**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN**

**HỌC PHẦN :XỬ LÝ ẢNH VÀ THỊ GIÁC MÁY TÍNH**

**Đề 09 : Xây dựng hệ thống phân loại cảm xúc qua khuôn mặt của con người**

**Giảng viên hướng dẫn: Lương Thị Hồng Lan**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Mã sinh viên** | **Sinh viên thực hiện** | **Lớp hành chính** |
| **1** | 20212993 | Nguyễn Văn Dũng | DCCNTT12.10.4 |
| **2** | 20211196 | Vũ Xuân Lộc | DCCNTT12.10.4 |
| **3** | 20211062 | Đới Sỹ Mạnh | DCCNTT12.10.4 |
| **4** | 20211125 | Phạm Duy Mạnh | DCCNTT12.10.4 |
| **5** | 20210973 | Lê Văn Tuấn | DCCNTT12.10.4 |

**Bắc Ninh, năm 2024**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÀI TẬP LỚN**

**HỌC PHẦN :XỬ LÝ ẢNH VÀ THỊ GIÁC MÁY TÍNH**

**Đề 09 : Xây dựng hệ thống phân loại cảm xúc qua khuôn mặt của con người**

**Giảng viên hướng dẫn: Lương Thị Hồng Lan**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Mã sinh viên** | **Sinh viên thực hiện** | **Lớp hành chính** |
| **1** | 20212993 | Nguyễn Văn Dũng | DCCNTT12.10.4 |
| **2** | 20211196 | Vũ Xuân Lộc | DCCNTT12.10.4 |
| **3** | 20211062 | Đới Sỹ Mạnh | DCCNTT12.10.4 |
| **4** | 20211125 | Phạm Duy Mạnh | DCCNTT12.10.4 |
| **5** | 20210973 | Lê Văn Tuấn | DCCNTT12.10.4 |

**Bắc Ninh, năm 2024**

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á  **KHOA CÔNG NGHỆ THÔNG TIN** | **KỲ THI KẾT THÚC HỌC PHẦN**  **HỌC KỲ 1, NĂM HỌC 2024** – **2025** |

|  |  |
| --- | --- |
| **PHIẾU CHẤM THI BÀI TẬP LỚN KẾT THÚC HỌC PHẦN**  **HỌC PHẦN :XỬ LÝ ẢNH VÀ THỊ GIÁC MÁY TÍNH**  **Đề 09 : Xây dựng hệ thống phân loại cảm xúc qua khuôn mặt của con người**  **Giảng viên hướng dẫn: Lương Thị Hồng Lan**  **Lớp Tín chỉ: XATGMT.03.K12.04.LH.C04.1\_LT** | |
| **Cán bộ chấm thi 1**  *(Ký và ghi rõ họ tên)*  **Lương Thị Hồng Lan** | **Cán bộ chấm thi 2**  *(Ký và ghi rõ họ tên)* |

| **TT** | **TIÊU CHÍ** | **THANG ĐIỂM** | **Nguyễn Văn Dũng** | **Vũ Xuân Lộc** | **Đới Sỹ Mạnh** | **Phạm Duy Mạnh** | **Lê Văn Tuấn** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 20212993 | 20211196 | 20211062 | 20211125 | 20210973 |
| **1** | **Nội dung báo cáo trên Word đầy đủ** | **3.5** |  |  |  |  |  |
| 1.1 | Có bố cục rõ ràng (mục lục, phần mở đầu, nội dung chính, kết luận). | 0,5 |  |  |  |  |  |
| 1.2 | Nội dung phân tích rõ ràng, logic. | 0,5 |  |  |  |  |  |
| 1.3 | Có dẫn chứng, số liệu minh họa đầy đủ. | 0,5 |  |  |  |  |  |
| 1.4 | Ngôn ngữ và trình bày chuẩn, không lỗi chính tả. | 0,5 |  |  |  |  |  |
| 1.5 | Có trích dẫn tài liệu tham khảo đúng quy cách. | 0,5 |  |  |  |  |  |
| 1.6 | Được trình bày chuyên nghiệp (canh lề, font chữ, khoảng cách dòng hợp lý). | 0,5 |  |  |  |  |  |
| 1.7 | Tài liệu đầy đủ, bám sát yêu cầu của đề bài. | 0,5 |  |  |  |  |  |
| **2** | **Nội dung thuyết trình đầy đủ** | **1.0** |  |  |  |  |  |
| 2.1 | Trình bày tự tin, phát âm rõ ràng, mạch lạc. | 0,5 |  |  |  |  |  |
| 2.2 | Nội dung thuyết trình đúng trọng tâm, không lan man. | 0,5 |  |  |  |  |  |
| **3** | **Slides báo cáo đầy đủ nội dung + Hỏi đáp** | **3.0** |  |  |  |  |  |
| 3.1 | Slides có bố cục rõ ràng (mở đầu, nội dung, kết luận). | 0,5 |  |  |  |  |  |
| 3.2 | Thiết kế slides đẹp, chuyên nghiệp (màu sắc, hình ảnh minh họa). | 0,5 |  |  |  |  |  |
| 3.3 | Nội dung trên slides ngắn gọn, dễ hiểu, súc tích. | 0,5 |  |  |  |  |  |
| 3.4 | Nội dung slides phù hợp với nội dung báo cáo. | 0,5 |  |  |  |  |  |
| 3.5 | Trả lời câu hỏi đầy đủ, chính xác. | 0,5 |  |  |  |  |  |
| 3.6 | Trả lời câu hỏi tự tin, thuyết phục. | 0,5 |  |  |  |  |  |
| **4** | **Code đầy đủ** | **2.5** |  |  |  |  |  |
| 1.1 | Code được trình bày rõ ràng, có chú thích đầy đủ. | 0,5 |  |  |  |  |  |
| 1.2 | Code chạy đúng, không lỗi. | 0,5 |  |  |  |  |  |
| 1.3 | Code tối ưu, không dư thừa. | 0,5 |  |  |  |  |  |
| 1.4 | Đáp ứng đầy đủ các yêu cầu chức năng theo đề bài. | 0,5 |  |  |  |  |  |
| 1.5 | Có tính sáng tạo hoặc cải thiện so với yêu cầu. | 0,5 |  |  |  |  |  |
| **TỔNG ĐIỂM BẰNG SỐ:** | | **10** |  |  |  |  |  |
| **TỔNG ĐIỂM BẰNG CHỮ:** | | *Mười tròn* |  |  |  |  |  |

**Mục lục**

[Lời nói đầu 9](#_Toc184559520)

[Chương 1 Cơ sở lý thuyết 10](#_Toc184559521)

[1.1 Nhận dạng 10](#_Toc184559522)

[1.1.1 Nhận dạng là gì? 10](#_Toc184559523)

[1.2.1 Ứng dụng 11](#_Toc184559524)

[1.2 Các phương pháp sử dụng trong nhận dạng 14](#_Toc184559525)

[1.2.1 Sử dụng đặc trưng biên 14](#_Toc184559526)

[1.2.2 Học sâu 16](#_Toc184559527)

[1.3 Ngôn ngữ lập trình và các thư viện sử dụng 19](#_Toc184559528)

[1.3.1 Ngôn ngữ lập trình 19](#_Toc184559529)

[1.3.2 Các thư viện 25](#_Toc184559530)

[Chương 2 Xây dựng mô hình 29](#_Toc184559531)

[2.1 Mô Tả Bài Toán 29](#_Toc184559532)

[2.2 Xây Dựng Hệ Thống 31](#_Toc184559533)

[Chương 3 Kết quả thực nghiệm 35](#_Toc184559534)

[3.1 Dữ liệu 35](#_Toc184559535)

[3.2 Độ đo 36](#_Toc184559536)

[3.2 Kết quả thực nghiệm 36](#_Toc184559537)

[Kết luận 39](#_Toc184559538)

[Tài liệu tham khảo 40](#_Toc184559539)

# Danh mục các từ viết tắt

|  |  |  |
| --- | --- | --- |
| Từ viết tắt | Nội dung | Diễn giải |
| **AI** | Artificial Intelligence | Trí tuệ nhân tạo |
| **CNN** | Convolutional Neural Network | Mạng thần kinh tích chập |
| **DL** | Deep Learning | Học sâu |
| **ML** | Machine Learning | Học máy |
| **ReLU** | Rectified Linear Unit | Hàm chỉnh lưu |
| **RNN** | Recurrent Neural Network | Mạng hồi quy |
| **LSTM** | Long Short Term Memory | Mạng bộ nhớ dài ngắn |
| **DBM** | Deep Boltzmann Machine | Máy Boltzmann hạn chế |
| **DBN** | Deep Belief Network | Mạng niềm tin sâu |
| **ANN** | Artificial Neural Network | Mạng thần kinh nhân tạo |
| **MLP** | Multi-layer Perceptron | Perceptron đa lớp |
| **DNN** | Deep neural network | Mạng thần kinh sâu |
| **NLP** | Natural Language Processing | Xử lý ngôn ngữ tự nhiên |

**Danh mục hình ảnh bảng biểu**

[Hình 1. Nhận diện khuôn mặt 10](#_Toc184651891)

