**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KỸ THUẬT MÁY TÍNH**

**NGUYỄN DUY MINH**

**PHẠM HOÀI PHƯƠNG**

**KHÓA LUẬN TỐT NGHIỆP**

**ROBOT DẪN ĐƯỜNG: XÂY DỰNG BẢN ĐỒ VÀ XÁC ĐỊNH VỊ TRÍ SỬ DỤNG THỊ GIÁC MÁY TÍNH**

**NAVIGATION ROBOT: BUILDING MAP AND DETERMINING LOCATION USING COMPUTER VISION**

**KỸ SƯ NGÀNH KỸ THUẬT MÁY TÍNH**

**TP. HỒ CHÍ MINH 2020**

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KỸ THUẬT MÁY TÍNH**

**NGUYỄN DUY MINH**

**PHẠM HOÀI PHƯƠNG**

**KHÓA LUẬN TỐT NGHIỆP**

**ROBOT DẪN ĐƯỜNG: XÂY DỰNG BẢN ĐỒ VÀ XÁC ĐỊNH VỊ TRÍ SỬ DỤNG THỊ GIÁC MÁY TÍNH**

**NAVIGATION ROBOT: BUILDING MAP AND DETERMINING LOCATION USING COMPUTER VISION**

**KỸ SƯ NGÀNH KỸ THUẬT MÁY TÍNH**

**GIẢNG VIÊN HƯỚNG DẪN**

**ThS. HÀ LÊ HOÀI TRUNG**

**TP. HỒ CHÍ MINH 2020**

**DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN**

|  |  |
| --- | --- |
| **ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** |
|  | *TP. HCM, ngày…..tháng…..năm……..* |

**NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP**

**CỦA CÁN BỘ HƯỚNG DẪN**

|  |  |  |
| --- | --- | --- |
| **Tên khóa luận:** | | |
| **ROBOT DẪN ĐƯỜNG: XÂY DỰNG BẢN ĐỒ VÀ XÁC ĐỊNH VỊ TRÍ**  **SỬ DỤNG THỊ GIÁC MÁY TÍNH** | | |
| **Nhóm SV thực hiện:** | | **Cán bộ hướng dẫn:** |
| Nguyễn Duy Minh | 15520483 | ……………………………….. |
| Phạm Hoài Phương | 15520665 | ……………………………….. |
| **Đánh giá Khóa luận**   1. Về cuốn báo cáo:   Số trang Số chương  Số bảng số liệu Số hình vẽ  Số tài liệu tham khảo Sản phẩm  Một số nhận xét về hình thức cuốn báo cáo:  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................   1. Về nội dung nghiên cứu:   …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................   1. Về chương trình ứng dụng:   …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................   1. Về thái độ làm việc của sinh viên:   …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  **Đánh giá chung:**Khóa luận đạt/không đạt yêu cầu của một khóa luận tốt nghiệp kỹ sư/ cử nhân, xếp loại Giỏi/ Khá/ Trung bình  **Điểm từng sinh viên:**  Nguyễn Duy Minh**:………../10**  Phạm Hoài Phương**:………../10** | | |

|  |  |
| --- | --- |
|  | Người nhận xét  (Ký tên và ghi rõ họ tên) |

|  |  |
| --- | --- |
| **ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** |
|  | *TP. HCM, ngày…..tháng…..năm……..* |

**NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP**

**CỦA CÁN BỘ PHẢN BIỆN**

|  |  |  |
| --- | --- | --- |
| **Tên khóa luận:** | | |
| **ROBOT DẪN ĐƯỜNG: XÂY DỰNG BẢN ĐỒ VÀ XÁC ĐỊNH VỊ TRÍ**  **SỬ DỤNG THỊ GIÁC MÁY TÍNH** | | |
| **Nhóm SV thực hiện:** | | **Cán bộ phản biện:** |
| Nguyễn Duy Minh | 15520483 | ……………………………….. |
| Phạm Hoài Phương | 15520665 | ……………………………….. |
| **Đánh giá Khóa luận**   1. Về cuốn báo cáo:   Số trang Số chương  Số bảng số liệu Số hình vẽ  Số tài liệu tham khảo Sản phẩm  Một số nhận xét về hình thức cuốn báo cáo:  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................   1. Về nội dung nghiên cứu:   …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................   1. Về chương trình ứng dụng:   …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................   1. Về thái độ làm việc của sinh viên:   …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  …………………………………………………………………………………………………...........................................  **Đánh giá chung:**Khóa luận đạt/không đạt yêu cầu của một khóa luận tốt nghiệp kỹ sư/ cử nhân, xếp loại Giỏi/ Khá/ Trung bình  **Điểm từng sinh viên:**  Vũ Ngọc Thịnh**:………../10**  Nguyễn Trọng Hoàng**:………../10** | | |

|  |  |
| --- | --- |
|  | Người nhận xét  (Ký tên và ghi rõ họ tên) |

**LỜI CẢM ƠN**

|  |  |
| --- | --- |
| ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** |

