**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KỸ THUẬT MÁY TÍNH**

**VĂN TRUNG DUY**

**TRẦN VÕ HẢO**

**KHÓA LUẬN TỐT NGHIỆP**

**XÂY DỰNG CHƯƠNG TRÌNH NHẬN DẠNG**

**NGÔN NGỮ KÝ HIỆU HỖ TRỢ NGƯỜI KHIẾM THÍNH**

**KỸ SƯ NGÀNH KỸ THUẬT MÁY TÍNH**

**TP. HỒ CHÍ MINH, 2015**

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KỸ THUẬT MÁY TÍNH**

**VĂN TRUNG DUY – 10520347**

**TRẦN VÕ HẢO - 10520336**

**KHÓA LUẬN TỐT NGHIỆP**

**XÂY DỰNG CHƯƠNG TRÌNH NHẬN DẠNG**

**NGÔN NGỮ KÝ HIỆU HỖ TRỢ NGƯỜI KHIẾM THÍNH**

**KỸ SƯ NGÀNH KỸ THUẬT MÁY TÍNH**

**GIẢNG VIÊN HƯỚNG DẪN**

**Ths. ĐỖ THẾ LUÂN**

**TP. HỒ CHÍ MINH, 2015**

DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo Quyết định số …………………… ngày ………………….. của Hiệu trưởng Trường Đại học Công nghệ Thông tin.

* 1. …………………………………………. – Chủ tịch.
  2. …………………………………………. – Thư ký.
  3. …………………………………………. – Ủy viên.
  4. …………………………………………. – Ủy viên.

|  |  |
| --- | --- |
| 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:** | | |
| **XÂY DỰNG CHƯƠNG TRÌNH NHẬN DẠNG NGÔN NGỮ KÝ HIỆU HỖ TRỢ NGƯỜI KHIẾM THÍNH** | | |
| **Nhóm SV thực hiện:** | | **Cán bộ hướng dẫn:** |
| Văn Trung Duy | 10520347 | Đỗ Thế Luân |
| Trần Võ Hảo | 10520336 |  |
| **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 ……… yêu cầu của một khóa luận tốt nghiệp kỹ sư/ cử nhân, xếp loại ……………  **Điểm từng sinh viên:**  Văn Trung Duy**:………../10**  Trần Võ Hảo**:………../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:** | | |
| **XÂY DỰNG CHƯƠNG TRÌNH NHẬN DẠNG NGÔN NGỮ KÝ HIỆU HỖ TRỢ NGƯỜI KHIẾM THÍNH** | | |
| **Nhóm SV thực hiện:** | | **Cán bộ phản biện :** |
| Văn Trung Duy | 10520347 |  |
| Trần Võ Hảo | 10520336 |  |
| **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 ……… yêu cầu của một khóa luận tốt nghiệp kỹ sư/ cử nhân, xếp loại ……………  **Điểm từng sinh viên:**  Văn Trung Duy**:………../10**  Trần Võ Hảo**:………../10** | | |

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

# LỜI CẢM ƠN

Để hoàn thành được khóa luận tốt nghiệp này, chúng tôi xin chân thành cảm ơn quý thầy cô Trường Đại học Công nghệ thông tin nói chung và quý thầy cô khoa Kỹ thuật máy tính nói riêng đã truyền đạt cho chúng tôi kiến thức và những kinh nghiệm quý báu trong suốt thời gian ngồi trên ghế nhà trường.

Xin gửi lời cảm ơn chân thành đến thầy ThS. Đỗ Thế Luân, người đã dành những thời gian quý báu của mình, trực tiếp hướng dẫn tận tình cho chúng tôi hoàn thành khóa luận tốt nghiệp. Bên cạnh đó còn có sự hỗ trợ sát xao của thầy Cao Văn Hưng, người đã nắm vai trò hỗ trợ chúng tôi trong đồ án 2 và giúp chúng tôi quyết định phát triển lên khóa luận tốt nghiệp.

Xin chân thành cảm ơn sự giúp đỡ của các bạn bè, anh chị, những người đã luôn động viên chúng tôi để cố gắng hết mình hoàn thành tốt khóa luận tốt nghiệp này một cách hiệu quả.

Đặc biệt, chúng con xin cảm ơn Cha Mẹ, anh chị em trong gia đình, những người đã sinh thành và nuôi dưỡng chúng con đến ngày hôm nay. Cảm ơn những lời động viên của Cha Mẹ dành cho chúng con, tạo thêm động lực để chúng con vượt qua những khó khăn gặp phải trong suốt quãng thời gian học tập cũng như trong cuộc sống.

Một lần nữa xin chân thành cảm ơn đến tất cả những người đã quan tâm đến khóa luận của chúng tôi. Tuy nhiên, trong quá trình làm việc không thể tránh khỏi những sai sót, rất mong sự đóng góp ý kiến nhiệt tình của Thầy Cô và các bạn.

Sinh viên thực hiện

**Văn Trung Duy**

**Trần Võ Hảo**

Khoa Kỹ thuật máy tính. Lớp KTMT2010

|  |  |
| --- | --- |
| 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: XÂY DỰNG CHƯƠNG TRÌNH NHẬN DẠNG NGÔN NGỮ KÝ HIỆU HỖ TRỢ NGƯỜI KHIẾM THÍNH** | | | |
| **Cán bộ hướng dẫn: ĐỖ THẾ LUÂN** | | | |
| **Thời gian thực hiện:**Từ ngày 15/09/2014 đến ngày 19/01/2015 | | | |
| **Sinh viên thực hiện:**  **VĂN TRUNG DUY - 10520347**  **TRẦN VÕ HẢO - 10520336** | | | |
| **Nội dung đề tài:** *Xây dựng được chương trình nhận dạng hỗ trợ giao tiếp với người câm điếc thông qua hình ảnh bàn tay được lấy từ camera. Chương trình viết bằng ngôn ngữ Matlab trên Desktop.* | | | |
| **Kế hoạch thực hiện** | | | |
| **Thời gian** | **Công việc** | | **Người thực hiện** |
| **15/09 – 29/09** | **Xây dựng bộ dữ liệu cho chương trình** | | **Cả hai** |
| **30/09/2014 – 30/11/2014** | **Tìm hiều các kĩ thuật xử lý hình ảnh để rút trích ra hình ảnh cần nhận dạng** | |
| **01/12/2015 – 01/01/2015** | **Tìm hiểu các thuật toán nhận dạng** | |
| **02/01 – 19/01** | **Hoàn thành ứng dụng và kiểm tra, viết báo cáo** | |
| **Xác nhận của CBHD**  (Ký tên và ghi rõ họ tên) | | **TP. HCM, ngày….tháng …..năm…..**  **Sinh viên**  (Ký tên và ghi rõ họ tên) | |

**MỤC LỤC**

[LỜI CẢM ƠN 8](#_Toc409254835)

[ĐỀ CƯƠNG CHI TIẾT 9](#_Toc409254836)

[LỜI MỞ ĐẦU v](#_Toc409254837)

[TÓM TẮT KHÓA LUẬN 1](#_Toc409254838)

[Chương 1. TỔNG QUAN ĐỀ TÀI 2](#_Toc409254839)

[**1.1.** **LÝ DO CHỌN ĐỀ TÀI** 2](#_Toc409254840)

[**1.2.** **LỊCH SỬ NGHIÊN CỨU ĐỀ TÀI** 3](#_Toc409254841)

[**1.3.** **MỤC TIÊU NGHIÊN CỨU** 4](#_Toc409254842)

[**1.3.1.** **Mục tiêu tổng quát** 4](#_Toc409254843)

[**1.3.2.** **Mục tiêu cụ thể** 5](#_Toc409254844)

[**1.4.** **KHÁCH THỂ, ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU** 5](#_Toc409254845)

[**1.4.1.** **Đối tượng nghiên cứu** 5](#_Toc409254846)

[**1.4.2.** **Phạm vi nghiên cứu** 5](#_Toc409254847)

[**1.5.** **Ý NGHĨA LÝ LUẬN VÀ THỰC TIỄN CỦA ĐỀ TÀI** 5](#_Toc409254848)

[**1.6.** **THUẬN LỢI VÀ KHÓ KHĂN** 6](#_Toc409254849)

[**1.6.1.** **Thuận lợi** 6](#_Toc409254850)

[**1.6.2.** **Khó khăn** 6](#_Toc409254851)

[Chương 2. **CƠ SỞ LÝ THUYẾT VÀ CÁC KHÁI NIỆM LIÊN QUAN** 8](#_Toc409254852)

[**2.1.** **Cơ sở lý luận** 8](#_Toc409254853)

[**2.3.1** **Lý thuyết về ngôn ngữ ký hiệu Việt Nam** 8](#_Toc409254854)

[**2.3.2** **Đặc điểm của ngôn ngữ ký hiệu** 11](#_Toc409254855)

[**2.3.3** **Ngôn ngữ ký hiệu và cuộc sống** 11](#_Toc409254856)

[**2.3.4** **Những điểm giống và khác nhau giữa Ngôn ngữ ký hiệu Việt Nam và Ngôn ngữ ký hiệu Mỹ (American Sign Laguage – ASL)** 12](#_Toc409254857)

[**2.3.5** **Bài toán phân lớp dữ liệu** 12](#_Toc409254858)

[**2.3.6** **Lý thuyết thuật toán k – Nearest Neighbors (k-NN)** 13](#_Toc409254859)

[**2.3.7** **Lý thuyết thuật toán Support Vector Machine (SVM)** 15](#_Toc409254860)

[**2.3.8** **Lý thuyết thuật toán Random Forest (RF)** 18](#_Toc409254861)

[**2.3.9** **Lý thuyết xử lý hình ảnh** 19](#_Toc409254862)

[**2.1.9.3** **Một số không gian màu** 19](#_Toc409254863)

[**2.1.9.4** **Phương pháp chuyển đổi định dạng** 26](#_Toc409254864)

[**2.2.** **Phương pháp nghiên cứu** 27](#_Toc409254865)

[**2.3.1** **Phương pháp nghiên cứu trực tiếp** 27](#_Toc409254866)

[**2.3.2** **Phương pháp nghiên cứu gián tiếp** 27](#_Toc409254867)

[**2.2.2.1** **Phương pháp thử - sai** 28](#_Toc409254868)

[**2.2.2.2** **Học máy** 29](#_Toc409254869)

[**2.3.** **Các khái niệm liên quan** 29](#_Toc409254870)

[**2.3.1** **Giới thiệu về Matlab** 29](#_Toc409254871)

[**2.3.2** **Camera Microsoft® LifeCam HD-5000** 30](#_Toc409254872)

[Chương 3: **PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG** 32](#_Toc409254873)

[**3.1.** **Mô hình tổng quan của hệ thống** 32](#_Toc409254874)

[**3.2.** **Tiến trình xử lý** 32](#_Toc409254875)

[**3.2.1.** **Tiến trình huấn luyện** 35](#_Toc409254876)

[**3.2.2.** **Phân tích chức năng tiền xử lý ảnh (Pre-processing)** 35](#_Toc409254877)

[**3.2.3.** **Tiến trình nhận dạng** 43](#_Toc409254878)

[Chương 4: **KẾT QUẢ THỬ NGHIỆM VÀ ĐÁNH GIÁ** 45](#_Toc409254879)

[**4.1.** **Qui trình thử nghiệm** 45](#_Toc409254880)

[**4.2.** **Kết quả thử nghiệm** 45](#_Toc409254881)

[**4.3.** **Kết luận, kiến nghị và hướng phát triển** 48](#_Toc409254882)

[**4.3.1.** **Kết luận** 48](#_Toc409254883)

[**4.3.2.** **Kiến nghị** 49](#_Toc409254884)

[**4.3.3.** **Hướng phát triển** 49](#_Toc409254885)

[**DANH MỤC TÀI LIỆU THAM KHẢO** 50](#_Toc409254886)

**DANH MỤC BẢNG BIỂU, HÌNH ẢNH**

**DANH MỤC HÌNH ẢNH:**

[Hình 2.1: Bảng ký tự đầy đủ Ngôn ngữ ký hiệu Việt Nam 9](#_Toc409254887)

[Hình 2.2: Siêu phẳng ngăn cách 2 lớp, H3 (màu xanh lá cây) không chia tách hai lớp dữ liệu. H1 (màu xanh lơ) phân tách hai lớp với lề nhỏ và H2 (màu đỏ) phân tách với lề cực đại. 16](#_Toc409254889)

[Hình 2.3: Siêu phẳng phân chia hai tập mẫu 17](#_Toc409254890)

[Hình 2.4: Không gian màu RGB 20](#_Toc409254891)

[Hình 2.5: Điểm ảnh được xác định tại tọa độ (2, 3) tương ứng với RGB lần lượt 0.5176 0.1608 0.0627 24](#_Toc409254892)

[Hình 2.6: Dải màu trong không gian màu YUV. 25](#_Toc409254893)

[Hình 3.1: Tổng quan về hệ thống xử lý nhận dạng ngôn ngữ ký hiệu Việt Nam 32](#_Toc409254894)

[Hình 3.2:Sơ đồ các tiến trình xử lý trong chương trình 33](#_Toc409254895)

[Hình 3.3: Mẫu ảnh của các ký tự 34](#_Toc409254896)

[Hình 3.4: Sơ đồ các bước trong tiền xử lý ảnh 35](#_Toc409254897)

[Hình 3.5:(1) Ảnh ban đầu, (2) Mô hình Kovac, (3) Không gian màu YCrCb, (4) kết hợp cả hai không gian màu 37](#_Toc409254898)

[Hình 3.6: Minh họa cách thức các phần cổ tay dư thừa 39](#_Toc409254899)

[Hình 3.7: Minh họa quá trình Thinning một ảnh nhị phân 41](#_Toc409254900)

[Hình 3.8: Ảnh chia thành 16 vùng bằng nhau 42](#_Toc409254901)

[Bảng 2.1: Ví dụ phân tích k – Nearest Neighbors 15](#_Toc409254888)

[Bảng 4.1: Kết quả kiểm tra bằng Confusion Matrix. Đơn vị: % 47](#_Toc409254905)

[Biểu đồ 4.1: Kết quả kiểm tra bằng thuật toán k-NN 46](#_Toc409254902)

[Biểu đồ 4.2: Kết quả kiểm tra bằng thuật toán SVM 46](#_Toc409254903)

[Biểu đồ 4.3: Kết quả kiểm tra bằng thuật toán Random Forest (RF) 47](#_Toc409254904)

**DANH MỤC TỪ VIẾT TẮT**

**VSL** Vietnam Sign Language

**ASL** American Sign Language

**SVM** Support Vector Machine

**k-NN** k – Nearest Neighbors

**RF** Random Forest

# LỜI MỞ ĐẦU

Một trong những phương thức phổ biến nhất mà người khiếm thính sử dụng để giao tiếp đó là ngôn ngữ ký hiệu. Một hệ thống chương trình nhận dạng ngôn ngữ ký hiệu Việt Nam sẽ hỗ trợ họ rất nhiều trong việc mở rộng ngoại giao với những người bình thường. Bài báo này mô tả chi tiết việc thiết kế hệ thống xử lý, xây dựng tập dữ liệu (Dataset), quá trình máy học (training) và nhận dạng các ký tự trong bảng chữ cái Việt Nam. Hiện tại, đây chỉ là một dự án nhỏ để hỗ trợ người khiếm thính tại Việt Nam.