[Hình 2. Nhận diện biển số xe 12](#_Toc184651892)

[Hình 3. Ứng dụng trong y tế 13](#_Toc184651893)

[Hình 4. Python 19](#_Toc184651894)

[Hình 5. Html 20](#_Toc184651895)

[Hình 6.Css 22](#_Toc184651896)

[Hình 7. Mô hình CNN 33](#_Toc184651897)

[Hình 8. Kết quả thực nghiệm với ảnh 37](#_Toc184651898)

[Hình 9. Thực nghiệm webcam với 1 người 37](#_Toc184651899)

[Hình 10.Thực nghiệm webcam với nhiều người 38](#_Toc184651900)

**Phân công nhiệm vụ**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thành viên** | **Vai trò** | **Nhiệm vụ** |
| 1 | Nguyễn Văn Dũng | Nhóm trưởng | Code+Tổng hợp Word+ Chỉnh sửa +Trình bày word |
| 2 | Vũ Xuân Lộc | Thành viên | Word chương 1 |
| 3 | Đới Sỹ Mạnh | Thành viên | Word chương 2 |
| 4 | Phạm Duy Mạnh | Thành viên | Word chương 3 |
| 5 | Lê Văn Tuấn | Thành viên | Powerpoint |

# Lời nói đầu

Trong bối cảnh sự phát triển vượt bậc của công nghệ thông tin và trí tuệ nhân tạo, xử lý ảnh và thị giác máy tính đã trở thành những lĩnh vực nghiên cứu trọng điểm với nhiều ứng dụng thực tiễn. Một trong những lĩnh vực nghiên cứu hấp dẫn và đầy tiềm năng là phân loại cảm xúc qua khuôn mặt. Đây là bài toán không chỉ mang lại giá trị trong khoa học mà còn đóng vai trò quan trọng trong các ứng dụng thực tế, như chăm sóc sức khỏe tâm lý, cải thiện trải nghiệm người dùng, hay hỗ trợ hệ thống giám sát thông minh.

Môn học "Xử lý Ảnh và Thị giác Máy tính" cung cấp những kiến thức nền tảng và công cụ mạnh mẽ để giải quyết các vấn đề liên quan đến phân tích và xử lý dữ liệu hình ảnh. Qua đó, học viên không chỉ nắm vững lý thuyết mà còn phát triển kỹ năng áp dụng vào các dự án thực tế.

Đề tài "Xây dựng hệ thống phân loại cảm xúc qua khuôn mặt của con người" được chọn để thực hiện trong học phần này với mục tiêu:

* Nghiên cứu các phương pháp xử lý ảnh hiện đại và các thuật toán học máy/học sâu.
* Ứng dụng các kỹ thuật trích xuất đặc trưng từ khuôn mặt để nhận diện cảm xúc.
* Xây dựng và đánh giá hệ thống phân loại cảm xúc có tính chính xác và hiệu quả.

Hệ thống này có tiềm năng ứng dụng rộng rãi trong nhiều lĩnh vực như giáo dục, y tế, giao tiếp thông minh, và các sản phẩm công nghệ cao. Quá trình thực hiện không chỉ giúp phát triển tư duy logic và kỹ năng giải quyết vấn đề mà còn khuyến khích sự sáng tạo trong việc áp dụng kiến thức lý thuyết vào thực tiễn.

Trong quá trình nghiên cứu và xây dựng hệ thống, nhóm thực hiện đã cố gắng tìm hiểu các tài liệu khoa học, thử nghiệm các phương pháp xử lý và phân loại khác nhau để đưa ra giải pháp tối ưu. Mặc dù đã nỗ lực hết mình, nhưng chắc chắn sẽ không tránh khỏi những hạn chế và sai sót. Chúng em rất mong nhận được sự góp ý và hỗ trợ từ giảng viên và các bạn đồng hành để hoàn thiện hơn.

Cuối cùng, chúng em xin gửi lời cảm ơn chân thành đến giảng viên hướng dẫn và bộ môn **"**Xử lý Ảnh và Thị giác Máy tính**"** đã hỗ trợ và tạo điều kiện để nhóm có thể hoàn thành đề tài này.

# Chương 1 Cơ sở lý thuyết

## 1.1 Nhận dạng

### 1.1.1 Nhận dạng là gì?

**Nhận dạng** là một quá trình hoặc hệ thống nhằm xác định và phân loại một đối tượng, sự kiện, hoặc hiện tượng dựa trên các đặc điểm, dữ liệu thu thập được từ môi trường. Khái niệm này được ứng dụng rộng rãi trong nhiều lĩnh vực khoa học và công nghệ, như trí tuệ nhân tạo (AI), học máy (machine learning), xử lý ảnh, xử lý ngôn ngữ tự nhiên (NLP), và các hệ thống an ninh. [1]

Nhận dạng có thể hiểu như việc hệ thống mô phỏng khả năng nhận thức của con người, giúp máy tính hoặc các thiết bị thông minh "hiểu" và "phân tích" các yếu tố như hình ảnh, âm thanh, văn bản, hoặc chuyển động.

A person's face identification

Description automatically generated

Hình 1. Nhận diện khuôn mặt

* Các loại nhận dạng chi tiết
* **Nhận dạng hình ảnh**: Phân tích và xác định nội dung trong ảnh hoặc video, ví dụ nhận dạng vật thể, khuôn mặt, hoặc cảnh quan.
* **Nhận dạng giọng nói:** Xác định nội dung hoặc nhận diện người nói từ tín hiệu âm thanh.
* **Nhận dạng văn bản (Text Recognition):** Tự động phát hiện và trích xuất thông tin từ văn bản, kể cả chữ viết tay hoặc tài liệu scan.
* **Nhận dạng hành động và cử chỉ**: Phân tích và nhận diện các chuyển động, hành động hoặc cử chỉ của con người từ video hoặc cảm biến.
* **Nhận dạng sinh trắc học**: Sử dụng các đặc điểm sinh học duy nhất của con người để nhận diện cá nhân, ví dụ vân tay, võng mạc, hoặc đặc điểm khuôn mặt.

### 1.2.1 Ứng dụng

* **Nhận dạng khuôn mặt (Face Recognition)**
* **Ngân hàng:** Xác thực khách hàng khi thực hiện giao dịch trực tuyến hoặc tại ATM.
* **Lực lượng pháp luật:** Tìm kiếm đối tượng truy nã trong đám đông, xác minh danh tính tại hiện trường.
* **Sân bay:** Kiểm soát an ninh, xác minh hành khách một cách nhanh chóng và chính xác.
* **Quản lý truy cập:** Kiểm soát ra vào các khu vực hạn chế, theo dõi sự hiện diện của nhân viên.
* **Marketing:** Phân tích cảm xúc khách hàng dựa trên biểu cảm khuôn mặt để cải thiện trải nghiệm mua sắm.
* **Nhận diện biển số xe (ANPR)**
* **Thu phí tự động:** Tính phí dựa trên biển số xe, loại bỏ các trạm thu phí truyền thống.
* **Giám sát tốc độ:** Phát hiện các phương tiện vượt quá tốc độ cho phép.
* **Tìm kiếm xe mất cắp:** Xây dựng cơ sở dữ liệu biển số xe để hỗ trợ công tác điều tra.
* **Quản lý bãi đỗ xe:** Tự động hóa quá trình vào/ra bãi đỗ xe, tính phí chính xác. [1]

A car in a parking lot

Description automatically generated

Hình 2. Nhận diện biển số xe

* **Phân loại đối tượng trong ảnh (Object Detection)**
* **An ninh:** Phát hiện các đối tượng lạ, hành vi đáng ngờ trong các khu vực công cộng.
* **Tự lái xe:** Nhận biết người đi bộ, xe đạp, các phương tiện khác, biển báo giao thông để đưa ra quyết định lái xe an toàn.
* **Sản xuất:** Kiểm soát chất lượng sản phẩm, phát hiện lỗi sản xuất.
* **OCR (Optical Character Recognition)**
* **Lưu trữ dữ liệu:** Số hóa tài liệu, tạo cơ sở dữ liệu văn bản dễ tìm kiếm.
* **Dịch vụ khách hàng:** Tự động hóa quá trình xử lý các đơn hàng, yêu cầu của khách hàng.
* **Giáo dục:** Tạo ra các công cụ học tập tương tác, hỗ trợ người khiếm thị.
* **Ứng dụng trong y tế (Medical Imaging)**
* **Chẩn đoán hình ảnh:** Hỗ trợ bác sĩ trong việc phát hiện các bệnh lý sớm, tăng độ chính xác của chẩn đoán.
* **Phẫu thuật:** Hỗ trợ phẫu thuật bằng cách cung cấp hình ảnh 3D chi tiết về cơ thể bệnh nhân.
* **Nghiên cứu y học:** Phân tích một lượng lớn dữ liệu hình ảnh để tìm ra các mối liên hệ mới, phát triển các phương pháp điều trị hiệu quả.

