**ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP THÁI NGUYÊN**

**KHOA CƠ KHÍ**

**--------🙠🙡🕮🙣🙢---------**

****

**ĐỒ ÁN TỐT NGHIỆP**

**CHUYÊN NGÀNH CƠ ĐIỆN TỬ**

**THIẾT KẾ HỆ THỐNG PHÂN LOẠI SẢN PHẨM BẰNG XỬ LÍ ẢNH**

|  |  |  |  |
| --- | --- | --- | --- |
| **GIẢNG VIÊN HƯỚNG DẪN** | | **: TS.LÊ THỊ THU THỦY** | |
| **SINH VIÊN** | | **: ĐOÀN QUỐC CƯỜNG** | |
| **MSSV** | | **: K185520114060** | |
| **LỚP** | | **: 54CĐT.02** | |
| **SINH VIÊN** | | **:VŨ VĂN ĐẠT** | |
| **MSSV** | | **: K185520114061** | |
| **LỚP** | | **: 54CĐT.02** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | |  | |  | |
| **KHOA CƠ KHÍ** | |  | | Học kỳ: I | |
| Bộ môn Cơ điện tử | |  | | Năm học: 2022 -2023 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PHIẾU GIAO ĐỀ TÀI ĐỒ ÁN TỐT NGHIỆP** | | | | |
| Thời gian thực hiện: 8 tuần | | | | |
| Ngày giao nhiệm vụ: …./…./202…; | Ngày hoàn thành: …./…./202… | | | |
| Họ và tên sv: ………………………………... | | | MSSV:………………... | Lớp:……………. | |
| Họ và tên sv: ………………………………... | | | MSSV:………………... | Lớp:……………. | |
| Họ và tên sv: ………………………………... | | | MSSV:………………... | Lớp:……………. | |
| Ngày …./…/202… | | Ngày …./…/202… | | |
| **T/M BỘ MÔN CHUYÊN MÔN** | | **CB Hướng dẫn** | | |
| *(ký, ghi rõ họ tên)* | | *(ký, ghi rõ họ tên)* | | |

|  |
| --- |
| 1. **Tên đề tài:** |
| **THIẾT KẾ HỆ THỐNG PHÂN LOẠI SẢN PHẨM THEO HÌNH DẠNG**  **ỨNG DỤNG XỬ LÝ ẢNH VÀ NGÔN NGỮ LẬP TRÌNH C#** |
| 1. **Thông số kỹ thuật yêu cầu:** |
| **…………………………………………………………………………………………….**  **…………………………………………………………………………………………….**  **…………………………………………………………………………………………….**  **…………………………………………………………………………………………….**  **…………………………………………………………………………………………….** |
| 1. **Nội dung các phần thuyết minh và tính toán** |
| * Mở đầu * Chương 1: Tổng quan về hệ thống. * Chương 2: Thiết kế hệ thống chấp hành. * Chương 3: Tính toán và lựa chọn các module. * Chương 4: Thiết kế hệ thống điều khiển. * Chương 5: Thi công sản phẩm. * Kết luận. |
| 1. **Kết quả đạt được** |
| **…………………………………………………………………………………………….**  **…………………………………………………………………………………………….** |

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

|  |
| --- |
| Thái Nguyên, ngày…tháng…năm…..  GIÁO VIÊN HƯỚNG DẪN  (*Ký và ghi rõ họ tên)* |

**NHẬN XÉT CỦA GIÁO VIÊN CHẤM**

|  |
| --- |
| Thái Nguyên, ngày…tháng…năm…..  GIÁO VIÊN HƯỚNG DẪN  (*Ký và ghi rõ họ tên)* |

**MỤC LỤC**

[LỜI NÓI ĐẦU 1](#_Toc123742130)

[CHƯƠNG 1: TỔNG QUAN VỀ HỆ THỐNG 2](#_Toc123742131)

[1.2.1 Thu nhận ảnh (Image Acquisition) 3](#_Toc123742132)

[1.2.2 Phân đoạn (Segmentation) hay phân vùng ảnh 4](#_Toc123742133)

[1.2.3 Biểu diễn ảnh (Image Representation) 4](#_Toc123742134)

[1.2.4 Nhận dạng và nội suy ảnh (Image Recognition and Interpretation) 4](#_Toc123742135)

[1.2.5 Cơ sơ tri thức (Knowledge Base) 4](#_Toc123742136)

[1.2.6 Mô tả 4](#_Toc123742137)

[**1.3 Những vấn đề trong xử lý ảnh** 5](#_Toc123742138)

[1.3.1. Điểm ảnh (Picture Element) 5](#_Toc123742139)

[1.3.2. Ảnh số 5](#_Toc123742140)

[1.3.3. Phân loại ảnh 5](#_Toc123742141)

[1.3.4. Quan hệ giữa các điểm ảnh 5](#_Toc123742142)

[1.3.5. Lọc nhiễu 6](#_Toc123742143)

[1.3.6. Phương pháp phát hiện biên 7](#_Toc123742144)

[1.3.7. Phân đoạn ảnh 8](#_Toc123742145)

[1.3.8. Các phép toán hình thái Morphology 9](#_Toc123742146)

[**1.4 Phân tích yêu cầu** 10](#_Toc123742147)

[**1.5 Kết luận chương :** 10](#_Toc123742148)

[CHƯƠNG 2: THIẾT KẾ HỆ THỐNG CƠ KHÍ 11](#_Toc123742149)

[**2.1 Giới thiệu về bảng thí nghiệm** 11](#_Toc123742150)

[CHƯƠNG 3: TỔNG QUAN VỀ CÁC LINH KIỆN VÀ CƠ CẤU CHẤP HÀNH 16](#_Toc123742151)

[3.1.1 Đặc điểm: 16](#_Toc123742152)

[3.1.2 Nguyên lí hoạt động. 17](#_Toc123742153)

[3.1.3 Góc quan sát camera và ống kính camera 17](#_Toc123742154)

[3.3.1 Van tiết lưu 21](#_Toc123742155)

[3.3.2 Van điện từ 22](#_Toc123742156)

[3.3.2.2 Nguyên lí hoạt động: 22](#_Toc123742157)

[CHƯƠNG 4: GIỚI THIỆU VỀ PHẦN MỀM VÀ LẬP TRÌNH HỆ THỐNG 32](#_Toc123742158)

[4.1.1 Giới thiệu về ngôn ngữ C# 32](#_Toc123742159)

[4.1.2 Đặc trưng của C# 32](#_Toc123742160)

[36](#_Toc123742161)

[ Kết quả đạt được 60](#_Toc123742162)

[ Những mặt hạn chế 60](#_Toc123742163)

LỜI NÓI ĐẦU

Ngày nay, sự phát triển mạnh mẽ của khoa học công nghệ những trang thiết bị hiện đại phục vụ công nghiệp hóa hiện đại hóa đất nước. Gớp phần vào sự phát triển đó thì ngành kỹ thuật điện tử đã góp phần không nhỏ trong sự nghiệp xây dựng và phát triển đất nước. Trong đó sự tích hợp mạch điện-điện tử ngày càng trở nên thiết yếu khi mà công nghệ ngày càng phát triển hơn tiến với thời đại của vi xử lý vi mạch những mạch cồng kềnh chiếm nhiều diện tích đã được loại bỏ dần và thay vào đó là các mạch siêu nhỏ gọn ngày càng được ưa chuộng.

Với yêu cầu nghiên cứu và tìm hiểu về một thiết bị đo có vai trò rất quan trọng trong đời sống và được sử dụng phổ biến trong tương lai đó là “ hệ thống phân loại sản phẩm”. Chúng em đã vận dụng những kiến thức và năng lực tích lũy trong thời gian học tập ở nhà trường và tài liệu khác để hoàn thành đề tài “ **Hệ thống phân loại sản phẩm bằng xử lý ảnh: Hình dạng(C#)**”. Được sự hướng dẫn chỉ bảo tận tình của thầy giáo và vận dụng tất cả các kỹ năng kiến thức trong suốt thời gian học tập tại trường chúng em đã hoàn thành theo đúng yêu cầu. Với đề tài này chúng em mong muốn có kiến thức hiểu biết sâu sắc và thực tế hơn góp một phần nhỏ vào sự phát triển của cơ điện tử và nền tự động hóa nước nhà. Tuy nhiên những kiến thức chúng em tích lũy được vẫn còn thiếu sót và còn thiếu kinh nghiệm trong việc trình bày, do đó chúng em rất mong nhận được sự góp ý và bổ sung của các thầy cô và các bạn theo dõi đề tài.

Chúng em xin chân thành cảm ơn thầy giáo hướng dẫn **TS. Lê Thị Thu Thủy** cùng các thầy(cô) trong bộ môn Cơ điện tử, Trường Đại Học Kỹ Thuật Công Nghiệp đã hướng dẫn và tạo điều kiện giúp đỡ chúng em hoàn thành đề tài này.

Chúng em xin chân thành cảm ơn!

CHƯƠNG 1: TỔNG QUAN VỀ HỆ THỐNG

1.1 Đặt vấn đề

Ngày nay, xã hội càng phát triển, công nghiệp hóa,hiện đại hóa ngày càng được nâng cao để phát triển đất nước và cải thiện cuộc sống của người dân .Vì vậy việc ứng dụng khoa học kĩ thuật ngày càng rộng rãi, phổ biến và mang lại hiệu quả cao trong hầu hết các lĩnh vực kinh tế, kĩ thuật cũng như đời sống xã hội.

Xét điều kiện cụ thể ở nước ta trong công cuộc công nghiệp hóa,hiện đại hóa

Sử dụng ngày càng nhiều thiết bị hiện đại để điều khiển tự động các quá trình sản xuất, gia công và chế biến sản phẩm…Điều này dẫn đến việc hình thành các hệ thống sản xuất linh hoạt , cho phép tự động hóa mức cao trên cơ sở sử dụng các máy CNC , robot công nghiệp . Trong đó có một khâu quan trọng ảnh hưởng đến chất lượng của sản phẩm là hệ thống phân loại sản phẩm. Hệ thống phân loại nhằm chia ra các nhóm có cùng thuộc tính với nhau để thực hiện đóng gói hay loại bỏ sản phẩm hỏng.Hiện nay để phân loại sản phâm người ta thường sử dụng các loại cảm biến với các chức năng khác nhau để phân loại sản phẩm để phân loại sản phẩm theo mong muốn như cảm biến phân loại theo màu sắc, cảm biến phân loại theo hình dạng… Những cảm biến này có ưu điểm chung là quá trình lắp đặt và vận hành tương đối đơn giản nhưng lại dễ gây nhiễu. Do đó , dựa trên nền tảng kiến thức đã học , vốn hiểu biết về điện tử và công nghệ xử lí ảnh cùng với sự đồng ý cảu giáo viên hướng dẫn **TS.Lê Thị Thu Thủy** , nhóm chúng em chọn đề tài “THIẾT KẾ HỆ THỐNG PHÂN LOẠI SẢN PHẨM BẰNG XỬ LÍ HÌNH ẢNH”.

1.2 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 mẻ 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.

Xử lý ảnh là kỹ thuật áp dụng trong việc tăng cường và xử lý các ảnh thu nhận từ các thiết bị như camera, webcam… Do đó, xử lý ảnh đã được ứng dụng và phát triển trong rất nhiều lĩnh vực quan trọng như:

* Trong lĩnh vực quân sự: xử lý và nhận dạng ảnh quân sự.
* Trong lĩnh vực giao tiếp người máy: nhận dạng ảnh, xử lý âm thanh, đồ họa.
* Trong lĩnh vực an, bảo mật: nhận diện khuôn mặt người, nhận diện vân tay, mẫu mắt, …
* Trong lĩnh vực giải trí: trò chơi điện tử.
* Trong lĩnh vực y tế: Xử lý ảnh y sinh, chụp X quang, MRI, …

Các phương pháp xử lý ảnh bắt đầu từ các ứng dụng chính: nâng cao chất lượng và phân tích ảnh. Ứng dụng đầu tiên được biết đến là nâng cao chất lượng ảnh báo được truyền từ Luân đôn đến New York từ những năm 1920. Vấn đề nâng cao chất lượng ảnh có liên quan tới phân bố mức sáng và độ phân giải của ảnh. Việc nâng cao chất lượng ảnh được phát triển vào khoảng những năm 1955. Điều này có thể giải thích được vì sau thế chiến thứ hai, máy tính phát triển nhanh tạo điều kiện cho quá trình xử lý ảnh số được thuận lợi hơn. Năm 1964, máy tính đã có khả năng xử lý và nâng cao chất lượng ảnh từ mặt trăng và vệ tinh Ranger 7 của Mỹ bao gồm: làm nổi đường biên, lưu ảnh. Từ năm 1964 đến nay, các phương tiện xử lý, nâng cao chất lượng, nhận dạng ảnh phát triển không ngừng. Các phương pháp tri thức nhân tạo như mạng nơ-ron nhân tạo, các thuật toán xử lý hiện đại và cải tiến, các công cụ nén ảnh ngày càng được áp dụng rộng rãi và thu được nhiều kết quả khả quan hơn.

Sau đây, ta sẽ xét các bước cần thiết trong quá trình xử lý ảnh. Đầu tiên, ảnh tự nhiên từ thế giới bên ngoài được thu nhận qua các thiết bị thu (như Camera, máy chụp ảnh). Trước đây, ảnh thu qua Camera là các ảnh tương tự (loại Camera ống kiểu CCIR). Gần đây với sự phát triển của công nghệ, ảnh màu hoặc đen trắng được lấy ra từ Camera, sau đó nó được chuyển trực tiếp thành ảnh số tạo thuận lợi cho xử lý tiếp theo. Mặt khác ảnh có thể được quét từ vệ tinh chụp trực tiếp bằng máy quét ảnh. Các bước cơ bản trong xử lý ảnh bao gồm:

Diagram

Description automatically generated

*Hình 1.2 Các bước cơ bản trong xử lý ảnh*

1.2.1 Thu nhận ảnh (Image Acquisition)

Ảnh có thể nhận qua camera màu hoặc trắng đen. Thường ảnh nhận qua camera là ảnh tương tự (loại camera ống chuẩn CCIR với tần số 1/25, mỗi ảnh 25 dòng), cũng có loại camera đã số hóa (như loại CCD – Change Coupled Device) là loại photodiot tạo cường độ sáng tại mỗi điểm ảnh.

Camera thường dùng là loại quét dòng, ảnh tạo ra có dạng hai chiều. Chất lượng ảnh thu nhận được phụ thuộc vào thiết bị thu, vào môi trường (ánh sáng, phong cảnh).

Sau bộ thu nhận, ảnh có thể nhiễu độ tương phản thấp nên cần đưa vào bộ tiền xử lý để nâng cao chất lượng. Chức năng chính của bộ tiền xử lý là lọc nhiễu, nâng độ tương phản để làm ảnh rõ hơn, nét hơn.

1.2.2 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 phụ thuộc rất nhiều vào công đoạn này.

1.2.3 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 gọi là trích chọn đặc trưng (Feature Extration) 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. Ví dụ: trong nhận dạng ký tự trên phong bì thư, chúng ta miêu tả các đặc trưng của từng ký tự giúp phận biệt ký tự này với ký tự khác.

1.2.4 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 (hoặc lưu) từ trước. Nội suy là phán đoán theo ý nghĩa trên cơ sở nhận dạng. Ví dụ: một loạt chữ số và nét gạch ngang trên phong bì thư có thể nội suy thành mã điện thoại. Có nhiều cách phân loại ảnh khác nhau về ảnh. Theo lý thuyết về nhận dạng, các mô hình toán học về ảnh được phân theo hai loại nhận dạng ảnh cơ bản:

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

Một số đối tượng nhận dạng khá phổ biến hiện nay 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…

1.2.5 Cơ sơ tri thức (Knowledge Base)

Như đã nói ở trên, 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 nhiễu. 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 mong muốn 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.