**ĐỀ CƯƠNG CHI TIẾT**

|  |  |
| --- | --- |
| **TÊN ĐỀ TÀI: ROBOT DẪN ĐƯỜNG: XÂY DỰNG BẢN ĐỒ VÀ XÁC ĐỊNH VỊ TRÍ SỬ DỤNG THỊ GIÁC MÁY TÍNH** | |
| **Cán bộ hướng dẫn: Th.S Hà Lê Hoài Trung** | |
| **Thời gian thực hiện:** Từ ngày 09/09/2019 đến ngày 29/12/2019 | |
| **Sinh viên thực hiện:**  **<Nguyễn Duy Minh – MSSV: 15520483>**  **<Phạm Hoài Phương – MSSV: 15520665>** | |
| **Nội dung đề tài:***(Mô tả chi tiết mục tiêu, phạm vi, đối tượng, phương pháp thực hiện, kết quả mong đợi của đề tài)*  *Mục tiêu, phạm vi, đối tượng:*  Xây dựng một hệ thống có khả năng vẽ bản đồ của một khu vực bất kì (tòa nhà, văn phòng, khu dân cư, ...)  Xây dựng thuật toán tự xác định ví của robot ở những nơi đã đi qua trong bản đồ.  Hiện thực hệ thống tại tầng 6 tòa nhà E của trường Đại học Công Nghệ Thông Tin.  Phương pháp thực hiện:   * Tìm hiểu và xây dựng giải thuật vẽ bản đồ. * Tìm hiểu các giải thuật phân tích đặc trưng ảnh dựa vào camera và sử dụng thư viện mở openCV. * Tìm hiểu và xây dựng thuật toán xác định vị trí. * Tiến hành ghép nối các module và thực hiện demo để đánh giá kết quả. | |
| **Kế hoạch thực hiện:**   |  |  |  |  | | --- | --- | --- | --- | | Thời gian bắt đầu | Thời gian kết thúc | Công việc | Lưu ý | | 15.9 | 22.9 | Lên ý tưởng |  | | 22.9 | 6.10 | Robot : Tìm hiểu các thiết bị để lựa chọn phần cứng cho xe.  Thị Giác : Tìm hiểu về thị giác máy tính. |  | | 6.10 | 20.10 | Robot : Lực chọn các mục tiêu ban đầu cho xe về tốc độ, kiểu dáng….  Thị Giác : Tìm hiểu các thuật toán phát hiện đường line cho xe. |  | | 20.10 | 3.11 | Robot: Tiến hành lựa chọn khung xe.  Thị Giác :Tiến hành đánh gia các thuật toán phát hiện đường line đã tìm hiểu và hiện thực bằng thự viện mở OpenCV. |  | | 3.11 | 10.11 | Báo cáo tiến độ giữa kỳ |  | | 10.11 | 17.11 | Robot : Tiến hành lập trình giao tiếp các module lại phần cứng lại với nhau.  Thị Giác : Tìm hiểu các giải thuật phát hiện đối tượng bằng camera. |  | | 17.11 | 24.11 | Robot : Xây dựng phương pháp di chuyển cho xe.  Thị Giác :Từ các giải thuật tìm hiểu tiến hành hiện thực phát hiện đối tượng ứng dụng thư viện mở openCV. |  | | 24.11 | 1.12 | Robot : Tiến hành lập trình kết nối các module lại phần cứng lại với nhau.  Thị Giác : Tìm hiểu các giải thuật phát hiện đối tượng bằng camera. |  | | 1.12 | 8.12 | Robot : Xây dựng phương pháp di chuyển cho xe.  Thị Giác : Dựa vào thuật toán tìm hiểu lưa chọn thuật toán phù hợp để hiển thực và đánh giá. |  | | 8.12 | 15.12 | Tiến hành kết nối thị phần thị giác vào xe. |  | | 15.12 | 5.1 | Tiến hành sửa lỗi, chạy demo và đánh giá kết quả. |  | | 5.1 | 11.1 | Hoàn thiện sản phẩm cuối cùng |  | | 11.1 | 15.1 | Phản biệt khóa luận tốt nghiêp |  | | 15.1 | 22.1 | Chuẩn bị bảo vệ khóa luận tốt nghiệp |  | | 22.1 | 25.1 | Bảo vệ khóa luận tốt nghiệp |  | | |
| **Xác nhận của CBHD**  (Ký tên và ghi rõ họ tên)  **Ths.Hà Lê Hoài Trung** | **TP. HCM, ngày….tháng …..năm…..**  **Sinh viên**  (Ký tên và ghi rõ họ tên)  **Phạm Hoài Phương Nguyễn Duy Minh** |

MỤC LỤC

[Tóm tắt khóa luận 19](#_Toc26819856)

[Chương 1. GIỚI THIỆU ĐỀ TÀI 20](#_Toc26819857)

[1.1. Lý do chọn đề tài 20](#_Toc26819858)

[1.2. Mục tiêu đề tài 20](#_Toc26819859)

[1.3. Khách thể, đối tượng và phạm vi nghiên cứu 21](#_Toc26819860)

[1.3.1. Khách thể nghiên cứu 21](#_Toc26819861)

[1.3.2. Đối tượng nghiên cứu 21](#_Toc26819862)

[1.3.3. Phạm vi nghiên cứu 21](#_Toc26819863)

[1.4. Thuận lợi và khó khăn 21](#_Toc26819864)

[1.4.1. Thuận lợi 21](#_Toc26819865)

[1.4.2. Khó khăn 21](#_Toc26819866)

[Chương 2. TỔNG QUAN VỀ THƯ VIỆN OPENCV VÀ THỊ GIÁC MÁY TÍNH 22](#_Toc26819867)

[2.1. Thị giác máy tính là gì 22](#_Toc26819868)

