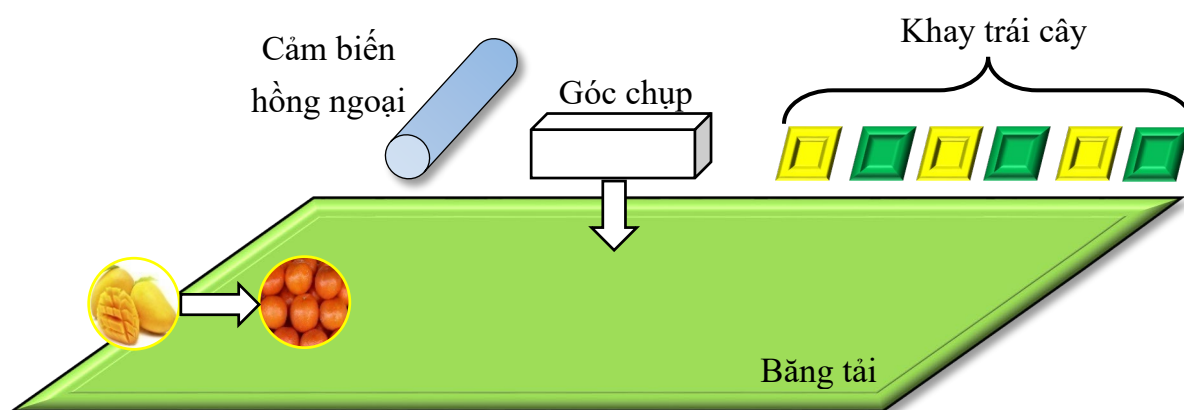
1.4. Đề xuất mô hình phân loại trái cây trên băng tải tự động.

Thiết bị đo được vuông góc với băng tải phụ và đặt cách băng tải phụ khoảng 30cm hoặc sẽ tự động thay đổi độ cao với yêu cầu của tiêu dung sao cho quả khi chạy qua thiết bị thì bề mặt quả xoài nằm gần trung tâm giữa camera và khung hình của camera có thể chứa toàn bộ bề mặt của quả.

Hệ thống băng chuyền là các mắt xích nối tiếp nhau, mỗi mắt sẽ đặt quả lên và có thể xoay đa hướng, sử dụng các động cơ đẩy để đẩy quả vào các ô phân loại.



Hình 1.4. Mô hình mô phỏng quá trình phân loại trái cây



Hình 1.5. Mô hình tổng quan thiết bị phân loại trái cây

1.5. Tổng quan hệ thống phần mềm

1.5.1. Hệ điều hành

Từ những ưu nhược điểm và cân nhắc vấn đề đặt ra của bài toán cần giải, từ một số hệ điều hành phổ biến như Raspbian, Ubuntu, Win 10 Iot CoreOSMC và OpenELEC, chúng em quyết định sử dụng hệ điều hành Raspbian Jesse là hệ điều hành chính cho máy tính nhúng Raspberry Pi.

Hệ điều hành Raspbian Jesse có nền tảng xây dựng từ hệ điều hành Ubuntu, là hệ điều hành dễ sử dụng, khá phổ biến trong cộng đồng người sử dụng Raspberry Pi nên có thể nhận được sự hỗ trợ từ cộng đồng người sử dụng. Raspbian là một phiên bản không chính thức của Debian Wheezy – đây là một phiên bản Linux có tiếng. Raspbian rất đơn giản và quen thuộc. Nó là nền tảng rất tốt cho những người mới bắt đầu làm quen với Raspberry nói riêng và Linux nói chung.

Nhược điểm của nó là giao diện đơn giản, cổ điển và rất khó hào nhoáng. Nhưng hiệu suất của hệ điều hành đem lại là cực kì cao và được tối ưu rất tốt khi so sánh với các hệ điều hành đang phổ biến trên Pi như Ubuntu Mate hay Win10 IotCore.

1.5.2. Thư viện xử lý ảnh

OpenCV (Open Source Computer Vision) là một thư viện mã nguồn mở nói về thị giác máy tính với hơn 500 hàm và 2500 thuật toán đã tối ưu để xử lý ảnh và các vấn đề liên quan đến thị giác máy tính thay cho con người. OpenCV được thiết kế một cách tối ưu, sử dụng tối đa sức mạnh củ dòng chip đa lõi để thực hiện các phép tính toán trên thời gian thực, tức là tốc độ đáp ứng đủ nhanh cho các phép tính toán thông thường và đơn giản. OpenCV là thư viện thiết kế để chạy trên nhiều nền tảng khác nhau (cross – platfrom), nghĩa là nó có thể chạy trên nhiều hệ điều hành Windows, Mac, Ubuntu... việc sử dụng thư viện OpenCV tuân theo những quy định sử dụng mã nguồn mở BSD do đó có thể sử dụng thư viện này vào các mục đích thương mại và phi thương mại. Năm 2005, OpenCV đã được sử dụng trên Stanley, chiếc xe đã giành được giải Grand DARPA 2005[3]. OpenCV hỗ trợ rất nhiều thuật toán liên quan đến thị giác máy tính và học máy ngày càng mở rộng. Cho đến nay thì OpenCV vẫn là một thư viện mở, được phát triển bởi nguồn quỹ không lợi nhuận và được sử hưởng ứng từ cộng đồng.[4]

1.5.3. Ngôn ngữ lập trình

Ngôn ngữ lập trình là một hệ thống được ký hiệu hóa dùng để miêu tả những phép tính toán thông qua máy tính trong một dạng mà cả máy tính lẫn con người đều có thể đọc và miêu tả được. Là một tập con của ngôn ngữ máy tính đã chuẩn hóa và được thiết kế để truyền những chỉ thị cho các thiết bị có bộ xử lý. Ngôn ngữ lập trình được sử dụng để tạo ra các chương trình với mục đích điều khiển máy tính thực hiện các công việc nào đó thông qua các câu lệnh.

Đối với các ngôn ngữ phổ biến, phát triển lâu dài người ta thường có các tiêu chuẩn chính thức do ngôn ngữ đó. Đồng thời, tổ chức các hội thảo để mở rộng, bổ sung các tiêu chuẩn trước đó. Ví dụ như ngôn ngữ C++, các hội đồng tiêu chuẩn ANSI C++ và ISO C++ đã tổ chức đến 13 cuộc hội thảo để điều chỉnh, sửa đổi và nâng cấp ngôn ngữ này.

Ngôn ngữ lập trình bao gồm những chi tiết kỹ thuật sau:

- Dữ liệu và cấu trúc dữ liệu.
- Câu lệnh và dòng điều khiển.
- Các tên hàm và tham số.
- Các cơ chế tham khảo và sự tái sử dụng.

Vai trò của ngôn ngữ lập trình

- Xác định hiểu rõ các yêu cầu cần thiết của khách hàng.
- Xác định và phân tích rõ hệ thống sẽ làm được những gì theo quan điểm của người dùng.

- Đưa ra cách thức giải quyết vấn đề bằng ngôn ngữ lập trình cụ thể.
- Ghép nối các bộ phận của hệ thống và kiểm tra xem có vận hành đúng thiết kế không.[5]

1.6. Lý do chọn đề tài

Trái cây là một loại lương thực thực phẩm không thể thiếu đối với trong cuộc sống của con người khi cung cấp cho cơ thể chúng ta nhiều chất dinh dưỡng, chất xơ và các loại vitamin giúp con người có thể khỏe mạnh hơn về cả thể chất lẫn tinh thần, giảm nguy cơ tránh mắc bệnh tật.

Trái cây được người tiêu dùng lựa chọn hàng đầu và nhu cầu này luôn là lựa chọn hàng đầu đặc biệt là trong mùa hè. Công dụng nổi bật của nó có thể kể đến như giảm cân, chăm sóc mắt, tăng cường hệ miễn dịch kiểm soát cholesterol, tăng trí nhớ và độ tập trung. Cùng với đó, tiêu chí về trái cây tươi của người tiêu dùng được đặt ra ngày càng khắt khe và đòi hỏi ngành công nghiệp chế biến các thực phẩm thiết yếu để đáp ứng nhu cầu khách hàng.

Ngành công nghệ chế biến phân loại trái cây tại Việt Nam đa số đều là các doanh nghiệp vừa và nhỏ, chưa có yếu tố tập trung sản xuất từng giai đoạn nhất định như các nước khác, vì vậy các thiết bị máy móc tự động hóa còn rất hạn chế trong các cơ sở trung thu mua trái cây, các công đoạn chủ yếu được làm cách thủ công, có thể gây thiệt hại lớn cho các doanh nghiệp khi tiêu tốn nhiều nhân lực cũng như gây ra sai số trong quá trình phân loại.

Trong thời đại công nghiệp hóa, hiện đại hóa hiện nay, việc nâng cấp thiết bị máy móc thực hiện những công việc tự động được đặt ra ngày càng nhiều để có thể giải quyết vấn đề hiện nay là giảm số lượng công nhân cũng như tiết kiệm diện tích của cơ sở, giảm sai số phân loại khi thực hiện thủ công và chuyển sang thực hiện bằng máy móc, giúp cho doanh nghiệp có thể quản lý nhà máy một cách tốt hơn. Các dây chuyền phân loại trái cây sẽ được nâng cấp với một chi phí hết sức hợp lý, giảm các lỗi do thao tác thủ công gây ra.

Từ đó chúng em thống nhất cùng sự giúp đỡ của giáo viên hướng dẫn để đưa ra đề tài phù hợp có tên:

“ỨNG DỤNG XỬ LÝ ẢNH TRONG HỆ THỐNG PHÂN LOẠI TRÁI CÂY SỬ DỤNG RASBERRY VÀ ARDUINO”

Chương 2: Cơ sở lý thuyết và công nghệ

2.1. Tổng quan về xử lý ảnh

“Xử lý ảnh là một lĩnh vực mang tính khoa học và công nghệ. Nó là một ngành khoa học mới so với nhiều ngành khoa học khác nhưng tốc độ phát triển của nó rất nhanh, kích thích các trung tâm nghiên cứu, ứng dụng đặc biệt là máy tính chuyên dụng riêng cho nó”. [6]

Xử lý ảnh là lĩnh vực với nhiều kiến thức cơ sở khác nhau được áp dụng vào trong công việc tăng cường và xử lý ảnh thu nhận được từ các thiết bị như camera, webcam.... Vì vậy, xử lý ảnh đã được ứng dụng và phát triển trong các nhiều lĩnh vực như:

- Trong lĩnh vực quân sự được sử dụng để xử lý ảnh và nhận dạng ảnh, xử lý âm thanh đồ họa
- Trong lĩnh vực an toàn bảo mật dùng nhận diện khuôn mặt người, nhận diện vân tay hay là mẫu mắt...
- Trong các lĩnh vực giải trí như trò chơi điện tử
- Trong lĩnh vực y tế áp dụng vào chụp chụp X quang, MRI...

Các phương pháp xử lý ảnh bắt đầu từ các ứng dụng: nâng cao chất lượng và phân tích ảnh. Vấn đề nâng cao chất lượng ảnh có ảnh hưởng đến tới phân bố mức sáng và độ phân giải của ảnh. Máy tính phát triển nhanh tạo điều kiện cho quá trình xử lý ảnh được thuận tiện và sắc nét nhiều màu hơn. Năm 1964, máy tính đã có chức năng xử lý và nâng cao được chất lượng ảnh của mặt trăng và vệ tinh như: làm nổi đường biên, lưu lại ảnh. Từ năm 1964 đến nay, các thiết bị xử lý, nâng cao về chất lượng, nhận dạng hình ảnh phát triển không ngừng. Các phương pháp liên quan đến tri thức nhân tạo như mạng Nơ-ron nhân tạo, các công cụ nén ảnh, thuật toán xử lý hiện đại, được cải tiến và ngày một áp dụng rộng rãi và đem lại nhiều kết quả khả quan.

Sau đây, ta sẽ xét các bước cần thiết cho quá trình xử lý ảnh. Đầu tiên từ thế giới bên ngoài được thu nhận qua các thiết bị thu như (Camera, máy chụp ảnh). Gần đây với sự phát triển của khoa học công nghệ, các ảnh màu hoặc đen trắng được lấy ra từ Camera, sau đó được chuyển trực tiếp thành ảnh số tạo thuận lợi cho việc xử lý tiếp theo. Ngoài ra, ảnh có thể quét từ vệ tinh chụp trực tiếp bằng máy quét ảnh.

2.1.1. Thu nhận ảnh (Image Acquisition)

Ảnh có thể nhận từ Camera màu hoặc trắng đen. Các ảnh thường nhận qua camera là ảnh tương tự (loại camera ống chuẩn CCIR có tần số 1/25Hz, mỗi ảnh khoảng 25

dòng), ngoài ra còn có loại camera đã số hóa (loại CCD - Charge Coupled Device) là loại photodiode dùng để tạo cường độ sáng tại mỗi điểm ảnh.

Camera thường được dùng là loại quét dòng, ảnh tạo ra có dạng hai chiều. Chất lượng mà ảnh thu nhận được tùy phụ thuộc vào thiết bị thu.

2.1.2. Phân đoạn (Segmentation) hay phân vùng ảnh

Phân vùng ảnh là tách một ảnh đầu vào thành các vùng thành phần để biểu diễn phân tích, nhận dạng ảnh. Ví dụ: để nhận dạng chữ (hoặc mã vạch) trên phong bì thư cho mục đích phân loại bưu phẩm, cần chia các câu chữ về địa chỉ hoặc tên người thành các từ, các chữ, các số (hoặc các vạch) riêng biệt để nhận dạng. Đây là phần phức tạp khó khăn nhất trong xử lý ảnh và cũng dễ gây lỗi, làm mất độ chính xác của ảnh. Kết quả nhận dạng ảnh còn phụ thuộc rất nhiều vào công đoạn này.

2.1.3. Biểu diễn ảnh (Image Representation)

Đây là phần sau phân đoạn chứa các điểm ảnh của vùng ảnh (ảnh đã phân đoạn) cộng với mã liên kết ở các vùng lân cận. Việc biến đổi các số liệu này thành dạng thích hợp là cần thiết cho xử lý tiếp theo bằng máy tính. Việc chọn các tính chất để thể hiện ảnh được gọi là trích chọn đặc trưng (Feature Extraction) gắn với việc tách các đặc tính của ảnh dưới dạng các thông tin định lượng hoặc làm cơ sở để phân biệt lớp đối tượng này với đối tượng khác trong phạm vi ảnh nhận được.

2.1.4. Nhận dạng và nội suy ảnh (Image Recognition and Interpretation)

Nhận dạng ảnh là quá trình xác định ảnh. Quá trình này thường thu được bằng cách so sánh với mẫu chuẩn đã được học từ trước. Nội suy là phán đoán theo ý nghĩa trên cơ sở nhận dạng.

- Nhận dạng theo tham số.
- Nhận dạng theo cấu trúc.

Một số đối tượng nhận dạng khá phổ biến được áp dụng trong khoa học và công nghệ là: nhận dạng ký tự (chữ in, chữ viết tay, chữ ký điện tử), nhận dạng văn bản (Text), nhận dạng vân tay, nhận dạng mã vạch, nhận dạng mặt người, ...

2.1.5. Cơ sở tri thức (Knowledge Base)

Ảnh là một đối tượng khá phức tạp về đường nét, độ sáng tối, dung lượng điểm ảnh, môi trường để thu ảnh phong phú kéo theo nhiều. Trong nhiều khâu xử lý và phân tích ảnh ngoài việc đơn giản hóa các phương pháp toán học đảm bảo tiện lợi cho xử lý, người ta bắt chước quy trình tiếp nhận và xử lý ảnh theo cách của con người. Trong các bước xử lý đó, nhiều khâu hiện nay đã xử lý theo các phương pháp trí tuệ con người. Vì vậy, ở đây các cơ sở tri thức được phát huy.

2.1.6. Mô tả

Ảnh sau khi số hóa sẽ lưu vào bộ nhớ, hoặc truyền sang các khâu tiếp theo để phân tích. Nếu lưu trữ ảnh trực tiếp từ các ảnh thô, đòi hỏi dung lượng bộ nhớ cực lớn và không hiệu quả theo quan điểm ứng dụng và công nghệ. Thông thường, các ảnh được gọi là các đặc trưng ảnh như: biên ảnh, vùng ảnh.

2.1.7. Những vấn đề trong xử lý ảnh

a) Điểm ảnh (Picture Element)

Là đơn vị cơ bản nhất để tạo nên một bức ảnh kỹ thuật số. Địa chỉ của điểm ảnh được xem như là một tọa độ (x, y) nào đó. Một bức ảnh kỹ thuật số, có thể được tạo ra bằng cách chụp hoặc bằng một phương pháp đồ họa nào khác, được tạo nên từ hàng ngàn hoặc hàng triệu pixel riêng lẻ. Bức ảnh càng chứa nhiều pixel thì càng chi tiết. Một triệu pixel thì tương đương với 1 megapixel."

b) Ảnh số

Ảnh số là tập hợp hữu hạn các điểm ảnh với mức xám phù hợp dùng để mô tả ảnh gần với ảnh thật. Số điểm ảnh xác định độ phân giải của ảnh. Ảnh có độ phân giải càng cao thì càng hiển thị rõ nét các đặc điểm của tấm hình càng làm cho tấm ảnh trở nên thực và sắc nét hơn. Một hình ảnh có tín hiệu hai chiều, nó được xác định bởi hàm toán học $f(x, y)$ trong đó x và y là hai tọa độ theo chiều ngang và chiều dọc. Các giá trị của $f(x, y)$ tại bất kỳ điểm nào là cung cấp các giá trị điểm ảnh (pixel) tại điểm đó của một hình ảnh

c) Phân loại ảnh

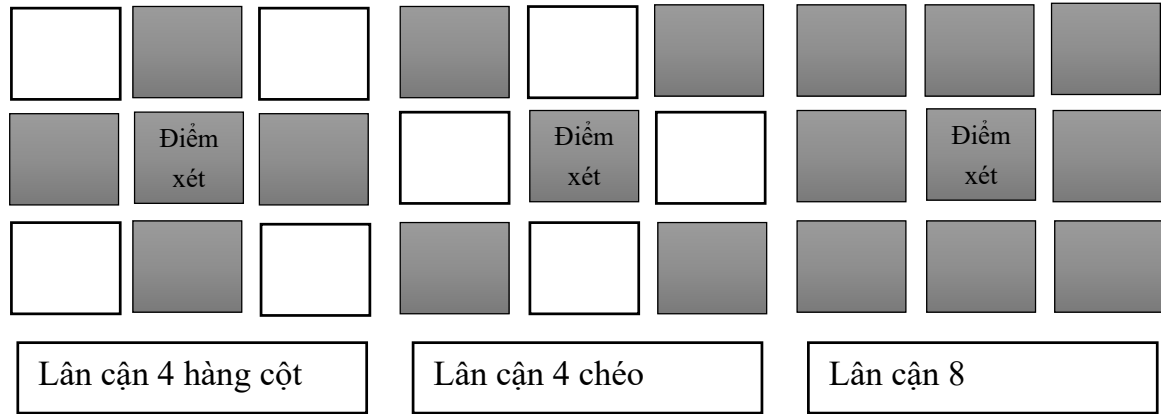
Mức xám của điểm ảnh là cường độ sáng, gán bằng một giá trị tại điểm đó. Các mức ảnh xám thông thường: 16, 32, 64, 128, 256. Mức được sử dụng thông dụng nhất là 256, tức là dùng 1 byte để biểu diễn mức xám.

Trong đó:

- Ảnh nhị phân: Là ảnh có hai màu trắng và đen chỉ có 2 giá trị 0 và 1 và chỉ sử dụng 1 bit dữ liệu trên 1 điểm ảnh
- Ảnh đen trắng: là ảnh có 2 màu đen và trắng (không chứa màu khác) với mức xám ở các điểm ảnh có thể khác nhau.
- Ảnh màu: Là ảnh kết hợp của 3 màu cơ bản lại với nhau để tạo ra một thế giới
- Màu sinh động. Người ta thường dùng 3byte để mô tả mức màu, tức là có 16,7 triệu màu.

d) Quan hệ giữa các điểm ảnh

Lân cận điểm ảnh: được nói một cách hài hước như là hàng xóm của cái điểm ảnh. Có 2 loại lân cận cơ bản là lân cận 4 và lân cận 8.



Hình 2.1. *Lân cận 4 và lân cận 8*

4 điểm ảnh lân cận 4 theo cột và hàng với tọa độ lần lượt là $(x+1, y)$, $(x-1, y)$, $(x, y+1)$, $(x, y-1)$ ký hiệu là tập $N4(p)$. 4 điểm ảnh lân cận 4 theo đường chéo có tọa độ lần lượt là $(x+1, y+1)$, $(x+1, y-1)$, $(x-1, y+1)$, $(x-1, y-1)$ ký hiệu là tập $ND(p)$. Tập 8 điểm ảnh lân cận 8 là hợp của 2 tập trên: $N8(p) = N4(p) + ND(p)$.

Liên kết ảnh: Các mối liên kết của ảnh được xem như là mối liên kết của 2 điểm ảnh gần nhau, có 3 loại liên kết: liên kết 4, liên kết 8, liên kết m (liên kết hỗn hợp). Trong ảnh đa mức xám, ta có thể đặt V chứa nhiều giá trị như $V = \{\text{tập con}\}$. Cho p có tọa độ (x, y) .

Liên kết 4: hai điểm ảnh p và q có giá trị thuộc về tập V được gọi là liên kết 4 của nhau nếu q thuộc về tập $N4(p)$.

Liên kết 8: hai điểm ảnh p và q có giá trị thuộc về tập V được gọi là liên kết 8 của nhau nếu q thuộc về tập $N8(p)$.