1.2.6 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.

## **1.3 Những vấn đề trong xử lý ảnh**

1.3.1. Điểm ảnh (Picture Element)

Là đơn vị cơ bản nhất để tạo nên một bước 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.

1.3.2. 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 thể hiện rõ nét các đặt điểm của tấm hình càng làm cho tấm ảnh trở nên thực và sắc nét hơn. Một hình ảnh là một 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.

1.3.3. Phân loại ảnh

Mức xám của đi

ểm 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 265, tức là dùng 1byte để biểu diễn mức xám. Trong đó:

* Ảnh nhị phân: Là ảnh có 2 mức trắng và đen, chỉ có 2 giá trị 0 và 1 và chỉ sử dụng 1bit dữ liệu trên 1 điểm ảnh
* Ảnh đen trắng: Là ảnh có hai màu đen, 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 khoảng 16,7 triệu mức màu.

1.3.4. 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.

Shape

Description automatically generated

*Hình 1.3.4: Lân cận 4 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) (1.1)

Liên kết ảnh: Các mối liên kết của ảnh được xem như là mối liên kiết của 2 điểm ảnh gần nhau, có 3 loại liên kết: liên kết 4, liên kết 8, lên kết m (liên kết hỗn hợp). Trong ảnh đa mức xám, ta có thể đặt V chứa nhiều giá trị như V = {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 V.

1.3.5. 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 sử dụng 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 nhiễu bằng lọc tuyến tính: Khi chụp ảnh có thể xuất hiện nhiều loại nhiễu vào quá trình xử lý ảnh, nên ta cần phải lọc nhiễu. Gồm các phương pháp cơ bản lọc trung bình, lọc thông thấp, … Ví dụ lọc trung bình: Với lọc trung bình, mỗi điểm ảnh được thay thế bằng trung bình trọng số của các điểm lân cận.[1]

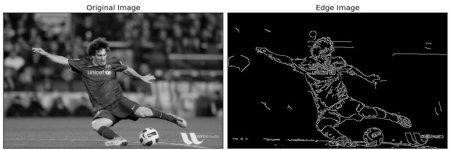
Làm trơn nhiễu bằng lọc phi tuyến: Các bộ lọc phi tuyến cũng hay được dùng trong kỹ thuật tăng cường ảnh. Một số phương pháp lọc cơ bản bộ lọc trung vị, lọc ngoài, … Với lọc trung vị, điểm ảnh đầu vào sẽ được thay thế bởi trung vị các điểm ảnh còn lọc giả trung vị sẽ dùng trung bình cộng của hai giá 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 số thường được chọn sao cho số điểm ảnh trong cửa sổ là lẻ.

Lọc ngoài: Giả thiết có ngưỡng nào đó 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.

1.3.6. Phương pháp phát hiện biên

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



*Hình 1.3.6 : Hình tách biên*

Điểm biên: Một điểm ảnh được coi 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 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 cục bộ 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, xét ví dụ sau: khi người họa sỹ muốn vẽ một danh nhân, họa sỹ chỉ cần vẽ vài đường nứt tốc họa mà không cần vẽ một cách đầy đủ.

Như vậy, phát hiện biên một cách lý tưởng là phát hiện được tất cả các đường biên trong các đối tượng. Định nghĩa toán học của biên ở trên 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: Tách biên theo đạo hàm bậc một: 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 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 (tìm được từ bước 2) 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 kiết 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.

1.3.7. 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 cần khảo sát ra khỏi phần nội

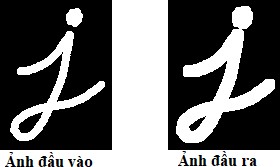
dung còn lại của ảnh, hay phân chia các đối tượng trong ảnh thành những đối tượng riêng biệt. Như vậy quá trình phân đoạn ảnh là quá trình giảm bớt số lượng thông tin trong ảnh và chỉ giữ lại những thông tin cần thiết cho ứng dụng. Do đó phân đoạn ảnh là quá trình loại bỏ các đối tượng không quan tâm 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 lớn 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:

* Các kỹ thuật cục bộ: dựa trên các đặc tính cục bộ của các điểm ảnh và các lân cận của nó
* Các kỹ thuật tổng thể: phân đoạn một ảnh dựa trên cơ sở của thông tin lấy từ tổng thể như sử dụng biểu đồ mức xám histogram.
* Các 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 các nối các điểm ảnh sao cho nó đồng nhất với nhau.

1.3.8. Các phép toán hình thái Morphology

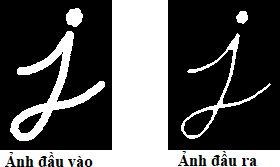
Khái niệm Morphology trong xử lý ảnh số khởi nguồn từ một ngành của sinh học, nghiên cứu về hình thể và cấu trúc của động thực vật. Đây là một công cụ giúp rút trích các thành phần trong ảnh nhị phân, biễu diễn và mô tả chúng dưới dạng các vùng hoặc dạng như các đường biên, xương và bao lồi. Kỹ thuật morphology cũng được áp dụng trên ảnh xám cho các công đoạn tiền/hậu xử lý ảnh (pre or post processing). Các phép toán đầu tiên trong Morphology được phát triển vào những năm 1964 bởi Georges Matheron (1930 - 2000) và Jean Serra (1940) tại trường đại học École des Mines de Paris, Pháp. Matheron là tiến sĩ hướng dẫn của Jean, họ cố gắng xác định số lượng đặc tính của khoáng sản thông qua “thin cross section” và công việc cho ra kết quả là một phương pháp tiếp cận mới, cũng như sự tiến bộ trong hình học tích phân và tô pô. Từ đó đến hết năm 1970, Morphology xử lý cơ bản với các ảnh nhị phân, tạo ra các phép toán và kĩ thuật như: Dilation, Erosion, Opening, Closing…

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



*Hình 1.3.8 a: Phép giản*

Phép co (Erosion): Phép toán Erosion là thao tác xói mòn/co hẹp các đối tượng ảnh đơn sắc. Nếu như phép dãn có thể nói là thêm điểm ảnh vào trong đối tượng ảnh, làm cho đối tượng ảnh trở nên lớn hơn thì phép co sẽ làm cho đối tượng ảnh trở nên nhỏ hơn, ít điểm ảnh.



*Hình 1.3.8 b : 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 hổ 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, Như phải kể đến Matlap, hay ngôn ngữ Python, …

## **1.4 Phân tích** yêu **cầu**

* Có khả năng phân loại 3 loại sản phẩm: tròn-vuông-tam giác.
* Thông số kĩ thuật cần đạt được:

- Hệ thống đảm bảo độ cứng vững kết cấu, độ chính xác khi làm việc.

- Hệ thống đảm bảo an toàn khi vận hành .

- Hệ thống làm việc ổn định, lâu dài.

- Dễ bảo trì , sữa chữa khi có sự cố.

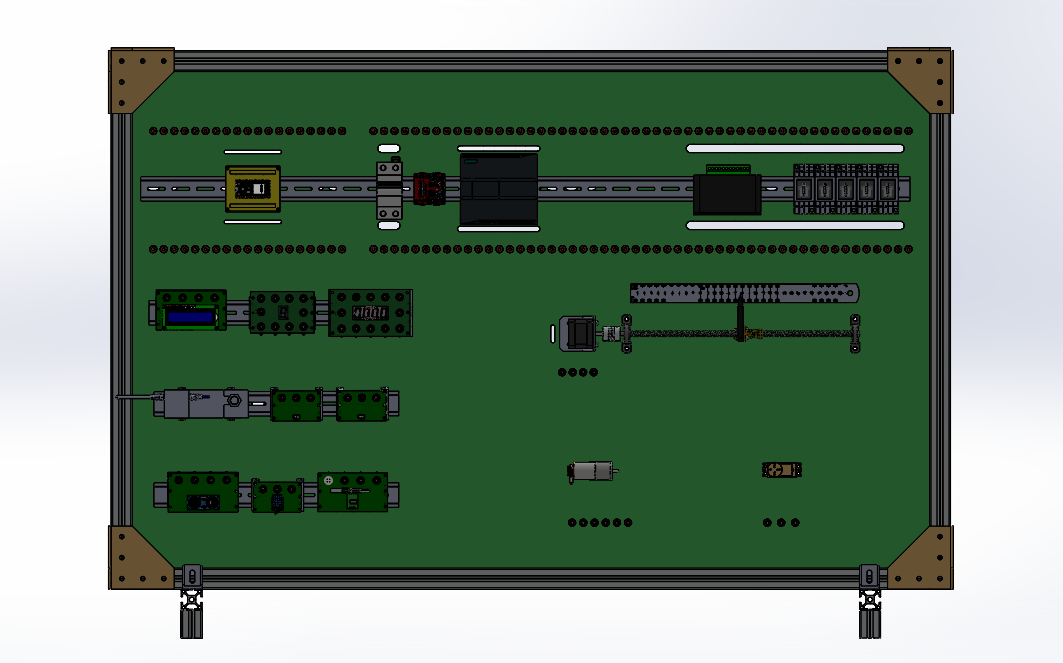
- Có khả năng kết hợp với những hệ thống khác để tạo nên một dây chuyền sản xuất mở rộng.

**1.5 Kết luận chương :**

Nêu được tổng quan về một hệ thống phân loại nói chung và hệ thống phân loại sản phẩm bằng sử lí ảnh: hình dạng C# nói riêng và sự cần thiết của hệ thống trong sản xuất. Để hoạt động được thì hệ thống cũng cần phải đảm bảo an toàn khi vận hành , chính xác, làm việc ổn định lâu dài và dễ bảo trì sửa chữa khi có sự cố .

CHƯƠNG 2: THIẾT KẾ HỆ THỐNG CƠ KHÍ

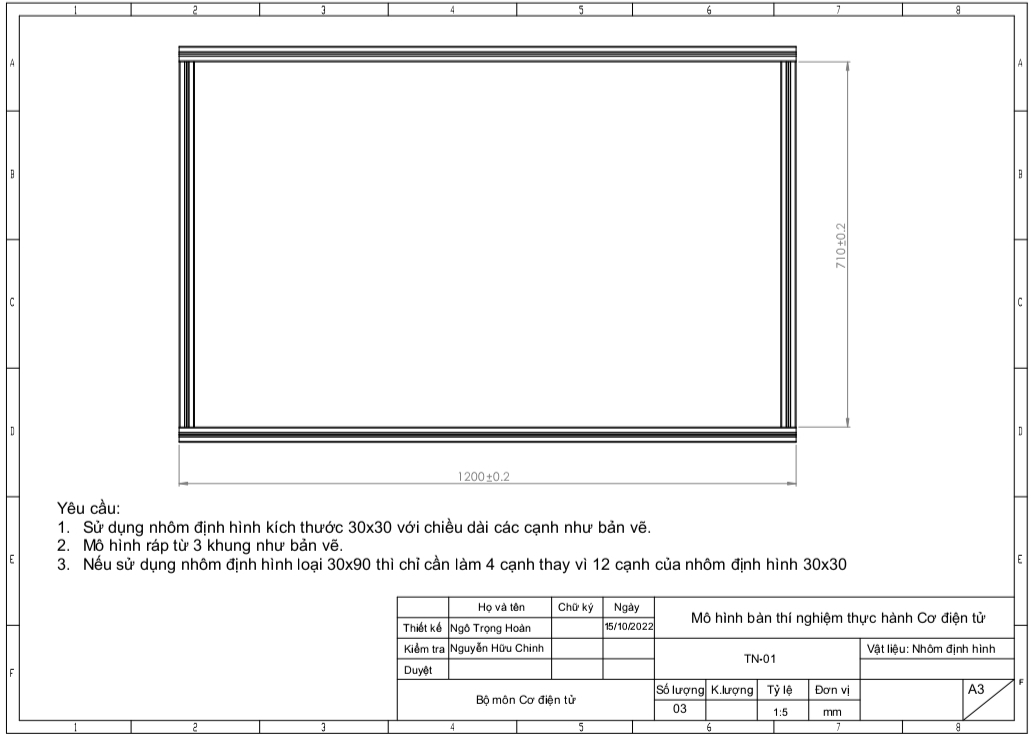
## **2.1 Giới** thiệu **về bảng thí nghiệm**



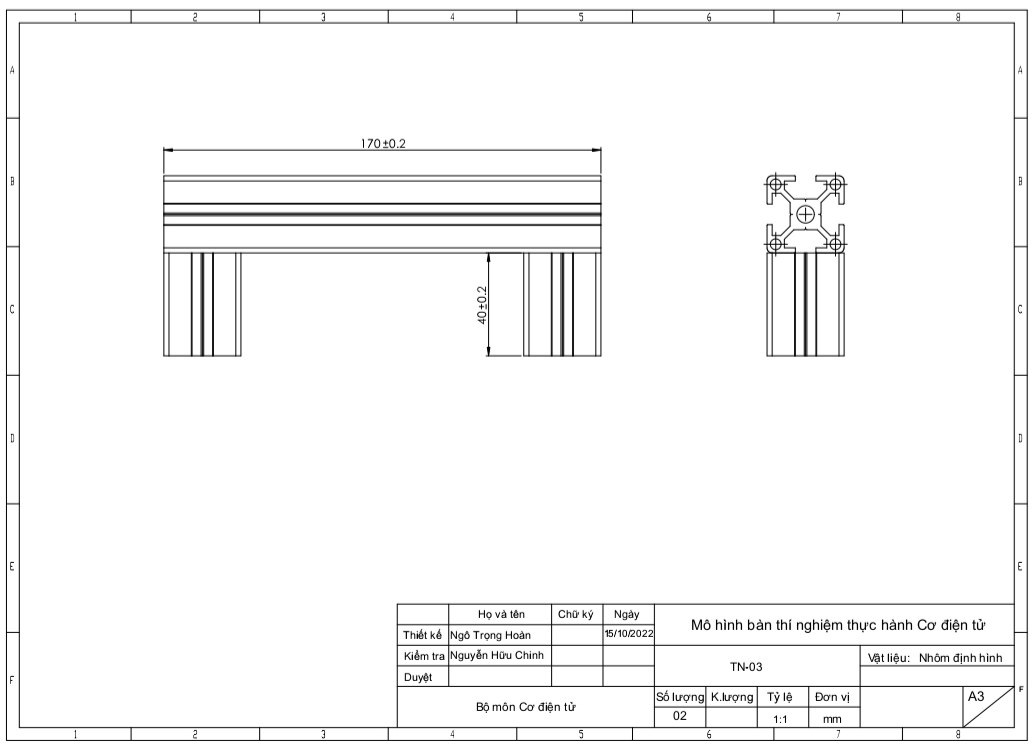
*Hình 2.1: Bảng thí nghiệm*

* Cấu tạo của bảng thực hành:
* Khung: cấu tạo từ các thanh nhôm định hình 30\*30 và tấm mica.
* Các bộ điều khiển: PLC fx3U 24MT, ESP32.
* Cảm biến: cảm biến quang, cảm biến nhiệt độ (PT100), cảm biến cân nặng (loadcell), cảm biến độ ẩm (DHT11), cảm biến khoảng cách (HRC04),…
* Hiển thị: màn hình LCD, led 7 thanh.
* Cơ cấu chấp hành: động cơ encoder GA25, động cơ servo MG996, động cơ bước.
* Aptomat 2 pha, relay 14 chân.
* Nguồn tổ ong

