|  |  |
| --- | --- |
|  |  |
|  | BỘ CÔNG THƯƠNG  TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI  --------------------------------------- |
|  |
| ĐỒ ÁN TỐT NGHIỆP |
| NGÀNH CÔNG NGHỆ THÔNG TIN |
| TÊN ĐỀ TÀI: |
| XÂY DỰNG HỆ THỐNG PHÁT HIỆN VÀ PHÂN LOẠI BIỂN SỐ XE SỬ DỤNG DEEPLEARNING |
|  |
| CBHD: TS. PHẠM VĂN HIỆP |
|  | Sinh viên: NGUYỄN QUANG SÁNG |
| Mã số sinh viên: 2020601249  Lớp: CNTT01-K15 |
|  |
|  |
|  |
| Hà Nội – Năm 2024 |
|  |

MỤC LỤC

[DANH MỤC HÌNH ẢNH 5](#_Toc167176441)

[DANH MỤC BẢNG BIỂU 7](#_Toc167176442)

[LỜI CẢM ƠN 8](#_Toc167176443)

[MỞ ĐẦU 9](#_Toc167176444)

[1. Lí do chọn đề tài 9](#_Toc167176445)

[2. Mục đích nghiên cứu của đề tài 10](#_Toc167176446)

[3. Đối tượng và phạm vi nghiên cứu của đồ án 11](#_Toc167176447)

[3.1 Đối tượng nghiên cứu: 11](#_Toc167176448)

[3.2 Phạm vi nghiên cứu: 11](#_Toc167176449)

[4. Ý nghĩa khoa học và thực tiễn của đề tài: 11](#_Toc167176450)

[CHƯƠNG I: GIỚI THIỆU TỔNG QUAN 12](#_Toc167176451)

[1.1 Thực trạng giao thông Việt Nam và những vấn đề tồn tại 12](#_Toc167176452)

[1.1.1 Thực trạng giao thông Việt Nam 12](#_Toc167176453)

[1.1.2 Ý tưởng xây dựng hệ thống phát hiện và phân loại biển số xe 13](#_Toc167176454)

[1.2 Tổng quan về hệ thống xử lý ảnh 13](#_Toc167176455)

[1.2.1 Tổng quan về xử lý ảnh 13](#_Toc167176456)

[1.2.2 Thu nhận ảnh: 15](#_Toc167176457)

[1.2.3 Phát hiện ảnh 22](#_Toc167176458)

[1.2.4 Phân loại ảnh 24](#_Toc167176459)

[1.2.5 Phân vùng ảnh 27](#_Toc167176460)

[1.2.6 Nhận dạng ảnh 30](#_Toc167176461)

[1.3 Tổng quan về Deep Learning và bài toán nhận dạng biển số xe 37](#_Toc167176462)

[1.3.1 Tổng quan về Deep Learning 37](#_Toc167176463)

[1.3.2 Bài toán nhận diện biển số xe 39](#_Toc167176464)

[1.4 Các kết quả nghiên cứu trong và ngoài nước 40](#_Toc167176465)

[1.4.1 Một số nghiên cứu đã và đang thực hiện tại nước ta 40](#_Toc167176466)

[1.4.2 Các kết quả nghiên cứu nước ngoài 41](#_Toc167176467)

[1.4.3 Định hướng của đề tài 42](#_Toc167176468)

[CHƯƠNG II: HỆ THỒNG PHÁT HIỆN VÀ PHÂN LOẠI BIỂN SỐ XE SỬ DỤNG DEEPLEARNING 43](#_Toc167176469)

[2.1 Khái quát chung về hệ thống phát hiện và phân loại biển số xe 43](#_Toc167176470)

[2.3 Phát hiện biển số xe 44](#_Toc167176471)

[2.3.1 Tổng quát bài toán phát hiện biển số xe 44](#_Toc167176472)

[2.3.2 Giới thiệu thuật toán Yolov8 trong bài toán phát hiện đối tượng 45](#_Toc167176473)

[2.3.3 Thu thập và gán nhãn dữ liệu 47](#_Toc167176474)

[2.3.4 Đào tạo mô hình 49](#_Toc167176475)

[2.4 Phân loại biển số xe 49](#_Toc167176476)

[2.4.1 Tổng quát bài toán phân loại biển số xe 49](#_Toc167176477)

[2.4.2 Giới thiệu thuật toán Resnet34 trong bài toán phân loại 50](#_Toc167176478)

[2.4.3 Thu thập và gán nhãn dữ liệu 52](#_Toc167176479)

[2.4.4 Đào tạo mô hình 54](#_Toc167176480)

[2.5 Nhận dạng kí tự biển số xe 54](#_Toc167176481)

[2.5.1 Tổng quát về module nhận dạng kí tự biển số xe 54](#_Toc167176482)

[2.5.2 Giới thiệu thư viện mã nguồn mở EasyOCR 55](#_Toc167176483)

[2.5.3 Tích hợp thư viện EasyOCR vào module nhận diện biển số xe 57](#_Toc167176484)

[2.2 Phân tích thiết kế hệ thống phát hiện và phân loại biển số xe 58](#_Toc167176485)

[2.2.1 Phân tích một số hệ thống nhận diện biển số xe hiện nay 58](#_Toc167176486)

[2.2.2 Yêu cầu,mục tiêu và ưu điểm của hệ thống 61](#_Toc167176487)

[2.2.3 Thiết kế kiến trúc hệ thống 62](#_Toc167176488)

[2.2.4 Thiết kế cơ sở dữ liệu 63](#_Toc167176489)

[2.2.5 Thiết kế giao diện người dùng 63](#_Toc167176490)

[CHƯƠNG III: THỰC NGHIỆM, CÀI ĐẶT VÀ MÔ PHỎNG HỆ THỐNG 65](#_Toc167176491)

[3.1 Kết quả thực nghiệm mô hình 65](#_Toc167176492)

[3.1.1 Mô hình phát hiện biển số xe 65](#_Toc167176493)

[3.1.2 Mô hình phân loại biển số xe 65](#_Toc167176494)

[3.2 Giao diện hệ thống 67](#_Toc167176495)

[3.3 Kết quả thực nghiệm hệ thống 71](#_Toc167176496)

[3.3.1 Test hệ thống bằng ảnh trong thư viện 71](#_Toc167176497)

[3.3.2 Test hệ thống bằng video trong thư viện 72](#_Toc167176498)

[3.3.3 Test hệ thống bằng webcam 73](#_Toc167176499)

[KẾT LUẬN 75](#_Toc167176500)

[TÀI LIỆU THAM KHẢO 76](#_Toc167176501)

[PHỤ LỤC 77](#_Toc167176502)

# DANH MỤC HÌNH ẢNH

[Hình 1.1:Sơ đồ tổng quát hệ thống xử lí ảnh 15](#_Toc166874736)

[Hình 1.2:Hình ảnh CCD camera 16](#_Toc166874737)

[Hình 1.3: Hình ảnh màn hình video 17](#_Toc166874738)

[Hình 1.4: Hình ảnh máy tính 18](#_Toc166874739)

[Hình 1.5: Hệ tọa độ RGB 19](#_Toc166874740)

[Hình 1.6: Biểu diễn mức xám của ảnh số 21](#_Toc166874741)

[Hình 1.7: Đường biên của ảnh 24](#_Toc166874742)

[Hình 1.8: Các bước xử lí và phân tích ảnh 24](#_Toc166874743)

[Hình 1.9: Lược đồ rắn lượn và cách chọn ngưỡng 29](#_Toc166874744)

[Hình 1.10: Mô hình nơ-ron nhân tạo 37](#_Toc166874745)

[Hình 2.1:Sơ đồ hệ thống phát hiện và phân loại biển số xe 45](#_Toc166874746)

[Hình 2.2: Kiến trúc mô hình Yolov8 48](#_Toc166874747)

[Hình 2.3: Quá trình gán nhãn bouding box biển số xe 50](#_Toc166874748)

[Hình 2.4: Kiến trúc mô hình Resnet34 52](#_Toc166874749)

[Hình 2.5: Mẫu dữ liệu bài toán phân loại biển số xe 55](#_Toc166874750)

[Hình 2.6: Sử dụng thư viện EasyOCR nhận diện kí tự biển số xe 59](#_Toc166874751)

[Hình 2.7: Quá trình gán nhãn mô hình nhận diện kí tự 60](#_Toc166874752)

[Hình 2.8: Giao diện hệ thống quản lí bãi xe 61](#_Toc166874753)

[Hình 3.1: Kết quả đánh giá quá trình trainning mô hình phát hiện biển số 66](#_Toc166874754)

[Hình 3.2: Kết quả đánh giá mô hình phân loại biển số mờ 67](#_Toc166874755)

[Hình 3.3: Kết quả đánh giá mô hình phân loại biển số bị che khuất 67](#_Toc166874756)

[Hình 3.4: Giao diện giới thiệu của hệ thống 68](#_Toc166874757)

[Hình 3.5: Giao diện nhận diện và phân loại biển số xe 69](#_Toc166874758)

[Hình 3.6: Giao diện chọn file từ thư viện 69](#_Toc166874759)

[Hình 3.7: Giao diện hiển thị kết quả nhận diện và phân loại ảnh biển số xe 70](#_Toc166874760)

[Hình 3.8: Giao diện hiển thị kết quả nhận diện và phân loại biển số bằng webcam 71](#_Toc166874761)

[Hình 3.9: Giao diện hiển thị kết quả nhận diện và phân loại video biển số xe 71](#_Toc166874762)

[Hình 3.10: Chọn ảnh trong thư viện 72](#_Toc166874763)

[Hình 3.11: Kết quả nhận diện và phân loại ảnh biển số xe 72](#_Toc166874764)

[Hình 3.12: Kết quả sau khi xuất ra file excel 73](#_Toc166874765)

[Hình 3.13: Chọn video trong thư viện 73](#_Toc166874766)

[Hình 3.14: Kết quả nhận diện và phân loại video biển số xe 74](#_Toc166874767)

[Hình 3.15: Mở webcam để bắt đầu nhận diện 74](#_Toc166874768)

[Hình 3.16: Kết quả nhận diện và phân loại biển số xe bằng webcam 75](#_Toc166874769)

# DANH MỤC BẢNG BIỂU

[Bảng 1.1 So sánh khả năng làm việc của bộ não và máy tính 36](#_Toc166875149)

# LỜI CẢM ƠN

Trong thời gian làm đồ án tốt nghiệp, em đã nhận được nhiều sự giúp đỡ, đóng góp ý kiến và chỉ bảo nhiệt tình của thầy cô và bạn bè.

Em xin gửi lời cảm ơn chân thành đến thầy ***TS. Phạm Văn Hiệp***, giảng viên khoa Công Nghệ Thông Tin trường Đại Học Công Nghiệp Hà Nội, người đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình em làm đồ án.

Em cũng xin chân thành cảm ơn các thầy cô trong trường Đại học Công Nghiệp Hà Nội nói chung và Khoa Công Nghệ Thông Tin nói riêng đã dạy dỗ cho em kiến thức về các môn đại cương cũng như các môn chuyên ngành, giúp em có được cơ sở lý thuyết vững vàng và tạo điều kiện cho em hoàn thành đồ án tốt nghiệp này.

Cuối cùng, em xin chân thành cảm ơn thầy cô và bạn bè, đã luôn tạo điều kiện, quan tâm, giúp đỡ, động viên em trong suốt quá trình học tập và hoàn thành đồ án tốt nghiệp.

Em xin chân thành cảm ơn!!

Sinh viên thực hiện

***Nguyễn Quang Sáng***

# MỞ ĐẦU

## 1. Lí do chọn đề tài

Trong lĩnh vực trí tuệ nhân tạo (AI), deep learning đã trở thành một công nghệ quan trọng và mạnh mẽ, mở ra những tiềm năng vô tận trong việc giải quyết các vấn đề phức tạp, đặc biệt là trong lĩnh vực thị giác máy tính. Trong thời đại số hóa và tiến bộ công nghệ, deep learning đã đóng vai trò quan trọng trong việc tạo ra những giải pháp sáng tạo và hiệu quả cho nhiều ứng dụng thực tế, bao gồm cả hệ thống nhận diện biển số xe.

Hệ thống nhận diện biển số xe đã trở thành một phần không thể thiếu trong quản lý giao thông, an ninh và an toàn trên đường. Sử dụng deep learning đã mang lại những tiến bộ đáng kể trong việc cải thiện hiệu suất và độ chính xác của các hệ thống nhận diện biển số xe. Bằng cách sử dụng các mô hình mạng nơ-ron sâu như Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs) và các biến thể của chúng, hệ thống có khả năng tự động học và nhận diện các biển số xe trong nhiều điều kiện khác nhau

Nhận thấy đây là một lĩnh vực mang nhiều ứng dụng thực tiễn và có tiềm năng phát triển cao trong tương lai cùng với sự đam mê nghiên cứu về cả hai lĩnh vực deeplearning và giao thông vậy nên đồ án tốt nghiệp xin trình bày một đề tài là *Xây dựng hệ thống phát hiện và phân loại biển số xe*. Trong đồ án sẽ trình bày về các mô hình deep learning phổ biến được sử dụng trong lĩnh vực này, cũng như đi sâu vào các phương pháp tiền xử lý dữ liệu và tối ưu hóa mô hình. Đồng thời, đồ án cũng sẽ thực hiện một số bài toán cụ thể về cách mà deep learning đã được ứng dụng trong các hệ thống nhận diện biển số xe thực tế

## 2. Mục đích nghiên cứu của đề tài

Mục đích nghiên cứu của đề tài xây dựng hệ thống phát hiện và phân loại biển số xe là:

* Tìm hiểu và tối ưu hóa mô hình: Nghiên cứu nhằm phát triển các thuật toán và mô hình để cải thiện độ chính xác và tốc độ phát hiện biển số xe. Mục tiêu là tạo ra một hệ thống nhận diện biển số xe chính xác và hiệu quả trong nhiều điều kiện ánh sáng, thời tiết và góc chụp khác nhau.
* Tăng cường tính ứng dụng: Đối với các nhà phát triển ứng dụng, mục tiêu là xây dựng hệ thống nhận diện biển số xe có thể tích hợp dễ dàng vào các ứng dụng thực tế như hệ thống giao thông thông minh, quản lý bãi đỗ xe, hoặc giám sát an ninh.
* Cải thiện quản lý giao thông và an ninh: Đối với cơ quan quản lý giao thông và an ninh, mục tiêu là sử dụng hệ thống nhận diện biển số xe để tăng cường khả năng giám sát và quản lý giao thông trên các tuyến đường, đồng thời cải thiện an ninh và an toàn cho cộng đồng.
* Nghiên cứu và phát triển công nghệ: Mục tiêu cuối cùng là đóng góp vào việc nghiên cứu và phát triển các công nghệ nhận diện biển số xe, từ đó nâng cao hiểu biết và khả năng áp dụng của cộng đồng nghiên cứu và thực tiễn trong lĩnh vực này.

## 3. Đối tượng và phạm vi nghiên cứu của đồ án

### 3.1 Đối tượng nghiên cứu:

Đây là một trong những ứng dụng xử lí ảnh nhận được nhiều sự quan tâm nghiên cứu về cả hai mặt lý thuyết và thực hiện chương trình. Nội dung của đồ án tập trung nghiên cứu các quá trình xử lí ảnh: phân vùng biển số, tách các ký tự, nhận biết các kí tự, phần mềm quản lý cơ sở dữ liệu biển số xe máy. Đề tài tập trung nghiên cứu về các mô hình deeplearning và áp dụng các thuật toán này để giải quyết các bài toán phục vụ cho bài toán nhận diện và phân loại biển số xe.

### 3.2 Phạm vi nghiên cứu:

Chủ yếu dựa vào các tài liệu và chạy mô phỏng nhận dạng biển số xe máy trên phần mềm, chưa có mô hình thực tế.

Việc có nhiều biển số xe với định dạng và độ sáng khác nhau gây khó khăn cho việc nhận dạng. Do quá trình nhận dạng dựa vào phương pháp xử lý ảnh và trích xuất biển số từ ảnh chụp nên độ sáng khác nhau làm tăng độ phức tạp trong quá trình nhận dạng.

## 4. Ý nghĩa khoa học và thực tiễn của đề tài:

Đề tài phát hiện và phân loại biển số xe mang lại ý nghĩa khoa học bằng cách đóng góp vào lĩnh vực trí tuệ nhân tạo và thị giác máy tính. Nghiên cứu này cung cấp các phương pháp và mô hình mới, từ đó nâng cao hiểu biết về deep learning và tạo ra cơ sở cho sự phát triển tiếp theo trong lĩnh vực nhận diện biển số xe.

Ý nghĩa thực tiễn của đề tài nằm ở việc cải thiện quản lý giao thông và an ninh, hỗ trợ phát triển các ứng dụng thông minh và tự động hóa trong các lĩnh vực như giao thông, đô thị thông minh và an ninh. Hơn nữa, việc sử dụng hệ thống nhận diện biển số xe cũng giúp tiết kiệm thời gian và công sức của con người trong quy trình nhận diện và phân loại, đồng thời đảm bảo an toàn và thuận tiện cho cộng đồng trong việc di chuyển hàng ngày

# CHƯƠNG I: GIỚI THIỆU TỔNG QUAN

## 1.1 Thực trạng giao thông Việt Nam và những vấn đề tồn tại

### 1.1.1 Thực trạng giao thông Việt Nam

Thực trạng giao thông Việt Nam Đã từ rất lâu rồi, tình trạng giao thông tại Việt Nam, đặc biệt là tại các thành phố lớn như Hà Nội hay Hồ Chí Minh, ùn tắc giao thông luôn là vấn đề nhức nhối đối với người dân sinh sống trong khu vực đó. Ngày nay, tình trạng ùn tắc giao thông, tai nạn giao thông lại càng trở lên nghiêm trọng hơn, xảy ra hầu hết trên các tỉnh thành trong cả nước. Với việc chuyển hướng từ một đất nước có nền kinh tế kế hoạch hóa tập trung sang một nền kinh tế thị trường tạo điều kiện phát triển mạnh cho nền kinh tế Việt Nam. Kinh tế phát triển, đời sống người dân được nâng cao kèm theo đó là những vấn đề liên quan đến giao thông đường bộ, ùn tắc, tai nạn giao thông mà ít ai nhận ra mối liên hệ giữa chúng. Đã có rất nhiều giải pháp đặt ra như đẩy mạnh việc cải tạo và phát triển cơ sở hạ tầng, tổ chức sắp xếp và điều hành hướng dẫn giao thông hợp lý, xử lý mạnh các vi phạm về trật tự an toàn giao thông và giải tỏa hành lang an toàn giao thông; tăng cường tuyên truyền, nâng cao ý thức chấp hành pháp luật khi tham gia giao thông và xây dựng văn hóa giao thông. Nhưng tình trạng ùn tắc giao thông vẫn diễn biến phức tạp, các điểm có nguy cơ xảy ra ùn tắc có chiều hướng gia tăng tại các khu vực đông dân cư, các tuyến giao thông trọng điểm, các trục chính ra vào đô thị ,… gây ảnh hưởng lớn đến giao thông đô thị, nhất là vào các dịp lễ Tết, khi diễn ra các lễ hội sự kiện lớn, tai nạn giao thông ngày càng gia tăng đang còn là một vấn đề bức xúc của toàn xã hội

### 1.1.2 Ý tưởng xây dựng hệ thống phát hiện và phân loại biển số xe

Tình trạng giao thông ngày nay đang đối mặt với nhiều thách thức, từ kẹt xe đến vi phạm luật giao thông và cảnh báo an ninh. Để giải quyết những vấn đề này, một ý tưởng hữu ích là xây dựng hệ thống phát hiện và phân loại biển số xe. Hệ thống này có thể phát hiện và ghi nhận biển số xe trên các phương tiện di chuyển chậm trong khu vực kẹt xe, từ đó cung cấp dữ liệu về lưu lượng giao thông và cảnh báo về vi phạm giao thông như vượt đèn đỏ hoặc vượt quá tốc độ giới hạn.

Bên cạnh đó, hệ thống cũng có thể được sử dụng để quản lý bãi đỗ xe công cộng bằng cách tự động ghi nhận và kiểm soát việc đỗ xe. Điều này giúp tối ưu hóa sử dụng không gian và tăng cường hiệu quả quản lý. Ngoài ra, hệ thống còn có thể được triển khai để giám sát an ninh trong các khu vực công cộng như bến xe buýt hoặc ga đường sắt, giúp phát hiện và cảnh báo về các hoạt động đáng ngờ hoặc các phương tiện không mong muốn.

## 1.2 Tổng quan về hệ thống xử lý ảnh

### 1.2.1 Tổng quan về xử lý ảnh

Ngày nay kỹ thuật xử lí ảnh đã được ứng dụng rộng rãi ở rất nhiều lĩnh vực, trong sản xuất cũng như trong đời sống. Ví dụ các hệ thống xử lí ảnh vệ tinh để phân tích không gian vũ trụ, hệ thống thăm dò địa chất, hệ thống phân tích tế bào sinh học và gần gũi nhất với chúng ta là các phần mềm hiển thị và xử lí ảnh chuyên dụng như Photoshop, ACD See…

Một hệ thống xử lý ảnh là hệ thống thực hiện các chức năng thu nhận ảnh đầu vào, thực hiện phép xử lý để tạo ảnh hoặc kết quả phân tích, nhận dạng ở đầu ra đáp ứng các yêu cầu và các ứng dụng cụ thể.