A doctor looking at a patient in a ct scan

Description automatically generated

Hình 3. Ứng dụng trong y tế

* **Phân tích và nhận dạng cử chỉ (Gesture Recognition)**
* **Giao diện người máy:** Điều khiển robot, máy tính bằng cử chỉ.
* **Trò chơi điện tử:** Tạo ra các trò chơi tương tác cao, trải nghiệm người dùng chân thực hơn.
* **Hỗ trợ người khuyết tật:** Điều khiển các thiết bị thông minh bằng cử chỉ.
* **Nhận diện sản phẩm trong thương mại điện tử**
* **Tìm kiếm hình ảnh:** Cho phép người dùng tìm kiếm sản phẩm bằng hình ảnh thay vì nhập từ khóa.
* **So sánh giá:** So sánh giá của cùng một sản phẩm trên các trang web khác nhau.
* **Tư vấn mua sắm:** Đề xuất các sản phẩm tương tự hoặc bổ sung dựa trên sản phẩm mà người dùng đang xem.
* **Phát hiện và phân loại hình ảnh vệ tinh**
* **Giám sát môi trường:** Theo dõi sự thay đổi của rừng, biển, đô thị.
* **Quản lý thiên tai:** Phát hiện các khu vực bị ảnh hưởng bởi thiên tai, hỗ trợ công tác cứu hộ.
* **Quản lý đô thị:** Lập bản đồ đô thị, theo dõi sự phát triển của các khu vực.
* **Tiềm năng phát triển**
* **AI và Deep Learning:** Sự phát triển của trí tuệ nhân tạo và học sâu giúp các hệ thống nhận dạng hình ảnh trở nên chính xác và thông minh hơn.
* **Ứng dụng trong thực tế ảo (VR) và tăng cường thực tế (AR):** Tạo ra các trải nghiệm tương tác sống động hơn. [3]
* **An ninh quốc gia:** Phân tích hình ảnh vệ tinh để phát hiện các hoạt động bất hợp pháp, bảo vệ biên giới.

## 1.2 Các phương pháp sử dụng trong nhận dạng

### 1.2.1 Sử dụng đặc trưng biên

Sử dụng đặc trưng đường biên (Edge Features) là một trong những phương pháp quan trọng trong xử lý ảnh và nhận diện đối tượng. Đặc trưng đường biên tập trung vào việc trích xuất thông tin liên quan đến ranh giới giữa các vùng trong ảnh, giúp xác định hình dạng và cấu trúc của đối tượng.

Biên ảnh là những điểm mà tại đó hàm độ sáng của ảnh liên tục có bước nhảy hoặc biến thiên nhanh. Cơ sở toán học của việc phát hiện và tách biên là phép toán đạo hàm, phương pháp này còn được gọi là phương pháp phát hiện biên trực tiếp. Tập hợp các điểm biên tạo thành đường biên(edge) hay đường bao (boundary) của ảnh. Ví dụ trong một ảnh nhị phân một điểm có thể được gọi là biên nếu đó là điểm đen và có ít nhất một điểm trắng lân cận.

Đặc trưng đường biên chủ yếu được sử dụng trong các bài toán nhận diện đối tượng, phân đoạn ảnh và phân tích hình ảnh. Cụ thể:

* Nhận diện đối tượng: Đặc trưng đường biên giúp phát hiện các đối tượng bằng cách nhận diện ranh giới giữa chúng và nền.
* Phân đoạn ảnh: Dùng để tách các đối tượng ra khỏi nền dựa trên sự thay đổi cường độ sáng.
* Phân tích hình dạng: Phát hiện các hình dạng trong ảnh như hình tròn, hình vuông, giúp phân loại và nhận diện hình ảnh.
* Quản lý ảnh y tế: Dùng trong các ứng dụng như nhận diện tổn thương hoặc các khối u trên ảnh X-quang hoặc CT.

**Ưu điểm:**

* Nhận diện chính xác các ranh giới:
* Đặc trưng đường biên rất hiệu quả trong việc xác định ranh giới giữa các đối tượng và nền trong ảnh, điều này cực kỳ quan trọng trong các bài toán phân đoạn ảnh và nhận diện đối tượng.
* Tăng cường tính chính xác trong nhận diện hình dạng:
* Việc phát hiện đường biên giúp mô hình hiểu rõ cấu trúc và hình dạng của đối tượng, nâng cao khả năng nhận diện hình ảnh.
* Giảm dung lượng tính toán:
* Thay vì xử lý toàn bộ ảnh, việc sử dụng đặc trưng đường biên giúp giảm thiểu số lượng dữ liệu cần xử lý, từ đó giúp tăng tốc độ xử lý và tiết kiệm tài nguyên.
* Có thể kết hợp với các phương pháp khác:
* Đặc trưng đường biên có thể dễ dàng kết hợp với các đặc trưng khác như màu sắc và kết cấu để cải thiện hiệu quả nhận diện đối tượng trong ảnh.
* Hiệu quả trong phân tích các ảnh có độ tương phản cao:
* Đặc trưng đường biên hoạt động tốt khi ảnh có sự thay đổi mạnh về cường độ sáng giữa các đối tượng và nền.

**Nhược điểm:**

* Nhạy cảm với nhiễu:
* Đặc trưng đường biên có thể bị nhiễu bởi các yếu tố không mong muốn như hạt nhiễu trong ảnh, gây ra những đường biên giả hoặc sai lệch. Điều này cần phải xử lý qua các bước tiền xử lý, ví dụ như làm mịn ảnh.
* Không hiệu quả trong ảnh có độ tương phản thấp:
* Khi sự thay đổi giữa các đối tượng và nền không rõ ràng (ảnh có độ tương phản thấp), đặc trưng đường biên có thể không phát hiện được các đường biên rõ ràng.
* Khó phát hiện đối tượng nhỏ hoặc phức tạp:
* Đặc trưng đường biên khó nhận diện các đối tượng nhỏ hoặc các đối tượng có nhiều chi tiết phức tạp, như tóc hoặc các họa tiết nhỏ trong ảnh.
* Không chứa thông tin nội dung đối tượng:
* Đặc trưng đường biên chỉ cung cấp thông tin về ranh giới của đối tượng mà không mang lại thông tin về nội dung bên trong đối tượng, như màu sắc hoặc kết cấu.
* Phụ thuộc vào ngưỡng và tham số:
* Các thuật toán phát hiện đường biên như Canny hay Sobel phụ thuộc vào việc chọn đúng ngưỡng (threshold). Nếu ngưỡng không phù hợp, có thể dẫn đến việc phát hiện quá ít hoặc quá nhiều đường biên.

### 1.2.2 Học sâu

Học sâu (Deep Learning) là một nhánh của trí tuệ nhân tạo (AI) và học máy (Machine Learning), tập trung vào việc xây dựng và huấn luyện các mô hình mạng nơ-ron nhân tạo (Artificial Neural Networks) với nhiều lớp (layers) để mô phỏng quá trình học tập của bộ não con người.

* Định Nghĩa Học Sâu:

Học sâu là một kỹ thuật học máy sử dụng các mạng nơ-ron sâu (Deep Neural Networks) để tự động học các đặc trưng từ dữ liệu. Thay vì cần phải thiết kế thủ công các đặc trưng cho mô hình, học sâu cho phép mô hình tự học từ dữ liệu đầu vào bằng cách tối ưu hóa các tham số thông qua các lớp nơ-ron. [2]

* Các Thành Phần Cơ Bản
* Mạng Nơ-ron Nhân Tạo (Artificial Neural Networks - ANN)
* Nơ-ron: Đơn vị tính toán cơ bản, lấy dữ liệu đầu vào, áp dụng trọng số (weights), cộng với độ chệch (bias), và thông qua một hàm kích hoạt (activation function) để tạo ra đầu ra.
* Lớp (Layer):
* Lớp đầu vào (Input Layer): Nhận dữ liệu từ bên ngoài.
* Lớp ẩn (Hidden Layers): Xử lý và học các đặc trưng từ dữ liệu.
* Lớp đầu ra (Output Layer): Trả về kết quả dự đoán.
* Hàm kích hoạt: Quyết định đầu ra của nơ-ron dựa trên tín hiệu đầu vào. Các hàm phổ biến:
* Sigmoid
* ReLU (Rectified Linear Unit)
* Tanh
* Softmax
* Mạng Nơ-ron Sâu (Deep Neural Networks - DNN)

Mạng nơ-ron sâu là mạng có nhiều lớp ẩn. Số lượng lớp ẩn tăng giúp mạng có khả năng học các đặc trưng phức tạp hơn.

* Học Có Giám Sát (Supervised Learning)

Sử dụng dữ liệu được gán nhãn (label).

Ví dụ: Phân loại hình ảnh mèo/dog (có nhãn "cat" và "dog").

* Học Không Giám Sát (Unsupervised Learning)

Xử lý dữ liệu không có nhãn.

Ví dụ: Phân cụm (clustering), giảm chiều dữ liệu (dimensionality reduction).

* Học Tăng Cường (Reinforcement Learning)

Học thông qua việc nhận phản hồi từ môi trường (phần thưởng hoặc hình phạt).

Ví dụ: AI chơi game cờ vua.