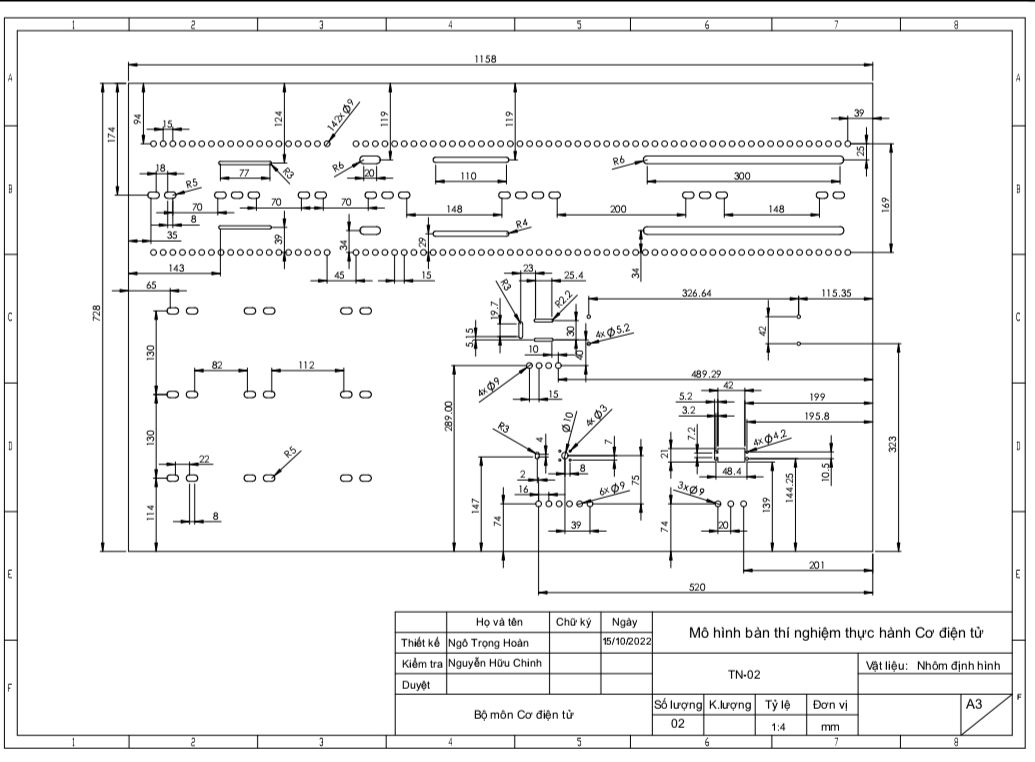
Bộ thực hành thí nghiệm có các thiết bị cơ bản giúp tìm hiểu và thực hành các đề tài cơ bản, giúp hiểu hơn về các thiết bị điện tử và các cảm biến cũng như các cơ cấu chấp hành, giúp sinh viên có cơ hội được tiếp xúc và tiếp cận các môn học một cách hấp dẫn và thực tiễn nhất.

2.2 Các bản vẽ thiết kế

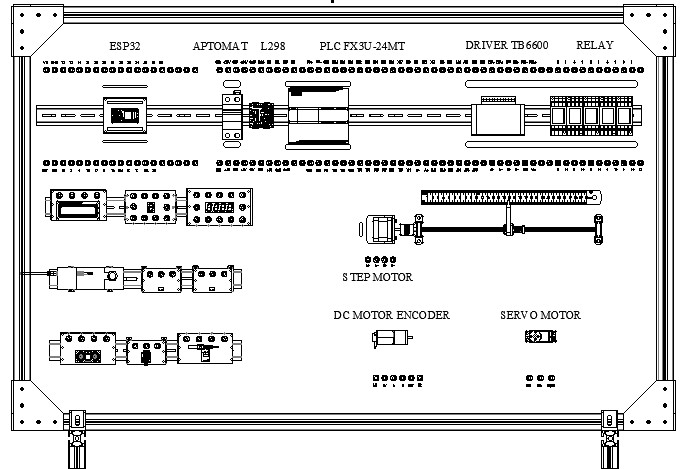
*Hình 2.2a: Khung nhôm*



*Hình 2.2b: Đế đỡ*



*Hình 2.2c: Tấm Mica*

*****Hiình 2.2d: Sơ đồ chân bảng thực hành*

CHƯƠNG 3: TỔNG QUAN VỀ CÁC LINH KIỆN VÀ CƠ CẤU CHẤP HÀNH

3.1 **Camera**

*Hiình 3.1: Một số loại camera trong thực tế*

Camera là một thiết bị ghi hình mà bạn có thể ghi lại hình ảnh trong một khoảng thời gian nào đó, lưu trữ và xem lại bất cứ khi nào bạn muốn

Chức năng của Camera là :

* **Tính năng giám sát 24/7**
* **Tính năng quan sát hồng ngoại ban đêm**
* **Tính năng ghi hình và lưu trữ hình ảnh**
* **Tính năng tìm kiếm và xem lại hình ảnh**
* **Tính năng xem video hình ảnh camera quan sát qua mạng.**

3.1.1 Đặc điểm:

* **Mỗi Camera thường có 3 dây:**
* Tín hiệu hình.
* RS485.
* Dây cấp nguồn.
* **Có 3 cách để phân loại Camera:**
* Phân loại theo kĩ thuật hình ảnh.
* Phân loại theo đường truyền.
* Phân loại theo tính năng sử dụng.
* **Các thành phần cơ bản của camera :**
* Camera: Phần thu hình.
* Ống kính: Để tăng độ quan sát (chỉ bắt buộc đối với Camera màu thân lớn).
* Adapter: Cung cấp nguồn cho Camera.
* Chân đế: Để bắt vào tường.
* Jack cắm: Để nối camera với dây Cable.
* Dây Cable: Để truyền tín hiệu.
* Card: Để kết và thu hình vào máy vi tính.
* Đầu ghi kỹ thuật số: Để thu hình và phát hình thông qua màn hình chuyên dụng của Camera hoặc Tivi.

3.1.2 Nguyên lí hoạt động.

* Hình ảnh truyền qua ống kính camera và hình thành trên mặt CCD - ma trận sử dụng CFA - màng lọc màu cung cấp thông tin các bộ chuyển đổi analog sang số, thiết lập nên tín hiệu số.
* Tín hiệu số sau khi được hình thành sẽ được truyền tới bộ chip xử lý.
* Bộ chip và bộ khuếch đại xử lý thông tin tạo nên tín hiệu hình ảnh, video.
* V-Driver: chịu trách nhiệm cho sự hình thành độ sáng và quét ngang của CDD bên trong camera.
* IRIS Driver: làm nhiệm vụ điều khiển ống kính camera đồng bộ với các tín hiệu bên ngoài.
* Bộ nhớ flash: bên trong chứa các phàn mềm chip xử lý các hình ảnh, âm thanh, tín hiệu,... Thông qua đó, người dùng có thể truy cập qua các phím điều khiển chỉnh sửa một số thiết lập.RS485: làm nhiệm vụ kiểm soát điều khiển bộ vi xử lý . Ví dụ điều khiển khởi động đèn hồng ngoại ban đêm của camera,...
* Chip I/O: làm nhiệm vụ một bộ giao tiếp ra vào với bàn điều khiển xuất tín hiệu điều khiển led tín hiệu báo động chuyển động. Đối với chip xử lý trong camera hiện đại được tích hợp nhiều chức năng bên trong như bù sáng, tự động điều chỉnh độ tương phản,...

3.1.3 Góc quan sát camera và ống kính camera

Trong tài liệu kỹ thuật thường không ghi góc mở, mà ghi thông số tiêu cự thay cho góc mở. Có thể sử dụng bảng quy đổi sau:

|  |  |
| --- | --- |
| **Tiêu cự** | **Góc mở** |
| 2.1 mm | 138°36’ |
| 2.5 mm | 100°24’ |
| 2.8 mm | 85°36’ |
| 3.6 mm | 79°36’ |
| 4 mm | 61°36’ |
| 6 mm | 34°24’ |
| 8 mm | 34°24’ |
| 12 mm | 22°42’ |
| 16 mm | 21°30’ |

*Bảng 3.1.3: Bảng quy đổi tiêu cự camera và góc mở*

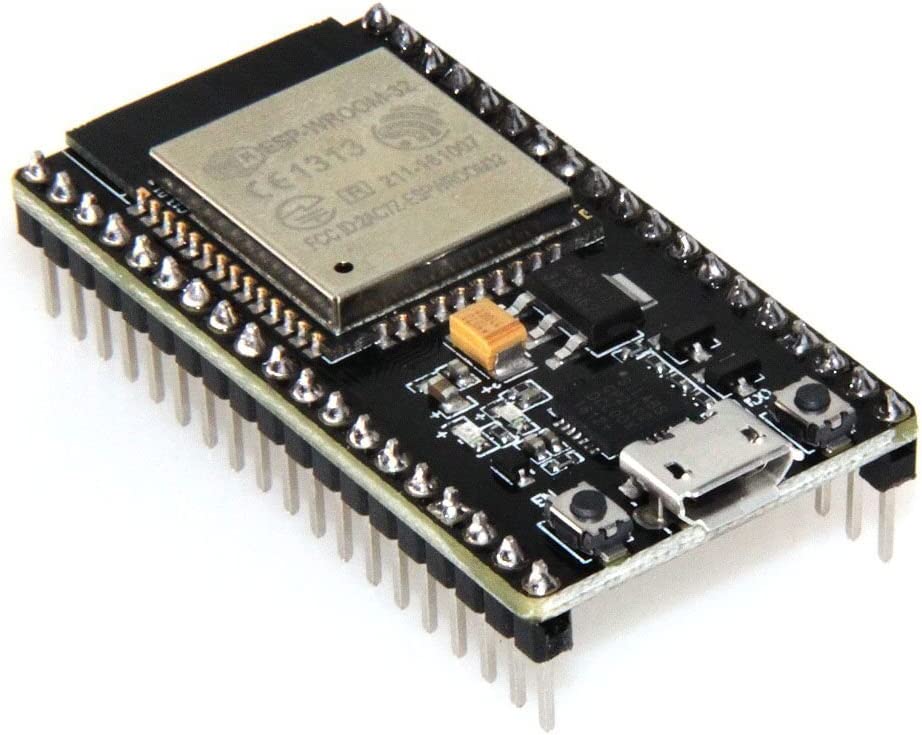
**Các thông số khác:**

* Auto Gain Control: Tự động kiểm soát độ lợi
* Backlight Compensation: Bù ánh sáng ngược
* Auto Electronic Shutter: Tự động chống sốc điện.

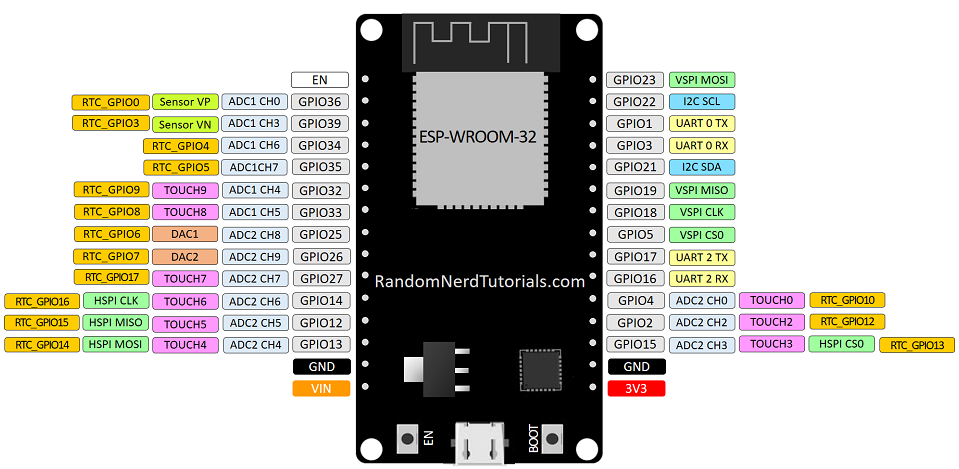
**Tính ứng dụng của camera:**

Hiện nay camera được xem là phương pháp thuận lợi được sử dụng thông dụng tại gia đình, nơi làm việc, công ty,... Đây là sản phẩm khoa học có hệ thống quan sát kỹ thuật tiên tiến đáp ứng nhu cầu trong thời đại phát triển mạnh mẽ như hiện nay. Các thế hệ camera ngày càng tiên tiến, tích hợp nhiều tính năng vượt trội có thể ghi lại hình ảnh trong một khoảng thời gian trong một phạm vi không gian nhất định và có thể lưu trữ trong thời gian dài. Với chức năng ghi lại hình ảnh, camera ngày nay được ứng dụng trong nhiều lĩnh vực khác nhau như giám sát, an ninh, giáo dục,... Một hệ thống đặt tại vị trí thích hợp giúp bạn quan sát, theo dõi toàn bộ ngôi nhà, nhà máy, ghi hình dạy học online,...

3.2 Vi điều khiển ESP 32



*Hình 3.2a: Hình ảnh vi điều khiển ESP32*



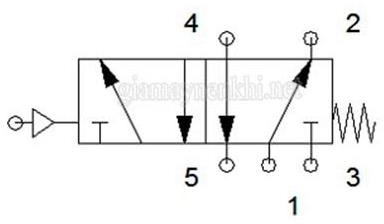
*Hình 3.2.b: Hình ảnh vi điều khiển ESP32*

* Cấu hình ESP32:
* CPU
* CPU: Xtensa Dual-Core LX6 microprocessor.
* Chạy hệ 32 bit
* Tốc độ xử lý 160MHZ up to 240 MHz
* Tốc độ xung nhịp đọc flash chip 40mhz --> 80mhz (tùy chỉnh khi lập trình)
* RAM: 520 KByte SRAM
* 520 KB SRAM liền chip –(trong đó 8 KB RAM RTC tốc độ cao – 8 KB RAM RTC tốc độ thấp (dùng ở chế độ DeepSleep).
* Hỗ trợ 2 giao tiếp không dây
* Wi-Fi: 802.11 b/g/n/e/i
* Bluetooth: v4.2 BR/EDR and BLE
* Hỗ trợ tất cả các loại giao tiếp
* 8-bit DACs( digital to analog) 2 cổng
* Analog(ADC) 12-bit 16 cổng.
* I²C – 2 cổng
* UART – 3 cổng
* SPI – 3 cổng (1 cổng cho chip FLASH )
* I²S – 2 cổng
* SD card /SDIO/MMC host
* Slave (SDIO/SPI)
* Ethernet MAC interface with dedicated DMA and IEEE 1588 support
* CAN bus 2.0
* IR (TX/RX)
* Băm xung PWM (tất cả các chân )
* Ultra low power analog pre-amplifier’
* Cảm biến tích hợp trên chip esp32
* 1 cảm biến Hall (cảm biến từ trường)
* 1 cảm biến đo nhiệt độ
* Cảm biến chạm (điện dung) với 10 đầu vào khác nhau.
* Bảo mật
* IEEE 802.11 standard security features all supported, including WFA, WPA/WPA2 and WAPI
* Secure boot
* Flash encryption
* 1024-bit OTP, up to 768-bit for customers
* Cryptographic hardware acceleration: AES, SHA-2, RSA, elliptic curve cryptography (ECC), random number generator (RNG)
* Nguồn điện hoạt động:
* Nhiệt độ hoạt động -40 + 85C
* Điện áp hoạt động: 2.2-3.6V
* Số cổng GPIOs : 34
* Điểm mạnh của ESP32 so với các dòng khác

Có thể nói ESP32 là sự nâng cấp hoàn hảo của ESP8266, với ESP8266 phù hợp với các dự án nhỏ và tiết kiệm chi phí. ESP32 lại phù hợp với các dự án phức tạp hơn, tốc độ xử lý cao hơn và tích hợp nhiều ngoại vi mạnh mẽ hơn.

