

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN TỐT NGHIỆP
NGÀNH: CÔNG NGHỆ THÔNG TIN
CHUYÊN NGÀNH: CÔNG NGHỆ PHẦN MỀM

ĐỀ TÀI:
**NHẬN DIỆN CẢM XÚC TRÊN KHUÔN MẶT
ỨNG DỤNG TRÍ TUỆ NHÂN TẠO**

Người hướng dẫn: **NINH KHÁNH DUY**
Sinh viên thực hiện: **HUỲNH BÁ THUẬN**
Số thẻ sinh viên: 102200034
Lớp: **20T1**

Đà Nẵng, 6/2024

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN TỐT NGHIỆP

NGÀNH: CÔNG NGHỆ THÔNG TIN
CHUYÊN NGÀNH: CÔNG NGHỆ PHẦN MỀM

ĐỀ TÀI:
NHẬN DIỆN CẢM XÚC TRÊN KHUÔN MẶT
ỨNG DỤNG TRÍ TUỆ NHÂN TẠO

Người hướng dẫn: TS. NINH KHÁNH DUY
Sinh viên thực hiện: HUỖNH BÁ THUẬN
Số thẻ sinh viên: 102200034
Lớp: 20T1

Đà Nẵng, 6/2024

TÓM TẮT

Tên đề tài: Nhận diện cảm xúc trên khuôn mặt ứng dụng trí tuệ nhân tạo

Sinh viên thực hiện: Huỳnh Bá Thuận

Số thẻ SV: 102200034

Lớp: 20T1

Bài toán nhận dạng cảm xúc bắt đầu được nghiên cứu từ những năm 1970 nhưng kết quả đạt được vẫn còn nhiều hạn chế. Hiện nay vấn đề này vẫn đang được nhiều người quan tâm bởi tính hấp dẫn cùng những vấn đề phức tạp của nó. Đây cũng là đề tài có tính ứng dụng cao như trong lĩnh vực y tế để nâng cao khả năng tương tác của robot với bệnh nhân, trong lĩnh vực quảng cáo để theo dõi cảm xúc của khách hàng từ đó đề ra phương hướng phát triển thực tế.

Vì tính ứng dụng của bài toán nên em đã thực hiện đề án này. Mục tiêu của đề án này là nghiên cứu và đánh giá về các phương pháp nhận dạng mặt người trong việc nhận dạng ra 7 cảm xúc: Vui, buồn, giận dữ, tự nhiên, bất ngờ, ghê tởm, khóc trên video, hình ảnh. Bằng cách sử dụng các phương pháp nhận dạng mặt người để xác định có người xuất hiện hay không từ đó phân tích các đặc điểm nhận dạng trên gương mặt đó để tìm ra cảm xúc gương mặt thông qua tập giá trị cảm xúc có sẵn.

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Huỳnh Bá Thuận

Số thẻ sinh viên: 102200034

Lớp: 20T1

Khoa: Công nghệ thông tin

Ngành: Công nghệ phần mềm

1. Tên đề tài đồ án: Nhận diện cảm xúc trên khuôn mặt ứng dụng trí tuệ nhân tạo
2. Đề tài thuộc diện: ☐ Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện
3. Các số liệu và dữ liệu ban đầu: không có
4. Nội dung các phần thuyết minh và tính toán:

Chương 1: Trình bày bài toán nhận diện cảm xúc trên khuôn mặt, cách triển khai giải quyết bài toán và các công cụ.

Chương 2: Trình bày những lý thuyết về thuật toán phát hiện khuôn mặt và kiến trúc CNN.

Chương 3: Trình bày về nguồn gốc của dữ liệu huấn luyện, phân bố của tập dữ liệu và phân tích chi tiết về phương pháp thực hiện và kết quả đạt được của quy trình huấn luyện và phát triển mô hình.

Chương 4: Trình bày quá trình xây dựng ứng dụng cho người dùng sử dụng.

Chương 5: Tổng kết lại những những kết quả đạt được đề hướng phát triển trong tương lai.

5. Các bản vẽ, đồ thị (ghi rõ các loại và kích thước bản vẽ): không có

6. Họ tên người hướng dẫn: TS. Ninh Khánh Duy

7. Ngày giao nhiệm vụ đồ án: 08/04/2024.

8. Ngày hoàn thành đồ án: 20/06/2024.

Đà Nẵng, ngày tháng năm 2024

Trưởng Bộ môn

Người hướng dẫn

LỜI NÓI ĐẦU

Đề tài “Nhận dạng cảm xúc trên khuôn mặt ứng dụng trí tuệ nhân tạo” là nội dung em chọn để nghiên cứu và làm đề án môn tốt nghiệp sau bốn năm theo học chương trình của khoa Công nghệ thông tin tại trường Đại học Bách Khoa – Đại học Đà Nẵng.

Để hoàn thành quá trình nghiên cứu và hoàn thiện đề án này, lời đầu tiên em xin chân thành cảm ơn sâu sắc đến thầy Ninh Khánh Duy. Thầy đã trực tiếp chỉ bảo và hướng dẫn em trong suốt quá trình nghiên cứu để em có thể hoàn thiện đề án này. Ngoài ra em xin chân thành cảm ơn các Thầy, Cô trong khoa Công nghệ thông tin đã giảng dạy và truyền thụ các kiến thức chuyên ngành trong những năm qua.

Cuối cùng, xin cảm ơn những người bạn của khoa Công nghệ thông tin đã hỗ trợ và giúp đỡ em trong những lúc khó khăn hay những lúc cần lời khuyên và sự nhận xét.

Trân trọng cảm ơn!

CAM ĐOAN

Tôi tên là Huỳnh Bá Thuận, sinh viên lớp 20T1, ngành Công nghệ phần mềm, khoa Công nghệ thông tin, Trường Đại học Bách khoa – Đại học Đà Nẵng.

Tôi xin cam đoan:

Đồ án tốt nghiệp với đề tài "Nhận diện cảm xúc trên khuôn mặt ứng dụng trí tuệ nhân tạo" là kết quả của quá trình nghiên cứu và làm việc nghiêm túc của bản thân tôi, dưới sự hướng dẫn của thầy Ninh Khánh Duy.

Tôi cam kết không sao chép, vi phạm bản quyền hoặc sử dụng trái phép các tài liệu, tài nguyên của bất kỳ cá nhân hay tổ chức nào. Mọi thông tin, số liệu trích dẫn trong đồ án đều được ghi rõ nguồn gốc và có chú thích đầy đủ theo quy định.

Tôi xin chịu trách nhiệm hoàn toàn trước nhà trường về tính chính xác và trung thực của những nội dung đã trình bày trong đồ án. Trân trọng!

Đà Nẵng, ngày 20 tháng 06 năm 2024

Sinh viên thực hiện

Huỳnh Bá Thuận

MỤC LỤC

TÓM TẮT

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

LỜI NÓI ĐẦU.....	i
CAM ĐOAN.....	ii
MỤC LỤC	iii
DANH SÁCH HÌNH VẼ	v
DANH SÁCH BẢNG.....	vi
DANH SÁCH CÁC CHỮ VIẾT TẮT	vii
MỞ ĐẦU	1
1. Mục đích thực hiện đề tài.....	1
2. Mục tiêu đề tài	1
3. Phạm vi và đối tượng nghiên cứu.....	1
4. Phương pháp nghiên cứu	1
5. Cấu trúc đồ án tốt nghiệp	2
Chương 1: TỔNG QUAN ĐỀ TÀI.....	3
1.1. Tổng quan về bài toán nhận dạng cảm xúc.....	3
1.2. Nhiệm vụ đề tài	4
1.3. Mô tả công việc	4
1.4. Phương pháp giải quyết	4
1.4.1. Khó khăn và thách thức	4
1.4.2. Phát hiện khuôn mặt (Face Detection).....	5
1.4.3. Phân loại cảm xúc (Facial Emotion Recognition)	5
Chương 2: CƠ SỞ LÝ THUYẾT	7
2.1. Phát hiện khuôn mặt	7
2.1.1. Phát hiện khuôn mặt (Face Detection).....	7
2.1.2. Định vị gương mặt	7
2.1.3. Phát hiện gương mặt.....	14
2.2. Mạng nơron tích chập	15

2.2.1. Giới thiệu sơ lược về mạng CNN (Convolutional Neural NetWorks)	15
2.2.2. Lớp Convolution	16
2.2.3. Lớp Pooling	16
2.2.4. Lớp Fully Connected	17
2.2.5. Lớp phi tuyến ReLU (Rectified Linear Unit)	17
Chương 3: GIẢI PHÁP NHẬN DẠNG CẢM XÚC TRÊN KHUÔN MẶT.....	19
3.1. Pipeline tổng quan.....	19
3.2. Phát hiện khuôn mặt (Face Detection)	19
3.2.1. Mô tả.....	20
3.2.2. Giải pháp	20
3.2.3. Kết quả thử nghiệm.....	20
3.3. Phân loại cảm xúc (Emotion Classification)	21
3.3.1. Thu thập dữ liệu.....	22
3.3.2. Tiền xử lý dữ liệu	22
3.3.3. Huấn luyện và hiệu chỉnh mô hình	24
3.3.4. Đánh giá mô hình	32
Chương 4 : XÂY DỰNG ỨNG DỤNG	34
4.1. Môi trường phát triển	34
4.2. Xây dựng API.....	35
4.3. Thiết kế giao diện.....	35
Chương 5 : KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	38
5.1. Kết Luận.....	38
5.2. Hướng phát triển	38
TÀI LIỆU THAM KHẢO.....	1

DANH SÁCH HÌNH VẼ

Hình 2.1 Các đặc trưng Haar – Like cơ bản	7
Hình 2.2 Các đặc trưng Haar – like cạnh (Edge features).....	8
Hình 2.3 Các đặc trưng Haar – like đường (Line features)	8
Hình 2.4 Các đặc trưng Haar – like xung quanh tâm (Center-surround features)	8
Hình 2.5 Tính giá trị ảnh tích phân tại điểm có tọa độ (x,y)	9
Hình 2.6 Tính nhanh giá trị của vùng ảnh D	10
Hình 2.7 Kết hợp các bộ phân loại yếu thành bộ phân loại mạnh.....	12
Hình 2.8 Các bộ phân lớp của Classifier of Cascade	13
Hình 2.9 Cấu trúc của phương pháp nhận dạng gương mặt bằng Haar – like – Adaboost	14
Hình 2.10 Sơ đồ nhận dạng gương mặt.....	14
Hình 2.11 Cấu trúc một mạng CNN.....	16
Hình 2.12 Tính tích chập giữa ảnh đầu vào 5x5 và Kernel 3x3.....	16
Hình 2.13 : Minh họa về Max Pooling	17
Hình 2.14 Minh họa về lớp Fully Connected	17
Hình 2.15 Hàm ReLU và tốc độ hội tụ khi so sánh với hàm tanh.	18
Hình 2.16 Hàm ReLU và tốc độ hội tụ khi so sánh với hàm tanh_	18
Hình 3.1 Giải pháp tổng quan của hệ thống	19
Hình 3.2 Kết quả thử nghiệm model FD	20
Hình 3.3 Quy trình xây dựng và kiểm thử mô hình phân loại cảm xúc dùng CNN.....	21
Hình 3.4 Phân bố tập dữ liệu huấn luyện	22
Hình 3.5 Thay đổi kích thước ảnh.....	23
Hình 3.6 Chuyển ảnh xám	23
Hình 3.7 Kết quả tăng cường dữ liệu	24
Hình 3. 8 Cấu trúc mô hình	28
Hình 3.11 Accuracy trên tập train và validation	30
Hình 3.12 Loss trên tập train và validation	31
Hình 3. 13 Biến thiên learning rate khi huấn luyện.....	31
Hình 4.1 Giao diện màn hình chính.....	35
Hình 4.2 Nhận diện bằng ảnh	36
Hình 4.3 Nhận diện realtime	37

DANH SÁCH BẢNG

Bảng 1.1 So sánh ưu nhược điểm của Machine Learning và Deep Learning	5
Bảng 3.1 Mô tả cấu trúc và kết quả huấn luyện của từng tổ hợp trên tập validation	25
Bảng 3.2 Các siêu tham số huấn luyện cho mô hình trên tập train	29
Bảng 3. 3 Đánh giá tốc độ thực thi của mô hình	33
Bảng 4.1 Giải thích các chức năng trên giao diện.....	36

DANH SÁCH CÁC CHỮ VIẾT TẮT

CNN	Convolution Neural Network
FACS	Facial Action Coding System
FER	Facial Expression Recognition
ML	Machine Learning
DL	Deep Learning
FD	Face Detection
EC	Emotion Classification
GPU	Graphics Processing Unit
ReLU	Rectified Linear Unit

MỞ ĐẦU

1. Mục đích thực hiện đề tài

Đề tài "Nhận diện cảm xúc bằng khuôn mặt ứng dụng trí tuệ nhân tạo" được thực hiện nhằm xây dựng một hệ thống có khả năng tự động nhận diện và phân loại các trạng thái cảm xúc của con người thông qua hình ảnh khuôn mặt. Việc nhận diện cảm xúc chính xác có thể ứng dụng trong nhiều lĩnh vực như giáo dục, y tế, marketing, và công nghệ giải trí, nhằm cải thiện trải nghiệm người dùng và tăng cường hiệu quả trong các tương tác người-máy.

2. Mục tiêu đề tài

Xây dựng một mô hình trí tuệ nhân tạo có khả năng nhận diện các cảm xúc cơ bản như vui, buồn, giận dữ, sợ hãi, ngạc nhiên, kinh tởm và bình thường từ hình ảnh khuôn mặt.

Đánh giá hiệu suất của mô hình thông qua các chỉ số đánh giá như độ chính xác, độ nhạy, và độ đặc hiệu.

Tạo ra một giao diện người dùng trực quan cho phép thử nghiệm và trình diễn khả năng nhận diện cảm xúc của hệ thống.

3. Phạm vi và đối tượng nghiên cứu

Phạm vi nghiên cứu: Đề tài tập trung vào việc nhận diện cảm xúc từ hình ảnh khuôn mặt tĩnh và động. Phạm vi nghiên cứu bao gồm việc thu thập dữ liệu, xử lý dữ liệu, xây dựng và huấn luyện mô hình, và kiểm tra hiệu quả của mô hình trên các bộ dữ liệu kiểm thử.

Đối tượng nghiên cứu: Các trạng thái cảm xúc cơ bản của con người được biểu hiện qua nét mặt, bao gồm vui, buồn, giận dữ, sợ hãi, ngạc nhiên, kinh tởm và bình thường. Đối tượng nghiên cứu cũng bao gồm các thuật toán và kỹ thuật trí tuệ nhân tạo, đặc biệt là học sâu (deep learning) và mạng nơ-ron tích chập (convolutional neural networks).