* Cách Hoạt Động
* Truyền Tín Hiệu Tiến (Forward Propagation):
* Dữ liệu đi qua từng lớp của mạng nơ-ron.
* Mỗi lớp thực hiện các phép tính toán để tạo ra đầu ra.
* Đầu ra cuối cùng được so sánh với giá trị thực tế để tính toán lỗi.
* Lan Truyền Ngược Lỗi (Backward Propagation):
* Dùng thuật toán lan truyền ngược để điều chỉnh trọng số và độ chệch của mạng.
* Dựa vào đạo hàm của hàm mất mát (loss function) để tối ưu hóa.
* Hàm Mất Mát (Loss Function):

Đo lường độ sai lệch giữa đầu ra dự đoán và giá trị thực tế.

Ví dụ:

* Mean Squared Error (MSE) cho bài toán hồi quy.
* Cross-Entropy cho bài toán phân loại.
* Thuật Toán Tối Ưu (Optimization Algorithms):
* Sử dụng để giảm hàm mất mát.
* Các thuật toán phổ biến:
* Gradient Descent
* Adam
* RMSprop
* Các Loại Mạng Nơ-ron Chính
* Mạng Nơ-ron Tích Chập (Convolutional Neural Networks - CNN)
* Sử dụng chủ yếu trong xử lý ảnh.
* Các lớp tích chập giúp phát hiện đặc trưng như cạnh, góc, hoặc chi tiết trong ảnh.
* Mạng Nơ-ron Hồi Quy (Recurrent Neural Networks - RNN)
* Dùng cho dữ liệu tuần tự như văn bản, âm thanh, hoặc chuỗi thời gian.
* Biến thể nổi bật: LSTM (Long Short-Term Memory), GRU (Gated Recurrent Unit).[2]
* Mạng Nơ-ron Đa Nhánh (Generative Adversarial Networks - GANs)
* Gồm hai mạng: generator (tạo dữ liệu giả) và discriminator (phân biệt dữ liệu thật/giả).
* Ứng dụng: Tạo ảnh, video giả lập, deepfake.
* Mạng Transformer
* Hiện đại, hiệu quả cho xử lý ngôn ngữ tự nhiên (NLP) và dữ liệu tuần tự.
* Ví dụ: BERT, GPT.

## 1.3 Ngôn ngữ lập trình và các thư viện sử dụng

### 1.3.1 Ngôn ngữ lập trình

* Python

Python là một ngôn ngữ lập trình bậc cao, thông dịch, và đa năng, được thiết kế để dễ học và dễ sử dụng. Nó được tạo ra bởi **Guido van Rossum** vào năm 1991 và ngày nay là một trong những ngôn ngữ lập trình phổ biến nhất thế giới.[7]

A blue and yellow snake logo

Description automatically generated

Hình 4. Python

**Đặc điểm chính của Python**

* **Cú pháp dễ đọc và học**: Cú pháp của Python đơn giản, gần gũi với ngôn ngữ tự nhiên, giúp lập trình viên dễ dàng nắm bắt ngay cả khi mới bắt đầu.
* **Đa nền tảng**: Python hoạt động trên hầu hết các hệ điều hành phổ biến như Windows, macOS, Linux, và có thể được sử dụng trên các thiết bị IoT.
* **Hướng đối tượng**: Python hỗ trợ lập trình hướng đối tượng (OOP), nhưng cũng rất linh hoạt với các phong cách lập trình thủ tục hoặc hàm.
* **Thư viện phong phú**: Python cung cấp hàng nghìn thư viện tích hợp sẵn và có thể mở rộng với các thư viện bên thứ ba như:
* **NumPy**, **Pandas**, **Matplotlib**: Xử lý dữ liệu.
* **TensorFlow**, **PyTorch**, **scikit-learn**: Học máy và AI.
* **Django**, **Flask**: Phát triển web.
* **OpenCV**, **Pillow**: Xử lý ảnh.
* **Cộng đồng lớn và hỗ trợ mạnh mẽ**: Python có một cộng đồng người dùng và nhà phát triển lớn, tài liệu phong phú, giúp việc học và giải quyết vấn đề trở nên dễ dàng.
* **Đa năng**: Python có thể được sử dụng trong nhiều lĩnh vực, từ phát triển phần mềm, phân tích dữ liệu, học máy, đến lập trình web, nhúng và tự động hóa.
* Html

HTML (HyperText Markup Language) là ngôn ngữ đánh dấu được sử dụng để tạo và cấu trúc nội dung trên các trang web. Đây là thành phần cốt lõi của web, thường được sử dụng cùng với CSS (Cascading Style Sheets) để định dạng giao diện và JavaScript để thêm chức năng tương tác.



Hình 5. Html

**Đặc điểm chính của HTML**

* **Cấu trúc nội dung**: HTML tổ chức nội dung thành các phần như tiêu đề, đoạn văn, danh sách, liên kết, hình ảnh, video, v.v.
* **Dễ học và sử dụng**: HTML có cú pháp đơn giản, dễ tiếp cận ngay cả đối với người mới học lập trình.
* **Không phải ngôn ngữ lập trình**: HTML không có logic như các ngôn ngữ lập trình mà chỉ là ngôn ngữ đánh dấu để mô tả nội dung.

**Ưu điểm của HTML**

* **Dễ học và sử dụng**:
* Cú pháp đơn giản, thân thiện với người mới bắt đầu.
* Không cần công cụ phức tạp, chỉ cần một trình soạn thảo văn bản và trình duyệt web.
* **Tính phổ biến và hỗ trợ tốt**:
* HTML được hỗ trợ trên tất cả các trình duyệt web hiện đại.
* Có tài liệu và cộng đồng lớn, dễ dàng tìm kiếm tài nguyên học tập hoặc giải quyết vấn đề.
* **Tương thích với các công nghệ khác**:
* Dễ dàng tích hợp với CSS và JavaScript để xây dựng các ứng dụng web mạnh mẽ.
* Làm việc tốt với các công cụ và framework như Bootstrap, React, Angular.
* **Không cần cấu hình phức tạp**: Không yêu cầu cài đặt môi trường hay server phức tạp khi phát triển.
* **Tính linh hoạt và mở rộng**:
* Hỗ trợ đa dạng các loại nội dung như văn bản, hình ảnh, âm thanh, video, biểu mẫu.
* Hỗ trợ các thẻ HTML5 mới để cải thiện khả năng tương tác và hỗ trợ đa phương tiện.

**Nhược điểm của HTML**

* **Không có tính năng động**: HTML chỉ là ngôn ngữ đánh dấu, không thể xử lý logic hoặc thực hiện các tác vụ phức tạp như các ngôn ngữ lập trình.
* **Thiếu khả năng bảo mật**: Dữ liệu hiển thị bằng HTML dễ bị sao chép hoặc tấn công nếu không được bảo vệ đúng cách.
* **Phụ thuộc vào các ngôn ngữ khác**: Để tạo một ứng dụng web hoàn chỉnh, HTML cần phải được kết hợp với CSS, JavaScript và các công nghệ backend.
* **Độ phức tạp tăng khi dự án lớn**: Quản lý cấu trúc HTML trở nên khó khăn khi ứng dụng trở nên phức tạp và có nhiều trang.
* **Hạn chế trong việc tối ưu hóa SEO nếu không biết cách**: Sử dụng sai thẻ hoặc không tối ưu cấu trúc HTML có thể ảnh hưởng đến khả năng xếp hạng trên công cụ tìm kiếm.
* **Css**

CSS là ngôn ngữ được sử dụng để định dạng và làm đẹp nội dung của trang web được tạo bằng HTML. Nó giúp cải thiện giao diện của trang web, từ màu sắc, phông chữ, đến bố cục và các hiệu ứng hình ảnh.

A blue and white logo

Description automatically generated

Hình 6.Css

**Đặc điểm chính của CSS**

* **Tách biệt nội dung và giao diện**: HTML để xây dựng cấu trúc nội dung, CSS để định dạng giao diện, giúp dễ bảo trì và quản lý mã nguồn.
* **Tái sử dụng mã**: Các quy tắc CSS có thể được áp dụng cho nhiều phần tử hoặc trang web, tiết kiệm thời gian viết mã.
* **Tương thích với tất cả các trình duyệt**: CSS hỗ trợ trên hầu hết các trình duyệt, giúp website hoạt động nhất quán trên nhiều nền tảng.
* **Tính kế thừa và phân tầng (Cascading)**: Các quy tắc CSS có thể kế thừa từ cha sang con, và áp dụng theo thứ tự ưu tiên (Inline > Internal > External).

**Ưu điểm của CSS**

* **Cải thiện hiệu suất làm việc**: Giúp thay đổi giao diện nhanh chóng mà không cần sửa đổi HTML.
* **Tăng tính linh hoạt**: Dễ dàng tùy chỉnh giao diện cho từng thiết bị (responsive design).
* **Tách biệt nội dung và thiết kế**: Giúp mã nguồn HTML sạch hơn, dễ đọc và quản lý.
* **Hỗ trợ hiệu ứng và hoạt hình**: CSS cho phép tạo ra các hiệu ứng chuyển động mà không cần JavaScript.
* **Dễ dàng duy trì và cập nhật**: Sử dụng tệp CSS bên ngoài để thay đổi giao diện của toàn bộ trang web chỉ bằng một lần chỉnh sửa.