ESP8266 là 17 chân GPIO, ADC độ phân giải 10 bit, 8 kênh PWM mềm trong khi đó ESP 32 có tới 30/36 chân GPIO, 18 kênh ADC độ phân giải 12-bit, 16 kênh PWM mềm, Touch Sensor, Hall Effect Sensor, Ethernet MAC Interface, Cảm biến nhiệt độ được tích hợp sẵn.

Về bộ nhớ ESP32 có thêm 4MB External Flash và 520KB SRAM (static random access memory) trong đó 8 KB RAM RTC tốc độ cao – 8 KB RAM RTC tốc độ thấp (dùng ở chế độ DeepSleep).

3.3 Van dùng trong hệ thống: Van đảo chiều 5/2

*Hình 3.3: Van điện từ một đầu điện*

Thông số kỹ thuật van:

* Loại van: 5 cửa 2 vị trí
* Điện áp đầu vào: 220V
* Kích thước cửa: Cửa vào, Cửa ra: 1/4”, Cửa xả 1/8”
* Môi trường làm việc: Khí nén
* Áp suất làm việc: 0.15 - 0.8 MPa
* Nhiệt độ cho phép: -5°C đến 60°C
* Sai số điện áp cho phép: ±10%
* Thời gian đáp ứng: 0.05s

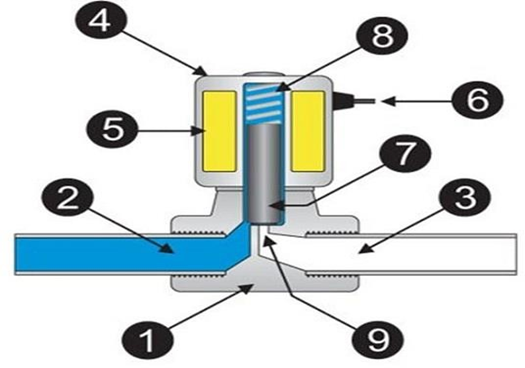
3.3.1 Van tiết lưu

*Hình 3.3.1: Van tiết lưu*

Van tiết lưu là loại khớp nối nhanh dây hơi, 1 đầu có gen ngoài vặn vào thiết bị, 1 đầu có nối nhanh vào dây hơi, đặc biệt có thêm núm điều chỉnh áp lực của hơi.Van có chất liệu 1 đầu bằng hợp kim, phần còn lại làm bằng nhựa PE có độ bền cao, núm điều chỉnh bằnghợp kim.

3.3.2 Van điện từ

Van điện từ là một thiết bị cơ điện được sử dụng để kiểm soát dòng chảy chất lỏng hoặc khí. Van điện từ hay còn gọi là solenoid valve hoạt động với điện áp 220V hoặc 24V và được điều khiển thông qua một cuộn dây.

*  Cấu tạo van điện từ:

Hình 3.3.2: Cấu tạo van điện từ

1. Thân van
2. Môi chất
3. Ống rỗng dẫn lưu chất đi qua
4. Vỏ cuộn coil
5. Coil điện hay được gọi là đầu điện
6. Dây điện
7. Trục van
8. Lò xo
9. Khe hở để lưu chất đi qua

3.3.2.2 Nguyên lí hoạt động:

Dưới sự tác động của dòng điện được cấp chạy qua cuộn dây, từ trường sẽ được sinh ra tạo một lực đẩy piston di chuyển đi lên nhằm mở thông van. Đây là nguyên tắc hoạt động cơ bản được sử dụng nhằm đóng mở van điện từ. Tuy nhiên với những van có đặc điểm cấu tạo khác nhau thì quá trình này cũng sẽ có những điểm không giống nhau.

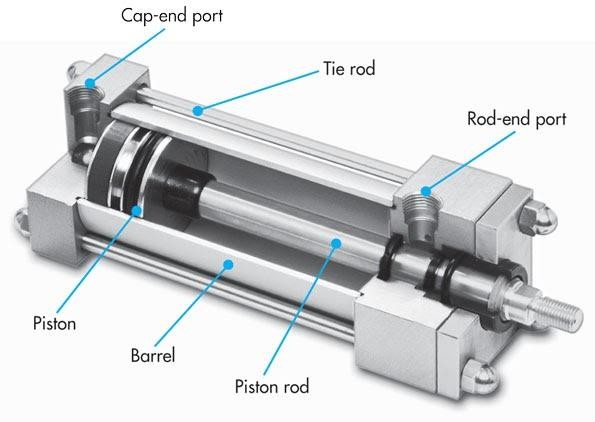
Cụ thể, với van điện từ thường đóng, có nghĩa là ở trạng thái bình thường cửa van sẽ đóng, dòng lưu chất không được chảy qua.Khi điện được cấp vào van, lõi dây đồng bên trong coil điện sẽ sinh ra từ trường (theo nguyên lý nam châm điện). Lực từ trường này sẽ truyền qua trục, tác động đến hút piston đang làm nhiệm vụ đóng cửa van. Van mở cửa và dòng lưu chất chảy qua. Khi ngắt điện, cuộn coil không sinh ra từ trường. Lực từ sẽ mất và lực của lò xo sẽ làm pittong đóng cửa van. Van trở về trạng thái ban đầu.

Và ngược lại với loại van điện từ thường mở. Ở trạng thái bình thường, van mở để dòng lưu chất đi qua. Khi cung cấp điện, từ trường được sinh ra sẽ tác động đến pittong làm cửa van đóng lại. Dòng chất ngưng chảy qua. Khi ngắt điện, từ trường mất và van trở lại trạng thái ban đầu.

Một số hệ thống hiện nay đã kết hợp thành công van solenoid với các thiết bị hẹn giờ, cảm biến để van có thể tự động làm việc theo cài đặt sẵn, đảm bảo thời gian và công suất ổn định.

* Ưu điểm:
* Thời gian chuyển đổi trạng thái làm việc của van hay nói cách khác là đóng mở rất nhanh. Theo như quan sát và thì thời gian trung bình khoảng từ 1-3s.
* Khi vận hành êm ái, ổn định, ít tiếng ồn, không rung và an toàn với con người.
* Thiết kế nhỏ gọn, vật liệu bền, thuận tiện cho việc di chuyển, lắp đặt, thay thế trong mọi không gian, vị trí và môi trường có tính chất khắc nghiệt khác nhau như: nước nóng, nước lạnh, hơi, hóa chất…
* Van có chế độ đóng/mở tự động kết hợp với các thiết bị hẹn giờ, cảm biến sẽ ít tiêu tốn điện năng, giảm việc sử dụng nhân công điều khiển trực tiếp từ đó tiết kiệm các chi phí.
* Nhược điểm:
  + Do sử dụng điện năng để hoạt động nên sẽ bị ảnh hưởng khi mất điện hoặc điện gặp sự cố, chập chờn khiến van không đóng mở kịp thời theo đúng yêu cầu.
  + Vào thời điểm đầu hành trình đối với van điện từ thường mở và cuối hành trình đối với van thường đóng, van sẽ có sự chênh lệch áp suất của lá van. Chính vì thế mà lực cần để tác động lên van sẽ lớn nên động cơ sẽ bị quá tải ở thời điểm này.
  + Van không thích hợp dùng cho nước bẩn hoặc dòng lưu chất nhiều cặn, tạp chất. Điều này sẽ ảnh hưởng đến khả năng đóng kín và mở cửa của van, làm tắc nghẽn hoặc kẹt van khiến năng suất hoạt động giảm đi đáng kể.

3.4 Xy lanh khí nén



*Hình 3.4: Cấu tạo xi lanh khí nén*

* Định nghĩa:

Xy lanh khí nén là một thiết bị cơ học không thể thiếu trong những loại máy nó hoạt động được là do dựa trên một nguyên tắc sử dụng khí nén thông thường là không khí, nguyên tắc hoạt động của nó sẽ truyền một lực có sẵn và làm chuyển một nguồn năng lượng tích tụ từ bên trong thành luồng khí nén động lực.

* Nguyên tắc hoạt động:

Sử dụng nguồn lực từ bên trong vì vậy mà không hề tiêu tốn hay chịu bất cứ tác động nào từ dòng năng lượng bên ngoài. Khi có lực nén khí cấp vào xy lanh khí nén với một áp suất nhất định sẽ làm nở không khí và tác động một lực nhất định đẩy piston di chuyển một cách từ từ theo hướng mình mong muốn.

Xy lanh khí nén hoạt động một cách tự động thông qua cấp khí nén được cài đặt sẵn bằng một con van điện từ khí nén có tác dụng cấp khí cho xy lanh piston có thể đi hết hành trình đòi hỏi lượng khí trong xy lanh phải được thải ra ngoài. Nguồn khí cung cấp cho xy lanh sẽ được nén ở một mức áp suất và nhiệt độ nhất định. Nhờ sự chênh lệch giữa mức áp suất trong xy lanh và áp suất ở môi trường bên ngoài khiến cho không khí bị giãn nở và năng lượng khí nén được chuyển đổi thành động năng. Do đó sự chuyển đổi năng lượng được thực hiện không phụ thuộc vào môi trường bên ngoài. Khi cung cấp nguồn khí cho xy lanh thì không khí sẽ được nén và chiếm không gian trong lòng xy lanh. Lượng khí này chiếm không gian lớn dần lên và khiến cho piston di chuyển làm sinh ra công giúp cho các thiết bị hoạt động.

* Với các thông số kỹ thuật:
* Loại: Khí nén.
* Hoạt động: Tác dụng kép.
* Môi trường hoạt động: Khí nén.
* Áp suất vận hành lớn nhất: 1.0 Mpa.
* Áp suất vận hành nhỏ nhất: 0.05 MPa
* Nhiệt độ của khí: -10°C đến 70°C
* Vật liệu bôi trơn: Không yêu cầu
* Tốc độ piston: 50 đến 750 mm/s
* Giảm chấn: Bằng cao su.
* Động năng cho phép: 0.4J.

3.5 Cảm biến tiệm cận

Cảm biến tiệm cận E3F-DS30C4 NPN là dòng cảm biến tiệm cận phát hiện vật cản tới gần bằng tia hồng ngoại cho độ phản hồi nhanh, có độ chính xác cao, dễ sử dụng.

* **Thông số kỹ thuật**
* Loại: cảm biến tiệm cận hồng ngoại
* Cảm biến vật cản hồng ngoại E3F-DS30C4 NPN 6-36V
* Dải điện áp làm việc: DC 6-36VDC. *Hình 3.5: Cảm biến tiệm cận*
* Khoảng cách phát hiện: 10-30cm có thể điều chỉnh
* Đầu ra NPN
* Chất liệ sản phẩm: nhựa
* Có led hiển thị ngõ ra màu đỏ
* Kích thước: 18 x 45 (mm)
* **Sơ đồ chân:**
* Màu nâu: VCC, nguồn dương 6 – 36VDC.
* Màu xanh dương: GND, nguồn âm 0VDC
* Màu đen: Chân tín hiệu ngõ ra cực thu hở NPN, cần phải có trở kéo để tạo thành mức cao.

|  |  |
| --- | --- |
| Input | 110 – 220VAC |
| Output | 24V 5A |
| Công suất | 120W |
| Trọng lượng | 400g |
| Kích thước | 19,8x9,8,4 cm |

3.6 Nguồn tổ ong 24V 5A

*Hình 3.6: Nguồn tổ ong 24V 5 Bảng 3.6: Thông số kỹ thuật*

Nguồn tổ ong 24V 5A hay còn gọi là nguồn một chiều được thiết kế để chuyển đổi diện áp từ nguồn xoay chiều 220V thành nguồn một chiều 12VDC để cung cấp cho các thiết bị hoạt động. Nguồn tổ ong được sử dụng rộng rãi trong các thiết bị điện và dân dụng.

3.7 Relay trung gian

*Hình 3.7: Relay 5v modul 4*

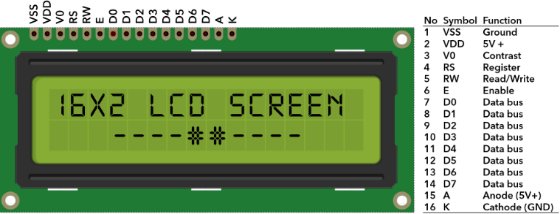
* Giới thiệu về relay trung gian

Module 4 relay 5V với Opto cách ly gồm 4 rơ le hoạt động tại điện áp 5VDC, chịu được hiệu điện thế lên đến 250VAC 10A. Module 4 relay 5V với Opto cách ly được thiết kế chắc chắn, khả năng cách điện tốt. Trên module đã có sẵn mạch kích relay sử dụng transistor và IC cách ly quang giúp cách ly hoàn toàn mạch điều khiển (vi điều khiển) với rơ le bảo đảm vi điều khiển hoạt động ổn định. Có sẵn header rất tiện dụng khi kết nối với vi điều khiển.

Module 4 relay 5V với opto cách ly sử dụng kích H/L Module 4 relay 5V với Opto cách ly sử dụng chân kích mức Thấp (0V), khi có tín hiệu 0V vào chân IN thì relay sẽ nhảy qua thường hở của relay ứng dụng với relay module khá nhiều bao gồm cả điện DC hay AC.

* Thông số kỹ thuật
* Điện áp hoạt động: 5VDC.
* Dòng tiêu thụ: 200mA/1Relay.
* Tín hiệu kích: High (5V) hoặc Low (0V) chọn bằng Jumper.
* Nguồn nuôi: 5VDC.
* Tiếp điểm đóng ngắt max: 250VAC-10A hoặc 30VDC-10A .
* Kích thước : 72mm \* 55mm \* 19mm.

3.8 Màn hình hiển thị LCD 16x2

LCD có chức năng nhận hoặc truyền tín hiệu từ vi điều khiển và hiển thị trạng thái ra màn

*Hình 3.8: LCD 16x2 và các chân*

|  |  |  |
| --- | --- | --- |
| **Chân** | **Ký hiệu** | **Chức năng** |
| 1 | Vss (GND) | 0 V |
| 2 | Vdd (VCC) | 5 VDC |
| 3 | Vee | Điều chỉnh độ tương phản, nguồn từ 0-5VDC |
| 4 | RS | Lựa chọn thanh ghi (Thanh ghi lệnh và thanh ghi dữ liệu)  RS=0 : Thanh ghi lệnh (khi ghi) tức là data trên chân D0-D7 là lệnh  RS=1 : Thanh ghi dữ liệu (Khi ghi và đọc): tức là data trên chân D0-D7 là dữ liệu |
| 5 | R/W | R/W=0 : Ghi dữ liệu (vi điều khiển ->LCD)  R/W=1 : Đọc dữ liệu (LCD-> Vi điều khiển) |
| 6 | E | E=0 : Vô hiệu hóa LCD  E=1 : LCD hoạt động  E chuyển từ 1->0: Bắt đầu ghi/ đọc LCD |
| 7 | D0 | Chân dữ liệu bit 0 |
| 8 | D1 | Chân dữ liệu bit 1 |
| 9 | D2 | Chân dữ liệu bit 2 |
| 10 | D3 | Chân dữ liệu bit 3 |
| 11 | D4 | Chân dữ liệu bit 4 |
| 12 | D5 | Chân dữ liệu bit 5 |
| 13 | D6 | Chân dữ liệu bit 6 |
| 14 | D7 | Chân dữ liệu bit 7 |
| 16 | A | Cực dương led nền LCD |
| 17 | K | Cực âm led nền LCD |

*Bảng 3.8 : Chức năng của các chân LCD*

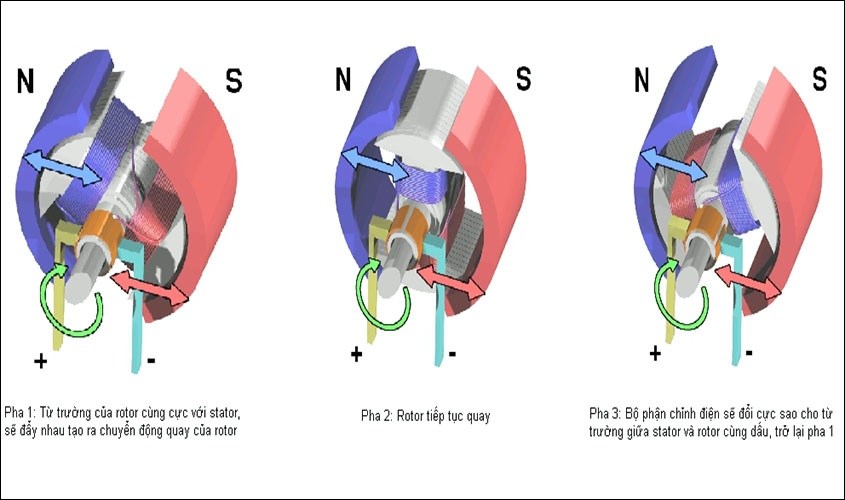
3.9 Động cơ DC



*Hình 2.1.4a: Động cơ DC*

Động cơ DC (Direct Current Motors) là động cơ điều khiển bằng dòng điện 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. Đầu dây ra của động cơ thường gồm 2 dây: dây nguồn-VCC và dây tiếp đất-GND.