4. Phương pháp nghiên cứu

Thu thập dữ liệu: Sử dụng các bộ dữ liệu công khai có sẵn như FER-2013, CK+, hoặc thu thập dữ liệu mới thông qua việc chụp ảnh khuôn mặt của các tình nguyện viên trong các trạng thái cảm xúc khác nhau.

Xử lý dữ liệu: Tiền xử lý dữ liệu bao gồm cắt, xoay, và chuẩn hóa hình ảnh khuôn mặt. Dữ liệu được chia thành các tập huấn luyện, tập kiểm tra và tập xác thực.

Huấn luyện, hiệu chỉnh mô hình để tìm cấu trúc có độ chính xác cao nhất: Sử dụng các mô hình học sâu như mạng nơ-ron tích chập (CNN) để xây dựng hệ thống nhận diện cảm xúc.

Đánh giá mô hình: Sử dụng các chỉ số đánh giá như độ chính xác, độ nhạy, độ đặc hiệu và ma trận nhầm lẫn để đánh giá hiệu suất của mô hình trên tập dữ liệu kiểm tra.

Triển khai và thử nghiệm: Xây dựng một giao diện người dùng cho phép người dùng tải lên hình ảnh và nhận diện cảm xúc. Thử nghiệm hệ thống với các hình ảnh ngoài tập dữ liệu huấn luyện để đánh giá khả năng tổng quát hóa của mô hình.

5. Cấu trúc đề án tốt nghiệp

Chương 1: Giới thiệu tổng quan

- o Nội dung chương này giới thiệu bài toán nhận diện cảm xúc trên khuôn mặt, cách triển khai giải quyết bài toán và các công cụ

Chương 2: Cơ sở lý thuyết

- o Nội dung chương này diễn giải những lý thuyết về thuật toán phát hiện khuôn mặt và kiến trúc CNN.

Chương 3: Giải pháp nhận diện cảm xúc trên khuôn mặt

- o Nội dung chương này chủ yếu trình bày về nguồn gốc của dữ liệu huấn luyện, phân bố của tập dữ liệu và phân tích chi tiết về phương pháp thực hiện và kết quả đạt được của quy trình huấn luyện và phát triển mô hình.

Chương 4: Xây dựng ứng dụng

- o Nội dung chương này sẽ trình bày quá trình xây dựng ứng dụng cho người dùng sử dụng.

Chương 5: Kết luận và hướng phát triển

- o Nội dung chương này tổng kết lại những kết quả đạt được đề hướng phát triển trong tương lai.

Chương 1: TỔNG QUAN ĐỀ TÀI

1.1. Tổng quan về bài toán nhận dạng cảm xúc

Nhận dạng cảm xúc mặt người (Facial Expression Recognition - FER) giữ vai trò quan trọng trong tương tác giữa người và máy, trong các hệ thống an ninh với sự phổ biến của các camera an ninh ở các sân bay, văn phòng, trường đại học, máy ATM, ngân hàng... FER còn có thể được sử dụng trong nghiên cứu tâm lý học hành vi, chăm sóc khách hàng hay trong các hệ thống khuyến nghị dựa trên hình ảnh.

Biểu hiện trên khuôn mặt thể hiện tâm trạng hoặc trạng thái cảm xúc của một cá nhân ở một thời điểm cụ thể như buồn, hạnh phúc, tức giận... Paul Ekman[1] cho thấy sáu cảm xúc phổ biến là buồn (sad), hạnh phúc (happy), tức giận (angry), sợ hãi (fear), ghê tởm (disgust) và ngạc nhiên (surprise). Phát hiện khuôn mặt là bước đầu tiên của nhận dạng cảm xúc khuôn mặt, trong đó khuôn mặt được xác định từ ảnh đầu vào và loại bỏ các đối tượng khác (nếu có). Sau khi đã xác định được khuôn mặt người, bước tiếp theo thực hiện trích xuất đặc trưng và biểu diễn các đặc trưng đó. Với đặc trưng thu được, công việc cuối cùng là phân loại các đặc trưng vào một trong 6 loại cảm xúc phổ biến ở trên. Nhiều công trình nghiên cứu đã được thực hiện nhằm nâng cao độ chính xác cho bài toán FER. Ở giai đoạn phân loại, rất nhiều thuật toán cũng đã được nghiên cứu và công bố như Mô hình Markov ẩn (Hidden Markov Model)[7], Naïve Bayesian[8], KNN (K-nearest Neighbours)[3], SVM (Support Vector Machine)[12], Mạng nơ ron (Neural Networks - NN)[8], Mạng nơ ron tích chập (Convolutional Neural Networks – CNN)[4]... Trong đó, CNN thể hiện ưu thế vượt trội với khả năng xử lý tập dữ liệu lớn cũng như rất nhiều kiến trúc mạng được nghiên cứu và công bố... Với những ưu điểm đó của LBP và CNN, luận văn tập trung tìm hiểu, nghiên cứu kiến trúc mà dữ liệu đầu vào của CNN là đặc trưng LBP hay còn gọi là mạng nơ ron tích chập mẫu nhị phân cục bộ (LBP-CNN – Local

Đồ án này trình bày về những nghiên cứu của em trong đề tài ứng dụng xử lý ảnh để nhận dạng cảm xúc. Nhận dạng cảm xúc ở đây là xác định xem biểu cảm của gương mặt người thuộc lớp biểu cảm nào trong số tập biểu cảm đã được định nghĩa từ trước, ở đây có 7 biểu cảm đó là: Vui (Happy), buồn (Sad), giận dữ (Angry), ngạc nhiên (Surprise), khóc (Fear), ghê tởm (Disgust), tự nhiên (Neutral).

Đề án có trình bày mô hình và thuật toán như: Mạng neural tích chập CNN (Convolution Neural Network), thuật toán Violet – Jones, giải thuật Adaboost và bộ phân loại Cascade. Đề án này tập trung nghiên cứu phương pháp tìm đặc trưng trên gương mặt, sau đó trích các điểm đặc trưng này để làm mẫu huấn luyện trên các mạng Neural. Cuối cùng đánh giá kết quả đạt được sau quá trình kiểm tra và đối chiếu.

1.2. Nhiệm vụ đề tài

Nội dung 1: Tìm hiểu về Python, thư viện OpenCV, các công cụ xây dựng model CNN.

Nội dung 2: Tìm hiểu về xử lý ảnh, tìm hiểu các thuật toán huấn luyện và nhận dạng gương mặt người sử dụng OpenCV trong Python.

Nội dung 3: Tìm hiểu về CNN, thiết kế, xây dựng hệ thống nhận dạng cảm xúc trên gương mặt ứng dụng CNN.

1.3. Mô tả công việc

- Mô tả input và output của hệ thống:

+ *Input*: hình ảnh con người hoặc hình ảnh realtime từ camera.

+ *Output*: Kết quả nhận diện cảm xúc.

- Đề tài này sẽ tập trung vào việc xây dựng một hệ thống nhận diện cảm xúc khuôn mặt và giải quyết các vấn đề sau:

+ **Face detection**: Nhận diện khuôn mặt trong hình ảnh.

+ **Facial emotion recognition (FER)**: Phân tích khuôn mặt thu được từ bước trên để phân loại cảm xúc của khuôn mặt.

- Kết quả dự kiến đạt được:

+ Hệ thống hoàn chỉnh để xử lý bài toán **Facial emotion recognition**.

+ Website dịch vụ.

1.4. Phương pháp giải quyết

1.4.1. Khó khăn và thách thức

Tuy đạt được những kết quả tương đối khả quan do sự phát triển của nhiều phương pháp như trí tuệ nhân tạo, mạng nơ ron và đã có nhiều nghiên cứu về đề tài này [2][5][6][7]... nhưng bài toán nhận dạng cảm xúc mặt người vẫn còn tồn tại nhiều khó khăn và thách thức. Khó khăn trong việc phát hiện mặt người khi ảnh đầu vào có nhiều ngoại vật khác, hướng nghiêng của mặt hay mặt bị che khuất một phần cũng là một thách thức lớn. Ngoài ra khi xác định được khuôn mặt người, kích thước ảnh mặt

được cắt ra thường nhỏ nên độ phân giải tương đối thấp, chưa kể một số ảnh đầu vào có độ phân giải thấp gây khó khăn cho bước tiếp theo là trích xuất và biểu diễn đặc trưng.

Nhằm lẫn giữa các cảm xúc với nhau khi phân loại cũng là thách thức cho bài toán FER vì mức độ tương đồng. Hay môi trường sẽ tác động đến việc nhận biết các cảm xúc, như ánh sáng quá chói làm hệ thống không phân biệt sự khác biệt này.

1.4.2. Phát hiện khuôn mặt (Face Detection)

Trình phân loại tầng dựa trên tính năng Haar được đề xuất vào năm 2001 bởi Paul Viola và Michael Jones.[11] Đây là một thuật toán trích xuất đặc trưng cổ điển, có thể phát hiện hiệu quả các đối tượng cụ thể bằng cách sử dụng các đặc trưng Haar. Hiệu quả của nó được cải thiện nhờ đồ thị tích phân. Thuật toán AdaBoost được áp dụng để huấn luyện bộ phân loại mạnh. Trình phân loại tầng dựa trên tính năng Haar đã được sử dụng rộng rãi trong nhận diện khuôn mặt do hiệu suất tuyệt vời của nó.

1.4.3. Phân loại cảm xúc (Facial Emotion Recognition)

Trong lĩnh vực trí tuệ nhân tạo (AI) và khoa học dữ liệu, Machine Learning (ML) và Deep Learning (DL) là hai phương pháp chủ yếu và quan trọng nhất hiện nay.[2][6][10] Cả hai đều đã mang lại những tiến bộ vượt bậc trong việc giải quyết các bài toán phức tạp, từ nhận dạng hình ảnh đến phân tích dữ liệu lớn.

Bảng 1.1 So sánh ưu nhược điểm của Machine Learning và Deep Learning

Tên	Ưu điểm	Nhược điểm
ML[2]	Hiệu suất tốt với dữ liệu nhỏ Khả năng xử lý đặc trưng lớn Dễ hiểu	Khó khăn trong xử lý dữ liệu không tuyến tính Trích xuất đặc trưng thủ công Hiệu suất hạn chế
DL[9][10]	- Khả năng lọc các đặc trưng phức tạp một cách tự động - Xử lý tốt với dữ liệu không tuyến tính - Hiệu suất cao với dữ liệu lớn	- Đòi hỏi tài nguyên tính toán lớn - Dễ bị overfitting - Khó giải thích

Trong bối cảnh hiện nay, với sự phát triển nhanh chóng của công nghệ và khối lượng dữ liệu ngày càng lớn, việc chọn Deep Learning, đặc biệt là CNNs, cho đề tài là một quyết định hợp lý. CNNs không chỉ mang lại hiệu suất cao hơn trong các bài toán liên quan đến hình ảnh mà còn giúp tự động hóa quá trình trích xuất đặc trưng, giảm thiểu công sức và tối ưu hóa quá trình phát triển [4][6]. Dù DL đòi hỏi tài nguyên tính toán lớn hơn, nhưng với những tiến bộ trong phần cứng và dịch vụ điện toán đám mây, việc triển khai các mô hình DL ngày càng trở nên dễ dàng và khả thi hơn.

Chương 2: CƠ SỞ LÝ THUYẾT

2.1. Phát hiện khuôn mặt

2.1.1. Phát hiện khuôn mặt (Face Detection)

Đây là một lĩnh vực nghiên cứu của ngành Computer Vision tự động xác định hoặc phát hiện khuôn mặt nào đó từ một bức hình ảnh kỹ thuật số hoặc một khung hình video từ một nguồn video. Một trong những cách để thực hiện điều này là so sánh các đặc điểm khuôn mặt chọn trước từ hình ảnh và một cơ sở dữ liệu về khuôn mặt.

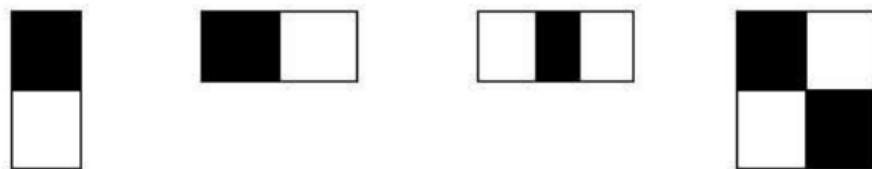
Có rất nhiều phương pháp giải quyết bài toán xác định khuôn mặt người trên ảnh 2D dựa vào các hướng tiếp cận khác nhau. Phương pháp Haar – Like – Adaboost của hai tác giả Paul Viola và Michael J.Jones là phương pháp xác định mặt người dựa theo hướng tiếp cận diện mạo.

Hướng tiếp cận diện mạo tức là hệ thống phát hiện khuôn mặt sẽ được học các mẫu của khuôn mặt từ một tập ảnh mẫu. Sau khi quá trình học hay quá trình huấn luyện (Training) thực hiện xong, hệ thống sẽ rút ra được những tham số để phục vụ cho quá trình nhận dạng. Vì vậy tiếp cận trên diện mạo còn được biết đến với tên gọi tiếp cận theo phương pháp học máy. Bài báo cáo tập trung chủ yếu vào hướng tiếp cận diện mạo, sử dụng phương pháp Haar – like – Adaboost.

2.1.2. Định vị gương mặt

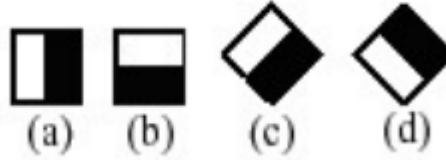
- Đặc trưng Haar – Like

Đặc trưng Haar Like được tạo thành bằng việc kết hợp các hình chữ nhật đen, trắng với nhau theo một trật tự, một kích thước nào đó. Hình dưới đây mô tả 4 đặc trưng Haar Like cơ bản như sau:

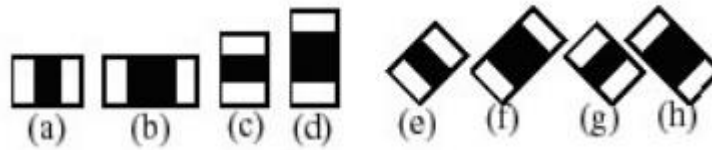


Hình 2.1 Các đặc trưng Haar – Like cơ bản

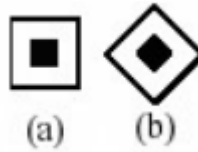
Để phát hiện khuôn mặt các đặc trưng Haar Like cơ bản trên được mở rộng (Lienhart, Kurnov, & Pisarevsky, 2002; Lienhart & Maydt, 2002) thành nhóm các đặc trưng cạnh, đặc trưng đường và đặc trưng trung tâm.