Từ những thực tế nêu trên nhóm quyết định nghiên cứu tìm hiểu lý thuyết, các kỹ thuật và phương pháp để “*Xây dựng chương trình nhận dạng ngôn ngữ ký hiệu hỗ trợ người khiếm thính*”, đây là đề tài có tính ứng dụng trong thực tế rất cao. Kết quả nghiên cứu xây dựng thành công chương trình nhận dạng được các ký hiệu ký tự của ngôn ngữ ký hiệu người khiếm thính thông qua một camera thông thường kết hợp với máy tính để cho ra kết quả hiện thị dạng chữ trên màn hình.

# TÓM TẮT KHÓA LUẬN

Nội dung đề tài được trình bày trong 4 chương. Cụ thể, *Chương 1* trình bày khái quát về đề tài như lý do chọn đề tài, lịch sử nghiên cứu, mục tiêu là gì? những thuận lợi và khó khăn khi thực hiện đề tài. *Chương 2* đề cập tới các cơ sở lý thuyết liên quan tới đề tài. *Chương 3* trình bày chi tiết quá trình phân tích và xây dựng hệ thống. Kết quả và quá trình thử nghiệm hệ thống trong thực tế xem hệ thống tương tác với môi trường như thế nào sẽ được trình bày *Chương 4* ngoài ra trong chương này còn có thêm kết luận, đánh giá và một số kiến nghị sẽ được nói rõ trong phần chi tiết của đề tài.

# Chương 1. TỔNG QUAN ĐỀ TÀI

* 1. **LÝ DO CHỌN ĐỀ TÀI**

Ngôn ngữ ký hiệu là phương thức chính mà người khiếm thính dùng để truyền tải thông tin giao tiếp hàng ngày thông qua các cử chỉ, điệu bộ, ngôn ngữ hình thể và biểu hiện trên khuôn mặt thay thế từ ngữ. Không chỉ riêng Việt Nam mà hầu hết các nước trên thế giới đều có bộ ngôn ngữ ký hiệu dành riêng cho người khiếm thính. Để giúp họ giao tiếp được với nhau thì đòi hỏi mỗi người trong số họ phải biết về ngôn ngữ ký hiệu, học nó và phải áp dụng được nó.

Song song việc giao tiếp giữa cộng đồng người khiếm thính với nhau, thì một câu hỏi được đặt ra là làm thế nào người bình thường có thể giao tiếp được với người khiếm thính? Bởi lẽ người bình thường đâu phải ai cũng biết về ngôn ngữ ký hiệu, vậy làm sao có thể giúp cho người bình thường có thể hiểu được người khiếm thính muốn nói những gì? Để giải đáp được vấn đề đã nêu ra, một trong các nước tiên phong như Mỹ, Nhật, Ấn Độ, Tây Ban Nha đã thiết lặp ra một hệ thống được gọi với cái tên chung là “*hệ thống thông dịch ngôn ngữ ký hiệu*”, hệ thống này có thể giúp cho người khiếm thính và người bình thường giao tiếp dễ dàng với nhau.

Bên cạnh các nước tiên phong đi đầu trong về ngôn ngữ ký hiệu, ở Việt Nam sự phát triển về mặt ngôn ngữ ký hiệu còn rất chậm và hầu như chưa có, có chăn là một bài báo của một tác giả có đề cặp tới việc này mà nhóm chúng tôi đã nêu ở phần mở đầu, nhà nước thì chưa có chính sách tích cực trong vấn đề này. Chính vì điều đó đòi hỏi chúng ta phải nổ lực nhiều hơn nữa để có thể mang lại sự tiện nghi, xóa đi mặc cảm mà cộng đồng người khiếm thính đang phải gánh chịu. Thế nên nhóm chúng tôi chọn đề tài này với mục đích hi vọng có thể giúp ích cho họ, giúp họ có thể hòa nhập với xã hội như bao người bình thường.

Thêm vào đó, dự án này cũng có một vài đóng góp tích cực trong lĩnh vực xử lý ảnh và dịch máy. Ngoài việc hỗ trợ người khiếm thính, ngôn ngữ ký hiệu còn có vai trò rất quan trọng trong việc tương tác với người máy. Hiện tại, ngôn ngữ ký hiệu có mặt trong một số ứng dụng trong một vài lĩnh vực như robotics, video games. Vì vậy, dự án này là một công cụ có tiềm năng lớn và mang tính thương mại cao trong tương lai.

* 1. **LỊCH SỬ NGHIÊN CỨU ĐỀ TÀI**

Có rất nhiều chủ đề liên quan đề cập đến việc nhận dạng các ngôn ngữ ký hiệu trên thế giới để giúp cho việc giao tiếp giữa người khiếm thính với những người bình thường trở nên dễ dàng hơn. Một vài ngôn ngữ có ký hiệu sai khác nhau như tiếng Anh, tiếng Nhật, tiếng Tây Ban Nha, tiếng Hindi… đang được nhận dạng thông qua các hệ thống khác nhau.

Một hệ thống nhận dạng ngôn ngữ ký hiệu Mỹ do Justin K. Chen xây dựng dựa vào thiết bị Microsoft Kinect với số ký tự được đưa vào hệ thống để nhận dạng là 10. Xây dựng trên Matlab, dựa vào đặc trưng là độ sâu của ảnh, màu da tay để áp dụng vào nhận dạng và cho độ chính xác 98.20%.

Cũng công việc nhận dạng ngôn ngữ ký hiệu Mỹ, tác giả Wing Kwong Chung và cộng sự đã xây dựng hệ thống dựa vào camera thông thường, với thuật toán Haar Wavelet Decomposition trong không gian màu YIQ. Thực hiện nhận dạng thời gian thực với 15 ký tự, cho kết quả độ chính xác 94.89% và thời gian nhận dạng vào khoảng 0.4 giây cho một ký hiệu.

Một chương trình khác nhận dạng ngôn ngữ ký hiệu Ấn Độ do tác giả Padmavathi . S và cộng sự dựa trên thuật toán Neural Network. Ở đây tác giả sử dụng 2 bàn tay để thực hiện những ký tự, không sử dụng thêm bất kỳ bộ phận phần cứng nào. Công việc huấn luyện và nhận dạng thông qua camera, các đặc trưng là các góc độ của bàn tay, số lượng ngón tay sau khi hình ảnh đã lọc ra được vùng màu bàn tay. Các đặc trưng được lưu trữ như những vec-tơ. Dưới điều kiện tốt về ánh sáng và môi trường ít nhiễu, kết quả cho thấy độ chính xác khá cao, lên đến 99%.

Khác với các dự án trên Rokade Rajeshree phát triển hệ thống nhận dạng những con số thông qua camera dựa vào các phương thức Thinning. Thực hiện trong không gian màu YCrCb, các đặc trưng ảnh ở đây là Endpoints và các đường sau khi Thinning ảnh, kết hợp tính khoảng cách Horizontal để áp dụng vào quá trình nhận dạng. Trung bình độ chính xác nhận dạng được là 92.13% với thời gian trung bình nhận dạng một ký hiệu là 8.5 giây. Liên quan đến ngôn ngữ ký hiệu Mỹ, Rohit Sharma và cộng sự đã xây dựng nên một hệ thống nhận dạng 26 ký tự thông qua hai thuật toán k - Nearest Neighbors (k-NN) và Support Vector Machine (SVM). Cũng dựa vào màu da tay trong không gian màu YCrCb là chính và đặc trưng ở đây chủ yếu là đưa về dạng ảnh Binary. Nhận dạng thông qua camera thông thường cho kết quả độ chính xác 90.10%.

Hiện tại ở Việt Nam, hệ thống nhận dạng ngôn ngữ ký hiệu của Việt Nam hầu như chưa xuất hiện. Theo thông tin mà nhóm chúng tôi tìm thấy, một bài báo duy nhất có công việc liên quan. Đó là bài báo có tựa “Recognizing Postures in Vietnamese Sign Language with MEMS Accelerometers” bởi Bui, TD. Vietnam National University Hanoi City. Tuy nhiên, bài báo này tập trung chủ yếu vào việc thiết kế một chiếc găng tay với bộ cảm biến có thể nhận ra các tư thế như “không gian” và “các điểm” đại diện cho 23 ký tự trong bộ chữ cái tiếng Việt. Hơn thế nữa phương pháp này hoàn toàn khác với những gì chúng tôi thực hiện.

* 1. **MỤC TIÊU NGHIÊN CỨU**
     1. **Mục tiêu tổng quát**

Ngiên cứu và xây dựng nên một hệ thống nhận dạng ngôn ngữ ký hiệu bao gồm một camera kết nối trực tiếp với máy tính dùng để ghi nhận lại hình ảnh bàn tay đang hiện thực các ký tự trong bảng chữ cái của ngôn ngữ ký hiệu Việt Nam, sau khi qua các tiến trình xử lý đối chiếu ảnh ghi nhận được, từ đó đưa ra kết quả nhận dạng của ký tự. Qua đó hiểu được quy trình xử lý ảnh dựa trên ngôn ngữ Matlab để có kết quả đáng mong đợi đồng thời hiểu được một số thuật toán nhận dạng, máy học.

* + 1. **Mục tiêu cụ thể**

Nghiên cứu xây dựng bộ dữ liệu cho quá trình huấn luyện và kiểm tra, là các hình ảnh biểu thị các ký tự trong bảng chữ cái của ngôn ngữ ký hiệu Việt Nam.

Nghiên cứu các thuật toán xử lý hình ảnh trên ngôn ngữ Matlab được ghi nhận từ camera kết nối với máy tính thông qua cổng USB.

Nghiên cứu các thuật toán nhận dạng như k – Nearest Neighbors (k – NN), Support Vector Machine (SVM) và Random Forest (RF), qua đó chọn thuật toán tối ưu nhất để áp dụng vào đề tài.

* 1. **KHÁCH THỂ, ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU**
     1. **Đối tượng nghiên cứu**

Hệ thống nhận dạng ngôn ngữ ký hiệu Việt Nam hoàn toàn dựa trên một máy tính kết nối một camera thông qua cổng USB ghi nhận hình ảnh các cử chỉ của bàn tay.

* + 1. **Phạm vi nghiên cứu**

Thiết kế một hệ thống nhận dạng ngôn ngữ ký hiệu dựa trên nền tảng ngôn ngữ Matlab, camera sẽ ghi nhận hình ảnh bàn tay để đưa vào quá trình xử lý, trong quá trình này ảnh đầu vào sẽ được xử lý và quy về tập các vec-tơ đặc trưng, tiếp đến là quá trình nhận dạng, ở đây các vec-tơ đặc trưng này sẽ được so sánh đối chiếu với các tập vec-tơ đặc trưng đã tạo sẵn gọi là dữ liệu huấn luyện. Kết quả sau khi so sánh bằng các thuật toán nhận dạng sẽ là ký tự dạng chữ viết tương ứng với ký hiệu của bàn tay.

* 1. **Ý NGHĨA LÝ LUẬN VÀ THỰC TIỄN CỦA ĐỀ TÀI**

Việc nghiên cứu và xây dựng nên một hệ thống nhận dạng ngôn ngữ ký hiệu rất có ý nghĩa đối với cộng đồng người khiếm thính ở Việt Nam.

So với các quốc gia đi tiên phong trong lĩnh vực hỗ trợ cộng đồng người khiếm thính, vấn đề này đối với nước ta hầu như không được đề cập và ít được quan tâm nhiều tới. Việc tạo nên một ứng dụng hỗ trợ họ mang một ý nghĩa to lớn về mặt lợi ích xã hội, giúp người khiếm thính có thể tự tin hơn trong giao tiếp, từ đó dễ dàng tìm kiếm công việc, mở rộng các mối quan hệ ngoại giao như những người bình thường. Bên cạnh đó kết quả nghiên cứu khóa luận sẽ cho thấy một cái nhìn tổng quan về các lý thuyết xử lý ảnh, thuật toán nhận dạng, quy trình huấn luyện và kiểm tra diễn ra trong hệ thống. Kết quả sẽ là bước đà cho việc phát triển hệ thống trong tương lai của sinh viên khoa Kỹ thuật máy tính nói riêng và hầu hết sinh viên trong nước, người quan tâm đến vấn đề này nói chung.