**Nhược điểm của CSS**

* **Không hoàn toàn bảo mật**: Tệp CSS dễ bị sao chép hoặc xem bởi bất kỳ ai có quyền truy cập vào trang web.
* **Vấn đề tương thích trình duyệt**: Một số thuộc tính hoặc tính năng mới của CSS có thể không hoạt động trên các trình duyệt cũ.
* **Phức tạp khi dự án lớn**: Quản lý các tệp CSS trở nên khó khăn khi dự án phát triển lớn, đặc biệt khi không tổ chức rõ ràng.
* **Phụ thuộc vào trình duyệt**: Các hiệu ứng và tính năng hiển thị có thể khác nhau giữa các trình duyệt nếu không kiểm tra kỹ lưỡng.
* **JavaScript**

JavaScript là ngôn ngữ lập trình được sử dụng để tạo ra các tính năng tương tác trên các trang web. Nó chạy trong trình duyệt và giúp tạo ra các hiệu ứng động, kiểm tra tính hợp lệ của biểu mẫu, xử lý các sự kiện và thậm chí xây dựng các ứng dụng web hoàn chỉnh.

**Đặc điểm chính của JavaScript**

* **Ngôn ngữ lập trình phía client**: JavaScript thường được sử dụng để xử lý và thực thi các tác vụ trên phía client (trình duyệt) mà không cần kết nối với máy chủ.
* **Tương tác động với HTML và CSS**:JavaScript có thể thay đổi nội dung của trang web (DOM) và áp dụng các kiểu CSS động.
* **Xử lý sự kiện**:JavaScript có thể phản hồi các sự kiện như nhấp chuột, di chuột, nhập văn bản, v.v., giúp tạo ra trải nghiệm người dùng tương tác.
* **Hỗ trợ lập trình hướng đối tượng (OOP)**:JavaScript hỗ trợ lập trình hướng đối tượng, cho phép sử dụng các lớp và đối tượng trong việc phát triển ứng dụng.
* **Có thể chạy trên cả client và server**: Với Node.js, JavaScript có thể chạy trên server, giúp phát triển ứng dụng web toàn diện (full-stack).
* **Ưu điểm của JavaScript**
* **Tương tác trực tiếp và động**: JavaScript giúp các trang web trở nên sống động với các hiệu ứng động, thay đổi nội dung trang mà không cần tải lại trang.
* **Hỗ trợ đa nền tảng**: JavaScript có thể chạy trên hầu hết tất cả các trình duyệt và hệ điều hành.
* **Được hỗ trợ rộng rãi**: JavaScript là ngôn ngữ phổ biến nhất cho phát triển web và được cộng đồng hỗ trợ mạnh mẽ, với nhiều tài nguyên học tập và thư viện sẵn có.
* **Chạy nhanh**: Vì JavaScript được thực thi trên client (trình duyệt), các tác vụ sẽ được xử lý nhanh chóng mà không cần phải gửi yêu cầu đến máy chủ.
* **Tính linh hoạt cao**: JavaScript có thể được sử dụng cho nhiều mục đích khác nhau, từ phát triển giao diện người dùng đến xây dựng ứng dụng web và server.
* **Nhược điểm của JavaScript**
* **Vấn đề bảo mật**: Mã JavaScript có thể bị thay đổi hoặc khai thác nếu không được kiểm soát tốt, gây nguy cơ bảo mật cho ứng dụng.
* **Khó quản lý mã nguồn lớn**: Khi dự án phát triển lớn, mã JavaScript có thể trở nên khó quản lý nếu không sử dụng các kỹ thuật tổ chức mã đúng cách (như modularization, design patterns).
* **Không đồng nhất giữa các trình duyệt**: Các trình duyệt có thể xử lý JavaScript khác nhau, mặc dù vấn đề này đã được cải thiện đáng kể, nhưng vẫn có những khác biệt nhỏ.
* **Quản lý bộ nhớ**: JavaScript không có cơ chế thu gom rác (garbage collection) tối ưu trong mọi trường hợp, điều này có thể dẫn đến sự rò rỉ bộ nhớ nếu không được kiểm soát.

### 1.3.2 Các thư viện

* OpenCv **(Open Source Computer Vision Library)**

OpenCV là một thư viện mã nguồn mở nổi tiếng dành cho xử lý ảnh và thị giác máy tính (Computer Vision). Được phát triển bởi **Intel** vào năm 2000 và hiện nay được hỗ trợ bởi một cộng đồng lớn, OpenCV là một công cụ mạnh mẽ cho các ứng dụng xử lý ảnh, phát hiện đối tượng, nhận dạng khuôn mặt, và nhiều hơn nữa.[4]

* **Đặc điểm nổi bật của OpenCV**
* **Xử lý ảnh nhanh chóng và hiệu quả**:Hỗ trợ các thao tác xử lý ảnh cơ bản và phức tạp, từ làm mờ ảnh, chuyển đổi màu, đến phát hiện cạnh và đối tượng.
* **Tương thích đa nền tảng**:Hoạt động trên nhiều hệ điều hành như Windows, macOS, Linux, và Android/iOS.
* **Hỗ trợ nhiều ngôn ngữ**:OpenCV có thể được sử dụng với Python, C++, Java, và Matlab.
* **Tích hợp mạnh mẽ với thư viện AI**:OpenCV hỗ trợ tích hợp với các framework học máy và học sâu như TensorFlow, PyTorch, ONNX.
* **Hỗ trợ video và xử lý thời gian thực**:Xử lý ảnh từ camera và video trực tiếp, phù hợp với các ứng dụng giám sát hoặc phân tích hành vi.
* **Ứng dụng của OpenCV**
* **Nhận diện khuôn mặt**: Sử dụng phương pháp Haar Cascades hoặc HOG + SVM để phát hiện và nhận diện khuôn mặt trong ảnh hoặc video.
* **Phát hiện và theo dõi đối tượng**: Xác định các đối tượng chuyển động hoặc tĩnh từ camera, video giám sát.
* **Phân loại và nhận diện đối tượng**: Kết hợp với các mô hình học sâu để phân loại và phát hiện đối tượng trong ảnh/video.
* **Xử lý ảnh y tế**: Phân tích hình ảnh từ X-quang, MRI để phát hiện bệnh.
* **Nhận diện ký tự (OCR)**: Kết hợp với Tesseract để nhận diện chữ viết trong ảnh.
* **Xử lý ảnh nghệ thuật và hiệu ứng**:Áp dụng các bộ lọc, tạo hiệu ứng như tranh sơn dầu, vẽ phác thảo.
* **Thị giác máy tính trong robot**: Sử dụng OpenCV trong các hệ thống tự động như xe tự hành, robot công nghiệp.
* Ưu điểm:
* Thư viện mã nguồn mở và miễn phí.
* Hỗ trợ xử lý thời gian thực.
* Cộng đồng lớn và nhiều tài liệu học tập.
* Nhược điểm:
* Một số tính năng cần tích hợp thêm thư viện bên ngoài.
* Đôi khi cần kiến thức cơ bản về xử lý ảnh để sử dụng hiệu quả.
* **NumPy(Numerical Python)**

**NumPy** là một thư viện mạnh mẽ trong Python, được thiết kế để làm việc với mảng dữ liệu lớn (arrays) và thực hiện các tính toán số học hiệu quả. Đây là công cụ cơ bản cho các lĩnh vực như khoa học dữ liệu, học máy, và xử lý tín hiệu, thường được sử dụng kết hợp với các thư viện khác như Pandas, Matplotlib, và TensorFlow.

* **Đặc điểm nổi bật của NumPy**
* **Mảng đa chiều (ndarray)**: NumPy cung cấp đối tượng chính là ndarray, hỗ trợ lưu trữ và xử lý dữ liệu nhiều chiều hiệu quả hơn so với danh sách (list) trong Python.
* **Tính toán nhanh chóng**: Các phép toán được thực hiện trực tiếp trên mảng NumPy được tối ưu hóa, nhanh hơn so với sử dụng vòng lặp Python thông thường.
* **Các hàm toán học tích hợp**: NumPy cung cấp nhiều hàm toán học như cộng, trừ, nhân, chia, căn bậc hai, logarit, lượng giác, v.v.
* **Hỗ trợ đại số tuyến tính**: Bao gồm các phép tính như nhân ma trận, tìm định thức, tính eigenvalues, và giải hệ phương trình.
* **Tương thích với dữ liệu từ bên ngoài**: NumPy có thể đọc và ghi dữ liệu từ file CSV, txt, hoặc thậm chí từ các định dạng nhị phân.
* **Tích hợp tốt với các thư viện khác**: NumPy là nền tảng của nhiều thư viện Python phổ biến như Pandas, Scikit-learn, và Matplotlib.
* **Ứng dụng của NumPy**
* **Khoa học dữ liệu**: Làm việc với dữ liệu lớn, xử lý dữ liệu nhanh chóng trước khi phân tích.
* **Học máy**: Là nền tảng cho các thư viện học máy như TensorFlow, Scikit-learn.
* **Đại số tuyến tính**: Giải quyết các bài toán liên quan đến ma trận và vector.
* **Xử lý tín hiệu**: Áp dụng các phép toán và hàm số học để phân tích tín hiệu.
* Tensorflow