Hình 2.2 Các đặc trưng Haar – like cạnh (Edge features)



Hình 2.3 Các đặc trưng Haar – like đường (Line features)



Hình 2.4 Các đặc trưng Haar – like xung quanh tâm (Center-surround features)

Các hình phía trên miêu tả các đặc trưng mở rộng của các đặc trưng Haar – like cơ sở.

Giá trị của một đặc trưng Haar – like là sự khác biệt giữa tổng các giá trị xám của các pixel trong vùng “đen” với tổng các giá trị xám của các pixel trong vùng “trắng”

$$f(x) = \text{Tổng vùng đen(pixel)} - \text{Tổng vùng trắng(pixel)}$$

Từ những giá trị đặc trưng này đưa vào bộ phân loại Adaboost ta sẽ xác được ảnh có khuôn mặt hay không.

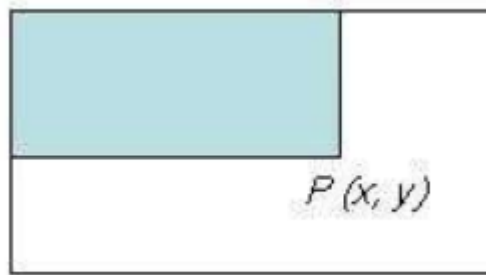
Sử dụng giá trị này, so sánh với các giá trị của các giá trị Pixel thô, các đặc trưng Haar – like có thể tăng/giảm sự thay đổi bên trong hay bên ngoài lớp khuôn mặt người (in-class/out-of-class), do đó sẽ làm cho bộ phân loại dễ hơn.

Như vậy ta có thể thấy rằng, để tính các giá trị của đặc trưng Haar-like, ta phải tính tổng của các vùng pixel trên ảnh.

Nhưng để tính toán các giá trị của các đặc trưng Haar-like cho tất cả các vị trí trên ảnh đòi hỏi chi phí tính toán khá lớn, do không đáp ứng được cho các ứng dụng đòi hỏi tính run-time. Do đó Viola và Jones đưa ra một khái niệm gọi là Integral Image (Ảnh tích hợp).[3]

- Ảnh tích hợp (Integral Image):

Đây là một công cụ giúp việc tính toán các giá trị đặc trưng Haar – Like nhanh hơn. Ảnh tích hợp là một mảng hai chiều với kích thước bằng kích thước của ảnh cần tính giá trị đặc trưng Haar – Like, với mỗi phần tử của mảng này được tính bằng cách tính tổng của điểm ảnh phía trên (dòng 1) và bên trái (cột 1) của nó. Bắt đầu từ vị trí trên, bên trái đến vị trí dưới, phải của ảnh, việc tính toán này đơn thuần chỉ dựa trên phép cộng số nguyên đơn giản, do đó tốc độ thực hiện rất nhanh. Dưới đây là mô tả cách tính ảnh tích phân.



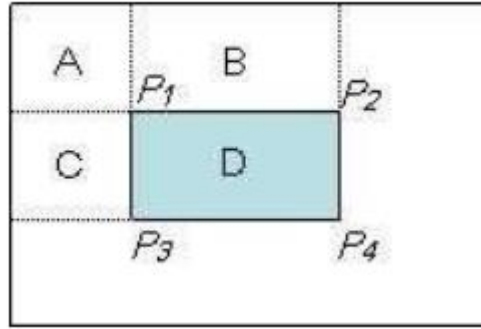
Hình 2.5 Tính giá trị ảnh tích phân tại điểm có tọa độ (x,y)

Giá trị của ảnh tích phân tại điểm P có tọa độ (x,y) được tính như sau:

$$P(x,y) = \sum_{x' \leq x, y' \leq y} i(x',y')$$

Sau khi đã tính được ảnh tích hợp, việc tính tổng các giá trị mức xám tại một vùng ảnh bất kỳ nào đó trên ảnh thực hiện theo cách sau, ví dụ tính giá trị của vùng D trong hình 5 như sau:

$$D = A + B + C + D - (A + B) - (A + C) + A$$



Hình 2.6 Tính nhanh giá trị của vùng ảnh D

Nếu như là ảnh xám bình thường thì để tính D ta phải tính tổng tất cả các giá trị điểm ảnh trong D, miền D càng lớn thì số phép cộng càng nhiều. Nhưng với ảnh tích hợp dù miền D có kích thước như thế nào thì D cũng chỉ cần tính thông qua 4 giá trị tại 4 đỉnh.

$$\text{Đặt: } P_1 = A, P_2 = A + B, P_3 = A + C, P_4 = A + B + C + D$$

$$\Rightarrow D = P_1 + P_4 - P_2 - P_3$$

Khi áp dụng vào tính toán các giá trị đặc trưng ta có thể thấy:

Đặc trưng hai hình chữ nhật (đặc trưng cạnh) được tính thông qua 6 giá trị điểm ảnh tích hợp

Đặc trưng ba hình chữ nhật (đặc trưng đường) và đặc trưng tâm – xung quanh được tính thông qua 8 điểm ảnh tích hợp.

Đặc trưng 4 hình chữ nhật (đặc trưng chéo) được tính thông qua 9 giá trị điểm ảnh tích hợp.

Trong khi đó nếu tính như định nghĩa thì các giá trị cần tính toán lên tới hàng trăm. Điều này làm tăng tốc độ xử lý một cách đáng kể.

- AdaBoost

AdaBoost (Freund & Schapire, 1995) là một bộ phân loại mạnh phi tuyến phức, hoạt động trên nguyên tắc kết hợp tuyến tính các bộ phân loại yếu (weak classifiers) để tạo nên một bộ phân loại mạnh (strong classifiers). AdaBoost sử dụng trọng số (weight) để đánh dấu các mẫu khó nhận dạng. Trong quá trình huấn luyện cứ mỗi bộ phân loại yếu được xây dựng thì thuật toán sẽ tiến hành cập nhật lại trọng số để chuẩn bị cho việc xây dựng bộ phân loại tiếp theo. Cập nhật bằng cách tăng trọng số của các

mẫu nhận dạng sai và giảm trọng số của các mẫu được nhận dạng đúng bởi bộ phân loại yếu vừa được xây dựng. Bằng cách này thì bộ phân loại sau có thể tập trung vào các mẫu mà bộ phân loại trước đó làm chưa tốt. Cuối cùng các bộ phân loại yếu sẽ được kết hợp lại tùy theo mức độ tốt của chúng để tạo nên một bộ phân loại mạnh.

Bộ phân loại yếu h_k được biểu diễn như sau:

$$h_k = \begin{cases} 1 & \text{nếu } p_k f_k(x) < p_k \theta_k \\ 0 & \text{ngược lại} \end{cases}$$

Trong đó:

x : Cửa sổ con cần xét.

θ_k : Ngưỡng.

f_k : Giá trị của đặc trưng Haar – like .

p_k : Hệ số quyết định chiều của phương trình.

Khi giá trị của đặc trưng Haar – like k : f_k tại cửa sổ con x vượt qua một ngưỡng θ_k thì bộ phân lớp $h_k(x)$ sẽ kết luận cửa sổ con x là khuôn mặt ($h_k(x) = 1$), còn nếu f_k không vượt qua ngưỡng đó thì không là khuôn mặt.

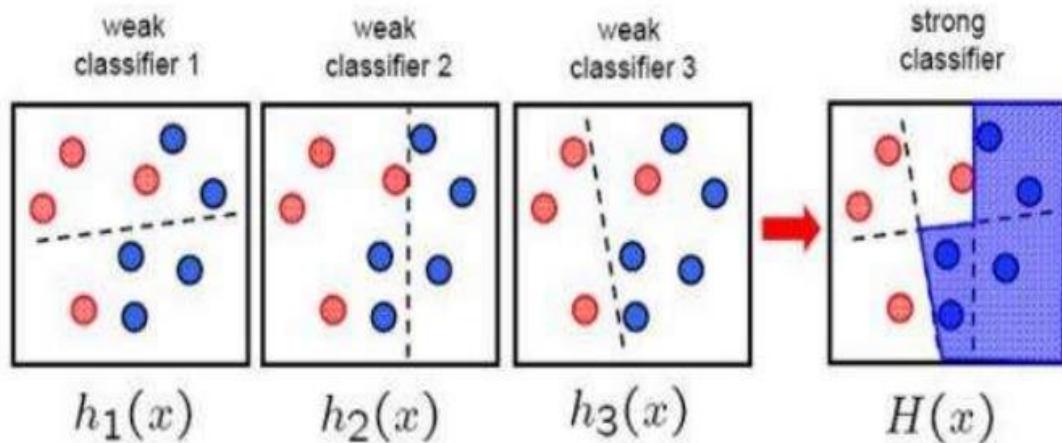
AdaBoost sẽ kết hợp các bộ phân loại yếu thành các bộ phân loại mạnh như sau:

$$H(x) = \text{Sign}(\sum_{m=1}^M \alpha^m h^m(x))$$

Trong đó:

$\alpha_m \geq 0$: hệ số chuẩn hóa cho các bộ phân loại yếu

h_m : Các bộ phân loại yếu



Hình 2.7 Kết hợp các bộ phân loại yếu thành bộ phân loại mạnh

Sử dụng phương pháp máy học Adaboost để xây dựng bộ phân loại mạnh có thể có độ chính xác trên 60%.

- Mô hình phân tầng Cascade

Cascade of Boosted Classifiers là mô hình phân tầng với mỗi tầng là một mô hình AdaBoost sử dụng bộ phân lớp yếu là cây quyết định với các đặc trưng Haar – Like.

Trong quá trình huấn luyện, bộ phân lớp phải duyệt qua tất cả các đặc trưng của mẫu trong tập huấn luyện. Giả sử, bằng Adaboost ta có được một bộ phân lớp

mạnh gồm 10 bộ phân lớp yếu, nếu như vậy thì tại tất cả các cửa sổ con trên tấm ảnh ta đều phải dùng cả 10 bộ phân lớp yếu. Việc này tốn rất nhiều thời gian. Mặc dù ta thấy những cửa sổ thật sự làm khuôn mặt rất ít và tại những cửa sổ không phải là khuôn mặt ta có thể loại bỏ mà chỉ cần dùng một bộ phân lớp mạnh gồm ít hơn 10 bộ phân lớp yếu. Vì vậy người ta giải quyết vấn đề này bằng chuỗi các bộ phân lớp (Cascade of Classifier).

Ta sẽ có một chuỗi các bộ phân lớp, trong đó mỗi bộ phân lớp được xây dựng bằng thuật toán Adaboost.

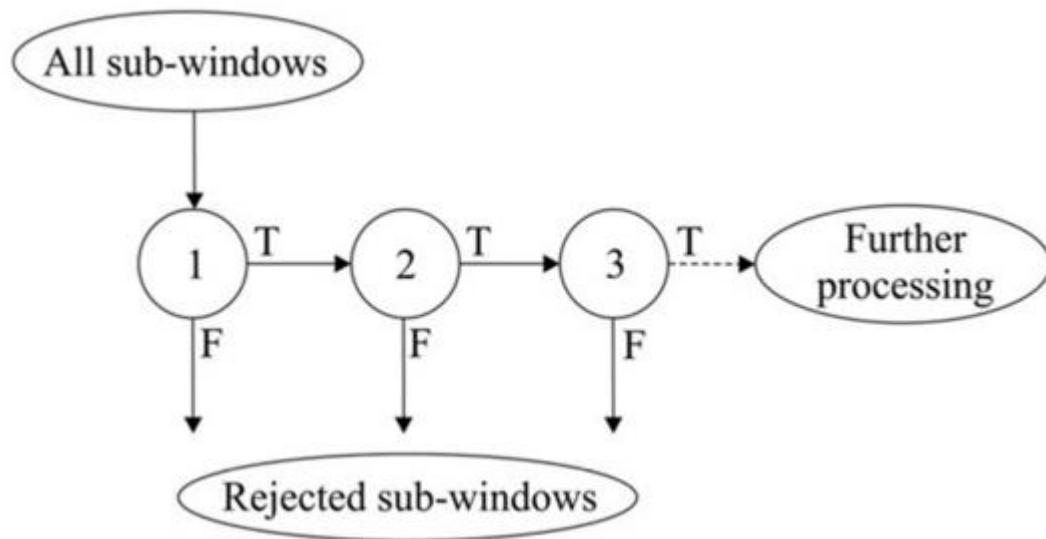
Bây giờ, ta đưa tất cả các cửa sổ con đi qua chuỗi các bộ phân lớp này:

- Bộ phân lớp đầu tiên sẽ loại bỏ phần lớn các cửa sổ không phải khuôn mặt (negative sub window). Ở đây, bộ phân lớp này rất đơn giản và do đó, độ phức tạp tính toán cũng rất thấp. Tất nhiên, vì rằng nó đơn giản nên trong số các cửa sổ được nhận

dạng là khuôn mặt sẽ có một lượng lớn cửa sổ bị nhận dạng sai (Không phải là khuôn mặt).

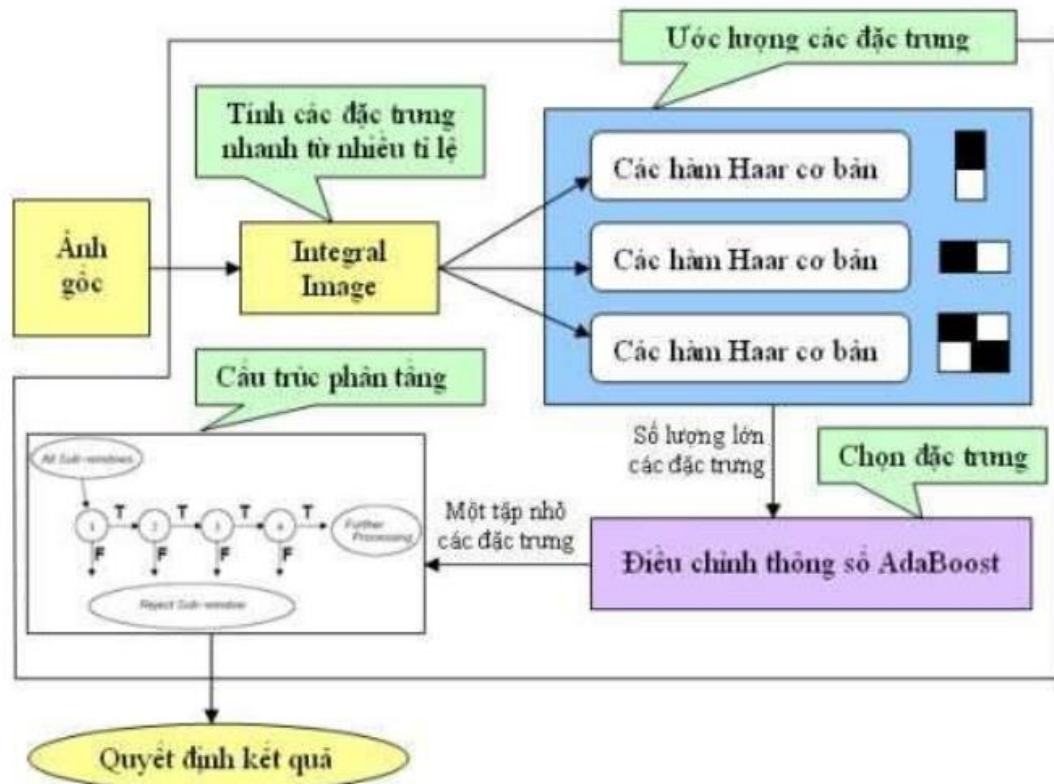
- Những cửa sổ được cho đi qua bởi bộ phân lớp đầu sẽ được xem xét bởi bộ phân lớp sau đó: nếu bộ phân lớp cho rằng đó không phải là khuôn mặt thì ta loại bỏ; Nếu bộ phân lớp cho rằng đó là khuôn mặt thì ta lại cho đi qua và chuyển đến bộ phân lớp phía sau.