* 1. **THUẬN LỢI VÀ KHÓ KHĂN**
     1. **Thuận lợi**

Trong quá trình làm luận văn chúng tôi nhận được sự động viên, giúp đỡ nhiệt tình từ người thân trong gia đình, bạn bè và sự hỗ trợ tận tình của giảng viên hướng dẫn khóa luận này. Bên cạnh đó, chúng tôi cũng nhận được sự hỗ trợ rất nhiều từ những thầy cô trong Khoa kỹ thuật máy tính như là hỗ trợ chúng tôi bằng việc cho chúng tôi mượn thiết bị camera, các thầy cô cũng cùng giáo viên hướng dẫn chỉ bảo cho chúng tôi, cung cấp tài liệu bổ ích để chúng tôi có thể hoàn thành tốt luận văn ngày hôm nay.

Sự sát cánh hợp tác phân công trong công việc là một phần quan trọng đóng góp cho sự thành công của luận văn. Mặt khác, sự chia sẽ các kiến thức cũng như kinh nghiệm của các anh đi trước cũng là động lực để chúng tôi hoàn thành tốt luận văn này.

* + 1. **Khó khăn**

Bên cạnh những thuận lợi nêu trên cũng còn có những khó khăn tồn tại nhất định. Khó khăn trong việc tìm kiếm tài liệu, các ứng dụng liên quan, tìm hiểu chọn ra thuật toán nhận dạng tốt cho đề tài, suy nghĩ tìm ra các đặc trưng của ảnh để áp dụng vào các công đoạn xử lý ảnh cũng như nhận dạng ảnh được chính xác hơn.

Từ những kiến thức lý thuyết đến việc thực hiện các sản phẩm thực tế gặp rất nhiều khó khăn, các kiến thức hầu hết là mới so với những kiến thức đã học tại trường vì vậy mà gặp những khó khăn nhất định trong quá trình làm luận văn.

Chương 2. **CƠ SỞ LÝ THUYẾT VÀ CÁC KHÁI NIỆM LIÊN QUAN**

* 1. **Cơ sở lý luận**
     1. **Lý thuyết về ngôn ngữ ký hiệu Việt Nam**

Ngôn ngữ ký hiệu là phương thức chính mà người khiếm thính dùng để truyền tải thông tin thông qua các cử chỉ, ngôn ngữ hình thể và biểu hiện trên khuôn mặt thay thế từ ngữ. Trong ngôn ngữ nói, từ ngữ được thể hiện bằng cách sử dụng lưỡi, môi và âm vực để tạo thành tiếng. Nhưng đối với những người khiếm thính (đặc biệt là những người điếc), âm thanh của lời nói họ không nghe được, và chỉ một phần nhỏ âm thanh của lời nói có thể được nhìn thấy qua môi. Ngôn ngữ ký hiệu dựa trên ý tưởng thị giác là công cụ hữu ích nhất mà một người điếc có để giao tiếp và nhận thông tin.

Ngôn ngữ ký hiệu Việt Nam (Vietnam Sign Language – VSL) là một ngôn ngữ hoàn toàn riêng biệt. Nó chứa tất cả các tính năng cơ bản của ngôn ngữ, nó có những quy định riêng của mình để phát âm, trật tự từ và ngữ pháp phức tạp. Hai đặc trưng quan trọng nhất của ngôn ngữ ký hiệu Việt Nam là tính giản lượt và sự nhấn mạnh.

*Ví dụ*:

Ngôn ngữ nói: “Anh có khỏe không ạ ?” (Are you ok ?)

VSL: “KHỎE không ?” (OK ?)

Cũng như với các ngôn ngữ khác, cách diễn đạt ý tưởng cụ thể trong VSL biến đổi nhiều như người sử dụng VSL. Ngoài những khác biệt trong cấu trúc, VSL có dấu và từ địa phương ở một số vùng miền. Cũng như các từ trong ngôn ngữ nói tiếng Anh khác nhau tại nhiều vùng khác nhau trong nước, VSL có các biến thể ở từng khu vực trong nhịp điệu của ký hiệu, hình thức và các phát âm. Dân tộc và tuổi tác là một vài yếu tố khác có ảnh hưởng đến việc sử dụng VSL và góp phần vào sự đa dạng của nó.



Hình 2.1: Bảng ký tự đầy đủ Ngôn ngữ ký hiệu Việt Nam

VSL có tất cả 23 ký hiệu đại diện cho 23 ký tự (Hình. 1.), bao gồm: a, b, c, d, đ, e, g, h, i, k, l, m, n, o, p, q, r, s, t, u, v, x, y.

Bên cạnh đó, còn có 2 ký hiệu là “dấu mũ” và “dấu râu” để biểu thị nguyên âm khác như: ă, â, ô, ê, ư, ơ. Nó thường hiện diện theo sau các ký hiệu nguyên âm tương ứng.

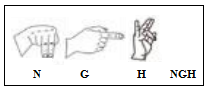
Ví dụ:

Ngôn ngữ nói: Â = A + ^

VSL:

Thêm vào đó, tiếng Việt cũng có phụ âm đôi, phụ âm ba như: ch, gh, kh, ng, ngh, nh, ph, th, tr. Biểu diễn tương tự cũng như trên.

Ví dụ:

Ngôn ngữ nói: NGH = N + G + H

VSL:

Để thực hiện các tông độ, trong VSL chúng ta có 5 ký hiệu đại diện cho 5 tông độ bao gồm: tông lên, tông xuống, tông rơi, tông uốn và tông gãy. (*Hình 2.1*)

Để truyền đạt một từ trong ngôn ngữ ký hiệu Việt Nam, người sử dụng phải sử dụng cử chỉ liên tục.

*Ví dụ*:

Ngôn ngữ nói: “TÔI LÀ NHUNG”

VSL: T + O + ^ + I L + A + ` N + H + U + N + G

* + 1. **Đặc điểm của ngôn ngữ ký hiệu**

Cũng giống như ngôn ngữ nói, ngôn ngữ ký hiệu của từng quốc gia, thậm chí từng vùng miền trong một quốc gia rất đổi khác nhau. Điều đó phụ thuộc vào mỗi quốc gia, vùng miền có lịch sử, văn hóa, tập quán khác nhau nên ký hiệu biểu thị sự vật hiện tượng cũng vì thế mà khác nhau. Ví dụ như để chỉ màu hồng thì ở Hà Nội người ta xoa xoa vào má (tức là ám chỉ má hồng), còn riêng ở trong Nam, thành phố Hồ Chí Minh lại chỉ vào bờ môi (ám chỉ môi hồng). Bên cạnh đó, khác biệt cũng trở nên lớn hơn trên tầm quốc gia, dẫn đến sự khác biệt về hệ thống từ vựng và ngữ pháp ngôn ngữ ký hiệu giữa các nước.

Song ở mọi nơi trên thế giới cũng có sự tương đồng nhất định. Ví dụ: ký hiệu “ăn cơm” thì các nước hầu như là làm như nhau là động tác giả cầm cái chén ăn cơm, ký hiệu “lái xe ô tô” thì động tác giả cầm vô lăng ô tô quay quay… Trong mỗi chúng ta, bất kể ai cũng đều có sẵn 30% kiến thức ngôn ngữ ký hiệu. Do ngôn ngữ ký hiệu phát triển hơn trong cộng đồng người khiếm thính, nên những người thuộc cộng đồng này của hai quốc gia khác nhau hầu như có thể hiểu được những gì đối phương muốn nói và vì thế họ có khả năng giao tiếp tốt hơn hai người bình thường nhưng họ không biết ngoại ngữ.

* + 1. **Ngôn ngữ ký hiệu và cuộc sống**

Ngôn ngữ ký hiệu chính là cuộc sống, bởi vì nó bắt nguồn từ cuộc sống. Mặt dù chúng ta có nhận ra điều đó hay không, thì hàng ngày chúng ta đã và đang sử dụng ngôn ngữ ký hiệu rất nhiều. Một biện pháp đơn giản nhất để chúng ta có thể nhận ra tầm quan trọng của ngôn ngữ không lời là chúng ta hãy thử nói chuyện mà nhắm mắt và hoàn toàn không cử động cơ thể, một lát sau bạn sẽ nhận ra rằng hiệu quả của cuộc trò chuyện rất thấp, bên cạnh đó khi nói chuyện cùng với sự cử động cơ thể, ánh mắt thì sẽ thu hút rất nhiều đối với đối phương đang theo dõi chúng ta, và nó cũng giúp người tiếp thu khắc sâu được nội dung cuộc trò chuyện hơn.

Để diễn đạt tính từ “to lớn” thì chúng ta dùng đôi tay khoát một vòng tròn lớn trong không gian. Để biểu thị động từ “viết” thì chúng ta cầm một cây viết và thực hiện hành động viết thì người kia sẽ hiểu ra ngay hành động của chúng ta. Để diễn đạt động từ “ngủ” thì chúng ta áp tay lên má và nhắm mắt lại, còn nếu muốn gõ cửa thì chúng ta giả bộ gõ gõ vào một cánh cửa tưởng tượng trong không khí… Vậy nên, ngôn ngữ ký hiệu cũng rất phổ biến trong cuộc sống chúng ta mà đôi khi chúng ta không nhận ra sự tồn tại của nó, nó vẫn phát triển và giúp cho cuộc sống chúng ta tiện lợi và thoải mái hơn so với sự giao tiếp khô khan bằng âm thanh là chính.

* + 1. **Những điểm giống và khác nhau giữa Ngôn ngữ ký hiệu Việt Nam và Ngôn ngữ ký hiệu Mỹ (American Sign Laguage – ASL)**

Nhìn chung, VSL có nhiều điểm tương đồng với ASL. Tuy nhiên, cũng có một vài điểm khác biệt như dưới đây:

* Tiếng Việt có phụ âm đôi và phụ âm ba. Để hiện thực phụ âm này, hai hoặc ba phụ âm sẽ kết hợp lại với nhau và thực hiện liên tục.
* Thêm vào đó, các nguyên âm như A, E, O, U cũng có sự khác biệt. Ví dụ: Ă, Â, Ê, Ô, Ơ, Ư.
* Phương thức mà VSL và ASL hiện thực các ký hiệu số hoàn toàn giống nhau.
  + 1. **Bài toán phân lớp dữ liệu**

Bài toán phân lớp một đối tượng dữ liệu vào một hay nhiều lớp cho trước nhờ vào một mô hình phân lớp mà mô hình này được xây dựng dựa trên một tập hợp các đối tượng dữ liệu đã được gán nhãn từ trước gọi là tập dữ liệu huấn luyện. Quá trình phân lớp còn được gọi là quá trình gán nhãn cho các đối tượng dữ liệu.

Như vậy, nhiệm vụ của bài toán phân lớp dữ liệu là cần xây dựng mô hình phân lớp để khi có một dữ liệu mới vào thì mô hình phân lớp sẽ cho biết dữ liệu đó thuộc lớp nào.

Có nhiều bài toán phân lớp dữ liệu, như phân lớp nhị phân, phân lớp đa lớp, phân lớp đa trị… Phân lớp nhị phân là quá trình tiến hành việc phân lớp dữ liệu vào một trong hai lớp khác nhau dựa vào việc dữ liệu đó có hay không một số đặc tính theo quy định của bộ phân lớp.

Phân lớp đa lớp là quá trình phân lớp với số lượng lớp lớn hơn hai. Như vậy, tập hợp dữ liệu trong miền xem xét được phân chia thành nhiều lớp chứ không đơn thuần chỉ là hai lớp như trong bài toán phân lớp nhị phân. Về bản chất, bài toán phân lớp nhị phân là trường hợp riêng của bài toán phân lớp đa lớp.

Trong phân lớp đa trị, mỗi đối tượng dữ liệu trong tập huấn luyện cũng như các đối tượng mới sau khi được phân lớp có thể thuộc vào từ hai lớp trở lên. Ví dụ theo như khó khăn đã nêu, thì ký tự A có thể được nhận dạng thuộc vào lớp S, T và R, nguyên nhân là do biểu hiện của ký tự có nét gần giống nhau.

* + 1. **Lý thuyết thuật toán k – Nearest Neighbors (k-NN)**

k-Nearest Neighbors algorithm (k-NN) được sử dụng rất phổ biến trong lĩnh vực Data Mining. k-NN là phương pháp để phân lớp các đối tượng dựa vào khoảng cách gần nhất giữa đối tượng cần xếp lớp với tất cả các đối tượng trong tập dữ liệu huấn luyện.

Một đối tượng được phân lớp dựa vào k láng giềng của nó. K là số nguyên dương được xác định trước khi thực hiện thuật toán. Người ta thường dùng khoảng cách Euclidean để tính khoảng cách giữa các đối tượng.

Thuật toán k-NN dùng trong phân lớp:

* Xác định giá trị tham số K (số láng giềng gần nhất).
* Tính khoảng cách giữa đối tượng cần phân lớp với tất cả các đối tượng trong tập dữ liệu huấn luyện (thường sử dụng khoảng cách Euclidean, Cosine…).
* Sắp xếp khoảng cách theo thứ tự tăng dần và xác định k láng giềng gần nhất với đối tượng cần phân lớp.
* Lấy tất cả các lớp của k láng giềng gần nhất đã xác định.
* Dựa vào phần lớn lớp của láng giềng gần nhất để xác định lớp cho đối tượng

Áp dụng cho bài toán phân lớp văn bản: Khi cần phân loại một văn bản mới, thuật toán sẽ tính khoảng cách (khoảng cách Euclidean, Cosine…) của tất cả các văn bản trong tập huấn luyện đến văn bản này để tìm ra k văn bản gần nhất (gọi là k “Neighbor”), sau đó dùng các khoảng cách này đánh trọng số cho tất cả chủ đề. Trọng số của một chủ đề chính là tổng tất cả các văn bản trong k láng giềng có cùng chủ đề, chủ đề nào không xuất hiện trong k láng giềng sẽ có trọng số bằng 0. Sau đó các chủ đề sẽ được sắp xếp theo mức độ giảm dần và các chủ đề có trọng số cao sẽ được chọn là chủ đề của văn bản cần phân loại.