[2.2. Thư viện OpenCV 23](#_Toc26819869)

[2.2.1. OpenCV là gì? 23](#_Toc26819870)

[2.2.2. Lý do sử dụng OpenCV 23](#_Toc26819871)

[2.2.3. Cấu trúc cơ bản của OpenCV 24](#_Toc26819872)

[2.2.4. Các câu lệnh cơ bản trong OpenCV 25](#_Toc26819873)

[Chương 3. GIẢI THUẬT THỊ GIÁC MÁY TÍNH 26](#_Toc26819874)

[3.1. Thuật toán vẽ bản đồ 26](#_Toc26819875)

[3.1.1. Giới thiệu bài toán lập bản đồ 26](#_Toc26819876)

[3.1.2. SLAM là gì? 26](#_Toc26819877)

[3.1.3. Vấn đề của SLAM 27](#_Toc26819878)

[3.1.4. Thuật toán SLAM 28](#_Toc26819879)

[3.1.5. Ứng dụng của SLAM vào thực tế 28](#_Toc26819880)

[3.2. Thuật toán xác định vị trí 28](#_Toc26819881)

DANH MỤC HÌNH VẼ

No table of figures entries found.

DANH MỤC BẢNG

[Table 2.1 các bước tìm đường đi ngắn nhất 16](#_Toc10842194)

# Tóm tắt khóa luận

Ngày nay, sự phát triển của robot ngày càng được chú trọng và được ứng dụng vào hầu hết tất cả lĩnh vực trong đời sống con người: Robot hút bụi, robot vận chuyển hàng hóa tại các nhà kho, robot phân loại hàng hóa trong dịch vụ bưu chính, robot thám hiểm, … Chúng ta có thể thấy rằng robot đã trở thành một nhu cầu không thể thiếu của con người. Ở Việt Nam, robot cũng là lĩnh vực mà các bạn sinh viên quan tâm nhiều. Tuy nhiên, robot áp đụng thị giác máy tính vẫn đang còn mới mẻ. Vì vậy, …

Đề tài khóa luận “ROBOT DẪN ĐƯỜNG: XÂY DỰNG BẢN ĐỒ VÀ XÁC ĐỊNH VỊ TRÍ SỬ DỤNG THỊ GIÁC MÁY TÍNH”, mong muốn xây dựng được một hệ thống có thể xây dựng bản đồ và xác định được vị trí của robot ở trong bản đồ đó (thay vì sử dụng GPS hoặc các thiết bị ngoại vi khác). Từ hệ thống này, có thể làm tiền đề cho các bạn sinh viên (hoặc những người quan tâm đến đề tài này) có thể áp dụng được và xây dựng được một Robot hoàn chỉnh có khả năng dẫn đường trong các văn phòng, trung tâm thương mại, … Bản báo cáo này sẽ trình bày các nội dung lí thuyết cơ bản về thị giác máy tính, các thuật toán vẽ bản đồ, xác định vị trí cũng như các nghiên cứu, hiện thực và thử nghiệm dựa vào các video mô phỏng hoạt động di chuyển của robot.

# Chương 1. GIỚI THIỆU ĐỀ TÀI

## Lý do chọn đề tài

Hiện nay, trong các khu dân cư, trung tâm thương mại, trường học, … Với diện tích rộng lớn, một người ở xa đến sẽ khó khăn trong việc di chuyển đến khu vực bất kì nếu không có bản đồ khu vực, tòa nhà hoặc sự giúp đỡ của những người xung quanh, … Nhất là đối với những người ở những vùng quê, ít tiếp xúc với cuộc sống nơi thành thị, đa phần họ sẽ ngại bắt chuyện với người lạ.

Chính vì vậy, nhóm đã quyết định thực hiện đề tài “Robot dẫn đường: Xây dựng bản đồ và xác định vị trí sử dụng thị giác máy tính”, là một phần của hệ thống Robot dẫn đường, với mong muốn xây dựng được một hệ thống hoàn chỉnh, xác định vị trí chính xác, nhanh chóng, và là tiền đề cho các nhóm có hứng thú với đề tài này có thể áp dụng và xây dựng được một Robot hoàn chỉnh để có thể đưa vào thử nghiệm và sử dụng.

## Mục tiêu đề tài

Mục tiêu đặt ra là thiết kế được một hệ thống ứng dụng thị giác máy tính để có thể xây dựng bản đồ những nơi đã đi qua và có thể xác định được vị trí bất kì trong bản đồ đã vẽ.

Từ mục tiêu của đề tài, nhóm đã xác định được các công việc phải hoàn thành:

* Tìm hiểu về thị giác máy tính và thư viện mở OpenCV để có thể hiện thực được các thuật toán
* Tìm hiểu và hiện thực thuật toán SLAM để có thể xây dựng được bản đồ.
* Tìm hiểu và hiện thực các thuật toán phát hiện đặc trưng ảnh để hỗ trợ trong việc xác định vị trí.
* Kết nối hai module để có được hệ thống hoàn chỉnh và tiến hành chạy demo đánh giá kết quả.

## Khách thể, đối tượng và phạm vi nghiên cứu

### Khách thể nghiên cứu

* Tìm hiểu các hoạt động, cách sử dụng board Jetson Nano, một dạng máy tính thu nhỏ trong một board mạch. Chúng ta có thể xây dựng các giải thuật cho thị giác máy tính trên board này.
* Thư viện OpenCV là thư viện mã nguồn mở, đáp ứng hầu hết các yêu cầu về xử lý ảnh.