Liên kết m: hai điểm ảnh p và q có giá trị thuộc về tập V được gọi là liên kết M của nhau nếu thỏa 1 trong 2 điều kiện sau: q thuộc về tập $N4(p)$, q thuộc về tập $ND(p)$ và giao của hai tập $N4(p)$, $N4(q)$ không chứa điểm ảnh nào có giá trị thuộc.

e) **Lọc nhiễu**

Ảnh thu nhận được thường sẽ bị nhiễu nên cần phải loại bỏ nhiễu. Các toán tử không gian dùng trong kỹ thuật tăng cường ảnh được phân nhóm theo công dụng: làm trơn nhiễu, nổi biên. Để làm trơn nhiễu hay tách nhiễu, người ta thường dùng các bộ lọc tuyến tính (lọc trung bình, thông thấp) hoặc lọc phi tuyến (trung vị, giả trung vị, lọc đồng hình). Từ bản chất của nhiễu (thường tương ứng với tần số cao) và từ cơ sở lý thuyết lọc là: bộ lọc chỉ cho tín hiệu có tần số nào đó thông qua, để lọc nhiễu người ta thường dùng lọc thông thấp (theo quan điểm tần số không gian) hay lấy tổ hợp tuyến tính để san bằng (lọc trung bình). Để làm nổi cạnh (ứng với tần số cao), người ta dùng các bộ lọc thông cao, lọc Laplace. Phương pháp lọc nhiễu. Chia làm 2 loại: lọc tuyến tính, lọc phi tuyến.

Làm trơn nhiều bằng lọc phi tuyến: Các bộ lọc phi tuyến cũng thường được dùng trong kỹ thuật tăng cường ảnh. Một số phương pháp lọc cơ bản bộ lọc trung vị, lọc ngoài... Với lọc trung vị, điểm ảnh đầu vào sẽ được thay thế bởi trung vị các điểm ảnh còn lọc giả trung vị sẽ dùng trung bình cộng của hai giá trị “trung vị” (trung bình cộng của max và min).

Lọc trung vị: Kỹ thuật này đòi hỏi giá trị các điểm ảnh trong cửa sổ phải xếp theo thứ tự tăng hay giảm dần so với giá trị trung vị. Kích thước cửa sổ thường được chọn sao cho số điểm ảnh trong cửa sổ là lẻ.

Lọc ngoài: Giả thiết có ngưỡng cho các mức nhiễu (có thể dựa vào lược đồ xám). Tiến hành so sánh giá trị độ xám của một điểm ảnh với trung bình số học 8 lân cận của nó. Nếu sai lệch lớn hơn ngưỡng, điểm ảnh này được coi như nhiễu. Trong trường hợp đó, thay thế giá trị của điểm ảnh bằng giá trị trung bình 8 lân cận vừa tính được.

f) Phương pháp phát hiện biên

Biên là một trong những vấn đề chúng ta cần quan tâm trong xử lý ảnh. Vì ở giai đoạn phân đoạn ảnh chủ yếu dựa vào biên.

Điểm biên: Một điểm ảnh được gọi là điểm biên nếu có sự thay đổi nhanh hoặc đột ngột về mức xám (hoặc màu). Ví dụ trong ảnh nhị phân, điểm đen gọi là điểm biên nếu lân cận nó có ít nhất một vài điểm trắng. Đường biên (đường bao: boundary): tập hợp các điểm biên liên tiếp tạo thành một đường biên hay đường bao.

Ý nghĩa của đường biên trong xử lý: ý nghĩa đầu tiên của đường biên là một loại đặc trưng tiêu biểu trong phân tích, nhận dạng ảnh. Thứ hai, người ta sử dụng biên làm phân cách các vùng xám (màu) cách biệt. Ngược lại, người ta cũng sử dụng các vùng ảnh để tìm đường phân cách. Tầm quan trọng của biên: để thấy rõ tầm quan trọng của biên.

Như vậy, phát hiện biên là phát hiện được tất cả các đường biên trong các đối tượng. Định nghĩa toán học của biên ở là cơ sở cho các kỹ thuật phát hiện biên. Điều quan trọng là sự biến thiên giữa các điểm ảnh thường nhỏ, trong khi đó biến thiên độ sáng của điểm biên thường là khá lớn khi qua biên. Xuất phát cơ sở này người ta thường sử dụng hai phương pháp phát hiện biên như sau: Có 2 phương pháp cơ bản là: một là tạo gradient của hai hướng và trực giao trong ảnh, hai là dùng tập đạo hàm có hướng. Tách biên theo đạo hàm bậc hai: được thực hiện trên một số dạng vi phân bậc 2 để làm xuất hiện biên. Có hai dạng của phương pháp đạo hàm bậc hai đã được nghiên cứu là: phương pháp Laplace và đạo hàm trực tiếp.

Bộ tách biên Canny: Phương pháp phát hiện này được sử dụng phổ biến vì nó có nhiều ưu điểm hơn so với các phương pháp khác. Các bước thực hiện: Làm phẳng dùng bộ lọc Gauss; Sau đó, Gradient cục bộ của biên độ và hướng được tính. Tìm điểm

ảnh có biên độ lớn nhất dùng kỹ thuật nonmaximal suppression; Các điểm ảnh đỉnh được chia làm hai ngưỡng T1 và T2, $T1 < T2$. Các điểm ảnh đỉnh có giá trị lớn hơn T2 được gọi là Strong và nằm trong khoảng T1 và T2 được gọi là Weak. Liên kết các điểm ảnh Weak có 8 kết nối đến điểm ảnh Strong.

Phương pháp gradient: Gradient là một vector có các thành phần biểu thị tốc độ thay đổi giá trị của điểm ảnh theo 2 hướng x và y, hay có thể nói là nó đại diện cho sự thay đổi về hướng và độ lớn của một vùng ảnh." $\nabla(X.Y) = f_x = a_x = f(x+dx, y) - f(x,y)$

g) Phân đoạn ảnh

Phân đoạn ảnh là bước then chốt trong xử lý ảnh. Giai đoạn này nhằm phân tích ảnh thành các vùng có cùng tính chất nào đó dựa theo biên hay các vùng liên thông. Tiêu chuẩn để xác định các vùng liên thông có thể là cùng mức xám, cùng màu hay cùng độ nhóm

Quá trình phân đoạn ảnh nhằm tách đối tượng ra khỏi phần nội dung còn lại của ảnh, hay phân chia các đối tượng trong ảnh thành những đối tượng riêng biệt. Như vậy quá trình phân đoạn ảnh là quá trình có thể giảm bớt số lượng thông tin trong ảnh và chỉ giữ lại những thông tin cần thiết cho ứng dụng. Do đó phân đoạn ảnh là quá trình loại bỏ các đối tượng không quan tâm trong ảnh. Có nhiều phương pháp phân đoạn ảnh khác nhau. Trong đó quá trình phân đoạn ảnh sử dụng một ngưỡng giá trị xám để phân đoạn ảnh ra thành các đối tượng và nền là phương pháp đơn giản nhất. Lúc này các điểm ở bên dưới ngưỡng giá trị xám thuộc về nền còn những điểm ảnh ở bên trên ngưỡng giá trị xám thuộc về đối tượng. Phương pháp phân đoạn ảnh này hiệu quả hơn đối với ảnh nhị phân, văn bản in hay đồ họa... Dựa vào đặc tính vật lý của vùng ảnh, các kỹ thuật phân đoạn vùng có thể được chia làm 3 loại:

- Kỹ thuật tổng thể: phân đoạn một ảnh dựa trên cơ sở các thông tin lấy từ tổng thể như sử dụng biểu đồ mức xám.
- Kỹ thuật chia, nối và phát triển dựa trên các khái niệm tương đồng về hình dạng và tính đồng nhất. Hai vùng có thể được nối lại với nhau và liền kề bên nhau. Các vùng không đồng nhất có thể được chia thành các vùng nhỏ. Một vùng có thể được phát triển bằng cách nối các điểm ảnh sao cho nó đồng nhất với nhau.

2.1.8. Các phép toán hình thái Morphology

Khái niệm Morphology trong xử lý ảnh số khởi nguồn từ một ngành của sinh học, nghiên cứu về hình thể và cấu trúc của động thực vật. Đây là một công cụ giúp rút ra các thành phần trong ảnh nhị phân, biểu diễn và mô tả chúng dưới dạng các vùng hoặc dạng như các đường biên, xương và bao lồi.

Phép giãn (Dilation): Phép toán Dilation là thao tác giãn nở/phình to các đối tượng ảnh đơn sắc.



Hình 2.2. Phép co

a) Những định dạng của ảnh

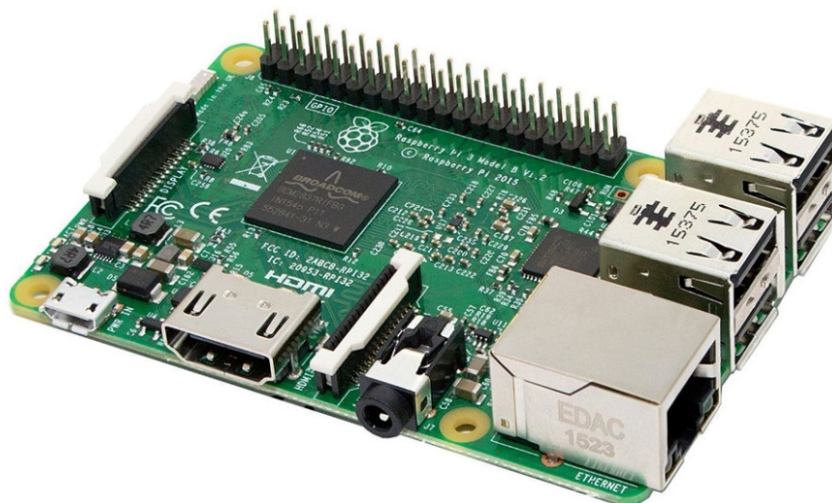
Ảnh thu nhận được sau quá trình số hóa thường được lưu lại cho các quá trình xử lý tiếp theo hay truyền đi. Trong quá trình truyền của kỹ thuật xử lý ảnh, tồn tại nhiều định dạng khác nhau từ ảnh đen trắng như định dạng IMG, ảnh đa cấp xám cho đến ảnh màu (BMP, JPEG, GIF).

b) Các phần mềm hỗ trợ xử lý ảnh

Hiện nay xử lý ảnh được giảng dạy trường đại học và ứng dụng vào thực tế rất nhiều như các phần mềm chỉnh sửa hình ảnh hay nhận biết khuôn mặt. Chính vì thế có rất nhiều công cụ để chúng ta lập trình ứng dụng vào thực tế.

2.2. Tìm hiểu về Raspberry Pi 3

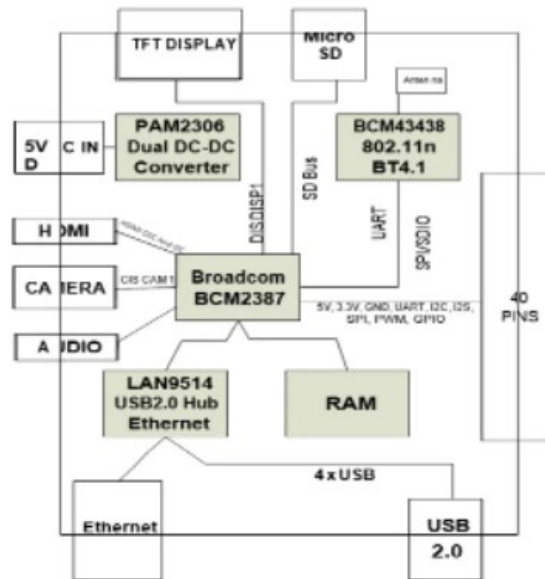
2.2.1. Khái niệm



Hình 2.3. Raspberry Pi 3

Pi là một máy vi tính rất nhỏ gọn, kích thước hai cạnh chỉ cỡ một cái thẻ ATM. Người ta đã được tích hợp mọi thứ cần thiết trong đó để bạn sử dụng như một cái máy vi tính. Trên bo mạch của Pi có CPU, GPU, RAM, khe cắm thẻ microSD, Wi-Fi, Bluetooth và 4 cổng USB 2.0. Khi mua Pi về, bạn chỉ việc cài hệ điều hành (thực ra là copy/paste cái thư mục vô thẻ nhớ), gắn chuột, bàn phím và màn hình là bắt đầu sử dụng được rồi (hoặc cao cấp hơn xịu là remote desktop từ một máy khác qua, hoặc SSH).[7]

2.2.2. Thông tin cấu hình Raspberry Pi 3



Hình 2.4. Sơ đồ Raspberyy Pi 3 và Modul B

Thông tin cấu hình:

- Bộ xử lý IBroadcom BCM2835 tốc độ xử lý 12gtz 64-bit qual-core ARM Cortex A53
- Mạng Wireless LAN chuẩn 802.11 b/g/n
- Bộ xử lý đa phương tiện Video Core IV® Dual Core
- Bộ nhớ Ram 1GB
- Hỗ trợ tất cả các bản phân phối ARM GNU / Linux mới nhất và Windows 10 iot
- Đầu nối micro usb cho nguồn điện 2,5A 5VDC
- Cổng mạng lx 10/100 +
- Đầu nối video / âm thanh lx HDMI
- Đầu nối video / âm thanh lx RCA
- USB 2.0 ports
- 40 GPIO pins
- Chip antenna

- Kết nối hiển thị DSI
- Khe cắm thẻ nhớ

2.2.3. Ưu điểm của Raspberry Pi

Giá rẻ: chỉ từ 5 USD thôi là bạn đã mua được một cái Pi.

- Đơn giản, dễ dùng, tiết kiệm không gian: dùng làm các công việc văn phòng đơn giản, gõ Word, Excel hay tạo PowerPoint, lướt web.
- Tiêu thụ rất ít điện
- Có tính di động cao: có thể bỏ vào túi mang đi khắp nơi, thích hợp để làm máy nghe nhạc di động, máy đọc ebook, máy dò pass Wi-Fi, máy chơi game cầm tay.

Raspberry Pi 3 có cấu hình cao nhất hiện nay (trước đó có Pi, Pi 2, ngoài ra còn có bản siêu nhỏ: Pi Zero và Pi Zero W). Pi 3 có giá chỉ 35 USD nhưng thường là chúng ta sẽ mua thêm các phụ kiện ví dụ như case cho Pi 3, tản nhiệt cho chip, adapter nguồn, thẻ nhớ.

2.2.4. Ứng dụng của Raspberry Pi

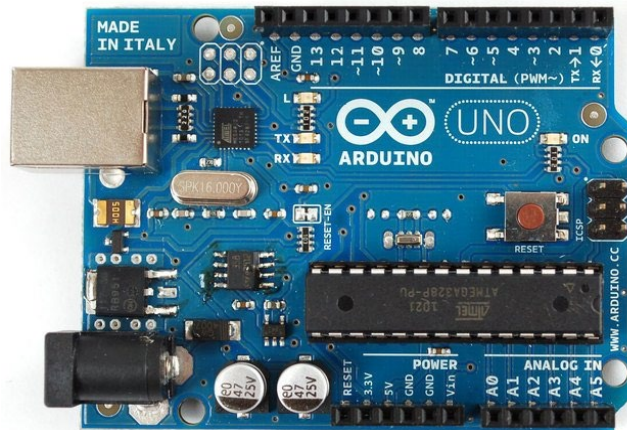
- Đầu coi phim HD giống như Android Box, hỗ trợ KODI đầy đủ.
- Máy chơi game cầm tay, console, game thùng. Chơi như máy điện tử bằng ngày xưa, giả lập được nhiều hệ máy.
- Cắm máy tải Torrent 24/24.
- Dùng làm VPN cá nhân.
- Biến ổ cứng bình thường thành ổ cứng mạng (NAS).
- Làm camera an ninh, quan sát từ xa.
- Hiện thị thời tiết, hiện thị thông tin mạng nội bộ...
- Máy nghe nhạc, máy đọc sách.
- Làm thành một cái máy Terminal di động có màn hình, bàn phím, pin dự phòng để sử dụng mọi lúc mọi nơi, dò pass Wi-Fi...
- Làm thiết bị điều khiển Smart Home, điều khiển mọi thiết bị điện tử trong nhà.
- Điều khiển robot, máy in không dây từ xa, Airplay...

2.3. Giới thiệu về Arduino

2.3.1. Khái niệm

Arduino thật ra là một bo mạch vi xử lý được dùng để lập trình với các thiết bị phân cứng như cảm biến, động cơ, đèn hoặc các thiết bị khác. Đặc điểm nổi bật của Arduino là môi trường phát triển ứng dụng cực kỳ dễ sử dụng, với một ngôn ngữ lập trình có thể học một cách nhanh chóng ngay cả với người mới bắt đầu vào lập trình. Và

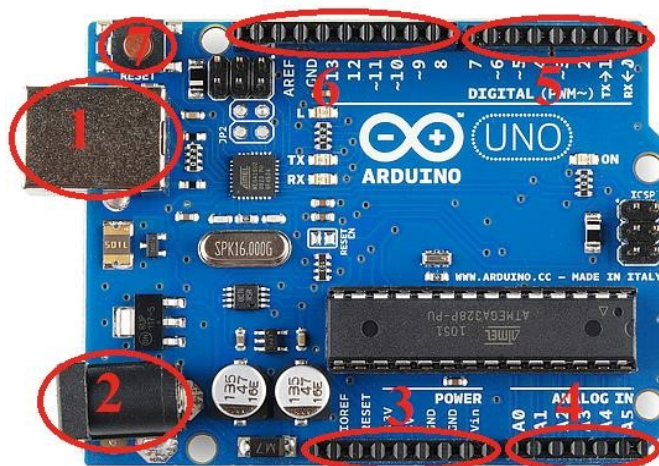
điều làm nên hiện tượng Arduino chính là mức giá rất thấp và tính chất nguồn mở từ phần cứng tới phần mềm.[8]



Hình 2.5. Arduino Uno

Arduino Uno là sử dụng chip Atmega328. Nó có 14 chân digital I/O, 6 chân đầu vào (input) analog.

2.3.2. Sơ đồ chân của Arduino



Hình 2.6. Arduino Uno.

a) Cổng USB (1).

Arduino sử dụng cáp USB để giao tiếp với máy tính. Thông qua cáp USB chúng ta có thể Upload chương trình cho Arduino hoạt động, ngoài ra USB còn là nguồn cho Arduino.

b) Nguồn (2 và 3).

Ngoài sử dụng USB làm nguồn thì chúng ta có thể sử dụng nguồn ngoài thông qua jack cắm 2.1mm (cực dương ở giữa) hoặc có thể sử dụng 2 chân Vin và GND để cấp nguồn cho Arduino.

Bo mạch hoạt động với nguồn ngoài ở điện áp từ 5 – 20 volt. Chúng ta có thể cấp một áp lớn hơn tuy nhiên chân 5V sẽ có mức điện áp lớn hơn 5 volt. Nếu sử dụng nguồn

lớn hơn 12 V thì sẽ có hiện tượng nóng và làm hỏng mạch. Khuyến cáo các bạn nên dùng nguồn ổn định là 5 đến dưới 12 volt.

Chân 5V và chân 3.3V (Output voltage): các chân này dùng để lấy nguồn ra từ nguồn mà chúng ta đã cung cấp cho Arduino. Lưu ý: không được cấp nguồn vào các chân này vì sẽ làm hỏng Arduino.

GND: chân mass.

c) Chip Atmega328.

Chip Atmega328 Có 32K bộ nhớ flash trong đó 0.5k sử dụng cho bootloader. Ngoài ra còn có 2K SRAM, 1K EEPROM.

d) Input và Output (4, 5 và 6).

Arduino Uno có 14 chân digital với chức năng input và output sử dụng các hàm pinMode (), digitalWrite () và digitalRead().

Cũng trên 14 chân digital này chúng ta còn một số chân chức năng đó là:

- Serial: chân 0 (Rx), chân 1 (Tx). Hai chân này dùng để truyền (Tx) và nhận (Rx) dữ liệu nối tiếp TTL. Chúng ta có thể sử dụng nó để giao tiếp với cổng COM của một số thiết bị hoặc các linh kiện có chuẩn giao tiếp nối tiếp.
- PWM (pulse width modulation): các chân 3, 5, 6, 9, 10, 11 trên bo mạch có dấu “~” là các chân PWM chúng ta có thể sử dụng nó để điều khiển tốc độ động cơ, độ sáng của đèn...
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK), các chân này hỗ trợ giao tiếp theo chuẩn SPI.
- I2C: Arduino hỗ trợ giao tiếp theo chuẩn I2C. Các chân A4 (SDA) và A5 (SCL) cho phép chúng ta giao tiếp giữa Arduino với các linh kiện có chuẩn giao tiếp là I2C.

e) Reset (7)

- Dùng để reset Arduino.

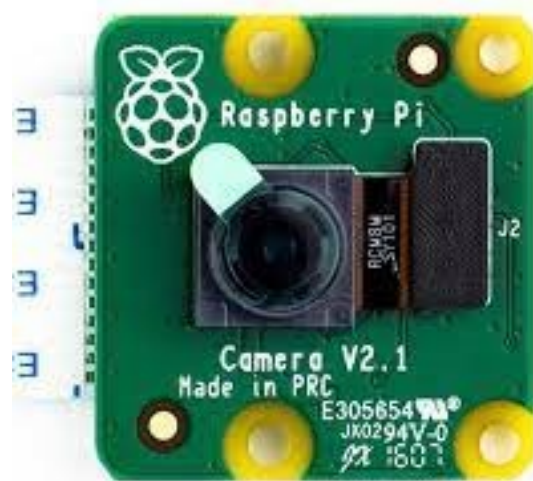
2.4. Giới thiệu về Camera Pi

2.4.1. Khái niệm

Raspberry Pi Camera là module camera được chính Raspberry Pi Foundation thiết kế và đưa vào sản xuất đại trà từ tháng 5/2013. Camera module ra đời đã thỏa lòng rất nhiều tín đồ yêu thích Raspberry. Trước khi xuất hiện camera, điều duy nhất bạn có thể làm để thêm khả năng nhận biết hình ảnh, quay phim, chụp hình cho RPi là sử dụng 1 webcam cắm vào cổng USB. Với các webcam Logitech tích hợp sẵn định dạng xuất mjpeg sẽ giúp Raspberry xử lý nhanh hơn. Nhưng các webcam Logitech lại có giá thành khá cao, nhất là các webcam có độ phân giải lớn. Bạn không tốn thêm cổng USB in ảo

cho Camera vì Camera được gắn chắc chắn vào socket CSI. Điều này giúp hạn chế tình trạng ngắn băng thông cho chip xử lý USB trên mạch Raspberry. Chiều dài cáp nối camera đã được tính toán cẩn thận khi vừa đạt được độ dài cần thiết trong khi vẫn đảm bảo tốc độ truyền hình ảnh từ module về Raspberry Pi.[9]

Raspberry Pi Camera Module V2 có một cảm biến 8 megapixel của Sony IMX219 (so với cảm biến 5 megapixel OmniVision 05647 trên Camera Module phiên bản cũ)



Hình 2.7. Camera Pi v2.1.