Khoảng cách giữa 2 văn bản chính là độ tương tự giữa 2 văn bản đó, 2 văn bản có giá trị độ tương tự càng lớn thì khoảng cách càng gần nhau.

*Ví dụ 1* : Dùng công thức Cosine để tính độ tương tự giữa 2 văn bản:

Văn bản A: Tôi là học sinh.

Văn Bản B: Tôi là sinh viên.

Văn bản C: Tôi là giáo viên.

Biểu diễn văn bản theo dạng vector:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Tôi | là | học | sinh | viên | giáo |
| Văn bản A | 1 | 1 | 1 | 1 | 0 | 0 |
| Văn bản B | 1 | 1 | 0 | 1 | 1 | 0 |
| Văn bản C | 1 | 1 | 0 | 0 | 1 | 1 |

Bảng 2.1: Ví dụ phân tích k – Nearest Neighbors

Vector A = (1,1,1,1,0,0)

Vector B = (1,1,0,1,1,0)

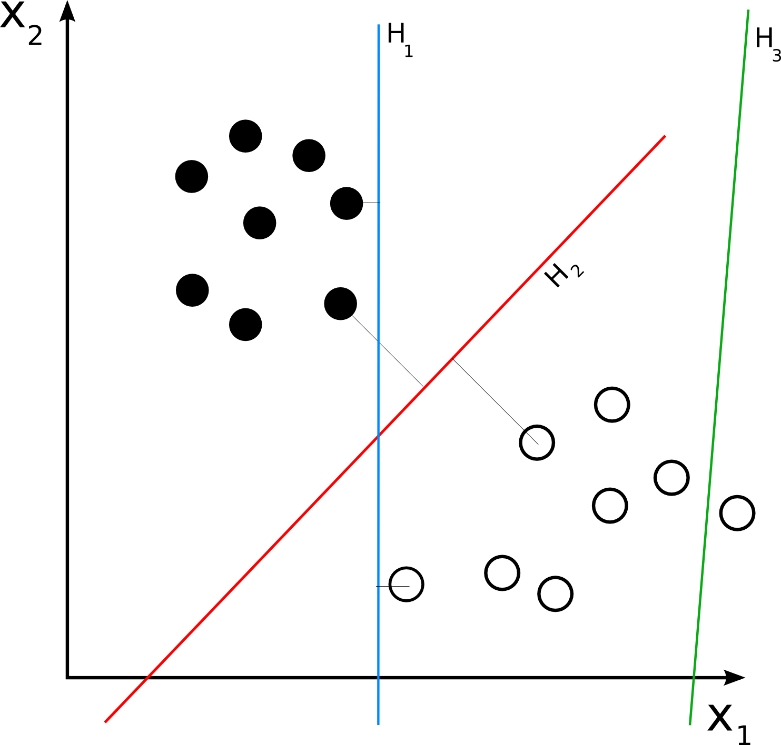
Vector C = (1,1,0,0,1,1)

Điều đó cho thấy văn bản A tương tự văn bản B hơn so với C.

* + 1. **Lý thuyết thuật toán Support Vector Machine (SVM)**

Thuật toán SVM (SVM - viết tắt 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. SVM dạng chuẩn nhận dữ liệu vào và phân loại chúng vào hai lớp khác nhau. Do đó SVM là một thuật toán phân loại nhị phân. Với một bộ các đối tượng huấn luyện thuộc hai lớp cho trước, thuật toán huấn luyện SVM xây dựng một mô hình SVM để phân loại các đối tượng khác vào hai lớp đó. Một mô hình SVM là một cách biểu diễn các điểm trong không gian và lựa chọn ranh giới giữa hai lớp sao cho khoảng cách từ các lớp trong tập huấn luyện tới ranh giới là xa nhất có thể. Các đối tượng cần kiểm tra cũng được biểu diễn trong cùng một không gian và được thuật toán dự đoán thuộc một trong hai lớp tùy thuộc vào đối tượng đó nằm ở phía nào của ranh giới.

*Đặt vấn đề*: Phân loại thống kê là một nhiệm vụ phổ biến trong máy học. Trong mô hình học có giám sát, thuật toán được cho trước một số điểm dữ liệu cùng với nhãn của chúng thuộc một trong hai lớp cho trước. Mục tiêu của thuật toán là xác định xem một điểm dữ liệu mới sẽ được thuộc về lớp nào. Mỗi điểm dữ liệu được biểu diễn dưới dạng một vector p-chiều, và ta muốn biết liệu có thể chia tách hai lớp dữ liệu bằng một siêu phẳng p − 1 chiều. Đây gọi là phân loại tuyến tính. Có nhiều siêu phẳng có thể phân loại được dữ liệu. Một lựa chọn hợp lý trong chúng là siêu phẳng có lề lớn nhất giữa hai lớp.



Hình 2.2: Siêu phẳng ngăn cách 2 lớp, H3 (màu xanh lá cây) không chia tách hai lớp dữ liệu. H1 (màu xanh lơ) phân tách hai lớp với lề nhỏ và H2 (màu đỏ) phân tách với lề cực đại.

*SVM tuyến tính:* SVM cho bài toán phân lớp tuyến tính có hình thức đơn giản cho việc phân lớp là phân lớp nhị phân: phân biệt giữa các đối tượng thuộc về một trong hai lớp: dương (+1) hoặc âm (-1). SVM sử dụng hai khái niệm để giải quyết vấn đề này là phân biệt biên rộng và hàm nhân (kernel). Ranh giới quyết định chia không gian thành hai tập tùy thuộc vào dấu của hàm f(x) =<w,x> + b. Với bất kỳ một tập dữ liệu khả tách tuyến tính có tồn tại một mặt phẳng phân lớp tất cả các điểm dữ liệu.

*SVM biên cứng*: SVM biên cứng được áp dụng đối với dữ liệu khả tách tuyến tính và nó cho kết quả phân lớp một cách chính xác với tất cả các loại dữ liệu dạng này.

Huấn luyện SVM: Giả sử tập mẫu có được gồm m phần tử là (x1,y1), (x2,y2), …, (xm, ym), trong đó xm  Rn  còn ym  {-1,1} là phân lớp của xi .

Cần xác định một siêu phẳng mà có thể tách biệt được 2 lớp trên:



Hình 2.3: Siêu phẳng phân chia hai tập mẫu

Giả sử phương trình siêu phẳng cần tìm là trong đó w là pháp vector của siêu phẳng . Ta có hai bất phương trình sau:

Lúc đó những support vector xi thỏa mãn phương trình thì nằm trên siêu phẳng H1, phương trình thì nằm trên siêu phẳng H2 , suy ra khoảng cách phân hoạch d giữa H1 và H2 là :

Do đó để có d lớn nhất thì phải nhỏ nhất hay nói cách khác phải đi tìm cực tiểu của .

Xét bài toán nhiều phân lớp với SVM: Để phân nhiều lớp thì kỹ thuật SVM nguyên thủy sẽ chia không gian dữ liệu thành 2 phần và quá trình này lặp lại nhiều lần. Khi đó hàm quyết định phân dữ liệu vào lớp thứ i của tập n, hai lớp sẽ là:

Những phần tử x là support vector sẽ thỏa điều kiện:

Như vậy, bài toán phân nhiều lớp sử dụng phương pháp SVM hoàn toàn có thể thực hiện giống như bài toán hai lớp. Bằng cách sử dụng chiến lược "một - đối - một” (one - against - one). Giả sử bài toán cần phân loại có k lớp (k > 2), chiến lược "một - đối - một” sẽ tiến hành k(k-l)/2 lần phân lớp nhị phân sử dụng phương pháp SVM. Mỗi lớp sẽ tiến hành phân tách với k -1 lớp còn lại để xác định k -1 hàm phân tách dựa vào bài toán phân hai lớp bằng phương pháp SVM.

* + 1. **Lý thuyết thuật toán Random Forest (RF)**

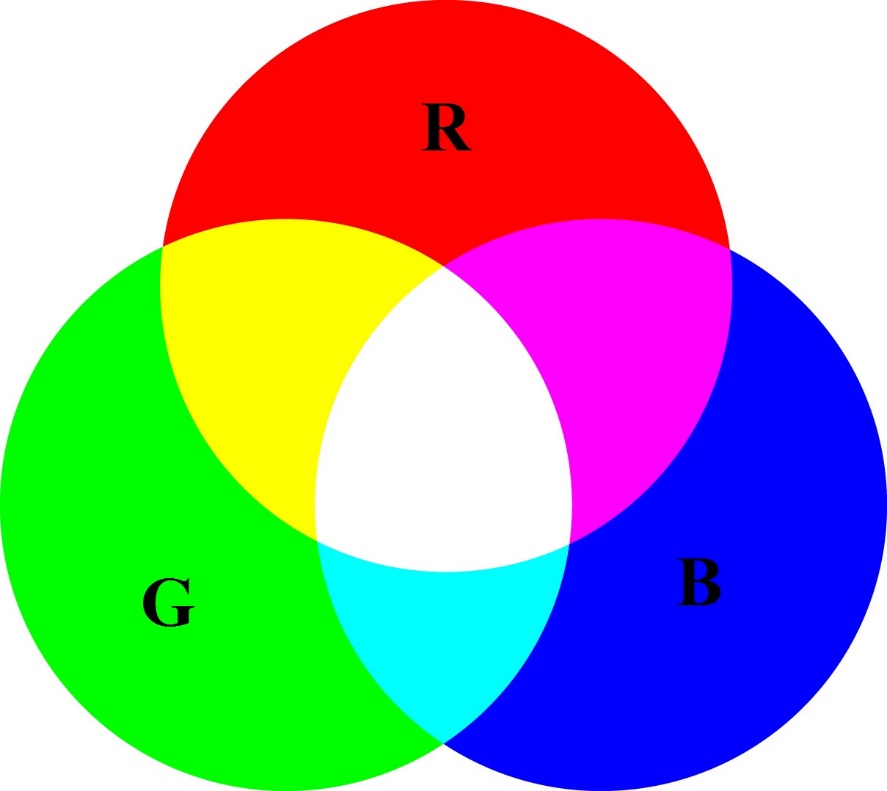
Thuật toán Random Forest là một thuật toán đặc biệt dựa trên kỹ thuật lắp ghép (ensemble techniques). Về mặt bản chất thuật toán RF được xây dựng dựa trên nền tảng thuật toán phân lớp cây quyết định CART sử dụng kỹ thuật có tên gọi là bagging. Kỹ thuật này cho phép lựa chọn một nhóm nhỏ các thuộc tính tại mỗi nút của cây để phân chia cho mức tiếp theo của cây phân lớp. Bằng cách chia nhỏ không gian tìm kiếm thành các cây nhỏ hơn như vậy cho phép thuật toán có thể phân loại một cách nhanh chóng cho dù không gian thuật toán rất lớn. Các tham số đầu vào của thuật toán khá đơn giản bao gồm số các thuộc tính được chọn trong mỗi lần phân chia. Giá trị mặc định của tham số này là căn bậc hai của p (với p là số lượng các thuộc tính). Tương tự như thuật toán cây quyết định CART, RF vẫn sử dụng công thức Gini là công thức tính toán cho việc phân chia cây. Số lượng cây được tạo ra là không hạn chế và cũng không sử dụng bất kỳ kỹ thuật để hạn chế mở rộng cây, cây được tạo ra có kích thước lớn tối đa. Chúng ta phải lựa chọn tham số cho biết số lượng cây (ntree) sẽ được sinh ra sao cho đảm bảo rằng sẽ mỗi một thuộc tính sẽ được kiểm tra một vài lần. Thuật toán sử dụng kỹ thuật OOB (out-of-bag) để xây dựng tập huấn luyện và phương pháp kiểm tra trên nó.

* + 1. **Lý thuyết xử lý hình ảnh**
       1. **Một số không gian màu**

**RGB**

Mô hình màu RGB sử dụng mô hình bổ sung trong đó ánh sáng đỏ, xanh lá cây và xanh lam được tổ hợp với nhau theo nhiều phương thức khác nhau để tạo thành các màu khác. Từ viết tắt RGB trong tiếng Anh có nghĩa là đỏ (red), xanh lá cây (green) và xanh lam (blue) là ba màu gốc trong các mô hình ánh sáng bổ sung.

Cũng lưu ý rằng mô hình màu RGB tự bản thân nó không định nghĩa thế nào là "đỏ", "xanh lá cây" và "xanh lam" một cách chính xác, vì thế với cùng các giá trị như nhau của RGB có thể mô tả các màu tương đối khác nhau trên các thiết bị khác nhau có cùng một mô hình màu. Trong khi chúng cùng chia sẻ một mô hình màu chung, không gian màu thực sự của chúng dao động một cách đáng kể.