- Những bộ phân lớp càng về sau thì càng phức tạp hơn, đòi hỏi sự tính toán nhiều hơn. Người ta gọi những cửa sổ con (mẫu) mà bộ phân lớp không loại bỏ được là những mẫu khó nhận dạng. Những mẫu này càng đi sâu vào trong chuỗi các bộ phân lớp thì càng khó nhận dạng. Chỉ những cửa sổ đi qua được tất cả các bộ phân lớp thì ta mới quyết định đó là khuôn mặt.



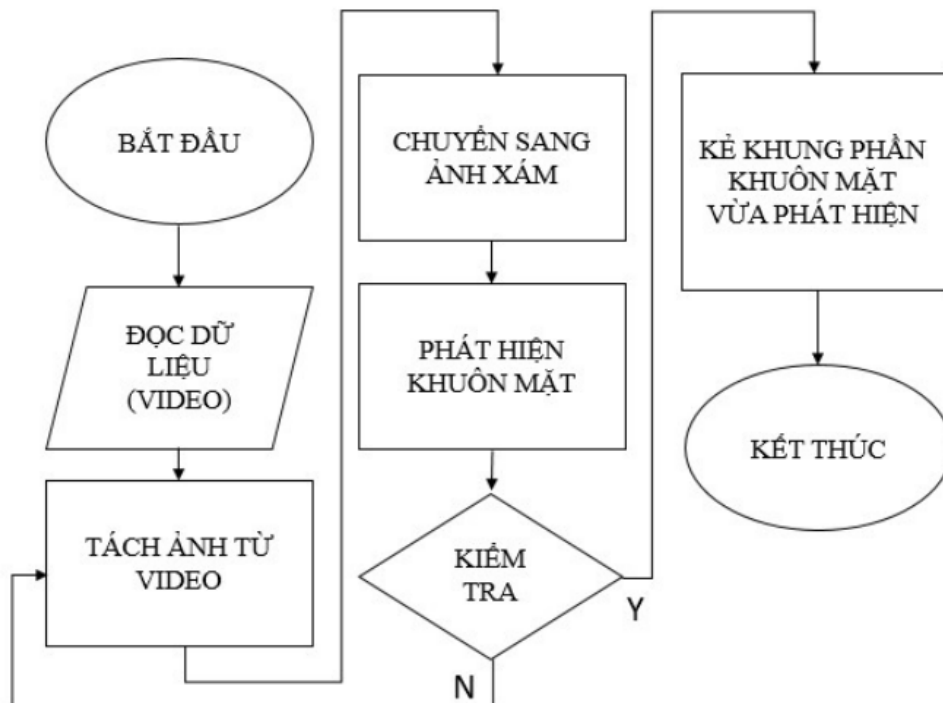
Hình 2.8 Các bộ phân lớp của Classifier of Cascade

Mô hình Cascade of Classifiers được xây dựng nhằm rút ngắn thời gian xử lý, giảm thiểu nhận dạng lầm (False Alarm) cho bộ phân loại. Cascade Trees gồm nhiều tầng (Stage hay còn gọi là Layer), mỗi tầng là một mô hình Adaboost với bộ phận lớp yếu là các cây quyết định. Một mẫu để được phân loại là đối tượng thì nó cần phải đi qua hết tất cả các tầng. Các tầng sau được huấn luyện bằng những mẫu âm Negative (không phải mặt người) mà tầng trước nó nhận dạng sai, tức là nó sẽ tập trung học từ các mẫu background khó hơn, do đó sự kết hợp các tầng AdaBoost này lại sẽ giúp bộ phân loại giảm thiểu nhận dạng lầm. Với cấu trúc này, những mẫu background dễ nhận dạng sẽ bị loại ngay từ những tầng đầu tiên, giúp đáp ứng tốt nhất thời gian xử lý và vẫn duy trì được hiệu quả phát hiện khuôn mặt.



Hình 2.9 Cấu trúc của phương pháp nhận dạng gương mặt bằng Haar – like – Adaboost

2.1.3. Phát hiện gương mặt



Hình 2.10 Sơ đồ nhận dạng gương mặt

2.2. Mạng nơ-ron tích chập

2.2.1. Giới thiệu sơ lược về mạng CNN (Convolutional Neural Networks)

Mạng nơ-ron tích chập (CNNs)(Lecun, 1998) là một mô hình học sâu có thể xây dựng được các hệ thống phân loại với độ chính xác cao. Ý tưởng của CNNs được lấy cảm hứng từ khả năng nhận biết thị giác của não người. Để có thể nhận biết được các hình ảnh trong vỏ não người có hai loại tế bào là tế bào đơn giản và tế bào phức tạp (Hubel, 1968). Các tế bào đơn giản phản ứng với các mẫu hình dạng cơ bản ở các vùng kích thích thị giác và các tế bào phức tạp tổng hợp thông tin từ các tế bào đơn giản để xác định các mẫu hình dạng phức tạp hơn. Khả năng nhận biết các hình ảnh của não người là một hệ thống xử lý hình ảnh tự nhiên mạnh mẽ và tự nhiên nên CNNs được phát triển dựa trên ba ý tưởng chính:

- Tính kết nối cục bộ (Local connectivity hay compositionality).
- Tính bất biến (Location invariance).
- Tính bất biến đối với quá trình chuyển đổi cục bộ (Invariance to local transition)(LeCun, 2015).

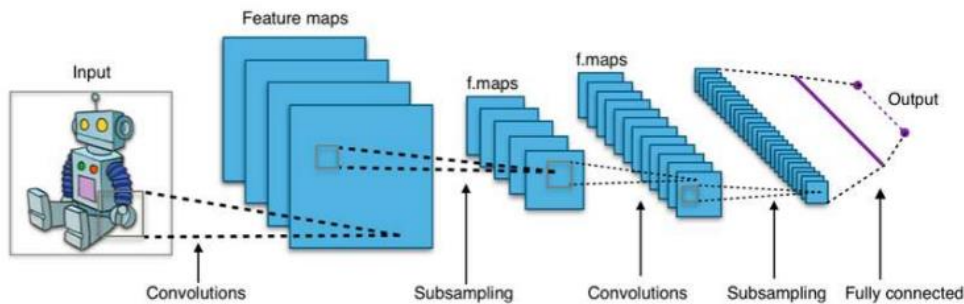
CNNs là một dạng mạng nơ-ron chuyên dụng để xử lý các dữ liệu dạng lưới 1 chiều như dữ liệu âm thanh, dữ liệu MGE hoặc nhiều chiều như dữ liệu hình ảnh, video. Cấu trúc cơ bản của CNNs gồm các lớp tích chập (Convolution layer), lớp phi tuyến (Nonlinear layer) và lớp lọc (Pooling layer), như Hình 11.

Các lớp tích chập kết hợp với các lớp phi tuyến sử dụng các hàm phi tuyến như ReLU hay tanh để tạo ra thông tin trừu tượng hơn (Abstract/higher-level) cho các lớp tiếp theo. Các lớp liên kết trong CNNs Hình 11: Convolutional Neural Network được với nhau thông qua cơ chế tích chập. Lớp tiếp theo là kết quả tích chập từ lớp trước đó vì vậy CNNs có được các kết nối cục bộ vì mỗi nơ-ron ở lớp tiếp theo sinh ra từ một bộ lọc được áp đặt lên một vùng cục bộ của lớp trước đó.

Nguyên tắc này được gọi là kết nối cục bộ (Local connectivity). Mỗi lớp như vậy được áp đặt các bộ lọc khác nhau. Một số lớp khác như lớp poolong/subsampling dùng để lọc lại các thông tin hữu ích hơn bằng cách loại bỏ các thông tin nhiễu.

Trong suốt quá trình huấn luyện, CNNs sẽ tự động học các tham số cho các lớp. Lớp cuối cùng được gọi là lớp kết nối đầy đủ (Fully connect layer) dùng để phân lớp. Mạng nơ-ron tích chập dùng để phân lớp dữ liệu MGE có kiến trúc mạng gồm nhiều tầng. Lớp đầu vào của mạng (Input layer) là một ma trận cường độ biểu hiện gen của

dữ liệu MGE. Lớp đầu ra của mạng (Output layer) là số lớp của dữ liệu. Giữa lớp đầu vào và đầu ra gồm nhiều tầng tích chập và một lớp kết nối đầy đủ. Mỗi tầng gồm một lớp tích chập, một lớp phi tuyến và một lớp lọc. Dữ liệu khi được huấn luyện qua các tầng tích chập sẽ được phân lớp phủ lớp kết nối đầy đủ sử dụng bộ phân lớp Softmax.

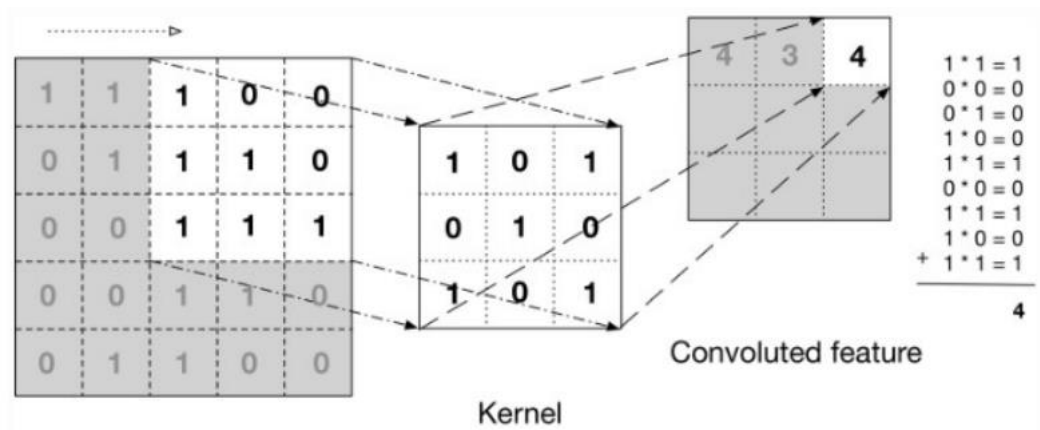


Hình 2.11 Cấu trúc một mạng CNN

2.2.2. Lớp Convolution

Convolution layer là lớp đầu tiên để trích đặc trưng từ ảnh đầu vào. Convolution duy trì mối quan hệ giữa các pixel bằng cách học các đặc trưng của bức hình sử dụng các kernel(filter).

Ví dụ: Để tính tích chập giữa ảnh đầu vào 5x5 và kernel 3x3 bằng cách trượt kernel trên ảnh đầu vào với stride = 1 và giá trị tích chập là kết quả của dot product giữa kernel và ảnh đầu vào như phía bên dưới.[4]



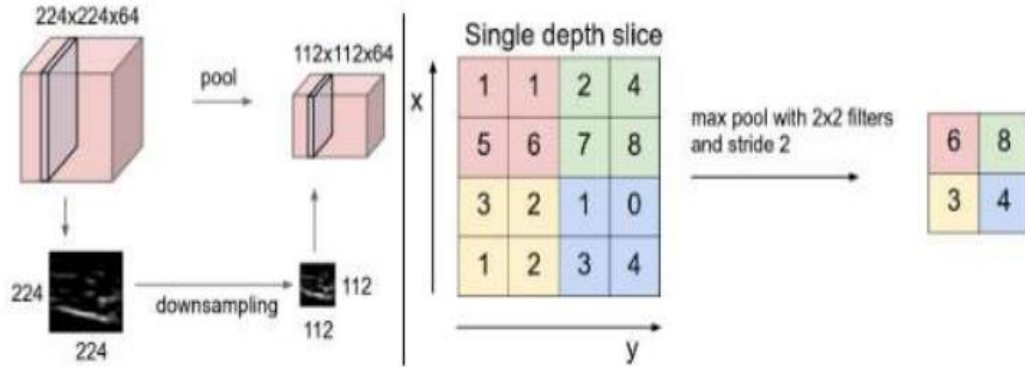
Hình 2.12 Tính tích chập giữa ảnh đầu vào 5x5 và Kernel 3x3

2.2.3. Lớp Pooling

Chức năng của lớp pooling là giảm dần kích thước không gian (chiều dài và chiều rộng) của khối đầu vào để giảm khối lượng tính toán trong mạng cũng như để chống lại hiện tượng Over-fitting.

Giống như hoạt động của lớp Convolution thì lớp Pooling cũng có một cửa sổ trượt, cửa sổ này trượt trên toàn bộ ma trận đầu vào và lấy giá trị lớn nhất từ các giá trị của ảnh và cửa sổ trượt qua (hình thức này gọi là Max Pooling).