### Đối tượng nghiên cứu

* Các sử dụng thư viện OpenCV.
* Các thuật toán xử lý ảnh
* Các cài đặt và sử dụng thư viện OpenCV sử dụng ngôn ngữ Python lên board Jetson nano.

### Phạm vi nghiên cứu

* Thiết kế hệ thống đơn giản, sử dụng camera để quay lại video mô phỏng di chuyển của robot. Chạy demo trên board Jetson nano để xây dựng bản đồ và xác định vị trí của robot.

## Thuận lợi và khó khăn

### Thuận lợi

Thư viện mã nguồn mở OpenCV đáp ứng hầu hết tất cả yêu cầu của hệ thống. Vì vậy, nhóm dễ dàng tiếp cận, tìm hiểu và hiện thực các thuật toán xử lý ảnh.

Xác định được vấn đề cần giải quyết nên nhóm đã lên kế hoạch thực hiện chi tiết, công việc được phân chia rõ ràng.

Sự hướng dẫn nhiệt tình của giảng viên hướng dẫn mỗi khi nhóm gặp khó khăn, trợ giúp, đốc thúc nhóm đảm bảo tiến bộ hoàn thành đề tài

### Khó khăn

Bên cạnh những thuận lợi khi thực hiện, nhóm còn gặp những khó khăn nhất định.

# TỔNG QUAN VỀ THƯ VIỆN OPENCV VÀ THỊ GIÁC MÁY TÍNH

## Thị giác máy tính là gì

Thị giác máy tính (tiếng Anh: computer vision) là một lĩnh vực bao gồm các phương pháp thu nhận, xử lý ảnh kỹ thuật số, phân tích và nhận dạng các hình ảnh và, nói chung là dữ liệu đa chiều từ thế giới thực để cho ra các thông tin số hoặc biểu tượng. Việc phát triển lĩnh vực này có bối cảnh từ việc sao chép các khả năng thị giác con người bởi sự nhận diện và hiểu biết một hình ảnh mang tính điện tử. Sự nhận diện hình ảnh có thể xem là việc giải quyết vấn đề của các biểu tượng thông tin từ dữ liệu hình ảnh qua cách dùng các mô hình được xây dựng với sự giúp đỡ của các ngành lý thuyết học, thống kê, vật lý và hình học. Thị giác máy tính cũng được mô tả là sự tổng thể của một dải rộng các quá trình tự động và tích hợp và các thể hiện cho các nhận thức thị giác. (Nguồn: wikipedia)

Thị giác máy tính là một môn học khoa học liên quan đến lý thuyết đằng sau các hệ thống nhân tạo có trích xuất các thông tin từ các hình ảnh. Dữ liệu hình ảnh có thể nhiều dạng, chẳng hạn như chuỗi video, các cảnh từ đa camera, hay dữ liệu đa chiều từ máy quét y học. Thị giác máy tính còn là một môn học kỹ thuật, trong đó tìm kiếm việc áp dụng các mô hình và các lý thuyết cho việc xây dựng các hệ thống thị giác máy tính. (Nguồn: wikipedia)

Các lĩnh vực con của thị giác máy tính bao gồm tái cấu trúc cảnh, dò tìm sự kiện, theo dõi video, nhận diện bố cục đối tượng, học, chỉ mục, đánh giá chuyển động và phục hồi ảnh. (Nguồn: wikipedia)

Hiện tại, lĩnh vực thị giác máy tính đang ngày càng được ưa chuộng, có rất nhiều thay đổi sau mỗi năm. Các lĩnh vực áp dụng thị giác máy tính có rất nhiều hướng nghiên cứu và giải pháp hiện thực khác nhau, mỗi giải pháp giải đều cho ra một kết quả nhất định.

## Thư viện OpenCV

### OpenCV là gì?

OpenCV là một thư viện mã nguồn mở hàng đầu cho thị giác máy tính (computer vision), xử lý ảnh và máy học, và các tính năng tăng tốc GPU trong hoạt động thời gian thực.

Là một sản phẩn được cấp phép BSD (là một giấy phép phần mềm tự do với các điều kiện đơn giản được sử dụng rộng rãi cho phần mềm máy tính) giúp các doanh nghiệp dễ dàng sử dụng và đổi mã. Nó có giao diện C/C++, Python, Matlab và hỗ trợ Windows, Linux, Mac OS và Android.

OpenCV có hệ thống đơn giản, dễ dàng tiếp cận và sử dụng. Lập trình viên có thể dễ dàng tạo ra các ứng dụng đơn giản về thị giác máy tính như phát hiện vật thể, nhận diện màu sắc, hình dáng, … Ngoài ra OpenCV còn hỗ trợ các thuật toán bậc cao như nhận diện khuôn mặt, cử chỉ, hoạt động của con người.

