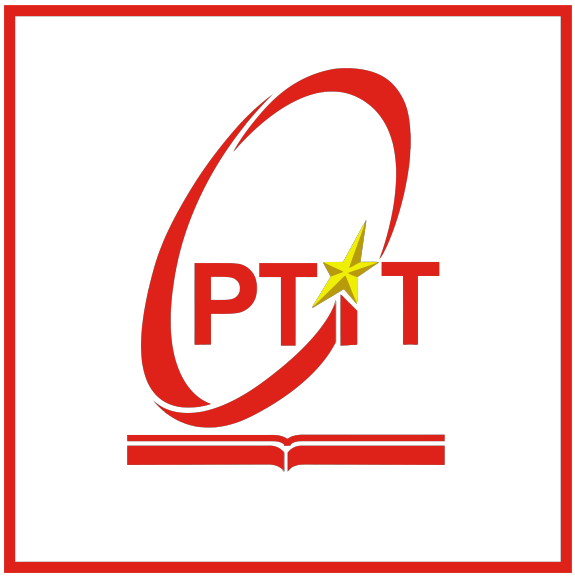
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN 1



LẬP TRÌNH PYTHON

BÀI TẬP LỚN

MÔ HÌNH NHẬN DIỆN CẢM XÚC

Giảng viên PGS.TS Nguyễn Trọng Khánh

Thành viên Nguyễn Tuấn Anh - B23DCCN039
Trần Văn Trọng - B23DCCN850
Nguyễn Văn Hùng - B23DCCN360
Phạm Công Hồng Quân - B23DCCN682

HÀ NỘI – 2025

Mục lục

Mô tả

Trong kỷ nguyên trí tuệ, trí tuệ nhân tạo (AI) đang phát triển nhanh chóng và cách mạng hóa nhiều lĩnh vực. Một lĩnh vực đặc biệt thu hút sự chú ý là việc dự đoán cảm xúc dựa trên phân tích khuôn mặt, đặc biệt với sự ảnh hưởng ngày càng tăng của mạng xã hội và các nền tảng giao tiếp. Bài nghiên cứu này giới thiệu một phương pháp sáng tạo sử dụng mạng nơ-ron tích chập (CNN) để giải quyết thách thức này. Bằng cách mô phỏng các mạng nơ-ron phức tạp trong con người và động vật, CNN cung cấp một công cụ mạnh mẽ để trích xuất các đặc trưng quan trọng từ hình ảnh khuôn mặt. Phương pháp đề xuất huấn luyện và triển khai các mô hình CNN có khả năng dự đoán chính xác cả giới tính lẫn cảm xúc. Các mô hình được huấn luyện trên một bộ dữ liệu đa dạng thu thập từ nhiều nguồn trực tuyến. Kết quả thực nghiệm rất hứa hẹn, cho thấy hiệu quả của các mô hình CNN. Đối với dự đoán cảm xúc, mô hình đạt độ chính xác 82% trên tập huấn luyện và 76% trên tập kiểm tra. Những kết quả này chứng minh tiềm năng của CNN trong việc nắm bắt chính xác thông tin về cảm xúc từ biểu cảm khuôn mặt. Đáng chú ý, các mô hình phát triển có hiệu năng thời gian thực, làm cho chúng phù hợp với các ứng dụng nhận diện khuôn mặt trong thực tế. Sự kết hợp giữa các kỹ thuật trí tuệ nhân tạo tiên tiến và CNN hứa hẹn sẽ nâng cao hiểu biết cũng như ứng dụng thực tiễn trong nhận diện giới tính và cảm xúc. Tóm lại, nghiên cứu này làm nổi bật những khả năng to lớn mà trí tuệ nhân tạo và mạng nơ-ron tích chập mang lại trong lĩnh vực nhận diện giới tính và cảm xúc, mở ra cơ hội cho những tiến bộ trong tương lai.