Camera Module có thể được sử dụng để quay video độ nét cao, cũng như chụp hình ảnh tĩnh. Nó cũng khá dễ dàng để sử dụng cho người mới bắt đầu, nhưng cũng có rất nhiều giải pháp mở rộng để cung cấp cho người dùng yêu cầu cao. Có rất nhiều demo của người dùng về công dụng của Camera Module như chụp Time-Lapse, SlowMotion và rất nhiều ứng dụng khác.

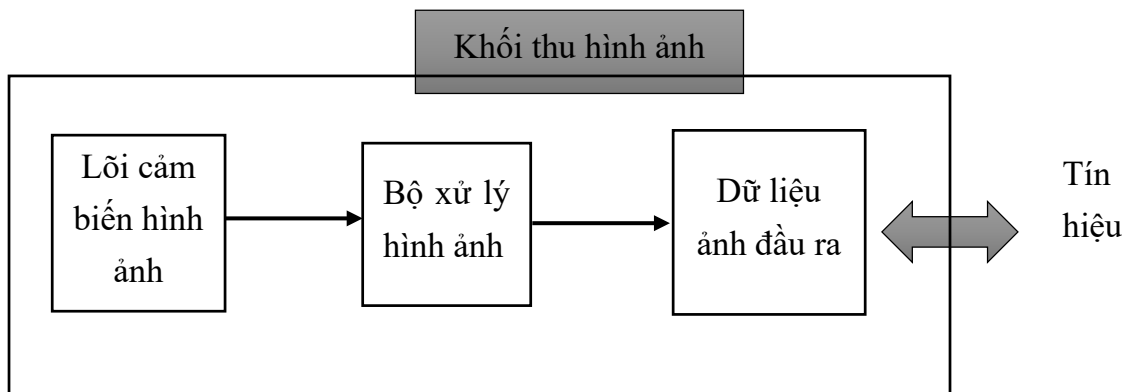
Raspberry Pi Camera Module V2 là một bước nhảy vọt về chất lượng hình ảnh, màu sắc trung thực và hiệu suất ánh sáng thấp. Đặc biệt nó hỗ trợ video lên tới 1080P30, 720P60 và video Iode VGA90, cũng như chế độ chụp hình. Dĩ nhiên, nó vẫn sử dụng đoạn cáp 15cm qua cổng CSI trên Raspberry Pi.

Thông tin cấu hình Camera Pi v2.1

- Ống kính tiêu cự cố định - Cảm biến độ phân giải 8 megapixel cho khả năng chụp ảnh kích thước 3280 x 2464.
- Hỗ trợ video 1080p30, 720p60 và 640x480290.
 - Kích thước 25mm x 23mm x 9mm.
 - Trọng lượng chỉ hơn 3g.

Kết nối với Raspberry Pi thông qua cáp ribbon đi kèm dài 15cm.

Camera Module được hỗ trợ với phiên bản mới nhất của Raspbian



Hình 2.8. Sơ đồ khối Camera Raspberry Pi

2.4.2. Ứng dụng

Bạn có thể dùng Camera và Raspberry để thiết lập hệ thống phát hiện chuyển động. Hệ thống hoạt động bằng cách sử dụng hình ảnh từ Camera kết hợp với chương trình xử lý hình ảnh và đưa ra lệnh điều khiển nếu phát hiện thấy có chuyển động. Đặc biệt hữu ích với các bạn yêu thích làm phim, bạn có thể sử dụng để quay những góc quay khó hoặc những cảnh quay độc mà chỉ với máy quay gọn nhẹ nhất mới làm được. Ngoài ra, bạn có thể sử dụng camera để quay các đoạn phim tinuue-lapse (ghép nhiều hình lại với nhau) đang được rất nhiều người dùng trên thế giới thực hiện.

2.5. Cảm biến hồng ngoại E18-D80NK

2.5.1. Khái quát

Cảm biến vật cản hồng ngoại E18-D80NK dùng ánh sáng hồng ngoại để xác định khoảng cách tới vật cản cho độ phản hồi nhanh và rất ít nhiễu do sử dụng mắt nhận và phát tia hồng ngoại theo tần số riêng biệt. Cảm biến có thể chỉnh khoảng cách bảo mong muốn thông qua biến trở, ngõ ra cảm biến ở dạng cực thu hỏ nên cần thêm 1 trở trờ lên nguồn ở chân tín hiệu khi sử dụng.[10]



Hình 2.9. Cảm biến hồng ngoại

2.5.2. Thông số kỹ thuật

- Nguồn điện cung cấp: 5V DC.
- Khoảng cách phát hiện vật: 3 - 80cm.
- Có thể điều chỉnh khoảng cách qua biến trở.
- Dòng kích ngõ ra: 300A.
- Ngõ ra dạng NPN cực thu hở giúp tùy biến được điện áp ngõ ra, trở treo lên áp
- bao nhiêu sẽ tạo thành điện áp ngõ ra bấy nhiêu.
- Chất liệu sản phẩm nhựa. Có led hiển thị ngõ ra màu đỏ.
- Kích thước: 1.8cm (D) x 7.0cm (L).

2.5.3. Sơ đồ chân

- Màu nâu: VCC, nguồn dương 5V DC.
- Màu xanh dương: GND, nguồn âm 0V DC.
- Màu đen: Chân tín hiệu ngõ ra cực thu hở NPN, cần phải có trở kéo để tạo thành mức cao

2.5.4. Ứng dụng

Các ứng dụng của cảm biến E18-D80NK: Xe tự động tránh vật cản, trong các dây chuyền sản xuất tự động.

2.6. Động cơ DC

2.6.1. Khái quát

Động cơ một chiều DC (DC là từ viết tắt của "Direct Current Motors") là động cơ điều khiển bằng dòng có hướng xác định hay nói dễ hiểu hơn thì đây là loại động cơ chạy bằng nguồn điện áp DC- điện áp 1 chiều (khác với điện áp AC xoay chiều). Động cơ thường gồm hai dây (dây nguồn- VCC và dây tiếp đất- GND).

DC là một động cơ một chiều với cơ năng quay liên tục.



Hình 2.10. Động cơ DC

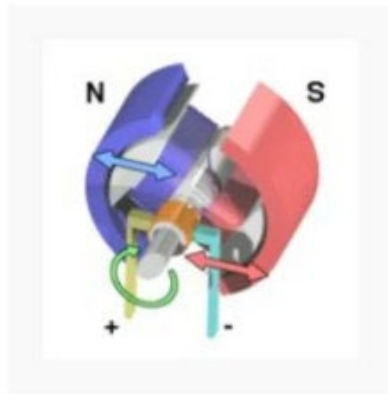
2.6.2. Cấu tạo và nguyên lý hoạt động

a) Cấu tạo

Gồm có 3 phần chính stator (phần cảm), rotor (phần ứng), và phần cổ góp chỉnh lưu.

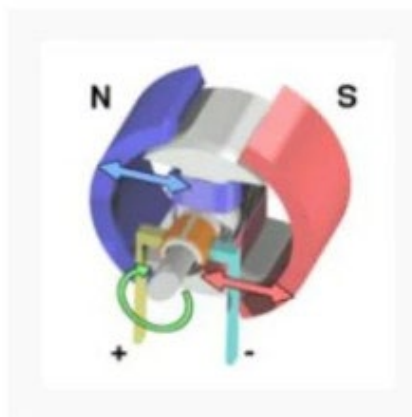
- Stator của động cơ điện 1 chiều thường là 1 hay nhiều cặp nam châm vĩnh cửu, hay nam châm điện.
- Rotor có các cuộn dây quấn và được nối với nguồn điện một chiều.
- Bộ phận chỉnh lưu, nó có nhiệm vụ là đổi chiều dòng điện trong khi chuyển động quay của rotor là liên tục. Thông thường bộ phận này gồm có một bộ cổ góp và một bộ chổi than tiếp xúc với cổ góp.

b) Nguyên lý hoạt động



Hình 2.11. Pha 1 động cơ DC

Pha 1: Từ trường của Rotor cùng cực với stator, sẽ đẩy nhau tạo ra chuyển động quay của Rotor.

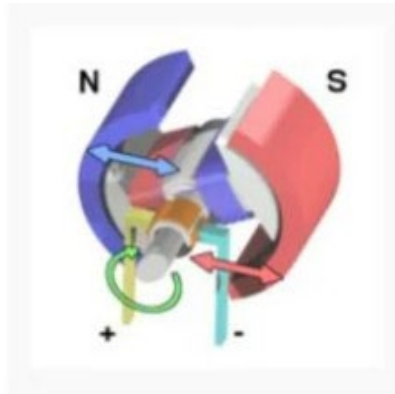


Hình 2.12. Pha 2 động cơ DC

Pha 2: Rotor tiếp tục quay

Pha 3: Bộ phận chỉnh điện sẽ đổi cực sao cho từ trường giữa stator và rotor cùng dấu, trở lại pha 1. Nếu trục của một động cơ điện một chiều được kéo bằng 1 lực ngoài, động cơ sẽ hoạt động như một máy phát điện một chiều, và tạo ra một sức điện động

cảm ứng Electromotive force (EMF). Khi vận hành bình thường, rotor khi quay sẽ phát ra một điện áp gọi là sức phản điện động counter-EMF (CEMF) hoặc sức điện động đối



Hình 2.13. Pha 3 động cơ DC

kháng, vì nó đối kháng lại điện áp bên ngoài đặt vào động cơ. Sức điện động này tương tự như sức điện động phát ra khi động cơ được sử dụng như một máy phát điện (như lúc ta nối một điện trở tải vào đầu ra của động cơ, và kéo trục động cơ bằng một ngẫu lực bên ngoài). Như vậy điện áp đặt trên động cơ bao gồm 2 thành phần: sức phản điện động, và điện áp giáng tạo ra do điện trở nội của các cuộn dây phần ứng. Dòng điện chạy qua động cơ được tính theo biểu thức sau:

$$I = (V_{\text{nguồn}} - V_{\text{Phần điện động}}) / R_{\text{Phần cứng}}$$

Công suất cơ mà động cơ đưa ra được, được tính bằng:

$$P = I * (V_{\text{phần điện động}})$$

2.6.3. Ứng dụng

Ngày nay động cơ điện DC được dùng trong hầu hết mọi lĩnh vực, từ các động cơ nhỏ dùng trong lò vi sóng để chuyển động đĩa quay, hay trong các máy đọc đĩa (máy chơi CD hay DVD), đến các đồ nghề như máy khoan, hay các máy gia dụng như máy giặt, sự hoạt động của thang máy hay các hệ thống thông gió cũng dựa vào động cơ điện. Ở nhiều nước động cơ điện được dùng trong các phương tiện vận chuyển, đặc biệt trong các đầu máy xe lửa.

Trong công nghệ máy tính: Động cơ điện được sử dụng trong các ổ cứng, ổ quang (chúng là các động cơ bước rất nhỏ).

2.7. Động cơ Servo MG996G

2.7.1. Giới thiệu động cơ Servo MG996G

Động cơ RC Servo MG996 là loại thường được sử dụng nhiều nhất trong các thiết kế Robot hoặc dẫn hướng xe. Động cơ RC Servo MG996 có lực kéo mạnh, các khớp và bánh răng được làm hoàn toàn bằng kim loại nên có độ bền cao, động cơ được tích hợp sẵn Driver điều khiển động cơ bên trong theo cơ chế phát xung - quay góc nên rất dễ sử dụng.



Hình 2.14. Động cơ servo MG996R

Thông số kỹ thuật

- Chung loại: Analog RC Servo.
- Điện áp hoạt động: 4.8-6.6VDC
- Kích thước: 40mm x 20mm x 43mm
- Trọng lượng: 55g
- Lực kéo:
 - 3.5 kg-cm (180.5 ozin) at 4.8V-1.5A
 - 5.5 kg-cm (208.3 ozin) at 6V-1.5A
- Tốc độ quay:
 - 0.17sec / 60 degrees (4.8V no load)
 - 0.13sec / 60 degrees (6.0V no load)

2.7.2. Ứng dụng

Trong ngành công nghiệp động cơ servo được sử dụng trong các máy công cụ, đóng gói, tự động hóa nhà máy, xử lý vật liệu, in chuyển đổi. Ngoài ra chúng còn được sử dụng để chế tạo cánh tay robot vì sử dụng chuyển đổi trơn tru và chính xác.

2.8. Hệ thống băng tải

2.8.1. Giới thiệu

Băng tải (băng chuyền) hiểu đơn giản là một máy cơ khí dùng để vận chuyển các đồ vật từ điểm này sang điểm khác, từ vị trí A sang vị trí B. Thay vì vận chuyển sản phẩm bằng công nhân vừa tốn thời gian, chi phí nhân công lại tạo ra môi trường làm việc lộn xộn thì băng chuyền tải có thể giải quyết điều đó. Nó giúp tiết kiệm sức lao động, số lượng nhân công, giảm thời gian và tăng năng suất lao động.[11]

Vì vậy băng chuyền, băng tải là một trong những bộ phận quan trọng trong dây chuyền sản xuất, lắp ráp của các nhà máy, xí nghiệp. Góp phần tạo nên một môi trường sản xuất hiện đại, khoa học và giải phóng sức lao động mang lại hiệu quả kinh tế cao cho công ty.



Hình 2.15. Băng tải

2.8.2. Cấu tạo

- Khung băng tải: thường được làm bằng nhôm định hình, thép sơn tĩnh điện hoặc inox
- Dây băng tải: Thường là dây băng PVC dày 2mm và 3mm hoặc dây băng PU dày 1.5mm.
- Động cơ chuyển động: Là động cơ giảm tốc công suất 0.2KW, 0.4KW, 0.75KW, 1.5KW, 2.2KW.
- Bộ điều khiển băng chuyền: Thường gồm có biến tần, sensor, timer, PLC...
- Cơ cấu truyền động gồm có: Rulo kéo, con lăn đỡ, không xích...
- Hệ thống bàn thao tác trên băng chuyền thường bằng gỗ, thép hoặc inox trên mặt có dán thảm cao su chống tĩnh điện.
- Hệ thống đường khí nén và đường điện có ổ cắm để lấy điện cho các máy dung trên băng chuyền.

2.8.3. Ứng dụng

Băng tải, băng chuyền được ứng dụng trong các ngành sản xuất công nghiệp, chế biến thực phẩm, khai thác...nhằm giúp hỗ trợ tối đa cho doanh nghiệp trong quá trình vận chuyển hàng hóa nhanh chóng, an toàn, tiện lợi. Ngoài ra với hệ thống băng tải còn giúp doanh nghiệp tối ưu hóa được chi phí, tiết kiệm thời gian, hạn chế nguồn nhân lực mang lại hiệu quả kinh tế cao đồng thời còn giúp cho hệ thống sản xuất ngày càng được tự động hóa theo hướng hiện đại.

2.9. LCD

2.9.1. Giới thiệu

Ngày nay, thiết bị hiển thị LCD (Liquid Crystal Display) được sử dụng trong rất nhiều các ứng dụng của VĐK. LCD có rất nhiều ưu điểm so với các dạng hiển thị khác: Nó có khả năng hiển thị kí tự đa dạng, trực quan (chữ, số và kí tự đồ họa), dễ dàng đưa

vào mạch ứng dụng theo nhiều giao thức giao tiếp khác nhau, tốn rất ít tài nguyên hệ thống và giá thành rẻ...[12]



Hình 2.16. LCD 16x02

2.9.2. Cấu tạo

Cấu tạo LCD 16*2 có 2 hàng mỗi hàng 16 chân



Hình 2.17. Cấu tạo LCD

Trong 16 chân của LCD được chia làm 3 dạng tín hiệu như sau:

- Các chân cấp nguồn: chân số 1 nối mass (0V), chân số 2 là VDD nối với nguồn 5V, chân số 3 dùng để chỉnh contrast thường nối với biến trở.
- các chân điều khiển: chân số 4 là chân RS dùng để điều khiển lựa chọn thanh ghi. Chân R/W dùng để điều khiển quá trình đọc và ghi. Chân E là chân cho phép dạng xung chốt.
- Các chân dữ liệu DB0-DB7: là chân từ số 7 đến 14 dùng để trao đổi dữ liệu giữa thiết bị điều khiển và LCD.

- Chân 15 nối nguồn +5V hoặc 4,2V đối với LED, chân 16 nối GND.

Chân	Kí hiệu	Mô tả
1	VSS	Chân nối đất cho LCD, khi thiết kế mạch ta nối chân này với GND của mạch điều khiển
2	VDD	Chân cấp nguồn cho LCD, khi thiết kế mạch ta nối chân này với VCC=5V của mạch điều khiển
3	VEE	Điều chỉnh độ tương phản của LCD
4	RS	Chân chọn thanh ghi (Register select). Nối chân RS với logic “0” (GND) hoặc logic “1” (VCC) để chọn thanh ghi. + Logic “0”: Bus DB0-DB7 sẽ nối với thanh ghi lệnh IR của LCD (ở chế độ “ghi” - write) hoặc nối với bộ đếm địa chỉ của LCD (ở chế độ “đọc” - read) + Logic “1”: Bus DB0-DB7 sẽ nối với thanh ghi dữ liệu DR bên trong LCD
5	R/W	Chân chọn chế độ đọc/ghi (Read/Write). Nối chân R/W với logic “0” để LCD hoạt động ở chế độ ghi, hoặc nối với logic “1” để LCD ở chế độ đọc
6	E	Chân cho phép (Enable). Sau khi các tín hiệu được đặt lên bus các lệnh chỉ được chấp nhận khi có 1 xung cho phép của chân E. + Ở chế độ ghi: Dữ liệu ở bus sẽ được LCD chuyển vào(chấp nhận) thanh ghi bên trong nó khi phát hiện một xung (high-to-low transition) của tín hiệu chân E. Ở chế độ đọc: Dữ liệu sẽ được LCD xuất ra DB0-DB7 khi phát hiện cạnh lên (low-to-high transition) ở chân E và được LCD giữ ở bus đến khi nào chân E xuống mức thấp
7- 14	DB0-DB7	Tám đường của bus dữ liệu dùng để trao đổi thông tin với MPU, Có 2 chế độ sử dụng 8 đường bus này: + Chế độ 8 bit: Dữ liệu được truyền trên cả 8 đường, với bit MSB là bit DB7. + Chế độ 4 bit : Dữ liệu được truyền trên 4 đường từ DB4 tới DB7, bit MSB là DB7
15		Nguồn dương cho đèn nền
16		GND cho đèn nền

Bảng 2-1: Cấu tạo LCD 16x02

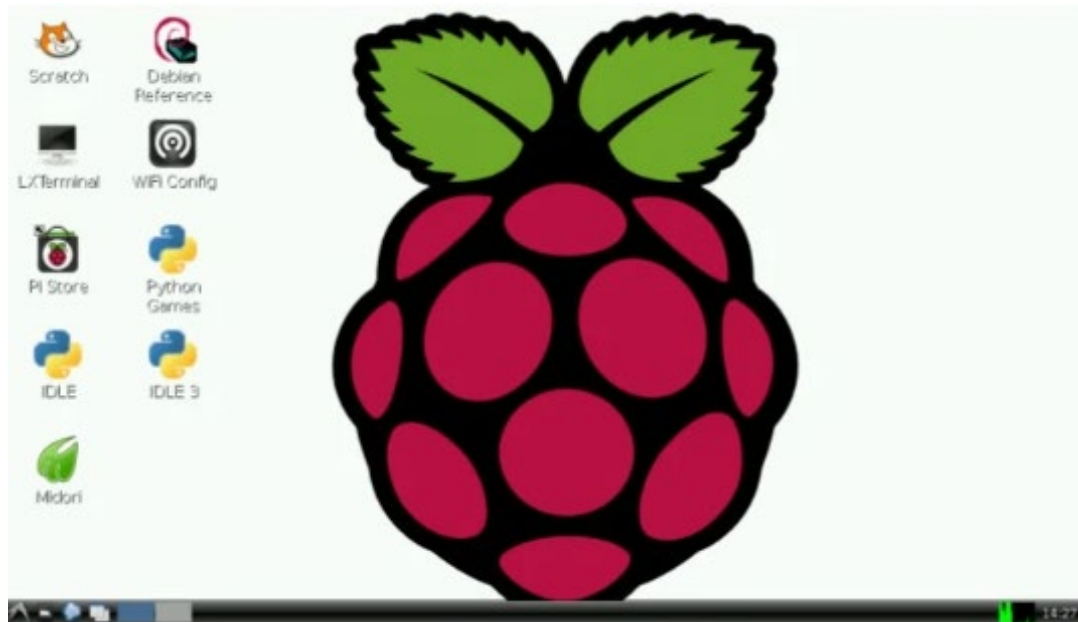
2.9.3. Ứng dụng

LCD thường được sử dụng trong các mạch điện tử, hiển thị thời gian thực, giá trị kết quả, hiệu ứng.

2.10. Hệ điều hành trên Raspberry Pi

Raspberry Pi có rất nhiều hệ điều hành hỗ trợ như Raspbian, Ubuntu Mate, PiNet, RiscOS... nhưng trong đó Raspbian là hệ điều hành cơ bản, phổ biến nhất và do chính Raspberry Pi Foundation cung cấp. Nó cũng được hãng khuyến cáo sử dụng, nhất là cho người mới bắt đầu làm quen với Raspberry Pi.[13]

Raspbian có dung lượng sau khi giải nén là khoảng gần 4GB, bạn cần tối thiểu 1 cái thẻ 4GB để có thể sử dụng Raspbian. Tuy nhiên, nên sử dụng thẻ tối thiểu 8GB vì bạn cần cài thêm các ứng dụng khác nữa



Hình 2.18. Hệ điều hành trên Raspberry

- Raspbian được hướng đến người dùng có mục đích:
- Sử dụng Raspberry Pi như máy tính văn phòng để lướt web, soạn văn bản, check mail và thi thoảng nghe nhạc/xem phim.
- Nghiên cứu phát triển các thiết bị điều khiển tự động.
- Sử dụng như một máy chủ cung cấp các dịch vụ như web, file server, printer, server, ...