Ví dụ về Max Pooling với stride = 2 và cửa sổ trượt với kích thước 2x2

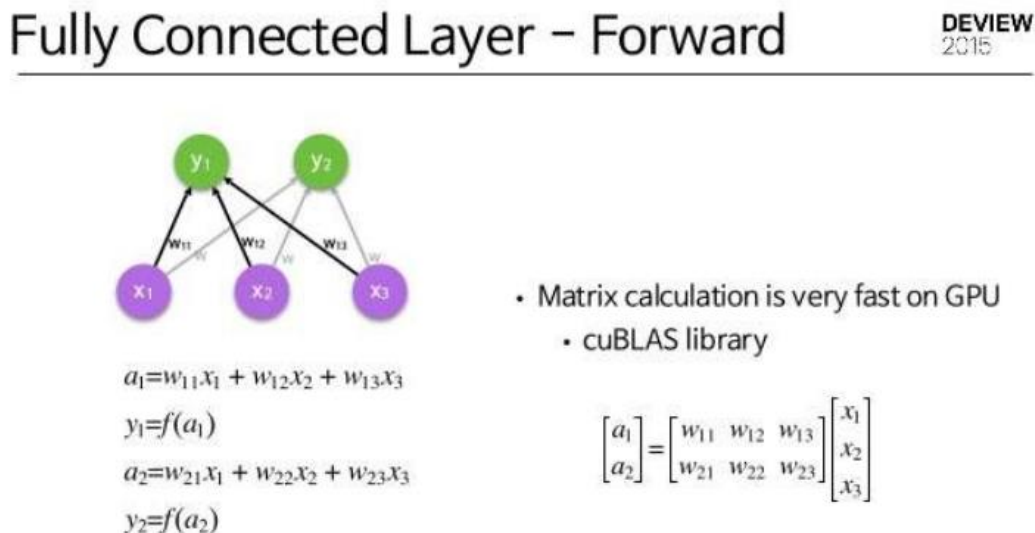


Hình 2.13 : Minh họa về Max Pooling

Lớp Pooling chỉ hoạt động trên từng lớp của khối ma trận Input đưa vào, không ảnh hưởng đến chiều sâu của khối ma trận Input.[4]

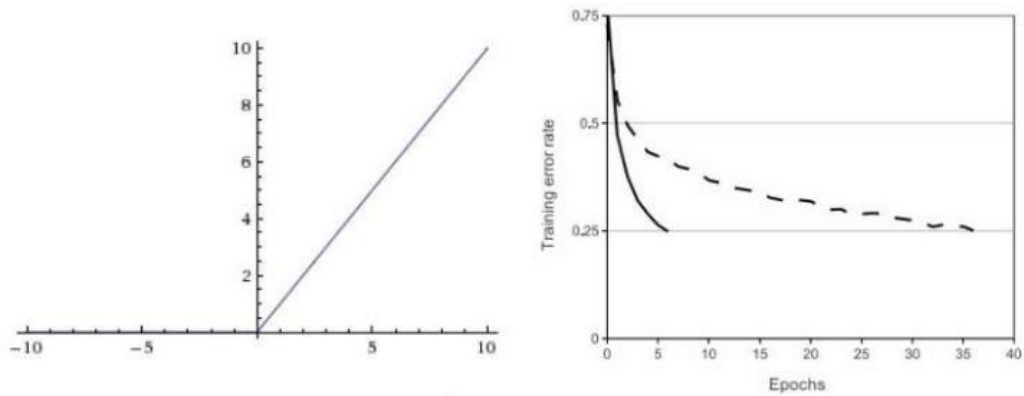
2.2.4. Lớp Fully Connected

Fully connected là lớp cuối cùng của mạng CNN. Đây là một lớp đầy đủ kết nối (fully connected layer). Lớp này nối mọi nơon từ lớp max pooled tới mọi nơon của tầng ra.[2]



Hình 2.14 Minh họa về lớp Fully Connected

2.2.5. Lớp phi tuyến ReLU (Rectified Linear Unit)



Hình 2.15 Hàm ReLU và tốc độ hội tụ khi so sánh với hàm tanh.

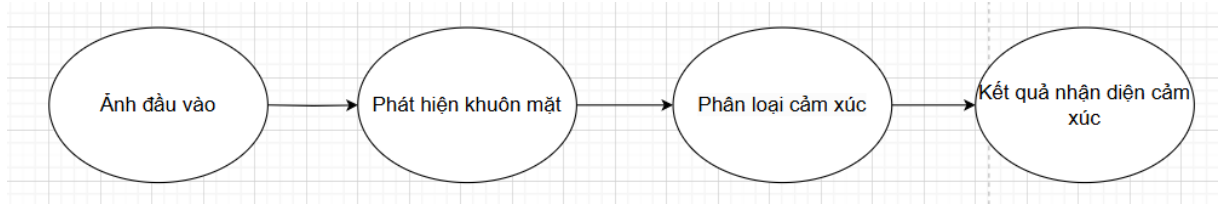
ReLU (Rectified Linear Unit) được sử dụng rộng rãi gần đây vì tính đơn giản của nó. Đồ thị của hàm ReLU được minh họa trên Hình 5 (trái)). Nó có công thức toán học $f(s) = \max(0, s)$ - rất đơn giản. Ưu điểm chính của nó là:

- ReLU được chứng minh giúp cho việc training các Deep Networks nhanh hơn rất nhiều (theo Krizhevsky et al.). Hình 5 (phải) so sánh sự hội tụ của SGD khi sử dụng hai activation function khác nhau: ReLU và tanh. Sự tăng tốc này được cho là vì ReLU được tính toán gần như tức thời và gradient của nó cũng được tính cực nhanh với gradient bằng 1 nếu đầu vào lớn hơn 0, bằng 0 nếu đầu vào nhỏ hơn 0.

- Mặc dù hàm ReLU không có đạo hàm tại $s = 0$, trong thực nghiệm, người ta vẫn thường định nghĩa $\text{ReLU}'(0) = 0$ và khẳng định thêm rằng, xác suất để input của một unit bằng 0 là rất nhỏ.[4]

Chương 3: GIẢI PHÁP NHẬN DẠNG CẢM XÚC TRÊN KHUÔN MẶT

3.1. Pipeline tổng quan



Hình 3.1 Giải pháp tổng quan của hệ thống

Hình ảnh trên là một sơ đồ khối mô tả quy trình nhận diện cảm xúc từ ảnh đầu vào. Quy trình này bao gồm các bước sau:

- Ảnh đầu vào: Đây là bước đầu tiên, nơi ảnh được nhập vào hệ thống.
- Phát hiện khuôn mặt: Ảnh đầu vào được xử lý để phát hiện các khuôn mặt trong ảnh.
- Phân loại cảm xúc: Sau khi khuôn mặt được phát hiện, hệ thống tiến hành phân loại cảm xúc dựa trên các đặc điểm khuôn mặt.
- Kết quả nhận diện cảm xúc: Cuối cùng, hệ thống đưa ra kết quả về cảm xúc được nhận diện từ ảnh đầu vào với nhãn và độ tin cậy dự đoán cho nhãn đó.

Sơ đồ này cho thấy một cách trực quan các bước cần thiết để thực hiện việc nhận diện cảm xúc từ ảnh.

3.2. Phát hiện khuôn mặt (Face Detection)

Bước đầu tiên không thể thiếu trong nhận dạng cảm xúc mặt người là phát hiện khuôn mặt. Nếu bước phát hiện khuôn mặt được thực hiện tốt thì sẽ làm tăng hiệu quả nhận dạng của hệ thống. Chúng tôi sử dụng thuật toán do Viola-Jones đề xuất với đặc trưng Haar để xác định vùng trong ảnh có khuôn mặt người. Thuật toán này cho kết quả khá cao và thời gian xử lý nhanh. Với dữ liệu khuôn mặt để nhận biết được cảm xúc thì tư thế khuôn mặt phải được chụp thẳng (frontface) và không có mang bất kỳ phụ kiện này che khuất các đặc trưng, đây là tiền đề để phương pháp của Viola-Jones cho kết quả rất cao và nhanh [11].

Chúng tôi sau khi phát hiện cho vùng khuôn mặt chúng tôi chuẩn hóa kích thước khuôn mặt để chuẩn bị cho giai đoạn kế tiếp. Để phân loại cảm xúc của khuôn mặt thì chúng tôi giả thiết không làm mất tính tổng quát là trong mỗi ảnh chỉ có tối đa một khuôn mặt.

3.2.1. Mô tả

- Module này được sử dụng để xác định các vùng khuôn mặt nằm trong ảnh đầu vào.

+ *Input*: ảnh đầu vào.

+ *Output*: tọa độ các vùng khuôn mặt đã được nhận dạng.

3.2.2. Giải pháp

- *Mô hình sử dụng*: sử dụng model pre-trained áp dụng thuật toán Haar cascade cho bài toán object detection.

- Sử dụng thư viện OpenCV cài đặt Haar Cascade để phát hiện các đối tượng như khuôn mặt.[11]

3.2.3. Kết quả thử nghiệm



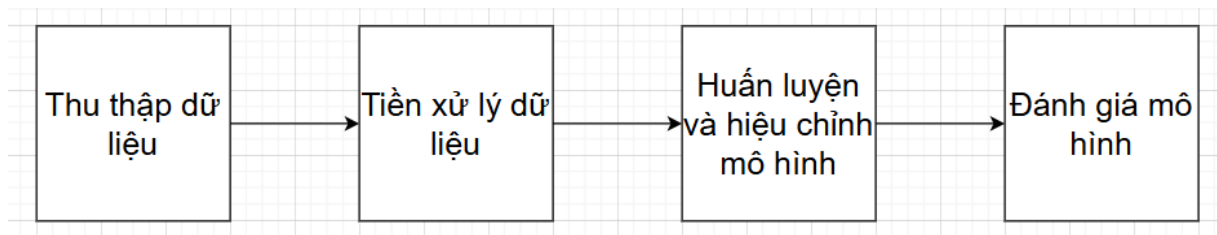
Hình 3.2 Kết quả thử nghiệm model FD

- Mô hình phát hiện khuôn mặt tốt với cả ảnh 1 người và ảnh nhiều người.
- Một số gương mặt không rõ ràng hoặc bị che khuất sẽ không phát hiện được

3.3. Phân loại cảm xúc (Emotion Classification)

Mô tả module

- Module này nhằm phân loại cảm xúc trên khuôn mặt thu được từ module trước
 - + *Input*: vùng khuôn mặt đã được crop từ **Module FD** đã được chuyển sang ảnh xám và có kích thước 48x48.
 - + *Output*: kết quả phân loại cảm xúc khuôn mặt



Hình 3.3 Quy trình xây dựng và kiểm thử mô hình phân loại cảm xúc dùng CNN

Pipeline trong hình gồm bốn bước chính trong quá trình phát triển mô hình học máy. Dưới đây là mô tả chi tiết về từng bước:

Thu thập dữ liệu:

- Đây là bước đầu tiên và quan trọng nhất trong quy trình phát triển mô hình học máy. Dữ liệu được thu thập từ nhiều nguồn khác nhau như cơ sở dữ liệu, web scraping, khảo sát, hoặc các nguồn dữ liệu công khai.
- Dữ liệu cần phải đủ lớn và đa dạng để đảm bảo mô hình có thể học một cách chính xác.

Tiền xử lý dữ liệu:

- Sau khi thu thập, dữ liệu thường không hoàn hảo và cần phải được làm sạch. Các bước tiền xử lý bao gồm xử lý giá trị thiếu, loại bỏ dữ liệu ngoại lai, chuẩn hóa và biến đổi dữ liệu (như chuyển đổi định dạng, mã hóa các biến phân loại).
- Mục tiêu của bước này là tạo ra một bộ dữ liệu sạch và có cấu trúc phù hợp cho quá trình huấn luyện mô hình.

Huấn luyện và hiệu chỉnh mô hình:

- Trong bước này, dữ liệu đã tiền xử lý được sử dụng để huấn luyện mô hình học máy. Các thuật toán học máy được áp dụng để tìm ra các mẫu và mối quan hệ trong dữ liệu.
- Quá trình hiệu chỉnh cũng được thực hiện để điều chỉnh các siêu tham số của mô hình nhằm tối ưu hóa hiệu suất của nó.

Đánh giá mô hình:

- Sau khi mô hình được huấn luyện, nó cần phải được kiểm thử và đánh giá để xác định hiệu suất của nó trên các bộ dữ liệu chưa từng thấy trước đây (bộ dữ liệu kiểm thử).
- Các chỉ số đánh giá như độ chính xác, độ nhạy, độ đặc hiệu,... được sử dụng để đánh giá chất lượng của mô hình.
- Nếu mô hình đạt yêu cầu, nó sẽ được triển khai; nếu không, quá trình huấn luyện và hiệu chỉnh sẽ được thực hiện lại.

3.3.1. Thu thập dữ liệu

3.3.1.1. Nguồn dữ liệu

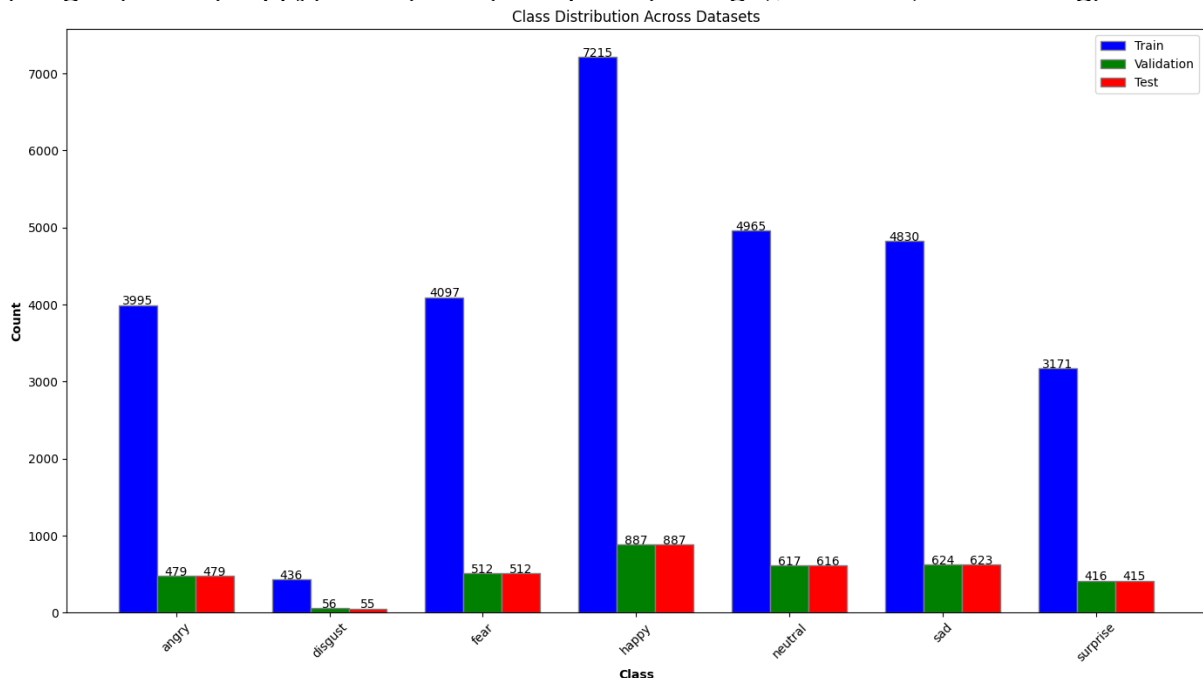
- Tên tập dữ liệu: FER-2013
- Nguồn: [FER-2013 \(kaggle.com\)](https://www.kaggle.com/c/fer2013)
- Số lượng: 35887 mẫu

3.3.1.2. Phân chia dữ liệu

- Phân chia bộ dữ liệu ban đầu thành 3 tập train – val – test theo tỉ lệ 80 – 10 – 10:

- + Train: 28709 mẫu
- + Validation: 3591 mẫu
- + Test: 3587 mẫu

- Gán nhãn từ 0 đến 6 tương ứng với các cảm xúc: Giận (Angry), Ghê tởm (Disgust), Vui (Happy), Sad (Buồn), Surprise (Bất ngờ), Neutral (Bình thường)



Hình 3.4 Phân bố tập dữ liệu huấn luyện

3.3.2. Tiền xử lý dữ liệu

3.3.2.1. Thay đổi kích thước ảnh (Resizing)

Để đảm bảo tất cả các ảnh có cùng kích thước đầu vào cho mạng neural, tất cả các ảnh cần được thay đổi kích thước về cùng một kích thước cố định 48x48 pixel.