* **Cấu tạo:**
* Gồm có 3 phần chính: stator (phần cảm), rotor (phần ứng) và phần cổ góp-chỉnh lưu.
* Stator của động cơ điện một chiều thường là một hay nhiều cặp nam châm vĩnh cửu, hay nam châm điện.
* Rotor có các cuộn dây cuốn và được nối với nguồn điện một chiều.
* Bộ phận chỉnh lưu có nhiệm vụ đổi chiều dòng điện trong khi chuyển động quay của rotor là liên tục. Thông thường bộ phận này thường có một bộ phận cổ góp và một bộ chổi than tiếp xúc với cổ góp.
* **Nguyên lý hoạt động**:
* Động cơ DC lấy năng lượng điện thông qua dòng điện trực tiếp và chuyển đổi năng lượng điện này thành vòng quay cơ học. Khi động cơ DC lấy năng lượng điện sẽ tạo ra một từ trường trong Stator. Từ trường này sẽ thu hút và đẩy lùi nam châm trên Roto, làm cho Roto quay.
* Nguyên lý hoạt động này chúng ta còn được biết với cái tên “Quy tắc bàn tay trái” được dạy trong môn Vật Lý. Tuy nhiên, để cỗ máy có thể hoạt động thì Roto cần quay liên tục không ngừng nghỉ, người ta sẽ gắn bộ chuyển đổi vào bàn chải đã kết nối với dòng điện để cung cấp nguồn điện cho cuộn dây động cơ.
* Ngoài ra, tốc độ quay của mỗi loại động cơ DC sẽ thay đổi và khác nhau theo từng chu kỳ thời gian, có thể là vòng/phút hoặc nghìn vòng/phút tùy thuộc vào việc ứng dụng đối với từng thiết bị khác nhau.
* Có thể nói động cơ DC là loại động cơ đơn giản và phổ biến nhất, được áp dụng trong nhiều ngành công nghiệp sản xuất và chế tạo như trong thiết bị gia dụng, lắp ráp quạt trần, dao cạo thông minh, cửa sổ điện ô tô,...



*Hình 3.9a: Sơ đồ nguyên lý hoạt động của động cơ DC*

Trong hệ thống này sử dụng động cơ encoder GA25:



*Hình 3.9b: Động cơ encoder GA25*

* **Thông số kỹ thuật:**
* Điện áp cấp cho động cơ hoạt động: 3 – 12VDC.
* Điện áp cấp cho Encoder hoạt động: 3.3VDC.
* Đĩa Encoder 11 xung, hai kênh A-B.
* Đường kính động cơ: 25mm.
* Đường kính trục: 4mm.

3.10 Tính toán lựa chọn xy lanh:

Khi tính toán lực cần chú ý đến chiều chuyển động của cán pittông và lực tác động lên cần pittông.

Lực tác động khi cần pittông đi ra: FA = A1.pe2. η

* A1: Diện tích mặt đáy pittông(m2)
* D: Đường kính mặt đáy pitông(m).
* Pe2 :áp suất khí nén trong xilanh(bar).
* η: Hiệu suất xilanh, thông thường η = 0,8

Lực tác động khi cần pittông đi vào: FA = A2.pe2. η

* A2 Diện tích vòng găng pittông:
* d: Đường kính cán pittông

**Kết Luận:**

Do khối lượng của vật thể không quá nặng (~ 100g) nên nhóm quyết định chọn loại xy lanh có thông số kỹ thuật như sau:

* Chất liệu: Hợp kim & Kim Loại.
* Đường kính xy lanh: 10cm; Hành trình: 100mm
* Điều kiện làm việc: Không khí
* Áp Lực tối đa :1.0 MPa (8.5 kg/cm2)

**3.12 Tính chọn cơ cấu chấp hành cho băng tải**

* **Thông số đầu vào:**
* Năng suất: 15sp/ph
* Drl =0.03m
* dph  = 0.03m
* Hệ số ma sát trượt: μ=0.3
* Hiệu suất: η=80%

Dựa vào năng suất và đường kính sản phẩm, ta tính được:

V= 15x0,0.03 =0.45 (m/ph)



Xét loại động cơ xoay chiều 50Hz (220V) hãng Oriental Motor

i=

* + Chọn i= 250
* **Tính momen xoắn cần thiết:**

Khối lượng của tải m=0.1+1,2=1.3(kg)

F= m.g.µ = 1.3x9.8x0.3 =3.822(N)



Chọn hệ số an toàn SF= 2



Dựa vào Catalog của nhà sản suất chọn sơ bộ dòng động cơ: **2IK6GC-250**

Momen khởi động:

Momen quán tính quy đổi trục động cơ:



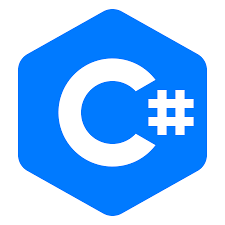
Kiểm nghiệm động cơ: Động cơ thỏa mãn các tiêu chí momen khởi động và momen quán tính.

* Kết luận: Ta chọn sử dụng dòng động cơ ký hiệu **2IK6GC-250.**

CHƯƠNG 4: GIỚI THIỆU VỀ PHẦN MỀM VÀ LẬP TRÌNH HỆ THỐNG

4.1 Ngôn ngữ lập trình C# là gì ?

4.1.1 Giới thiệu về ngôn ngữ C#

* C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000, trong đó người dẫn đầu là Anders Hejlsberg và Scott Wiltamuth.
* C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và nó được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.
* C# được thiết kế cho Common Language Infrastructure (CLI), mà gồm Executable Code và Runtime Environment, cho phép chúng ta sử dụng các ngôn ngữ high-level đa dạng trên các nền tảng và cấu trúc máy tính khác nhau.
* C# với sự hỗ trợ mạnh mẽ của .NET Framework giúp cho việc tạo một ứng dụng Windows Forms hay WPF (Windows Presentation Foundation), . . . trở nên rất dễ dàng.
* *Hình 4.1b: Động cơ encoder GA25*

4.1.2 Đặc trưng của C#

C# là ngôn ngữ lập trình chuyên nghiệp được sử dụng rộng rãi bởi vì:

* **C# là ngôn ngữ đơn giản**

Như ta đã biết thì ngôn ngữ C# dựng trên nền tảng C++ và Java nên ngôn ngữ C# khá đơn giản. Nếu chúng ta thân thiện với C và C++ hoậc thậm chí là Java, chúng ta sẽ thấy C# khá giống về diện mạo, cú pháp, biểu thức, toán tử và những chức năng khác được lấy trực tiếp từ ngôn ngữ C và C++, nhưng nó đã được cải tiến để làm cho ngôn ngữ đơn giản hơn. Một vài trong các sự cải tiến là loại bỏ các dư thừa, hay là thêm vào những cú pháp thay đổi.

* **C# là ngôn ngữ hiện đại**

Một vài khái niệm khá mới mẻ khá mơ hồ với các bạn vừa mới học lập trình, như xử lý ngoại lệ, những kiểu dữ liệu mở rộng, bảo mật mã nguồn..v..v... Đây là những đặc tính được cho là của một ngôn ngữ hiện đại cần có. Và C# chứa tất cả các đặt tính ta vừa nêu trên. Các bạn sẽ dần tìm hiểu được các đặt tính trên qua các bài học trong series này.

* **C# là một ngôn ngữ lập trình thuần hướng đối tượng**

Lập trình hướng đối tượng(tiếng Anh: Object-oriented programming, viết tắt: OOP) là một phương pháp lập trình có 4 tính chất. Đó là tính trừu tượng (*abstraction*), tính đóng gói (encapsulation), tính đa hình (polymorphism) và tính kế thừa (inheritance). C# hỗ trợ cho chúng ta tất cả những đặc tính trên. Và để hiểu rõ hơn thì chúng ta sẽ có một chương trình bày về phần này.

* **C# là một ngôn ngữ ít từ khóa**

C# được mô tả là ngôn ngữ sử dụng giới hạn những từ khóa (gồm khoảng 80 từ khóa và mười mấy kiểu dữ liệu xây dựng sẵn). Nếu bạn nghĩ rằng ngôn ngữ có càng nhiều từ khóa thì sẽ càng mạnh mẽ hơn. Điều này không phải sự thật, lấy ví dụ ngôn ngữ C# làm điển hình nhé. Nếu bạn học sâu về C# bạn sẽ thấy rằng ngôn ngữ này có thể được sử dụng để làm bất cứ nhiệm vụ nào.

Ngoài những đặc điểm trên thì còn một số ưu điểm nổi bật của C#:

* C# có cấu trúc khá gần gũi với các ngôn ngữ lập trình truyền thống, nên cũng khá dể dàng tiếp cận và học nhanh với C#.
* C# có thể biên dịch trên nhiều nền tảng máy tính khác nhau.
* C# được xây dựng trên nền tảng của C++ và Java nên nó được thừa hưởng những ưu điểm của ngôn ngữ đó.
* C# là một phần của .NET Framework nên được sự chống lưng khá lớn đến từ bộ phận này.
* C# có IDE Visual Studio cùng nhiều plug-in vô cùng mạnh mẽ.

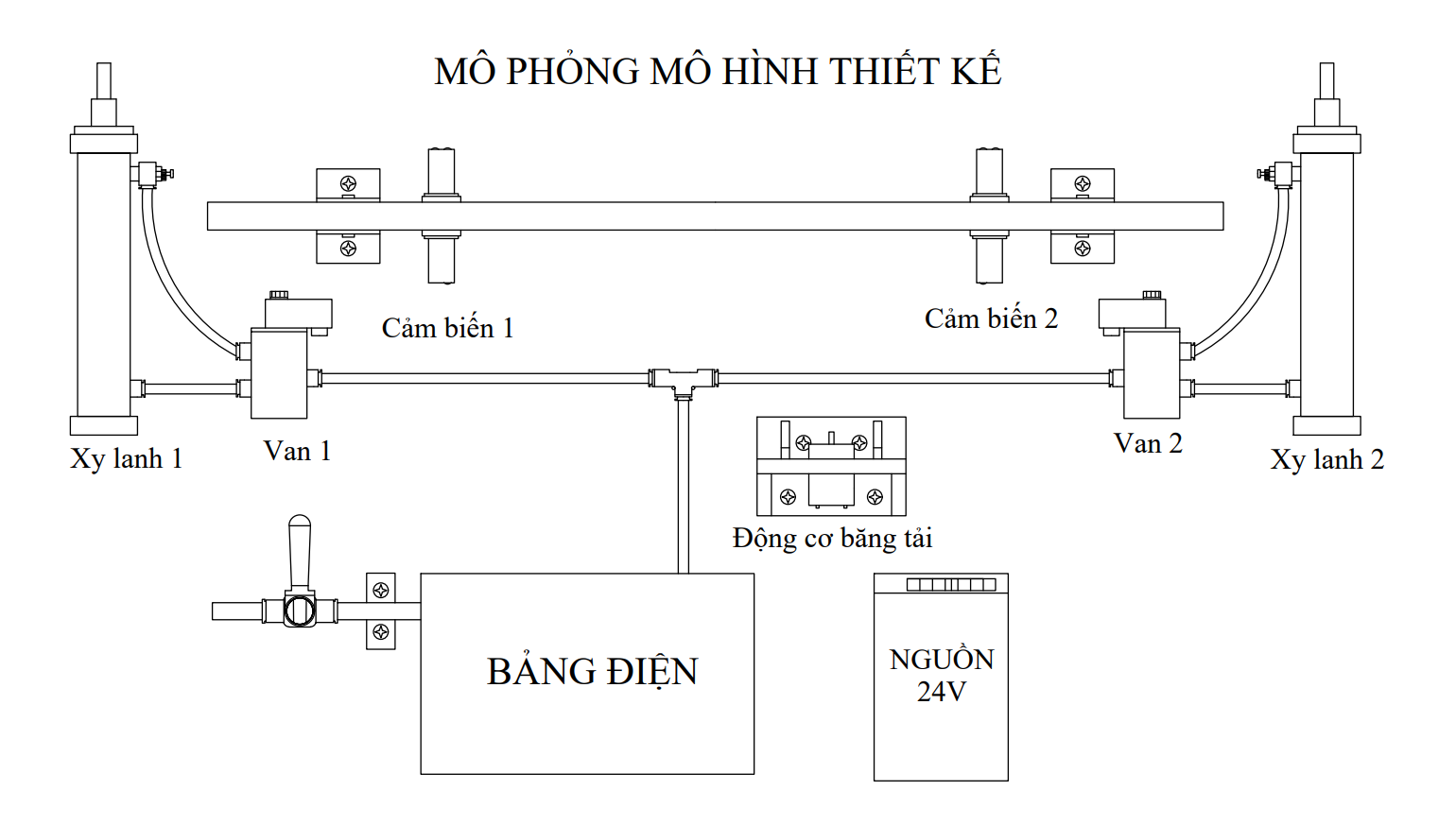
**4.1.3 Đặc trưng của C#**

Aforge.net là một nền tảng C# thiết kế cho các nhà phát triển hoặc nghiên cứu trong lĩnh vực thị giác máy và trí tuệ nhân tạo- xử lý ảnh, mạng thần kinh nhân tạo, các thuật toán di truyền, logic mờ, học máy, robot.