Trong phạm vi của đề tài, đồ án xin giới hạn trong việc giới thiệu một hệ thống xử lý ảnh ứng dụng nhận dạng và ra quyết định trên thực tế.

Nhận dạng ảnh

Camera

Thu nhận ảnh

Phân tích ảnh

Lưu trữ

Hình 1.1:Sơ đồ tổng quát hệ thống xử lí ảnh

Sơ đồ khối tổng quát của hệ thống này được thể hiện trong hình 1.1, trong đó gồm ba khối chức năng cơ bản:

* ***Khối thu nhận ảnh***: thực hiện chức năng thu nhận ảnh và thực hiện quá trình số hóa (lưu giữ theo định dạng yêu cầu).
* ***Khối phân tích ảnh***: trước hết hệ thống tiến hành bước tiền xử lý ảnh với mục đích tăng cường, cải thiện chất lượng ảnh, làm nổi các đặc trưng cơ bản của ảnh hay làm cho ảnh gần giống nhất với trạng thái gốc. Sau đó, là quá trình phân tích ảnh và trích chọn đặc trưng của ảnh ví dụ như biên, điểm gấp khúc, điểm kết thúc, điểm chữ thập…
* ***Khối nhận dạng:*** dựa vào các đặc trưng đã thu nhận từ quá trình phân tích ảnh trước đó thực hiện quá trình nhận dạng, đưa ra các quyết định ứng với các ứng dụng cụ thể.

### 1.2.2 Thu nhận ảnh:

#### 1.2.2.1 Các thiết bị thu nhận ảnh:

Hai thành phần cho công đoạn này là linh kiện nhạy với phổ năng lượng điện từ trường, loại thứ nhất tạo tín hiệu điện ở đầu ra tỷ lệ với mức năng lượng mà bộ cảm biến (đại diện là camera); loại thứ hai là bộ số hoá.

* **Lý thuyết về camera:**



Hình 1.2:Hình ảnh CCD camera

Tổng quát có hai kiểu camera: kiểu camera dùng đèn chân không và kiểu camera chỉ dùng bán dẫn. Đặc biệt là trong lĩnh vực này, camera bán dẫn thường hay được dùng hơn camera đèn chân không. Camera bán dẫn cũng được gọi là CCD camera do dùng các thanh ghi dịch đặc biệt gọi là thiết bị gộp (Charge-Coupled Devices- CCDs). Các CCD này chuyển các tín hiệu ảnh sang từ bộ cảm nhận ánh sáng bổ trợ ở phía trước camera thành các tín hiệu điện mà sau đó được mã hóa thành tín hiệu TV. Loại camera chất lượng cao cho tín hiệu ít nhiễu và có độ nhậy cao với ánh sáng. Khi chọn camera cần chú ý đến các thấu kính từ 18 đến 108 mm.

* **Bộ cảm biến ảnh:**

Máy chụp ảnh, camera có thể ghi lại hình ảnh (phim trong máy chụp, vidicon trong camera truyền hình). Có nhiều loại máy cảm biến (Sensor) làm việc với ánh sáng nhìn thấy và hồng ngoại như: Micro Densitometers, Image Dissector, Camera Divicon, linh kiện quang điện bằng bán dẫn. Các loại cảm biến bằng chụp ảnh phải số hoá là phim âm bản hoặc chụp ảnh. Camera divicon và linh kiện bán dẫn quang điện có thể cho ảnh ghi trên băng từ có thể số hoá. Trong Micro Densitometer phim và ảnh chụp được gắn trên mặt phẳng hoặc cuốn quang trống. Việc quét ảnh thông qua tia sáng (ví dụ tia Laser) trên ảnh đồng thời dịch chuyển mặt phim hoặc quang trống tương đối theo tia sáng. Trường hợp dùng phim, tia sáng đi qua phim.

Bây giờ chúng ta đề cập đến tất cả các khối trong hệ thống:

1. **Thiết bị nhận ảnh:**

Chức năng của thiết bị này là số hóa một băng tần số cơ bản của tớn hiệu truyền hình cung cấp từ một camera, hoặc từ một đầu máy VCR. Ảnh số sau đó được lưu trữ trong bộ đệm chính. Bộ đệm này có khả năng được địa chỉ hóa (nhờ một PC) đến từng điểm bằng phần mềm. Thông thường thiết bị này có nhiều chương trình con điều khiển để có thể lập trình được thông qua ngôn ngữ C. Khi mua một thiết cần chú ý cácc điểm sau:

* Thiết bị có khả năng số hóa ảnh ít nhất 8 bit (256 mức xỏm) và ảnh thu được phải có kích thước ít nhất là 512×512 điểm hoặc hơn.
* Thiết bị phải chứa một bộ đệm ảnh để lưu trữ một hoặc nhiều ảnh có độ phân giải 512×512 điểm ảnh.
* Thiết bị phải được kèm một bộ đầy đủ thư viện các chương trình con có khả năng giao diện với các chương trình C viết bằng Turbo C hoặc Microsoft C.
* Một số thiết bị cho phép tuỳ chọn sử dụng cả hai chế độ văn bản và đồ hoạ trên cùng một màn hình hoặc hai màn hình riêng biệt. Mặc dù chi tiết này là không cần thiết, nhưng nó sẽ rất có giá trị trong trường hợp bị giới hạn về không gian lắp đặt hoặc khả năng tài chính.

1. **Màn hình video:**



Hình 1.3: Hình ảnh màn hình video

Một số nhà sản xuất (như Sony) sản xuất các loại màn hình đen trắng chất lượng cao. Nên sử dụng loại màn hình chất lượng cao, vì màn hình chất lượng thấp có thể làm bạn nhầm lẫn kết quả. Một màn hình 9 inch là đủ cho yêu cầu làm việc. Để hiển thị ảnh màu, nên dùng một màn hình đa hệ.

1. **Máy tính:**



Hình 1.4: Hình ảnh máy tính

Cần có một máy tính Pentium 4 hoặc cấu hình cao hơn. Để chắc chắn, các máy này phải có sẵn các khe cắm cho phần xử lý ảnh. Các chương trình thiết kế và lọc ảnh có thể chạy trên bất kỳ hệ thống nào. Các chương trình con hiển thị ảnh dựng vỉ mạch VGA và có sẵn trên đĩa kèm theo. Các chương trình con hiển thị ảnh cũng hỗ trợ cho hầu hết các vỉ mạch SVGA.

Kết quả nhận dạng được lưu dưới dạng chuỗi các ký tự (character string) được đưa vào chương trình quản lý cơ sở dữ liệu biển số xe máy. Tại đây cán bộ trông xe sẽ tiến hành các thao tác nghiệp vụ như: in vé xe, thu tiền, lưu vào cơ sở dữ liệu. Tuỳ thuộc vào kết quả xử lý của chương trình quản lý hệ thống Barrier sẽ được tiến hành đóng mở tương ứng.

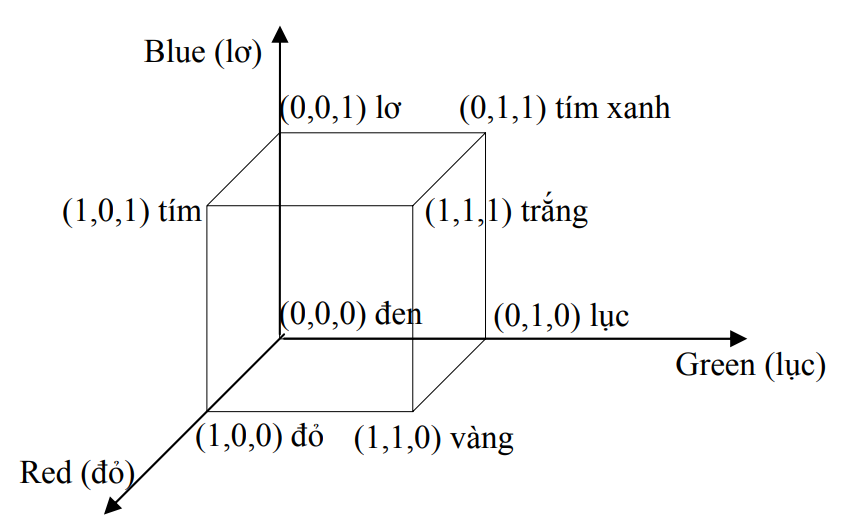
#### 1.2.2.2 Hệ tọa độ màu:

* **Khái niệm:**

Tổ chức quốc tế về chuẩn hóa màu CIE (Commission Internationale d’Eclairage) đưa ra một số chuẩn để biểu diễn màu. Các hệ này có các chuẩn riêng. Hệ chuẩn màu CIE-RGB dùng 3 màu cơ bản R, G, B và ký hiệu để phân biệt với các chuẩn khác. Như đã nêu trên, một màu là tổ hợp của các màu cơ bản theo một tỷ lệ nào đó. Như vậy, mỗi pixel ảnh màu ký hiệu , được viết:

=(T: ký hiệu chuyển vị)

Người ta dùng hệ tọa độ ba màu R-G-B (tương ứng với hệ tọa độ x-y-z) để biểu diễn màu như sau:



X

Y

Z

Hình 1.5: Hệ tọa độ RGB

Trong cách biểu diễn này ta có công thức:

đỏ + lục + lơ =1

Công thức này gọi là công thức Maxwell. Trong hình trên, tam giác tạo bởi ba đường đứt đoạn gọi là tam giác Maxwell. Màu trắng trong hệ tọa độ này được tính bởi:

=(++ ) = 1

#### 1.2.2.3 Một số phương pháp biểu diễn ảnh:

Sau bước số hóa, ảnh sẽ được lưu trữ hay chuyển sang giai đoạn phân tích. Trước khi đề cập đến vấn đề lưu trữ ảnh, cần xem xét ảnh sẽ được biểu diễn ra sao trong bộ nhớ máy tính. Dưới đây giới thiệu một số phương pháp biểu diễn thường dùng chi tiết và tường minh hơn:

- Biểu diễn mã loạt dài (Run-length Code)

- Biểu diễn mã xích (Chain Code)

- Biểu diễn mã tứ phân (Quad Tree Code)

#### 1.2.2.4 Các định dạng ảnh cơ bản:

* **Khái niệm chung:**

Ảnh thu được sau quá trình số hóa thường được lưu lại cho các quá trình xử lý tiếp theo hay truyền đi. Trong quá trình phát triển của kỹ thuật xử lý ảnh, tồn tại nhiều định dạng ảnh khác nhau từ ảnh đen trắng (với định dạng IMG), ảnh đa cấp xám cho đến ảnh màu: (BMP, GIF, JPEG…). Tuy các định dạng này khác nhau, song chúng đều tuân theo một cấu trúc chung nhất. Nhìn chung, một tệp ảnh bất kỳ thường bao gồm 3 phần:

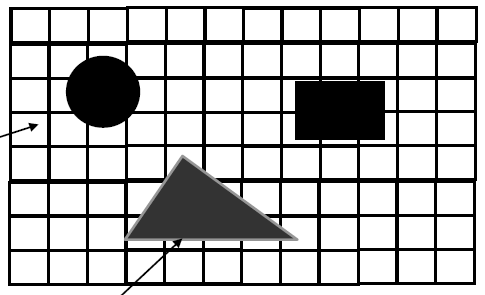
* Mào đầu tệp (Header)
* Dữ liệu nén (Data Compression)
* Bảng màu (Palette Color)
* **Quy trình đọc một tệp ảnh:**

Trong quá trình xử lý ảnh, đầu tiên phải tiến hành đọc tệp ảnh và chuyển vào bộ nhớ của máy tính dưới dạng ma trận số liệu ảnh. Khi lưu trữ dưới dạng tệp, ảnh là một khối gồm một số các byte. Để đọc đúng tệp ảnh ta cần hiểu ý nghĩa các phần trong cấu trúc của tệp ảnh như đã nêu trên. Trước tiên, ta cần đọc phần mào đầu (Header) để lấy các thông tin chung và thông tin điều khiển. Việc đọc này sẽ dừng ngay khi ta không gặp được chữ ký (Chữ ký ở đây thường được hiểu là một mã chỉ ra định dạng ảnh và đời (version) của nó) mong muốn.

Dựa vào thông tin điều khiển, ta xác định được vị trí bảng màu và đọc nó vào bộ nhớ. Cuối cùng, ta đọc phần dữ liệu nén. Sau khi đọc xong các khối dữ liệu ảnh vào bộ nhớ ta tiến hành nén dữ liệu ảnh. Căn cứ vào phương pháp nén chỉ ra trong phần Header ta giải mã được ảnh. Cuối cùng là khâu hiện ảnh. Dựa vào số liệu ảnh đã giải nén, vị trí và kích thước ảnh, cùng sự trợ giúp của bảng màu ảnh được hiện lên trên màn hình.

#### 1.2.2.5 Khái niệm ảnh đen trắng và ảnh màu:

Ảnh có thể biểu diễn dưới dạng tín hiệu tương tự hoặc tín hiệu số. Trong biểu diễn số của các ảnh đa mức xám, một ảnh được biểu diễn dưới dạng một ma trận hai chiều. Mỗi phần tử của ma trận biểu diễn cho mức xám hay cường độ của ảnh tại vị trí đó.



Độ sáng trung bình trong

mỗi hình chữ nhật = giá trị

một điểm ảnh.

Pixel or PEL

Hình 1.6: Biểu diễn mức xám của ảnh số

Trong Hình 1.6, một lưới chia ô vuông tưởng tượng được đặt lên ảnh. Độ lớn mỗi ô vuông của lưới xác định kích thước của một điểm ảnh. Mức xám của một điểm được tính bằng cường độ sáng trung bình tại mỗi ô vuông này. Mắt lưới càng nhỏ thì chất lượng ảnh càng cao. Trong kỹ thuật truyền hình tiên tiến, (mục đích là cung cấp cho người xem), hình ảnh cần chất lượng cao với độ phân giải gấp hai lần so với các chuẩn hiện nay.

* **Ảnh đen trắng:**

Ảnh đen trắng chỉ bao gồm 2 màu: màu đen và màu trắng. Người ta phân mức đen trắng đó thành L mức. Nếu sử dụng số bit *B=8 bít* để mã hóa mức đen trắng (hay mức xám) thì *L* được xác định :

L= (trong ví dụ này ta có B=8 bit nên L=256 mức)

Nếu *L* bằng 2, *B=1*, nghĩa là chỉ có 2 mức: mức *0* và mức *1*, còn gọi *là ảnh nhị phân*. Mức 1 ứng với màu sáng, còn mức 0 ứng với màu tối. Nếu L lớn hơn 2 ta có ảnh đa cấp xám. Nói cách khác, với ảnh nhị phân mỗi điểm ảnh được mã hóa trên 1 bit, còn với ảnh 256 mức, mỗi điểm ảnh được mã hóa trên 8 bit. Như vậy, với ảnh đen trắng: nếu dùng 8 bit (1 byte) để biểu diễn mức xám, số các mức xám có thể biểu diễn được là 256. Mỗi mức xám được biểu diễn dưới dạng là một số nguyên nằm trong khoảng từ 0 đến 255, với mức 0 biểu diễn cho mức cường độ đen nhất và 255 biểu diễn cho mức cường độ sáng nhất. Ảnh nhị phân khá đơn giản, các phần tử ảnh có thể coi như các phần tử logic. Ứng dụng chính của nó được dùng theo tính logic để phân biệt đối tượng ảnh với nền hay để phân biệt điểm biên với điểm khác.

* **Ảnh màu:**

Ảnh màu theo lý thuyết của Thomas là ảnh tổ hợp từ 3 màu cơ bản: đỏ (R), lục (G), lơ (B) và thường thu nhận trên các dải băng tần khác nhau. Với ảnh màu, cách biểu diễn cũng tương tự như với ảnh đen trắng, chỉ khác là các số tại mỗi phần tử của ma trận biểu diễn cho ba màu riêng rẽ gồm: đỏ (red), lục (green) và lam (blue). Để biểu diễn cho một điểm ảnh màu cần 24 bit. 24 bit này được chia thành ba khoảng 8 bit. Mỗi màu cũng phân thành *L* cấp màu khác nhau (thường *L=256*). Mỗi khoảng này biểu diễn cho cường độ sáng của một trong các màu chính.

Do đó, để lưu trữ ảnh màu người ta có thể lưu trữ từng màu riêng biệt, mỗi màu lưu trữ như một ảnh đa cấp xám. Do đó, không gian nhớ dành cho một ảnh màu lớn gấp 3 lần một ảnh đa cấp xám cùng kích cỡ.

### 1.2.3 Phát hiện ảnh

#### 1.2.3.1 Giới thiệu về bài toán phát hiện ảnh

* **Một số khái niệm:**

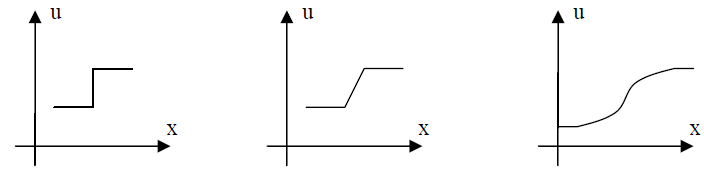
*Điểm biên:* Một điểm ảnh được coi là điểm biên nếu có sự thay đổi nhanh hoặc đột ngột về mức xám (hoặc màu). Ví dụ trong ảnh nhị phân, điểm đen gọi là điểm biên nếu lân cận nó có ít nhất một điểm trắng.

*Đường biên (đường bao: boundary):* tập hợp các điểm biên liên tiếp tạo thành một đường biên hay đường bao.

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

*Tầm quan trọng của biên:* để thấy rõ tầm quan trọng của biên, xét ví dụ sau: khi người họa sỹ muốn vẽ một danh nhân, họa sỹ chỉ cần vẽ vài đường nứt tốc họa mà không cần vẽ một cách đầy đủ.

*Mô hình biểu diễn đường biên, theo toán học:* điểm ảnh có sự biến đổi mức xám u(x) một cách đột ngột theo hình dưới:

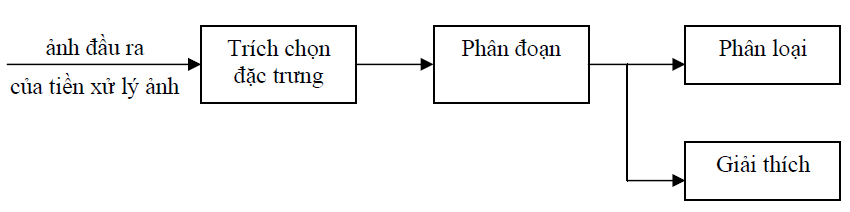


1. Đường biên lý tưởng b) Đường biên bậc thang c) Đường biên thực

Hình 1.7: Đường biên của ảnh

Các khái niệm và định nghĩa tóm tắt trên là cơ sở giúp ta hiểu và dùng để hiểu cách xây dựng, thiết kế các kỹ thuật phát hiện biên ảnh.

*Chú ý:* Phát hiện biên là một phần trong phân tích ảnh, sau khi đã lọc ảnh (hay tiền xử lý ảnh). Các bước của phân tích ảnh có thể mô tả theo sơ đồ dưới đây. Việc dò và tìm biên ảnh là một trong các đặc trưng thuộc khối trích chọn đặc trưng.

****

Hình 1.8: Các bước xử lí và phân tích ảnh

#### 1.2.3.2 Các phương pháp phát hiện ảnh

Từ định nghĩa toán học của biên người ta sử dụng hai phương pháp phát hiện biên như sau (phương pháp chính)

1. ***Phương pháp phát hiện biên trực tiếp****:* phương pháp này chủ yếu dựa vào sự biến thiên độ sáng của điểm ảnh để làm nổi biên bằng kỹ thuật đạo hàm.

* Nếu lấy đạo hàm bậc nhất của ảnh: ta có phương pháp Gradient
* Nếu lấy đạo hàm bậc hai của ảnh: ta có phương pháp Laplace.

Hai phương pháp này được gọi chung là phương pháp dò biên cục bộ.

Ngoài ra, người ta còn sử dụng phương pháp “đi theo đường biên” dựa vào công cụ toán học là nguyên lý quy hoạch động và được gọi là phương pháp dò biên tổng thể. Phương pháp dò biên trực tiếp có hiệu quả và ít bị tác động của nhiễu.

1. ***Phương pháp phát hiện biên gián tiếp***: Nếu bằng cách nào đấy, chúng ta thu được các vùng ảnh khác nhau thì đường phân cách giữa các vùng đó chính là biên. Nói cách khác, việc xác định đường biên của ảnh được thực hiện từ ảnh đã được phân vùng. Phương pháp dò biên gián tiếp khó cài đặt nhưng áp dụng tốt khi sự biến thiên độ sáng nhỏ.

*Chú ý*: Kỹ thuật dò biên và phân vùng ảnh là hai bài toán đối ngẫu của nhau.

* **Quy trình phát hiện biên:**

Bước 1: Do ảnh ghi được thường có nhiễu, bước một là phải lọc nhiễu.