Hình 3.5 Thay đổi kích thước ảnh

3.3.2.2. Chuyển đổi ảnh về thang màu xám (Grayscale Conversion):

Trong bài toán nhận diện cảm xúc, thông tin về màu sắc thường không cần thiết. Do đó, chuyển đổi ảnh về thang màu xám giúp giảm kích thước dữ liệu và giảm tải tính toán.



Hình 3.6 Chuyển ảnh xám

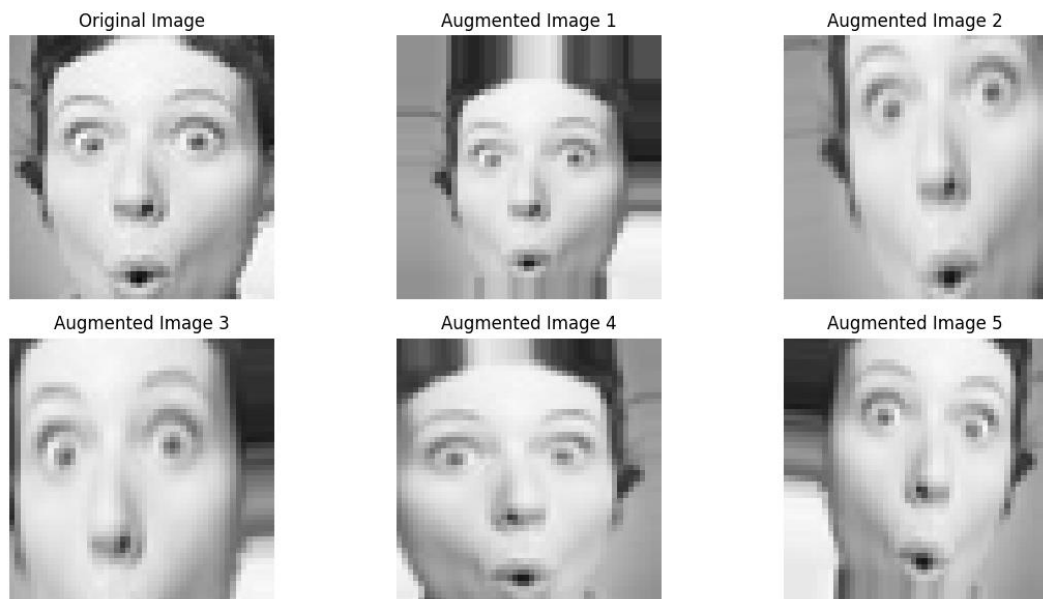
3.3.2.3. Chuẩn hóa giá trị pixel (Pixel Normalization):

Chia tất cả các giá trị pixel của ảnh cho 255 để đưa các giá trị này về khoảng [0, 1]. Đây là bước chuẩn hóa phổ biến trong xử lý ảnh và giúp tăng tốc độ hội tụ của mô hình.

3.3.2.4. Tăng cường dữ liệu cho tập train

- zoom_range=0.3: Tạo ra các ảnh phóng to hoặc thu nhỏ với một khoảng ngẫu nhiên lên tới 30%.

- horizontal_flip=True: Lật ngược ảnh theo chiều ngang ngẫu nhiên.
- rotation_range=10: Xoay ảnh ngẫu nhiên trong khoảng ± 10 độ.
- width_shift_range=0.1: Dịch chuyển ảnh theo chiều ngang ngẫu nhiên trong khoảng $\pm 10\%$ của chiều rộng ảnh.
- height_shift_range=0.1: Dịch chuyển ảnh theo chiều dọc ngẫu nhiên trong khoảng $\pm 10\%$ của chiều cao ảnh.
- shear_range=0.2: Áp dụng biến đổi cắt xén (shear transformation) lên tới 20%



Hình 3.7 Kết quả tăng cường dữ liệu

- Từ 1 ảnh gốc ban đầu có thể tạo ra nhiều ảnh tăng cường từ nhiều bộ phương pháp tăng cường khác nhau.

3.3.3. Huấn luyện và hiệu chỉnh mô hình

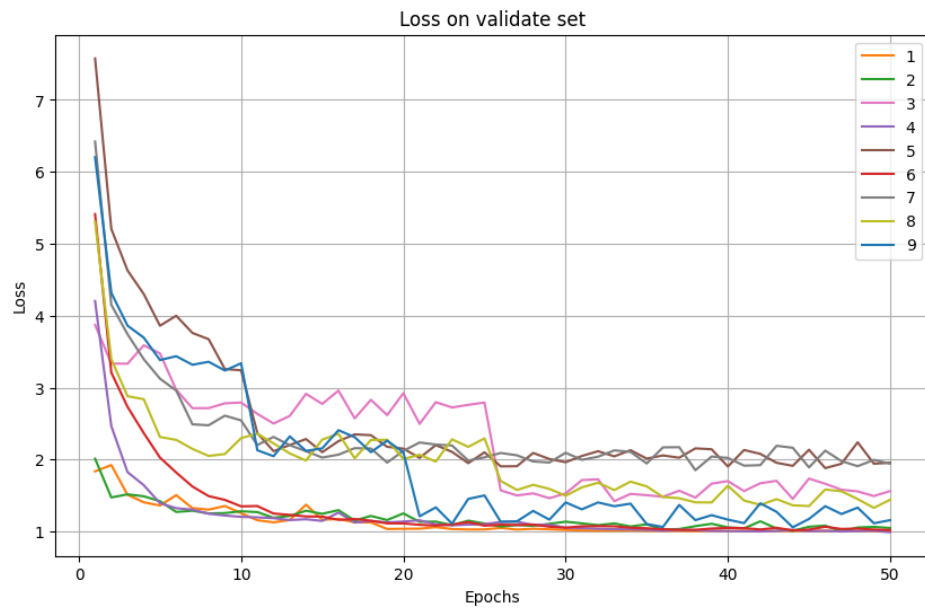
- Sau khi tham khảo, tìm hiểu về các cấu trúc mô hình CNN [4][9][10] tôi tiến hành lựa chọn và so sánh giữa 9 tổ hợp lớp ẩn trong mô hình CNN để tìm ra tổ hợp cho độ chính xác cao nhất trên tập validation
- Cả 9 tổ hợp đều có kiến trúc kết thúc như sau:
 - + Tầng Flatten chuyển đổi dữ liệu thành vector.
 - + Tầng Dense và hàm kích hoạt ReLU, sau đó là Dropout với tỷ lệ 0.5.
 - + Cuối cùng là tầng Dense với số neuron bằng số lượng lớp (07) và hàm kích hoạt softmax để thực hiện phân loại cảm xúc.
- Quá trình so sánh được thực hiện với các siêu tham số được set cố định cho cả 9 mô hình với giá trị cụ thể sau:

- + Epochs = 50
 - + Learning rate = 1e-4
 - + Batch size = 64
 - + Optimizer = ‘Adam’,
 - + Loss function = “categorical_crossentropy”
 - + Thực hiện giảm Learning rate nếu trong vòng 3 epoch không có sự cải thiện về val_accuracy.
 - + Thực hiện dừng sớm nếu trong vòng 6 epoch không có sự cải thiện về val_accuracy.
- Chi tiết mô tả tổ hợp và kết quả trên tập validation được trình bày dưới bảng sau:

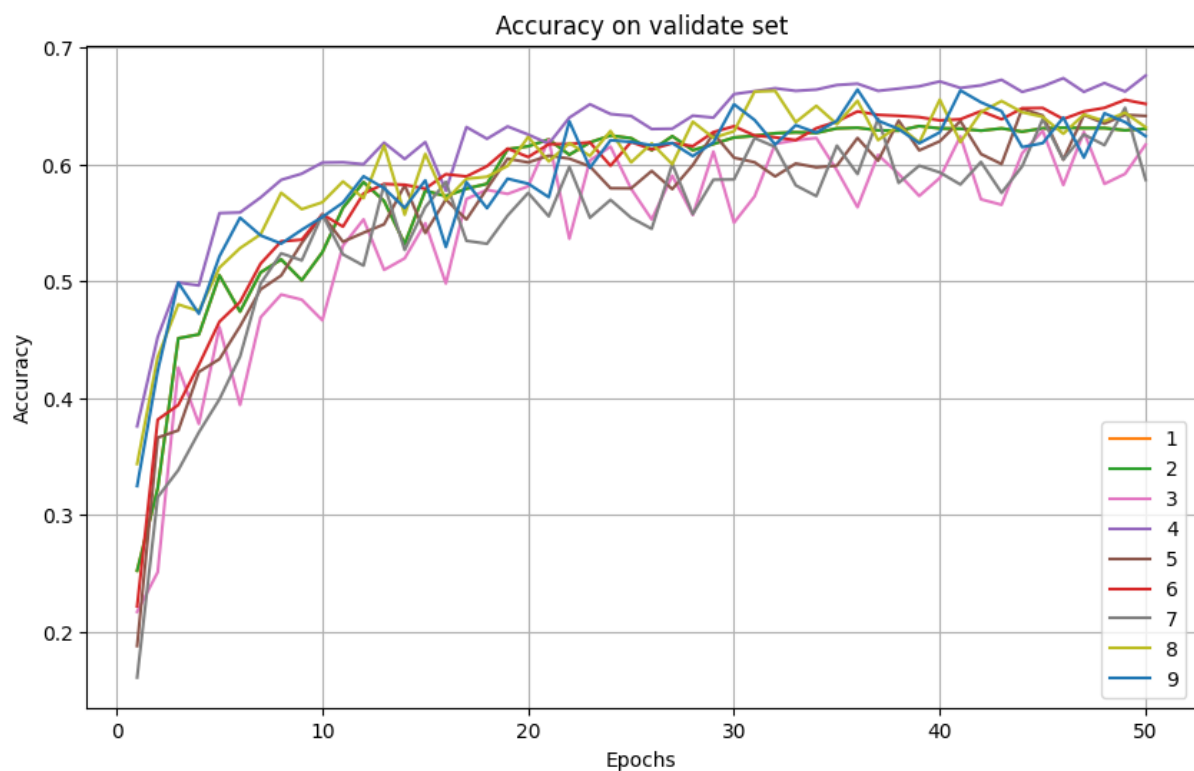
Bảng 3.1 Mô tả cấu trúc và kết quả huấn luyện của từng tổ hợp trên tập validation

STT	Mô tả cấu trúc	Val_accuracy	Val_loss
1	- 3 lớp Convolution với lần lượt 32, 64, 128 bộ lọc kích thước 3x3	0.63	1.36
2	- 3 lớp Convolution với lần lượt 32, 64, 128 bộ lọc kích thước 3x3 - Xen kẽ là 1 bộ gồm 3 lớp : + Batch Normalization và + MaxPooling 2x2, + Dropout(0.25)	0.60	1.85
3	- 3 lớp Convolution với lần lượt 32, 64, 128 bộ lọc kích thước 3x3 - Xen kẽ là 2 bộ gồm 3 lớp : + Batch Normalization và + MaxPooling 2x2, + Dropout(0.25)	0.57	1.35
4	- 4 lớp Convolution với lần lượt 32, 64, 128, 256 bộ lọc kích thước 3x3	0.63	1.21
5	- 4 lớp Convolution với lần lượt 32, 64, 128, 256 bộ lọc kích thước 3x3 - Xen kẽ là 1 bộ gồm 3 lớp : + Batch Normalization và	0.63	1.98

	+ MaxPooling 2x2, + Dropout(0.25)		
6	- 4 lớp Convolution với lần lượt 32, 64, 128, 256 bộ lọc kích thước 3x3 - Xen kẽ là 2 bộ gồm 3 lớp : + Batch Normalization và + MaxPooling 2x2, + Dropout(0.25)	0.67	0.98
7	- 5 lớp Convolution với lần lượt 32, 64, 128, 256, 512 bộ lọc kích thước 3x3	0.63	1.61
8	- 4 lớp Convolution với lần lượt 32, 64, 128, 256 bộ lọc kích thước 3x3 - Xen kẽ là 1 bộ gồm 3 lớp : + Batch Normalization và + MaxPooling 2x2, + Dropout(0.25)	0.64	1.02
9	- 4 lớp Convolution với lần lượt 32, 64, 128, 256 bộ lọc kích thước 3x3 - Xen kẽ là 1 bộ gồm 3 lớp : + Batch Normalization và + MaxPooling 2x2, + Dropout(0.25)	0.65	1.00



Hình 3.8 Loss của 9 tổ hợp trên tập validation



Hình 3.9 Accuracy của 9 tổ hợp trên tập Validation

Từ bảng trên ta chọn ra được tổ hợp số 6 gồm:

- 4 lớp Convolution với lần lượt 32, 64, 128, 256 bộ lọc kích thước 3x3
- Xen kẽ là 2 bộ gồm 3 lớp :
 - + Batch Normalization và
 - + MaxPooling 2x2,

+ Dropout(0.25)

Cho kết quả tốt nhất trên tập validation với accuracy 0.67 và loss 0.98. Do đó ta sẽ sử dụng cấu trúc mô hình này.

Cấu trúc mô hình cho độ chính xác tốt nhất :

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 48, 48, 32)	320
conv2d_1 (Conv2D)	(None, 48, 48, 64)	18496
batch_normalization (Batch Normalization)	(None, 48, 48, 64)	256
max_pooling2d (MaxPooling2D)	(None, 24, 24, 64)	0
dropout (Dropout)	(None, 24, 24, 64)	0
conv2d_2 (Conv2D)	(None, 24, 24, 128)	73856
conv2d_3 (Conv2D)	(None, 22, 22, 256)	295168
batch_normalization_1 (Batch Normalization)	(None, 22, 22, 256)	1024
max_pooling2d_1 (MaxPooling2D)	(None, 11, 11, 256)	0
dropout_1 (Dropout)	(None, 11, 11, 256)	0
flatten (Flatten)	(None, 30976)	0
dense (Dense)	(None, 1024)	31720448
dropout_2 (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 7)	7175
Total params: 32116743 (122.52 MB)		
Trainable params: 32116103 (122.51 MB)		
Non-trainable params: 640 (2.50 KB)		

Hình 3.10 Cấu trúc mô hình

Mô hình bao gồm các tầng như sau:

- Tầng tích chập đầu tiên với 32 bộ lọc kích thước 3x3, sử dụng hàm kích hoạt ReLU và padding 'same'.