Theo đánh giá, Raspbian hoạt động rất ổn định, tốc độ nhanh (đặc biệt là trên Raspberry Pi 3). Nhược điểm của nó là giao diện đơn giản, cổ điển và rất không hào nhoáng. Nếu bạn không quá quan tâm tới giao diện mà hướng đến hiệu năng thì Raspbian rất phù hợp cho bạn.

2.11. Giới thiệu ngôn ngữ Python và thư viện OpenCV

2.11.1. Ngôn ngữ Python

a) Giới thiệu ngôn ngữ Python

Python là một ngôn ngữ lập trình được sử dụng phổ biến ngày nay từ trong môi trường học đường cho tới các dự án lớn. Ngôn ngữ phát triển nhiều loại ứng dụng, phần mềm khác nhau như các chương trình chạy trên desktop, server, lập trình các ứng dụng web... Ngoài ra Python cũng là ngôn ngữ ưa thích trong xây dựng các chương trình trí tuệ nhân tạo trong đó bao gồm machine learning. Ban đầu, Python được phát triển để chạy trên nền Unix, nhưng sau này, nó đã chạy trên mọi hệ điều hành từ MS-DOS đến

Mac OS, 6S/2, Windows, Linux và các hệ điều hành khác thuộc họ Unix. Python do Guido van Rossum tạo ra năm 1990. Python được phát triển trong một dự án mã mở, do tổ chức phi lợi nhuận Python Software Foundation quản lý. Mặc dù sự phát triển của Python có sự đóng góp của rất nhiều cá nhân, nhưng Guido van Rossum hiện nay vẫn là tác giả chủ yếu của Python. Ông giữ vai trò chủ chốt trong việc quyết định hướng phát triển của Python.

b) Đặt điểm nổi bật của Python

Python là ngôn ngữ có hình thức đơn giản, cú pháp ngắn gọn, sử dụng một số lượng ít các từ khoá, do đó Python là một ngôn ngữ dễ học đối với người mới bắt đầu tìm hiểu. Python là ngôn ngữ có mã lệnh (source code hay đơn giản là code) không mấy phức tạp. Cả trường hợp bạn chưa biết gì về Python bạn cũng có thể suy đoán được ý nghĩa của từng dòng lệnh trong source code. Python có nhiều ứng dụng trên nhiều nền tảng, chương trình phần mềm viết bằng ngôn ngữ Python có thể được chạy trên nhiều nền tảng hệ điều hành khác nhau bao gồm Windows, Mac OSX và Linux.

2.11.2. Thư viện OpenCV

a) Giới thiệu

OpenCV (Open Source Computer Vision Library) là một thư viện mã nguồn mở, nó là miễn phí cho những ai bắt đầu tiếp cận với các học thuật. OpenCV được ứng dụng trong nhiều lĩnh vực như cho thị giác máy tính hay xử lý ảnh và máy học. Thư viện được lập trình trên các ngôn ngữ cấp cao: C++, C, Python, hay Java và hỗ trợ trên các nền tảng Window, Linux, Mac OS, iOS và Android. OpenCV đã được tạo ra tại Intel vào năm 1999 bởi Gary Bradsky, và ra mắt vào năm 2000. OpenCV có rất nhiều ứng dụng: Nhận dạng ảnh, xử lý hình ảnh, phục hồi hình ảnh/video, thực tế ảo... Ở đề tài này thư viện OpenCV được chạy trên ngôn ngữ Python. OpenCV được dùng làm thư viện chính để xử lý hình ảnh đầu vào và sau đó đi nhận dạng ảnh.[14]

b) Đặc điểm

OpenCV Là một thư viện mở nên sử dụng các thuật toán một cách miễn phí, cùng với việc chúng ta cũng có thể đóng góp thêm các thuật toán giúp Thư viện thêm ngày càng phát triển.

Các tính năng của thư viện OpenCV:

- Đối với hình ảnh, chúng ta có thể đọc và lưu hay ghi chúng.
- Về Video cũng tương tự như hình ảnh cũng có đọc và ghi.
- Xử lý hình ảnh có thể lọc nhiễu cho ảnh, hay chuyển đổi ảnh.
- Thực hiện nhận dạng đặc điểm của hình dạng trong ảnh.
- Phát hiện các đối tượng xác định được xác định trước như khuôn mặt, mắt, xe trong video hoặc hình ảnh.

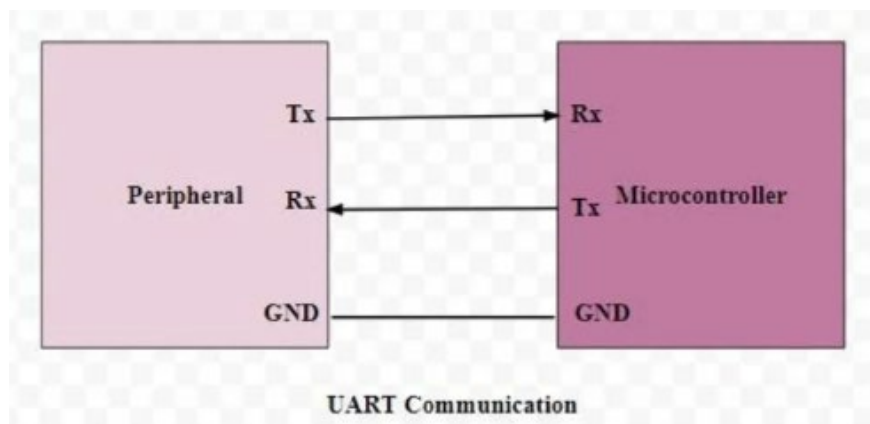
- Phân tích video ước lượng chuyển động của nó, trừ nền ra và theo dõi các đối tượng trong video.

2.12. Các chuẩn giao tiếp

2.12.1. Chuẩn giao tiếp UART

a) Giới thiệu

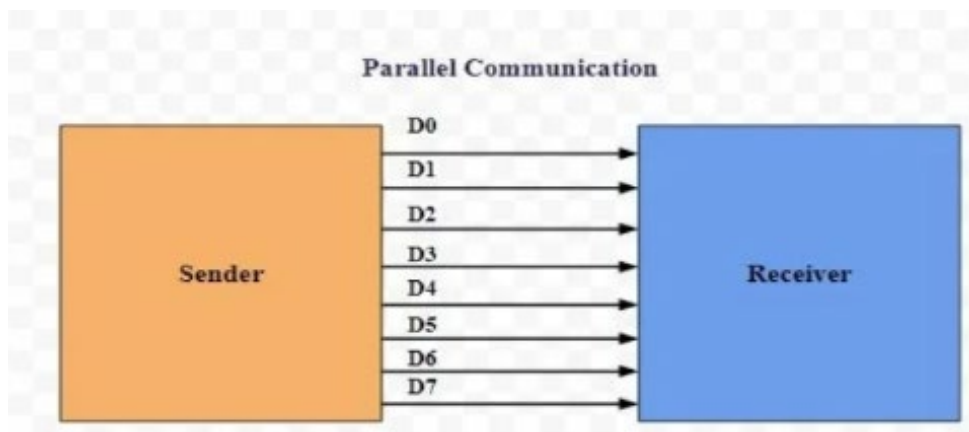
Các tên đầy đủ UART là “Universal Asynchronous Receiver / Transmitter”, và nó là một vi mạch sẵn có trong một vi điều khiển nhưng không giống như một giao thức truyền thông (I2C & SPI). Chức năng chính của UART là truyền dữ liệu nối tiếp. Trong UART, giao tiếp giữa hai thiết bị có thể được thực hiện theo hai cách là giao tiếp dữ liệu nối tiếp và giao tiếp dữ liệu song song.[15]



Hình 2.19. Giao tiếp UART

Trong giao tiếp dữ liệu nối tiếp, dữ liệu có thể được truyền qua một cái hoặc một đường dây ở dạng bit-bit và nó chỉ cần hai cáp. Truyền thông dữ liệu nối tiếp không đắt khi chúng ta so sánh với giao tiếp song song. Nó đòi hỏi rất ít mạch cũng như dây. Vì vậy, giao tiếp này rất hữu ích trong các mạch ghép so với giao tiếp song song

Trong giao tiếp dữ liệu song song, dữ liệu có thể được truyền qua nhiều cáp cùng một lúc. Truyền dữ liệu song song tốn kém nhưng rất nhanh, vì nó đòi hỏi phản cứng và cáp bổ sung. Các ví dụ tốt nhất cho giao tiếp này là máy in cũ, PCI, RAM, vv

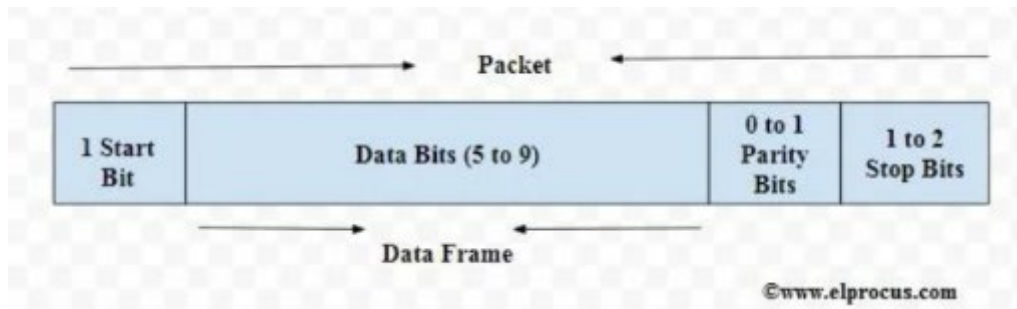


Hình 2.20. Truyền thông nối tiếp

b) Truyền thông UART

Trong giao tiếp này, có hai loại UART có sẵn là truyền UART và nhận UART và giao tiếp giữa hai loại này có thể được thực hiện trực tiếp với nhau. Đối với điều này, chỉ cần hai cáp để giao tiếp giữa hai UART, Luồng dữ liệu sẽ từ cả hai chân truyền (Tx) và nhận (Rx) của UARTs. Trong UART, việc truyền dữ liệu từ Tx UART sang Rx UART có thể được thực hiện không đồng bộ (không có tín hiệu CLK để đồng bộ hóa các bit o/p).

Việc truyền dữ liệu của UART có thể được thực hiện bằng cách sử dụng bus dữ liệu ở dạng song song bởi các thiết bị khác như vi điều khiển, bộ nhớ, CPU, vv. Sau khi nhận được dữ liệu song song từ bus, nó tạo thành gói dữ liệu bằng cách thêm ba bit như bắt đầu, dừng lại và trung bình. Nó đọc từng bit gói dữ liệu và chuyển đổi dữ liệu nhận được thành dạng song song để loại bỏ ba bit của gói dữ liệu. Tóm lại, gói dữ liệu nhận được bởi UART chuyển song song về phía bus dữ liệu ở đầu nhận.



Hình 2.21. UART

Start-bit: còn được gọi là bit đồng bộ hóa được đặt trước dữ liệu thực tế. Nói chung, một đường truyền dữ liệu không hoạt động được điều khiển ở mức điện áp cao. Để bắt đầu truyền dữ liệu, truyền UART kéo đường dữ liệu từ mức điện áp cao (1) xuống mức điện áp thấp (0). UART thu được thông báo sự chuyển đổi này từ mức cao sang mức thấp qua đường dữ liệu cũng như bắt đầu hiệu dữ liệu thực. Nói chung, chỉ có một start-bit.

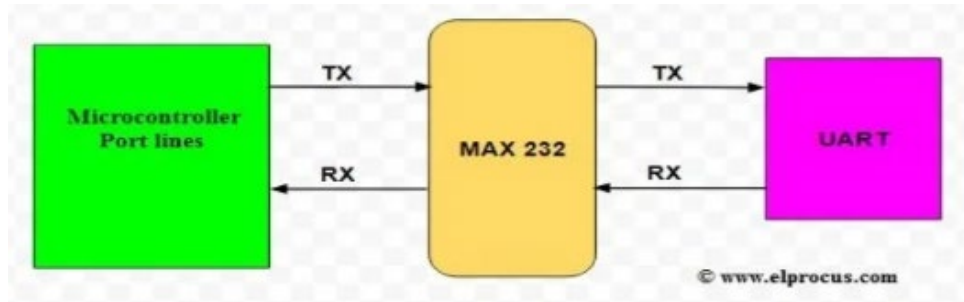
Bit dừng: được đặt ở phần cuối của gói dữ liệu. Thông thường, bit này dài 2 bit nhưng thường chỉ sử dụng 1 bit. Để dừng sóng, UART giữ đường dữ liệu ở mức điện áp cao.

Bit chẵn lẻ: cho phép người nhận đảm bảo liệu dữ liệu được thu thập có đúng hay không. Đây là một hệ thống kiểm tra lỗi cấp thấp & bit chẵn lẻ có sẵn trong hai phạm vi như Chẵn lẻ - chẵn lẻ cũng như Chẵn lẻ - lẻ. Trên thực tế, bit này không được sử dụng rộng rãi nên không bắt buộc.

Dữ liệu bit hoặc khung dữ liệu: Các bit dữ liệu bao gồm dữ liệu thực được truyền từ người gửi đến người nhận. Độ dài khung dữ liệu có thể nằm trong khoảng 5 & 8. Nếu bit chẵn lẻ không được sử dụng thì chiều dài khung dữ liệu có thể dài 9 bit. Nói

chung, LSB của dữ liệu được truyền trước tiên sau đó nó rất hữu ích cho việc truyền.

Giao diện UART: Hình dưới đây cho thấy UART giao tiếp với vi điều khiển. Giao tiếp UART có thể được thực hiện bằng ba tín hiệu như TXD, RXD và GND.



Hình 2.22. Giao tiếp UART

c) Ứng dụng của UART

UART thường được sử dụng trong các bộ vi điều khiển cho các yêu cầu chính xác và chúng cũng có sẵn trong các thiết bị liên lạc khác nhau như giao tiếp không dây, thiết bị GPS, mô-đun Bluetooth và nhiều ứng dụng khác.

Các tiêu chuẩn truyền thông như RS422 & TIA được sử dụng trong UART ngoại trừ RS232. Thông thường, UART là một IC riêng được sử dụng trong giao tiếp nối tiếp UART

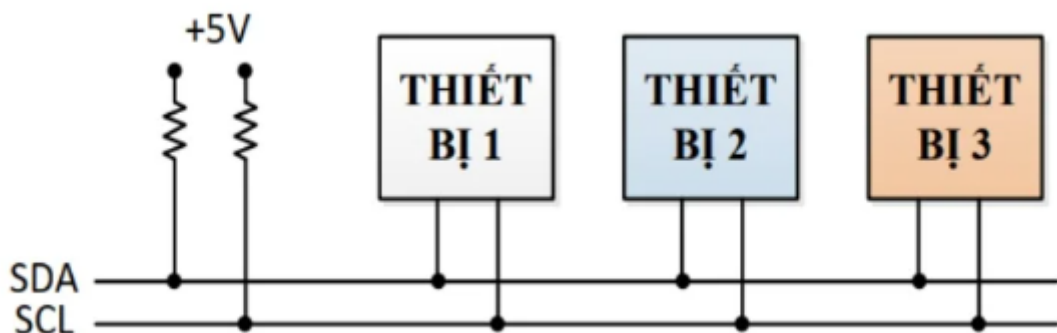
d) Ưu điểm và nhược điểm của UART

Những ưu và nhược điểm của UART bao gồm những điều sau đây:

- Nó chỉ cần hai dây để truyền dữ liệu
- Tín hiệu CLK là không cần thiết.
- Nó bao gồm một bit chẵn lẻ để cho phép kiểm tra lỗi
- Sắp xếp gói dữ liệu có thể được sửa đổi vì cả hai mặt được sắp xếp
- Kích thước khung dữ liệu tối đa là 4 bit
- Nó không chứa một số hệ thống phụ (hoặc)
- Tốc độ truyền của UART phải ở mức 10% của nhau

2.12.2. Chuẩn giao tiếp I2C

a) Giới thiệu



Hình 2.23. Giao tiếp I2C

I2C viết tắt của từ Inter-Integrated Circuit là một chuẩn truyền thông do hãng điện tử Philips Semiconductor sáng lập cho phép giao tiếp một thiết bị chủ với nhiều thiết bị tớ với nhau như hình.

Chuẩn giao tiếp I2C có 2 đường tín hiệu tên là SDA (serial data) có chức năng truyền tải dữ liệu và tín hiệu SCL (serial clock) truyền tải xung clock để dịch chuyển dữ liệu.[16]

Trong hệ thống truyền dữ liệu IDC thì thiết bị nào cung cấp xung clock thì được gọi là chủ (master), thiết bị nhận xung clock được gọi là tớ (slave).

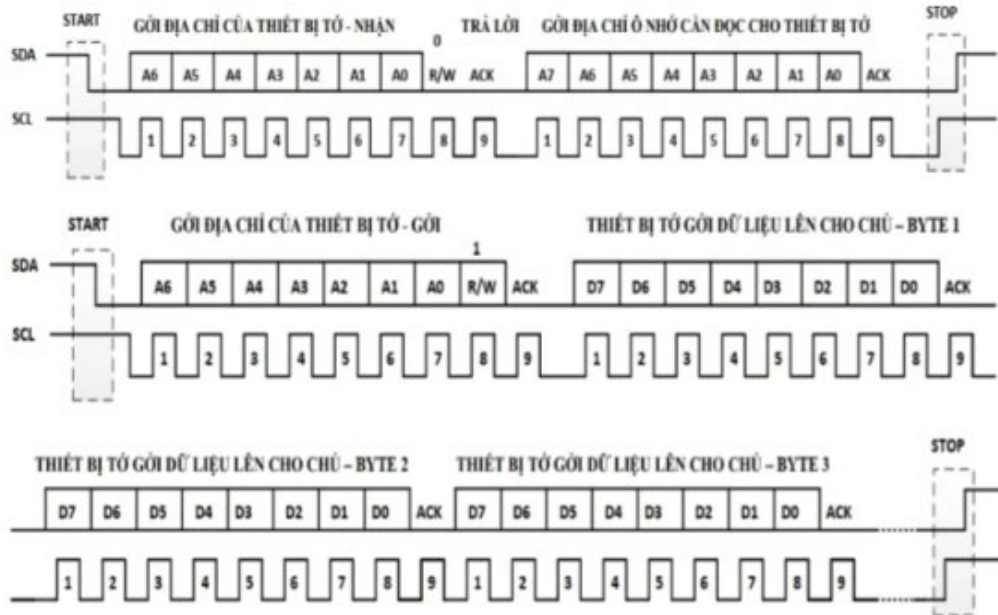
Thiết bị chủ chỉ có 1, thiết bị tớ thì có nhiều, mỗi thiết bị tớ sẽ có 1 địa chỉ độc lập, chuẩn truyền ban đầu dùng địa chỉ 7 bit nên có thể 1 chủ giao tiếp với 128 thiết bị tớ. Các thiết bị sau này tăng thêm số bit nên có thể giao tiếp nhiều hơn.

Giao diện I2C hỗ trợ tốc độ truyền chuẩn 100kHz hay tốc độ cao 400kHz. Ngoài ra còn hỗ trợ 7 hoặc 10 bit địa chỉ. Được thiết kế nhằm đơn giản hóa quá trình trao đổi với 2 kênh DMA cho truyền và nhận dữ liệu

b) Quy trình truyền dữ liệu chuẩn I2C

Quá trình thiết bị chủ ghi dữ liệu vào thiết bị tớ:

- Bước 1: Thiết bị chủ tạo trạng thái START để bắt đầu quá trình truyền dữ liệu - các thiết bị tớ sẽ ở trạng thái sẵn sàng nhận địa chỉ từ thiết bị chủ.
- Bước 2: Thiết bị chủ gửi địa chỉ của thiết bị tớ cần giao tiếp – khi đó tất cả các thiết bị tớ đều nhận địa chỉ và so sánh với địa chỉ của mình, các thiết bị tớ sau khi phát hiện không phải địa chỉ của mình thì chờ cho đến khi nào nhận trạng thái START mới. - Trong dữ liệu 8 bit thì có 7 bit địa chỉ và 1 bit điều khiển đọc/ghi (R/W): thì bit này bằng 0 để báo cho thiết bị tớ sẽ nhận byte tiếp theo.
- Bước 3: Thiết bị chủ chờ nhận tín hiệu bắt tay từ thiết bị tớ. Thiết bị tớ nào đúng địa chỉ thì phát 1 tín hiệu trả lời cho chủ biết.
- Bước 4: Thiết bị chủ tiến hành gửi địa chỉ của ô nhớ bắt đầu cần ghi dữ liệu, bit R/W ở trạng thái ghi.
- Bước 5: Thiết bị chủ chờ nhận tín hiệu trả lời từ thiết bị tớ.
- Bước 6: Thiết bị chủ tiến hành gửi dữ liệu để ghi vào thiết bị tớ, mỗi lần ghi 1 byte, sau khi gửi xong thì tiến hành chờ nhận tín hiệu trả lời từ thiết bị tớ, quá trình thực hiện cho đến byte cuối cùng xong rồi thì thiết bị chủ chuyển sang trạng thái STOP để chấm dứt quá trình giao tiếp với thiết bị tớ.