Bước 2: Làm nổi biên sử dụng các toán tử phát hiện biên.

Bước 3: Định vị biên. Chú ý rằng kỹ thuật nổi biên gây tác dụng phụ là gây nhiễu làm một số biên giả xuất hiện do vậy cần loại bỏ biên giả.

Bước 4: Liên kết và trích chọn biên.

### 1.2.4 Phân loại ảnh

#### 1.2.4.1 Giới thiệu về bài toán phân loại ảnh

Phân loại hình ảnh (Image classification) hay Nhận dạng hình ảnh (Image recognition) là một trong những tác vụ của thị giác máy tính, ở đó thuật toán xem xét và dán nhãn cho hình ảnh từ một tập danh mục được xác định và đào tạo trước.

Ví dụ, với một tập các hình ảnh, mỗi hình ảnh mô tả một con mèo hoặc một con chó, thuật toán sẽ “quan sát” toàn bộ dữ liệu và dựa trên hình dạng, màu sắc để hình thành giả thuyết liên quan đến nội dung của ảnh. Kết quả thu được là từ tập dữ liệu ban đầu, các hình ảnh chó/mèo đã được phân loại một cách tự động.

Thực tế, thị giác góp phần tạo nên 80-85% nhận thức của con người về thế giới. Hàng ngày, mỗi người phải thực hiện phân loại trên bất kỳ dữ liệu hình ảnh nào mà chúng ta bắt gặp. Do đó, mô phỏng nhiệm vụ phân loại với sự trợ giúp của mạng nơ-ron là một trong những ứng dụng đầu tiên của thị giác máy tính mà các nhà nghiên cứu nghĩ đến.

#### 1.2.4.2 Các kỹ thuật phân loại ảnh

Có nhiều thuật toán khác nhau được ứng dụng trong việc phân loại hình ảnh. Các thuật toán này được chia thành hai nhóm chính là Học có giám sát (supervised learning) và Học không giám sát (unsupervised learning).

* **Phân loại có giám sát**

Trong học máy có giám sát, thuật toán được huấn luyện trên một tập hình ảnh đã được dán nhãn. Từ dữ liệu mẫu này, thuật toán có thể trích xuất thông tin, phục vụ phân loại ngay cả những hình ảnh chưa từng nhìn thấy trước đó.

* *Phân loại nhãn đơn*

Phân loại nhãn đơn (Single-label classification) là tác vụ phổ biến nhất trong phân loại ảnh có giám sát. Theo đó, mỗi hình ảnh được đại diện bởi một nhãn/chú thích (a single label or annotation). Mô hình xuất ra một giá trị hoặc dự đoán duy nhất cho mỗi hình ảnh mà nó xử lý. Đầu ra từ mô hình là mã hóa One-hot (từng giá trị được biến đổi thành các đặc trưng nhị phân chỉ chứa giá trị 1 hoặc 0). Mã hóa One-hot có độ dài bằng số lớp và giá trị biểu thị xác suất hình ảnh thuộc về lớp này. Hàm Softmax được sử dụng để đảm bảo các xác suất tổng bằng một và xác suất tối đa được chọn làm đầu ra của mô hình. Mặc dù Softmax không có giá trị về mặt dự đoán, nhưng nó giúp ràng buộc đầu ra giữa 1 và 0, nhờ vậy, có thể đánh giá độ tin cậy của mô hình từ điểm Softmax. Một số ví dụ về bộ dữ liệu phân loại nhãn đơn bao gồm MNIST, SVHN, ImageNet, v.v. Phân loại nhãn đơn có thể được xếp vào phân loại đa lớp (Multiclass classification) hoặc phân loại nhị phân (binary classification).

* Phân loại đa nhãn

Phân loại đa nhãn là một tác vụ phân loại trong đó mỗi hình ảnh có thể chứa nhiều hơn một nhãn hoặc một số hình ảnh chứa đồng thời tất cả các nhãn. Phân loại đa nhãn xuất hiện phổ biến trong lĩnh vực xử lý hình ảnh y tế, khi một bệnh nhân có thể được chẩn đoán mắc nhiều bệnh dựa trên dữ liệu chụp X-quang.

* **Phân loại không giám sát**

Trong học máy không giám sát, thuật toán chỉ sử dụng dữ liệu thô để đào tạo.Các nhãn phân loại thường không xuất hiện trong kiểu học này và mô hình học bằng cách nhận dạng các mẫu trong tập dữ liệu huấn luyện.

Giống như phân loại có giám sát, các phương pháp dựa trên không giám sát cũng liên quan đến bước trích xuất đặc điểm với các thông tin chi tiết nhất về hình ảnh. Các đặc điểm này sau đó được xử lý bằng các phương pháp phân cụm tham số (Gaussian Mixture Models) và phi tham số (K-means) hoặc các thuật toán học không giám sát khác.

Các thuật toán và kỹ thuật phân loại của thị giác máy tính không chỉ giới hạn trong dữ liệu hình ảnh 2D đơn giản mà còn mở rộng ra dưới dạng Video và ảnh 3D.

* Phân loại video

Khác với Phân loại hình ảnh, vốn chỉ sử dụng các thuật toán Xử lý ảnh và Mạng thần kinh tích chập (CNN), các tác vụ Phân loại video sử dụng cả dữ liệu hình ảnh và dữ liệu tạm thời (liên quan đến thời gian). Để có thể áp dụng trực tiếp thuật toán phân loại hình ảnh tiêu chuẩn, mô hình phân loại video sẽ khai thác mối quan hệ giữa các khung hình khác nhau. Theo đó, các mạng thần kinh phù hợp với dữ liệu chuỗi thời gian như LSTM (Bộ nhớ ngắn hạn dài) và RNN (Mạng thần kinh hồi quy) sẽ kết hợp với CNN để phân tích mối quan hệ thời gian giữa các khung.

* Phân loại 3D

Khác biệt cơ bản của phân loại 3D khi so với phân loại 2D nằm ở cấu trúc của CNN và bản chất chuyển động của hạt nhân trượt (sliding kernel). Nhân trong phân loại dữ liệu 3D cũng là 3D và di chuyển dọc theo cả ba trục, khác với chuyển động thẳng hai trục trong CNNs 2D. CNN có khả năng nắm bắt dữ liệu không gian rất tốt, và do đó dễ dàng xử lý khi dữ liệu được đặt cách nhau trên ba trục. Bộ dữ liệu phân loại 3D dễ dàng được tìm thấy trong lĩnh vực y tế (vi dụ:  ảnh cộng hưởng từ não) và dữ liệu cấu trúc của các đại phân tử thu được từ Kính hiển vi điện tử lạnh.

### 1.2.5 Phân vùng ảnh

#### 1.2.5.1 Giới thiệu về bài toán phân vùng ảnh

Phân vùng ảnh là bước then chốt trong xử lý ảnh. Giai đoạn này nhằm phân tích ảnh thành những thành phần có cùng tính chất nào đó dựa theo biên hay các vùng liên thông. Tiêu chuẩn để xác định các vùng liên thông có thể là cùng mức xám, cùng màu hay cùng độ nhám... Trước hết cần làm rõ khái niệm "vùng ảnh" (Segment) và đặc điểm vật lý của vùng.

Vùng ảnh là một chi tiết, một thực thể trông toàn cảnh. Nó là một tập hợp các điểm có cùng hoặc gần cùng một tính chất nào đó : mức xám, mức màu, độ nhám… Vùng ảnh là một trong hai thuộc tính của ảnh. Nói đến vùng ảnh là nói đến tính chất bề mặt. Đường bao quanh một vùng ảnh (Boundary) là biên ảnh. Các điểm trong một vùng ảnh có độ biến thiên giá trị mức xám tương đối đồng đều hay tính kết cấu tương đồng.

Dựa vào đặc tính vật lý của ảnh, người ta có nhiều kỹ thuật phân vùng: phân vùng dựa theo miền liên thông gọi là phân vùng dựa theo miền đồng nhất hay miền kề; phân vùng dựa vào biên gọi là phân vùng biên. Ngoài ra còn có các kỹ thuật phân vùng khác dựa vào biên độ, phân vùng dựa theo kết cấu.

#### 1.2.5.2 Các phương pháp phân vùng ảnh

* **Phân vùng ảnh theo ngưỡng biên độ:**

Các đặc tính đơn giản, cần thiết nhất của ảnh là biên độ và các tính chất vật lý như : độ tương phản, độ truyền sáng, màu sắc hoặc đáp ứng phổ.

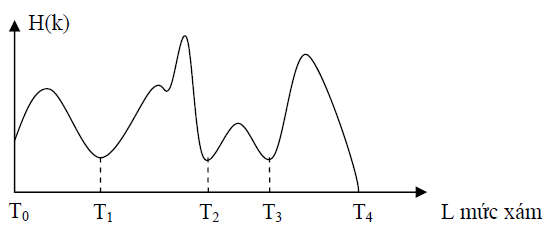
Như vậy, có thể dùng ngưỡng biên độ để phân vùng khi biên độ đủ lớn đặc trưng cho ảnh. Thí dụ, biên độ trong bộ cảm biến ảnh hồng ngoại có thể phản ánh vùng có nhiệt độ thấp hay vùng có nhiệt độ cao. Kỹ thuật phân ngưỡng theo biên độ rất có lợi đối với ảnh nhị phân như văn bản in, đồ họa, ảnh màu hay ảnh X-quang.

Việc chọn ngưỡng rất quan trọng. Nó bao gồm các bước :

* Xem xét lược đồ xám của ảnh để xác định các đỉnh và các khe. Nếu ảnh có dạng rắn lượn (nhiều đỉnh và khe), các khe có thể dùng để chọn ngưỡng.
* Chọn ngưỡng t sao cho một phần xác định trước η của toàn bộ số mẫu là thấp hơn t.
* Điều chỉnh ngưỡng dựa trên lược đồ xám của các điểm lân cận.
* Chọn ngưỡng theo lược đồ xám của những điểm thỏa mãn tiêu chuẩn chọn.Thí dụ, với ảnh có độ tương phản thấp, lược đồ của những điểm có biên độ Laplace g(m,n) lớn hơn giá trị t định trước (sao cho từ 5% đến 10% số điểm ảnh với Gradient lớn nhất sẽ coi như biên) sẽ cho phép xác định các đặc tính ảnh lưỡng cực tốt hơn ảnh gốc.
* Khi có mô hình phân lớp xác suất, việc xác định ngưỡng dựa vào tiêu chuẩn xác

suất nhằm cực tiểu xác suất sai số hoặc dựa vào một số tính chất khác của luật Bayes.

* Để hiểu rõ hơn nguyên tắc phân vùng dựa vào ngưỡng biên độ, xét thí dụ sau:



Hình 1.9: Lược đồ rắn lượn và cách chọn ngưỡng

Giả sử ảnh có lược đồ xám như hình 1.9, chọn các ngưỡng như hình trên với:

*=,…,=.* Ta có 5 ngưỡng và phân ảnh thành 4 vùng, ký hiệu là vùng thứ *k* của ảnh, *k=1,2,3,4*. Cách phân vùng theo nguyên tắc :

*P(m,n)* ∈  *nếu -1 ≤ P(m,n) < , k=1,2,3,4.*

Khi phân vùng xong, nếu ảnh rõ nét thì việc phân vùng coi như kết thúc. Nếu không, cần điều chỉnh ngưỡng.

* **Phân vùng ảnh theo miền đồng nhất:**

Kỹ thuật phân vùng ảnh thành các miền đồng nhất dựa vào các tính chất quan trọng nào đó của miền ảnh. Việc lựa chọn các tính chất của miền sẽ xác định tiêu chuẩn phân vùng. Tính đồng nhất của một miền ảnh là điểm chủ yếu xác định tính hiệu quả của việc phân vùng. Các tiêu chuẩn hay được dùng là sự thuần nhất về mức xám, màu sắc đối với ảnh màu, kết cấu sợi và chuyển động. Các phương pháp phân vùng ảnh theo miền đồng nhất thường áp dụng là :

* Phương pháp tách cây tứ phân
* Phương pháp cục bộ
* Phương pháp tổng hợp
* **Phân vùng ảnh theo kết cấu bề mặt:**

1. Phương pháp cấu trúc:

Kết cấu sợi có cấu trúc thuần nhất là những texels xác định, mà sự xuất hiện lặp đi lặp lại tuân theo một luật tất định hay ngẫu nhiên nào đấy. Một texel về thực tế là một nhóm các điểm ảnh có cùng một số tính chất bất biến lặp trên ảnh. Một texel cũng có định nghĩa theo mức xám, theo bề mặt hay tính đồng nhất đối với một số các tính chất như kích thước, hướng, lược đồ bậc hai (ma trận tương tranh).

Với các texel được phân bố ngẫu nhiên, tính kết cấu sợi tương ứng của nó được coi là yếu (Weak) ngược với qui luật phân bố tất định gọi là khỏe (Strong). Khi tính kết cấu sợi là yếu, luật phân bố có thể đo bởi:

* Mật độ gờ
* Các loạt dài của các texel liên thông tối đa
* Mật độ cực trị tương đối; số pixel trên một đơn vị diện tích có mức xám cực trị địa phương so với các lân cận.

Ngoài hai cách tiếp cận trên, người ta còn dùng cách tiếp cận khác bằng cách lấy tổ hợp 2 cách trên và gọi là kỹ thuật *mosaic*. Mô hình này biểu diễn các quá trình học ngẫu nhiên, thí dụ như khảm ngẫu nhiên hay đều của một mặt phẳng vào các đường cong lồi sẽ làm nổi lên tính kết cấu tế bào.

1. Tiếp cận theo tính kết cấu:

Khi đối tượng xuất hiện trên một nền có tính kết cấu cao, việc phân đoạn dựa vào tính kết cấu trở nên quan trọng. Nguyên nhân là kết cấu sợi thường chứa mật độ cao các gờ (edge) làm cho phân đoạn theo biên kém hiệu quả, trừ phi ta loại tính kết cấu. Việc phân đoạn dựa vào miền đồng nhất cũng có thể áp dụng cho các đặc trưng kết cấu và có thể dùng để phân đoạn các miền có tính kết cấu.

Nhìn chung, việc phân loại và phân vùng dựa vào kết cấu là một vấn đề phức tạp. Ở đây, tài liệu chỉ mang tính chất giới thiệu. Có thể giải quyết vấn đề này trong thực tế nếu ta biết trước các loại kết cấu (dựa vào quy luật hay các phân bố của nó).

### 1.2.6 Nhận dạng ảnh

#### 1.2.6.1 Giới thiệu về bài toán nhận dạng ảnh

Nhận dạng ảnh là giai đoạn cuối của các hệ thống xử lý ảnh. Nhận dạng ảnh dựa trên lý thuyết nhận dạng (Pattern Recognition) đã được đề cập trong nhiều sách về nhận dạng. Trong lý thuyết về nhận dạng nói chung và nhận dạng ảnh nói riêng có ba cách tiếp cận khác nhau:

- Nhận dạng dựa vào phân hoạch không gian.

- Nhận dạng dựa vào cấu trúc.

- Nhận dạng dựa vào kỹ thuật mạng nơron.

Hai cách tiếp cận đầu là cách tiếp cận kinh điển. Các đối tượng ảnh quan sát và thu nhận được phải trải qua giai đoạn tiền xử lý nhằm tăng cường chất lượng, làm nổi các chi tiết, tiếp theo là trích chọn và biểu diễn các đặc trưng, cuối cùng mới là giai đoạn nhận dạng. Cách tiếp cận thứ ba hoàn toàn khác. Nó dựa vào cơ chế đoán nhận, lưu trữ và phân biệt đối tượng mô phỏng theo hoạt động của hệ thần kinh con người. Do cơ chế đặc biệt, các đối tượng thu nhận bởi thị giác người không cần qua giai đoạn cải thiện mà chuyển ngay sang giai đoạn tổng hợp, đối sánh với các mẫu đã lưu trữ để nhận dạng.

* **Khái niệm nhận dạng.**

Nhận dạng là quá trình phân loại các đối tượng được biểu diễn theo một mô hình nào đó và gán chúng một tên (gán cho đối tượng một tên gọi, tức là một dạng) dựa theo những quy luật và mẫu chuẩn. Quá trình nhận dạng dựa vào những mẫu học biết trước gọi là *nhận dạng có thầy* hay học có thầy, trong những trường hợp ngược lại gọi *là học không có thầy*.

* **Bản chất của quá trình nhận dạng:**

Quá trình nhận dạng gồm 3 giai đoạn chính:

* Chọn mô hình biểu diễn đối tượng.
* Chọn luật ra quyết định (phương pháp nhận dạng) và suy diễn.
* Học trong nhận dạng.

Trong việc lựa chọn để biểu diễn đối tượng, đối tượng có thể được xác định theo cách định lượng (mô hình tham số) hay định tính (mô hình cấu trúc). Khi đối tượng đã được xác định, quá trình nhận dạng chuyển sang giai đoạn thứ hai-giai đoạn học (Learning). Học là giai đoạn cung cấp tri thức cho hệ thống. Mục đích học nhằm cải thiện, điều chỉnh việc phân loại tập đối tượng thành các lớp. Nhận dạng là tìm ra quy luật và các thuật toán để có thể gắn đối tượng vào một lớp hay nói một cách khác gán cho đối tượng một tên.

#### 1.2.6.2 Các phương pháp nhận dạng ảnh

* **Nhận dạng theo cấu trúc:**

Các đối tượng cần nhận dạng theo phương pháp này được biểu diễn bởi một câu trong ngôn ngữ L(G). Khi đó thao tác phân lớp chính là xem xét một đối tượng có thuộc văn phạm L(G) không. Nói cách khác nó có được sinh ra bởi các luật của văn phạm G hay không.

Như vậy sự phân lớp là theo cách tiếp cận cấu trúc đòi hỏi phải xác định:

* Xác định tập *V1* chung cho tất cả mọi đối tượng
* Xác định các quy tắc *P* để sản sinh ra một câu và chúng khác nhau đối với mỗi lớp.
* Ra quyết định: xác định một đối tượng *X* được biểu diễn bởi một câu *.* Nếu nhận biết bởi *L()* thì ta nói rằng *X* là một đối tượng thuộc loại *.*

Nói cách khác, việc ra quyết định phân lớp là dựa vào phân tích cú biểu diễn lớp . Pháp của văn phạm. Cũng như trong phân tích cú pháp ngôn ngữ, có phân tích trên xuống, dưới lên, việc nhận dạng theo cấu trúc cũng có thể thực hiện theo cách tương tự.

* **Nhận dạng dựa theo mạng Nơron:**

1. **Giới thiệu mạng nơron:**

Nghiên cứu về nơron thần kinh từ lâu đã trở thành đề tài được nhiều nhà khoa học quan tâm. Nhưng kỷ nguyên của mạng nơron chính thức được bắt đầu với báo cáo khoa học của Mc Culloch và Pitts năm 1943 miêu tả một phép tính logic của mạng nơron. Báo cáo này được công chúng đón nhận cho đến năm 1949 học thuyết về mạng nơron chính thức của Mc. Culloch và Pitts được mô tả chủ yếu trong bài giảng thứ hai trong bốn bài giảng mà Von Neumann đã phát biểu tại trường đại học Illinois. Sự phát triển tiếp theo của quá trình nghiên cứu mạng nơron được đánh dấu vào năm 1949 với việc xuất bản cuốn sách “*The Organization of Behavior: A neuropsychological Theory*” của Donald Olding Hebb. Đến năm 1952, cuốn sách của Ashby “*Design for a Brain*” đã mô tả những điều kiện cần và đủ đối với một hệ thống hoạt động giống như bộ não “đó là phải học để còn tồn tại trong môi trường luôn thay đổi và nhận được những cái nó cần”. Tiếp đó, năm 1954, Minsky đã viết luận án tiến sĩ mang tên *“Theory of Neural-analog Reinforcement Systems and Application to Brain-Model Problem”* tại trường Đại học Princeton, sau đó là bài báo của ông *“Steps Toward Artificial Interligence”* năm 1961 về việc học củng cố trong mạng nơron hiện nay. Một chủ đề khác được đánh giá cao là phát kiến về bộ nhớ liên kết của Taylor vào năm 1956, mở đầu một loạt các kết quả phát triển to lớn về sau. Các kết quả có thể kể đến là sự ra đời của mạng Perceptron được Frank Rosenblatt công bố vào năm 1957 và được coi là “mạng nơron truyền thẳng đơn giản nhất”. Tiếp đó năm 1960, mạng nơron khác được Bernard Widrow và Marcian Hoff giới thiệu là ADALINE (ADAptive LINear Element). Với mạng ADALINE lần đầu tiên kiểu hội tụ các mạch con chứa trọng số trước node tổng được sử dụng để phân lớp các mẫu. Năm 1969, Minsky và Papert xuất bản cuốn *“Perceptron, An Introduction to Computational Geometry”* chỉ ra những giới hạn trong mạng Perceptron một lớp đơn và đề nghị khắc phục trong mạng Perceptron nhiều lớp. Những năm 1970 nổi bật với các sự kiện: các mạng liên kết của Kohonen và Anderson (1972), Cognitron – mạng tự tổ chức nhiều lớp đầu tiên do Kunihiko Fukushima người Nhật giới thiệu năm 1975. Thời kì phát triển vượt bậc của quá trình nghiên cứu mạng nơron là những năm 80 với sự ra đời của một loạt các mạng nơron có giá trị: mạng Hopfied của John Hopfield (1982), SOM (Self-Organization Map) của Kohonen, máy Boltzmann của Ackley, Hinton và Sejnowski. Tiếp theo là sự ra đời của mạng Back- Propagation năm 1986 do D. Rumelhart, G. Hilton và R. Williams giới thiệu, sau đó là mạng ART (Adaptive Resonance Networks) vào năm 1987 của Gail Carpenter và Stephen Grossberg tại đại học Boston. Trong những năm 1990 việc ứng dụng các mô hình mạng nơron vào phục vụ các lĩnh vực trong cuộc sống được phát triển mạnh mẽ.