Hình 2.4: Không gian màu RGB

Các màu gốc có liên quan đến các khái niệm sinh học hơn là vật lí, nó dựa trên cơ sở phản ứng sinh lý học của mắt người đối với ánh sáng. Mắt người có các tế bào cảm quang có hình nón nên còn được gọi là tế bào hình nón, các tế bào này thông thường có phản ứng cực đại với ánh sáng vàng - xanh lá cây (tế bào hình nón L), xanh lá cây (tế bào hình nón M) và xanh lam (tế bào hình nón S) tương ứng với các bước sóng khoảng 564 nm, 534 nm và 420 nm. Ví dụ, màu vàng thấy được khi các tế bào cảm nhận màu xanh ánh vàng được kích thích nhiều hơn một chút so với tế bào cảm nhận màu xanh lá cây và màu đỏ cảm nhận được khi các tế bào cảm nhận màu vàng - xanh lá cây được kích thích nhiều hơn so với tế bào cảm nhận màu xanh lá cây.

Mặc dù biên độ cực đại của các phản xạ của các tế bào cảm quang không diễn ra ở các bước sóng của màu "đỏ", "xanh lá cây" và "xanh lam", ba màu này được mô tả như là các màu gốc vì chúng có thể sử dụng một cách tương đối độc lập để kích thích ba loại tế bào cảm quang.

Để sinh ra khoảng màu tối ưu cho các loài động vật khác, các màu gốc khác có thể được sử dụng. Với các loài vật có bốn loại tế bào cảm quang, chẳng hạn như nhiều loại chim người ta có lẽ phải nói là cần tới bốn màu gốc; cho các loài vật chỉ có hai loại tế bào cảm quang, như phần lớn các loại động vật có vú thì chỉ cần hai màu gốc.

Một trong những ứng dụng phổ biến nhất của mô hình màu RGB là việc hiển thị màu sắc trong các ống tia âm cực, màn hình tinh thể lỏng hay màn hình plasma, chẳng hạn như màn hình máy tính hay ti vi. Mỗi điểm ảnh trên màn hình có thể được thể hiện trong bộ nhớ máy tính như là các giá trị độc lập của màu đỏ, xanh lá cây và xanh lam. Các giá trị này được chuyển đổi thành các cường độ và gửi tới màn hình. Bằng việc sử dụng các tổ hợp thích hợp của các cường độ ánh sáng đỏ, xanh lá cây và xanh lam, màn hình có thể tái tạo lại phần lớn các màu trong khoảng đen và trắng. Các phần cứng hiển thị điển hình được sử dụng cho các màn hình máy tính trong năm 2003 sử dụng tổng cộng 24-bit thông tin cho mỗi điểm ảnh (trong tiếng Anh thông thường được biết đến như *bits per pixel* hay *bpp*). Nó tương ứng với mỗi 8-bit cho màu đỏ, xanh lá cây và xanh lam, tạo thành một tổ hợp 256 các giá trị có thể, hay 256 mức cường độ cho mỗi màu. Với hệ thống như thế, khoảng 16.7 triệu màu rời rạc có thể tái tạo.

**Các kiểu biểu diễn mô hình màu RGB:**

* Biểu diễn dạng số 24-bit

Khi biểu diễn dưới dạng số, các giá trị RGB trong mô hình 24 bpp thông thường được ghi bằng cặp ba số nguyên giữa 0 và 255, mỗi số đại diện cho cường độ của màu đỏ, xanh lá cây, xanh lam trong trật tự như thế. Ví dụ:

* (0, 0, 0) là màu đen.
* (255, 255, 255) là màu trắng.
* (255, 0, 0) là màu đỏ.
* (0, 255, 0) là màu xanh lá cây.
* (0, 0, 255) là màu xanh lam.
* (255, 255, 0) là màu vàng.
* (0, 255, 255) là màu xanh ngọc.
* (255, 0, 255) là màu hồng sẫm.

Định nghĩa trên sử dụng thỏa thuận được biết đến như là *toàn bộ khoảng RGB*. Thông thường, RGB cho video kỹ thuật số không phải là toàn bộ khoảng này. Thay vì thế video RGB sử dụng thỏa thuận với thang độ và các giá trị tương đối chẳng hạn như (16, 16, 16) là màu đen, (235, 235, 235) là màu trắng v.v. Ví dụ, các thang độ và giá trị tương đối này được sử dụng cho định nghĩa RGB kỹ thuật số trong CCIR 601.

* Kiểu 16-bit

Còn có kiểu 16 bpp, trong đó hoặc là có 5 bit cho mỗi màu gọi là kiểu 555 hay một bit còn lại cho màu xanh lá cây (vì mắt có thể cảm nhận màu này tốt hơn so với các màu khác) gọi là kiểu 565. Kiểu 24 bpp nói chung được gọi là thật màu, trong khi kiểu 16 bpp được gọi là cao màu.

* Kiểu 32 bit

Cái gọi là kiểu *32 bpp* phần lớn là sự đồng nhất chính xác với kiểu 24 bpp, do ở đây thực sự cũng chỉ có 8 bit cho mỗi màu thành phần, 8 bit dư đơn giản là không sử dụng (ngoại trừ khả năng sử dụng như là kênh alpha). Lý do của việc mở rộng của kiểu 32 bpp là vận tốc cao hơn mà phần lớn các phần cứng ngày nay có thể truy cập các dữ liệu được sắp xếp trong các địa chỉ byte có thể chia được ngang nhau theo cấp số của 2, so với các dữ liệu không được sắp xếp như vậy.

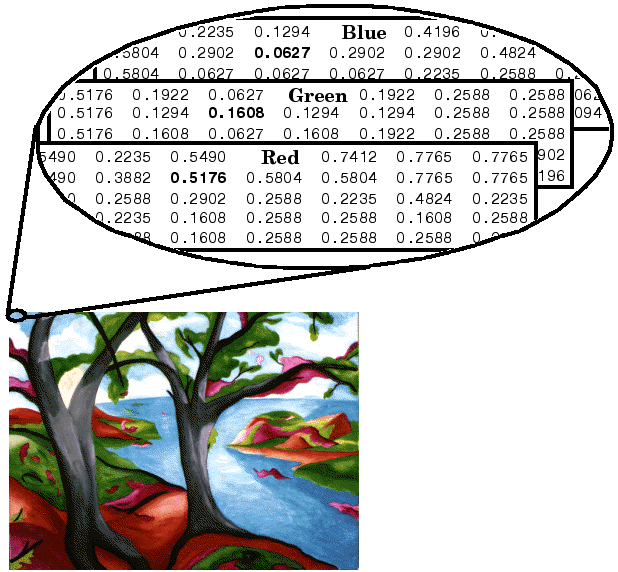
* Kiểu 48 bit

"Kiểu 16-bit" cũng có thể để chỉ tới 16 bit cho mỗi màu thành phần, tạo ra trong kiểu 48 bpp. Kiểu này làm cho nó có khả năng biểu thị 65.535 sắc thái mỗi màu thành phần thay vì chỉ có 255. Nó đầu tiên được sử dụng trong chỉnh sửa hình ảnh chuyên nghiệp, như Photoshop của Adobe để duy trì sự chính xác cao hơn khi có hơn một thuật toán lọc hình ảnh được sử dụng đối với hình ảnh đó. Với chỉ có 8 bit cho mỗi màu, các sai số làm tròn có xu hướng tích lũy sau mỗi thuật toán lọc hình ảnh được sử dụng và làm biến dạng kết quả cuối cùng.

* RGBA

Với nhu cầu về các hình ảnh ghép đã xuất hiện phương án của RGB trong đó thêm vào kênh 8 bit dư cho độ trong suốt, vì thế tạo ra định dạng 32 bpp. Kênh trong suốt được biết đến phổ biến hơn như là *kênh alpha*, vì thế định dạng này có tên là RGBA. Cũng lưu ý rằng vì nó không thay đổi bất kì cái gì trong mô hình RGB, nên RGBA không phải là một mô hình màu khác biệt, nó chỉ là định dạng tệp (file) trong đó bổ sung thêm thông tin về độ trong suốt cùng với thông tin về màu trong cùng một tệp.

**RGB trong Matlab:** Một ảnh RGB trong Matlab có thể là một mảng của lớp double, uint8 hoặc uint16. Một điểm ảnh (pixel) có thành phần (0, 0, 0) được hiển thị như là màu đen, và một điểm ảnh (1, 1, 1) được hiển thị như màu trắng. Ba thành phần đơn sắc cho mỗi màu tại một điểm ảnh được lưu trữ như một vec-tơ 3D trong không gian màu RGB với các thành phần (x, y, z) tương ứng (R, G, B). Ví dụ, các thành phần màu đỏ, xanh lá cây và màu xanh của các điểm ảnh (10, 5) được lưu trữ tương ứng trong RGB (10, 5, 1), RGB (10, 5, 2) và RGB (10, 5, 3).

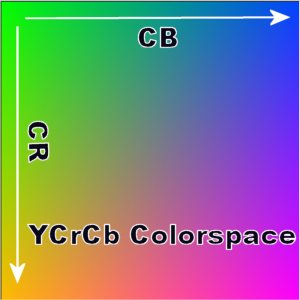


Hình 2.5: Điểm ảnh được xác định tại tọa độ (2, 3) tương ứng với RGB lần lượt 0.5176 0.1608 0.0627

**YUV**

Mô hình ***YUV*** qui định một không gian màu được tạo bởi một độ sáng và hai thành phần màu (*chrominance*). YUV được sử dụng trong hệ thống phát sóng truyền hình theo chuẩn PAL[[1]](#footnote-1), đây là chuẩn ở phần lớn các nước. Mô hình YUV giúp tạo ra màu đúng với nhận thức của con người hơn chuẩn RGB, là loại được dùng trong các thiết bị đồ hoạ máy tính.

Y đại diện cho thành phần độ sáng, U và V là đại diện cho các thành phần màu. Không gian màu YCbCr hay YPbPr, được sử dụng trong các thiết bị phát hình, đều xuất phát từ nó (Cb/Pb và Cr/Pr là những phiên bản biến thể của U và V), và đôi khi bị gọi một cách không chính xác là "YUV".



Hình 2.6: Dải màu trong không gian màu YUV.

Thuận lợi chính của hệ thống luminance/chrominance như ở trong YUV và các họ hàng của nó, YIQ và YDbDr, là ở chỗ chúng vẫn tương thích với hệ màu đen trắng của tivi tương tự. Tín hiệu Y về cơ bản giống với tín hiệu được truyền từ một máy thu hình trắng đen bình thường (với một ít thay đổi không đáng kể), và các tín hiệu U và V có thể được bỏ qua. Khi được dùng trong một thiết lập màu thì quá trình trừ đi được bảo toàn, kết quả là không gian màu gốc RGB.

Một lợi điểm khác là tín hiệu trong YUV có thể dễ dàng được xử lí để có thể loại bỏ bớt một số thông tin để giảm băng thông (*bandwidth*). Mắt con người thực sự có độ phân giải màu khá thấp: các ảnh màu có độ phân giải cao mà chúng ta thấy đều được xử lí bởi hệ thống hình ảnh (*visual system*) bằng cách kết hợp ảnh đen và trắng có độ phân giải cao và ảnh màu với độ phân giải thấp. Lợi dụng điểm này, các chuẩn như NTSC[[2]](#footnote-2) làm giảm lượng thông tin trong phần màu (*chrominance*) một cách đáng kể, để cho mắt người tự kết hợp chúng lại. Chẳng hạn, NTSC chỉ lưu lại 11% của màu xanh gốc và 30% của màu đỏ gốc, loại bỏ phần còn lại. Vì màu xanh đã được mã hoá trong tín hiệu Y, kết quả của tín hiệu U và V là khá nhỏ hơn so với tín hiệu RGB hay YUV được gởi đi. Việc lọc bỏ các tín hiệu xanh (*blue*) và đỏ (*red*) là không cần thiết nếu tín hiệu là ở định dạng YUV.

* + - 1. **Phương pháp chuyển đổi định dạng**

Việc chuyển đổi giữa các hệ màu thông thường được thực hiện thông qua các phép biến đổi ma trận. Để thực hiện chuyển từ RGB sang YCrCb ta áp dụng công thức tính sau:

Hoặc ở dạng ma trận như sau:

Chuyển ngược lại từ YCrCb sang RGB theo công thức:

* 1. **Phương pháp nghiên cứu**
     1. **Phương pháp nghiên cứu trực tiếp**

Đặc điểm của phương pháp này chính là xác định trực tiếp được lời giải qua một thủ tục tính toán (công thức, hệ luật, định luật…) hoặc qua các bước căn bản để có được lời giải. Ngoài ra, việc giải quyết vấn đề trên máy tính chỉ là thao tác lập trình hay là sự chuyển đổi lời giải từ ngôn ngữ bên ngoài sang các ngôn ngữ được sử dụng trong máy tính. Ở đây, chúng tôi đã sử dụng ngôn ngữ lập trình Matlab để thực hiện đề tài này.

Các nguyên lý được áp dụng trong phương pháp[[3]](#footnote-3) này được kể ra như sau:

*Nguyên lý chuyển đổi dữ liệu bài toán thành dữ liệu của chương trình*: Dữ liệu của bài toán sẽ được biểu diễn lại dưới dạng các biến của chương trình thông qua các quy tắc xác định của ngôn ngữ lập trình cụ thể.

*Nguyên lý chuyển đổi quá trình tính toán của bài toán thành các cấu trúc của chương trình*: Mọi quá trình tính toán đều có thể mô tả và thực hiện dựa trên ba cấu trúc cơ bản đó là cấu trúc tuần tự, cấu trúc rẽ nhánh và cấu trúc lặp.

*Nguyên lý phân chia bài toán ban đầu thành các bài toán nhỏ hơn*: Mọi vấn đề bài toán đều có thể giải quyết bằng các phân chia thành những vấn đề bài toán nhỏ hơn.