Hình 2.24. Dữ liệu I2C

Quá trình thiết bị chủ đọc dữ liệu vào thiết bị tớ:

- Bước 1: Thiết bị chủ tạo trạng thái START để bắt đầu quá trình truyền dữ liệu các thiết bị tớ sẽ ở trạng thái sẵn sàng nhận địa chỉ từ thiết bị chủ. -
- Bước 2: Thiết bị chủ gửi địa chỉ của thiết bị tớ cần giao tiếp khi đó tất cả các thiết bị tớ đều nhận địa chỉ và so sánh với địa chỉ mình, các thiết bị tớ sau khi phát hiện không phải địa chỉ của mình thì chờ cho đến khi nào nhận trạng thái START mới. Trong dữ liệu 8 bit thì có 7 bit địa chỉ và 1 bit điều khiển đọc/ghi (R/W): thì bit này bằng 0 để báo cho thiết bị tớ sẽ nhận byte tiếp theo.
- Bước 3: Thiết bị chủ chờ nhận tín hiệu bắt tay từ thiết bị tớ. Thiết bị tớ nào đúng địa chỉ thì phát 1 tín hiệu trả lời cho chủ biết.
- Bước 4: Thiết bị chủ tiến hành gửi địa chỉ của ô nhớ bắt đầu cần đọc dữ liệu, bit R/W ở trạng thái đọc.
- Bước 5: Thiết bị chủ chờ nhận tín hiệu trả lời từ thiết bị tớ
- Bước 6: Thiết bị chủ chuyển sang trạng thái STOP, bắt đầu lại trạng thái START, tiến hành gửi dữ liệu của thiết bị và bit RW bằng 1 để yêu cầu tớ gửi dữ liệu nội dung ô nhớ của địa chỉ đã nhận.
- Bước 7: Thiết bị chủ sau khi nhận sẽ báo tín hiệu trả lời, quá trình này thực hiện cho đến khi hết dữ liệu mong muốn thì thiết bị chủ tạo tín hiệu STOP để chấm dứt

c) Ứng dụng

Thường được dùng trong các khối truyền thông nối tiếp của vi điều khiển.

Chương 3: Tính toán và thiết kế

3.1. Giới thiệu

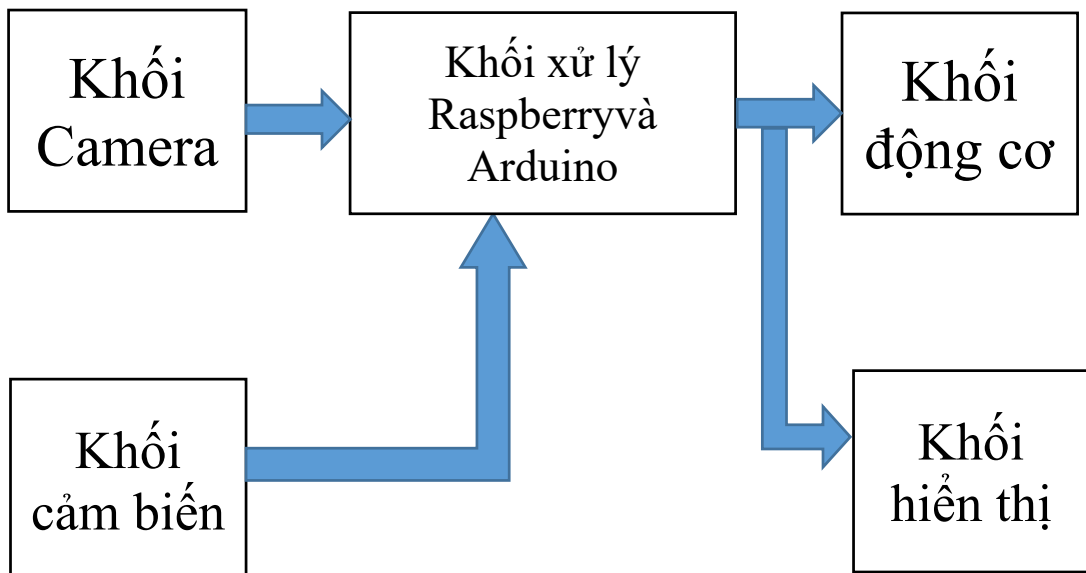
Ngày nay, phần lớn các thiết bị điện tử phát triển theo xu hướng tự động hóa, thông minh, chúng giao tiếp được với con người và hoạt động theo ý muốn con người, để làm được điều đó thì các thiết bị cảm biến, thuật toán hỗ trợ ra đời.

Đề tài “Ứng dụng xử lý ảnh trong hệ thống phân loại sản phẩm” với mục tiêu là phân loại sản phẩm theo màu sắc (đỏ, xanh, vàng) và các hình dạng. Dựa trên ngôn ngữ Python với thư viện chính là OpenCV, được thực hiện với bộ xử lý trung tâm là kit Raspberry và kit Arduino Uno, khối thu tín hiệu hình ảnh từ bên ngoài là Camera Raspberry Pi, điều khiển cơ cấu chấp hành, sau đó hiển thị kết quả trên màn hình LCD.

Đề tài chỉ xây dựng với mô hình nhỏ và gọn không làm thành dây chuyền sản xuất. Thời gian và tốc độ xử lý còn chậm, cùng với Camera chụp ảnh không mong muốn ở điều kiện thiếu ánh sáng mà phải được che kín và chiếu thêm đèn led để tăng cường sáng.

3.2. Tính toán và thiết kế hệ thống

3.2.1. Thiết kế sơ đồ khối hệ thống



Hình 3.1. Sơ đồ khối hệ thống.

Chức năng của từng khối:

Khối xử lý

- Raspberry Pi: có chức năng nhận tín hiệu hình ảnh từ khối thu tín hiệu hình ảnh khi có mệnh lệnh của Arduino Uno, tiến hành xử lý hình ảnh và

phân tích nhận dạng, sau đó trả lại kết quả cho Arduino Uno, tiếp tục chờ mệnh lệnh tiếp theo của Arduino Uno.

- Arduino Uno: Có chức năng nhận tín hiệu từ khối cảm biến sau đó ra lệnh cho Raspberry Pi hoạt động, rồi nhận dữ liệu trở về từ Raspberry Pi, tiến hành phân tích xử lý, điều khiển hoạt động của khối động cơ và gửi dữ liệu sang khối hiển thị.

Khối camera:

- Có chức năng thu thập tín hiệu hình ảnh từ thực tế chuyển về tín hiệu điện và gửi dữ liệu cho khối Raspberry Pi. Ở đây sử dụng Camera Raspberry

Khối cảm biến:

- Có chức năng gửi tín hiệu điện cho khối điều khiển khi có vật cản đi qua. Sử dụng cảm biến hồng ngoại E18-D80NK.

Khối động cơ:

- Được vận hành bởi khối điều khiển. Sử dụng động cơ DC và Xanh

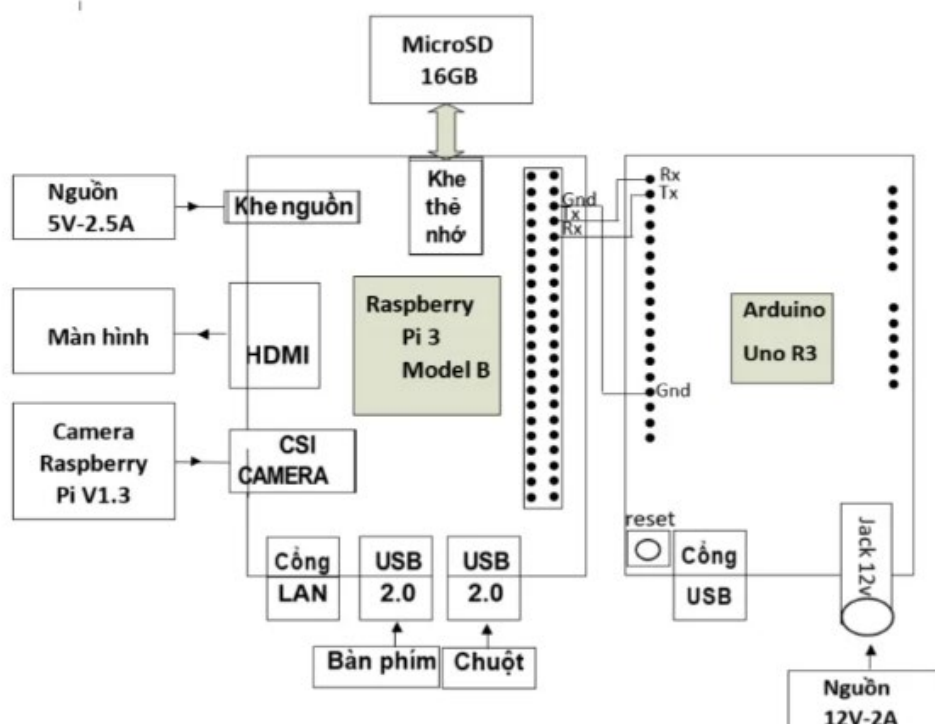
Khối hiển thị:

- Có chức năng hiển thị kết quả nhận được từ khối điều khiển. Sử dụng màn hình LCD16x2.

3.2.2. Sơ đồ kết nối hệ thống

a) Khối xử lý

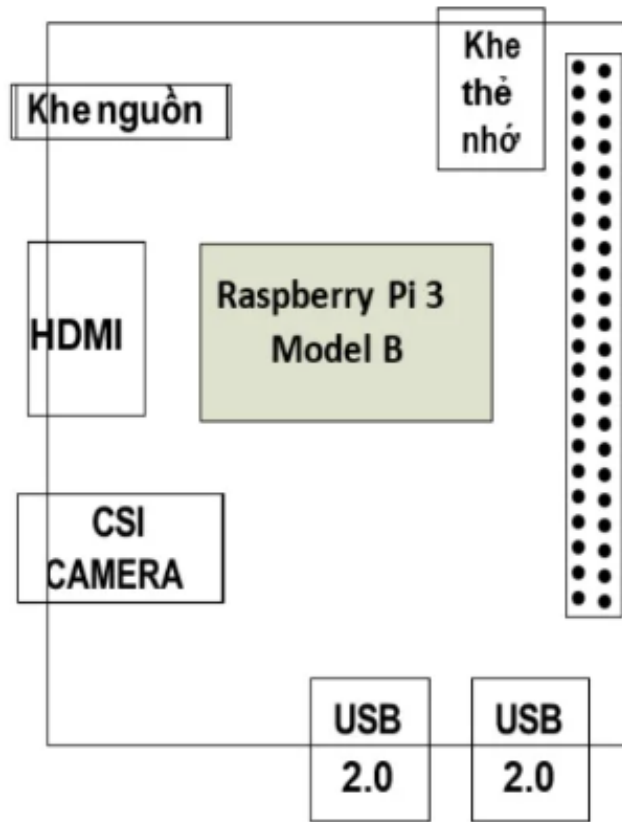
Raspberry Pi



Hình 3.2. Sơ đồ kết nối của Raspberry Pi.

Trên đây là sơ đồ kết nối của Raspberry Pi, khối thu tín hiệu hình ảnh từ Camera Raspberry Pi được kết nối vào cổng CIS Camera, Raspberry Pi và Arduino truyền nhận dữ liệu từ 2 chân Rx, Tx.

Chức năng của Kit Raspberry Pi là nhận tín hiệu hình ảnh từ khối Camera, sau đó xử lý ảnh và phân tích để đưa kết quả sang Arduino Uno.



Hình 3.3. Sơ đồ các cổng ngoại vi sử dụng

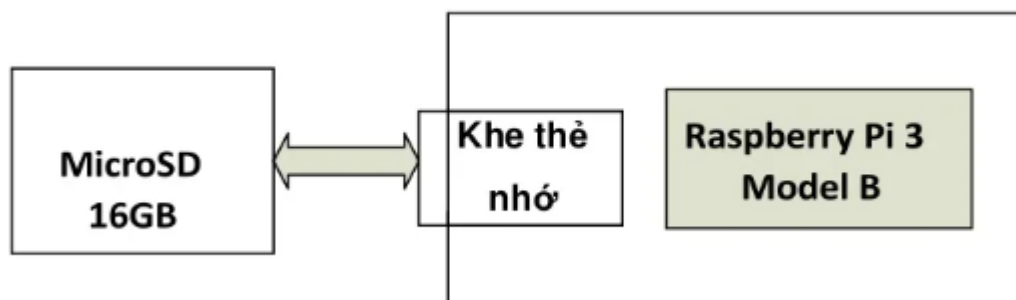
Cổng USB: Kit Raspberry có 4 cổng USB (Universal Serial Bus) nhưng chỉ sử dụng 2 cổng để kết nối với bàn phím và chuột. Với chuẩn USB 2.0 tốc độ cao, đường truyền đạt tốc độ tối đa đến 480 Mbps. Cáp USB gồm hai sợi nguồn (+5V và dây GND) cùng một cặp gồm hai sợi dây xoắn để mang dữ liệu.

Cổng HDMI: dùng một cổng HDMI (High-Definition Multimedia Interface) để kết nối với màn hình LCD để hỗ trợ lập trình, truyền dữ liệu theo chuẩn Giao Tiếp hình ảnh KTS, hỗ trợ âm thanh KTS 8 kênh.

Cổng CSI Camera: Mặt của mô hình này chính là Camera được kết nối thông qua cổng CSI Camera có 15 chân.

Nguồn: Dòng hoạt động của Raspberry 500-1000mA, cổng HDMI sử dụng 50mA, module Camera sử dụng 250mA, bàn phím và chuột sử dụng từ 100mA-1000mA. Tổng dòng tiêu thụ được khoảng 2300mA. Chính vì vậy ta sử dụng Adapter 5V và 2.5A để cấp cho máy tính nhúng cho Raspberry.

Chọn thẻ nhớ lưu dữ liệu: Muốn chạy được chương trình trên máy tính nhưng ta cần phải có hệ điều hành được cài sẵn trên thẻ nhớ. Vì hệ điều hành Raspbian chiếm 4GB dung lượng chưa kể các dữ liệu, chương trình, phần mềm liên quan, chính vì vậy chọn loại thẻ nhớ tối thiểu 8GB, ở đây nhóm chọn loại thẻ nhớ MicroSD 16GB có tốc độ đọc lên tới 48MB/s, tốc độ đọc cũng ảnh hưởng tới tốc độ xử lý dữ liệu của các chương trình chính vì vậy với tốc độ 48MB/s là tốc độ hợp lý.



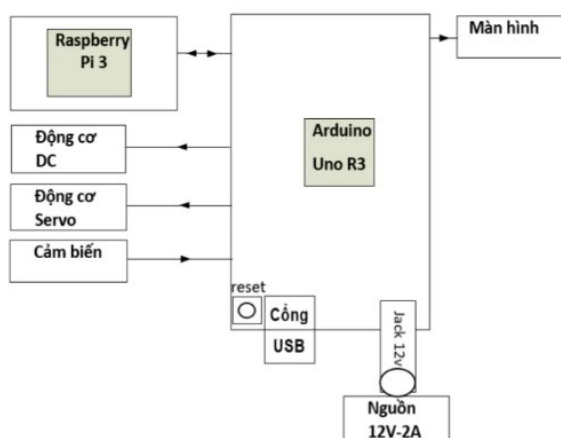
Hình 3.4. Sơ đồ kết nối của thẻ nhớ Raspberry.

Thiết bị	Dòng tiêu thụ	Ghi Chú
Raspberry Pi	500-1000 mA	Khi kit hoạt động mình nó
Module Camera	250 mA	
Cổng HDMI	50 mA	
Bàn Phím và chuột	100 Ma – 1000 mA	
Tổng	900 mA – 2300 mA	Chọn nguồn 2.5A

Bảng 3-1: Số liệu của các thiết bị

Từ bảng thống kê trên cho ta thấy, giao tiếp ngoại vi của Raspberry Pi là rất tốn kém năng lượng, chính vì thế để hệ thống hoạt động ổn định thì nên đầu tư mua | một bộ nguồn cung cấp cho Raspberry Pi.

Arduino Uno

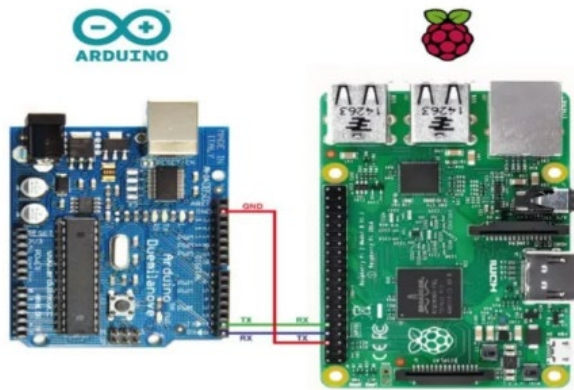


Hình 3.5. Sơ đồ kết nối của Arduino với Raspberry.

Giao tiếp giữa Raspberry Pi và Arduino Uno:

Trong suốt quá trình hoạt động của hệ thống, Raspberry Pi và Arduino Uno cần truyền nhận dữ liệu một cách liên tục, ở đây nhóm chọn kiểu truyền Uart vì trên cả 2 Kit điều khiển đều được tích hợp sẵn chuẩn truyền này nên quá trình thiết lập khá dễ dàng, đồng thời chuẩn Uart với tốc độ truyền nhận nhanh và ổn định, đáp ứng tốt được yêu cầu của hệ thống.

- Chân RX của Arduino Uno nối với chân TX của Raspberry Pi 3.
- Chân TX của Arduino Uno nối với chân RX của Raspberry Pi 3.
- Nối chung 2 chân GND.

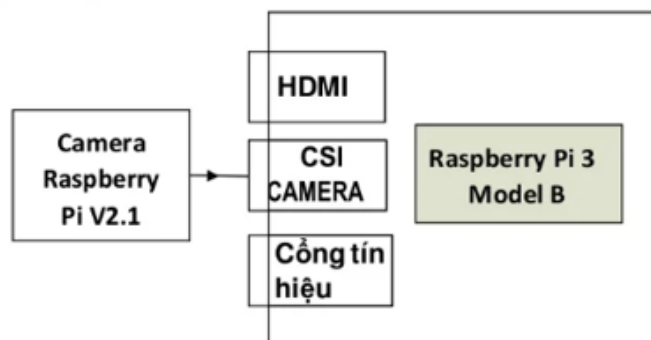


Hình 3.6. Giao tiếp Uart giữa Raspberry Pi và Arduino Uno.

b) Khối camera

Được xem như là bộ phận chính của mô hình. Có chức năng thu thập tín hiệu hình ảnh từ thực tế rồi gửi dữ liệu cho Raspberry Pi xử lý.

Ở đây để có chất lượng hình ảnh tốt để xử lý ảnh nên nó được bao bọc lại để tránh những ánh sáng từ bên ngoài. Ta dùng Camera có độ phân giải từ 8 Megapixels, nên chọn Camera Raspberry Pi V2.1 làm khối thu tín hiệu hình ảnh. Vì khối có ảnh hưởng lớn đến chất lượng ảnh cùng với việc nhận dạng và phân loại sản phẩm, chính vì vậy nhóm chọn Camera Pi V2.1 có độ phân giải tốt 8 Megapixels. Megapixel (MP) là đơn vị đo độ phân giải của thiết bị quang.

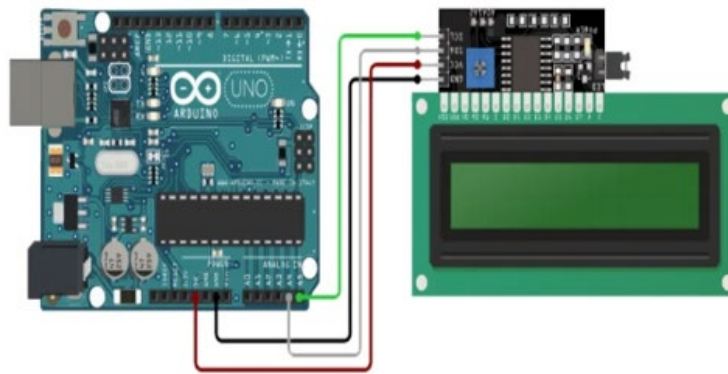


Hình 3.7. Sơ đồ kết nối Camera.

c) Khối hiển thị

Ở đây nhóm sử dụng màn hình LCD16x2 để hiển thị kết quả, LCD16x2 có các chuẩn giao tiếp với Arduino Uno: Giao tiếp chuẩn 4bit, chuẩn 8bit, chuẩn I2C. Nhóm quyết định chọn chuẩn I2C vì với chuẩn này việc lắp đặt khá dễ dàng, chỉ cần sử dụng 2 chân SDA, SCL.

- Chân SDA của I2C trên LCD được nối vào chân A4 trên Arduino Uno (chân SDA trên Arduino Uno)
- Chân SCL của IDC trên LCD được nối vào chân A5 trên Arduino Uno (chân SCL trên Arduino Uno)



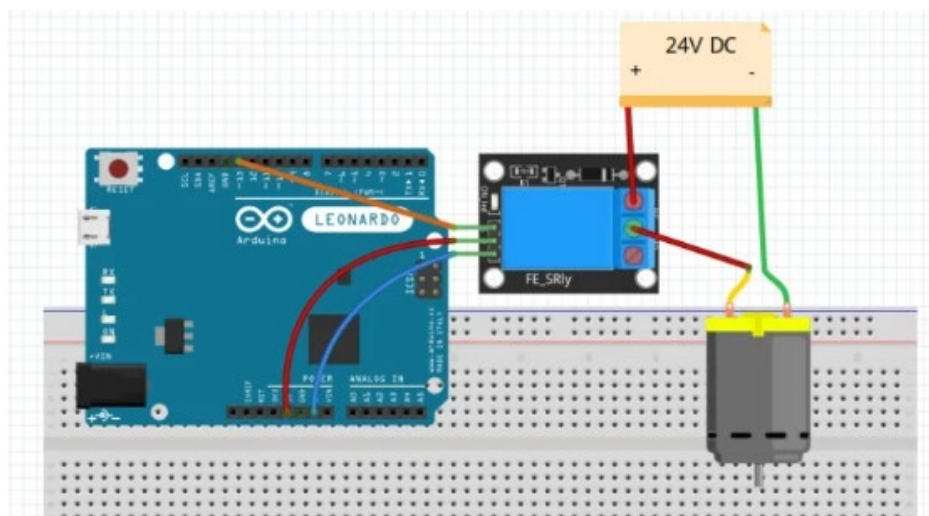
Hình 3.8. Giao tiếp I2C giữa Arduino Uno và LCD16x2.

d) Khối động cơ

Giao tiếp động cơ DC:

- Động cơ DC được sử dụng để điều khiển cho băng tải chạy, sử dụng động cơ DC 24V vì thế nhóm sử dụng Arduino Uno điều khiển Relay 5V để đóng ngắt quá trình chạy.