- Tầng tích chập thứ hai với 64 bộ lọc kích thước 3x3, sử dụng hàm kích hoạt ReLU và padding 'same'.
- Tiếp theo là tầng Batch Normalization và MaxPooling với kích thước 2x2, sau đó là Dropout với tỷ lệ 0.25.
- Tầng tích chập thứ ba với 128 bộ lọc kích thước 3x3, sử dụng hàm kích hoạt ReLU, padding 'same' và regularizer l2 với hệ số 0.01.
- Tầng tích chập thứ tư với 256 bộ lọc kích thước 3x3, sử dụng hàm kích hoạt ReLU và regularizer l2 với hệ số 0.01.
- Tiếp theo là tầng Batch Normalization và MaxPooling với kích thước 2x2, sau đó là Dropout với tỷ lệ 0.25.
- Tầng Flatten chuyển đổi dữ liệu thành vector.
- Tầng Dense với 1024 neuron và hàm kích hoạt ReLU, sau đó là Dropout với tỷ lệ 0.5.
- Cuối cùng là tầng Dense với số neuron bằng số lượng lớp (07) và hàm kích hoạt softmax để thực hiện phân loại cảm xúc.
- Nhằm giảm tình trạng over-fitting của kiến trúc CNN, kỹ thuật Dropout được sử dụng sau các tầng pooling và tầng fully-connected.

Huấn luyện mô hình có kết quả tốt nhất

Bảng 3.2 Các siêu tham số huấn luyện cho mô hình trên tập train

Tham số huấn luyện	Giá trị
Epochs	150
Learning rate	10^{-4}
Batch size	64
Optimization	Adam
Loss function	categorical crossentropy

Learning rate schedule:

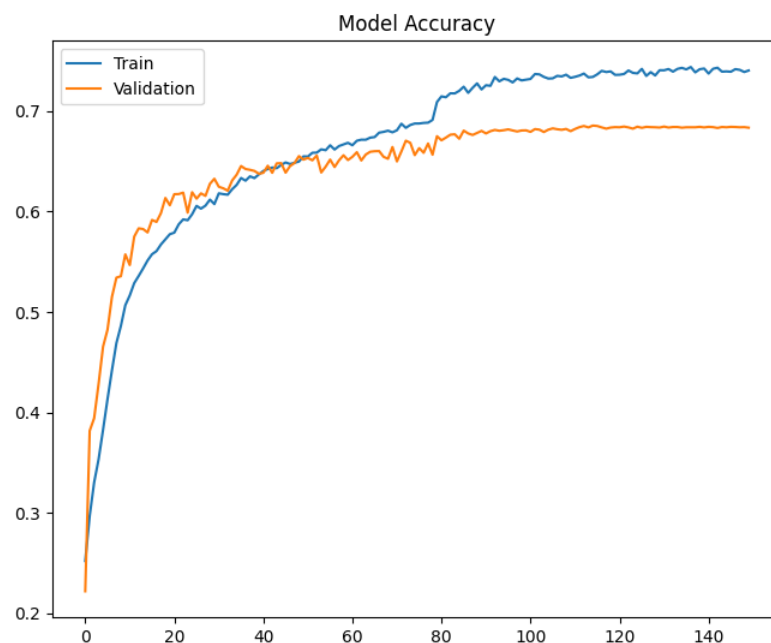
Điều chỉnh learning rate trong quá trình huấn luyện, chẳng hạn như giảm learning rate khi tiến trình huấn luyện không còn cải thiện đáng kể.

- `monitor='val_accuracy'`: Theo dõi độ chính xác trên tập validation.
- `factor=0.2`: Hệ số giảm tốc độ học là 0.2 khi được kích hoạt.
- `patience=6`: Số lượng epoch mà mô hình có thể không cải thiện trước khi giảm tốc độ học.
- `verbose=1`: Hiển thị thông tin khi giảm tốc độ học.
- `min_delta=0.0001`: Độ thay đổi tối thiểu để xem xét là cải thiện

Early stopping criteria:

Các tiêu chí để dừng quá trình huấn luyện sớm nếu hiệu suất trên tập validation không còn cải thiện, giúp tránh overfitting.

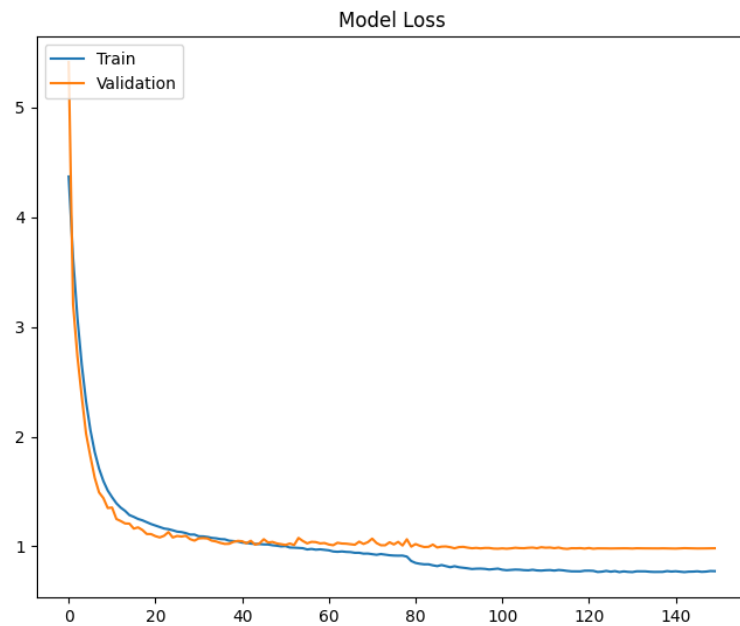
- `monitor='val_accuracy'`: Theo dõi độ chính xác trên tập validation.
- `min_delta=0.001`: Độ thay đổi tối thiểu mà mô hình phải có để không coi là cải thiện.
- `patience=3`: Số lượng epoch mà mô hình có thể không cải thiện trước khi dừng huấn luyện.
- `verbose=1`: Hiển thị thông tin khi dừng huấn luyện.
- `restore_best_weights=True`: Khôi phục lại các trọng số của mô hình về phiên bản tốt nhất.



final train accuracy = 79.59 , validation accuracy = 68.38

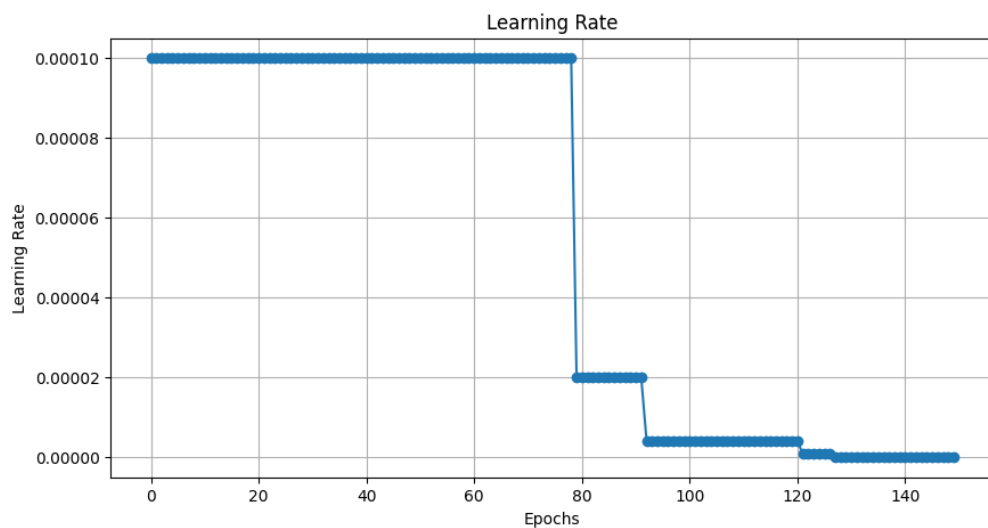
Hình 3.11 Accuracy trên tập train và validation

- Accuracy của mô hình trên tập val là 68.38% và trên tập train là 79.58%



Hình 3.12 Loss trên tập train và validation

- Loss trên tập trai là 0,98 và trên tập train là 0,68
- Kết quả cho thấy mô hình hội tụ tốt và không bị overfitting

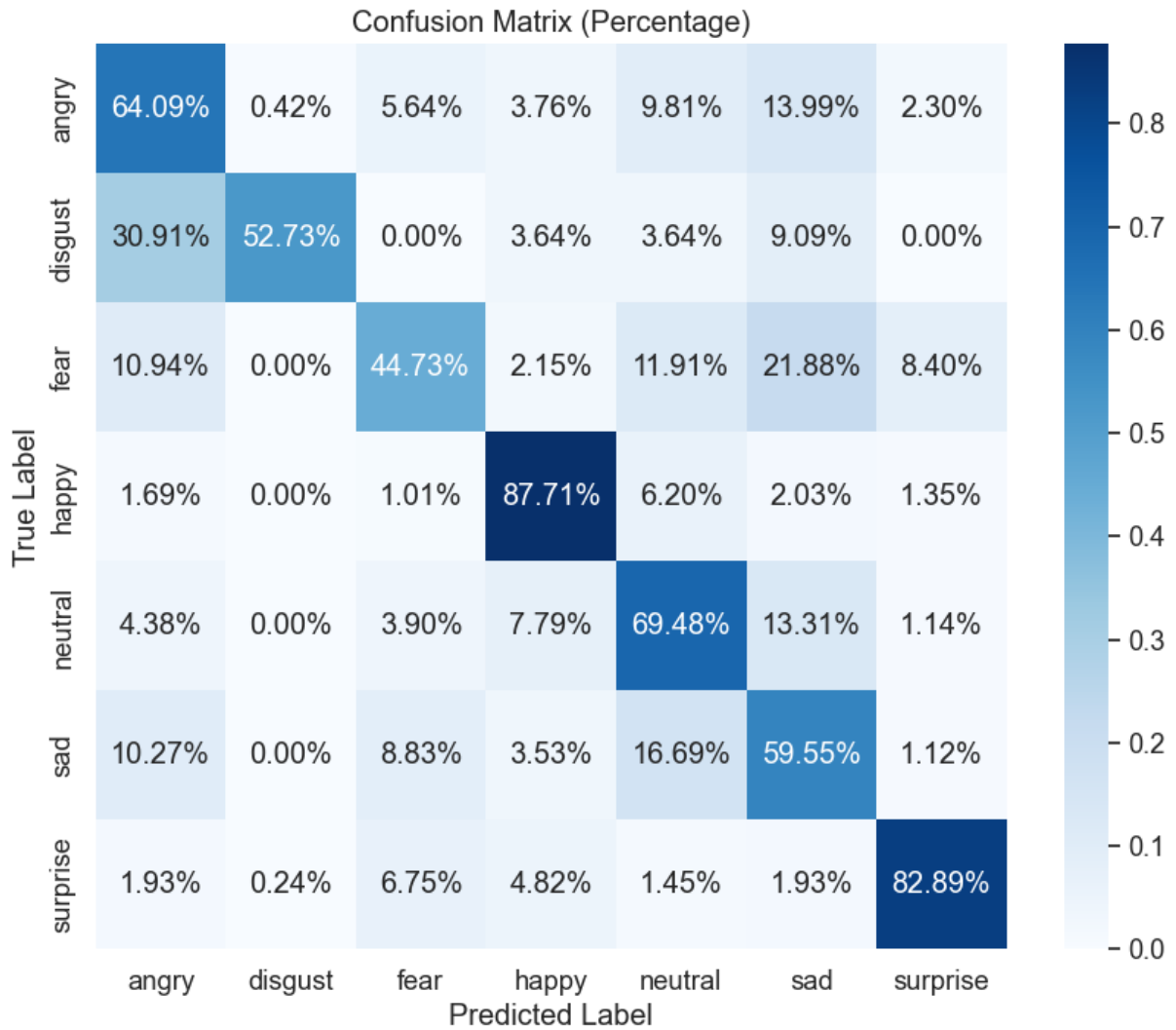


Hình 3.13 Biến thiên learning rate khi huấn luyện

- Learning rate có thay đổi trong quá trình huấn luyện và được biểu diễn trên hình.

3.3.4. Đánh giá mô hình

Việc đánh giá mô hình được thực hiện bằng cách sử dụng 3591 hình ảnh trên tập test. Trình phân loại cho độ chính xác 69,31%. Ma trận nhầm lẫn cho bảy lớp biểu hiện khuôn mặt được hiển thị bên dưới:



Hình 3.14 Ma trận nhầm lẫn cho 7 loại cảm xúc

Các bảng trên cho thấy mô hình này có độ chính xác cao nhất đối với cảm xúc vui với 87.71%, tiếp theo là ngạc nhiên với 82,89 %, trung lập với 69.48%, tức giận với 64.09%, buồn với 59.55%, ghê tởm với 52.73% và độ chính xác thấp nhất đối với cảm xúc sợ hãi là 44.73%.

Precision, recall and F1-score của từng lớp biểu cảm khuôn mặt được hiển thị bên dưới:

Class	Precision	Recall	F1-score
angry	0.62	0.64	0.63
disgust	0.91	0.53	0.67
fear	0.62	0.45	0.52
happy	0.87	0.88	0.87
neutral	0.61	0.69	0.65
sad	0.56	0.60	0.58
surprise	0.81	0.83	0.82
accuracy			0.69
macro avg	0.71	0.66	0.68
weighted avg	0.69	0.69	0.69

Hình 3.15 Metris cho 7 lớp cảm xúc

Accuracy tổng thể của mô hình là 69%. Mô hình hoạt động thực sự tốt về việc phân loại những cảm xúc tích cực dẫn đến điểm số chính xác tương đối cao về hạnh phúc và ngạc nhiên.

Nhìn chung, hiệu suất của mô hình có vẻ yếu hơn đối với những cảm xúc tiêu cực. Đặc biệt, cảm xúc sợ hãi và buồn có precision và f1-score thấp. Mô hình thường xuyên phân loại sai tức giận, sợ hãi và trung tính là buồn bã. Ngoài ra, nó bối rối nhất khi dự đoán khuôn mặt buồn và trung tính vì hai cảm xúc này có lẽ ít biểu cảm nhất (không bao gồm khuôn mặt đang khóc).

Tổng F1-score cũng là 0,69. Điểm F1 cao nhất cho hạnh phúc và bất ngờ có F1-score lần lượt là 0,87 và 0,82. Sự sợ hãi có F1-score thấp nhất là 0,52.