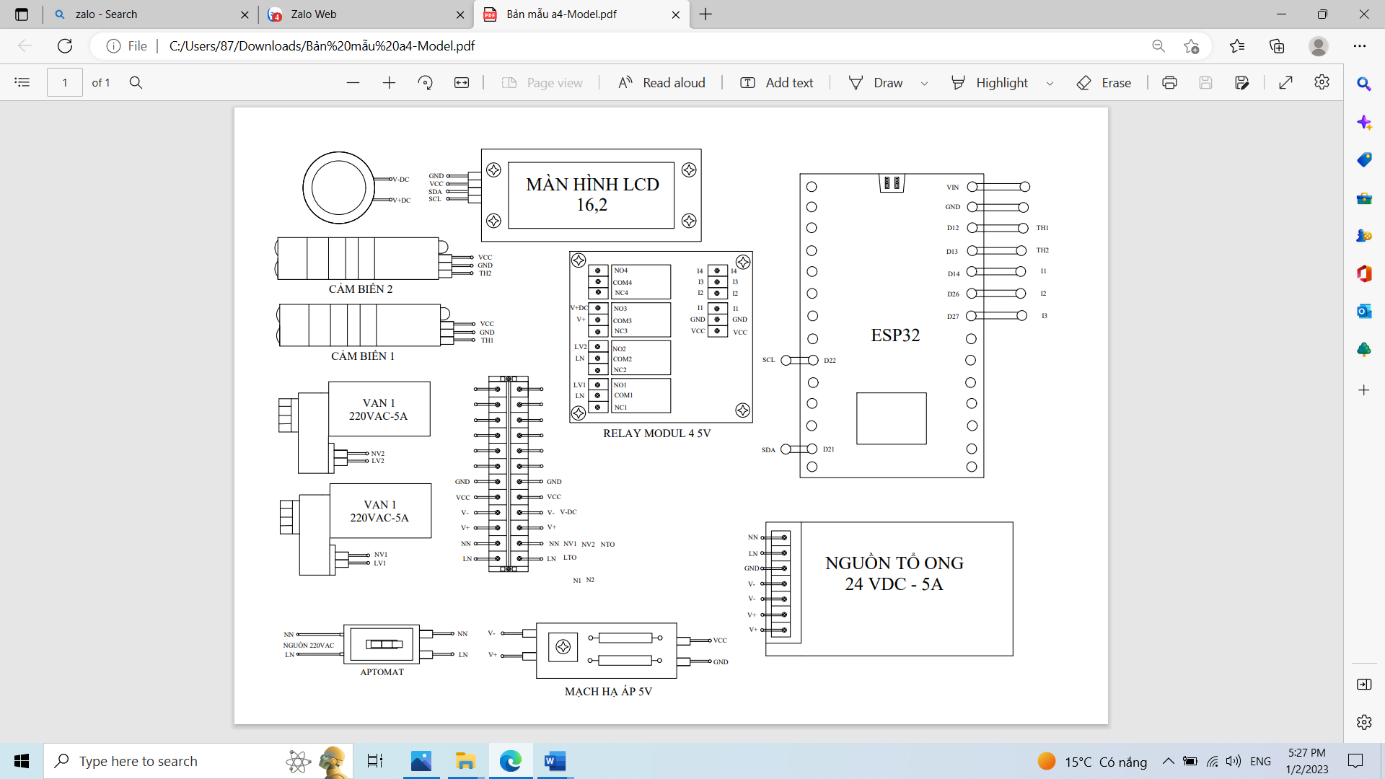
Nền tảng này là tập hợp các thư viện và các ứng dụng mẫu có các tính năng sau:

* Thư viện Aforge: Thư viện Aforge
* Thư viện AForge.Imaging: Thư viện Aforge xử lý hình ảnh
* Thư viện AForge.Video: Thư viện Aforge video camera
* Thư viện AForge.Video.DirectShow: Thư viện Aforge hiển thị video
* Thư viện AForge.Imaging.Filters: Thư viện Aforge thư viện lọc nhiễu
* Thư viện AForge.Math.Geometry: Thư viện Aforge thư viện xử lý thuật toán

4.2 Phác thảo mô hình điều khiển cơ cấu chấp hành



*Hình 4.2: Phác thảo ý tưởng mô hình thiết kế*

4.3 Sơ đồ đấu nối

*Sơ đồ 4.3: Sơ đồ mach điện*

4.3 Sơ đồ khối của hệ thống

Camera

Laptop

ESP32

Van

*Sơ đồ 4.3: Sơ đồ khối của hệ thống*

Chức năng của từng khối:

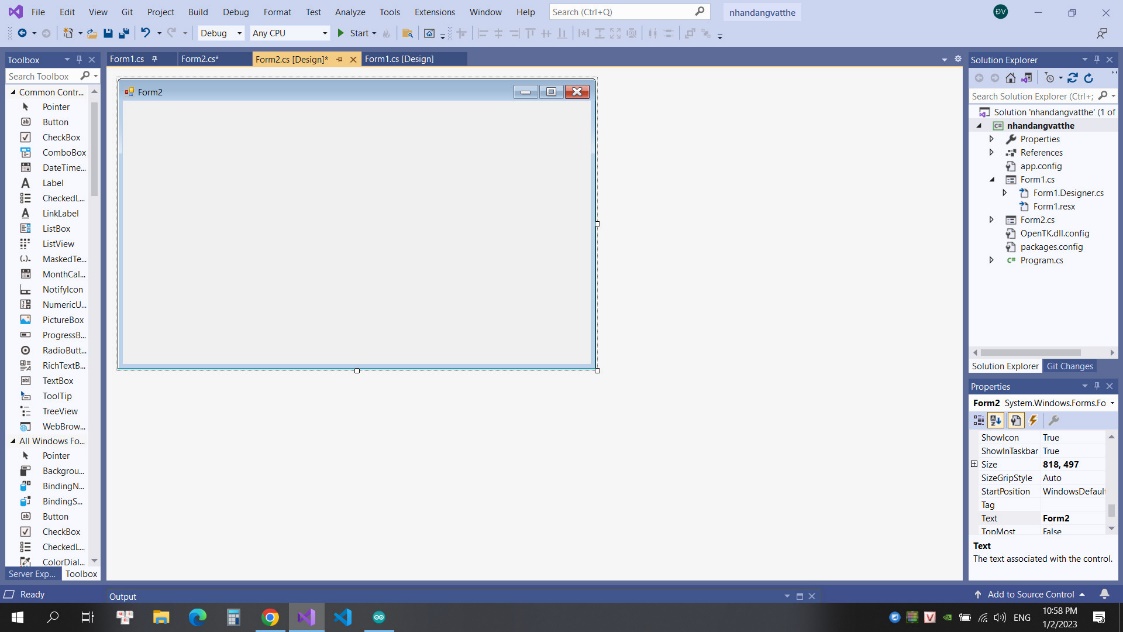
* Khối camera:
* Có chức năng thu nhận tín hiệu hình ảnh từ thực tế chuyển về tín hiệu điện và gửi dữ liệu cho laptop.
* Khối xử lý:
* Laptop: có chức năng nhận tín hiệu hình ảnh từ camera sau đó tiến hành xử lý hình ảnh và phân tích nhận dạng rồi trả kết quả cho ESP32.
* ESP32: có chức năng nhận tín hiệu từ laptop sau đó ra lệnh cho khối động cơ hoạt động rồi nhận dữ liệu trả về từ laptop tiến hành phân tích xử lý, điều khiển hoạt động của khối động cơ.
* Khối cơ cấu chấp hành:
* Piston có khả năng thực hiện mà không cần tới sự can thiệp trung gian, để có thể thực hiện đầy đủ với các điểm dừng trung gian, đồng thời được điều chỉnh để có thể kiểm soát được lượng mở rộng hoặc là rút lại của cần piston trong quá trình hoạt động của nó.

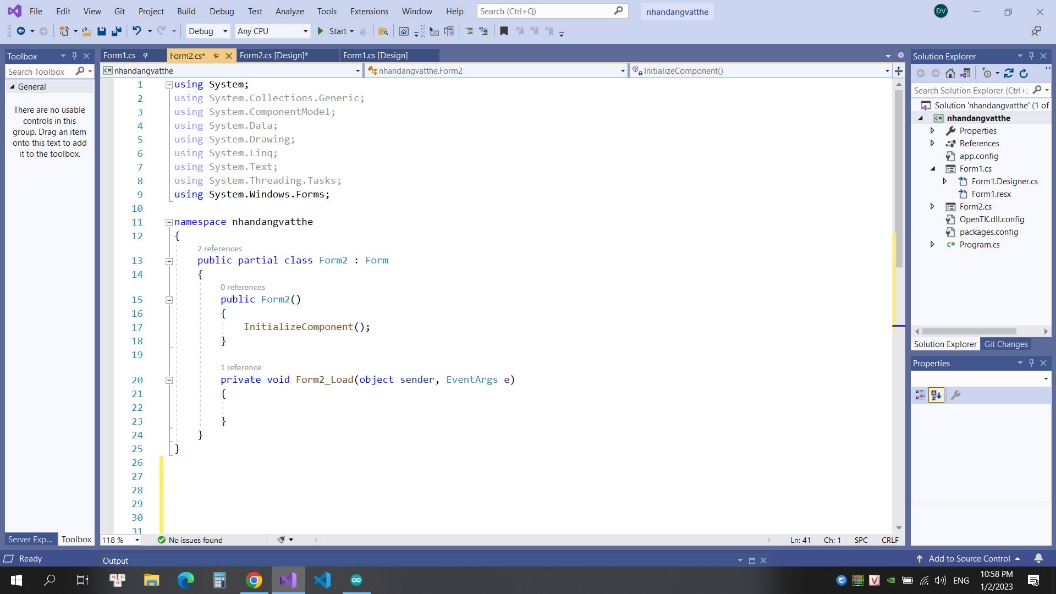
4.4 Lập trình và phần mềm Microsoft Visual Studio

**4.4.1 Giới thiệu về phần mềm Microsoft Visual Studio**

  Microsoft Visual Studio còn được gọi là " Trình soạn thảo mã nhiều người sử dụng nhất thế giới ", được dùng để lập trình C++ và C# là chính. Nó được sử dụng để phát triển chương trình máy tính cho Micrsoft Windows, cũng như các trang web, các ứng dụng web và các dịch vụ web. Visual Studio sử dụng nền tảng phát triển phần mềm của Microsoft như Windows API, Windows Forms, Windows Presentation Foundation, Windows Store và Microsoft Silverlight. Nó có thể sản xuất cả hai ngôn ngữ máy và mã số quản lý.

Visual Studio hỗ trợ nhiều ngôn ngữ lập trình khác nhau và cho phép trình biên tập mã và gỡ lỗi để hỗ trợ (mức độ khác nhau) hầu như mọi ngôn ngữ lập trình. Các ngôn ngữ tích hợp gồm có C,[4] C++ và C++/CLI (thông qua Visual C++), VB.NET (thông qua Visual Basic.NET), C# (thông qua Visual C#) và F# (như của Visual Studio 2010[5]). Hỗ trợ cho các ngôn ngữ khác như J++/J#, Python và Ruby thông qua dịch vụ cài đặt riêng rẽ. Nó cũng hỗ trợ XML/XSLT, HTML/XHTML, JavaScript và CSS.

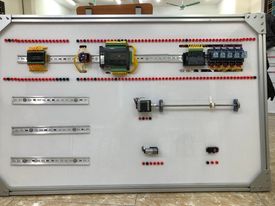
 **4.4.2 Lập trình trên Microsoft Visual Studio 2019**