Chân kích của Relay 5V được nối vào chân 0 của Arduino:



Hình 3.9. Kết nối Arduino Uno và động cơ DC.

- Động cơ XiLanh được sử dụng để đẩy sản phẩm nhằm phân loại theo cùng màu sắc.

6 chân điều khiển Xanh (Van khí nén) được nối tương ứng vào 6 chân 2,3,4,5,6,7 trên Arduino.

e) Khối cảm biến

Đề tài sử dụng cảm biến vật cản hồng ngoại E18- D80NK để phát hiện sản phẩm, gửi tín hiệu về Arduino Uno để thực hiện các quá trình tiếp theo.

Chân tín hiệu của cảm biến được kết nối vào chân số 10,11,12, A0, A1, A2 của Arduino Uno.



Hình 3.10. Kết nối cảm biến hồng ngoại và Arduino Uno.

f) Khối nguồn

Nguồn cấp cho Raspberry Pi 3 và Camera Pi

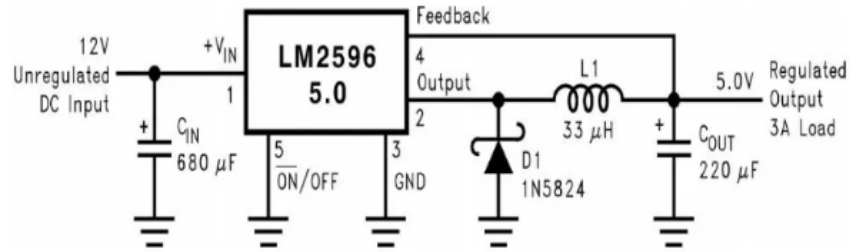


Hình 3.11. Adapter 5VDC – 2A.

- Raspberry Pi 3 hoạt động ổn định với nguồn 5VDC-2A, vì vậy ở đây nhóm sử dụng Adapter chuyên dụng cho Raspberry

Nguồn cung cấp cho Arduino Uno

- Nguồn cung cấp cho Arduino Uno R3 ở mức điện áp 7-12VDC, dòng điện max $I_{max}=50mA$, do đó nhóm sử dụng mạch hạ áp LM2596 để hạ điện áp 24VDC từ nguồn tổ ong còn 12VDC cấp cho Arduino Uno.



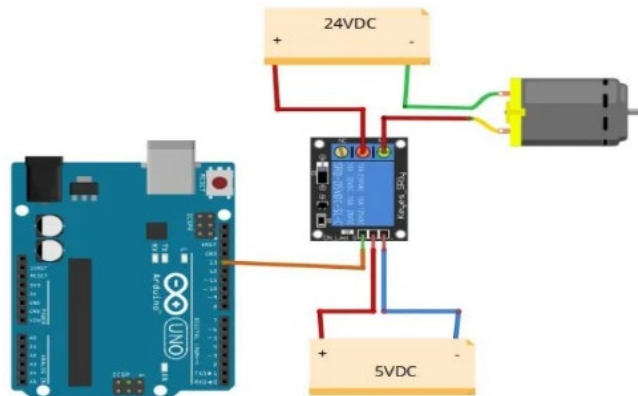
Hình 3.12. Sơ đồ nguyên lý mạch hạ áp LM2596.



Hình 3.13. LM2596

Nguồn cấp cho động cơ DC

- Ở đây nhóm sử dụng nguồn tổ ong cấp cho động cơ DC 24VDC để điều khiển hoạt động của băng tải, do điện áp hoạt động quá lớn so với nguồn hoạt động của Arduino Uno, nên sử dụng thêm module Relay 5VDC để Arduino Uno có thể đóng ngắt quá trình hoạt động của băng tải.



Hình 3.14. Mạch cấp nguồn cho động cơ DC 24V.

Nguồn cấp cho cảm biến E18-D80NK, LCD16x2, Relay 5V

Các thiết bị này có điểm chung là điều hoạt động tốt với điện áp 5VDC, chỉ khác dòng điện hoạt động.

- Cảm biến E18 – D80NK hoạt động với dòng điện $I_{max} = 300\text{mA}$.
- LCD 16x2 hoạt động với dòng $max = 2.5\text{mA}$.

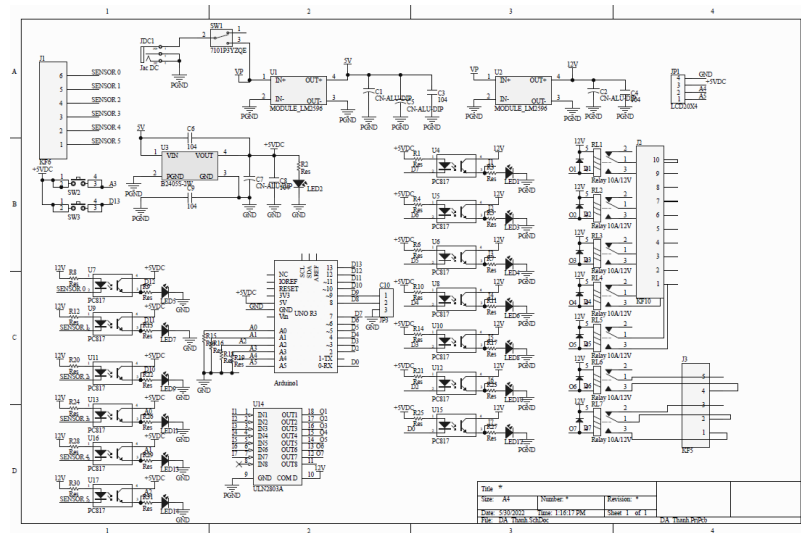
- Relay 5V hoạt động với dòng điện cho phép qua tối đa $I_{max} = 10A$, dòng kích
- $I_{kích} = 5\text{ mA}$.

Dòng điện tổng khoảng 1203 mA, vì vậy nhóm quyết định sử dụng mạch hạ áp LM2596 để chuyển điện áp 24VDC - 5A từ nguồn tổ ong còn 5VDC - 5A để cấp cho các thiết bị này.



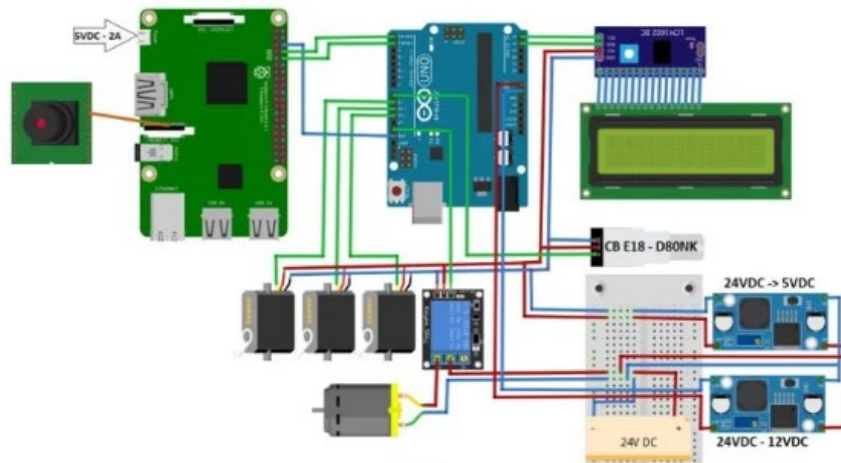
Hình 3.15. Nguồn tổ ong 24VDC – 5A.

g) Sơ đồ nguyên lý



Hình 3.16. Sơ đồ nguyên lý mạch.

3.2.3. Sơ đồ kết nối toàn mạch



Hình 3.17. Sơ đồ kết nối toàn mạch.

Chương 4: Thi công hệ thống

4.1. Giới thiệu

Đề tài xử lý ảnh hiện nay đã và đang đi vào trong giảng đường đại học từ sơ cấp đến nâng cao. Vì vậy, đề tài “Ứng dụng xử lý ảnh trong hệ thống phân loại sản phẩm” được ứng dụng thông qua việc xử lý hình ảnh, nhận dạng hình ảnh và phân loại sản phẩm theo màu sắc. Ở phần này sẽ trình bày quá trình thi công hệ thống.

Với đề tài “Ứng dụng xử lý ảnh trong hệ thống phân loại sản phẩm” bao gồm: Mô hình phân loại sản phẩm sử dụng sản phẩm là các phôi theo màu sắc (đỏ, xanh, vàng). Có thể phát hiện những sản phẩm không thuộc 3 màu trên.

Đề tài chỉ xây dựng với mô hình nhỏ và gọn không làm thành dây chuyền sản xuất. Mô hình được kết nối qua các dây tín hiệu nên sẽ không có phần thiết kế và thi công bo mạch.

4.2. Thi công hệ thống

4.2.1. Chuẩn bị phần cứng

Sau khi tính toán và thiết kế, chúng ta chuẩn bị linh kiện tiến hành lắp ráp.

STT	Tên Linh Kiện	Giá Trị	Dạng vỏ	Chú thích
1	Băng tải		Nhôm và nhựa	Bao gồm đế ròng rọc, dây băng tải
2	LCD	Điện áp hoạt động: 5V Dòng : 2.5 mA	LCD 16*2	
3	Kit Arduino	Điện áp hoạt động: 5V Dòng : 50 mA	Arduino Uno R3	
4	Kit Raspberry	Điện áp hoạt động: 5V Dòng: 1A	Kit Raspberry	Có quạt tản nhiệt
5	Camera Pi	Điện áp hoạt động: 5V Dòng: 250 mA	Camera Pi	6
6	Xilanh(Van Khí Nén)	Điện áp hoạt động: 24V		
7	Cảm biến	Điện áp hoạt động: 5V Dòng: 300mA	Nhựa	

8	Mạch hạ áp	Đầu vào: 3V-40V LM2596 Đầu ra: 1V-30V Dòng: 3A	LM2596	
9	Relay	Điện áp hoạt động: 5V Dòng cho phép qua tối đa: 10A Dòng kích: 5mA	Nhựa	
10	Domino		Nhựa cứng	
11	Led đơn	Điện áp hoạt động: 3V Dòng: 15mA	Led	
12	Bo mạch		Nhựa cứng	
13	Nguồn tổ ong	Điện áp vào: 220 VAC Điện áp ra: 24VDC Dòng ra: 10A	Thép	
14	Adapter	Điện áp vào: 220 VAC Điện áp ra: 5VDC Dòng ra: 2A	Nhựa cứng	
15	Động cơ DC	Điện áp hoạt động: 24V Dòng: 900mA	Hợp kim	

Bảng 4-1: Thông số chi tiết linh kiện.

4.2.2. Lắp ráp và kiểm tra

Là phần kết nối giữa các linh kiện và các module, tạo thành một khối thống nhất với nhau.

a) Phần băng tải

Lắp băng tải và động cơ thông qua chi tiết in 3D.

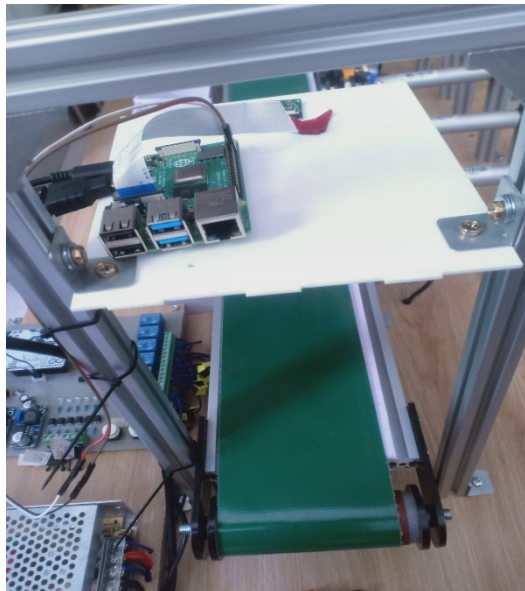


Hình 4.1. Băng tải và động cơ DC.

b) Phần máng trượt

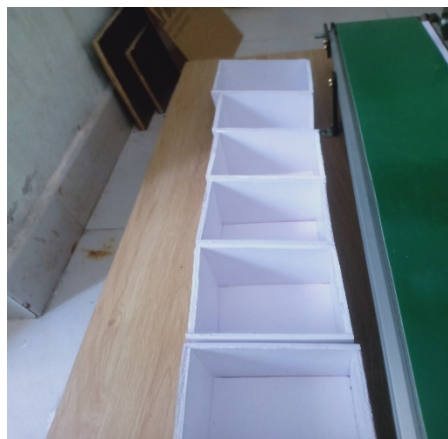
Máng trượt chia làm 2 phần:

- Phần thứ nhất là máng đẩy sản phẩm vào



Hình 4.2. Máng khi đưa sản phẩm vào.

- Phần thứ 2 là máng sản phẩm sau khi được phân loại màu



Hình 4.3. Máng sau khi phân loại màu.

Chúng được sắp xếp và gắn kết với băng tải, tiến hành vận ốc và cố định mảng trượt.

c) Phần hiển thị

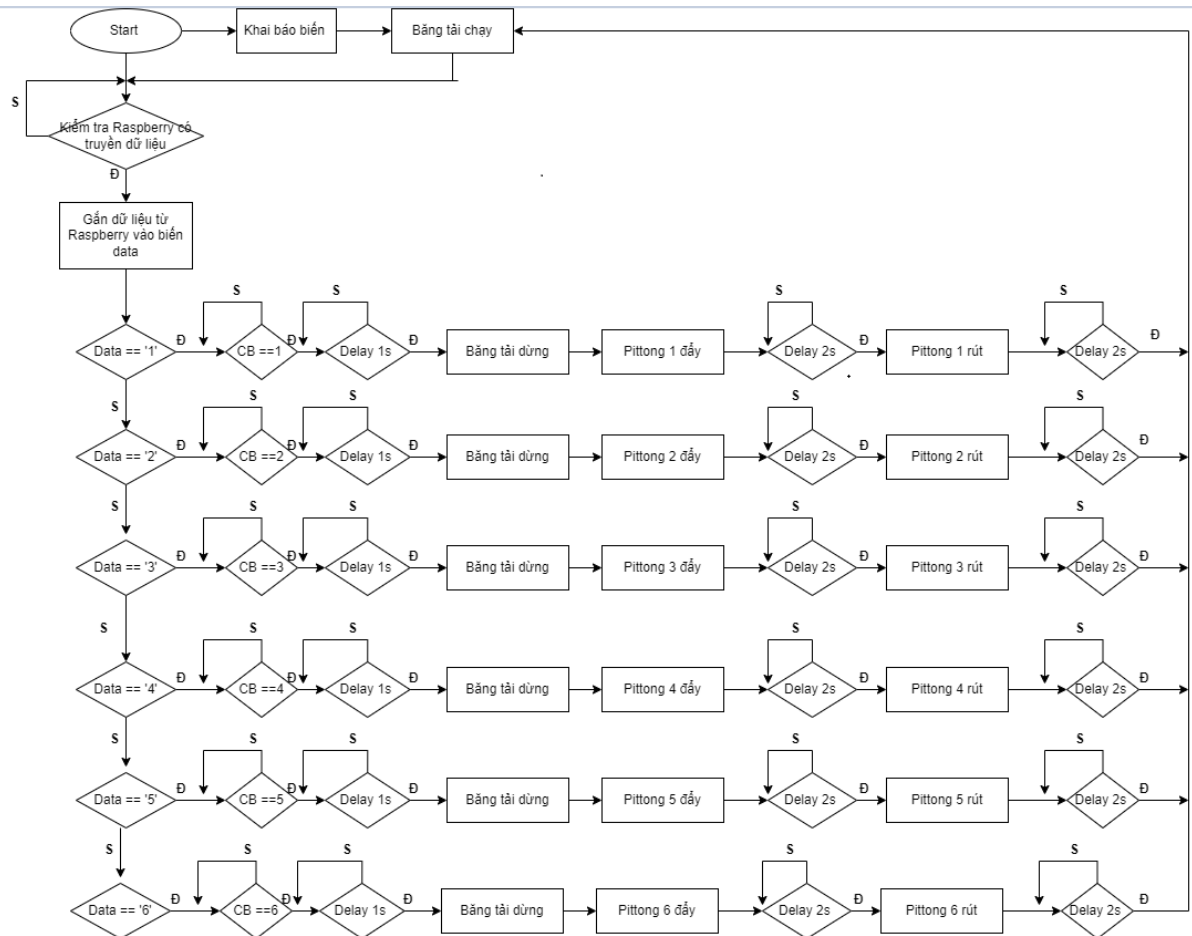
Kết nối module LCD16X2 I2C vào khối điều khiển và được đặt bên ngoài hộp điều khiển.

d) Phần Module điều khiển

- Lắp Raspberry Pi với Camera và Arduino
 - Kết nối thông qua giao tiếp Uart giữa Raspberry và Arduino Uno bao gồm chân Tx của Raspberry nối với Rx của Arduino Uno và Rx của Raspberry nối với Tx của Arduino Uno.
 - Kết nối Raspberry với Camera thông qua socket CSI Camera có sẵn trên kit Raspberry bằng cáp 15pin ribbon.
- Lắp ráp cảm biến, Van khí nén với Arduino
 - Kiểm tra chính xác màu sắc và chức năng từng loại dây để việc kết nối trở nên thuận tiện.
 - Cảm biến bao gồm có 3 dây: Đen là dây tín hiệu được kết nối vào pin 8 của Arduino và dây nâu được nối vào nguồn 5v, dây xanh được nối GND.
 - Van Khí nén bao gồm 3 dây: Cam là dây tín hiệu được nối vào pin 2,3,4,5,6,7 của Arduino và dây đỏ được nối vào nguồn 5v, dây nâu được nối GND.
- Lắp ráp Relay 5V với động cơ giảm tốc DC và Arduino
 - Khi kết nối Relay 5v với động cơ giảm tốc Dc bao gồm hai loại kết nối: thường đóng và thường mở.
 - Chọn kết nối phù hợp và tiện lợi cho việc lập trình trên kit Arduino. Tiếp điểm thường đóng khi lập trình trạng thái mức 1 là động cơ hoạt động và mức 0 là động cơ ngưng và ngược lại đối với tiếp điểm thường hở.
- Lắp ráp LM2596 với cảm biến, LCD, Relay
 - Hiệu chỉnh điện áp ngõ ra của Lm2596 từ 24V xuống 5V cấp cho Cảm biến, Lcd, Relay thông qua việc vặn biến trở.
 - Kết nối các dây nguồn và GND của các module với module Lm2596.
 - Nguồn Adapter 5V kết nối với Raspberry Pi 3
 - Kiểm tra nguồn có kết nối được với Raspberry.
 - Kiểm tra Raspberry có lên nguồn chưa. Nếu chưa thì kiểm tra lại kết nối.

4.3. Lập trình hệ thống

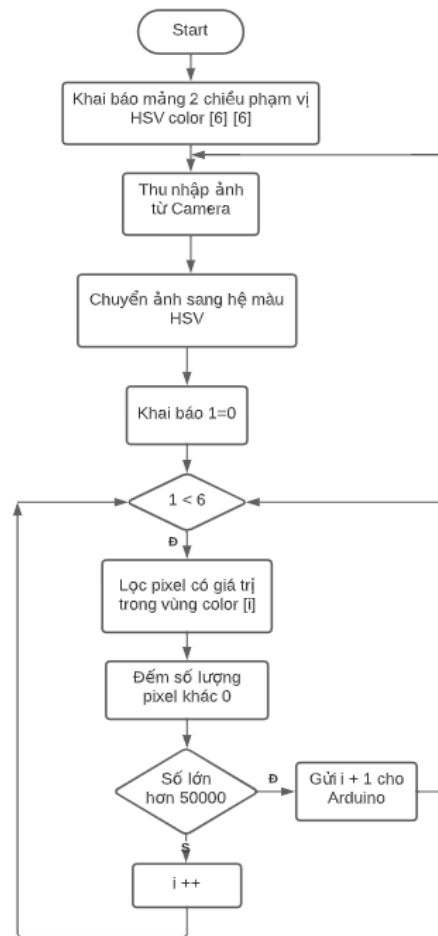
4.3.1. Lưu đồ giải thuật trên Arduino



Hình 4.4. Lưu đồ thuật toán trên Arduino Uno.

4.3.2. Lưu đồ thuật toán xử lý ảnh trên Raspberry Pi 3

Là chương trình con “Phân loại ảnh” của lưu đồ ở trên.



Hình 4.5. Lưu đồ thuật xử lý trên Raspberry Pi.

4.3.3. Hướng dẫn sử dụng và thao tác

Có 2 cách để cho Raspberry hoạt động: Hoạt động thủ công và hoạt động tự động.

Hoạt động thủ công:

- **Bước 1:** Sau khi cấp nguồn cho Raspberry, ta kết nối giữa máy tính và Raspberry thông qua dây LAN, tiến hành truy cập vào hệ điều hành của Raspberry như đã trình bày ở phần 4.3.3, giao diện hệ điều hành Raspbian xuất hiện.
- **Bước 2:** Truy cập vào cửa sổ Terminal để thực thi chương trình đã viết
- **Bước 3:** Tại cửa sổ Terminal, ta tiến hành gọi tên chương trình đã viết nhưng ở đây do sử dụng thư viện OpenCV nên ta cần phải gọi thư viện ra trước.

Lệnh gọi OpenCV:

- Source -/profile
- WorkonCV

- Sau khi đã gọi thư viện OpenCV, tiếp tục gọi chương trình cần thực thi, ở đây nhóm đặt tên chương trình là: “phongfinal.py”.
- Python phongfinal.py

Hoạt động tự động:

- Hoạt động tự động nghĩa là sau khi cấp nguồn vào Raspberry, Raspberry sẽ tự động chỉ đến đường dẫn và thực thi chương trình mà không cần phải trải qua các bước như ở hoạt động thủ công. Để thực hiện được điều đó:
- **Bước 1:** Sau khi cấp nguồn cho Raspberry, ta kết nối giữa máy tính và Raspberry thông qua dây LAN, tiến hành truy cập đến Terminal của Raspberry bằng phần mềm PuTTY.