TensorFlow là một framework mã nguồn mở được phát triển bởi Google, dùng để xây dựng và huấn luyện các mô hình học máy (machine learning) và học sâu (deep learning). Nó hỗ trợ một loạt các thuật toán học máy và các công cụ mạnh mẽ để triển khai mô hình học sâu như mạng nơ-ron nhân tạo (ANN), mạng nơ-ron tích chập (CNN), và mạng nơ-ron hồi tiếp (RNN).[6]

* **Đặc điểm chính của TensorFlow**
* **Mã nguồn mở và miễn phí**:TensorFlow là mã nguồn mở, có thể sử dụng miễn phí và có cộng đồng đóng góp lớn.
* **Hỗ trợ đa dạng nền tảng**:TensorFlow có thể chạy trên nhiều nền tảng khác nhau, từ máy tính cá nhân đến các thiết bị di động, máy chủ và hệ thống phân tán (distributed systems).
* **Tính linh hoạt cao**:TensorFlow hỗ trợ cả việc xây dựng các mô hình học máy đơn giản lẫn phức tạp, giúp người dùng dễ dàng tùy chỉnh mô hình theo nhu cầu.
* **Hỗ trợ GPU**:TensorFlow hỗ trợ sử dụng GPU (Graphics Processing Unit) để tăng tốc quá trình huấn luyện mô hình, đặc biệt là với các mô hình học sâu, giúp giảm thời gian huấn luyện.
* **Hệ sinh thái phong phú**:TensorFlow có một hệ sinh thái đầy đủ, bao gồm các công cụ như **TensorFlow Lite** (cho thiết bị di động), **TensorFlow.js** (dành cho trình duyệt), **TensorFlow Extended (TFX)** (cho triển khai mô hình), v.v.
* **Ưu điểm của TensorFlow**
* **Tính mở rộng**:TensorFlow hỗ trợ cả các mô hình nhỏ gọn cho các ứng dụng đơn giản và các mô hình phức tạp cho các hệ thống học sâu quy mô lớn.
* **Tính tương thích với các công cụ khác**:TensorFlow dễ dàng tích hợp với các công cụ khác như Keras (API cao cấp của TensorFlow), Apache Kafka, Apache Spark, và các nền tảng khác.
* **Hỗ trợ GPU và TPU**:TensorFlow có thể tận dụng sức mạnh của GPU và TPU (Tensor Processing Unit), giúp tăng tốc quá trình huấn luyện mô hình.
* **Cộng đồng hỗ trợ mạnh mẽ**:TensorFlow có một cộng đồng người dùng lớn và một kho tài liệu phong phú, giúp người dùng dễ dàng tìm thấy các ví dụ, tài liệu hướng dẫn và các bài toán đã giải quyết.
* **Tích hợp với Google Cloud**:TensorFlow có thể dễ dàng tích hợp với Google Cloud AI, giúp triển khai mô hình lên môi trường đám mây để phục vụ cho các ứng dụng quy mô lớn.
* **Nhược điểm của TensorFlow**
* **Khó học đối với người mới**:TensorFlow có cú pháp và cách sử dụng khá phức tạp đối với những người mới bắt đầu. Tuy nhiên, điều này đã được cải thiện nhờ vào sự ra đời của Keras, một API cấp cao hơn, dễ sử dụng hơn.
* **Thời gian học hỏi lâu**:Việc làm quen với các khái niệm học sâu và hiểu cách cấu trúc mô hình trong TensorFlow có thể mất thời gian, đặc biệt là đối với các mô hình phức tạp.
* **Cấu hình phức tạp cho môi trường phân tán**:Đối với các ứng dụng học sâu quy mô lớn yêu cầu môi trường phân tán, việc cấu hình TensorFlow có thể khá phức tạp và đụng phải nhiều vấn đề kỹ thuật.

# Chương 2 Xây dựng mô hình

## 2.1 Mô Tả Bài Toán

**Mục Tiêu Bài Toán:** Bài toán là nhận diện cảm xúc từ khuôn mặt người thông qua các bức ảnh hoặc video. Mục tiêu chính là phát triển một hệ thống có khả năng phân loại cảm xúc của người dùng từ bảy loại cảm xúc phổ biến: **Angry (Giận dữ), Disgust (Ghê tởm), Fear (Sợ hãi), Happy (Vui vẻ), Sad (Buồn), Surprise (Ngạc nhiên), và Neutral (Trung lập)**. Hệ thống này có thể được áp dụng trong nhiều lĩnh vực như an ninh (giám sát hành vi), chăm sóc sức khỏe (theo dõi cảm xúc bệnh nhân), và trong các ứng dụng giải trí như trò chơi video hoặc thậm chí trong lĩnh vực marketing để hiểu phản ứng của khách hàng.

Input

* Dữ liệu đầu vào:
  + Hình ảnh hoặc video chứa khuôn mặt người (định dạng phổ biến: .jpg, .png, .mp4, v.v.).
  + Đường dẫn tới dữ liệu hoặc camera trực tiếp.
* Đặc tả của input:
  + Đối với ảnh:
    - Ảnh cần rõ nét và khuôn mặt phải chiếm phần lớn khung hình.
    - Kích thước khuyến nghị: 48x48 (nếu sử dụng dữ liệu chuẩn hóa).
  + Đối với video:
    - Video cần có độ phân giải tối thiểu để nhận diện khuôn mặt.
    - Frame rate tối thiểu: 10 FPS để đảm bảo độ chính xác khi phân tích theo thời gian thực.

Output

* Kết quả đầu ra:
  + Nhãn cảm xúc tương ứng với khuôn mặt được nhận dạng.
  + Xác suất hoặc độ tin cậy của mỗi nhãn (nếu có).
* Dạng xuất ra:
  + Văn bản: Tên cảm xúc (ví dụ: "Happy", "Sad").
  + Hình ảnh: Hình ảnh gốc có khung bao quanh khuôn mặt và nhãn cảm xúc gắn trên khung.
  + Nếu là video, hiển thị nhãn cảm xúc theo thời gian thực (trên từng frame).
* Đặc tả của output:
  + Với ảnh:
    - Ảnh đầu ra có thể được lưu với định dạng .jpg, .png.
  + Với video:
    - Video đầu ra có thể lưu dưới định dạng .mp4 hoặc hiển thị trực tiếp.
* Ví dụ output:
  + output: Happy (Confidence: 95%)
  + Ảnh kết quả:
    - Ảnh gốc với nhãn cảm xúc ở mỗi khuôn mặt.
  + Đối với video:
    - Nhãn cảm xúc thay đổi theo từng khung hình hiển thị.

Luồng xử lý tổng quát

1. Tiền xử lý:
   * Đọc ảnh/video.
   * Dò tìm và cắt khuôn mặt (sử dụng các mô hình như Haar Cascade, MTCNN, hoặc DLIB).
2. Trích xuất đặc trưng:
   * Sử dụng mô hình CNN (ví dụ: VGG, MobileNet, hoặc ResNet).
3. Phân loại:
   * Áp dụng mạng nơ-ron hoặc mô hình được huấn luyện để phân loại cảm xúc.
4. Hiển thị hoặc lưu kết quả.

## 2.2 Xây Dựng Hệ Thống

Kỹ thuật sử dụng

* Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) là một trong những loại mạng nơ-ron phổ biến và mạnh mẽ nhất trong lĩnh vực thị giác máy tính và học sâu. CNN đặc biệt hiệu quả trong việc xử lý và phân tích dữ liệu có cấu trúc lưới như ảnh, video, và dữ liệu không gian 2D. Điều này giúp CNN trở thành một công cụ cực kỳ mạnh mẽ cho các bài toán như nhận diện đối tượng, phân loại ảnh, phân đoạn ảnh, và nhiều ứng dụng khác trong thị giác máy tính.

Convolutional Neural Network (CNN) bắt nguồn từ các nghiên cứu về mạng neuron và thị giác sinh học. Năm 1980, Kunihiko Fukushima phát minh Neocognitron, mô hình đầu tiên sử dụng các lớp tích chập và pooling, mô phỏng cơ chế thị giác tự nhiên. Sau đó, vào năm 1989, Yann LeCun và đồng nghiệp phát triển LeNet-5, CNN hiện đại đầu tiên, ứng dụng trong nhận diện chữ viết tay, đánh dấu bước ngoặt quan trọng trong ứng dụng thực tế của mạng neuron.

CNN thực sự bùng nổ vào năm 2012 với sự ra đời của AlexNet, kiến trúc mạng sâu sử dụng GPU để huấn luyện và đạt hiệu suất vượt trội trong cuộc thi ImageNet. Từ đó, hàng loạt cải tiến như VGGNet, ResNet, và EfficientNet ra đời, giúp CNN được áp dụng rộng rãi trong nhiều lĩnh vực như nhận diện hình ảnh, xử lý video, y học, và robot. CNN đã trở thành một trụ cột quan trọng của trí tuệ nhân tạo nhờ sự kết hợp giữa lý thuyết, dữ liệu lớn và tiến bộ phần cứng.