1. **Cấu trúc Nơron sinh học:**

Nơron sinh vật có nhiều dạng khác nhau như dạng hình tháp, dạng tổ ong, dạng rễ cây. Tuy khác nhau về hình dạng, chúng có cấu trúc và nguyên lý hoạt động chung. Một tế bào nơron gồm bốn phần cơ bản:

*Các nhánh và rễ:* Các nhánh và rễ là các bộ phận nhận thông tin, các đầu nhạy hoặc các đầu ra của các nơron khác bám vào rễ hoặc nhánh của một nơron. Khi các đầu vào từ ngoài này có sự chênh lệch về nồng độ K+ , Na+ hay Cl- so với nồng độ bên trong của nó thì xẩy ra hiện tượng thấm từ ngoài vào trong thông qua một cơ chế màng thấm đặc biệt. Hiện tượng thẩm thấu như vậy tạo nên một cơ chế truyền đạt thông tin với hàng ngàn hàng vạn lối vào trên một nơron sinh vật, ứng với hàng nghìn hàng vạn liên kết khác nhau. Mức độ thẩm thấu được đặc trưng bởi cơ chế màng tượng trưng bằng một tỷ lệ. Tỷ lệ đó được gọi là tỷ trọng hay đơn giản gọi là trọng (Weight).

*Thân thần kinh (Soma):* Thân thần kinh chứa các nhân và cơ quan tổng hợp protein. Các ion vào được tổng hợp và biến đổi. Khi nồng độ các ion đạt đến một giá trị nhất định, xảy ra quá trình phát xung (hay kích thích). Xung đó được phát ở các đầu ra của nơron. Dây dẫn đầu ra xung được gọi là dây thần kinh.

*Dây thần kinh (Axon):* Dây thần kinh là đầu ra. Đó là phương tiện truyền dẫn tín hiệu. Dây thần kinh được cấu tạo gồm các đốt và có thể dài từ micro mét đến vài mét tuỳ từng kết cấu cụ thể. Đầu ra này có thể truyền tín hiệu đến các nơron khác.

Mạng nơron sinh vật tổ chức thành từng lớp (layer). Ta có:

* Mạng một lớp: là tập hợp các phần tử nơron có đầu vào và đầu ra trên mỗi một phần tử. Nếu mạng nối đầu ra của các phần tử này với đầu vào của phần tử kia gọi là mạng tự liên kết (autoassociative).
* Mạng hai lớp: gồm một lớp đầu vào và một lớp đầu ra riêng biệt.
* Mạng nhiều lớp: gồm một lớp đầu vào và một lớp đầu ra riêng biệt. Các lớp nằm giữa lớp đầu vào và lớp đầu ra gọi là lớp ẩn (hidden layers).
* Mạng truyền thẳng: là mạng hai hay nhiều lớp mà quá trình truyền tín hiệu từ đầu ra lớp này đến đầu vào lớp kia theo một hướng.
* Mạng truyền ngược: là mạng mà trong đó một hoặc nhiều đầu ra của các phần tử lớp sau truyền ngược tới đầu vào của lớp trước.
* Mạng tự tổ chức: là mạng có khả năng sử dụng những kinh nghiệm của quá khứ để thích ứng với những biến đổi của môi trường (không dự báo trước). Loại mạng này thuộc nhóm hệ học, thích nghi không cần có tín hiệu chỉ đạo từ bên ngoài.

Bảng sau đây đưa ra những con số cụ thể để có thể so sánh những ưu việt của bộ não con người so với một máy tính cá nhân mức trung bình hiện nay.

Bảng 1.1 So sánh khả năng làm việc của bộ não và máy tính

|  |  |  |
| --- | --- | --- |
|  | Máy tính | Bộ não người |
| Đơn vị tính toán | Bộ xử lý trung tâm với 105 mạch logic | Mạng 1011 nơron |
| Bộ nhớ | 109 bit RAM | 1011 nơron |
| Bộ nhớ ngoài | Trên 1010 bit bộ nhớ ngoài | Với 1014 khớp nối thần kinh |
| Thời gian xử lý | 10-8 giây | 10-3 giây |
| Thông lượng | 109 bit/giây | 1014 bit/giây |
| Cập nhật thông tin | 105 bit/giây | 1014 nơron/giây |

Những nghiên cứu trên đây là cơ sở mở đường cho con người nghiên cứu phương pháp mô phỏng lại mạng nơron sinh vật, đó chính là mạng nơron nhân tạo.

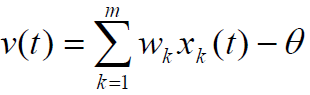
1. **Cấu trúc nơron nhân tạo:**

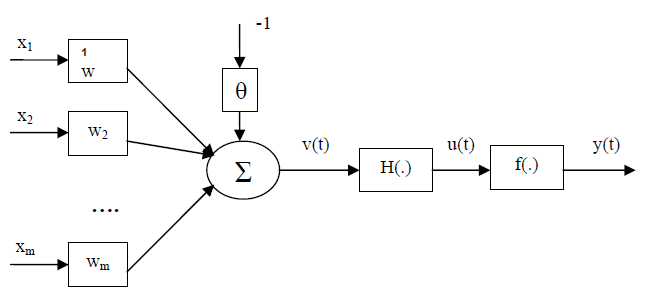
Trên cơ sở cấu trúc của nơron sinh học tổng quát người ta đề xuất mô hình nơron nhân tạo gồm 3 phần chính:

* Bộ tổng liên kết đầu vào
* Bộ động học tuyến tính
* Bộ phi tuyến

***Bộ tổng liên kết đầu vào:***

Là bộ tổng hợp các liên kết tại đầu vào của một phần tử nơron, mô tả như sau:





Hình 1.10: Mô hình nơ-ron nhân tạo

Trong đó:

V(t): Tổng tất cả các đầu vào mô tả toàn bộ thế năng tác động ở thân nơron.

Các đầu vào ngoài; với k là chỉ số chạy, k=1,2…m.

*m:* Số lượng đầu vào mô tả tín hiệu vào từ các đầu nhạy thần kinh

: Trọng liên kết ngoài giữa các đầu vào k tới nơron hiện tại.

y(t): Tín hiệu đầu ra nơron.

*θ*: Ngưỡng (là hằng số), xác định ngưỡng kích thích hay ức chế.

## 1**.3 Tổng quan về Deep Learning và bài toán nhận dạng biển số xe**

### 1.3.1 Tổng quan về Deep Learning

#### **1.3.1.1 Giới thiệu về Deep learning**

**Deep Learning**là một tập hợp con của học máy, về cơ bản là một mạng nơ-ron có ba lớp trở lên. Những mạng lưới thần kinh này cố gắng mô phỏng hành vi của não người cho phép deep learning “học” từ một lượng lớn dữ liệu. Trong khi mạng nơ-ron có một lớp vẫn có thể đưa ra các dự đoán gần đúng, các lớp ẩn bổ sung có thể giúp tối ưu hóa và tinh chỉnh để có độ chính xác. Deep Learning thúc đẩy nhiều ứng dụng và dịch vụ trí tuệ nhân tạo (AI) nhằm cải thiện tự động hóa, thực hiện các tác vụ phân tích và vật lý mà không cần sự can thiệp của con người. Công nghệ deep learning được ứng dụng rộng rãi trong các sản phẩm và dịch vụ hàng ngày chẳng hạn như trợ lý kỹ thuật số, điều khiển từ xa hỗ trợ giọng nói và phát hiện gian lận thẻ tín dụng cũng như các công nghệ mới nổi chẳng hạn như ô tô tự lái.

#### **1.3.1.2 Cách thức hoạt động của Deep learning**

**Học sâu deep learning** hoạt động bằng cách khám phá các cấu trúc phức tạp trong dữ liệu mà chúng trải nghiệm. Cụ thể là xây dựng các mô hình tính toán bao gồm nhiều lớp xử lý, mạng có thể tạo ra nhiều mức trừu tượng để biểu diễn dữ liệu. Ví dụ, một mô hình học sâu được gọi là mạng nơ-ron phức hợp có thể được đào tạo bằng cách sử dụng số lượng lớn (hàng triệu) hình ảnh, chẳng hạn như những hình ảnh có chứa mèo. Loại mạng nơ-ron này thường học hỏi từ các pixel có trong hình ảnh mà nó thu được. Nó có thể phân loại các nhóm pixel đại diện cho các đặc điểm của mèo, với các nhóm đặc điểm như móng vuốt, tai và mắt cho biết sự hiện diện của mèo trong hình ảnh. Đối với nhiều tác vụ, chẳng hạn như thị giác máy tính, nhận dạng giọng nói còn được gọi là xử lý ngôn ngữ tự nhiên, dịch máy và robot, hiệu suất của các hệ thống deep learning vượt xa so với các hệ thống machine learning. Điều này không có nghĩa là việc xây dựng các hệ thống học sâu là tương đối dễ dàng so với các hệ thống học máy thông thường. Mặc dù nhận dạng tính năng tự trị trong học sâu, hàng nghìn siêu tham số (nút bấm) cần được điều chỉnh để mô hình học sâu deep learning trở nên hiệu quả.

#### **1.3.1.3 Ưu điểm của Deep learning**

* **Tự động hóa các tính năng**
* **Tương thích tốt với kiểu dữ liệu phi cấu trúc**
* **Khả năng tự học tốt hơn**
* **Lợi thế về thuật toán phân tán và song song**
* **Tối ưu chi phí**
* **Phân tích nâng cao**
* **Khả năng mở rộng**

#### **1.3.1.4 Ứng dụng của Deep learning**

Một số ứng dụng của deep learning hiện nay, cụ thể:

* Người máy: Sự phát triển của khoa học công nghệ trong thời gian gần đây, đặc biệt trong lĩnh vực người máy được thúc đẩy bởi những tiến bộ trong AI và deep learning.
* Nông nghiệp: Ngày nay, deep learning cho phép nông dân triển khai thiết bị có thể nhìn thấy và phân biệt giữa cây trồng và cỏ dại. Khả năng này cho phép máy làm cỏ phun thuốc diệt cỏ một cách có chọn lọc lên cỏ dại và để các cây khác không bị ảnh hưởng.
* Hình ảnh y tế và chăm sóc sức khỏe: Học sâu đang có những bước tiến đáng kể vào việc cải thiện chất lượng, chăm sóc sức khỏe bằng cách dự đoán các sự kiện y tế từ dữ liệu hồ sơ sức khỏe điện tử.

### 1.3.2 Bài toán nhận diện biển số xe

#### 1.3.2.1 Giới thiệu về bài toán nhận dạng biển số xe:

Bài toán nhận dạng biển số xe là một trong những bài toán quan trọng trong lĩnh vực thị giác máy tính và xử lý ảnh, với ứng dụng rộng rãi trong lĩnh vực giao thông, an ninh, và quản lý đô thị. Mục tiêu của bài toán này là tự động nhận dạng và phân loại biển số xe trên hình ảnh hoặc video, giúp cải thiện hiệu suất và hiệu quả của các hệ thống giao thông, giám sát an ninh, và quản lý xe cộ.

Tầm quan trọng của bài toán nhận dạng biển số xe:

* An ninh giao thông: Phát hiện và nhận dạng biển số xe có thể giúp trong việc giám sát giao thông, xử lý vi phạm giao thông, và tăng cường an ninh đô thị.
* Quản lý đô thị: Nhận dạng biển số xe có thể được sử dụng để kiểm soát luồng xe cộ, quản lý đỗ xe, và thu thập dữ liệu về giao thông.
* Ứng dụng trong giao thông thông minh: Cung cấp thông tin về lưu lượng giao thông và dự đoán tình trạng giao thông để cải thiện việc đi lại trong thành phố.

#### 1.3.2.2 Các phương pháp nhận dạng biển số xe:

* Phương pháp dựa trên đặc trưng:

Sử dụng các kỹ thuật xử lý ảnh và trích xuất đặc trưng để phát hiện và nhận dạng vùng chứa biển số xe. Phương pháp này thường dựa vào các kỹ thuật như lọc thông tin, phân đoạn, và trích xuất đặc trưng hình học của biển số xe.

* Phương pháp dựa trên máy học:

Sử dụng các mô hình học máy, đặc biệt là học sâu (deep learning), để nhận dạng và phân loại biển số xe. Các mô hình học sâu như mạng nơ-ron tích chập (CNN) thường được sử dụng để tự động học và nhận dạng các đặc điểm của biển số xe.

#### 1.3.2.3 Khó khăn và thách thức:

* Biến thể về biển số: Sự đa dạng về kích thước, màu sắc, font chữ, và điều kiện ánh sáng của biển số xe có thể tạo ra thách thức trong việc nhận dạng chính xác.
* Nhiễu và ảnh nền: Nhiễu từ môi trường xung quanh và ảnh nền có thể làm cho quá trình nhận dạng trở nên khó khăn và phức tạp.
* Hiệu suất và tốc độ: Yêu cầu về hiệu suất và tốc độ của hệ thống nhận dạng biển số xe là một thách thức đối với các ứng dụng thời gian thực.

## 1.4 Các kết quả nghiên cứu trong và ngoài nước

### 1.4.1 Một số nghiên cứu đã và đang thực hiện tại nước ta

- Nghiên cứu đề xuất mô hình Yolo V5 ứng dụng trong nhận dạng biển số xe của nhóm tác giả Nguyễn Thành Lợi, Đào Xuân Phúc, Nguyễn Thị Tố Uyên , Nguyễn Hữu Phát thuộc trường Đại học Mở Hà Nội [3]*.* Nội dung bài nghiên cứu bày về các vấn đề liên quan đến mô hình Yolov5, bao gồm nguyên lý hoạt động, áp dụng mô hình để đào tạo cho dữ liệu từ đó nhận diện biển số xe và đánh giá mô hình. Kết quả chỉ ra mô hình có độ chính xác cao : mô hình nhận diện biển số xe độ chính xác trên tập val Precision đạt 99.7% và Recall đạt 99.9%,mô hình nhận diện kí tự đạt trên 97% với từng lớp kí tự (0-9 , A-Z)

- Bài nghiên cứu về “Kĩ thuật nhận dạng biển số xe và áp dụng vào bài toán quản lí bãi giữ xe” của nhóm tác giả Trần Thị Hương và Ngô Thị Kiều Hằng tại trường đại học Hà Tĩnh [6]. Nhóm tác giả nghiên cứu xây dựng hệ thống gửi xe thông minh phục vụ cho trường Đại Học Hà Tĩnh, Hệ thống quản lý vé xe gồm 3 chức năng chính: Quản lý dữ liệu, nhận dạng xe vào ra bãi, thống kê tìm kiếm thông tin. Chương trình đã đưa vào thử nghiệm trên

cả xe máy và ô tô, kết quả nhận diện chính xác biển số. In vé xe tự động, trích mã QR của vé, tránh trường hợp làm vé giả. Thực hiện lưu thời gian xe vào, và thời gian trả xe, tìm kiếm xe gửi theo vé xe. Thống kê số lượng xe gửi theo ngày, tuần, tháng. Chương trình đáp ứng yêu cầu đưa ra: nhận dạng biển số xe và thực hiện quá trình quản lý vé xe. Thử nghiêm 100 mẫu biển số xe khác nhau, khi chụp đúng góc, đủ ánh sáng thì hệ thống nhận dạng đúng 99%, 1 mẫu xe không nhận diện được do biển số xe bị mờ số, phải tự đánh thủ công biển số vào hệ thống quản lý

### 1.4.2 Các kết quả nghiên cứu nước ngoài

- Bài báo "End-to-End License Plate Detection and Recognition via Deep Neural Networks" của C. S. Chan, C. H. Chen, M. H. Nguyen và M. Savvides [4], công bố vào năm 2015 là một nghiên cứu đột phá trong lĩnh vực nhận diện và nhận dạng biển số xe sử dụng mạng neural sâu. Nghiên cứu này tập trung vào việc phát triển một hệ thống tự động có khả năng phát hiện và nhận dạng biển số xe từ hình ảnh, với mục tiêu đạt được hiệu suất cao và độ chính xác đáng tin cậy. Bằng việc áp dụng kiến trúc mạng neural sâu, đặc biệt là mạng Convolutional Neural Network (CNN), hệ thống đề xuất đã đạt được kết quả ấn tượng trên nhiều bộ dữ liệu đánh giá, bao gồm cả các bộ dữ liệu phổ biến như OpenALPR và SVHN dataset. Kết quả thử nghiệm đã chứng minh rằng phương pháp đề xuất không chỉ đạt được độ chính xác cao mà còn có độ ổn định và khả năng áp dụng thực tế. Bằng cách kết hợp phát hiện và nhận dạng trong một quy trình end-to-end, nghiên cứu này đã mở ra những triển vọng mới trong việc ứng dụng các kỹ thuật học sâu vào lĩnh vực nhận diện biển số xe

- "Bài báo "Vehicle License Plate Recognition Based on Multifeature Fusion and Improved YOLOv3" của S. Guo, J. Yang, X. Wang và X. Dai [5], được công bố vào năm 2020, là một nghiên cứu quan trọng trong lĩnh vực nhận dạng biển số xe dựa trên việc kết hợp nhiều đặc điểm và cải tiến của mô hình YOLOv3. Nghiên cứu này tập trung vào việc phát triển một hệ thống nhận dạng biển số xe với hiệu suất cao và độ chính xác đáng tin cậy, bằng cách kết hợp nhiều đặc điểm khác nhau của ảnh và cải tiến YOLOv3. Bằng cách sử dụng một phương pháp kết hợp đa đặc điểm, bao gồm cả đặc điểm hình ảnh và các đặc điểm khác như màu sắc và hình dạng, cùng với việc cải tiến mô hình YOLOv3, nghiên cứu đã đạt được kết quả ấn tượng trong việc nhận dạng biển số xe từ hình ảnh. Kết quả thử nghiệm trên các bộ dữ liệu đánh giá đã chứng minh rằng phương pháp đề xuất không chỉ có độ chính xác cao mà còn có khả năng ứng dụng thực tế trong các tình huống đa dạng. Bài báo này đã đóng góp một phần quan trọng trong việc phát triển các phương pháp nhận dạng biển số xe thông qua việc kết hợp nhiều đặc điểm và cải tiến của mô hình YOLOv3.

### 1.4.3 Định hướng của đề tài

Định hướng nghiên cứu của đề tài "Xây dựng hệ thống phát hiện và phân loại biển số xe sử dụng các thuật toán deeplearning bao gồm các phần sau:

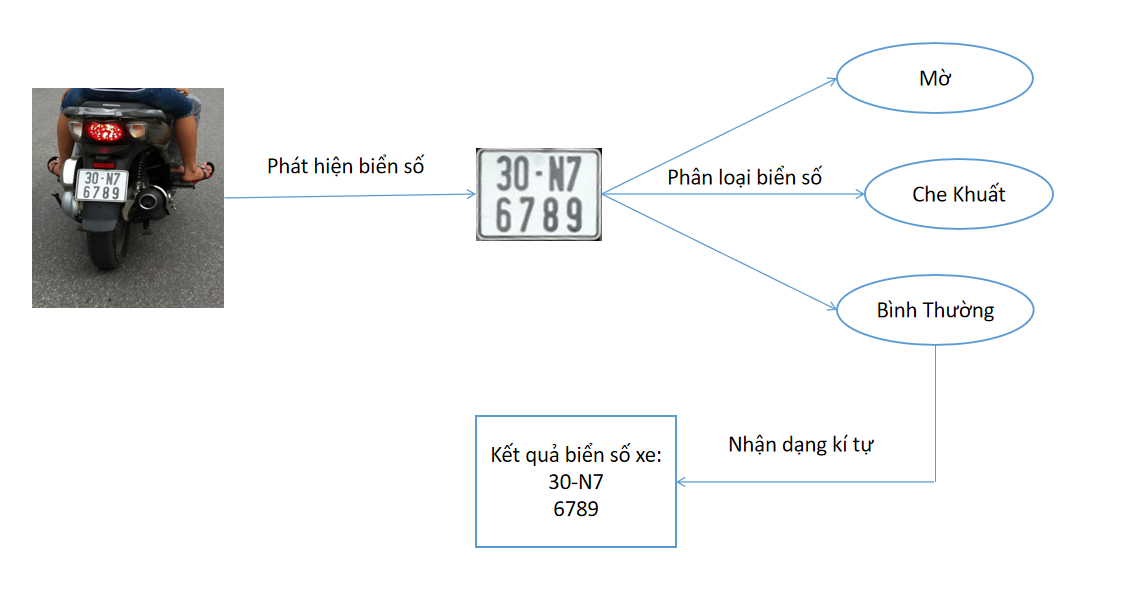
* Tổng quan về đề tài:
* Giới thiệu về vấn đề cụ thể mà hệ thống nhận diện và phân loại biển số xe nhằm giải quyết.
* Công nghệ và phương pháp:
* Trình bày các công nghệ và phương pháp chính được sử dụng trong đề tài.
* Giải thích về cách sử dụng mô hình YOLOv8 để nhận diện biển số xe và mô hình ResNet34 để phân loại biển số.
* Mô tả về việc sử dụng thư viện Easy OCR để nhận diện ký tự từ ảnh và giao diện người dùng được phát triển bằng Qt5 để tương tác với người dùng.
* Phương pháp triển khai:
* Trình bày chi tiết về quá trình triển khai của hệ thống, bao gồm thu thập dữ liệu, tiền xử lý dữ liệu, huấn luyện mô hình, triển khai giao diện người dùng và tích hợp các thành phần.
* Mô tả về quy trình phát triển và kiểm thử của hệ thống, bao gồm các bước kiểm thử chức năng và hiệu suất.
* Kết quả dự kiến:
* Đề xuất các kết quả dự kiến của đề tài, bao gồm độ chính xác của hệ thống trong việc nhận diện và phân loại biển số xe.
* Mô tả về các tính năng và khả năng của ứng dụng cuối cùng sau khi hoàn thành.
* Hướng phát triển tiếp theo:
* Đề xuất các hướng nghiên cứu và phát triển tiếp theo của đề tài, bao gồm việc cải thiện hiệu suất của hệ thống và mở rộng ứng dụng vào các lĩnh vực mới.