Ở đây ta nhập địa chỉ IP của Raspberry là: 192.168.137.87

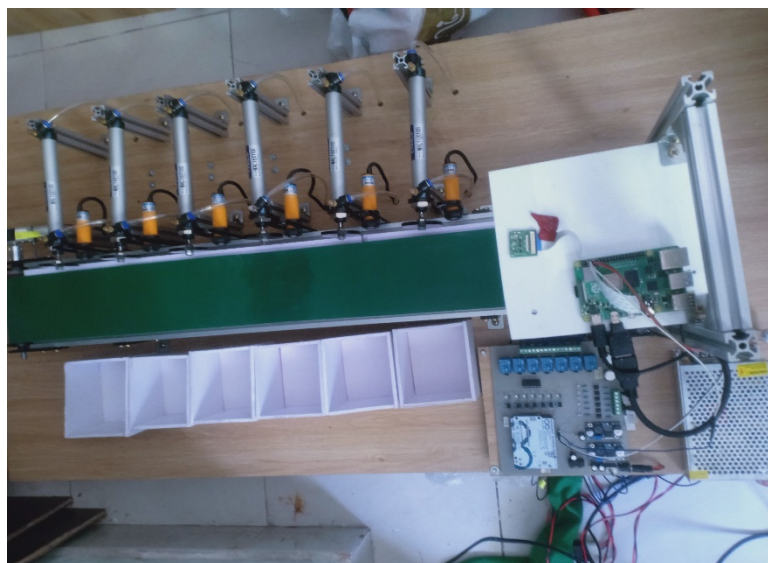
- **Bước 2:** Truy cập vào cửa sổ terminal và đăng nhập vào Raspbian để thực thi chương trình đã viết.

Tài khoản: pi

Mật khẩu: raspbian

- **Bước 3:** Tạo một file để lưu các lệnh thực thi.
 - Ta cần truy cập vào hệ điều hành Raspberry để tạo một file với đuôi .sh trong đường dẫn Home/pi lưu các lệnh cần thực thi.
- **Bước 4:** Mở tất cả các file đã có trong Raspberry và tiến hành mở file chạy tự động
 - Tiến hành gọi lệnh chạy tự động tại màn hình Terminal.
 - Nano start_final.hs.
 - Export VISUAL=nano; crontab-e.

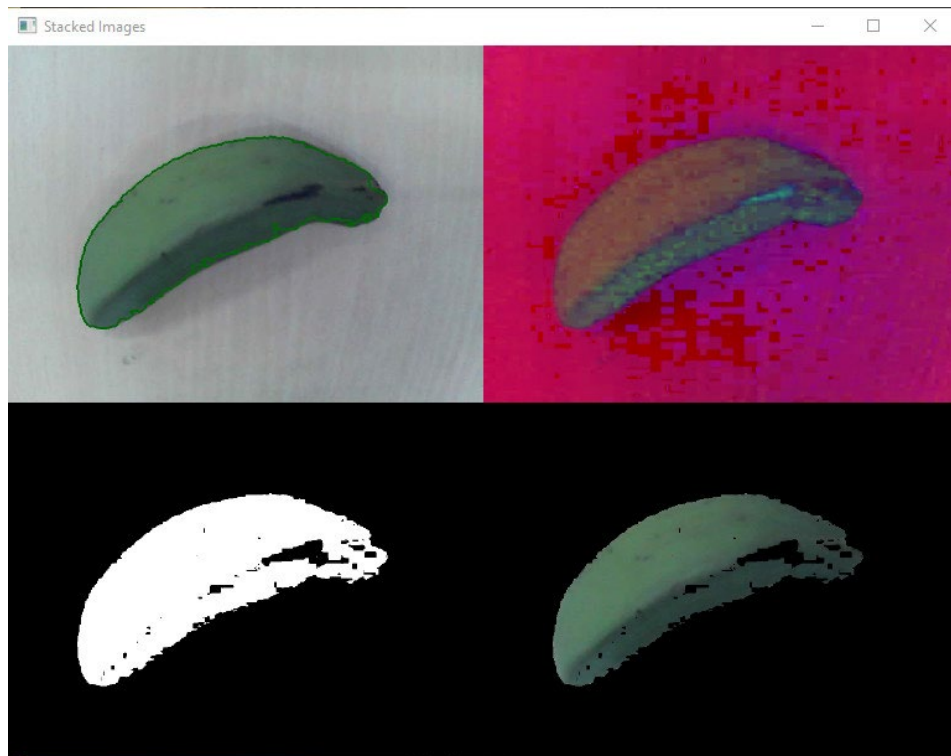
4.4. Mô hình hoàn thiện



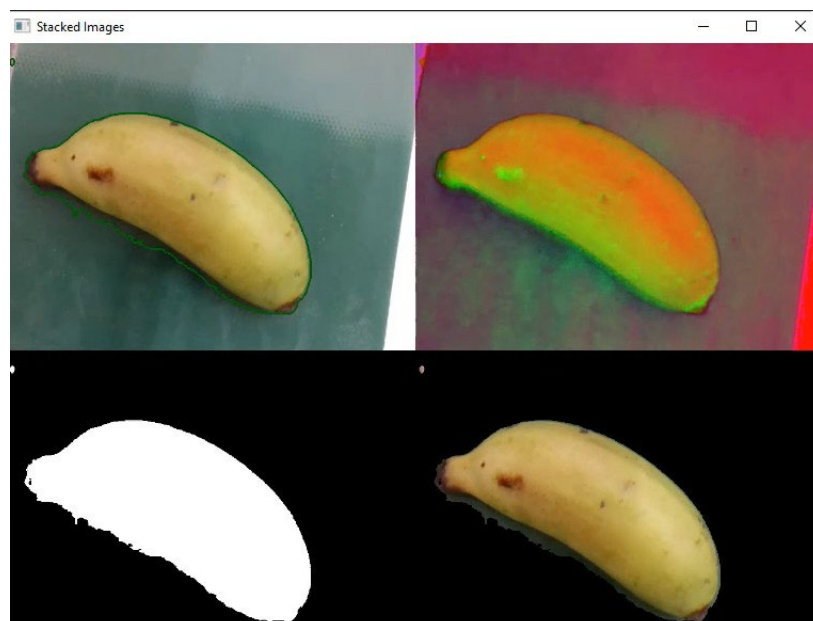
Hình 4.6. Mô hình thực tế

Mô hình hoạt động khá ổn định. Bộ xử lý trung tâm xử lý tín hiệu khá tốt. Phân loại thành công các sản phẩm đỏ, xanh, vàng, nếu không thuộc một trong các màu sắc kể trên thì được xem là sản phẩm khác, hệ thống máng cấp sản phẩm hoạt động ổn định.

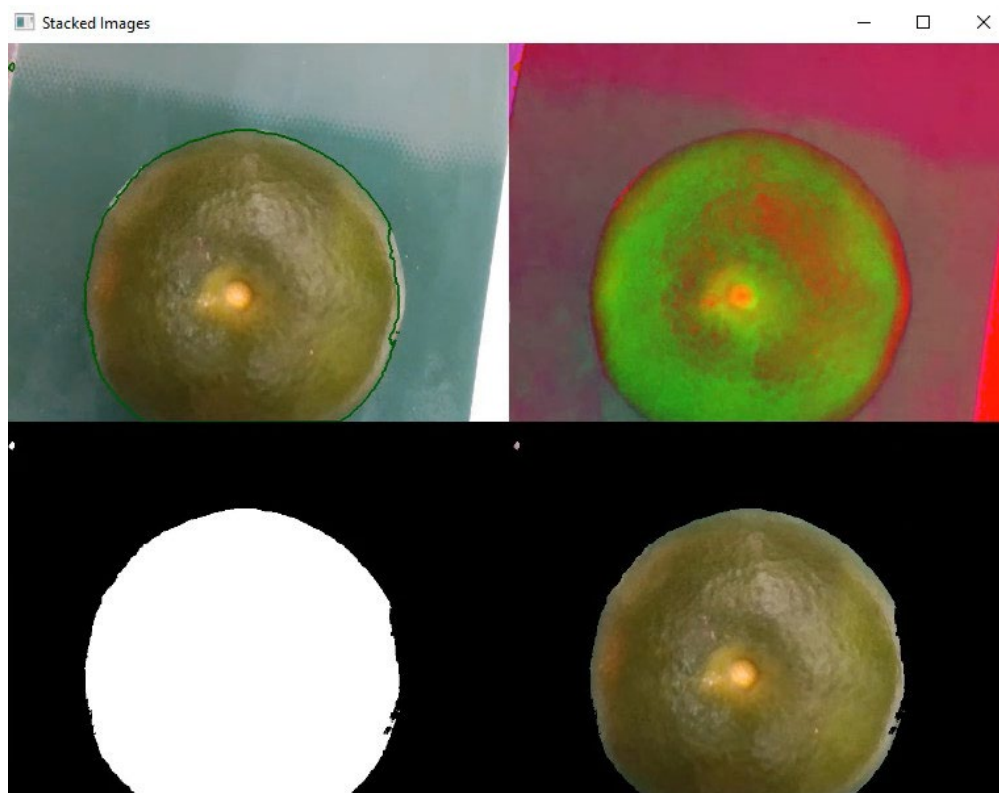
4.5. Kết quả tách biên nhận diện từ các quả



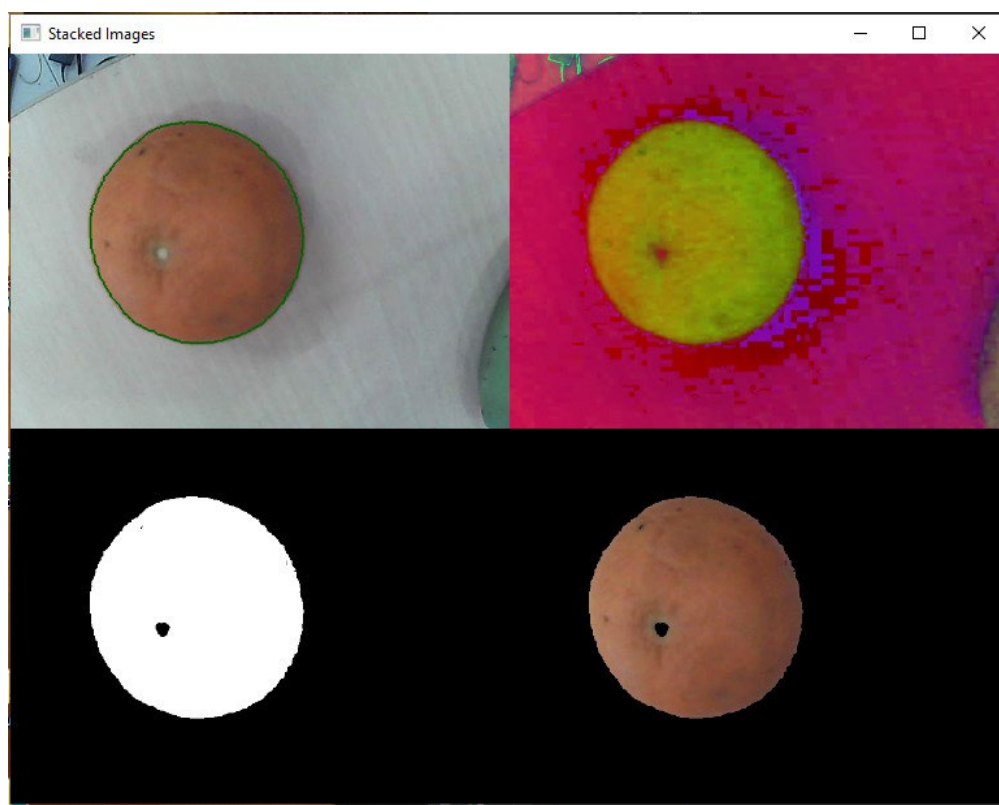
Hình 4.7. Kết quả nhận dạng của quả chuối xanh



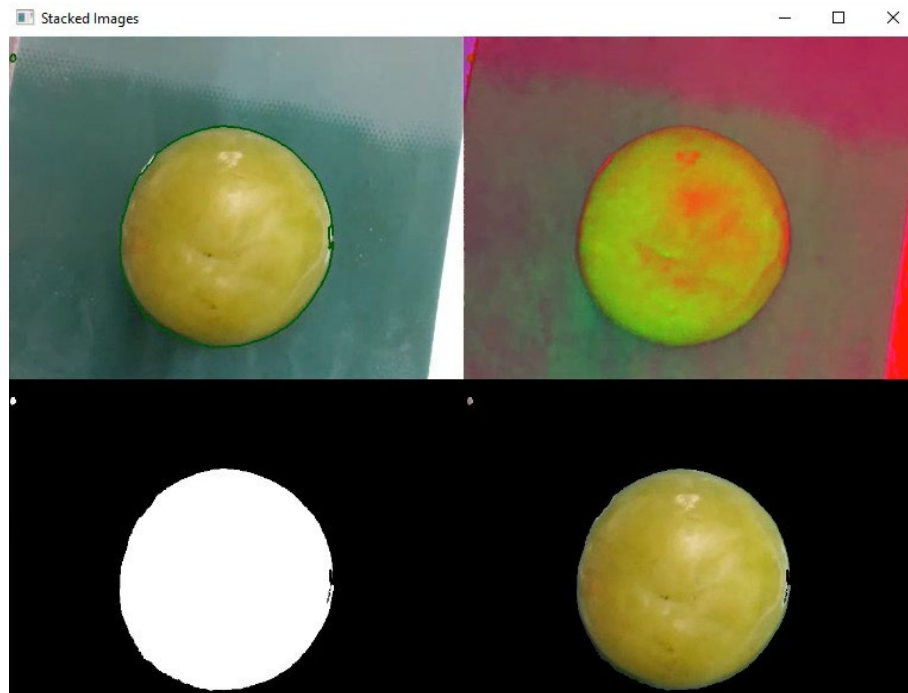
Hình 4.8. Kết quả nhận dạng của quả chuối vàng



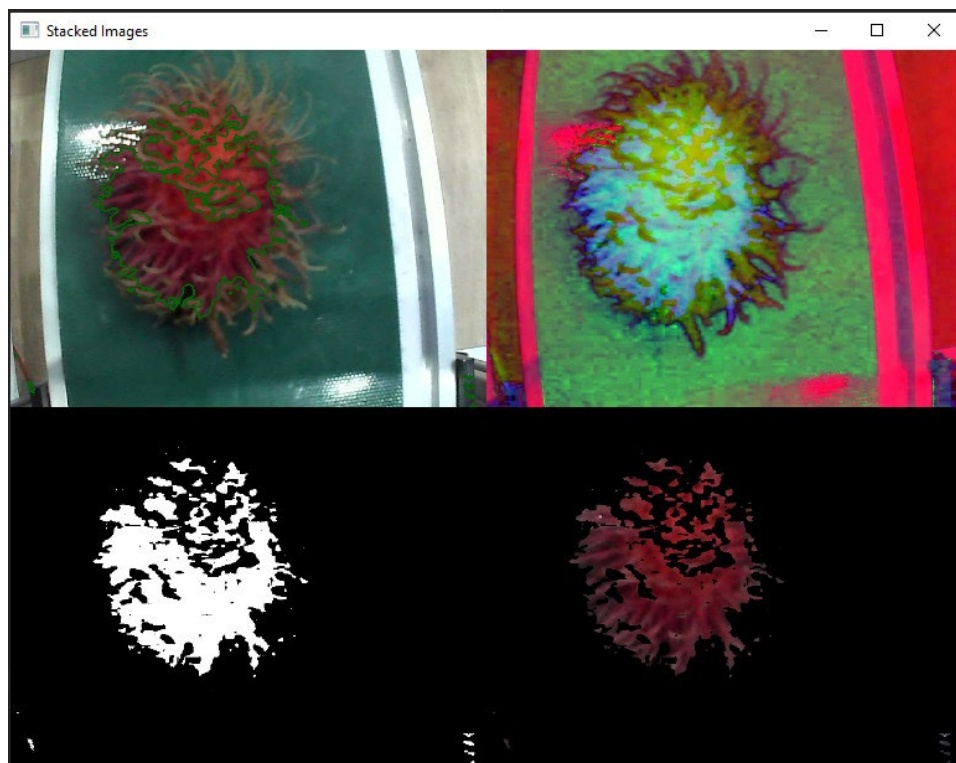
Hình 4.9. Kết quả nhận dạng của quả cam xanh



Hình 4.10. Kết quả nhận dạng của quả cam vàng



Hình 4.11. Kết quả nhận dạng của quả mận xanh



Hình 4.12. Kết quả nhận dạng của quả mận đỏ

Sau khi nhận diện các màu, Raspberry Pi sẽ tiến hành xuất ra kết quả và gửi chuỗi dữ liệu cho Arduino.

Nhận xét:

Sau các thực nghiệm trên các sản phẩm, hệ thống phân loại sản phẩm cho thấy hoạt động tương đối ổn định với độ chính xác tuyệt đối, không bị nhầm lẫn trong việc phân loại giữa các màu với nhau và phân loại rõ ràng từng sản phẩm.

4.6. Nhận xét và đánh giá

Qua quá trình thực hiện đề tài, nhóm đã biết sử dụng Raspberry các chức năng cơ bản chiếc máy tính nhúng, như việc tạo file mới, tạo project mới, tìm hiểu Python trên Raspberry, cách chạy chương trình Python,... Ngoài ra nhóm còn tự cài đặt hệ điều hành cho máy và kết nối với Camera Pi chụp ảnh từ Raspberry,... Đánh giá máy tính nhúng Raspberry một cách khách quan, máy tính được thiết kế nhỏ gọn, thuận tiện cho các mô hình không thích sự cồng kềnh, đáp ứng đủ các chức năng cơ bản như một chiếc máy tính bình thường, tốc độ xử lý tạm ổn cho các dự án nhỏ, hệ thống chạy ổn định, nhưng các dự án lớn cần cấu hình máy cao thì Raspberry chưa đáp ứng được nhu cầu. Vì thời gian thực hiện đề tài giới hạn, nên nhóm không đi sâu về các giao tiếp ngoại vi của Raspberry, như Wifi hay các chân GPIO... Chỉ dừng lại ở việc sử dụng các chức năng cơ bản của máy. Đối với Model Camera Raspberry Pi có độ phân giải không quá cao nhưng ngược lại nó có giá thành rẻ, cùng với chất lượng ảnh phù hợp với việc xử lý của đề tài. Tìm hiểu chi tiết về Arduino Uno cũng như cách giao tiếp giữa Arduino với máy tính nhúng Raspberry Pi, điều khiển được hoạt động của Xilanh, Relay, Cảm biến và LCD.

Sau khi qua thời gian tìm hiểu phương pháp nhận dạng, phân loại sản phẩm, nhóm đã phân loại thành công được ba màu cơ bản (đỏ, xanh, vàng), nếu không thuộc một trong ba màu kể trên thì được xem là sản phẩm khác, cùng với đó là hiển thị sản phẩm trên LCD.

KẾT LUẬN

- **Kết quả đạt được**

Sau khi tổng hợp các kết quả đạt được và đem so sánh với những yêu cầu và mục tiêu thiết kế cho thấy hệ thống đáp ứng tương đối đầy đủ, chính xác. Mô hình phần cứng hoạt động tốt, thiết bị nhỏ gọn, lắp đặt dễ dàng, có tính kinh tế, thuận tiện cho việc học tập và cũng có thể phát triển thành dây chuyền sản xuất. Nhận dạng tốt được hầu hết các màu sắc đề ra của sản phẩm, có đếm sản phẩm.

- **Những mặt hạn chế**

Ngoài những kết quả đạt được thì hệ thống vẫn còn những hạn chế sau: Tốc độ xử lý còn chậm, hệ thống bùồng ảnh chưa đáp ứng được ánh sáng tốt nhất, chỉ phân loại được sản phẩm theo màu sắc, không phát hiện được sản phẩm bị lỗi.

- **Hướng phát triển**

Từ những mặt hạn chế của đề tài, để đề tài hoạt động tốt hơn và có thể áp dụng vào thực tế sau này nhóm đã đề ra những hướng phát triển như sau: Làm thành dây chuyền sản xuất với cấu hình mạnh hơn, cải tiến thêm chức năng phát hiện được sản phẩm lỗi, tìm hiểu và phát triển thêm chức năng vận hành và giám sát từ xa.