CNN là một mô hình học sâu được thiết kế để tự động và hiệu quả học từ dữ liệu hình ảnh, trích xuất các đặc trưng không gian quan trọng mà không cần sự can thiệp thủ công. Với khả năng nhận diện các mẫu cơ bản như cạnh, góc, và các đối tượng phức tạp, CNN có thể sử dụng các đặc trưng này để đưa ra dự đoán chính xác.

Từ "Convolutional" trong CNN xuất phát từ thao tác tích chập (convolution) mà mô hình này sử dụng để xử lý dữ liệu hình ảnh. Convolution giúp mạng học và trích xuất các đặc trưng của hình ảnh, chẳng hạn như cạnh và hình dạng, thông qua các bộ lọc (filters) hoặc kernel. Đây là bước đầu tiên giúp mạng "nhìn" và hiểu được cấu trúc không gian của hình ảnh.

Khi bạn đưa một ảnh vào CNN, mỗi bộ lọc (kernel) sẽ "di chuyển" qua ảnh và áp dụng phép toán tích chập để tạo ra các đặc trưng. Sau đó, các đặc trưng này sẽ được đưa vào các lớp tiếp theo của mạng để học thêm các mẫu phức tạp hơn. Quá trình này giúp CNN phát hiện các đặc trưng từ đơn giản đến phức tạp, từ cạnh cơ bản đến đối tượng phức tạp như con người, xe hơi, động vật, v.v.

Convolution neural network gồm những lớp cơ bản sau:

* Convolutional layer

Đây chính là lớp đóng vai trò mấu chốt của CNN, khi layer này đảm nhiệm việc thực hiện mọi tính toán. Stride, padding, filter map, feature map là những yếu tố quan trọng nhất của convolutional layer.

* Cơ chế của CNN là tạo ra các filter áp dụng vào từng vùng hình ảnh. Các filter map này được gọi là ma trận 3 chiều, bên trong chứa các parameter dưới dạng những con số.
* Stride là sự dịch chuyển filter map theo pixel dựa trên giá trị từ trái sang phải.
* Padding: Là các giá trị 0 được thêm cùng lớp input.
* Feature map: Sau mỗi lần quét, một quá trình tính toán sẽ được thực hiện. Feature map sẽ thể hiện kết quả sau mỗi lần filter map quét qua input.

Relu layer

Còn có tên gọi khác là activation function, đây là một hàm được kích hoạt trong neural network. Có tác dụng mô phỏng các neuron có tỷ lệ truyền xung qua axon. Trong activation function chúng còn có hàm nghĩa là: Relu, Tanh, Sigmoid, Maxout, Leaky,… Relu layer được ứng dụng phổ biến trong việc huấn luyện nơ-ron do sở hữu nhiều ưu điểm tiên tiến.

* Pooling layer

Khi nhận phải đầu vào quá lớn, các lớp pooling layer sẽ được xếp giữa những lớp Convolutional layer nhằm mục đích giảm parameter. Pooling layer được chia thành 2 loại phổ biến là max pooling và average.

* Fully connected layer

Khi 2 lớp convolutional layer và pooling layer nhận được ảnh truyền, lớp này sẽ có nhiệm vụ xuất kết quả. Khi ta nhận được kết quả là model đọc được thông tin ảnh, ta cần phải tạo sự liên kết để cho ra nhiều output hơn. Đây chính là lúc các lập trình viên sử dụng fully connected layer. Hơn nữa, nếu fully connected layer có dữ liệu về hình ảnh thì chúng sẽ chuyển thành mục chưa được phân chia chất lượng.

A diagram of a network

Description automatically generated

Hình 7. Mô hình CNN

Suốt quá trình huấn luyện, CNN sẽ tự động học hỏi các giá trị thông qua lớp filter với “mẫu” là cách thức người dùng thực hiện. Điều này khá giống với cách bộ não con người nhận diện những vật thể trong tự nhiên.

Một cấu trúc cơ bản nhất của CNN sẽ bao gồm 3 phần chủ yếu, đó là:

* Local receptive field (trường cục bộ): Nhiệm vụ của trường cục bộ là phân tách và lọc dữ liệu cũng như thông tin ảnh, sau đó chọn ra các vùng ảnh có giá trị sử dụng cao nhất.
* Shared weights and bias (trọng số chia sẻ): Trong mạng CNN, thành phần này có tác dụng giảm thiểu tối đa lượng tham số có tác dụng lớn. Trong mỗi convolution sẽ chứa nhiều feature map khác nhau, mỗi feature lại có khả năng giúp nhận diện một số feature trong ảnh.
* Pooling layer (lớp tổng hợp): Pooling layer là lớp cuối cùng, với khả năng đơn giản hóa thông tin đầu ra. Khi đã hoàn tất tính toán và quét qua các lớp, pooling layer sẽ được tạo ra nhằm mục đích lược bớt các thông tin không cần thiết và tối ưu đầu ra. Điều này giúp người dùng nhận được kết quả ưng ý và đúng với yêu cầu hay mong muốn.
* Mạng CNN có gì đặc biệt?

Tự động trích xuất đặc trưng

Trái ngược với các phương pháp học máy truyền thống, nơi người ta phải trích xuất đặc trưng thủ công từ dữ liệu (ví dụ: thông qua histogram, SIFT, hoặc HOG), CNN tự động học và trích xuất các đặc trưng từ dữ liệu đầu vào. Điều này không chỉ giúp giảm bớt công việc thủ công mà còn mang lại khả năng nhận diện chính xác hơn khi mạng có thể tự phát hiện các đặc trưng quan trọng mà không bị ảnh hưởng bởi các quyết định của người lập trình.

Khả năng nhận diện đối tượng mạnh mẽ

CNN có khả năng nhận diện các đối tượng trong ảnh một cách mạnh mẽ và hiệu quả. Mạng này có thể phát hiện các đối tượng ngay cả khi chúng thay đổi về kích thước, vị trí hoặc góc độ trong ảnh, nhờ vào các lớp pooling và convolution giúp mạng học được các đặc trưng bất biến đối với các biến đổi này.

Khả năng xử lý ảnh có kích thước lớn

CNN có thể xử lý và học từ các ảnh có độ phân giải lớn mà không gặp phải vấn đề về việc trích xuất đặc trưng thủ công, làm cho nó đặc biệt hữu ích cho các bài toán yêu cầu phân tích hình ảnh quy mô lớn (ví dụ: nhận diện khuôn mặt hoặc phân loại ảnh với hàng triệu ảnh).

Các ứng dụng phổ biến của CNN

CNN đã đạt được nhiều thành tựu nổi bật trong các bài toán liên quan đến thị giác máy tính. Một số ứng dụng điển hình bao gồm:

* Nhận diện và phân loại đối tượng: CNN có thể phân loại ảnh thành các nhóm hoặc phân biệt các đối tượng trong ảnh (ví dụ: nhận diện mèo, chó, xe hơi).
* Nhận diện khuôn mặt: CNN được sử dụng trong nhiều ứng dụng nhận diện khuôn mặt, từ bảo mật cho đến ứng dụng phân tích cảm xúc.
* Phân đoạn ảnh: CNN có thể chia ảnh thành các vùng có ý nghĩa, ví dụ trong phân đoạn y tế, nơi mỗi vùng có thể đại diện cho một bộ phận cơ thể cụ thể.
* Nhận diện chữ viết tay: CNN cũng được áp dụng để nhận diện chữ viết tay, ví dụ trong việc quét các tài liệu

# Chương 3 Kết quả thực nghiệm

## 3.1 Dữ liệu

* Nguồn Dữ Liệu:
* Dữ liệu được sử dụng trong mô hình nhận diện cảm xúc là bộ dữ liệu **FER2013** (Facial Expression Recognition 2013), được chia thành các thư mục con tương ứng với từng cảm xúc (Angry, Disgust, Fear, Happy, Sad, Surprise, Neutral).
* Bộ dữ liệu này có sẵn trong một số trang web học máy như Kaggle và được sử dụng phổ biến trong các bài toán nhận diện cảm xúc từ khuôn mặt.
* Train:28709
* Test :7178
* Tiền Xử Lý Dữ Liệu:
* Dữ liệu ảnh được chuyển đổi sang dạng grayscale (ảnh xám) để giảm độ phức tạp tính toán và làm giảm số lượng thông tin không cần thiết, tập trung vào cấu trúc của khuôn mặt.
* Ảnh được rescale để các giá trị pixel nằm trong khoảng từ 0 đến 1, sử dụng công thức rescale=1./255. Điều này giúp chuẩn hóa các giá trị đầu vào và làm việc hiệu quả hơn với mô hình học sâu.
* Kỹ thuật tiền xử lý: Sử dụng ImageDataGenerator từ Keras, một công cụ tiện lợi để tự động tải và tiền xử lý dữ liệu ảnh. Cả dữ liệu huấn luyện và kiểm tra đều được chuẩn hóa (rescale) trước khi đưa vào mô hình.
* Chia Dữ Liệu:
* Bộ dữ liệu được chia thành hai phần chính:
  + Dữ liệu huấn luyện: Ở thư mục train\_data\_dir, bao gồm các ảnh phục vụ cho việc huấn luyện mô hình.
  + Dữ liệu kiểm tra: Ở thư mục test\_data\_dir, bao gồm các ảnh dùng để đánh giá hiệu quả của mô hình.