1 Giới thiệu

Các ứng dụng xử lý ảnh là những lĩnh vực thú vị, thu hút nhiều sự chú ý trong ngànhtrí tuệ nhân tạo, với nhiều vấn đề có tính ứng dụng cao trong thực tế. Đồng thời, với sự phát triển mạnh mẽ của các thuật toán học sâu (deep learning), đặc biệt là mạng nơ-ron tích chập (CNN), đã mang lại kết quả nổi bật trong các bài toán kiểm thử. Ví dụ: vào năm 2015, Kaiming He [2] đã đề xuất kiến trúc mạng ResNet và đạt được tỷ lệ lỗi rất thấp chỉ 3,57%. Vào năm 2012, Alex và các cộng sự nghiên cứu [1] đã đề xuất một mô hình sử dụng mạng CNN và giành chiến thắng trong cuộc thi ImageNet với tỷ lệ lỗi 15% — một cuộc thi có quy mô rất lớn về vấn đề nhận dạng và phát hiện đối tượng trong ảnh. Ở con người,khuôn mặt có thể biểu đạt trạng thái cảm xúc. Trong tâm lý học, việc nhận diện cảm xúc và giới tính từ khuôn mặt đóng vai trò quan trọng trong nhiều nghiên cứu và ứng dụng, chẳng hạn như: nhận diện cảm xúc, nghiên cứu tâm lý thích nghi, nghiên cứu nhận thức xã hội, phân tích giới tính. Do đó, việc xác định cảm xúc trên khuôn mặt là một vấn đề quan trọng, ý nghĩa và có khả năng ứng dụng thực tiễn rất lớn.

2 Nghiên cứu gần đây

Machine learning đã được chứng minh là rất hiệu quả trong các vấn đề phân loại, và một trong những nhánh của machine learning là học sâu (deep learning). Vấn đề nhận dạng cảm xúc là một trong những bài toán điển hình sử dụng thuật toán CNN và đã được nghiên cứu triển khai từ lâu. Tuy nhiên, các nghiên cứu trước đây chỉ dừng lại ở việc nghiên cứu từng đặc trưng riêng lẻ, chưa kết hợp cả hai đặc trưng trên. Trong dự án này, nhóm sẽ trình bày phương pháp xây dựng từng mô hình riêng lẻ và kết quả của mô hình. Đầu tiên, nhóm sẽ lần lượt đi qua từng bài toán để có tổng quan chung nhất

về các nghiên cứu liên quan đến bài báo này. Vấn đề nhận diện và dự đoán cảm xúc đã được nhiều công ty áp dụng để đánh giá chất lượng dịch vụ khi phân tích trạng thái cảm xúc của khách hàng trong quá trình tương tác với sản phẩm. Tại ParallelDots, họ đã áp dụng tâm lý học, mô hình hóa cảm xúc con người và trí tuệ nhân tạo để nhận diện cảm xúc. Với công nghệ này, nhà tuyển dụng có thể đánh giá mức độ tự tin của ứng viên và đưa ra quyết định xem ứng viên đó có phù hợp với vị trí làm việc tiếp xúc trực tiếp với khách hàng hay không.

3 Đề xuất thuật toán

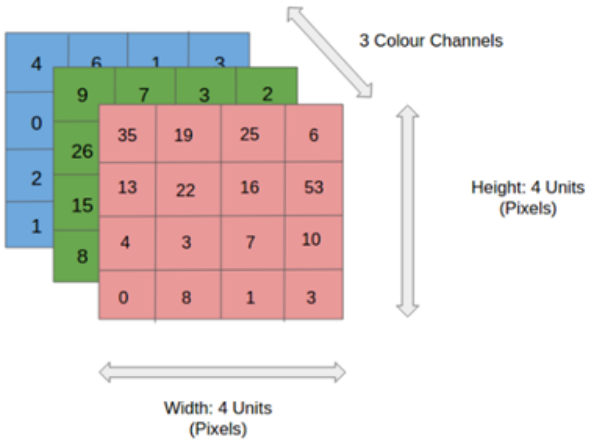
3.1 Mạng nơ-ron tích chập

3.1.1 Giới thiệu về CNN

Mạng nơ-ron tích chập (CNN) là một kiến trúc mạng mạnh mẽ, được thiết kế đặc biệt cho các bài toán nhận diện và phân loại hình ảnh. Trong số các ứng dụng khác nhau của CNN, một lĩnh vực nổi bật là nhận diện khuôn mặt con người. Với khả năng học và trích xuất các đặc trưng quan trọng từ hình ảnh, CNN đã được sử dụng rộng rãi trong lĩnh vực nhận diện khuôn mặt. Chúng đặc biệt mạnh trong việc nắm bắt các chi tiết và mẫu phức tạp trên khuôn mặt, giúp xác định chính xác từng cá nhân dựa trên đặc trưng khuôn mặt.

Thông qua việc sử dụng các tầng tích chập (convolutional layers), tầng gộp (pooling layers) và tầng kết nối đầy đủ (fully connected layers), CNN cho phép chúng ta tận dụng sức mạnh của học sâu (deep learning) để giải quyết các thách thức trong nhận diện khuôn mặt với độ chính xác và hiệu quả ấn tượng.