# CHƯƠNG II: HỆ THỒNG PHÁT HIỆN VÀ PHÂN LOẠI BIỂN SỐ XE SỬ DỤNG DEEPLEARNING

## 2.1 Khái quát chung về hệ thống phát hiện và phân loại biển số xe

Luồng xử lí của hệ thống phát hiện và phân loại biển số xe bao gồm các phần chính sau:

* ***Phát hiện biển số***: Khối này có chức năng tách biển số từ ảnh chụp bằng các phương pháp xử lý ảnh. Kết quả của khối là ảnh màu RBG (Red Green Blue) được cắt ra từ ảnh chụp. Đây là một công việc rất khó khăn vì ta không biết được vị trí chính xác của biển số. Ngoài ra, còn phụ thuộc vào độ sáng và chất lượng của ảnh chụp.
* ***Phân loại biển số***: Sau khi phát hiện và tách được biển số, ảnh trích xuất vùng biển số sẽ đi qua khối phân loại nhằm lọc và loại ra những biển số bị mờ và bị che khuất gây ảnh hưởng đến khối nhận diện kí tự
* ***Nhận dạng ký tự***: Sau khi phân loại, ảnh trích xuất vùng biển số đưa vào khối nhận dạng dạng để tiến hành nhận dạng từng ký tự trong chuỗi



Hình 2.1:Sơ đồ hệ thống phát hiện và phân loại biển số xe

## 2.3 Phát hiện biển số xe

### 2.3.1 Tổng quát bài toán phát hiện biển số xe

Bài toán phát hiện biển số xe là một phần quan trọng trong hệ thống nhận diện biển số xe, một ứng dụng của trí tuệ nhân tạo trong lĩnh vực thị giác máy tính. Mục tiêu chính của bài toán này là xác định vị trí và giới hạn của biển số xe trên các hình ảnh hoặc video. Quá trình này cung cấp thông tin cần thiết để nhận dạng và phân tích các biển số, từ đó hỗ trợ trong việc quản lý giao thông, an ninh, giám sát và nhiều ứng dụng khác.

Tính chất chính của bài toán phát hiện biển số xe bao gồm:

* Đa dạng về hình dạng và kích thước: Biển số xe có thể có nhiều loại hình dạng và kích thước, tùy thuộc vào loại phương tiện và quy định của từng quốc gia hoặc khu vực.
* Điều kiện ánh sáng và nền: Hệ thống phải xử lý các điều kiện ánh sáng khác nhau, từ ánh sáng ban ngày sáng chói đến ánh sáng yếu hoặc ánh sáng ban đêm. Ngoài ra, các nền phức tạp cũng đặt ra thách thức cho quá trình phát hiện.
* Che khuất và biến dạng: Biển số xe có thể bị che khuất bởi các vật thể khác hoặc bị biến dạng do góc chụp, tốc độ di chuyển hoặc điều kiện môi trường.
* Thời gian thực: Trong một số ứng dụng, hệ thống phải phát hiện và nhận dạng biển số xe một cách nhanh chóng và hiệu quả để đáp ứng yêu cầu về thời gian thực.
* Độ chính xác: Đối với các ứng dụng quan trọng như giám sát giao thông hoặc an ninh, độ chính xác của quá trình phát hiện là yếu tố quan trọng, đảm bảo không bỏ sót hoặc nhận diện sai các biển số.

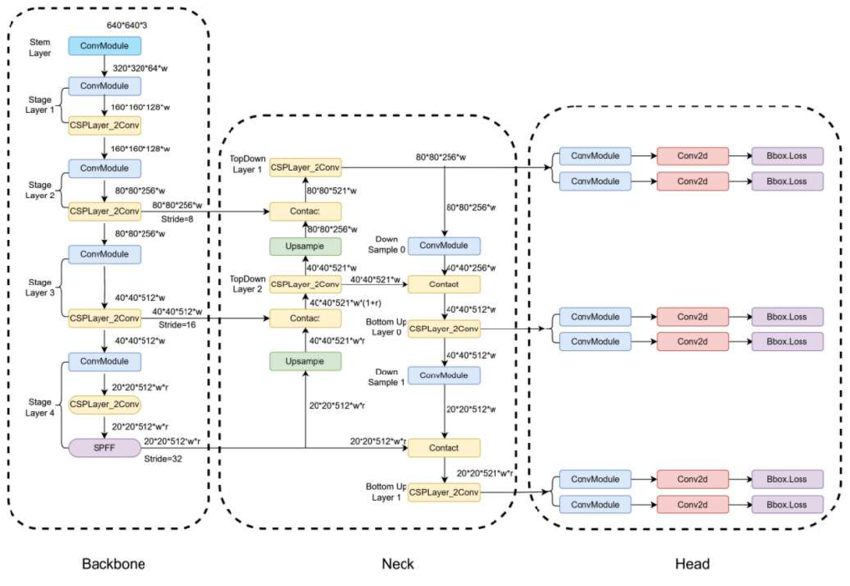
### 2.3.2 Giới thiệu thuật toán Yolov8 trong bài toán phát hiện đối tượng

#### 2.3.2.1. Tổng quan về thuật toán YOLOv8

YOLOv8 (You Only Look Once version 8) là một trong những thuật toán tiên tiến nhất trong lĩnh vực nhận dạng và phát hiện đối tượng trong ảnh và video. YOLOv8 là phiên bản cải tiến từ các phiên bản trước đó như YOLOv3 và YOLOv4, với hiệu suất và độ chính xác cao hơn.

#### 2.3.2.2. Nguyên Tắc Hoạt Động

* + ***Chia ảnh thành lưới ô vuông (Grid):***
* YOLOv8 chia mỗi ảnh thành một lưới ô vuông với kích thước cố định. Mỗi ô vuông trong lưới có khả năng dự đoán một hoặc nhiều đối tượng.
  + ***Dự đoán đối tượng trong mỗi ô vuông:***
* Mỗi ô vuông dự đoán một số lượng hộp giới hạn (bounding box) và xác suất tương ứng cho các lớp đối tượng có thể có trong ô vuông đó.
* Mỗi hộp giới hạn được biểu diễn bằng các thuộc tính: tọa độ của góc trên trái, chiều rộng, chiều cao, và xác suất dự đoán.
  + ***Kết hợp dự đoán trên toàn bộ ảnh:***
* Quá trình dự đoán được thực hiện song song trên toàn bộ ảnh, giúp YOLOv8 nhanh chóng và hiệu quả.
* Mỗi hộp giới hạn được đưa ra sau đó được kết hợp và lọc thông qua thuật toán giữ lại các hộp giới hạn quan trọng.
  + ***Đánh giá và lọc kết quả:***
* Kết quả dự đoán cuối cùng là tập hợp các hộp giới hạn đã được lọc, với mỗi hộp giới hạn có kèm theo xác suất của lớp đối tượng và tọa độ của hộp giới hạn đó.



Hình 2.2: Kiến trúc mô hình Yolov8

#### 2.3.2.3. Ưu Điểm

* Tốc độ nhanh: YOLOv8 thực hiện dự đoán một cách hiệu quả trên toàn bộ ảnh, giúp giảm thiểu thời gian xử lý.
* Độ chính xác cao: YOLOv8 đạt được độ chính xác cao trong việc nhận diện và phát hiện đối tượng, với khả năng nhận diện nhiều lớp đối tượng khác nhau.

#### 2.3.2.4. Ứng Dụng

* YOLOv8 được áp dụng rộng rãi trong nhiều lĩnh vực như bảo mật, tự động hóa, xe tự lái, nhận diện khuôn mặt, và nhận dạng biển số xe.
* Với tốc độ và độ chính xác cao, YOLOv8 đang trở thành một công cụ quan trọng trong lĩnh vực thị giác máy tính và ứng dụng trí tuệ nhân tạo.

### 2.3.3 Thu thập và gán nhãn dữ liệu

#### 2.3.3.1 Thu thập dữ liệu

* Thu thập ảnh chứa biển số xe từ nhiều nguồn khác nhau như camera giao thông, ảnh đường phố, tập hợp các ảnh từ bộ dữ liệu công cộng.
* Đảm bảo dữ liệu thu thập được đa dạng về điều kiện ánh sáng, góc chụp, và môi trường để mô hình có khả năng tổng quát hóa tốt.
* Nhãn chứa thông tin về vị trí và định dạng của biển số xe trên ảnh. Định dạng nhãn được lưu trong file .txt tương ứng với mỗi ảnh, trong đó mỗi dòng chứa thông tin về một biển số xe cụ thể. Thông tin này bao gồm tọa độ của bounding box và label tương ứng với biển số.

#### 2.3.3.2 Gán nhãn dữ liệu

* Sử dụng tool labelme và thư viện OpenCV để cắt ảnh và đánh dấu vị trí của biển số xe trên ảnh.
* Tạo các bounding box bao quanh biển số và lưu thông tin này vào file nhãn tương ứng.
* Thực hiện các bước tiền xử lý như resize, normalize, và augment để chuẩn hóa và tăng cường dữ liệu.
* Chia dữ liệu thành các tập train/validation/test để đánh giá hiệu suất của mô hình.

#### 2.3.3.3 Kết quả đạt được

* Đã thu thập được 100.000 hình ảnh thô các phương tiện giao thông bằng các phương thức như: crawl, internet, luồng camera, agument dữ liệu
* Dùng tool labelme để gán nhãn bouding box trả kết quả về file .json tương ứng. Đưa các file nhãn về kiểu file .txt theo đúng định dạng đầu vào của mô hình YOLO
* Xử lí ảnh trước khi đưa vào model bằng các phương pháp: rotate, resize, normalize
* Chia dữ liệu thành các tập train/validation/test theo tỉ lệ 7/2/1



Hình 2.3: Quá trình gán nhãn bouding box biển số xe

### 2.3.4 Đào tạo mô hình

* Sử dụng dữ liệu đã chuẩn bị để huấn luyện mô hình yolov8 trên máy chủ .
* Tiến hành huấn luyện trên các tập dữ liệu train và validation, đánh giá hiệu suất và điều chỉnh tham số để cải thiện mô hình. Đặt batch size của mô hình là 64 và số lượng epoch là 100
* Log kết quả mỗi epoch ra file .txt để theo dõix

## 2.4 Phân loại biển số xe

### 2.4.1 Tổng quát bài toán phân loại biển số xe

Bài toán phân loại biển số xe trong hệ thống nhận diện biển số xe là một phần quan trọng để xác định trạng thái và chất lượng của các biển số trên các hình ảnh hoặc video. Mục tiêu chính của bài toán này là phân loại biển số xe thành các loại: mờ, che khuất hoặc bình thường, dựa trên các đặc điểm như độ rõ nét, độ sáng và độ phân giải. Quá trình này giúp cải thiện hiệu suất của hệ thống nhận diện biển số xe và tăng khả năng nhận diện chính xác.

Tính chất chính của bài toán phân loại biển số xe bao gồm:

* Phân loại mờ: Các biển số mờ là những biển số bị mờ hoặc không rõ nét do nhiều nguyên nhân như chất lượng hình ảnh kém, điều kiện ánh sáng không tốt hoặc góc chụp không lý tưởng.
* Phân loại che khuất: Các biển số che khuất là những biển số bị che khuất bởi các vật thể khác trong hình ảnh, làm giảm khả năng nhận diện.
* Phân loại bình thường: Các biển số bình thường là những biển số có độ rõ nét cao và không bị che khuất, giúp quá trình nhận diện và phân loại diễn ra một cách chính xác.

### 2.4.2 Giới thiệu thuật toán Resnet34 trong bài toán phân loại

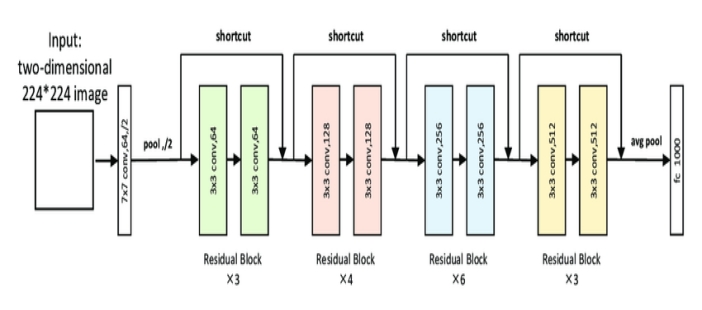
#### 2.4.2.1. Tổng quan về thuật toán Resnet34

ResNet34 (Residual Network 34) là một trong những kiến trúc mạng neural network sâu phổ biến và mạnh mẽ trong lĩnh vực computer vision. Được giới thiệu bởi Microsoft Research vào năm 2015, ResNet34 đã đạt được nhiều thành tựu trong việc giải quyết các bài toán phức tạp như nhận dạng đối tượng, phân loại ảnh và nhận dạng khuôn mặt.

#### 2.4.2.2. Nguyên Tắc Hoạt Động

ResNet34 dựa trên một kiến trúc mạng neural network sâu có khả năng học được các biểu diễn sâu của dữ liệu. Điểm đặc biệt của ResNet34 là sự sáng tạo trong cơ chế skip connections (còn được gọi là residual connections).

* ***Cơ Chế Skip Connections:***
* Trong các mạng neural network truyền thống, thông tin được truyền theo một luồng tuần tự từ lớp đầu vào đến lớp đầu ra. Khi mạng trở nên sâu, việc huấn luyện trở nên khó khăn do hiện tượng vanishing gradient, trong đó gradient giảm đáng kể khi lan truyền ngược từ lớp cuối cùng về lớp đầu vào.
* ResNet34 giải quyết vấn đề này bằng cách sử dụng các skip connections để tạo ra các đường dẫn shortcut giữa các lớp, cho phép thông tin trực tiếp lan truyền qua các lớp. Cụ thể, mỗi lớp được kết nối với một hoặc nhiều lớp kế tiếp thông qua các identity mappings, cho phép thông tin lan truyền thông suốt mà không bị mất đi hay biến dạng.
* Cơ chế skip connections giúp giảm thiểu hiện tượng vanishing gradient, giúp mô hình có khả năng học được những biểu diễn sâu hơn và tăng độ chính xác của mạng.
* ***Kiến Trúc Mạng ResNet34:***
* ResNet34 có kiến trúc gồm 34 lớp, bao gồm các lớp convolution, batch normalization, activation (thường là ReLU), và các residual blocks.
* Mỗi residual block bao gồm hai hoặc ba lớp convolution với kích thước kernel nhỏ, được kết hợp với các skip connections để tạo ra đường dẫn ngắn.
* Các residual block có thể được xếp chồng lên nhau để tạo thành một mạng sâu có khả năng học được các biểu diễn phức tạp của dữ liệu.



Hình 2.4: Kiến trúc mô hình Resnet34

#### 2.4.2.3. Ưu Điểm

Thuật toán ResNet34 mang lại nhiều ưu điểm đáng chú ý, làm nổi bật kiến trúc này trong lĩnh vực computer vision:

* ***Hiệu Quả Trong Việc Huấn Luyện Mạng Sâu:***
* Cơ chế skip connections giúp giảm thiểu hiện tượng vanishing gradient và exploding gradient, làm cho việc huấn luyện các mạng sâu trở nên dễ dàng hơn. Điều này cho phép mô hình học được các biểu diễn sâu của dữ liệu một cách hiệu quả hơn.
* ***Tính Tổng Quát và Khả Năng Tính Toán:***
* ResNet34 là một kiến trúc mạng neural network rất tổng quát và có khả năng học được các biểu diễn phức tạp của dữ liệu. Các residual blocks có thể được xếp chồng lên nhau để tạo ra các mạng sâu có khả năng học được các đặc trưng từ cấp thấp đến cấp cao.
* Kiến trúc của ResNet34 cũng tối ưu về mặt tính toán, với việc sử dụng các lớp convolution nhỏ và hiệu quả.

#### 2.4.2.4. Ứng Dụng

ResNet34 và các biến thể của nó được sử dụng rộng rãi trong nhiều ứng dụng trong computer vision, bao gồm:

* Phân loại ảnh: ResNet34 có khả năng phân loại các loại đối tượng trong ảnh với độ chính xác cao.
* Nhận dạng khuôn mặt: ResNet34 được sử dụng trong các hệ thống nhận dạng khuôn mặt tự động.
* Xử lý ảnh y tế: ResNet34 được áp dụng trong việc phát hiện bệnh lý từ hình ảnh y tế như chụp X-quang và MRI.
* Xử lý video: ResNet34 có thể được sử dụng trong việc phân loại và nhận dạng đối tượng trên video.

### 2.4.3 Thu thập và gán nhãn dữ liệu

#### 2.4.3.1 Thu thập Dữ liệu:

* Thu thập ảnh: Thu thập ảnh chứa các biển số xe từ nhiều nguồn khác nhau như camera giám sát giao thông, ảnh từ xe hội viên cộng đồng, hoặc cơ sở dữ liệu ảnh công cộng.
* Chú ý đến biển số mờ và bị che khuất: Đảm bảo rằng dữ liệu cũng bao gồm các ảnh chứa biển số xe bị mờ và bị che khuất đủ đa dạng và phản ánh thực tế.

#### 2.4.3.2 Gán nhãn:

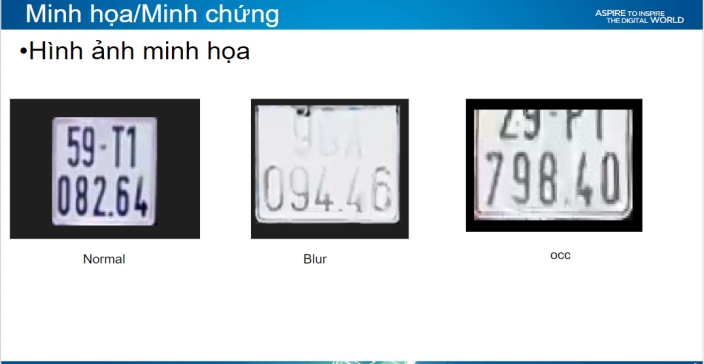
* Xác định biển số mờ và bị che khuất: Sử dụng các thuật toán hoặc kỹ thuật nhận dạng để xác định các ảnh có chứa biển số xe bị mờ và bị che khuất
* Nhãn hóa dữ liệu: Gán nhãn cho từng ảnh, đánh dấu liệu liệu có chứa biển số mờ ,bị che khuất hoặc bình thường bằng cách thêm nhãn vào file .txt chứa nhãn vào đường link dẫn đến ảnh tương ứng.

#### 2.4.3.3 Chuẩn bị Dữ liệu:

* Chia dữ liệu: Chia dữ liệu thành các tập huấn luyện, kiểm tra và validation để đảm bảo mô hình có thể tổng quát hóa tốt.
* Xử lí ảnh: Tiền xử lí ảnh nếu cần thiết như cắt, resize, hoặc xoay để chuẩn hóa dữ liệu và giảm nhiễu.