* + 1. **Phương pháp nghiên cứu gián tiếp**

*Phương pháp tiếp cận thu thập thông tin:* Tiếp cận là sự lựa chọn chỗ đứng để quan sát đối tượng nghiên cứu, là sự khởi đầu của quá trình tiếp xúc với đối tượng nghiên cứu, là sự đi trước của tư duy trước khi bắt tay thực hiện những thao tác cụ thể của quá trình thu thập thông tin. Tiếp cận bao gồm: Tiếp cận hệ thống có cấu trúc; Tiếp cận định tính và định lượng; Tiếp cận tất nhiên và ngẫu nhiên; Tiếp cận lịch sử và logic; Tiếp cận cá biệt và so sánh; Tiếp cận phân tích và tổng hợp.

*Phương pháp nghiên cứu tài liệu:* Mục đích nghiên cứu tài liệu là tìm hiểu lịch sử nghiên cứu, nắm bắt những nội dung đồng nghiệp đi trước đã làm, không mất thời gian lặp lại những công việc mà đồng nghiệp đi trước đã thực hiện. Nội dung phân tích có thể bao gồm: Phân tích nguồn, phân tích tác giả, phân tích nội dung và tổng hợp tài liệu.

*Phương* pháp *thực nghiệm:* Thực nghiệm là phương pháp thu thập thông tin bằng cách quan sát trong điều kiện có gây biến đổi đối tượng khảo sát một cách chủ định. Bằng cách thay đổi tham số, người nghiên cứu có thể thu được những kết quả mong muốn, như: Tách riêng từng phần thuần nhất của đối tượng nghiên cứu để quan sát; Biến đổi các điều kiện tồn tại của đối tượng nghiên cứu; Rút ngắn được thời gian tiếp cận trong quan sát; Tiến hành những thực nghiệm lặp lại nhiều lần để kiểm tra lẫn nhau; Không bị hạn chế về không gian và thời gian. Các phương pháp thực nghiệm: Thử và sai; Phương pháp Ơristic; Phương pháp tương tự (nghiên cứu trên các mô hình thí điểm).

* + - 1. **Phương pháp thử - sai**

Các nguyên lý áp dụng trong phương pháp này bao gồm:

*Nguyên lý vét cạn* toàn *bộ*: Đây là phương pháp đơn giản nhất, liệt kê tất cả các trường hợp có thể xảy ra.

*Nguyên lý mắt* lưới: Vận dụng nguyên lý này để sàn lọc ra những trường hợp nào phù hợp với các điều kiện đưa ra, giống như lưới bắt cá chỉ bắt được những con các có kích thước lớn hơn kích thước của mắt lưới.

*Nguyên lý giảm* độ *phức tạp của thử và sai*: Thu hẹp trường hợp trước và trong khi duyệt, đồng thời đơn giản hóa tối đa điều kiện chấp nhận một trường hợp.

*Nguyên lý thu gọn không gian tìm kiếm*: Loại bỏ những trường hợp hoặc nhóm trường hợp chắc chắn không dẫn đến lời giải.

* + - 1. **Học máy**

Học máy, có tài liệu gọi là Máy học, (tiếng Anh: *machine learning*) là một lĩnh vực của trí tuệ nhân tạo liên quan đến việc nghiên cứu và xây dựng các kĩ thuật cho phép các hệ thống "học" tự động từ dữ liệu để giải quyết những vấn đề cụ thể. Học máy rất gần với suy diễn thống kê (statistical inference) tuy có khác nhau về thuật ngữ. Dưới góc nhìn của trí tuệ nhân tạo, động lực chính học máy bởi là nhu cầu thu nhận tri thức (knowledge acquisition).

Có rất nhiều thuật toán để thực hiện phương pháp này như mạng nơ-ron nhân tạo, phương pháp biến thiên,… Trong chương trình này, chúng tôi quyết định sử dụng ba thuật toán k – Nearest Neighbors (k-NN), Support Vector Machine (SVM) và Random Forest (RF).

* 1. **Các khái niệm liên quan**
     1. **Giới thiệu về Matlab**

Matlab là một ngôn ngữ lập trình thực hành bậc cao được sử dụng để giải các bài toán về kỹ thuật. Matlab tích hợp được việc tính toán, thể hiện kết quả, cho phép lập trình, giao diện làm việc rất dễ dàng cho người sử dụng. Dữ liệu cùng với thư viện được lập trình sẵn cho phép người sử dụng có thể có được những ứng dụng sau đây.

Sử dụng các hàm có sẵn trong thư viện, các phép tính toán học thông thường.

* Cho phép lập trình tạo ra những ứng dụng mới.
* Cho phép mô phỏng các mô hình thực tế.
* Phân tích, khảo sát và hiển thị dữ liệu.
* Với phần mềm đồ hoạ cực mạnh.
* Cho phép phát triển, giao tiếp với một số phần mềm khác như C++, Fortran.

Matlab là một hệ thống tương giao, các phần tử dữ liệu là một mảng (mảng này không đòi hỏi về kích thước). Chúng cho phép giải quyết các vấn đề liên quan đến lập trình bằng máy tính, đặc biệt sử dụng các phép tính về ma trận hay vectơ và có thể sử dụng ngôn ngữ C học Fortran lập trình rồi thực hiện ứng dụng lập trình đó bằng các câu lệnh gọi từ Matlab. Matlab được viết tắt từ chữ “MATrix LABoratory” tức là thư viện về ma trận, từ đó phần mềm Matlab được viết nhằm cung cấp cho việc truy cập vào phần mềm ma trận một cách dễ dàng, phần mềm ma trận này được phát triển bởi các công trình Linpack và Eispack. Ngày nay Matlab được phát triển bởi Lapack và Artpack tạo nên một nghệ thuật phần mềm cho ma trận.

Dữ liệu của Matlab thể hiện dưới dạng ma trận (hoặc mảng - tổng quát), và có các kiểu dữ liệu được liệt kê sau đây:

* Kiểu đơn single, kiểu này có lợi về bộ nhớ dữ liệu vì nó đòi hỏi ít byte nhớ hơn, kiểu dữ liệu này không được sử dụng trong các phép tính toán học, độ chính xác kém hơn.
* Kiểu double kiểu này là kiểu thông dụng nhất của các biến trong Matlab.
* Kiểu Sparse.
* Kiểu uint8, uint8, uint16, uint64...
* Kiểu char ví dụ “Hello”.
* Kiểu cell.
* Kiểu Structure.

Trong Matlab kiểu dữ liệu double là kiểu mặc định sử dụng trong các phép tính số học.

* + 1. **Camera Microsoft® LifeCam HD-5000**

Được dùng để thu nhận hình ảnh cho quá trình huấn luyện và thành phần tiếp nhận hình ảnh trong quá trình nhận dạng. Dưới đây là một vài thông số kỹ thuật:

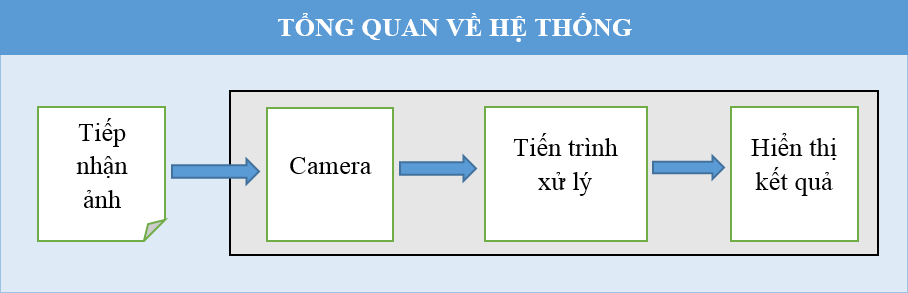
* Công nghệ cảm biến CMOS, độ phân giải 1280x720 pixel giao tiếp thông qua cổng USB.
* Có tất cả 11 khung ảnh hỗ trợ, trong đó có các khung ảnh thông dụng như: 1280x720, 640x480, 320x240.
* Chế độ tự động lấy nét, hỗ trợ 30 khung hình trên giây.

Chương 3: **PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

* 1. **Mô hình tổng quan của hệ thống**

Mục đích chính của đề tài là xây dựng chương trình nhận dạng các ký tự ký hiệu trong bảng chữ cái của Ngôn ngữ ký hiệu Việt Nam bằng một chương trình được xây dựng trên máy tính, cụ thể ở đây dựa trên ngôn ngữ lập trình Matlab để xây dựng, và thông qua một Camera kết nối ngoài để tiếp nhận các thông tin về hình ảnh. Nhằm phục vụ cho quá trình huấn luyện và nhận dạng.

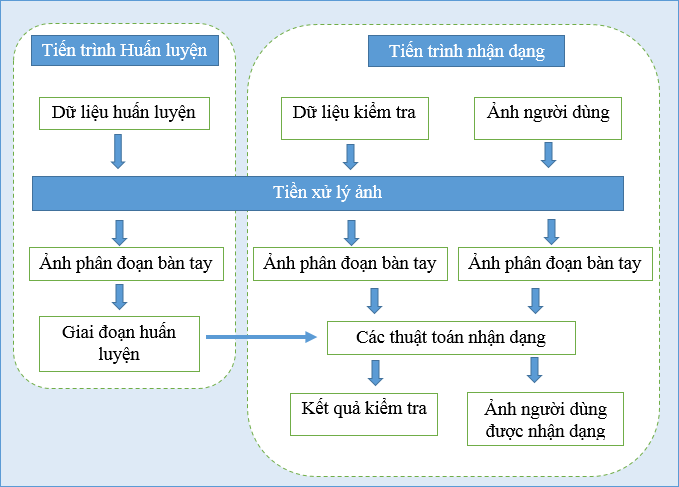
Nhìn chung, hệ thống bao gồm hai thành phần chính là một camera và một bộ máy tính để xử lý. Đầu tiên, người sử dụng thực hiện các ký hiệu ngôn ngữ trước camera. Sau đó, camera sẽ tiếp nhận hình ảnh và máy tính sẽ xử lý và cho ra kết quả nhận dạng hiển thị ra màn hình.



Hình 3.1: Tổng quan về hệ thống xử lý nhận dạng ngôn ngữ ký hiệu Việt Nam

* 1. **Tiến trình xử lý**

Một tập dữ liệu để phục vụ cho hệ thống thử nghiệm ban đầu được xây dựng. Chúng tôi chụp lại toàn bộ ký hiệu chữ cái trong bảng ký hiệu VSL bao gồm 23 ký hiệu chữ cái, 2 ký hiệu là “dấu râu” và “dấu mũ”, và thêm 1 ký hiệu “W” bằng camera Microsoft® LifeCam HD-5000. Microsoft® LifeCam HD-5000 được sử dụng để chụp lại các tư thế của bàn tay thực hiện ngôn ngữ ký hiệu với độ phân giải 640x480 pixel. Hình ảnh được chụp lại trên nền cố định, ở đây chúng tôi sử dụng nền xanh, tuy nhiên chúng ta có thể sử dụng các nền khác nhau nhưng phải đồng bộ toàn chương trình bằng một nền cố định. Với từng mẫu ký hiệu, chúng tôi viết một chương trình: chụp lại một ảnh cứ sau 2 giây và liên tiếp 50 ảnh với 4 người khác nhau. Tiếp theo, chúng tôi trộn lẫn những hình ảnh này lại với nhau theo từng mẫu. Và sau khi sàng lọc, chúng tôi có một tập dữ liệu với 26 ký hiệu và mỗi ký hiệu có 50 ảnh, phân chia thành hai nhóm: một nhóm 40 ảnh phục vụ cho quá trình huấn luyện, và nhóm 10 ảnh còn lại để thực hiện kiểm tra thử nghiệm trước khi hiện thực công việc kiểm tra thời gian thực bằng camera.



Hình 3.2:Sơ đồ các tiến trình xử lý trong chương trình



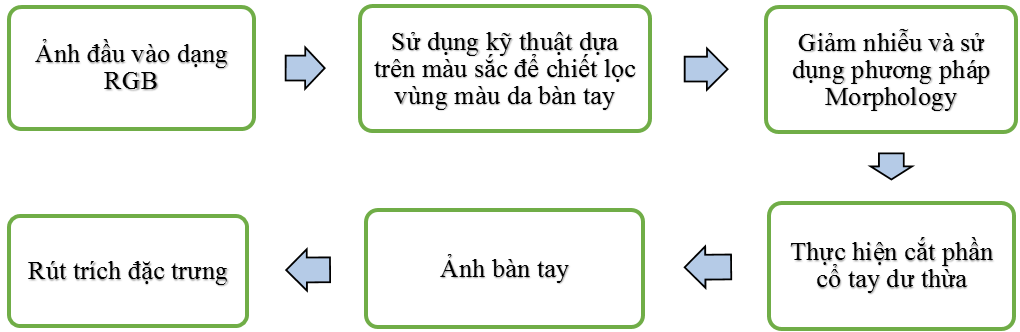
Hình 3.3: Mẫu ảnh của các ký tự

* + 1. **Tiến trình huấn luyện**

Ban đầu, toàn bộ ảnh huấn luyện sẽ được đưa vào hệ thống. Sau đó, công việc đầu tiên chúng tôi sẽ thực hiện tiền xử lý ảnh (Pre-processing) tất cả các ảnh huấn luyện này. Kết quả của bước tiền xử lý này là tập dữ liệu các con số từ việc rút trích đặc trưng của ảnh. Ở giai đoạn này, các thuật toán chỉ thực hiện lưu trữ các vec-tơ đặc trưng và gán nhãn cho các mẫu ảnh huấn luyện.