Đánh giá tốc độ xử lý:

- Thực hiện dự đoán 1000 mẫu ngẫu nhiên trên tập test 5 lần cho kết quả như sau:

Bảng 3. 3 Đánh giá tốc độ thực thi của mô hình

Lần	Thời gian thực thi (s)
1	2.85
2	2.58
3	2.67
4	2.64
5	3.15

- Ta thu được kết quả trung bình xử lý 1000 mẫu mất 2.78s

Chương 4 : XÂY DỰNG ỨNG DỤNG

4.1. Môi trường phát triển

- Python

Python là một ngôn ngữ lập trình bậc cao, được tạo ra bởi Guido van Rossum và phát hành lần đầu tiên vào năm 1991. Python nổi tiếng với cú pháp rõ ràng, dễ đọc, và dễ học, làm cho nó trở thành một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới.

Đặc điểm nổi bật:

- Cú pháp dễ hiểu: Python có cú pháp gần gũi với ngôn ngữ tự nhiên, giúp lập trình viên viết mã một cách nhanh chóng và hiệu quả.
- Đa năng: Python được sử dụng rộng rãi trong nhiều lĩnh vực như phát triển web, khoa học dữ liệu, trí tuệ nhân tạo, tự động hóa, và nhiều lĩnh vực khác.
- Thư viện phong phú: Python có một hệ sinh thái phong phú với hàng ngàn thư viện mạnh mẽ như NumPy, pandas, TensorFlow, và nhiều thư viện khác hỗ trợ nhiều lĩnh vực.

- Flask

Flask là một microframework dành cho phát triển web được viết bằng Python. Flask được thiết kế để dễ sử dụng và mở rộng, cho phép các lập trình viên phát triển các ứng dụng web một cách nhanh chóng và dễ dàng.

Đặc điểm nổi bật:

- Nhẹ và Linh hoạt: Flask là một microframework, nghĩa là nó cung cấp những tính năng cơ bản nhất cho phát triển web và cho phép lập trình viên tùy biến và mở rộng theo nhu cầu.
- Dễ học và sử dụng: Với cú pháp đơn giản và cấu trúc rõ ràng, Flask rất dễ học và sử dụng, đặc biệt là cho những người mới bắt đầu.
- Khả năng mở rộng: Flask hỗ trợ mở rộng dễ dàng thông qua các extension. Các extension phổ biến bao gồm Flask-SQLAlchemy, Flask-WTF, Flask-Login, và nhiều cái khác.

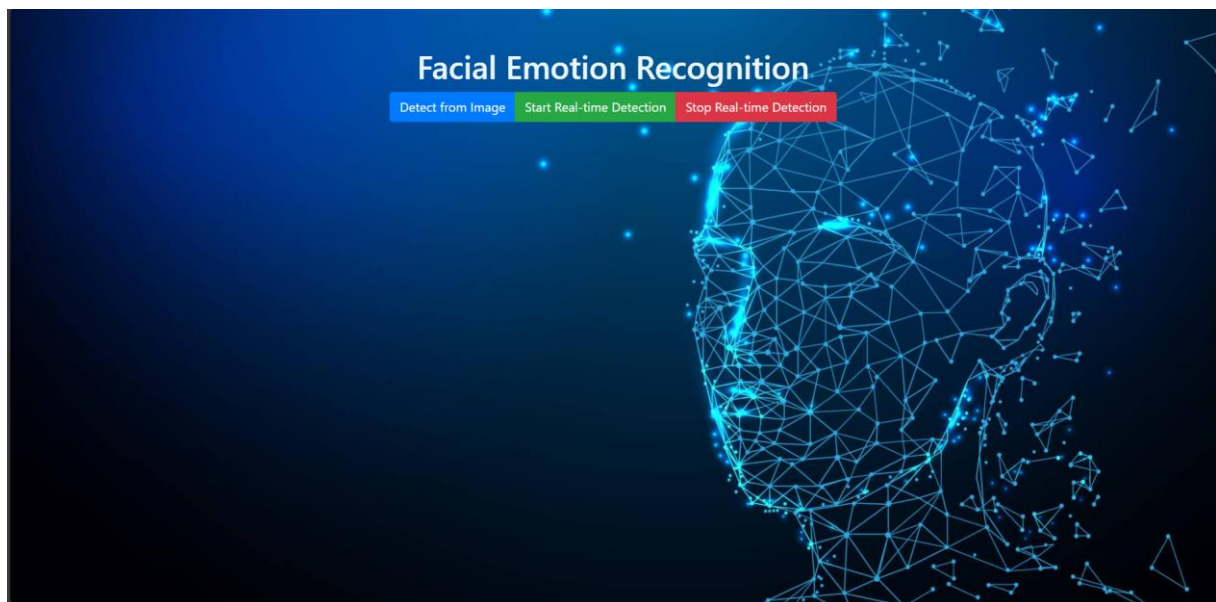
4.2. Xây dựng API

Việc xây dựng API backend với Flask là một bước quan trọng trong việc phát triển ứng dụng web nhận diện cảm xúc từ hình ảnh khuôn mặt. Dưới đây là mô tả chi tiết API này.

- Tên: API nhận diện cảm xúc
- Chức năng: Nhận ảnh đầu vào và trả về kết quả dự đoán cảm xúc và hình ảnh đã xử lý.
- Đầu vào: Ảnh từ người dùng được chuyển về dưới dạng Json
- Đầu ra: Hình ảnh đã xử lý với nhãn dự đoán và tập kết quả dự đoán độ tin cậy cho từng nhãn cảm xúc
- Base URL: https://your-domain.com/facial_emotion_recognition_realtime
- Phương thức HTTP: POST
- Cách thức hoạt động:
 - + Sử dụng Module Phát hiện khuôn mặt để vẽ vùng có khuôn mặt lên ảnh
 - + Sử dụng module Phân loại cảm xúc để dự đoán cảm xúc cho khuôn mặt được phát hiện và trả về kết quả

4.3. Thiết kế giao diện

Giao diện người dùng được xây dựng bằng html và javascript để xử lý các sự kiện gọi api, tải dữ liệu kết quả lên trang html



Hình 4.1 Giao diện màn hình chính

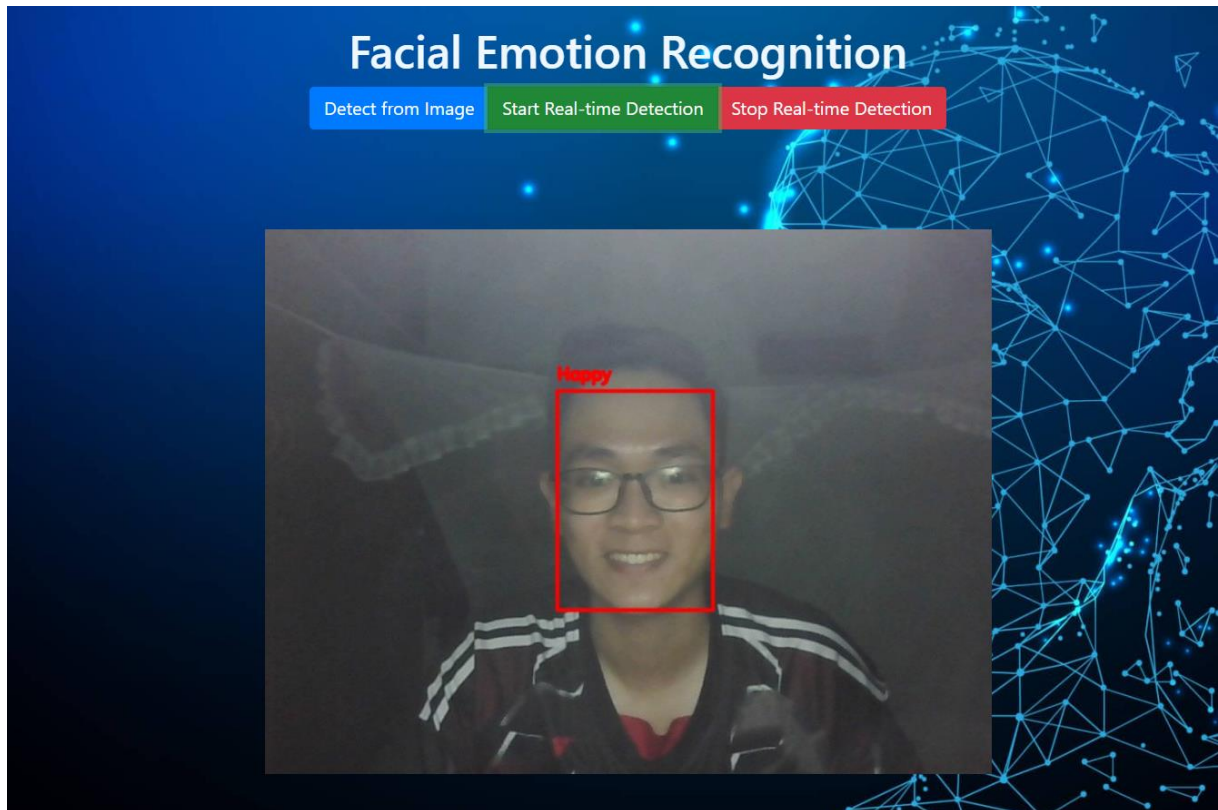
Bảng 4.1 Giải thích các chức năng trên giao diện

Action Name	Description
Detect from image	Khi người dùng nhấn vào nút, hệ thống sẽ yêu cầu người dùng nhập ảnh vào để hệ thống tiến hành nhận diện và đưa ra kết quả.
Start Realtime Detection	Khi người dùng nhấn vào nút, hệ thống sẽ yêu cầu người dùng chấp nhận để hệ thống sử dụng camera và tiến hành hiển thị kết quả thu được đã xử lý cho người dùng
Stop Realtime Detection	Tắt Realtime Detection



Hình 4.2 Nhận diện bằng ảnh

- Hình ảnh thử nghiệm ứng dụng với chức năng nhận diện cảm xúc từ ảnh.



Hình 4.3 Nhận diện realtime

- Hình ảnh thử nghiệm ứng dụng với chức năng nhận diện cảm xúc real time thông qua camera thiết bị.

Chương 5 : KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết Luận

Kết quả đạt được:

- Huấn luyện và hiệu chỉnh mô hình CNN để thu được mô hình phân loại cảm xúc từ hình ảnh với:

- Độ chính xác chung là 69,31%
- Độ chính xác cao nhất đối với cảm xúc vui với 87.71%, tiếp theo là ngạc nhiên với 82,89 %, trung lập với 69.48%, tức giận với 64.09%, buồn với 59.55%, ghê tởm với 52.73% và độ chính xác thấp nhất đối với cảm xúc sợ hãi là 44.73%.

- Thời gian thực thi 1000 mẫu trung bình là 2.78 giây

- Tích hợp mô hình để xây dựng API backend với Flask

- Xây dựng được ứng dụng web với giao diện thân thiện, dễ sử dụng, giúp người dùng có thể thử nghiệm một cách nhanh chóng và hiệu quả.

Hạn chế:

- Bộ dữ liệu vẫn còn khá ít, nhiều hạn chế nên mô hình vẫn chưa thật sự tổng quát tốt.
- Khả năng nhận diện cảm xúc khuôn mặt chưa hoàn hảo.
- Sự phụ thuộc vào độ phân giải hình ảnh.
- Khả năng tổng quát hóa chưa cao
- Mô hình yêu cầu tính toán cao.

5.2. Hướng phát triển

- Cải thiện độ chính xác:

- Mô hình lai ghép (Hybrid Models): Kết hợp các phương pháp khác nhau để tăng độ chính xác.

- Đa cảm xúc và cảm xúc phức tạp:

- Phát triển các mô hình có khả năng nhận diện các cảm xúc đa chiều và phức tạp hơn, không chỉ dừng lại ở các cảm xúc cơ bản.

- Ứng dụng thực tiễn:

- Chăm sóc sức khỏe: Giám sát cảm xúc của bệnh nhân, hỗ trợ tâm lý.
- Giáo dục: Phân tích phản ứng cảm xúc của học sinh để cải thiện phương pháp giảng dạy.
- Tương tác người-máy: Nâng cao trải nghiệm người dùng trong các hệ thống AI.

- Cải thiện về dữ liệu:

- Thu thập dữ liệu phong phú hơn: Bao gồm các biểu hiện cảm xúc trong nhiều ngữ cảnh và văn hóa khác nhau.
- Dữ liệu thời gian thực: Sử dụng các hệ thống thu thập dữ liệu trực tiếp từ các tình huống thực tế.

TÀI LIỆU THAM KHẢO

- [1] Ekman, Paul; Cordaro, Daniel (ngày 20 tháng 9 năm 2011). "What is Meant by Calling Emotions Basic". *Emotion Review*. **3** (4): 364–370. doi:[10.1177/1754073911410740](https://doi.org/10.1177/1754073911410740). ISSN 1754-0739.
- [2] Li, Shan, and Weihong Deng. "Deep facial expression recognition: A survey." *IEEE Transactions on Affective Computing* (2020).
- [3] L. Pham, T. H. Vu and T. A. Tran, "Facial Expression Recognition Using Residual Masking Network," *2020 25th International Conference on Pattern Recognition (ICPR)*, Milan, Italy, 2021, pp. 4513-4519, doi: 10.1109/ICPR48806.2021.9411919.
- [4] Meghana, Mohammad Yusuf Khan , Mahesh Bharti, Kalpesh Mohanta, Kanaiya "Facial Emotion Recognition using CNN", *Journal of Advanced Zoology* pp 966-970, 2023.
- [5] Huang, ZY., Chiang, CC., Chen, JH. *et al.* A study on computer vision for facial emotion recognition. *Sci Rep* 13, 8425 (2023). <https://doi.org/10.1038/s41598-023-35446-4>
- [6] I. Michael Revina, W.R. Sam Emmanuel "A Survey on Human Face Expression Recognition Techniques", 2018
- [7] Y. Wen, K. Zhang, Z. Li và Y. Qiao, "A Discriminative Feature Learning Approach for Deep Face Recognition," *ECCV 2016: Computer Vision – ECCV 2017*, tập 9911, pp. 499-515, 2017.
- [8] Md. Forhad Ali, Mehenag Khatun, Nakib Aman Turzo, "Facial Emotion Detection Using Neural Network", *International Journal of Scientific & Engineering Research* Volume 11 ISSN 2229-5518, 2020.
- [9] Harisu Abdullahi Shehu, Md. Haidar Sharif and Sahin Uyaver, "Facial Expression Recognition Using Deep Learning", *AIP Conference Proceedings* 2334, 070003 (2021), 02 March 2021
- [10] Nithya Roopa. S, "Emotion Recognition From Facial Expression Using Deep Learning", Vol.8, pp.91-95, August 2019.
- [11] Thuật toán Haar Cascade: [Lập trình công cụ phát hiện khuôn mặt bằng Python - QuanTriMang.com](https://quantrimang.com) [Accessed 12 June 2024]
- [12] Trang, Lê Thu. "Sử Dụng Phương Pháp Support Vector Machine Và Convolutional Neural Network Để Phân Loại Cảm Xúc Của Khuôn Mặt." *TNU Journal of Science and Technology*, 2021.