### Lý do sử dụng OpenCV

Thư viện OpenCV có hơn 2500 thuật toán được tối ưu hóa, được sử dụng để phát hiện và nhận diện khuôn mặt, nhận dạng đối tượng, phân loại hành động của con người trong video, theo dõi chuyển động của camera, theo dõi các vật thể chuyển động, trích xuất mô hình 3D của các vật thểm , … OpenCV có hơn 47 nghìn người dùng cộng đồng và số lượng lượt tải ước tính vượt quá 18 triệu. (Nguồn: opencv.org)

Thư viện được sử dụng rộng rãi trong các công ty, nhón nghiên cứu và các cơ quan chính phủ. Cùng với các công ty như Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota sử dụng OpenCV. Việc sử dụng OpenCV được triển khai bao gồm từ việc ghép các hình ảnh đường phố với nhau, phát hiện xâm nhập vào video giám sát ở Israel, theo dõi thiết bị khai thác mỏ ở Trung Quốc, giúp robot điều khiển và nhặt đồ vật tại Willow Garage, phát hiện tai nạn đuối nước ở châu Âu, Tây Ban Nha và New York, kiểm tra đường băng cho các mảnh vụn ở Thổ Nhĩ Kỳ, kiểm tra nhãn trên các sản phẩm tại các nhà máy trên toàn thế giới để phát hiện khuôn mặt nhanh ở Nhật Bản. (Nguồn: opencv.org)

### Cấu trúc cơ bản của OpenCV

Cấu trúc cơ bản của OpenCV được thể hiện qua hình sau:

A screenshot of a cell phone

Description automatically generated

Hình 2.1 Cấu trúc cơ bản của OpenCV

OpenCV được chia làm 4 thành phần chính:

* + CV (computer vision): Thành phần xử lý ảnh cơ bản và các thuật toán thị giác máy tính.
  + MLL (machine learning library): Thư viện máy học, bao gồm rất nhiều các lớp thống kê và công cụ gom nhóm.
  + HighGUI: Chứa các thủ tục I/O và các hàm dùng trong việc lưu trữ và xử lý hình ảnh, video.
  + CXCORE: Chứa các cấu trúc và nội dung của các dữ liệu cơ bản.

Phiên bản mới nhất (opencv-4.1.2) gồm các module sau:

* **Core**: Chứa các cấu trúc, class cơ bản mà OpenCV sẽ sử dụng trong việc lưu trữ và xử lý hình ảnh như Mat, Scale, Point, Vec, … và các phương thức cơ bản sử dụng cho các module khác.
* **Imgproc**: Đây là một module xử lý hình ảnh của OpenCV bao gồm các bộ lọc (filter) linear và non-linear và các phép biến đổi hình học (tranforrmation) như resize hoặc các phép biến đổi affine, các thuật toán liên quan đến histogram (biểu đồ) của hình ảnh.
* **Highui**: Là một module cho phép tương tác với người dùng trên UI (User Interface) như hiển thị hình ảnh, video capturing.
* **Features2d**: Module tìm các đặc trưng (feature) của hình ảnh. Trong module có hiện thực các thuật toán rút trích đặc trưng như PCA, …
* **Calib3d**: Hiệu chuẩn máy ảnh và xây dựng 3D.
* **Objdetect**: Module cho việc phát hiện các đối tượng như khuôn mặt, đôi mắt, người, xe hơi, … trong hinfh ảnh. Các thuật toán được sử dụng trong module này là Haar-like Features.
* **Ml**: Module này chứa các thuật toán về Machine Learning phục vụ cho các bài toán phân lớp (Classfitication) và bài toán gom nhóm (Clustering). Ví dụ như thuật toán SVM, ANN, …
* **Video**: Module phân tích video bao gồm ước lượng chuyển động, trừ nền và các thuật toán theo dõi đối tượng (object tracking).

### Các câu lệnh cơ bản trong OpenCV

**Đọc một bức ảnh:**

Sử dụng hàm cv2.imread(“file\_name”, option) để đọc một bức ảnh.

File\_name: có thể là bức ảnh tại thư mục hiện hành hoặc đường dẫn tới ảnh đó.

Option:

* cv2.IMREAD\_COLOR: Tải một hình ảnh màu.
* cv2.IMREAD\_GRAYSCALE: Tải hình ảnh ở chế độ màu xám.
* cv2.IMREAD\_UNCHANGED: Tải hình ảnh như vậy bao gồm cả kênh alpha.

**Hiển thị một bức ảnh:**

Sử dụng hàm cv2.imshow(‘window\_name’, image) để hiển thị một bức ảnh.

**Lưu trữ một bức ảnh:**

Sử dụng hàm cv2.write(“name\_of\_image.png”, Img) để lưu ảnh.

**Đọc dữ liệu từ camera/webcam hay mở video:**

Sử dụng hàm cv2.VideoCapture() để đọc dữ liệu từ camera hoặc mở video

Vd: để lấy dữ liệu từ camera:

cap = cv2.VideoCapture() // Để đọc video ta cần truyền đường dẫn video vào

while(True):

ret, frame = cap.read()

//Xuất từng frame ảnh ra màn hình

cv2.imshow('frame',frame)

cap.release()

# GIẢI THUẬT THỊ GIÁC MÁY TÍNH

## Thuật toán vẽ bản đồ

### Giới thiệu bài toán lập bản đồ

Bài toán lập bản đồ là một trong những vấn đề được nghiên cứu chủ yếu của ngảnh Robot học. Xét một robot đơn giản sử dụng một bộ bánh xe có gắn động cơ với các thiết bị vật lý để kiểm soát tốc độ, hướng di chuyển và những cảm biến để thu thập thông tin về môi trường. Nhiệm vụ của robot này là di chuyển để lập bản đồ những nơi con người không thể hoặc không muốn tiếp cận, đồng thời robot tự xác định được vị trí của nó so với những đối tượng xung quanh. Kỹ thuật để robot có thể thực hiện được nhiệm vụ này gọi là SLAM.