#### 2.4.3.4 Kết quả đạt được

* Đã thu thập được 10.000 hình ảnh thô các phương tiện giao thông bằng các phương thức như: crawl, internet, luồng camera, agument dữ liệu
* Dùng tool tự tạo để gán nhãn trả kết quả về file .txt theo định dạng mõi dòng là đường link dẫn đến ảnh và nhãn tương ứng
* Xử lí ảnh trước khi đưa vào model bằng các phương pháp: rotate, resize, normalize
* Chia dữ liệu thành các tập train/validation/test theo tỉ lệ 7/2/1 cho mỗi bộ dữ liệu



Hình 2.5: Mẫu dữ liệu bài toán phân loại biển số xe

### 2.4.4 Đào tạo mô hình

* Sử dụng dữ liệu đã chuẩn bị để huấn luyện mô hình Resnet34 trên máy chủ
* Tiến hành huấn luyện trên các tập dữ liệu train và validation, đánh giá hiệu suất và điều chỉnh tham số để cải thiện mô hình. Đặt batch size của mô hình là 64 và số lượng epoch là 50
* Log kết quả mỗi epoch ra file .txt để theo dõi

## 2.5 Nhận dạng kí tự biển số xe

### 2.5.1 Tổng quát về module nhận dạng kí tự biển số xe

Module nhận dạng kí tự biển số xe trong hệ thống phát hiện và phân loại biển số xe đóng vai trò quan trọng trong quá trình nhận diện và phân loại biển số. Nhiệm vụ của module này là nhận diện và trích xuất các kí tự từ biển số đã được phát hiện và phân loại, sau đó nhận dạng các kí tự này để tạo thành chuỗi biển số hoàn chỉnh.

Một số đặc điểm chính của module nhận dạng kí tự biển số xe bao gồm:

* Nhận dạng đa ký tự: Module này có khả năng nhận dạng các kí tự có thể xuất hiện trong biển số xe, bao gồm chữ cái, chữ số và các ký tự đặc biệt như dấu gạch ngang, dấu cách.
* Độ chính xác cao: Yêu cầu của việc nhận dạng kí tự là độ chính xác cao để đảm bảo rằng các kí tự được nhận dạng chính xác, tránh sai sót trong việc đọc biển số.
* Xử lý ảnh: Module này sử dụng các kỹ thuật xử lý ảnh để làm sạch và tách biển số thành các kí tự riêng biệt, tăng cường độ rõ nét và khả năng nhận dạng.
* Tích hợp dữ liệu: Module này tích hợp dữ liệu về các kí tự biển số từ nhiều nguồn khác nhau để đào tạo và kiểm thử mô hình nhận dạng.

### 2.5.2 Giới thiệu thư viện mã nguồn mở EasyOCR

#### 2.5.2.1 Giới thiệu tổng quan

EasyOCR là một thư viện nhận dạng ký tự (OCR) mã nguồn mở, được thiết kế để dễ dàng sử dụng và linh hoạt trong việc nhận dạng văn bản từ ảnh. Thư viện này được phát triển bởi một nhóm nghiên cứu và phát triển phần mềm tại Trung Quốc, và đã trở thành một trong những lựa chọn phổ biến cho các ứng dụng nhận dạng văn bản.

#### 2.5.2.2 Nguyên lí hoạt động

EasyOCR hoạt động dựa trên một kiến trúc mạng nơ-ron sâu, thường sử dụng mạng nơ-ron tích chập (CNN), để nhận dạng văn bản từ ảnh. Quá trình này bao gồm các bước sau:

* Tiền xử lý ảnh: Ảnh đầu vào được tiền xử lý để chuẩn bị cho quá trình nhận dạng, bao gồm việc chuyển đổi sang không gian màu thích hợp, cân bằng độ sáng và cắt ảnh thành các vùng quan trọng.
* Trích xuất đặc trưng: EasyOCR sử dụng mạng nơ-ron tích chập để trích xuất các đặc trưng quan trọng từ ảnh, như các đường cong và góc của các ký tự.
* Nhận dạng ký tự: Sau khi trích xuất đặc trưng, một mô hình học máy được sử dụng để dự đoán các ký tự trong ảnh. Mô hình này đã được huấn luyện trước trên các tập dữ liệu lớn chứa các ảnh chứa văn bản.

#### 2.5.2.3 Ưu điểm

Ưu điểm của EasyOCR:

* Dễ sử dụng: Thư viện này có API đơn giản và trực quan, cho phép người dùng dễ dàng tích hợp vào các ứng dụng Python của họ.
* Hiệu suất cao: EasyOCR sử dụng các mô hình học sâu tiên tiến để đạt được hiệu suất nhận dạng văn bản tốt trên nhiều loại ảnh và điều kiện ánh sáng khác nhau.
* Hỗ trợ nhiều ngôn ngữ: Thư viện hỗ trợ nhận dạng văn bản trong nhiều ngôn ngữ khác nhau trên thế giới, bao gồm cả các ngôn ngữ có ký tự đặc biệt như tiếng Trung, tiếng Nhật và tiếng Hàn.
* Tính linh hoạt và tùy chỉnh: EasyOCR là một thư viện mã nguồn mở, cho phép cộng đồng phát triển và cải thiện thư viện theo nhu cầu cụ thể của họ.

#### 2.5.2.4 Ứng dụng của EasyOCR:

* Nhận dạng biển số xe: EasyOCR có thể được sử dụng để nhận dạng biển số xe từ ảnh, giúp trong việc giám sát giao thông và quản lý xe cộ.
* Quét tài liệu: Thư viện này cũng có thể được sử dụng để quét và nhận dạng văn bản từ tài liệu giấy hoặc tệp ảnh, giúp trong việc tự động hóa việc xử lý tài liệu.
* Nhận diện văn bản trong ứng dụng di động: EasyOCR có thể được tích hợp vào các ứng dụng di động để nhận dạng văn bản từ ảnh chụp bằng camera của điện thoại di động, giúp trong việc tạo ra các ứng dụng ghi chú, quét danh thiếp, và nhiều ứng dụng khác.

### 2.5.3 Tích hợp thư viện EasyOCR vào module nhận diện biển số xe

Sau giai đoạn phân loại, tiếp theo là quá trình nhận dạng ký tự từ các ảnh đã được tách ra từ biển số xe, và trong đó chúng ta sử dụng thư viện mã nguồn mở EasyOCR. Dưới đây là các bước chi tiết trong quá trình này:

* Chuẩn bị dữ liệu đầu vào: Các ảnh chứa từng ký tự đã được tách ra từ biển số xe sẽ được sử dụng làm dữ liệu đầu vào cho quá trình nhận dạng. Đảm bảo rằng các ảnh này đã được tiền xử lý đủ để phù hợp với yêu cầu của EasyOCR.
* Sử dụng EasyOCR: Sử dụng thư viện EasyOCR để nhận dạng các ký tự từ các ảnh đã được chuẩn bị. EasyOCR cung cấp một giao diện đơn giản để sử dụng và hỗ trợ nhận dạng ký tự trên nhiều ngôn ngữ và kích thước font khác nhau.
* Thực hiện nhận dạng: Gọi hàm nhận dạng từ EasyOCR và truyền vào ảnh chứa ký tự. Thư viện sẽ tiến hành quá trình nhận dạng và trả về kết quả dưới dạng các chuỗi ký tự.
* Xử lý kết quả: Kết quả trả về từ EasyOCR sẽ được xử lý để loại bỏ các ký tự không hợp lệ hoặc không cần thiết và sắp xếp lại theo thứ tự đúng của biển số xe.
* Hiển thị kết quả: Kết quả cuối cùng của quá trình nhận dạng ký tự là chuỗi ký tự hoàn chỉnh của biển số xe. Chuỗi này có thể được hiển thị lên giao diện người dùng hoặc sử dụng cho các mục đích khác như lưu trữ hoặc xử lý tiếp theo.



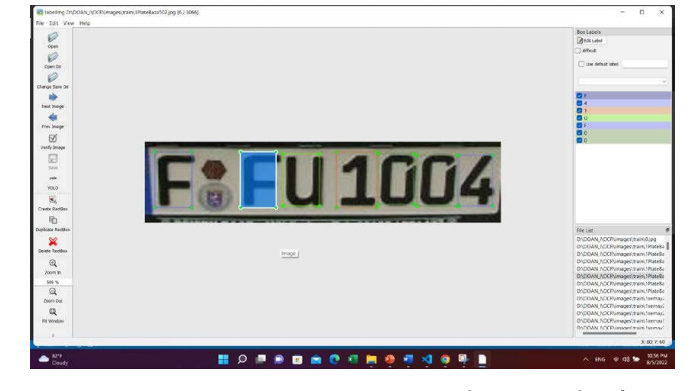
Hình 2.6: Sử dụng thư viện EasyOCR nhận diện kí tự biển số xe

## 2.2 Phân tích thiết kế hệ thống phát hiện và phân loại biển số xe

### 2.2.1 Phân tích một số hệ thống nhận diện biển số xe hiện nay

* Nghiên cứu đề xuất mô hình Yolo V5 ứng dụng trong nhận dạng biển số xe

Mô hình do nhóm tác giả Nguyễn Thành Lợi, Đào Xuân Phúc, Nguyễn Thị Tố Uyên [3]. Kết quả ra mô hình có độ chính xác cao: mô hình nhận diện biển số xe độ chính xác trên tập val Precision đạt 99.7% và Recall đạt 99.9%, mô hình nhận diện kí tự đạt trên 97% với từng lớp kí tự (0-9, A-Z)



Hình 2.7: Quá trình gán nhãn mô hình nhận diện kí tự

* *Ưu điểm*

-Tốc độ xử lý nhanh: sử dụng thuật toán Yolov5 giúp nhận diện đối tượng tốt trong thời gian thực.

- Độ chính xác của mô hình nhận diện kí tự tốt

* *Nhược điểm:*

-Độ chính xác của chương trình nhận dạng biển số xe phụ thuộc vào ánh sáng môi trường và chất lượng biển số xe

-Chưa phân loại được các biển nhiễu không thể nhận dạng

* Hệ thống quản lí bãi xe thông minh Đại Học Hà Tĩnh

Hệ thống do nhóm tác giả Trần Thị Hương và Ngô Thị Kiều Hằng tại trường đại học Hà Tĩnh [6] nghiên cứ và thực hiện. Thử nghiêm 100 mẫu biển số xe khác nhau, khi chụp đúng góc, đủ ánh sáng thì hệ thống nhận dạng đúng 99%, 1 mẫu xe không nhận diện được do biển số xe bị mờ số, phải tự đánh thủ công biển số vào hệ thống quản lý



Hình 2.8: Giao diện hệ thống quản lí bãi xe

* *Ưu điểm*

-Tốc độ xử lý nhanh: chương trình được viết trên giao diện ứng dụng Visual Studio 2012 bằng ngôn ngữ lập trình C#, kết hợp mã nguồn mở OpenCV nên tốc độ xử lý nhận dạng tương đối nhanh so với việc sử dụng các ngôn ngữ lập trình khác và công cụ hỗ trợ khác.

- Giao diện đơn giản, dễ sử dụng: giao diện được thiết kế dựa trên công cụ MFC có sẵn trong phần mềm Visual Studio với các nút lệnh trực

quan, dễ dàng thao tác.

- Tính kinh tế cao: Các giai đoạn quản lý, quá trình lưu trữ thống kê được thực hiện tự động, giảm thời gian tính toán thủ công.

* *Nhược điểm:*

- Độ chính xác của chương trình nhận dạng biển số xe phụ thuộc vào ánh sáng môi trường và chất lượng biển số xe

- Chưa phân loại được các biển nhiễu không thể nhận dạng

### 2.2.2 Yêu cầu,mục tiêu và ưu điểm của hệ thống

* Yêu cầu của hệ thống:
* Chính xác và đáng tin cậy: Hệ thống cần có khả năng nhận diện chính xác và đáng tin cậy các biển số xe, bao gồm cả trong các điều kiện ánh sáng khác nhau ,ở các góc độ khác nhau và phân loại được những biển khó đọc được thông tin .
* Tính thời gian thực: Yêu cầu hệ thống phải hoạt động ở tốc độ nhanh và có khả năng xử lý ảnh thời gian thực để phản ứng kịp thời với các tình huống giao thông.
* Khả năng tương thích và tích hợp: Hệ thống cần có khả năng tương thích với các loại camera khác nhau và có thể tích hợp vào các hệ thống giao thông tự động hiện có.
* Bảo mật và quyền riêng tư: Bảo mật thông tin liên quan đến biển số xe là rất quan trọng. Hệ thống cần có cơ chế bảo mật để đảm bảo an toàn thông tin cá nhân.
* Mục tiêu của hệ thống:
* Nâng cao hiệu suất giao thông: Mục tiêu chính của hệ thống là nâng cao hiệu suất giao thông bằng cách giúp giảm ùn tắc giao thông, tăng cường an toàn và giảm vi phạm luật giao thông.
* Tăng cường an ninh: Hệ thống giúp tăng cường an ninh giao thông bằng cách phát hiện và ghi nhận các hành vi vi phạm và xe bị truy nã.
* Tối ưu hóa quản lý giao thông: Dựa vào dữ liệu thu thập được từ hệ thống, các cơ quan quản lý giao thông có thể thực hiện các biện pháp điều tiết giao thông một cách hiệu quả hơn.
* Cải thiện trải nghiệm người dùng: Đối với người dùng cuối, mục tiêu là cung cấp một trải nghiệm thuận tiện và an toàn khi tham gia giao thông, bằng cách cung cấp thông tin và dịch vụ hữu ích.
* Ưu điểm của hệ thống so với các hệ thống khác:
* Xử lí trích xuẩt ảnh và xoay ảnh bằng thuật toán deeplearning nên sẽ cho ra đầu ra chứa vùng biển số chất lượng nhất và không bị méo ảnh
* Có module phân loại biển số xe để loại ra những biển số xe kém chất lượng ra khỏi quá trình nhận dạng khiến hệ thống chạy mượt và chính xác hơn

### 2.2.3 Thiết kế kiến trúc hệ thống

* Tiền xử lý dữ liệu: Dữ liệu hình ảnh thu thập được cần được tiền xử lý để chuẩn bị cho quá trình nhận diện và phân loại. Các bước tiền xử lý có thể bao gồm cắt ảnh, làm sạch nhiễu, điều chỉnh độ tương phản, cân bằng màu sắc, và các biến đổi khác.
* Phát hiện biển số xe: Sau khi xử lí dự liệu , ảnh sẽ được đi qua module phát hiện biển số xe để phát hiện và trích xuất vị trí của biển số xe trong ảnh
* Phân loại biển số xe: Sau khi vị trí của biển số xe đã được xác định, hệ thống phân loại biển số vào các loại khác nhau như biển số bình thường, biển số mờ hoặc biển số bị che khuất. Điều này giúp tăng cường hiệu suất nhận diện và phân loại sau này.
* Nhận diện kí tự: Khi biển số xe đã được phát hiện và phân loại, hệ thống tiếp tục nhận diện các ký tự trên biển số.
* Xử lý kết quả: Kết quả cuối cùng sau quá trình nhận diện và phân loại là một chuỗi các ký tự trên biển số xe. Hệ thống có thể tiếp tục xử lý kết quả này bằng cách lưu trữ, gửi đi thông báo hoặc kích hoạt các hành động khác tùy thuộc vào yêu cầu cụ thể.

### 2.2.4 Thiết kế cơ sở dữ liệu

Hệ thống nhận diện và phân loại biển số xe và lưu trữ dữ liệu vào file Excel, thông tin sẽ được lưu với các trường dữ liệu sau:

* ID: Mã số duy nhất định danh cho mỗi bản ghi trong cơ sở dữ liệu.
* Vị trí: Một chuỗi ký tự hoặc cặp tọa độ để lưu trữ vị trí vật lý hoặc vị trí trên hình ảnh của biển số xe.
* Loại biển số: Trạng thái của biển số xe sau khi phân loại, có thể là "Bình thường", "Mờ", "Che khuất"
* Ký tự: Một chuỗi ký tự để lưu trữ các ký tự được nhận diện từ biển số xe.

### 2.2.5 Thiết kế giao diện người dùng

Phân tích thiết kế giao diện hệ thống phát hiện và phân loại ảnh sử dụng thư viện mã nguồn mở Qt5 –thư viện hỗ trợ xây dựng ứng dụng desktop bằng pythn, trung vào các yếu tố sau:

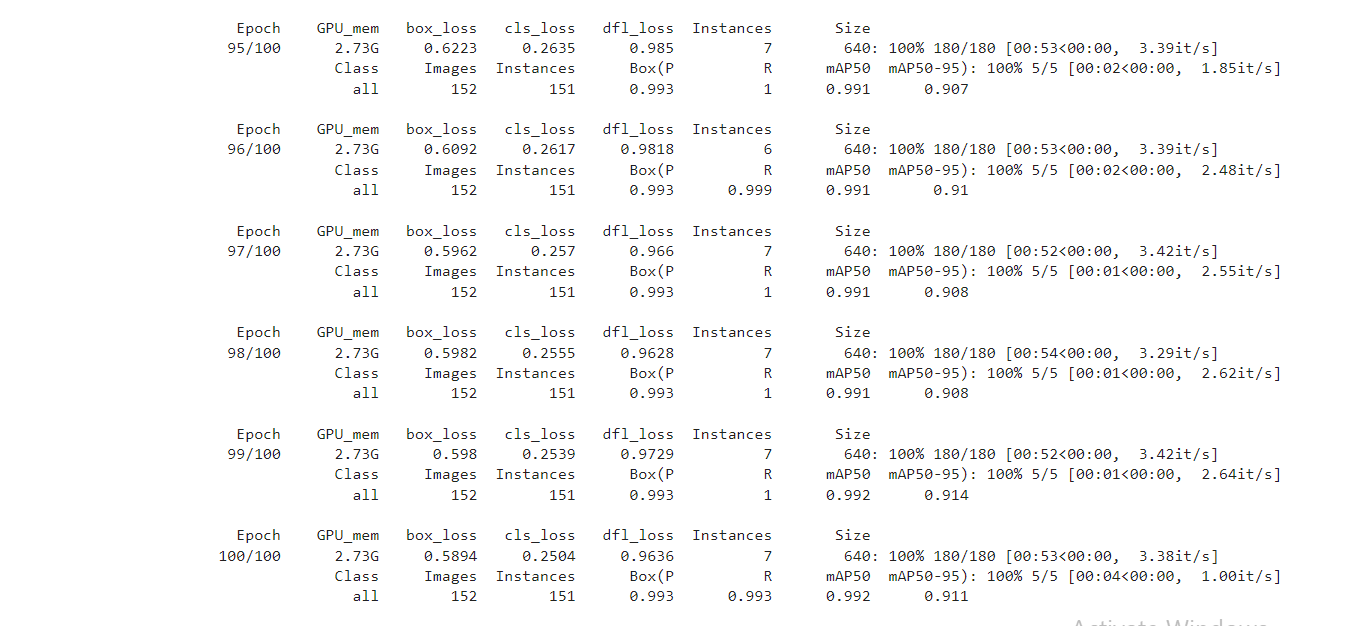
* Chức năng chính: hiển thị ảnh, thực hiện phát hiện và phân loại ảnh dựa trên mô hình đã được huấn luyện.
* Giao diện người dùng: Thiết kế giao diện người dùng thông qua Qt Designer hoặc mã Python. Giao diện cần bao gồm các phần tử như nút mở ảnh, ô hiển thị ảnh, và ô hiển thị kết quả phát hiện và phân loại.
* Tương tác người dùng: Xác định các tương tác người dùng, bao gồm việc mở ảnh từ thư mục, hiển thị ảnh trong ứng dụng, và hiển thị kết quả phát hiện và phân loại ảnh.
* Hiển thị ảnh: Sử dụng QLabel để hiển thị ảnh trong giao diện. Qt cung cấp các phương pháp để tải và hiển thị ảnh từ file hoặc từ các dữ liệu ảnh.
* Phát hiện và phân loại ảnh: Thực hiện quá trình phát hiện và phân loại ảnh bằng cách tích hợp các mô hình hoặc thuật toán tương ứng vào mã Python của bạn. Sau đó, hiển thị kết quả trên giao diện người dùng.
* Tích hợp các chức năng khác: Ngoài việc hiển thị và xử lý ảnh, tích hợp các chức năng khác như lưu trữ kết quả, thực hiện xử lý thêm trên ảnh, và tương tác với người dùng

# CHƯƠNG III: THỰC NGHIỆM, CÀI ĐẶT VÀ MÔ PHỎNG HỆ THỐNG

## 3.1 Kết quả thực nghiệm mô hình

### 3.1.1 Mô hình phát hiện biển số xe

- Kết quả sau quá trình training thu được mô hình với box loss là 0.5894, classification loss là 0.2504, Directional Feature Learning Loss là 0.9636



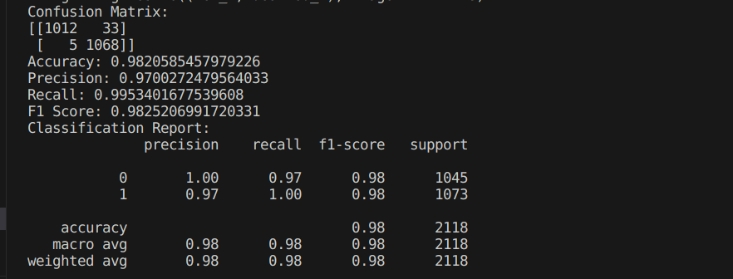
Hình 3.1: Kết quả đánh giá quá trình trainning mô hình phát hiện biển số

### 3.1.2 Mô hình phân loại biển số xe

#### 3.1.2.1 Mô hình phân loại mờ

* Đánh giá mô hình trên tập test

Đánh giá hiệu suất của mô hình trên tập dữ liệu test gồm 2.118 ảnh, kết quả thu được độ chính xác của mô hình phân loại biển số bị mờ là 98.2%, Precision đạt 97%, Recall đạt 99.5%