Chia tỷ lệ huấn luyện và kiểm tra:

* Dữ liệu được chia theo tỉ lệ 80-20 (80% cho huấn luyện và 20% cho kiểm tra). Cấu trúc này giúp mô hình học từ dữ liệu lớn và kiểm tra trên một lượng dữ liệu chưa thấy để đánh giá độ chính xác.

## 3.2 Độ đo

Công thức để tính độ đo Confidence (độ tin cậy) thường là giá trị xác suất cao nhất được mô hình dự đoán, biểu thị mức độ chắc chắn của mô hình về một nhãn cụ thể. Trong bài toán phân loại cảm xúc, confidence được tính như sau:

* Công Thức Tính Độ Đo Confidence:

Confidence = max(Softmax(Output))

Trong đó:

* + - Softmax(Output): Hàm softmax chuyển đổi output của mạng (logits) thành xác suất trên các nhãn.
    - max(...): Lấy giá trị xác suất cao nhất trong các xác suất của tất cả các nhãn.
* Diễn Giải:

Mô hình dự đoán một vector đầu ra cho mỗi hình ảnh hoặc khuôn mặt, ví dụ:

[0.1, 0.05, 0.15, 0.6, 0.05, 0.02, 0.03]

Đây là xác suất dự đoán cho các nhãn ['Angry', 'Disgust', 'Fear', 'Happy', 'Sad', 'Surprise', 'Neutral'].

* Giá trị lớn nhất trong vector này chính là confidence:

Confidence = 0.6 (cho nhãn 'Happy').

* Confidence được biểu diễn dưới dạng phần trăm trong ứng dụng:

Confidence Percentage = Confidence × 100%

Ví Dụ:

Nếu mô hình dự đoán cảm xúc 'Happy' với xác suất cao nhất là 0.6, độ tin cậy sẽ được hiển thị như sau:

Confidence Percentage = 0.6 × 100 = 60%.

## 3.2 Kết quả thực nghiệm

* Kết quả thực nghiệm với ảnh

A close-up of several men's faces

Description automatically generated

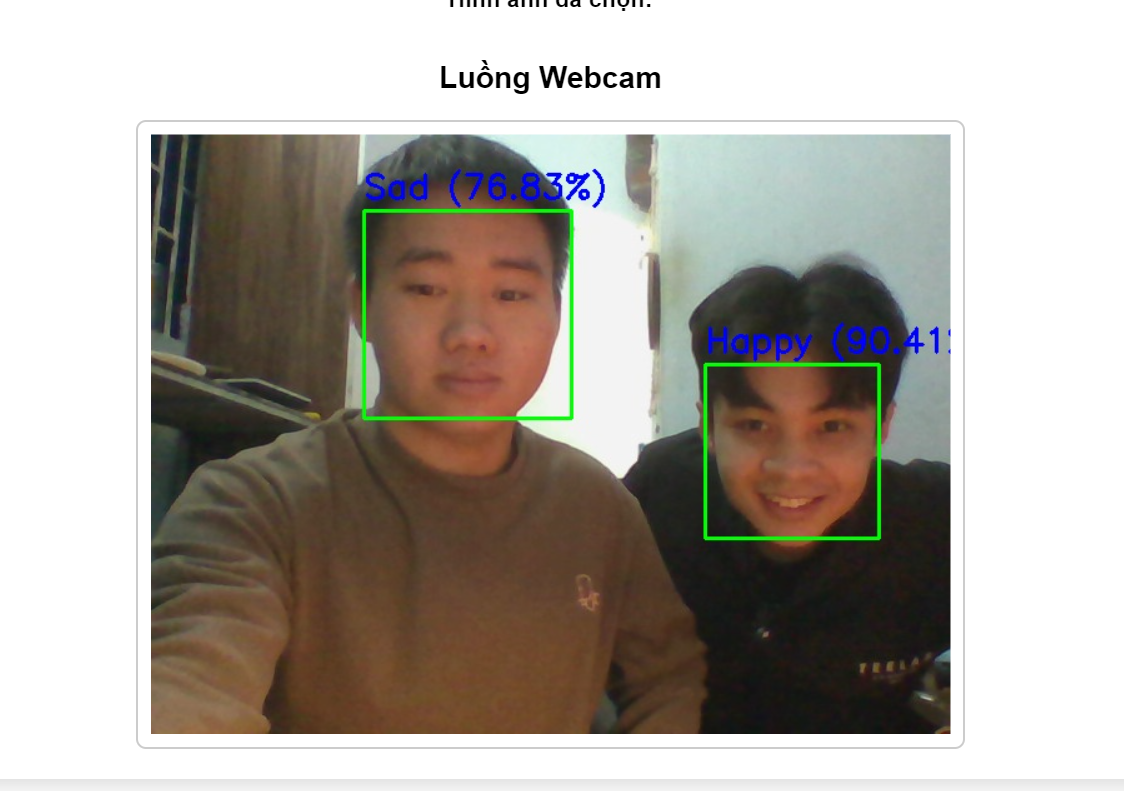
Hình 8. Kết quả thực nghiệm với ảnh

* Kết quả thực nghiệm với webcam

A person taking a selfie

Description automatically generated

Hình 9. Thực nghiệm webcam với 1 người



Hình 10.Thực nghiệm webcam với nhiều người

# Kết luận

* **Tóm tắt kết quả đạt được**:
* Bài tập nhận diện cảm xúc khuôn mặt đã sử dụng các kỹ thuật học máy và học sâu để nhận diện các cảm xúc cơ bản như vui, buồn, giận, sợ hãi, ngạc nhiên và khinh bỉ.
* Các thư viện Python như OpenCV, Dlib và fer đã được áp dụng để phát hiện khuôn mặt và phân loại cảm xúc.
* Mô hình đạt độ chính xác cao, nhưng vẫn gặp một số vấn đề như ảnh hưởng của ánh sáng, góc chụp và sự đa dạng trong biểu cảm khuôn mặt.
* **Khó khăn và hạn chế**:
* Mô hình chưa hoàn hảo do các yếu tố như ánh sáng không đồng đều, góc chụp khuôn mặt và sự thay đổi trong các biểu cảm cảm xúc.
* Một số cảm xúc có thể bị nhầm lẫn hoặc nhận diện không chính xác trong các tình huống thực tế.
* **Hướng phát triển tương lai**:
* Cải thiện độ chính xác: Thu thập thêm dữ liệu huấn luyện đa dạng hơn để tăng khả năng nhận diện các biểu cảm trong nhiều điều kiện khác nhau.
* Sử dụng các kỹ thuật học sâu: Áp dụng các mô hình học sâu như Mạng Neural Sâu (DNN) hoặc Mạng Chuyển Đổi (Transfer Learning) để cải thiện khả năng phân loại.
* Tích hợp vào các hệ thống thực tế: Mô hình có thể được áp dụng vào các lĩnh vực như trợ lý ảo, dịch vụ khách hàng thông minh, hoặc ứng dụng trong y tế để hỗ trợ chẩn đoán các rối loạn tâm lý.

# Tài liệu tham khảo

[1]**Nguyễn Ngọc Hoàng Anh** (2020), Xử Lý Ảnh với Python, Nhà xuất bản Khoa Học và Kỹ Thuật.(Hướng dẫn chi tiết cách xử lý ảnh cơ bản và nâng cao với Python, phù hợp cho người mới bắt đầu).

[2]**Nguyễn Tấn Trần Minh Khang** (2021), Deep Learning và Ứng Dụng trong Xử Lý Ảnh, Nhà xuất bản Bách Khoa Hà Nội.(Cuốn sách tập trung vào các ứng dụng học sâu cho xử lý ảnh, bao gồm phân loại, phát hiện đối tượng, và tạo ảnh).

[3]**Trịnh Minh Cường** (2019), Học Máy và Xử Lý Ảnh Thực Tế với Python, Nhà xuất bản Thông Tin và Truyền Thông.(Tích hợp kiến thức về Machine Learning và xử lý ảnh với Python, dùng thư viện phổ biến như OpenCV, PIL, và TensorFlow).

[4]**OpenCV Official Documentation**  
https://docs.opencv.org/ (Truy cập từ ngày 01/12/2024 - 03/12/2024)(Hướng dẫn chi tiết về các tính năng của thư viện OpenCV cho xử lý ảnh và video).

[5]**Imaging Library (Pillow)**  
<https://pillow.readthedocs.io/> (Truy cập từ ngày 01/12/2024 - 03/12/2024)(Tài liệu chính thức của Pillow, thư viện xử lý ảnh mạnh mẽ và dễ sử dụng trong Python).

[6]**TensorFlow Documentation**  
https://www.tensorflow.org/tutorials/images (Truy cập từ ngày 01/12/2024 03/12/2024)(Cung cấp các hướng dẫn và ví dụ thực hành về xử lý ảnh với các mô hình học sâu).

[7]**Python**   
https://toidicode.com/tong-quan-ve-python-161.html.Truy cập từ ngày 01/12/2024 03/12/2024)(Cung cấp các hướng dẫn và ví dụ thực hành về xử lý ảnh với các mô hình học sâu).