TÀI LIỆU THAM KHẢO

- [1] Bích Hồng (TTXVN/Vietnam - 03/04/2022), *Xuất khẩu rau quả chuyển dịch tích cực trong những tháng đầu năm*, [Xuất khẩu rau quả chuyển dịch tích cực trong những tháng đầu năm | Kinh doanh | Vietnam+ \(VietnamPlus\)](#)
- [2] Gabriel Oliver-Codina, Nuno Gracias và Antonio M. López (2016), *Image Based Mango Fruit Detection, Localisation and Yield Estimation Using Multiple View Geometry*, <https://www.mdpi.com/1424-8220/16/11/1915/htm>
- [3] Đại học Stanford hợp tác với Phòng thí nghiệm Nghiên cứu Điện tử Volkswagen (ERL) (2005), *Stanley(xe)*, [https://wivi.wiki/wiki/Stanley_\(vehicle\)](https://wivi.wiki/wiki/Stanley_(vehicle))
- [4] PGS.TS Nguyễn Quang Hoan (2006), *Xử lý ảnh*, <https://www.slideshare.net/elLeonNo1/gio-trnh-x-l-nh>
- [5] Trường Cao Đẳng Quốc Tế Hà Nội (2022), *Bạn biết gì về ngôn ngữ lập trình?*, <https://caodangquoctehanoi.edu.vn/tong-quan-ve-ngon-ngu-lap-trinh/>
- [6] Nguyễn Hiền Minh – Phan Thanh Phong (06/2019), *Ứng dụng xử lý ảnh trong hệ thống phân loại sản phẩm*, Đồ án tốt nghiệp đại học, Trường Đại học Sư phạm Kỹ thuật Hồ Chí Minh
- [7] Raspberry Pi Company (2020), *Raspberry Pi 3 Model B*, <https://www.raspberrypi.com/products/raspberry-pi-3-model-b/>
- [8] Arduino (5/2022), *Arduino*, <https://fr.wikipedia.org/wiki/Arduino>
- [9] Công ty Cổ phần Trung tâm Dữ liệu Internet Việt Nam (2012-2022), *Raspberry Pi Camera Module V2 8MP*, <https://raspberrypi.vn/san-pham/raspberry-pi-camera-module-v2-8mp>
- [10] Cửa hàng HSHOP ĐIỆN TỬ VÀ ROBOT (2020), *Cảm biến vật cản hồng ngoại E18-D80NK*, <https://hshop.vn/products/cam-bien-vat-can-hong-ngoai-e18-d80nk-4>
- [11] Công ty Cổ phần Vũ trụ Việt Nam (2018), *Băng tải mini*, <https://www.cosmovina.com/bang-tai-mini>
- [12] ElProCus (2013-2022), *LCD16x2 Pin Configuration and Its Working*, <https://www.elprocus.com/lcd-16x2-pin-configuration-and-its-working/>
- [13] Wikipedia (28/5/2022), *Raspberry Pi OS*, https://en.wikipedia.org/wiki/Raspberry_Pi_OS
- [14] GeeksforGeeks (2020-2022), *OpenCV Python Tutorial*, <https://www.geeksforgeeks.org/opencv-python-tutorial/>

- [15] AnalogDialogue (12/2020), *UART: A Hardware Communication Protocol Understanding Universal Asynchronous Receiver/Transmitter*,
<https://www.analog.com/en/analog-dialogue/articles/uart-a-hardware-communication-protocol.html#:~:text=By%20definition%2C%20UART%20is%20a,going%20to%20the%20receiving%20end.>
- [16] Circuit Basics (2016), *BASICS OF THE I2C COMMUNICATION PROTOCOL*,
<https://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/>

PHỤ LỤC

Code Arduino

```
// tao xanh ,chuoi vang , cam xanh , cam vang , chom chom do, mang cut tim
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 20, 4);
#define CB_1 12 //khai báo chân//
#define CB_2 11
#define CB_3 10
#define CB_4 A0
#define CB_5 A1
#define CB_6 A2
#define start_ A3
#define stop_ 13
#define CB_PH 9

#define xy_lanh_1 4
#define xy_lanh_2 2
#define xy_lanh_3 7
#define xy_lanh_4 5
#define xy_lanh_5 6
#define xy_lanh_6 3
#define bang_tai 8

bool status_CB_1 = false;
bool status_CB_2 = false;
bool status_CB_3 = false;
bool status_CB_4 = false;
bool status_CB_5 = false;
bool status_CB_6 = false;
bool status_CB_PH = false;

int gt_chuoi_xanh = 0;
int gt_chuoi_vang = 0;
int gt_cam_xanh = 0;
int gt_cam_vang = 0;
```



```
int gt_man_xanh = 0;
int gt_mangcut = 0;

int SL_tao_xanh = 0;
int SL_chuoi_vang = 0;
int SL_cam_xanh = 0;
int SL_cam_vang = 0;
int SL_chom = 0;
int SL_chch_do = 0;

int gt_xy_lanh_1 = 0;
int gt_xy_lanh_2 = 0;
int gt_xy_lanh_3 = 0;
int gt_xy_lanh_4 = 0;
int gt_xy_lanh_5 = 0;
int gt_xy_lanh_6 = 0;

int kho[2] = {};
int vi_tri = 0;
int vi_tri_xy_lanh1 = 0;
bool en_hd = false;
bool en_trai_cay = false;
int tg_en = 0;

void setup() {
    Serial.begin(9600);
    lcd.init();
    lcd.clear();
    lcd.backlight();

    lcd.setCursor(0, 0);
    lcd.print(" DE TAI: UNG DUNG ");
    lcd.setCursor(0, 1);
    lcd.print(" XU LY ANH TRONG ");
    lcd.setCursor(0, 2);
    lcd.print(" HE THONG PHAN LOAI ");
```

```
lcd.setCursor(0, 3);
lcd.print("    TRAI CAY    ");

pinMode(CB_1, INPUT_PULLUP); // nhập xuất chân Digital//
pinMode(CB_2, INPUT_PULLUP);
pinMode(CB_3, INPUT_PULLUP);
pinMode(CB_4, INPUT_PULLUP);
pinMode(CB_5, INPUT_PULLUP);
pinMode(CB_PH, INPUT);
pinMode(start_, INPUT_PULLUP);
pinMode(stop_, INPUT_PULLUP);

pinMode(xy_lanh_1, OUTPUT);
pinMode(xy_lanh_2, OUTPUT);
pinMode(xy_lanh_3, OUTPUT);
pinMode(xy_lanh_4, OUTPUT);
pinMode(xy_lanh_5, OUTPUT);
pinMode(xy_lanh_6, OUTPUT);
pinMode(bang_tai, OUTPUT);

digitalWrite(xy_lanh_1, !LOW); //nhập xuất chân Digital//
digitalWrite(xy_lanh_2, !LOW);
digitalWrite(xy_lanh_3, !LOW);
digitalWrite(xy_lanh_4, !LOW);
digitalWrite(xy_lanh_5, !LOW);
digitalWrite(xy_lanh_6, !LOW);
digitalWrite(bang_tai, !LOW);
delay(3000);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print(" DO AN TOT NGHIEP ");
lcd.setCursor(0, 1);
lcd.print("    HE THONG    ");
lcd.setCursor(0, 2);
lcd.print("    DANG DUNG    ");
}
```

```
void loop() {

    // Serial.print((String)digitalRead(CB_1) + '' + (String)digitalRead(CB_2) + ''
+ (String)digitalRead(CB_3) + '' +
    //          (String)digitalRead(CB_4) + '' + (String)digitalRead(CB_5) + '' +
(String)digitalRead(CB_6)
    //          + '' + (String)digitalRead(bang_tai) + "    ");
    //

    if (digitalRead(start_) == 1) {
        lcd.clear();
        en_hd = true;
        digitalWrite(bang_tai, !HIGH);
        lcd.setCursor(0, 0);
        lcd.print(" DO AN TOT NGHIEP ");
        lcd.setCursor(0, 1);
        lcd.print("TX : 0   CHV: 0 ");
        lcd.setCursor(0, 2);
        lcd.print("CX : 0   CV : 0 ");
        lcd.setCursor(0, 3);
        lcd.print("CC : 0   MC: 0 ");
    }

    while (Serial.available()) {
        char t = Serial.read();
    }

    while (en_hd == true) {
        ///Serial.println((String)kho[0]   + ' ' + (String)kho[1]   + ' ' +
(String)gt_chuoi_xanh + '' +
        //          (String)gt_chuoi_vang + ' ' + (String)gt_cam_xanh + ' ' +
(String)gt_cam_vang
        //          + '' + (String)gt_man_xanh + "    ");
        if (digitalRead(CB_PH) == 1 && status_CB_PH == false) {
```

```
    en_trai_cay = true;
    status_CB_PH = true;
}

if (en_trai_cay == true && tg_en <= 40) {
    while (Serial.available()) {
        char t = Serial.read();
    }
    tg_en++;
    if (tg_en==30) {
        digitalWrite(bang_tai, !LOW);
    }
}

if (tg_en == 41) {
    while (Serial.available()) {
        char t = Serial.read();
        if (t == '1' || t == '2' || t == '3' || t == '4' || t == '5' || t == '6') {
            tg_en = 42;
            if (t == '1') {
                SL_tao_xanh++;
                lcd.setCursor(5, 1);
                lcd.print((String)SL_tao_xanh + ' ');
            }
            else if (t == '2') {
                SL_chuoi_vang++;
                lcd.setCursor(16, 1);
                lcd.print((String)SL_chuoi_vang + ' ');
            }
            else if (t == '3') {
                SL_cam_xanh++;
                lcd.setCursor(5, 2);
                lcd.print((String)SL_cam_xanh + ' ');
            }
            else if (t == '4') {
                SL_cam_vang++;
            }
        }
    }
}
```

```

        lcd.setCursor(16, 2);
        lcd.print((String)SL_cam_vang + ' ');
    }
    else if (t == '5') {
        SL_chom++;
        lcd.setCursor(5, 3);
        lcd.print((String)SL_chom + ' ');
    }
    else if (t == '6') {
        SL_chch_do++;
        lcd.setCursor(16, 3);
        lcd.print((String)SL_chch_do + ' ');
    }
    kho[vi_tri] = t - 48;
    vi_tri = (vi_tri == 1) ? 0 : vi_tri + 1;
    digitalWrite(bang_tai, !HIGH);
}
}
}

if (tg_en >= 42 ) {
    while (Serial.available()) {
        char t = Serial.read();
    }
    tg_en++;
    if (tg_en > 180) {
        status_CB_PH = false;
        en_trai_cay = false;
        tg_en = 0;
    }
}

//////////CB_1//////////
if (digitalRead(CB_1) == 1 && status_CB_1 == false) {
    status_CB_1 = true;
    if (kho[vi_tri_xy_lanh1] == 1) {

```

```
    gt_xy_lanh_1++;
    kho[vi_tri_xy_lanh1] = 0;
    vi_tri_xy_lanh1 = (vi_tri_xy_lanh1 == 1) ? 0 : vi_tri_xy_lanh1 + 1;
}
else if (kho[vi_tri_xy_lanh1] != 0) {
    gt_chuoi_vang = kho[vi_tri_xy_lanh1];
    kho[vi_tri_xy_lanh1] = 0;
    vi_tri_xy_lanh1 = (vi_tri_xy_lanh1 == 1) ? 0 : vi_tri_xy_lanh1 + 1;
}
}
if (digitalRead(CB_1) == 0 && status_CB_1 == true) {
    status_CB_1 = false;
}

//////////CB_2//////////
if (digitalRead(CB_2) == 1 && status_CB_2 == false) {
    status_CB_2 = true;
if (gt_chuoi_vang == 2) {
    gt_xy_lanh_2++;
    gt_chuoi_vang = 0;
}
else if (gt_chuoi_vang != 0) {
    gt_cam_xanh = gt_chuoi_vang;
    gt_chuoi_vang = 0;
}
}
if (digitalRead(CB_2) == 0 && status_CB_2 == true) {
    status_CB_2 = false;
}

//////////CB_3//////////
if (digitalRead(CB_3) == 1 && status_CB_3 == false) {
    status_CB_3 = true;
    if (gt_cam_xanh == 3) {
        gt_xy_lanh_3++;
        gt_cam_xanh = 0;
    }
}
```

```
    }
    else if (gt_cam_xanh != 0) {
        gt_cam_vang = gt_cam_xanh;
        gt_cam_xanh = 0;
    }
}

if (digitalRead(CB_3) == 0 && status_CB_3 == true) {
    status_CB_3 = false;
}

//////////CB_4//////////
if (digitalRead(CB_4) == 1 && status_CB_4 == false) {
    status_CB_4 = true;
    if (gt_cam_vang == 4) {
        gt_xy_lanh_4++;
        gt_cam_vang = 0;
    }
    else if (gt_cam_vang != 0) {
        gt_man_xanh = gt_cam_vang;
        gt_cam_vang = 0;
    }
}

if (digitalRead(CB_4) == 0 && status_CB_4 == true) {
    status_CB_4 = false;
}

//////////CB_5//////////
if (digitalRead(CB_5) == 1 && status_CB_5 == false) {
    status_CB_5 = true;
    if (gt_man_xanh == 5) {
        gt_xy_lanh_5++;
        gt_man_xanh = 0;
    }
    else if (gt_man_xanh != 0) {
        gt_man_xanh = 0;
    }
}
```

```
}
if (digitalRead(CB_5) == 0 && status_CB_5 == true) {
    status_CB_5 = false;
}

//////////CB_6//////////
if (digitalRead(CB_6) == 1 && status_CB_6 == false) {
    status_CB_6 = true;
    gt_xy_lanh_6++;
}
if (digitalRead(CB_6) == 0 && status_CB_6 == true) {
    status_CB_6 = false;
}
delay(20);

//////////xy_lanh_1//////////
if (digitalRead(bang_tai) == 0 && gt_xy_lanh_1 >= 1) {
    gt_xy_lanh_1++;
    if (gt_xy_lanh_1 >= 50) {
        gt_xy_lanh_1 = 0;
        digitalWrite(bang_tai, !LOW);
        //Serial.println((String)digitalRead(bang_tai));
        digitalWrite(xy_lanh_1, !HIGH);
        delay(1500);
        digitalWrite(xy_lanh_1, !LOW);
        delay(1500);
        digitalWrite(bang_tai, !HIGH);
    }
}

//////////xy_lanh_2//////////
if (digitalRead(bang_tai) == 0 && gt_xy_lanh_2 >= 1) {
    gt_xy_lanh_2++;
    if (gt_xy_lanh_2 >= 50) {
        gt_xy_lanh_2 = 0;
        digitalWrite(bang_tai, !LOW);
```



```
digitalWrite(xy_lanh_2, !HIGH);
delay(1500);
digitalWrite(xy_lanh_2, !LOW);
delay(1500);
digitalWrite(bang_tai, !HIGH);
}
}

//////////xy lanh 3//////////
if (digitalRead(bang_tai) == 0 && gt_xy_lanh_3 >= 1) {
    gt_xy_lanh_3++;
    if (gt_xy_lanh_3 >= 50) {
        gt_xy_lanh_3 = 0;
        digitalWrite(bang_tai, !LOW);
        digitalWrite(xy_lanh_3, !HIGH);
        delay(1500);
        digitalWrite(xy_lanh_3, !LOW);
        delay(1500);
        digitalWrite(bang_tai, !HIGH);
    }
}

//////////xy lanh 4//////////
if (digitalRead(bang_tai) == 0 && gt_xy_lanh_4 >= 1) {
    gt_xy_lanh_4++;
    if (gt_xy_lanh_4 >= 50) {
        gt_xy_lanh_4 = 0;
        digitalWrite(bang_tai, !LOW);
        digitalWrite(xy_lanh_4, !HIGH);
        delay(1500);
        digitalWrite(xy_lanh_4, !LOW);
        delay(1500);
        digitalWrite(bang_tai, !HIGH);
    }
}

//////////xy lanh 5//////////
```

```
if (digitalRead(bang_tai) == 0 && gt_xy_lanh_5 >= 1) {
    gt_xy_lanh_5++;
    if (gt_xy_lanh_5 >= 50) {
        gt_xy_lanh_5 = 0;
        digitalWrite(bang_tai, !LOW);
        digitalWrite(xy_lanh_5, !HIGH);
        delay(1500);
        digitalWrite(xy_lanh_5, !LOW);
        delay(1500);
        digitalWrite(bang_tai, !HIGH);
    }
}

//////////xy lanh 6//////////
if (digitalRead(bang_tai) == 0 && gt_xy_lanh_6 >= 1) {
    gt_xy_lanh_6++;
    if (gt_xy_lanh_6 >= 50) {
        gt_xy_lanh_6 = 0;
        digitalWrite(bang_tai, !LOW);
        digitalWrite(xy_lanh_6, !HIGH);
        delay(1500);
        digitalWrite(xy_lanh_6, !LOW);
        delay(1500);
        digitalWrite(bang_tai, !HIGH);
    }
}

if (digitalRead(stop_) == 0) {
    en_hd = false;
    gt_xy_lanh_1 = gt_xy_lanh_2 = gt_xy_lanh_3 = 0;
    gt_xy_lanh_4 = gt_xy_lanh_5 = gt_xy_lanh_6 = 0;
    status_CB_1 = status_CB_2 = status_CB_3 = status_CB_4 = status_CB_5 =
status_CB_6 = false;
    kho[0] = kho[1] = 0;
    gt_chuoi_xanh = gt_chuoi_vang = gt_cam_xanh = gt_cam_vang =
gt_man_xanh = gt_mangcut = 0;
```

```
digitalWrite(bang_tai, !LOW);  
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print(" DO AN TOT NGHIEP ");  
lcd.setCursor(0, 1);  
lcd.print(" HE THONG ");  
lcd.setCursor(0, 2);  
lcd.print(" DANG DUNG ");  
}  
}  
}
```

Code Rasoberry:

```
import cv2
import serial
import numpy as np

cap = cv2.VideoCapture(0)
Arduin = serial.Serial('/dev/ttyACM0',9600)
def empty(a):
    pass

def stackImages(scale, imgArray):
    rows = len(imgArray)
    cols = len(imgArray[0])
    rowsAvailable = isinstance(imgArray[0], list)
    width = imgArray[0][0].shape[1]
    height = imgArray[0][0].shape[0]
    if rowsAvailable:
        for x in range(0, rows):
            for y in range(0, cols):
                if imgArray[x][y].shape[:2] == imgArray[0][0].shape[:2]:
                    imgArray[x][y] = cv2.resize(imgArray[x][y], (0, 0), None, scale,
scale)
                else:
                    imgArray[x][y] = cv2.resize(imgArray[x][y],
(imgArray[0][0].shape[1], imgArray[0][0].shape[0]),
                    None, scale, scale)
                if len(imgArray[x][y].shape) == 2: imgArray[x][y] =
cv2.cvtColor(imgArray[x][y], cv2.COLOR_GRAY2BGR)
            imageBlank = np.zeros((height, width, 3), np.uint8)
            hor = [imageBlank] * rows
            hor_con = [imageBlank] * rows
            for x in range(0, rows):
                hor[x] = np.hstack(imgArray[x])
            ver = np.vstack(hor)
    else:
        for x in range(0, rows):
            for y in range(0, cols):
                imgArray[x][y] = cv2.cvtColor(imgArray[x][y], cv2.COLOR_GRAY2BGR)
            imageBlank = np.zeros((height, width, 3), np.uint8)
            hor = [imageBlank] * rows
            hor_con = [imageBlank] * rows
            for x in range(0, rows):
                hor[x] = np.hstack(imgArray[x])
            ver = np.vstack(hor)
```

```

for x in range(0, rows):
    if imgArray[x].shape[:2] == imgArray[0].shape[:2]:
        imgArray[x] = cv2.resize(imgArray[x], (0, 0), None, scale, scale)
    else:
        imgArray[x] = cv2.resize(imgArray[x], (imgArray[0].shape[1],
imgArray[0].shape[0]), None, scale, scale)
    if len(imgArray[x].shape) == 2:
        imgArray[x] =
cv2.cvtColor(imgArray[x], cv2.COLOR_GRAY2BGR)
    hor = np.hstack(imgArray)
    ver = hor
return ver

cv2.namedWindow("TrackBars")
cv2.resizeWindow("TrackBars", 640, 240)
cv2.createTrackbar("Hue Min", "TrackBars", 0, 179, empty)
cv2.createTrackbar("Hue Max", "TrackBars", 179, 179, empty)
cv2.createTrackbar("Sat Min", "TrackBars", 0, 255, empty)
cv2.createTrackbar("Sat Max", "TrackBars", 255, 255, empty)
cv2.createTrackbar("Val Min", "TrackBars", 0, 255, empty)
cv2.createTrackbar("Val Max", "TrackBars", 255, 255, empty)
cv2.createTrackbar("so_anh", "TrackBars", 0, 5, empty)

tra_i_cay = [[29, 74, 23, 56, 242, 186], #tao xanh.
             [2, 68, 101, 30, 193, 253], #CHUOI VANG .
             [7, 61, 23, 29, 181, 163], #CAM XANH .
             [4, 108, 43, 28, 255, 225], #CAM VANG .
             [115, 93, 33, 179, 244, 203], #chom chom .
             [122, 0, 0, 168, 179, 120]] #mang cut

t=0
data = 0
tao = 0
while True:
    success, img = cap.read() #doc anh
    imgHSV = cv2.cvtColor(img, cv2.COLOR_BGR2HSV) #chuyen he RGB
sang HSV

```

```

h_min = cv2.getTrackbarPos("Hue Min", "TrackBars") #CHINH
h_max = cv2.getTrackbarPos("Hue Max", "TrackBars")
s_min = cv2.getTrackbarPos("Sat Min", "TrackBars")
s_max = cv2.getTrackbarPos("Sat Max", "TrackBars")
v_min = cv2.getTrackbarPos("Val Min", "TrackBars")
v_max = cv2.getTrackbarPos("Val Max", "TrackBars") #CHINH
#lower = np.array([h_min, s_min, v_min]) #chinh ma fu
#upper = np.array([h_max, s_max, v_max]) #chinh ma fu

lower = np.array(trai_cay[data][0:3])
upper = np.array(trai_cay[data][3:6])

mask = cv2.inRange(imgHSV, lower, upper)
imgResult = cv2.bitwise_and(img, img, mask=mask)

contours, hierarchy =
cv2.findContours(mask, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_NONE)
#TIM DUONG VIEN
for cnt in contours: #VE DUONG VIEN
    cv2.drawContours(img, cnt, -1, (5, 100, 0), 2)

imgStack = stackImages(0.6, ([img, imgHSV], [mask, imgResult]))
cv2.imshow("Stacked Images", imgStack)

if data == 0 and cv2.countNonZero(mask) > 25000 and
cv2.countNonZero(mask) < 75000:
    Arduin.write('1'.encode())
    print("tao xanh")

if data == 1 and cv2.countNonZero(mask) > 110000 and
cv2.countNonZero(mask) < 190000:
    Arduin.write('2'.encode()) #GUI SANG ARDUINO
    print("chuoai vang")

```

```
        if data == 2 and cv2.countNonZero(mask) > 140000 and
cv2.countNonZero(mask) < 230000:
            Arduin.write('3'.encode())
            print("cam xanh")
        if data == 3 and cv2.countNonZero(mask) > 230000 and
cv2.countNonZero(mask) < 300000:
            Arduin.write('4'.encode())
            print("cam vang")

        if data == 4 and cv2.countNonZero(mask) > 80000 and
cv2.countNonZero(mask) < 180000:
            Arduin.write('5'.encode())
            print("chom chom")

        if data == 5 and cv2.countNonZero(mask) > 130000 and
cv2.countNonZero(mask) < 190000:
            Arduin.write('6'.encode())
            print("mang cut")

    #print(cv2.countNonZero(mask)) #chinh mau
    #print(data)
    cv2.waitKey(1)
    data += 1
    if data == 6:
        data = 0
```