### SLAM là gì?

SLAM (simultaeous localization and mapping) là hệ thống sử dụng thông tin ảnh thu được từ camera để tái tạo môi trường bên ngoài, bằng cách đưa thông tin môi trường vào một map (2D hoặc 3D). Từ đó, thiết bị (robot, camera, xe) có thể định vị (localization) đang ở đâu, trạng thái, tư thế của nó trong map để tự động thiết lập đường đi (path planning) trong môi trường hiện tại.

Điều khiển tự động thiết bị robot chia làm 3 vấn đề chính: định vị (localization), tái tạo môi trường (mapping) và hoạch định đường đi (path planning).

Ban đầu, hai vấn đề định vị và tái tạo môi trường được nghiên cứu độc lập, tuy nhiên, sau khi nhận thấy:

* Định vị: cần xác định vị trí hiện tại của robot dựa vào bản đồ tái tạo.
* Tái tạo bản đồ: cần xác định vị trí của đối tượng trong bản đồ, để xây dựng bản đồ chính xác nhất, ít sai số.

Vì thế, trong giai đoạn 1985-1990, Raja Chatila và Jean-Paul Laumond (1985) và Randall Smith (1990) đã đề xuất gộp hai vấn đề với nhau để nghiên cứu. Một thời gian sau đó, SLAM ra đời đại diện cho localization và mapping.

### Vấn đề của SLAM

Để tái tạo môi trường vào map, thực thể phải nhận thông tin các phần tử trong môi trường xung quanh nhờ các cảm biến ngoại vi và cảm biến nội vi (cảm biến lazer, GPS, sóng âm thanh, …). Và hầu như mọi cảm biến đều bị ảnh hưởng bởi nhiễu và khả năng của chúng có hạn, thêm nữa là do các thiết bị quá đắt, cồng kềnh do trang bị nhiều thiết bị. Cảm biến GPS không thể hoạt động ở dưới nước, ở các hành tinh khác, và ngay cả trong nhà. Hệ thống SLAM lỗi khi hoạt động trong môi trường động, môi trường với phạm vi quá lớn, môi trường có quá nhiều hoặc quá ít đặc trưng (salient features), khi camera di chuyển không ổn định, và khi các cảm biến xảy ra xung đột.Những nguyên nhân này làm cho việc sử dụng chúng vào ứng dụng thực tế khó khăn.

Đã có những nghiên cứu nâng cao độ chính xác của mô hình SLAM bằng cách kết hợp các thông tin thu được từ các cảm biến. Tuy nhiên, vấn đề kinh phí làm cho hướng phát triển này ít đi vào thực tế. Đồng thời, việc hướng đến tạo ra một sản phẩm tốt, lại rẻ tiền là xu hướng tất yếu của thị trường. Và vì vậy, người ta hướng đến việc nghiên cứu làm sao chỉ sử dụng mỗi camera mà có thể xây dựng được một mô hình SLAM tốt. Điều này có nghĩa là sẽ ứng dụng những kiến thức trong lĩnh vực thị giác máy tính để xử lý và cấu trúc hình ảnh thu được từ camera. (Nguồn: stdio.vn)

### Thuật toán SLAM

### Ứng dụng của SLAM vào thực tế

* Xe hơi tự động lái
* Các thiết bị bay tự động
* Drone giao hàng
* Thiết bị thám hiểm không gian
* Robot giúp việc nhà
* Robot giúp người khiếm thị di chuyển

## Thuật toán Canny Edge Detection

## Thuật toán xác định vị trí

### Bài toán định vị cho robot

Hiện nay, có rất nhiều cảm biến khác nhau được dùng để giải quyết bài toán định bị cho robot như GPS, Lidar, Wifi. Tuy nhiên, hệ thống định vị GPS chỉ thích hợp sử dụng trong điều kiện thời tiết tốt, môi trường lớn, ngoài trời và không thích hợp sử dụng trong môi trường nhỏ. Sử dụng Lidar hoặc Wifi đòi hỏi phải có chi phí cao.

Trong nhiều năm gần đây, thị giác máy tính đang ngày càng phát triển và được ứng dụng nhiều vào lĩnh vực robot học. Việc sử dụng camera để thu thập thông tin hình ảnh của môi trường đã thu hút được rất nhiều sự quan tâm từ những người có hứng thú với lĩnh vực này. Ưu điểm chính của việc sử dụng camera là giá thành rẻ hơn rất nhiều so với các cảm biến khác trong sử dụng những hình ảnh thu được còn được ứng dụng vào bài toán xây dựng bản đồ, định vị và phát triển vật cản.

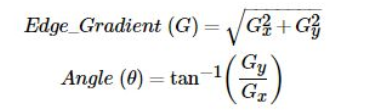
Trong đề tài khóa luận này, nhóm sử dụng 01 camera để áp dụng vào bài toán định vị cho robot. Nhóm đã nghiên cứu các thuật toán Trilateral Localization, SUFT, SIFT, ORB để so sánh và tìm ra thuật toán phù hợp nhất cho đề tài khóa luận này. Tuy nhiên, để hiện thực thuật toán Trilateral Localization, nhóm cần tìm hiểu thêm thuật toán Canny Edge Detection và SVM để hỗ trợ cho thuật toán này.