* + 1. **Phân tích chức năng tiền xử lý ảnh (Pre-processing)**

Mục tiêu của quá trình tiền xử lý ảnh là từ ảnh ban đầu được tiếp nhận từ camera, sẽ phân tích, giảm nhiễu và rút trích được những đặc trưng cần thiết từ ảnh. Dưới đây là các bước trong quá trình tiền xử lý ảnh:



Hình 3.4: Sơ đồ các bước trong tiền xử lý ảnh

Để chiết xuất được vùng màu da tay, hệ thống phải thực hiện việc tìm kiếm màu da. Thực chất quá trình này cũng tương tự với quá trình nhị phân hóa [[4]](#footnote-4), nhưng ở đây chúng tôi nhị phân hóa trực tiếp từ ba thông số R (đỏ), G (xanh lá), B (xanh dương) của ảnh. Sở dĩ không sử dụng nhị phân hóa theo một con số ngưỡng (T) cho trước là do quá trình nhị phân hóa này bị ảnh hưởng bởi ánh sáng nhiều hơn và không có một con số cố định cho đa phần môi trường khác nhau. Theo Barkoky, RGB là không gian màu thường được sử dụng nhất để lưu trữ và đại diện cho hình ảnh kỹ thuật số. Tuy nhiên, nó vẫn tồn tại một số yếu điểm, vì vậy chúng tôi sử dụng mô hình Kovac để xác định vùng màu da trong không gian màu RGB và so sánh với màu da từ không gian màu YCrCb để giảm những thành phần dư thừa trong các kênh màu của RGB.

Mô hình Kovac được áp dụng trong chương trình mô tả như sau:

(1)

Những con số trong mô hình Kovac được rút trích từ việc kiểm thử nhiều lần việc tìm kiếm màu da tay, để có thể từ ảnh màu RGB ban đầu, chúng tôi có thể chuyển sang ảnh nhị phân (trắng – đen) với màu da tay được xác định trong ảnh nhị phân ở đây là màu trắng.

Theo Kukharev, liên quan đến cụm màu da trong không gian màu YCrCb được cho:

(2)

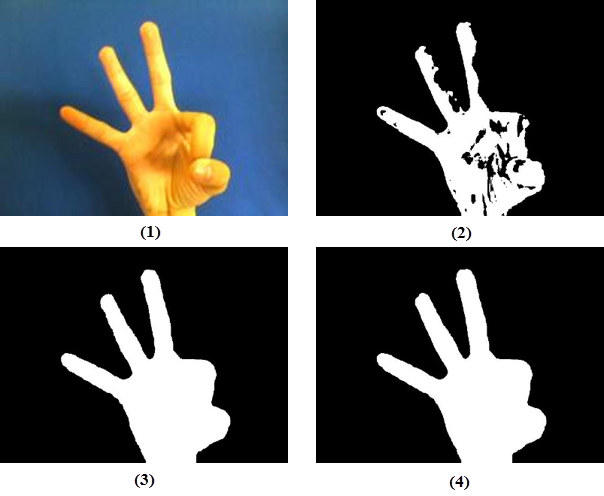
Y>80 và 135 < Cr < 180 và 85 < Cb < 135

Tuy nhiên, trong môi trường và sau khi phân tích chúng tôi rút ra được những con số khác để phù hợp hơn với hệ thống chương trình như sau:

(3)

Y>30 và 40 < Cr < 250 và 20 < Cb < 100

Với mục đích tìm kiếm màu da tay từ nhiều người khác nhau. Chúng tôi quyết định sử dụng kết hợp tất cả các ngưỡng màu (1) và (4) lại với nhau để đảm bảo chương trình có thể kiểm tra trên nhiều màu da tay của con người Việt Nam hơn.

****

Hình 3.5:(1) Ảnh ban đầu, (2) Mô hình Kovac, (3) Không gian màu YCrCb, (4) kết hợp cả hai không gian màu

Sau khi rút được vùng màu da tay, chúng tôi sử dụng phương pháp Morphology và bộ lọc để giảm đi các vùng nhiễu. Dưới đây là đoạn code hiện thực công việc lọc nhiễu:

for i=2:nRows-1,

for j=1:nColumns,

if ThresImage(i-1,j,1)==0 && ThresImage(i+1,j,1)==0,

ThresImage(i,j,1)=0;

end;

if ThresImage(i-1,j,1)==255 && ThresImage(i+1,j,1)==255,

ThresImage(i,j,1)=255;

end;

end;

end;

for i=1:nRows,

for j=2:nColumns - 1,

if ThresImage(i,j-1,1)==0 && ThresImage(i,j+1,1)==0,

ThresImage(i,j,1)=0;

end;

if ThresImage(i,j-1,1)==255 && ThresImage(i,j+1,1)==255,

ThresImage(i,j,1)=255;

end;

end;

end;

ThresImage = imfill(ThresImage,'holes');

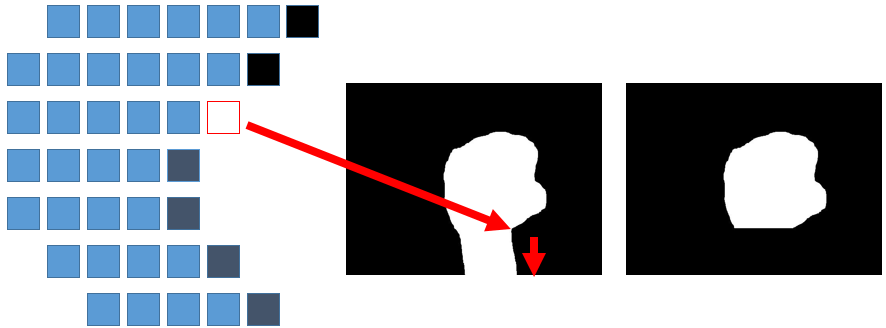
Mask = ones(5,5);

ThresImage = imdilate(ThresImage,Mask);%Get borders of skin regions

Ta dễ dàng thấy được, xét 2 điểm có tọa độ (i, j), (i, j+1) và điểm ảnh này có giá trị là 0 (màu đen) hay là 255 (màu trắng) dựa vào cách xác định sau:

Vì lý do một số ảnh có thể sẽ bị nhiễu do ánh sáng tác động đến vùng màu da tay có thể gây mất một số chi tiết, nên chúng tôi tạo ra một mặt nạ có giá trị là một ma trận 5x5 điền đầy giá trị 1 (tức một ảnh có kích thước 5x5 màu trắng) để lấp vào toàn bộ ảnh sau khi lọc nhiễu.

Bước kế tiếp trong quá trình tiền xử lý ảnh, chúng tôi các bỏ đi phần dư thừa sau khi đã chiết xuất được vùng màu da tay đó là cổ tay. Mục đích là giảm thiểu những điểm ảnh không cần thiết và giảm thời gian xử lý cho các bước tiếp theo của quá trình tiền xử lý ảnh sau này.



Hình 3.6: Minh họa cách thức các phần cổ tay dư thừa

Điểm ảnh cổ tay được xác định dựa vào cách thức như sau: đầu tiên, chúng tôi xét từng điểm một dọc biên bên phải của ảnh từ dưới đi lên, khi phát hiện thấy một điểm ảnh màu trắng nằm bên phải phía trên điểm ảnh đang được xét đến thì chúng tôi tiếp tục xét ba điểm bên phải liên tiếp lần lượt lên trên như *hình 3.5* (điểm được xét là hai chấm đen). Có hai trường hợp:

* Nếu một trong ba điểm ảnh được xét không mang giá trị là 255 thì chúng tôi tiếp tục xét điểm kế tiếp phía trên.
* Nếu cả ba điểm được xét có giá trị là 255 thì chúng tôi kết luận đây chính là điểm ảnh cổ tay cần xác định. Và chúng tôi thực hiện xóa đi vùng từ hàng của điểm cổ tay xuống.
* Trong trường hợp không xác định được điểm ảnh cổ tay thì chúng tôi lấy lại ảnh nhị phân.

Tiếp theo, chúng tôi tiến hành rút trích đặc trưng, dưới đây là quá trình thining ảnh để rút trích ra được số điểm endpoints và số điểm branchpoints.

Phương pháp Morphology được áp dụng ở đây là Thinning, một hình thái hoạt động được sử dụng để loại bỏ các điểm ảnh cận cảnh từ ảnh nhị phân, có phần giống như xói mòn (erosion). Nó có thể được sử dụng cho nhiều ứng dụng, nhưng là đặc biệt hữu ích cho việc tạo ra bộ khung xương (skeletonization) từ ảnh nhị phân. Trong chế độ này, nó thường được sử dụng để dọn dẹp các đầu ra của các cạnh bằng cách giảm tất cả các dòng với chiều dày điểm ảnh. Mục đích chính là giảm ngưỡng đầu ra của một cạnh, để lại một đường duy nhất từ một đường có độ dày nhỏ hơn nhưng vẫn bảo toàn chiều dài của các đường này (tức là các điểm ảnh ở hai đầu cực của đường không bị ảnh hưởng. Một thuật toán đơn giản để làm điều này được trình bày như bên dưới đây:

* Trường hợp 1, xóa điểm ảnh p nếu và chỉ nếu thỏa điều kiện G1, G2 và G3.
* Trường hợp 2, xóa điểm ảnh p nếu và chỉ nếu thỏa điều kiện G1, G2 và G3’.

***Điều kiện G1:***

Khi

Với

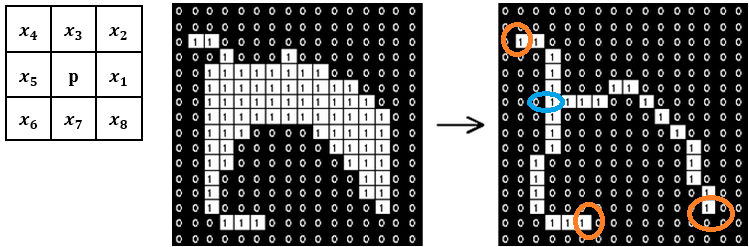
là giá trị lần lượt của 8 điểm xung quanh điểm p đang xét, bắt đầu từ điểm gần nhất phía đông và được đánh số theo chiều kim đồng hồ.

***Điều*** kiện ***G2:***

Khi

***Điều kiện G3:***

***Điều*** kiện ***G3’:***

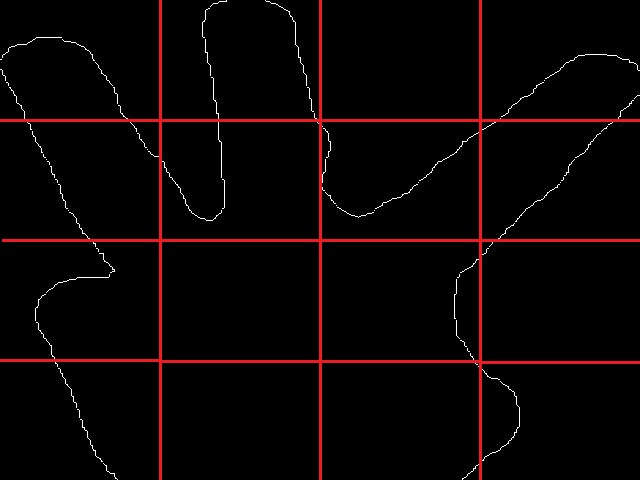


Hình 3.7: Minh họa quá trình Thinning một ảnh nhị phân

Từ *hình 3.6* ta cũng dễ dàng thấy được các điểm khoanh vùng màu đỏ sẽ là các điểm Endpoints và màu xanh sẽ là các điểm Branchpoints. Đó chính là hai trong tổng số 6 đặc trưng của một ảnh. Chúng tôi tiến hành rút trích thêm các đặc trưng khác như: chiều cao, chiều rộng, số điểm ảnh (pixel) trắng và số histogram, và chia ảnh thành nhiều vùng (zoning).

Trước khi chia ảnh thành 16 vùng, ta thực hiện chỉnh lại kích thước nhỏ hơn 10% so với ảnh gốc ban đầu (tức ảnh có kích thước hiện tại là 48x64). Nếu như sau quá trình cắt bỏ phần cổ tay dư thừa mà ảnh nhị phân có phần nằm lệch sang một hướng nhất định thì chúng tôi thực hiện crop (cắt) vùng ảnh nhị phân bằng cách xác định các điểm góc quan trọng của một ảnh như góc bên trái, phải, trên và dưới, sau đó tinh chỉnh lại kích thước 48x64. Kế đến, chúng tôi sử dụng thuật toán tìm đường biên của ảnh nhị phân (Edge). Xác định đường biên (Edge Detection) là quá trình địa phương hóa chuyển tiếp cường độ điểm ảnh. Có nhiều phương pháp xác định đường biên, ở đây chúng tôi sử dụng phương pháp Sobel, Sobel sử dụng hai mặt nạ (masks), một chiều ngang và một chiều dọc. Cả hai mặt nạ thường được sử dụng là ma trận 3x3. Khi một ảnh được áp dụng cho việc xác định đường biên, hệ thống sẽ tạo ra hai mặt nạ như trên, cả hai mặt nạ sẽ kết hợp lại với nhau để tạo ra một ảnh duy nhất. Mục đích là để xác định sự tồn tại và vị trí của các cạnh trong một ảnh. Hướng các cạnh thu được là hướng cơ bản của Gradient, gradient là hướng thay đổi của cường độ tối đa, thay đổi có thể là tăng hoặc giảm.

Zoning là phương pháp chia một ảnh thành nhiều vùng nhỏ để dễ dàng xử lý hơn. Trong chương trình này, chúng tôi chia ảnh thành 16 vùng nhỏ bằng nhau sau khi đã thực hiện xác định đường biên của ảnh (Edge Detection).



Hình 3.8: Ảnh chia thành 16 vùng bằng nhau

Bước tiếp theo, chúng tôi tính tổng số điểm ảnh màu trắng của từng vùng và cho ra 16 con số. Kế đến, chúng tôi tính số histogram (ở đây nghĩa là tính tổng số pixel trắng) theo chiều dọc của 16 vùng này và cho ra 4 con số. Như vậy ở đây chúng tôi có 20 con số được đưa vào tập đặc trưng của một ảnh. Tổng cộng một ảnh chúng tôi có 25 con số cho đặc trưng.