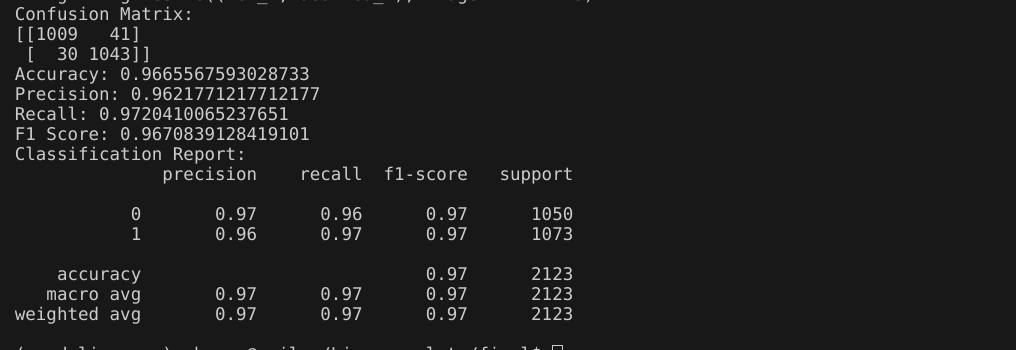
Hình 3.2: Kết quả đánh giá mô hình phân loại biển số mờ

* Hạn chế của mô hình
* Một số trường hợp bị mờ 1 góc nhỏ của biển số xe nhưng không bị mờ thông tin nhưng mô hình vẫn phân loại kết quả biển số bị mờ
* Một số trường hợp ảnh bị mờ do chất lượng của biển số thấp nhưng vẫn đọc được thông tin , mô hình phân loại kết quả biển bị mờ

#### 3.1.2.2 Mô hình phân loại che khuất

* Đánh giá mô hình trên tập test

Đánh giá hiệu suất của mô hình trên tập dữ liệu test gồm 2.122 ảnh, kết quả thu được độ chính xác của mô hình phân loại biển số bị che khuất là 96.6%, Precision đạt 96.2%, Recall đạt 97.2%



Hình 3.3: Kết quả đánh giá mô hình phân loại biển số bị che khuất

* Hạn chế của mô hình
* Một số trường hợp biển số chỉ bị khuất 1 phần nhỏ vẫn hiện rõ thông tin nhưng mô hình phân loại thành biển bị che khuất
* Một số trường hợp có vật khác che biển số mô hình như phân loại được biển bị che khuất

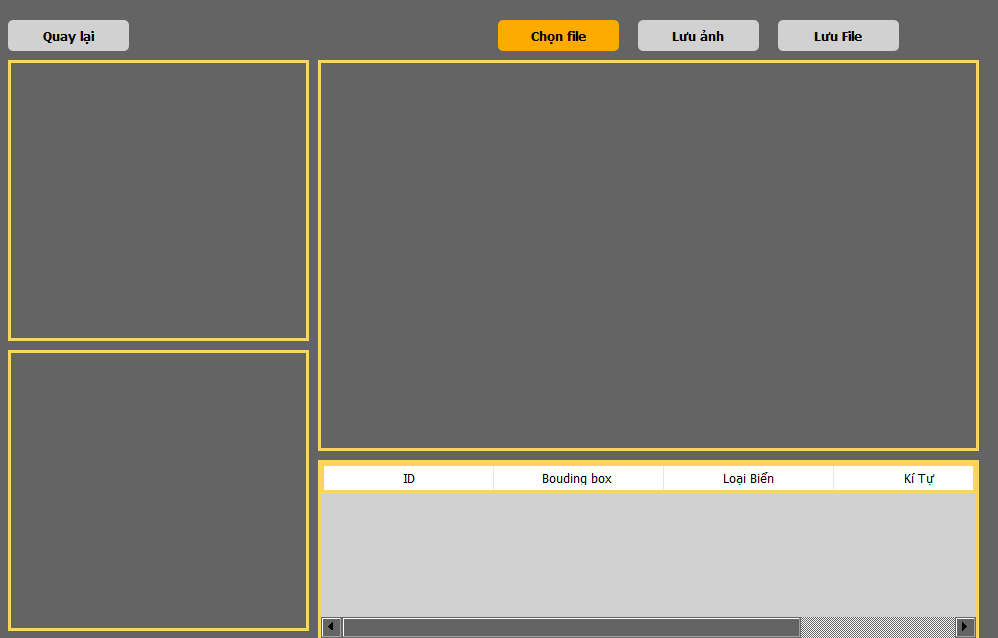
## 3.2 Giao diện hệ thống

* Giao diện giới thiệu của hệ thống:
* Có 3 tính năng chính tương ứng với định dạng 3 đầu vào khác nhau: Ảnh, Webcam và Video
* Người sử dụng click chuột vào các nút màu cam để chọn tính năng muốn sử dụng



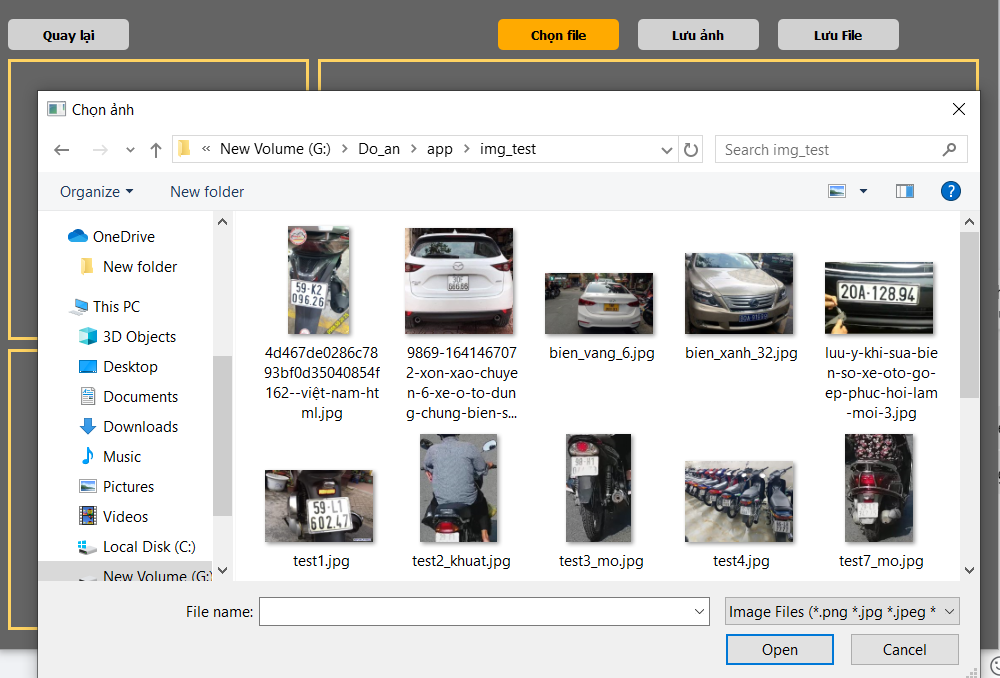
Hình 3.4: Giao diện giới thiệu của hệ thống

* Giao diện tính năng nhận diện và phân loại biển số xe đầu vào là ảnh :
* Giao diện này xuất hiện sau khi người sử dụng click chuột vào nút “Ảnh” trong giao diện giới thiệu



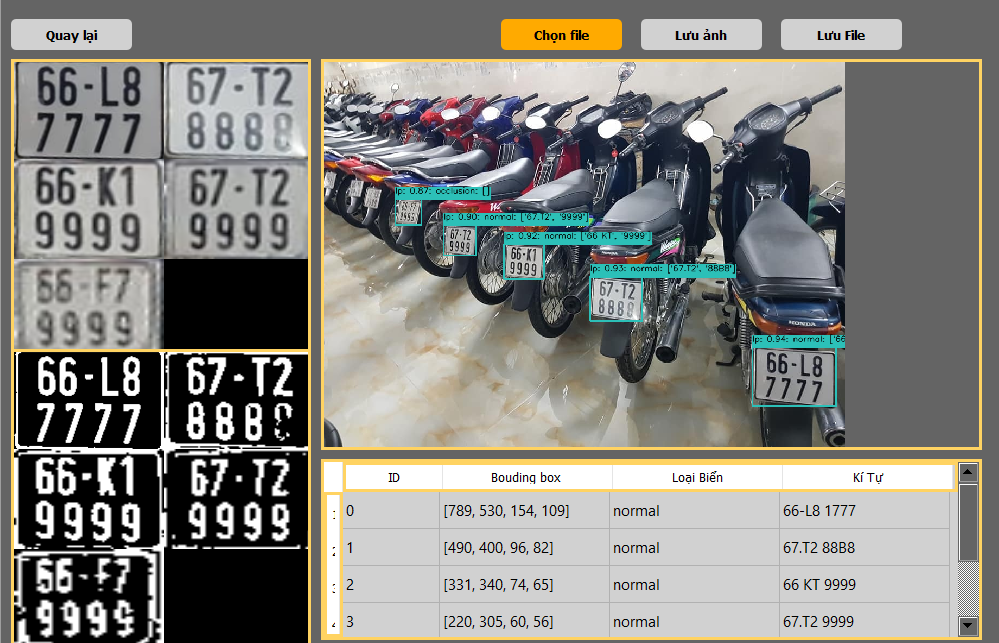
Hình 3.5: Giao diện nhận diện và phân loại biển số xe

* Giao diện chọn file từ thư viện:
* Giao diện này xuất hiện sau khi người sử dụng click chuột vào nút “Chọn file”
* Người sử dụng chọn 1 file ảnh trong thư viện để thực hiện quá trình nhận dạng và phân loại biển số



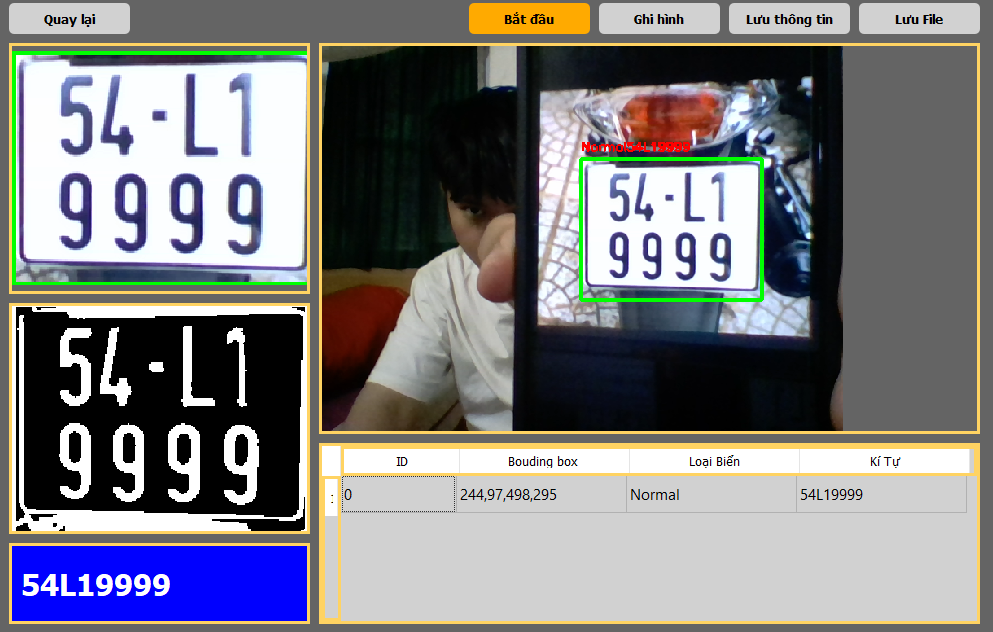
Hình 3.6: Giao diện chọn file từ thư viện

* Hiển thị kết quả nhận dạng và phân loại ảnh biển số xe:
* Sau quá trình nhận dạng và phân loại biển số hệ thống sẽ trả về hình ảnh kèm bouding box và loại biển số
* Thông tin được hiển thị trong bảng bên phải với các thông tin: ID, Tọa độ box, Loại biển, Kí tự. Người sử dụng có thể click nút “Lưu file” để xuất file excel



Hình 3.7: Giao diện hiển thị kết quả nhận diện và phân loại ảnh biển số xe

* Giao diện tính năng nhận diện và phân loại biển số xe đầu vào là webcam, :
* Giao diện này xuất hiện sau khi người sử dụng click chuột vào nút “Webcam” trong giao diện giới thiệu
* Người dung click nút “Bắt đầu” để bật webcam và thực hiện quá trình nhận dạng và phân loại biển số



Hình 3.8: Giao diện hiển thị kết quả nhận diện và phân loại biển số bằng webcam

* Giao diện tính năng nhận diện và phân loại biển số xe đầu vào là video :
* Giao diện này xuất hiện sau khi người sử dụng click chuột vào nút “Video” trong giao diện giới thiệu
* Người dung click nút “Mở file” để bật giao diện chọn file từ thư viện sau đó người dùng chọn video muốn phân tích và thực hiện quá trình nhận dạng và phân loại biển số

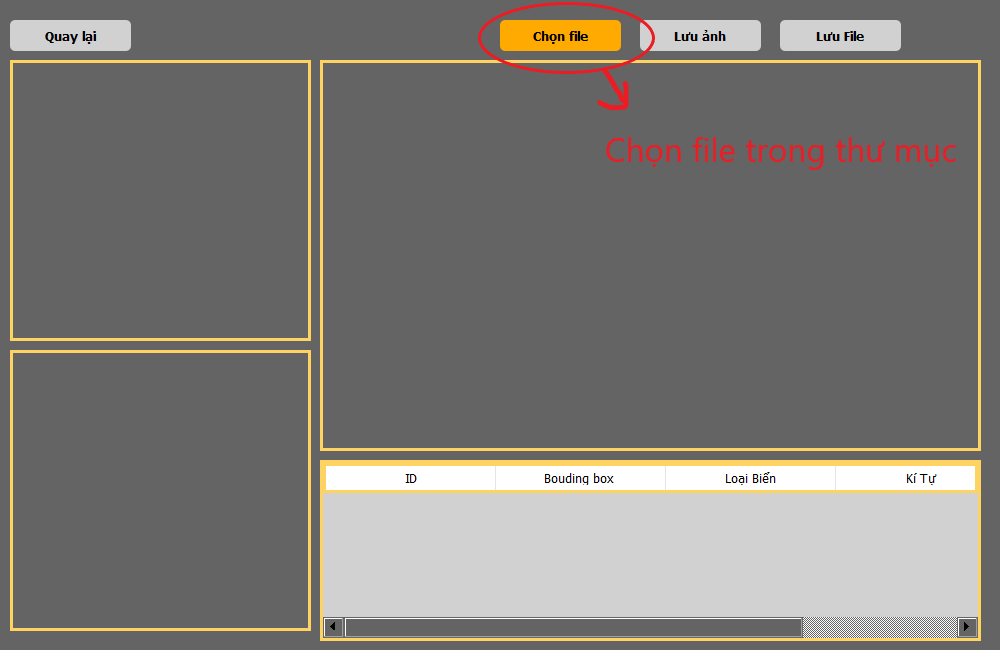


Hình 3.9: Giao diện hiển thị kết quả nhận diện và phân loại video biển số xe

## 3.3 Kết quả thực nghiệm hệ thống

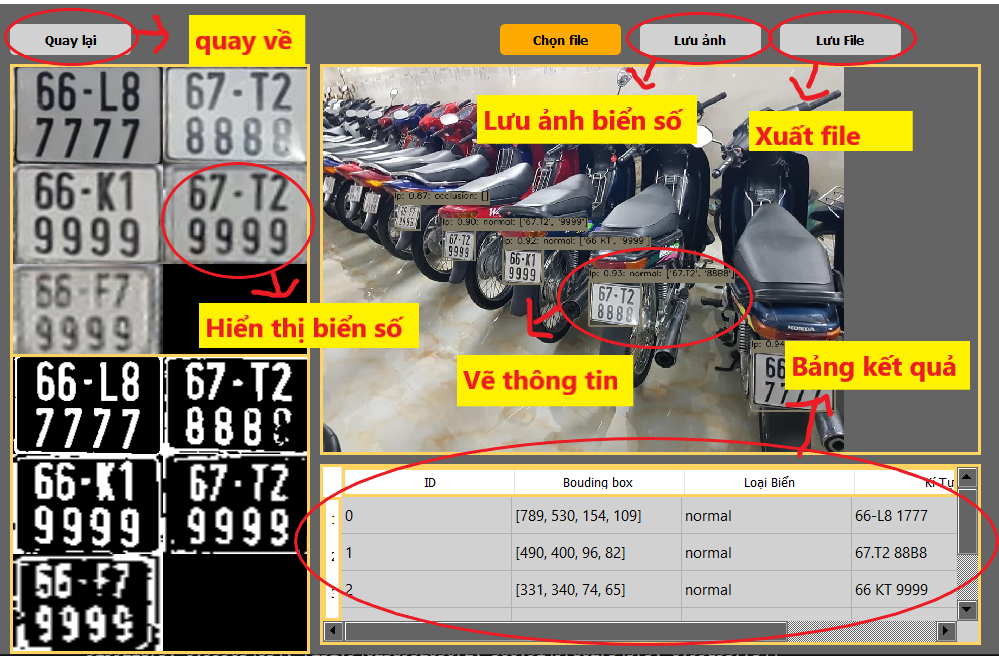
### 3.3.1 Test hệ thống bằng ảnh trong thư viện

* Click “Chọn file” để chọn ảnh muốn nhận diện trên hệ thống



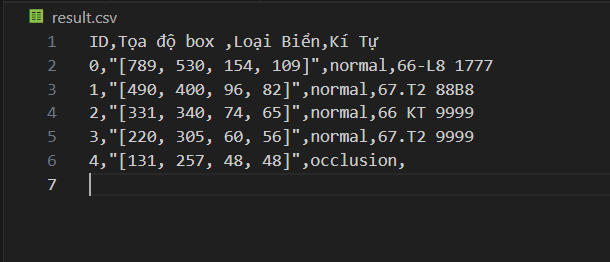
Hình 3.10: Chọn ảnh trong thư viện

* Kết quả nhận được
* Người dùng click “Lưu ảnh” để lưu ảnh tổng hợp trích xuất biển số
* Người dùng click “Lưu file” để xuất file kết quả



Hình 3.11: Kết quả nhận diện và phân loại ảnh biển số xe

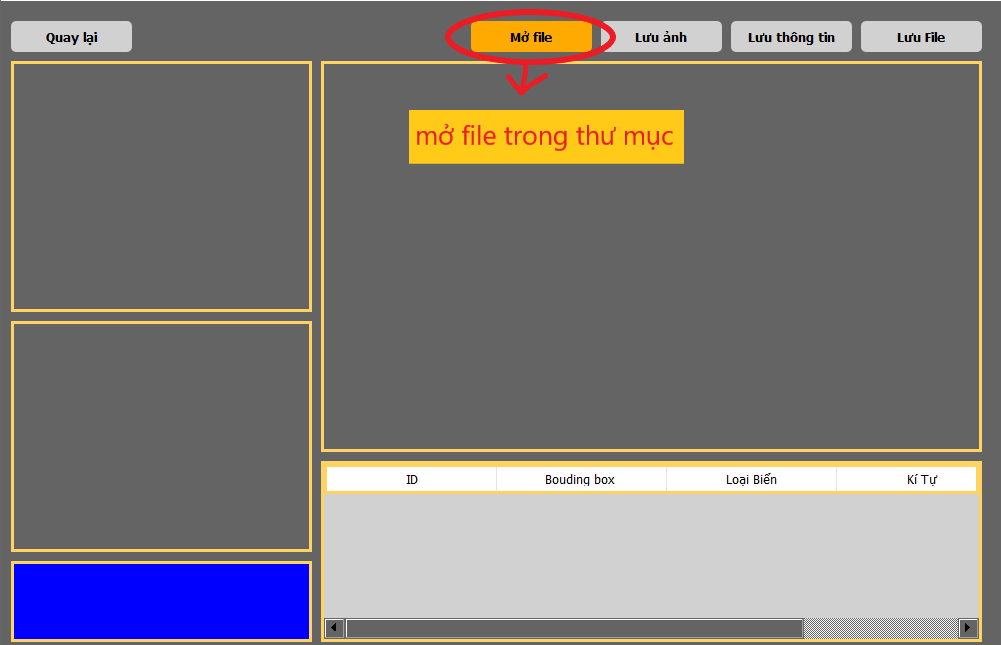
* Kết quả khi xuất ra file excel



Hình 3.12: Kết quả sau khi xuất ra file excel

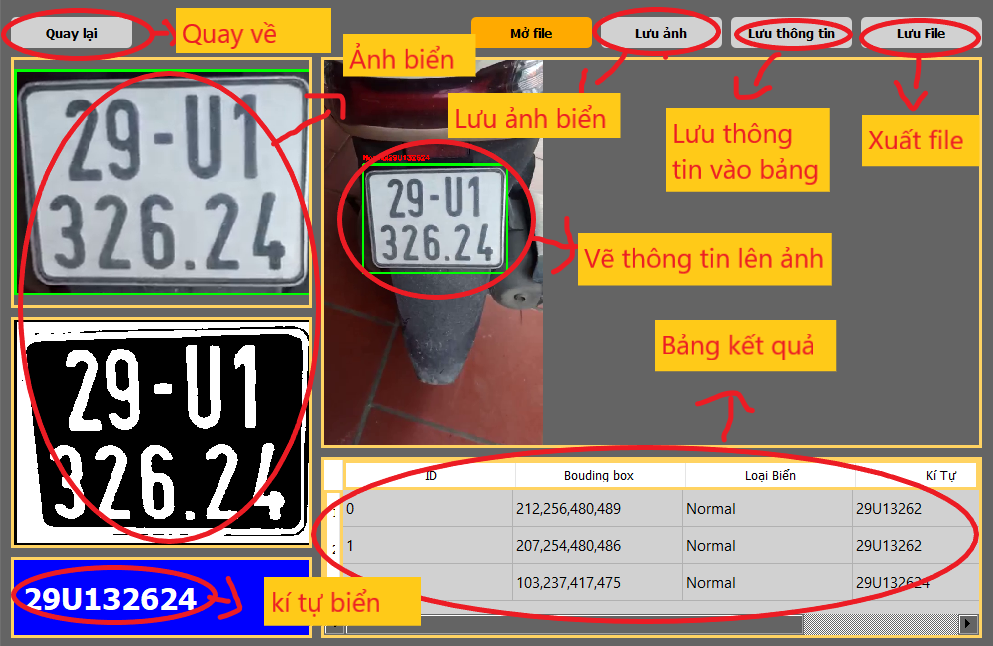
### 3.3.2 Test hệ thống bằng video trong thư viện