### Thuật toán Canny Edge Detection

Trong hình ảnh, thường tồn tại các thành phần như: vùng trơn, góc / cạnh và nhiễu. Cạnh trong ảnh mang đặc trưng quan trọng, thường là thuộc đối tượng trong ảnh (object). Do đó, để phát hiện cạnh trong ảnh, giải thuật Canny là một trong những giải thuật phổ biến / nổi tiếng nhất trong Xử lý ảnh.

Giải thuật phát hiện cạnh Canny gồm 4 bước chính sau:

* **Giảm nhiễu**: Làm mờ ảnh, giảm nhiễu dùng bộ lọc Gaussian kích thước 5x5. Kích thước 5x5 thường hoạt động tốt cho giải thuật Canny.
* **Tính Gradient và hướng gradient**: ta dùng bộ lọc Sobel X và Sobel Y (3x3) để tính được ảnh đạo hàm Gx và Gy. Sau đó, ta tiếp tục tính ảnh Gradient và góc của Gradient theo công thức. Ảnh đạo hàm Gx và Gy là ma trận (ví dụ: 640x640), thì kết quả tính ảnh đạo hàm Edge Gradient cũng là một ma trận (640x640), mỗi pixel trên ma trận này thể hiện độ lớn của biến đổi mức sáng ở vị trí tương ứng trên ảnh gốc. Tương tự, ma trận Angle cũng có cùng kích thước (640x640), mỗi pixel trên Angle thể hiện góc, hay nói cách khác là hướng của cạnh. Ví dụ dễ hiểu, nếu góc gradient là 0 độ, thì cạnh của ta trên ảnh sẽ là một đường thẳng đứng (tức tạo góc 90 độ so với trục hoành) (**vuông góc hướng gradient**). Khi tính toán, giá trị hướng gradient sẽ nằm trong đoạn [-180, 180] độ, ta không giữ nguyên các góc này mà gom các giá trị này về 4 bin đại diện cho 4 hướng: hướng ngang (0 độ), hướng chéo bên phải (45 độ), hướng dọc (90 độ) và hướng chéo trái (135 độ).



* **Non-maximum Suppression** (viết tắt NMS): loại bỏ các pixel ở vị trí không phải cực đại toàn cục. Ở bước này, ta dùng một filter 3x3 lần lượt chạy qua các pixel trên ảnh gradient. Trong quá trình lọc, ta xem xét xem độ lớn gradient của pixel trung tâm có phải là cực đại (lớn nhất trong cục bộ - local maximum) so với các gradient ở các pixel xung quanh. Nếu là cực đại, ta sẽ ghi nhận sẽ giữ pixel đó lại. Còn nếu pixel tại đó không phải là cực đại lân cận, ta sẽ set độ lớn gradient của nó về zero. Ta chỉ so sánh pixel trung tâm với 2 pixel lân cận theo **hướng gradient**. Ví dụ: nếu hướng gradient đang là 0 độ, ta sẽ so pixel trung tâm với pixel liền trái và liền phải nó. Trường hợp khác nếu hướng gradient là 45 độ, ta sẽ so sánh với 2 pixel hàng xóm là góc trên bên phải và góc dưới bên trái của pixel trung tâm. Tương tự cho 2 trường hợp hướng gradient còn lại.
* **Lọc ngưỡng**: ta sẽ xét các pixel dương trên mặt nạ nhị phân kết quả của bước trước. Nếu giá trị gradient vượt ngưỡng **max\_val** thì pixel đó **chắc chắn là cạnh**. Các pixel có độ lớn gradient nhỏ hơn ngưỡng **min\_val** sẽ bị loại bỏ. Còn các pixel nằm trong khoảng 2 ngưỡng trên sẽ được xem xét rằng nó có nằm liên kề với những pixel được cho là "chắc chắn là cạnh" hay không. Nếu liền kề thì ta giữ, còn không liền kề bất cứ pixel cạnh nào thì ta loại. Sau bước này ta có thể áp dụng thêm bước hậu xử lý loại bỏ nhiễu (tức những pixel cạnh rời rạc hay cạnh ngắn) nếu muốn. Ảnh minh họa về ngưỡng lọc:

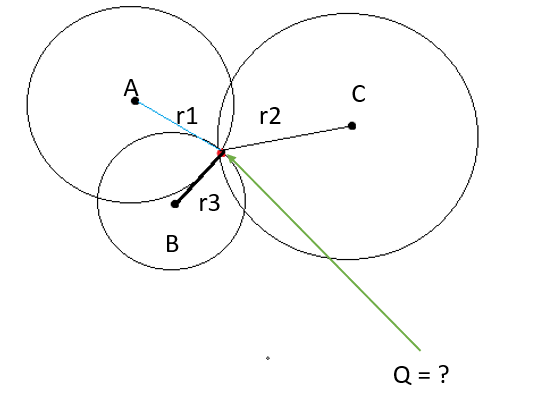
### Thuật toán SVM

SVM (Support Vector Machine) là một khái niệm trong thống kê và khoa học máy tính cho một tập hợp các phương pháp học có giám sát liên quan đến nhau để phân loại và phân tích hồi quy.