Những con số đặc trưng được rút trích được đưa vào một ma trận tập hợp từ tất cả các ảnh trong bộ dữ liệu huấn luyện, ma trận có kích thước 1040 cột x 25 hàng (1040 là tổng số ảnh có trong bộ dữ liệu huấn luyện). Sau đó, chúng tôi tiến hành chuẩn hóa các giá trị về những con số nhỏ hơn hoặc bằng 1 nhằm mục đích giảm thiểu những đặc trưng mang giá trị lớn lấn áp các đặc trưng có giá trị nhỏ. Ví dụ: số điểm ảnh trắng thường có giá trị rất lớn, vào khoảng 2000 trở lên, trong khi đó, số endpoints hay số branchpoints thường chỉ vào khoảng 3 – 60.

* + 1. **Tiến trình nhận dạng**

Dựa vào các thuật toán nhận dạng như k-NN, SVM, Random Forest. Chúng tôi tiến hành kiểm thử ban đầu dựa vào bộ ảnh kiểm tra. Sau đó tiến hành nhận dạng thông qua camera. Tiến trình nhận dạng cần một ma trận tổng số đặc trưng và ảnh nhận dạng, ảnh được đưa vào đi qua quá trình tiền xử lý ảnh (Pre-processing), sau đó được đưa vào hàm chức năng nhận dạng nhất định như VSLknn, VSLsvm hay TreeBaggering để nhận dạng và cho ra kết quả hiển thị lên màn hình dạng chữ viết.

Việc nhận dạng với thuật toán k-NN, ban đầu ta phải xác định trước một số k cho hàm chức năng VSLknn này. Số k là số láng giềng gần nhất mà hàm này sẽ xét khi thực hiện tác vụ nhận dạng. Không hẳn chọn k lớn thì tốt, số k lớn có thể ảnh hưởng tới thời gian nhận dạng một ký tự và sau nhiều lần chạy thử nghiệm thì chúng tôi quyết định chọn k = 8. Song song với chọn k, một tham số khác là các phương thức tính khoảng cách, ở đây chúng tôi chọn euclidean. Và một tham số quyết định làm thế nào để phân loại mẫu là “nearest”. Hàm chức năng được sử dụng trong Matlab ở đây là knnclassify.

Với thuật toán Random Forest, sử dụng ma trận đặc trưng của toàn bộ ảnh huấn luyện (kích thước 1040 hàng x 25 cột) chúng tôi gán thêm một cột thứ 26 nữa, cột này là những con số nguyên dùng để gán nhãn cho các lớp bởi hàm chức năng TreeBagger. Ma trận sau khi gán có kích thước 1040x26 chính là ma trận đặc trưng dùng cho việc nhận dạng sau này của thuật toán RF. Một đối tượng được tạo ra hội tụ các thông số được nêu rõ trong TreeBagger với các tham số như sau:

* nTrees: Số cây quyết định (Decision Tree) được tạo. Sau nhiều lần kiểm thử chúng tôi chọn nTrees = 10, cho kết quả và thời gian khá tốt.
* X: là những đặc trưng (features), có thể mang kiểu vector Cell của mảng String, Numberic vector, Ma trận (Matrix). Ở đây chúng tôi sử dụng ma trận.
* Y: là nhãn dùng để phân lớp đối tượng (classLabels), là cột thứ 26 vừa được gán ở bước trên.
* Method: ở đây chúng tôi sử dụng classification.

Chương 4: **KẾT QUẢ THỬ NGHIỆM VÀ ĐÁNH GIÁ**

* 1. **Qui trình thử nghiệm**

Chúng tôi thực hiện kiểm tra bẳng 2 cách: thứ nhất là kiểm tra bẳng ảnh từ bộ dữ liệu kiểm tra (10 ảnh cho một ký tự), và thứ hai là kiểm tra thông qua camera.

Các thành phần cần phải có để phục vụ cho quá trình kiểm tra:

* Một camera để ghi nhận hình ảnh của ngôn ngữ ký hiệu (giao tiếp qua cổng USB).
* Phông nền màu xanh.
* Chương trình Matlab.
* Một ma trận đặc trưng của tất cả các ảnh trong bộ dữ liệu huấn luyện.
* Máy tính.

Tiến hành thử nghiệm: Nhìn chung cả hai cách kiểm tra đều tương tự như nhau, thường thì kiểm tra bằng ảnh cho độ chính xác cao hơn kiểm tra bằng camera. Rất dễ hiểu vì khi kiểm tra bằng camera, camera lấy nét ảnh theo cơ chế tự động (auto-focus) đôi khi một số hình ảnh sẽ bị nhòe, từ đó gây trở ngại cho các bước trong tiến trình xử lý ảnh nên cho kết quả có phần sai lệch. Để giảm thời gian nhận dạng, chúng tôi tách riêng quá trình rút trích đặc trưng bộ ảnh huấn luyện, khi đi vào bước kiểm tra chúng tôi chỉ sử dụng ma trận đặc trưng đã có sẵn.

Sau khi để chuẩn bị đầy đủ các thành phần nêu trên, chúng tôi khởi động chương trình Matlab. Đầu tiên, một người sẽ đứng trước camera và thực hiện các ký hiệu ký tự trong bảng chữ cái của ngôn ngữ ký hiệu Việt Nam. Camera sẽ ghi nhận hình ảnh bằng cách chụp lại, trải qua các bước trong chương trình Matlab và cho kết quả hiển thị ra màn hình của máy tính.

* 1. **Kết quả thử nghiệm**

Dưới đây là biểu đồ kết quả khi thực hiện kiểm tra bằng tập dữ liệu ảnh kiểm tra. Mỗi ký tự có 10 ảnh kiểm tra.

Biểu đồ 4.1: Kết quả kiểm tra bằng thuật toán k-NN

Biểu đồ 4.2: Kết quả kiểm tra bằng thuật toán SVM

Biểu đồ 4.3: Kết quả kiểm tra bằng thuật toán Random Forest (RF)

Các biểu đồ trên chưa thật sự đánh giá tốt kết quả của ba thuật toán. Nên chúng tôi dùng Confusion Matrix để tính toán các thông số về độ chính xác như sau:

Bảng 4.1: Kết quả kiểm tra bằng Confusion Matrix. Đơn vị: %

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuật toán** | **k-NN** | **SVM** | **Random Forest** |
| **Accuracy** | 98.53 | 98.88 | 97.28 |
| **Sensitivity** | 80.77 | 99.42 | 64.62 |
| **Specificity** | 99.24 | 85.39 | 98.59 |

Việc kiểm tra cũng được thực hiện qua camera – cho kết quả trung bình thời gian nhận dạng một ký hiệu là 4 – 5 giây. Kết quả này phụ thuộc một phần vào phần cứng máy tính thực hiện. Ngoài ra việc kiểm các ký tự như Ă, Â, Ô, Ơ, Đ chỉ thực hiện theo cách kiểm tra các ký tự đầu vào:

* Nếu ảnh thứ nhất là ký tự A, O, U, E thì cho phép chương trình kiểm tra thêm một ký tự nữa. Nếu ký tự tiếp theo là một trong hai ký tự “dấu râu”, hay “dấu mũ” thì cho hiển thị ra màn hình ký tự tương úng Ă, Â, Ô, Ê, Ư.
* Cũng trường hợp trên, nhưng khi ảnh thứ hai được kiểm tra không thuộc một trong hai ký tự “dấu râu” hoặc “dấu mũ” thì chỉ cho phép hiển thị ra màn hình ký tự ban đầu, tức là A, O, U, E.

Đánh giá việc nhận dạng các kiểu ký tự NG, CH, NGH cũng tương tự như vậy, cho kết quả bị phụ thuộc vào các ký tự đơn nên giải pháp này chưa thật sự tốt.

* 1. **Kết luận, kiến nghị và hướng phát triển**
     1. **Kết luận**

Qua thời gian nghiên cứu đề tài này chúng em đã học được một số kỹ năng trong công việc, đạt được những kết quả sau:

* Nghiên cứu và xây dựng thành công chương trình nhận dạng các ký tự của ngôn ngữ ký hiệu.
* Nắm rõ được các thuật toán nhận dạng, máy học như k-NN, SVM hay Random Forest.
* Biết và xử lý được các vấn đề mà thường xuyên xảy ra đối với việc xây dựng hệ thống đó là một số lỗi thực tế nó không hẳn như lý thuyết, hiểu được từ lý thuyết tới thực tế là một quãng đường dài.
* Kỹ năng tìm tài liệu, phân tích và trích xuất thông tin từ các tài liệu, làm việc nhóm.
* Học được một số kỹ năng trong quá trình nghiên cứu khoa học.
* Cách viết báo cáo một nghiên cứu khoa học.
* Điểm mới của khóa luận là thiết kế thành công thành công chương trình nhận dạng ngôn ngữ ký hiệu Việt Nam, đây cũng được coi là một chương trình đầu tiên tại Việt Nam trong lĩnh vực nhận dạng ngôn ngữ ký hiệu hỗ trợ người khiếm thính tại Việt Nam.
  + 1. **Kiến nghị**

Trên khuôn khổ của khóa luận, nhóm tập trung nghiên cứu và xây dựng hệ thống có khả năng thu nhập dữ liệu từ camera thông thường. Trên cở sở này nhóm muốn phát triển, mở rộng thêm nên nhóm kiến nghị:

* Nghiên cứu và xây dựng hoàn thiện hơn các tính năng nhận dạng thông qua các cử chỉ phức tạp hơn như các ký hiệu động.
* Áp dụng các công nghệ hiện đại và tiên tiến trên thế giới vào hệ thống.
  + 1. **Hướng phát triển**

Chúng tôi sẽ thực hiện việc kiểm tra trên những máy tính với cấu hình mạnh hơn để so sánh và đánh giá. Hiện tại, hệ thống vẫn còn tồn tại một số nhược điểm. Đầu tiên, còn một số ký tự kép, các ký tự như Ă, Â, Ư, Ơ, Ô, việc nhận dạng chưa được tối ưu lắm. Một số ký tự có ký hiệu gần gống nhau như A với S và T, R với U, dấu râu với X, cho kết quả nhận dạng còn trùng nhau. Thêm vào đó, quá trình lấy ảnh còn bị ảnh hưởng nhiều bởi các yếu tố ánh sáng, nền ảnh. Về cơ bản, bộ dữ liệu huấn luyện không có bộ chuẩn nên chúng tôi tự xây dựng bộ dữ liệu riêng cho chương trình. Vì vậy kết quả đánh giá sẽ bị ảnh hưởng khá nhiều dựa vào độ tin cậy của bộ dữ liệu này.

Trong tương lai, chúng tôi sẽ hoàn thiện hơn các tính năng của hệ thống. Xử lý được các ký tự đặc biệt bằng cách tìm hiểu các kĩ thuật theo vết đối tượng trong video, khắc phục vấn đề bị ảnh hưởng bởi môi trường như ánh sáng, có nhiều vật dụng xung quanh vùng kiểm tra cũng như hoàn thiện bộ dữ liệu ảnh.

**DANH MỤC TÀI LIỆU THAM KHẢO**

**Tiếng việt:**

[1] TS. Đỗ Năng Toàn, TS. Phạm Việt Bình, *Giáo trình môn học Xử lý ảnh*, Đại Học Thái Nguyên, tháng 11 năm 2007.

[2] TS. Vũ Đức Lung. *Slide bài giảng “Phương pháp luận sáng tạo Khoa học – Công nghệ”*. ĐH Công nghệ Thông tin. 2013

[3] Nguyễn Thị Lan Anh, *Nghiên Cứu Thuật Toán Máy Học SVM và Ứng Dụng Trong Bài Toán Khám Phá Ý Kiến Phản Hồi của Khách Hàng Trên Website*, Học Viện Bưu Chính Viễn Thông, Hà Nội, năm 2013.

**Tiếng anh:**

[1] Lousia Lam, Seong-Whan Lee, *Thining Methodologies – A Comprehensice Survey*, IEEE, Vol 14, NO.9, September 1992.

[2] Elif AYBAR, *Sobel Edge Detection Method For Matlab*, Anadolu University.

[3] Keith Jack, Video Demystified – A Handbook for the Digital Engineer Third Edition, chapter 3.

**Một số website tham khảo:**

[1] [http://en.wikipedia.org](http://en.wikipedia.org/)

[2] [http://vi.wikipedia.org](http://vi.wikipedia.org/)

1. Phase Alternating Line – là hệ truyền hình được dùng phần lớn ở Châu Âu, châu Á. Chuẩn PAL có tần số quét ngang là 50Hz (tương đương 25 khung hình/s). [↑](#footnote-ref-1)
2. National Teltevision System Committee – là truyền hình được sử dụng hầu hết ở Bắc Mỹ và Nam Mỹ, chuẩn NTSC có tần số quét ngang là 60 Hz (tương đương 30 khung hình/s). [↑](#footnote-ref-2)
3. Tham khảo : Slide bài giảng “Phương pháp luận sáng tạo Khoa học – Công nghệ”, TS. Vũ Đức Lung, Trường ĐH QG TPHCM – ĐH CNTT. [↑](#footnote-ref-3)
4. Nhị phân hóa là quá trình biến một ảnh xám thành ảnh nhị phân. Ảnh nhị phân chỉ chứa hai giá trị 0 hoặc 1 (hoặc 0 và 255 tùy theo quy định câu trúc ảnh). Với một ngưỡng T cho trước, công thức để nhị phân ảnh là đặt nhị phân ảnh là đặt tất cả các giá trị của ảnh xám về 0 nếu chúng bé hơn T và 255 nếu chúng lớn hơn T. [↑](#footnote-ref-4)