*Hình 4.4.1.a: Thiết kế giao diện điều khiển*

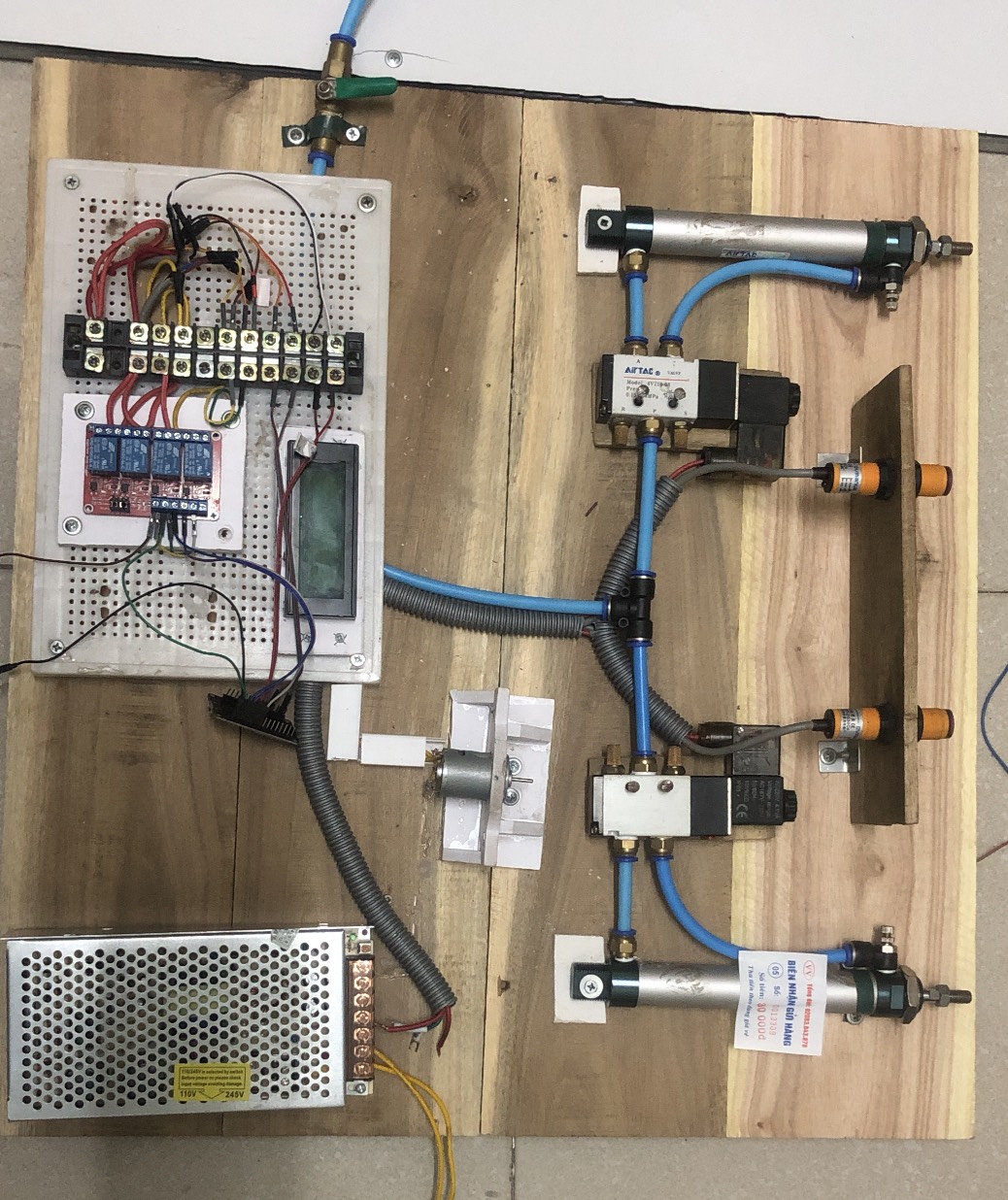
*Hình 4.4.1.b: Giao diện lập trình*

4.5 KẾT QUẢ ĐẠT ĐƯỢC

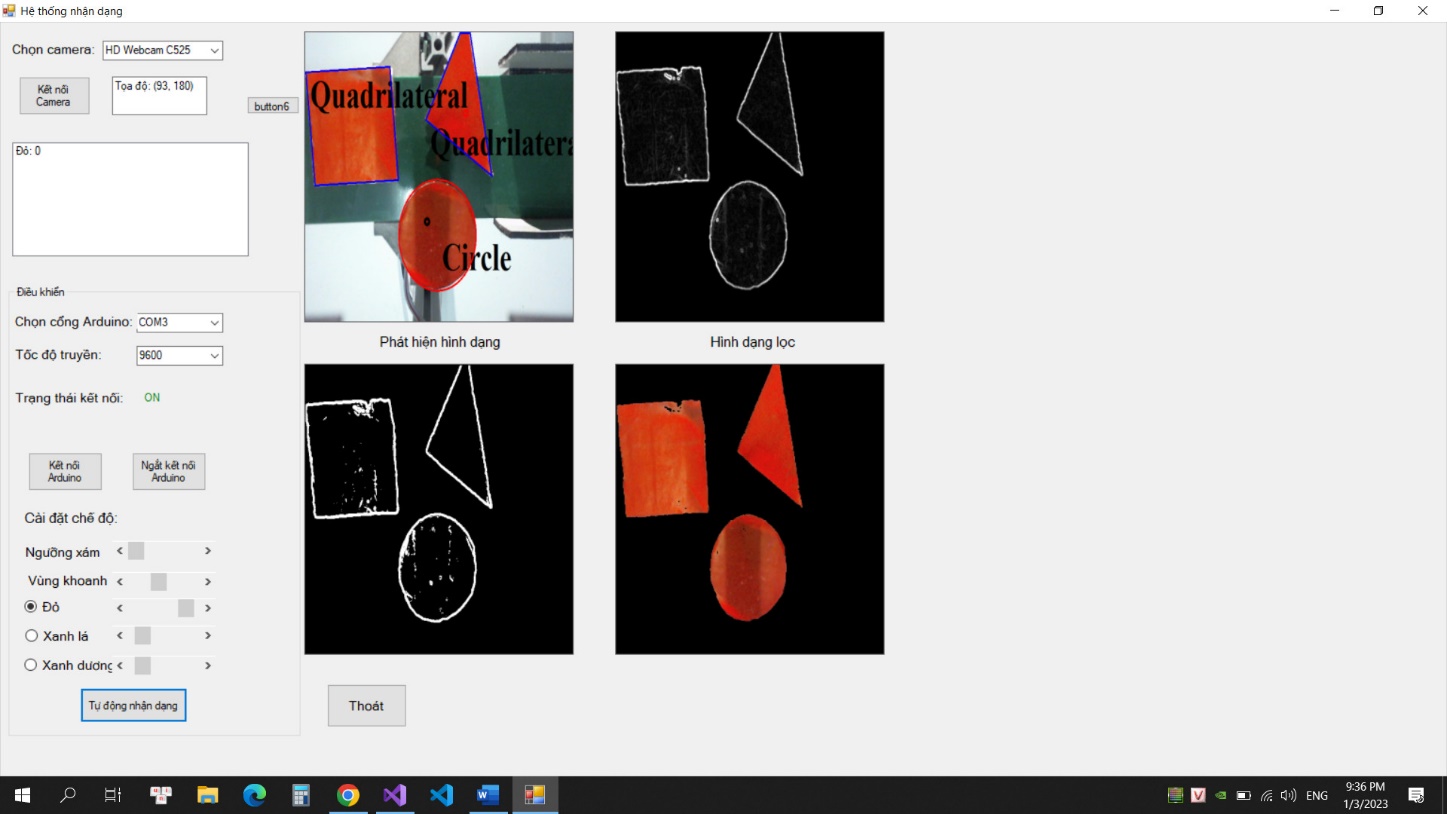
* Xây dựng mô hình thực tế

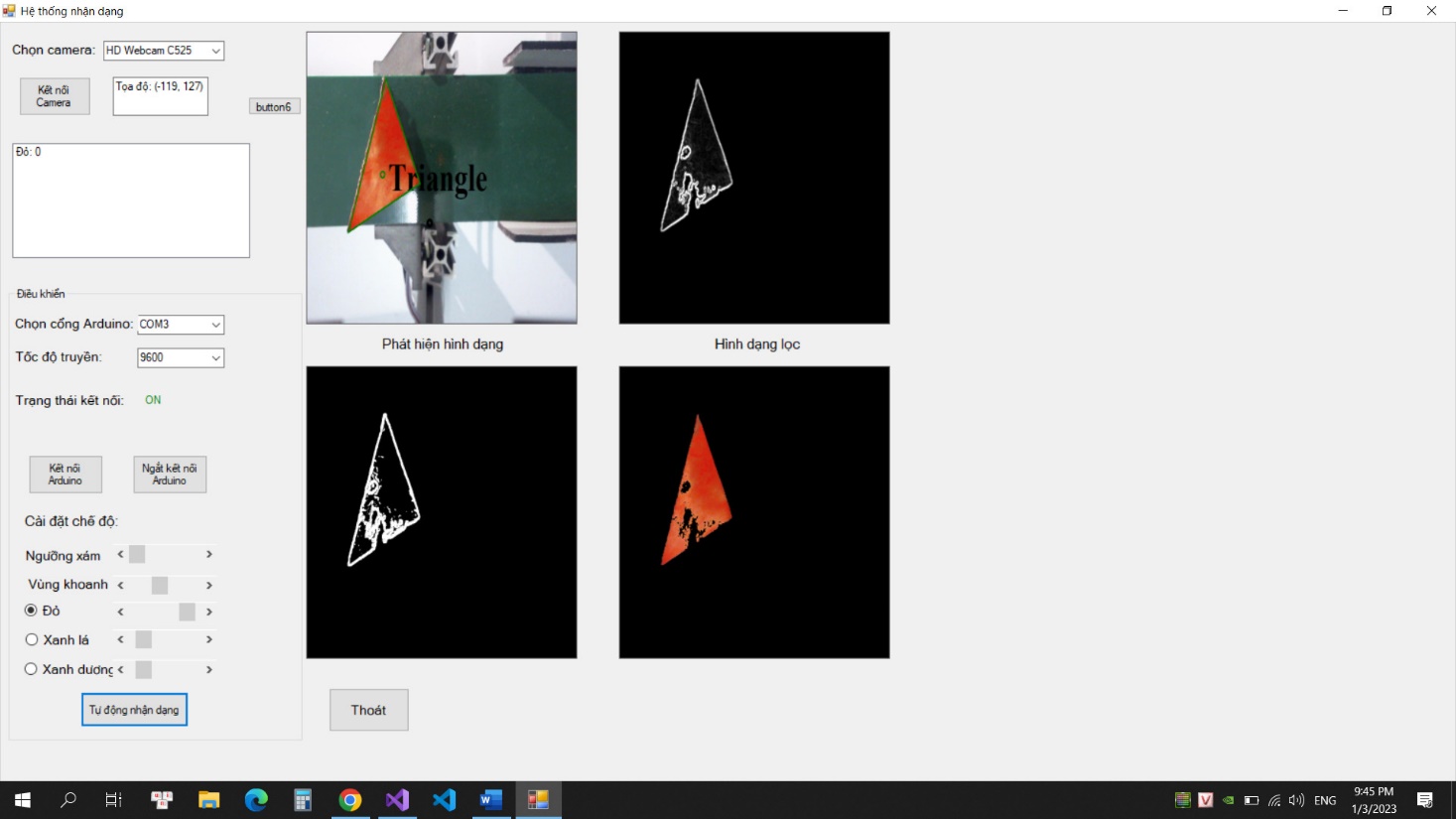


*Hình 4.5a: Mô hình thực hành bàn thí nghiệm*

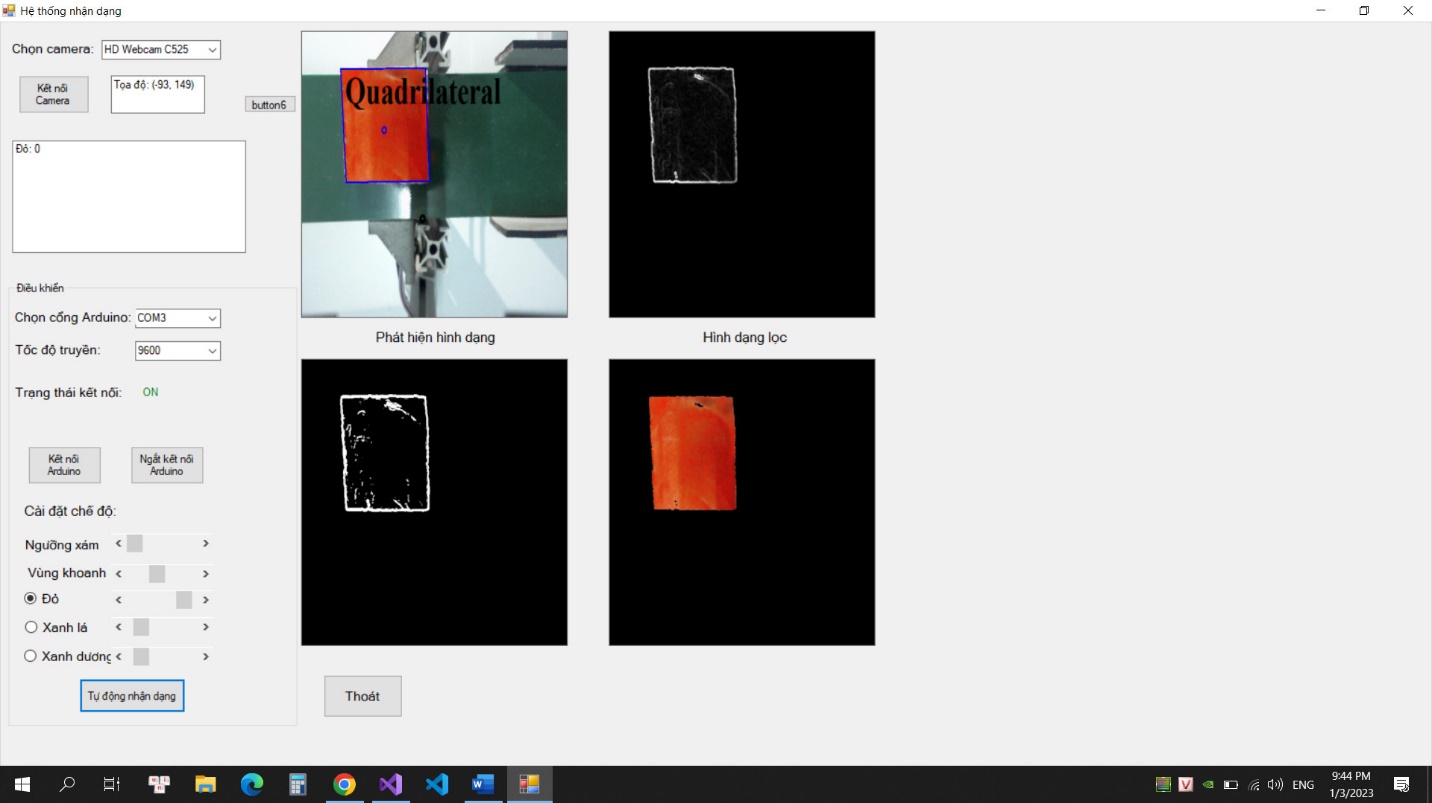


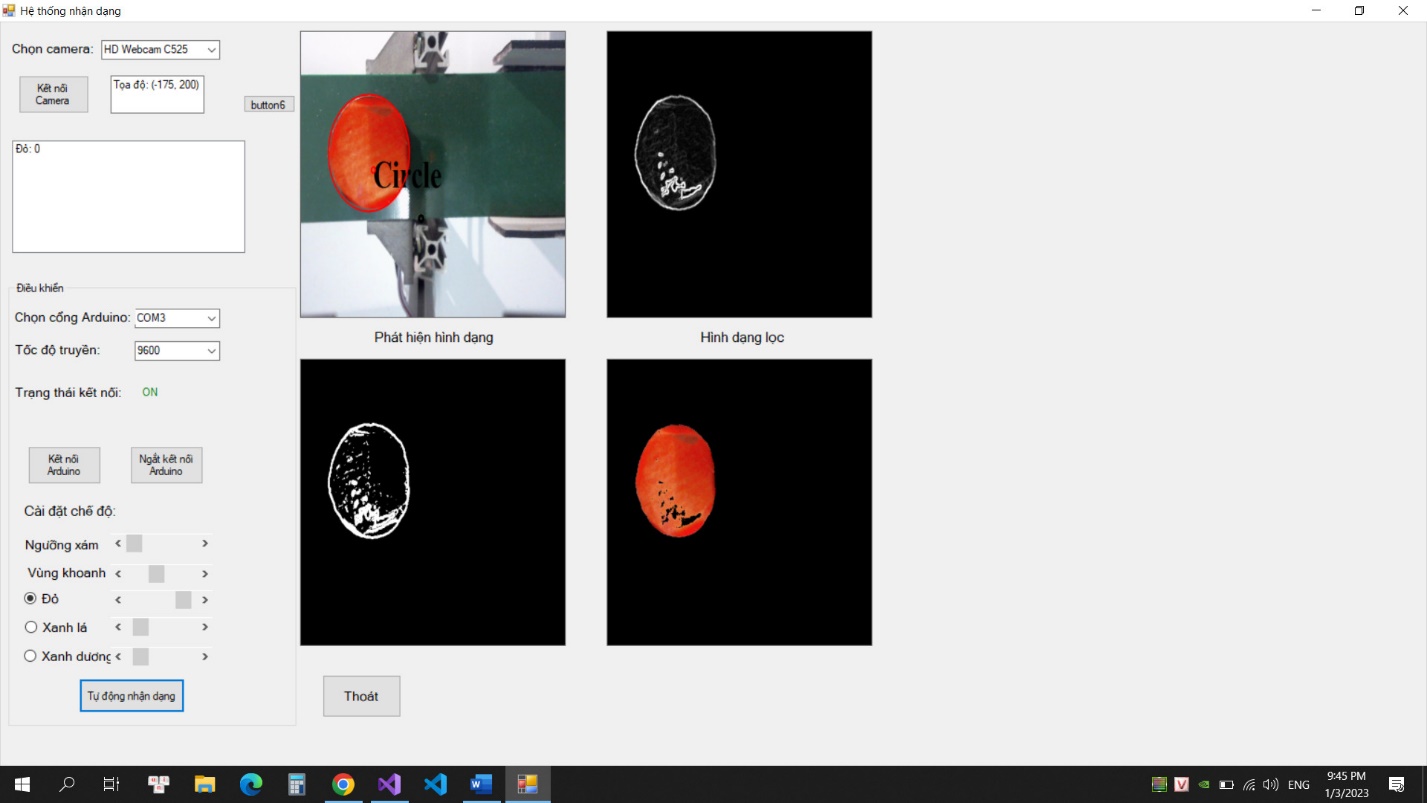
*Hình 4.5b: Mô hình điều khiển cơ cấu chấp hành*

* Hệ thống nhận dạng hình dạng

*******Hình 4.5a: Hệ thống nhận dạng các khối mầu cơ bản*

*Hình 4.5b: Hệ thống nhận dạng sản phẩm hình tam giác*

**

*Hình 4.5c: Hệ thống nhận dạng sản phẩm hình vuông*

*Hình 4.5d: Hệ thống nhận dạng sản phẩm hình tròn*

* **Nhận xét:**
* Kết quả đạt được là khá tốt
* -Hệ thống nhận dạng và phân loại được sản phẩm theo hình dạng bằng ngôn ngữ C#
* -Thiết kế thành công mô hình ứng dụng xử lý ảnh trong bài toán phân loại sản phẩm.