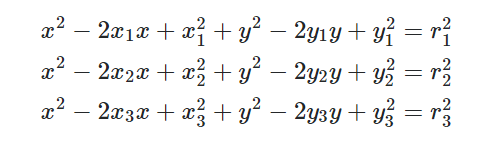
SVM là một thuật toán phân loại nhị phân, SVM nhận dữ liệu vào và phân loại chúng vào hai lớp khác nhau. Với một bộ các ví dụ luyện tập thuộc hai thể loại cho trước, thuật toán luyện tập SVM xây dựng một mô hình SVM để phân loại các ví dụ khác vào hai thể loại đó.

### Thuật toán Trilateral Localization

Cho 3 điểm đã biết A, B, C (với các tọa độ là là tọa độ các landmark đã sử dụng trong bản đồ), và khoảng cách từ robot tới các điểm A, B, C là r1, r2, r3. Áp dụng thuật toán Trilateral Localization, chúng ta có thể tính được vị trí của robot đang đứng (điểm P):



Hình 2.3 Thuật toán Trilateral localzation



### Thuật toán SIFT

SIFT (Scale – Invariant Feature Transform) là thuật toán tiếp cận để phát hiện và trích xuất ra những điểm đặc biệt (keypoint), hay còn gọi là đặc trưng SIFT (SIFT feature). Các đặc trưng này bất biến với việc thay đổi cường độ ánh sáng trong ảnh, thay đổi tỉ lệ ảnh, xoay ảnh, … Thuật toán này được David Lowe khởi xướng vào năm 1999, và đang được ứng dụng rộng rãi cho đến ngày nay. []

Ưu điểm của đặc trưng SIFT:

* Đặc trưng SIFT là những đặc trưng tự nhiên của ảnh, vì vậy, chúng bất biến đối với việc thay đổi tỉ lệ ảnh, xoay ảnh, ánh sáng, …
* Đặc trưng chứa đựng nhiều thông tin, phù hợp cho việc so khớp các đối tượng trong một khối dữ liệu đặc trưng.
* Rất nhiều đặc trưng sẽ được phát hiện ngay cả khi chỉ có một vài đối tượng xuất hiện trong ảnh.

Các giai đoạn thực hiện đặc trưng SIFT:

* Scale – space Extrema Detection: Giai đoạn đầu tiên là tìm kiếm tính toán trên tất cả các tỷ lệ và vị trí của hình ảnh. Được triển khai bằng nhiều chức năng Gaussian khác nhau để tìm ra những đặc trưng tiềm năng mà bất biến với việc thay đổi tỉ lệ và hướng.
* Keypoint localization: Từ những điểm đặc trưng tiềm năng ở giai đoạn một sẽ được chọn lọc làm điểm đặc trưng tốt nhất (keypoints).
* Oriented assignment: Mỗi điểm đặc trưng (keypoints) sẽ được gán cho một hoặc nhiều hướng dựa trên hướng Gradient của ảnh.
* Generation of keypoint descriptors: SIFT sẽ tính keypoint descriptions bằng cách lấy 16x16 điểm liền kề của keypoint đó, rồi chia thành 16 sub-blocks với kích thước 4x4. Với mỗi sub-block, ta sẽ tạo được 8 bin oritation. Do đó tất cả sẽ có 128 bin giá trị tương ứng với 1 vector biểu hiện cho keypoint descriptor.



Hình …

### So khớp các điểm đặc trưng (Feature matching)

Sau khi các điểm đặc trưng SIFT sau khi được trích xuất ra cần phải so khớp với các đặc trưng SIFT đã được lưu trong database. Sử dụng phương phaps Brute – Force matcher để so khớp 2 ảnh với nhau. Nó lấy những mô tả đặc trưng của ảnh này tính toán khoảng cách với tất cả mô tả đặc trưng của ảnh còn lại. Kết quả trả về là những cặp đặc trưng gần nhất.

Thư viện OpenCV hỗ trợ tạo đối tượng matcher BF bằng lệnh cv2.BFMatcher(cv2.NORM\_HAMMING, crossCheck=True) với cv2.NORM\_HARMMING là phép đo khoảng cách, ngoài ra còn có các tủy chọn như cv2.NORM\_L1, cv2.NORM\_L2, … Tham số còn lại crossCheck = True sẽ trả về những giá trị phù hợp nhất (kết hợp tốt nhất) .



Hình …

Kết quả của việc so khớp giữa 2 hình ảnh sử dụng thuật toán SIFT cho ra kết quả tốt, dễ dàng áp dụng vào bài toán xác định vị trí của robot. Tuy nhiên, thời gian để đưa ra kết quả khá lâu, không phù hợp khi nhúng xuống máy tính nhúng jetson nano. Vì vậy, nhóm đã tìm hiểu thêm các thuật toán tương tự (SUFT, ORB), so sánh và lựa chọn thuật toán phù hợp nhất, được tổng hợp ở bảng sau:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Time (sec) | Kp\_img1 | Kp\_img2 | Matches | Match rate |
| SIFT | 0.2 | 499 | 481 | 363 |  |
| SURF | 0.4 | 848 | 862 | 661 |  |
| ORB |  |  |  |  |  |

# XÂY DỰNG HỆ THỐNG VẼ BẢN ĐỒ VÀ XÁC ĐỊNH VỊ TRÍ CHO ROBOT.

## …

A circuit board

Description automatically generated

HÌnh …

A screenshot of a cell phone

Description automatically generated

Hình …

**TÀI LIỆU THAM KHẢO**