* Click “Mở file” để chọn video muốn nhận diện trên hệ thống



Hình 3.13: Chọn video trong thư viện

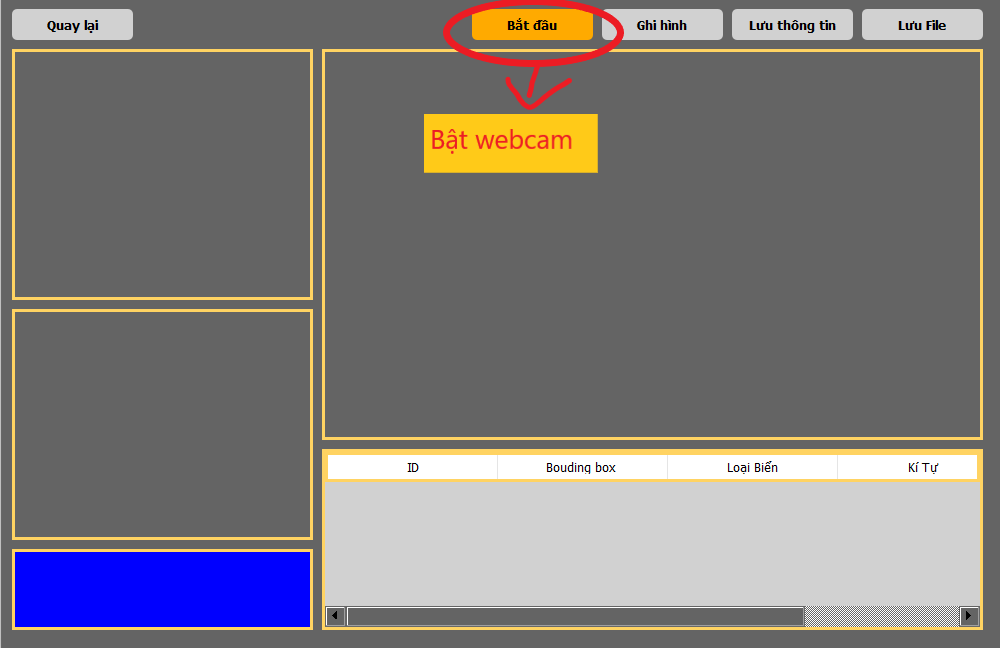
* Kết quả nhận được
* Người dùng click “Lưu ảnh” để lưu ảnh tổng hợp trích xuất biển số
* Người dùng click “Lưu thông tin” để lưu thông tin nhận dạng vào bảng
* Người dùng click “Lưu file” để xuất file kết quả



Hình 3.14: Kết quả nhận diện và phân loại video biển số xe

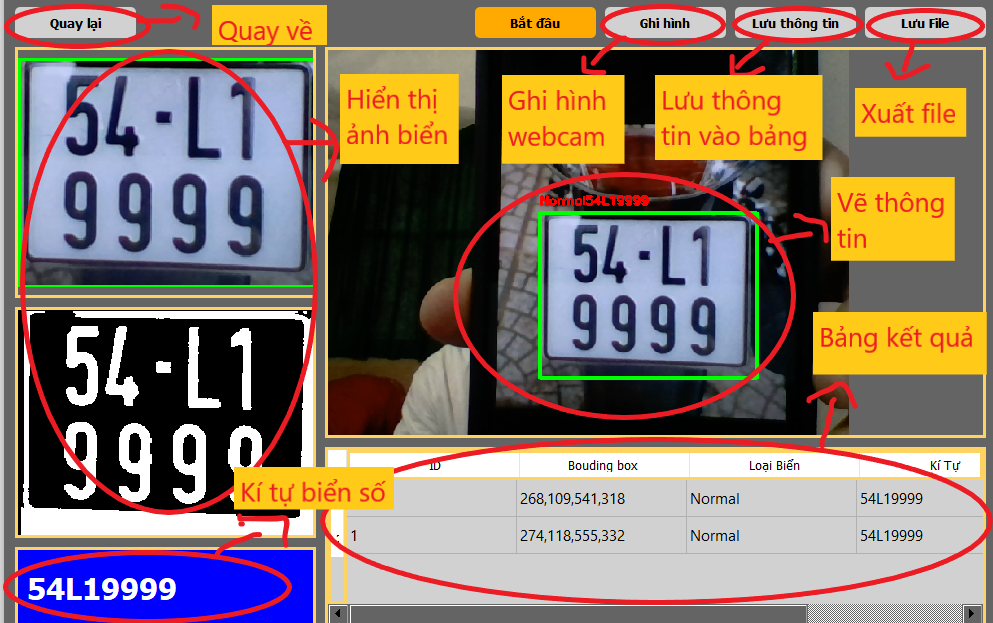
### 3.3.3 Test hệ thống bằng webcam

* Click ”Bắt đầu” để mở webcam và thực hiện quá trình nhận diện



Hình 3.15: Mở webcam để bắt đầu nhận diện

* Kết quả nhận được
* Người dùng click “Ghi hình” để lưu video webcam
* Người dùng click “Lưu thông tin” để lưu thông tin nhận dạng vào bảng
* Người dùng click “Lưu file” để xuất file kết quả



Hình 3.16: Kết quả nhận diện và phân loại biển số xe bằng webcam

# KẾT LUẬN

* **Kết quả đạt được**

Đồ án đã xử lí tốt đa số yêu cầu của đề tài như: phát hiện đa dạng biển số xe, phân loại biển số xe, nhận diện kí tự biển số xe. Đồ án xây dựng thành công luồng xử lí phát hiện ,phân loại và và nhận diện biển số xe với độ trễ so với thời gian thực rất thấp và đạt được độ chính xác cao.

* **Vấn đề còn tồn tại**

Độ chính xác của chương trình nhận diện và phân loại biển số xe phụ thuộc khá nhiều vào ánh sáng môi trường và tình trạng của biển số xe , nhiều trường hợp chất lượng biển và điều kiện ánh sáng kém nên dẫn đến nhận dạng kí tự bị sai và yếu tố này ta có thể điều chỉnh được

* **Hướng phát triển**

Để giải quyết các khó khăn của đề tài, người thực hiện xin đề xuất 1 số ý kiến

sau đây:

* Sử dụng camera chuyên dụng.
* Thiết lập môi trường ổn định xung quanh để làm tăng độ chính xác của ảnh chụp ( xác định 1 vị trí cố định để chụp ảnh).
* Tiếp tục thử nghiệm những mô hình Deeplearning mới nhằm nâng cao hiệu suất xử lí của hệ thống.

# TÀI LIỆU THAM KHẢO

[1] PGS.TS Nguyễn Quang Hoan (2006), *Giáo trình xử lý ảnh, Học viện công nghệ bưu chính viễn thông*, Xuất Bản Hà Nội.

[2] Lương Mạnh Bá, Nguyễn Thanh Thủy (2000), *Nhập môn xử lý ảnh số, Đại Học Bách Khoa Hà Nộ*i, Giáo dục.

[3] Nguyễn Thành Lợi, Đào Xuân Phúc,Nguyễn Thị Tố Uyên,Nguyễn Hữu Phát (2023) ,*Đề xuất mô hình Yolov5 ứng dụng trong nhận diện biển số xe*,Trường Điện-Điện tử,Đại học Bách Khoa Hà Nội

[4] Rayson Laroca, Evair Severo, Luiz A. Zanlorensi, Luiz S. Oliveira, Gabriel Resende Gonçalves, William Robson Schwartz, David Menotti (2018) , *A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector*

[5] Hui Li, Peng Wang, and Chunhua Shen (2019),*Towards End-to-End Car License Plates Detection and Recognition with Deep Neural Networks,* Northwestern Polytechnical University

[6] <https://sti.vista.gov.vn/tw/Pages/tai-lieu-khcn.aspx?ItemID=329734>

[7] <https://thigiacmaytinh.com/>

[8] <https://learnopencv.com/>

[9] <https://github.com/ultralytics/ultralytics>

[10] <https://phamdinhkhanh.github.io/2020/12/19/Resnet.html>

# PHỤ LỤC

**Code chương trình hệ thống phát hiện và phân loại biển số xe**

* Khởi tạo thư viện và các model cần thiết

import sys

from PyQt5.QtWidgets import QApplication, QMainWindow, QPushButton, QLabel, QVBoxLayout, QWidget, QFileDialog,QTableWidgetItem,QDialog

from PyQt5.uic import loadUi

from PyQt5.QtGui import QPixmap,QImage

from PyQt5.QtCore import QTimer,QThread,pyqtSignal

from PyQt5.QtCore import Qt

from utils import \*

import csv

import copy

import easyocr

from ultralytics import YOLO

LICENSE\_MODEL\_DETECTION\_DIR = 'model\_trained/license\_plate\_detector.pt'

license\_plate\_detector = YOLO(LICENSE\_MODEL\_DETECTION\_DIR)

blur\_path='model\_trained/blur\_runs-48-6288.pt'

occ\_path='model\_trained/occ\_runs-48-6288.pt'

att\_occ=AttributeDetector\_resnet(occ\_path)

att\_blur=AttributeDetector\_resnet(blur\_path)

# Khởi tạo mô hình Yolov8

model = Yolov8(onnx\_model='model\_trained/yolov8l\_lp.onnx', confidence\_thres=0.5, iou\_thres=0.5)

* Chương trình nhận diện và phân loại biển số xe đầu vào là ảnh

class MainWindow\_IMG(QMainWindow):

    def \_\_init\_\_(self):

        super().\_\_init\_\_()

        # Tải giao diện từ file .ui

        loadUi("GUI/main\_img.ui", self)

        self.pushButton.clicked.connect(self.reset\_data)

         # Tạo nút lưu và kết nối với sự kiện nhấn

        self.pushButton\_2.clicked.connect(self.save\_to\_csv)

        # Kết nối sự kiện nhấn nút pushButton với phương thức open\_image

        self.pushButton.clicked.connect(self.open\_image)

        path\_img\_ui1='img\_ui\z5312217369290\_f95ca37c21ebad9381e285b6319b68ee.jpg'

        img\_ui1=cv2.imread(path\_img\_ui1)

        self.display\_image(img\_ui1, self.label\_5)

    def open\_image(self):

        # Mở hộp thoại chọn tệp

        filename, \_ = QFileDialog.getOpenFileName(self, "Chọn ảnh", "", "Image Files (\*.png \*.jpg \*.jpeg \*.bmp \*.gif)")

        # Kiểm tra nếu người dùng đã chọn một tệp

        if filename:

            img = cv2.imread(filename)

            if img is not None:

                boxes\_info = model(img)

                count\_id=0

                for box\_info in boxes\_info:

                    box = box\_info['box']

                    score = box\_info['score']

                    class\_id = box\_info['class\_id']

                    key\_point= box\_info["keypoint"]

                    print(key\_point)

                    yolo\_kps = np.array(key\_point)

                    yolo\_kps=yolo\_kps[:,:2]

                    #print(yolo\_kps)

                    tmp = copy.deepcopy(yolo\_kps[3])

                    yolo\_kps[3]= yolo\_kps[2]

                    yolo\_kps[2]= tmp

                # print(yolo\_kps.shape)

                    yldm = copy.deepcopy(yolo\_kps)

                    #print(yldm)

                    ratio = 0

                    x1,y1,x2,y2 = get\_padding\_align(yldm[1],yldm[2],ratio)

                    x0,y0,x3,y3 = get\_padding\_align(yldm[0],yldm[3],ratio)

                    yldm[0]= np.array([x0,y0])

                    yldm[1]= np.array([x1,y1])

                    yldm[2]= np.array([x2,y2])

                    yldm[3]= np.array([x3,y3])

                    # print(type(yldm))

                    ypts= np.asarray([[yldm[0][0],yldm[0][1]],[yldm[1][0],yldm[1][1]],[yldm[3][0],yldm[3][1]],[yldm[2][0],yldm[2][1]]],dtype = np.float32)

                    print(ypts)

                    warped\_ = four\_point\_transform(img, ypts)

                    check\_occ = np.argmax(att\_occ(warped\_))

                    check\_blur=np.argmax(att\_blur(warped\_))

                    print(check\_occ)

                    print(check\_blur)

                    check=''

                    ocr=[]

                    if check\_occ==0 and check\_blur==0:

                        check = 'normal'

                        # process license plate

                        license\_plate\_crop\_gray = cv2.cvtColor(warped\_, cv2.COLOR\_BGR2GRAY)

                        reader = easyocr.Reader(['en'])

                        result = reader.readtext(license\_plate\_crop\_gray)

                        for detection in result:

                            ocr.append(detection[1])

                    elif check\_occ==0 and check\_blur==1:

                        check = 'blur'

                    elif check\_occ==1 and check\_blur==0:

                        check = 'occlusion'

                    else:

                        check = 'blur and occlusion'

                    model.draw\_detections(img, box, score, class\_id,check,ocr)

                    result\_string = ' '.join(ocr)

                    self.tableWidget.insertRow(count\_id)

                    self.tableWidget.setItem(count\_id, 0, QTableWidgetItem(str(count\_id)))

                    self.tableWidget.setItem(count\_id, 1, QTableWidgetItem(str(box)))  # Convert int to str

                    self.tableWidget.setItem(count\_id, 2, QTableWidgetItem(check))

                    self.tableWidget.setItem(count\_id, 3, QTableWidgetItem(result\_string))

                    count\_id+=1

            else:

                self.result\_label.setText("Không thể đọc ảnh")

        self.display\_image(img, self.label\_2)

    def display\_image(self, cvimage, label\_widget):

        # Chuyển đổi ảnh từ OpenCV sang QImage

        height, width, channel = cvimage.shape

        bytes\_per\_line = 3 \* width

        qimage = QImage(cvimage.data, width, height, bytes\_per\_line, QImage.Format\_RGB888).rgbSwapped()

        # Hiển thị ảnh lên QLabel

        pixmap = QPixmap.fromImage(qimage)

        label\_widget.setPixmap(pixmap.scaled(label\_widget.size(), Qt.KeepAspectRatio))

    def reset\_data(self):

        # Xóa văn bản của label

        self.label\_2.setText("")

        # Xóa dữ liệu của bảng

        self.tableWidget.setRowCount(0)

    def save\_to\_csv(self):

        file\_path, \_ = QFileDialog.getSaveFileName(self, 'Lưu file CSV', '.', 'CSV Files (\*.csv)')

        if file\_path:

            with open(file\_path, mode='w', newline='', encoding='utf-8') as file:  # Sử dụng encoding 'utf-8'

                writer = csv.writer(file)

                # Ghi tiêu đề của các cột

                headers = [self.tableWidget.horizontalHeaderItem(col).text() for col in range(self.tableWidget.columnCount())]

                writer.writerow(headers)

                # Ghi dữ liệu từ TableWidget vào tệp CSV

                for row in range(self.tableWidget.rowCount()):

                    row\_data = []

                    for col in range(self.tableWidget.columnCount()):

                        item = self.tableWidget.item(row, col)

                        if item is not None:

                            row\_data.append(item.text())

                        else:

                            row\_data.append("")

                    writer.writerow(row\_data)

* Chương trình nhận diện và phân loại biển số xe đầu vào là video

class VideoThread(QThread):

    change\_pixmap\_signal = pyqtSignal(QImage)

    def \_\_init\_\_(self, filename):

        super().\_\_init\_\_()

        self.filename = filename

        self.reader = easyocr.Reader(['en'])

    def run(self):

        cap = cv2.VideoCapture(self.filename)

        while cap.isOpened():

            ret, frame = cap.read()

            if ret:

                frame = cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)

                license\_detections = license\_plate\_detector(frame)[0]

                if len(license\_detections.boxes.cls.tolist()) != 0 :

                        for license\_plate in license\_detections.boxes.data.tolist() :

                            x1, y1, x2, y2, score, class\_id = license\_plate

                            cv2.rectangle(frame, (int(x1), int(y1)), (int(x2), int(y2)), (0, 255, 0), 3)

                            license\_plate\_crop = frame[int(y1):int(y2), int(x1): int(x2), :]

                            license\_plate\_crop\_gray = cv2.cvtColor(license\_plate\_crop, cv2.COLOR\_BGR2GRAY)

                            result = self.reader.readtext(license\_plate\_crop\_gray)

                            for detection in result:

                                bbox = detection[0]

                                text = detection[1]

                                cv2.putText(frame, text, (int(bbox[0][0]), int(bbox[0][1])), cv2.FONT\_HERSHEY\_SIMPLEX, 0.5, (0, 255, 0), 2)

                h, w, ch = frame.shape

                bytes\_per\_line = ch \* w

                convert\_to\_qt\_format = QImage(frame.data, w, h, bytes\_per\_line, QImage.Format\_RGB888)

                self.change\_pixmap\_signal.emit(convert\_to\_qt\_format)

            else:

                break

        cap.release()

class MainWindow\_VIDEO(QMainWindow):

    def \_\_init\_\_(self):

        super().\_\_init\_\_()

        loadUi("GUI/video.ui", self)

        self.pushButton.clicked.connect(self.select\_video)

    def select\_video(self):

        filename, \_ = QFileDialog.getOpenFileName(self, "Select Video File", "", "Video Files (\*.mp4 \*.avi)")

        if filename:

            self.thread = VideoThread(filename)

            self.thread.change\_pixmap\_signal.connect(self.update\_image)

            self.thread.start()

    def update\_image(self,image):

        self.label.setPixmap(QPixmap.fromImage(image).scaled(self.label.size(), Qt.KeepAspectRatio))

* Chương trình nhận diện và phân loại biển số xe đầu vào là ảnh

class WebcamThread(QThread):

    frame\_processed = pyqtSignal(QImage)

    def \_\_init\_\_(self):

        super().\_\_init\_\_()

        self.reader = easyocr.Reader(['en'])

        self.running = True

    def run(self):

        cap = cv2.VideoCapture(0)

        while self.running:

            ret, frame = cap.read()

            if not ret:

                break

            frame = cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)

            license\_detections = license\_plate\_detector(frame)[0]

            if len(license\_detections.boxes.cls.tolist()) != 0 :

                for license\_plate in license\_detections.boxes.data.tolist() :

                    x1, y1, x2, y2, score, class\_id = license\_plate

                    cv2.rectangle(frame, (int(x1), int(y1)), (int(x2), int(y2)), (0, 255, 0), 3)

                    license\_plate\_crop = frame[int(y1):int(y2), int(x1): int(x2), :]

                    license\_plate\_crop\_gray = cv2.cvtColor(license\_plate\_crop, cv2.COLOR\_BGR2GRAY)

                    result = self.reader.readtext(license\_plate\_crop\_gray)

                    for detection in result:

                        bbox = detection[0]

                        text = detection[1]

                        cv2.putText(frame, text, (int(bbox[0][0]), int(bbox[0][1])), cv2.FONT\_HERSHEY\_SIMPLEX, 0.5, (0, 255, 0), 2)

            qimage = QImage(frame.data, frame.shape[1], frame.shape[0], QImage.Format\_RGB888)

            self.frame\_processed.emit(qimage)

class MainWindow\_WEBCAM(QMainWindow):

    def \_\_init\_\_(self):

        super().\_\_init\_\_()

        loadUi("GUI/webcam.ui", self)

        self.pushButton.clicked.connect(self.start\_webcam)

        self.webcam\_thread = WebcamThread()

        self.webcam\_thread.frame\_processed.connect(self.display\_webcam)

    def start\_webcam(self):

        self.webcam\_thread.start()

    def display\_webcam(self, qimage):

        pixmap = QPixmap.fromImage(qimage)

        self.label.setPixmap(pixmap.scaled(self.label.size(), Qt.KeepAspectRatio)