**4.6 Sơ đồ giải thuật bài toán xử lý và phân loại sản phẩm**

4.7 Code lập trình:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Threading;

using System.Drawing.Imaging;

using AForge;

using AForge.Imaging;

using AForge.Video;

using AForge.Video.DirectShow;

using AForge.Imaging.Filters;

using AForge.Math.Geometry;

using System.IO.Ports;

namespace nhandangvatthe

{

public partial class Form1 : Form

{

FilterInfoCollection \_device;

VideoCaptureDevice \_CaptureDevice;

Bitmap anhvien, anhnhiphan, anhxam, lammoanh, locmau;

EuclideanColorFiltering \_colorFilter = new EuclideanColorFiltering();

System.Drawing.Font \_font = new System.Drawing.Font("Times New Roman", 48, FontStyle.Bold);

System.Drawing.SolidBrush \_brush = new System.Drawing.SolidBrush(System.Drawing.Color.Black);

SobelEdgeDetector \_edgeFilter = new SobelEdgeDetector();

Pen \_PictureboxPen = new Pen(Color.Black, 5);

bool \_blurFlag = false;

int ipenWidth = 3;

int iThreshold = 40, iRadius = 40;

int iColorMode = 1, iRedValue = 220, iGreenValue = 30, iBlueValue = 30;

public event EventHandler kiemtratrangthai;

private void hScrollBar2\_Scroll(object sender, ScrollEventArgs e)

{

iBlueValue = hScrollBar2.Value;

listBox1.Items.Add("Blue: " + iBlueValue.ToString());

ScrollDown();

}

private void hScrollBar1\_Scroll(object sender, ScrollEventArgs e)

{

iGreenValue = hScrollBar1.Value;

listBox1.Items.Add("Xanh lá: " + iGreenValue.ToString());

ScrollDown();

}

private void radioButton1\_CheckedChanged(object sender, EventArgs e)

{

if (radioButton1.Checked == true)

{

iColorMode = 1;

hScrollBar4.Value = 100;

iRadius = hScrollBar4.Value;

hScrollBar3.Value = 220;

hScrollBar2.Value = 30;

hScrollBar1.Value = 30;

listBox1.Items.Add("Đỏ: " + iRedValue.ToString());

ScrollDown();

}

}

private void hScrollBar4\_Scroll(object sender, ScrollEventArgs e)

{

iRadius = hScrollBar4.Value;

listBox1.Items.Add(iRadius.ToString());

ScrollDown();

}

private void hScrollBar5\_Scroll(object sender, ScrollEventArgs e)

{

iThreshold = hScrollBar5.Value;

listBox1.Items.Add(iThreshold.ToString());

ScrollDown();

}

private void radioButton2\_CheckedChanged(object sender, EventArgs e) //chọn màu xanh

{

if (radioButton2.Checked == true)

{

iColorMode = 2;

hScrollBar4.Value = 180;

iRadius = hScrollBar4.Value;

hScrollBar3.Value = 30;

hScrollBar2.Value = 30;

hScrollBar1.Value = 240;

listBox1.Items.Add("Xanh dương: " + iBlueValue.ToString());

ScrollDown();

}

}

private void radioButton3\_CheckedChanged(object sender, EventArgs e) //Chọn màu đỏ

{

if (radioButton3.Checked == true)

{

iColorMode = 3;

hScrollBar4.Value = 180;

iRadius = hScrollBar4.Value;

hScrollBar3.Value = 5;

hScrollBar2.Value = 240;

hScrollBar1.Value = 5;

listBox1.Items.Add("Xanh lá: " + iGreenValue.ToString());

ScrollDown();

}

}

private void hScrollBar3\_Scroll(object sender, ScrollEventArgs e) //Thang lọc màu đỏ

{

iRedValue = hScrollBar3.Value;

listBox1.Items.Add("Đỏ: " + iRedValue.ToString());

ScrollDown();

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e) //Chọn Camera

{

try

{

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void button4\_Click(object sender, EventArgs e) //Tự động

{

try

{

\_blurFlag = true;

}

catch

{

\_blurFlag = false;

}

}

private void button2\_Click(object sender, EventArgs e)

{

//ket noi arduino

try

{

if (serialPort1.IsOpen)

{

MessageBox.Show("Đã tồn tại kết nối");

}

else

{

serialPort1.PortName = comboBox2.Text;

serialPort1.BaudRate = Convert.ToInt32(comboBox3.Text);

serialPort1.Open();

MessageBox.Show("Kết nối thành công");

}

}

catch (Exception ex)

{

MessageBox.Show("Kết nối thất bại\n" + ex.Message);

}

}

private void button3\_Click(object sender, EventArgs e)

{

try

{

if (!serialPort1.IsOpen)

{

MessageBox.Show("Không có kết nối để đóng");

}

else

{

serialPort1.Close();

MessageBox.Show("Ngắt kết nối thành công");

}

}

catch (Exception ex)

{

MessageBox.Show("Ngắt kết nối thất bại\n" + ex.Message);

}

}

private void timer1\_Tick(object sender, EventArgs e)

{

if(this.kiemtratrangthai == null)

{

this.kiemtratrangthai += capnhattrangthai;

}

this.kiemtratrangthai(this, EventArgs.Empty);

}

private void capnhattrangthai(object sender, EventArgs e)

{

if(serialPort1.IsOpen)

{

label12.Text = "ON";

label12.ForeColor = Color.Green;

}

else

{

label12.Text = "OFF";

label12.ForeColor = Color.Red;

}

}

private void pictureBox2\_Click(object sender, EventArgs e)

{

}

private void button5\_Click(object sender, EventArgs e)

{

Application.Exit();

\_CaptureDevice.Stop();

}

private void button6\_Click(object sender, EventArgs e)

{

try

{

serialPort1.Write("1");

}

catch

{

MessageBox.Show("Loi dieu khien");

}

}

private void button7\_Click(object sender, EventArgs e)

{

try

{

serialPort1.Write("2");

}

catch

{

MessageBox.Show("Loi dieu khien");

}

}

private void button1\_Click(object sender, EventArgs e)

{

\_CaptureDevice = new VideoCaptureDevice(\_device[comboBox1.SelectedIndex].MonikerString);

\_CaptureDevice.NewFrame += new NewFrameEventHandler(get\_Frame);

\_CaptureDevice.Start();

}

public Form1()

{

InitializeComponent();

CheckForIllegalCrossThreadCalls = false;

}

private void Form1\_Load(object sender, EventArgs e)

{

try

{

\_device = new FilterInfoCollection(FilterCategory.VideoInputDevice);

foreach (FilterInfo device in \_device)

comboBox1.Items.Add(device.Name);

comboBox1.SelectedIndex = 0;

\_CaptureDevice = new VideoCaptureDevice();

comboBox2.DataSource = SerialPort.GetPortNames();

string[] tocdotruyen = { "1200", "2400", "4800", "9600", "19200", "38400", "57600", "115200" };

comboBox3.DataSource = tocdotruyen;

comboBox3.SelectedIndex = 3;

serialPort1.DataReceived += SerialPort1\_DataReceived;

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void SerialPort1\_DataReceived(object sender, SerialDataReceivedEventArgs e)

{

}

void ScrollDown()

{

try

{

listBox1.SelectedIndex = listBox1.Items.Count - 1;

listBox1.SelectedIndex = -1;

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void get\_Frame(object sender, NewFrameEventArgs eventArgs)

{

//Đưa ảnh vào khung nhận dạng - PictureBox

Bitmap anhnguon = (Bitmap)eventArgs.Frame.Clone();

pictureBox1.Image = Phathien(anhnguon);

pictureBox2.Image = anhvien;

pictureBox3.Image = anhnhiphan;

pictureBox4.Image = locmau;

}

private System.Drawing.Point[] ToPointsArray(List<IntPoint> points)

{

System.Drawing.Point[] array = new System.Drawing.Point[points.Count];

for (int i = 0, n = points.Count; i < n; i++)

{

array[i] = new System.Drawing.Point(points[i].X, points[i].Y);

}

return array;

}

private Bitmap Phathien(Bitmap \_bitmapSourceImage)

{

#region Color filtering by Euclidean filtering

switch (iColorMode)

{

case 1:

//\_colorFilter.CenterColor = new RGB(230, 30, 30);

iRedValue = hScrollBar3.Value;

iBlueValue = hScrollBar2.Value;

iGreenValue = hScrollBar1.Value;

\_colorFilter.CenterColor = new RGB((byte)iRedValue, (byte)iGreenValue, (byte)iBlueValue);

\_colorFilter.Radius = (short)iRadius;

locmau = \_colorFilter.Apply(\_bitmapSourceImage);

break;

case 2:

iRedValue = hScrollBar3.Value;

iBlueValue = hScrollBar2.Value;

iGreenValue = hScrollBar1.Value;

\_colorFilter.CenterColor = new RGB((byte)iRedValue, (byte)iGreenValue, (byte)iBlueValue);

\_colorFilter.Radius = (short)iRadius;

locmau = \_colorFilter.Apply(\_bitmapSourceImage);

break;

case 3:

iRedValue = hScrollBar3.Value;

iBlueValue = hScrollBar2.Value;

iGreenValue = hScrollBar1.Value;

\_colorFilter.CenterColor = new RGB((byte)iRedValue, (byte)iGreenValue, (byte)iBlueValue);

\_colorFilter.Radius = (short)iRadius;

locmau = \_colorFilter.Apply(\_bitmapSourceImage);

break;

}

#endregion

Grayscale \_grayscale = new Grayscale(0.2125, 0.7154, 0.0721);

anhxam = \_grayscale.Apply(locmau);

//nhận dạng tự động

if (\_blurFlag == true)

{

//Blur \_blurfilter = new Blur();

GaussianBlur \_blurfilter = new GaussianBlur(1.5);

lammoanh = \_blurfilter.Apply(anhxam);

anhvien = \_edgeFilter.Apply(lammoanh);

}

else if (\_blurFlag == false)

{

anhvien = \_edgeFilter.Apply(anhxam);

}

Threshold \_threshold = new Threshold(iThreshold);

anhnhiphan = \_threshold.Apply(anhvien);

BlobCounter \_blobCounter = new BlobCounter();

//Configure Filter

\_blobCounter.MinWidth = 70;

\_blobCounter.MinHeight = 70;

\_blobCounter.FilterBlobs = true;

\_blobCounter.ProcessImage(anhnhiphan);

Blob[] \_blobPoints = \_blobCounter.GetObjectsInformation();

Graphics \_g = Graphics.FromImage(\_bitmapSourceImage);

SimpleShapeChecker \_shapeChecker = new SimpleShapeChecker();

for (int i = 0; i < \_blobPoints.Length; i++)

{

List<IntPoint> \_edgePoint = \_blobCounter.GetBlobsEdgePoints(\_blobPoints[i]);

List<IntPoint> \_corners = null;

AForge.Point \_center;

float \_radius;

//nhận dạng hình chữ nhật

if (\_shapeChecker.IsQuadrilateral(\_edgePoint, out \_corners))

{

dieukhienxylanh1();

//Vẽ điểm tham chiếu của picturebox

\_g.DrawEllipse(\_PictureboxPen, (float)(pictureBox1.Size.Width),

(float)(pictureBox1.Size.Height),

(float)10, (float)10);

//Cài đặt bản vẽ cho đường viền của đối tượng được phát hiện

Rectangle[] \_rects = \_blobCounter.GetObjectsRectangles();

System.Drawing.Point[] \_coordinates = ToPointsArray(\_corners);

Pen \_pen = new Pen(Color.Blue, ipenWidth);

int \_x = \_coordinates[0].X;

int \_y = \_coordinates[0].Y;

//Cài đặt bản vẽ cho trọng tâm của đối tượng được phát hiện

int \_centroid\_X = (int)\_blobPoints[0].CenterOfGravity.X;

int \_centroid\_Y = (int)\_blobPoints[0].CenterOfGravity.Y;

// Vẽ điểm trọng tâm cho đối tượng

\_g.DrawEllipse(\_pen, (float)(\_centroid\_X), (float)(\_centroid\_Y), (float)10, (float)10);

//Xác định tọa độ

int \_deg\_x = (int)\_centroid\_X - pictureBox1.Size.Width;

int \_deg\_y = pictureBox1.Size.Height - (int)\_centroid\_Y;

textBox1.Text = ("Tọa độ: (" + \_deg\_x + ", " + \_deg\_y + ")");

//Vẽ đường viền cho đói tượng

if (\_coordinates.Length == 4)

{

string \_shapeString = "" + \_shapeChecker.CheckShapeType(\_edgePoint);

\_g.DrawString(\_shapeString, \_font, \_brush, \_x, \_y);

\_g.DrawPolygon(\_pen, ToPointsArray(\_corners));

}

}

//nhan dang hinh tron

if (\_shapeChecker.IsCircle(\_edgePoint, out \_center, out \_radius))

{

dieukhienxylanh2();

//tạo điểm gốc

\_g.DrawEllipse(\_PictureboxPen, (float)(pictureBox1.Size.Width),

(float)(pictureBox1.Size.Height),

(float)10, (float)10);

// Cài đặt bản vẽ cho đường viền của đối tượng được phát hiện

Rectangle[] \_rects = \_blobCounter.GetObjectsRectangles();

Pen \_pen = new Pen(Color.Red, ipenWidth);

string \_shapeString = "" + \_shapeChecker.CheckShapeType(\_edgePoint);

int \_x = (int)\_center.X;

int \_y = (int)\_center.Y;

///

/// Vẽ đường viền của đối tượng được phát hiện

///

\_g.DrawString(\_shapeString, \_font, \_brush, \_x, \_y);

\_g.DrawEllipse(\_pen, (float)(\_center.X - \_radius),

(float)(\_center.Y - \_radius),

(float)(\_radius \* 2),

(float)(\_radius \* 2));

//Vẽ điểm trọng tâm của đối tượng

int \_centroid\_X = (int)\_blobPoints[0].CenterOfGravity.X;

int \_centroid\_Y = (int)\_blobPoints[0].CenterOfGravity.Y;

\_g.DrawEllipse(\_pen, (float)(\_centroid\_X), (float)(\_centroid\_Y), (float)10, (float)10);

}

//Nhận dạng hình tam giác

if (\_shapeChecker.IsTriangle(\_edgePoint, out \_corners))

{

dieukhiendongco();

//Tạo gốc tọa độ

\_g.DrawEllipse(\_PictureboxPen, (float)(pictureBox1.Size.Width),

(float)(pictureBox1.Size.Height),

(float)10, (float)10);

// Cài đặt bản vẽ cho đường viền của đối tượng được phát hiện

Rectangle[] \_rects = \_blobCounter.GetObjectsRectangles();

Pen \_pen = new Pen(Color.Green, ipenWidth);

string \_shapeString = "" + \_shapeChecker.CheckShapeType(\_edgePoint);

int \_x = (int)\_center.X;

int \_y = (int)\_center.Y;

// Cài đặt bản vẽ cho trọng tâm của đối tượng được phát hiện

int \_centroid\_X = (int)\_blobPoints[0].CenterOfGravity.X;

int \_centroid\_Y = (int)\_blobPoints[0].CenterOfGravity.Y;

///Vẽ đường viền của đối tượng được phát hiện

\_g.DrawString(\_shapeString, \_font, \_brush, \_x, \_y);

\_g.DrawPolygon(\_pen, ToPointsArray(\_corners));

//Vẽ điểm trọng tâm của đối tượng

\_g.DrawEllipse(\_pen, (float)(\_centroid\_X), (float)(\_centroid\_Y), (float)10, (float)10);

}

}

return \_bitmapSourceImage;

}

private void dieukhiendongco()

{

try

{

serialPort1.Write("3");

}

catch

{

MessageBox.Show("Khong the dieu khien");

}

}

private void dieukhienxylanh2()

{

try

{

serialPort1.Write("2");

}

catch

{

MessageBox.Show("Khong the dieu khien");

}

}

private void dieukhienxylanh1()

{

try

{

serialPort1.Write("1");

}

catch

{

MessageBox.Show("Khong the dieu khien");

}

}

}

}

**CHƯƠNG 5: KẾT LUẬN**

5.1. 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 những gì mà sản phẩm đề ra
* 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 hình dạng đề ra của sản phẩm, có đếm sản phẩm.Tốc độ xử lý tương đối nhanh.
* Những mặt hạn chế

Ngoài những kết quả đạt được thì hệ thống vẫn còn những hạn chế sau: Hệ thống nhận diện bị sai lệch khhi ánh sáng không tốt, chỉ phân loại được sản phẩm theo hình dạng, không phát hiện được sản phẩm bị lỗi.

5.2 Hướng phát triển của đề tài

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] Phạm Thành Long, Đặng Doanh Hoằng, Thiết Kế Sản Phẩm CĐT, 2022.

[2] Giáo trình Thiết kế hệ thống CĐT, bộ môn CĐT, trường Đại Học kỹ thuật công nghiệp. [3] Kênh youtube : “Hyeonjun Park”, “AKHTAR JAMIL”

[4] Phần mềm : Arduino IDE, Microsoft Visual Studio,…..

Trang wed tham khảo :

[4] <https://nabajyotighosh.github.io>

[5] <https://create.arduino.cc>

[6]https://github.com/MrLacquer/CSharp-image-AForge/blob/master/2016310249\_ObjectDetecting/Form1.cs