Mạng nơ-ron tích chập (CNN) là một phương pháp được sử dụng để phân loại hình ảnh bằng cách xử lý và gán nhãn cho hình ảnh dựa trên nội dung của chúng. Khi được đưa vào một hình ảnh, CNN xem hình ảnh như một mảng các điểm ảnh (pixel), trong đó thông tin về chiều cao, chiều rộng và chiều sâu của hình ảnh được xác định.

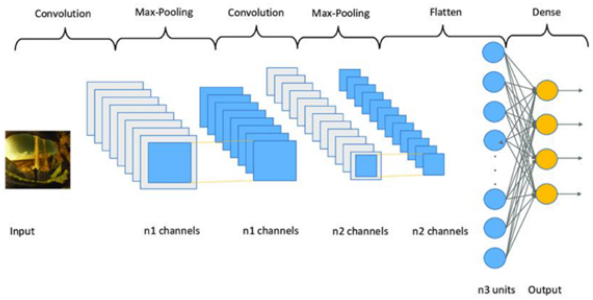


Hình 1: Cấu trúc một bức ảnh RGB

Mô hình CNN được sử dụng để huấn luyện và đánh giá, trong đó mỗi hình ảnh đầu vào sẽ đi qua một chuỗi các tầng tích chập (convolutional layers) với các bộ lọc (kernels). Sau đó, thông qua các tầng kết nối đầy đủ (fully connected layers) và sử dụng hàm

Softmax, mô hình sẽ phân loại các đối tượng với các giá trị xác suất nằm trong khoảng từ 0 đến 1.

Thông qua quá trình này, mô hình có khả năng hiểu và nhận diện các đặc trưng quan trọng trong hình ảnh, đồng thời sử dụng chúng để xác định cảm xúc của khuôn mặt.



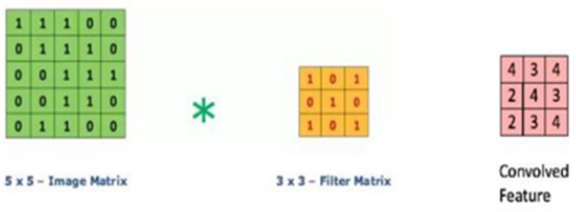
Hình 2: Sơ đồ của một mạng CNN

3.1.2 Tầng tích chập (Convolution Layer)

Tầng tích chập (Convolution) là một lớp quan trọng trong mạng nơ-ron tích chập (CNN), được sử dụng để trích xuất các đặc trưng từ hình ảnh đầu vào. Tầng tích chập giữ mối quan hệ giữa các điểm ảnh trong hình bằng cách áp dụng các bộ lọc nhỏ (filters) lên dữ liệu đầu vào.

Để hiểu cách tầng tích chập hoạt động, hãy xem xét một ma trận hình ảnh 5x5 với các giá trị điểm ảnh là 0 và 1. Đồng thời, chúng ta cũng có một ma trận bộ lọc 3x3. Bằng cách nhân ma trận hình ảnh với ma trận bộ lọc, chúng ta tạo ra một Feature Map, trong đó mỗi giá trị trong Feature Map đại diện cho một đặc trưng được tìm thấy trong hình ảnh gốc.

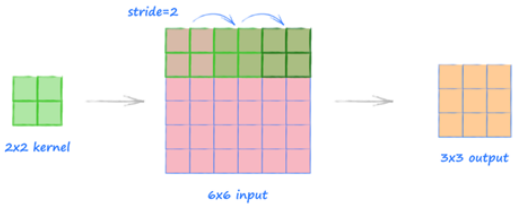
Thông qua quá trình tích chập, CNN học được các đặc trưng mức thấp như đường viền và góc cạnh, cũng như các đặc trưng mức cao như khuôn mặt và các đối tượng. Tầng tích chập giảm kích thước dữ liệu đầu vào và tạo ra các biểu diễn trừu tượng của hình ảnh, giúp huấn luyện mô hình hiệu quả hơn và cải thiện khả năng nhận diện, phân loại.



Hình 3: Mô tả quá trình tính tích chập

3.1.3 Stride

Stride (Bước nhảy) là khoảng cách giữa các vị trí liên tiếp của kernel trên ma trận đầu vào. Khi stride = 1, kernel di chuyển 1 pixel mỗi lần; khi stride = 2, kernel di chuyển 2 pixel mỗi lần; và tương tự với các giá trị stride khác. Ở mỗi vị trí di chuyển, kernel thực hiện phép tính trên các phần tử của ma trận đầu vào nằm trong vùng tương ứng với kernel.



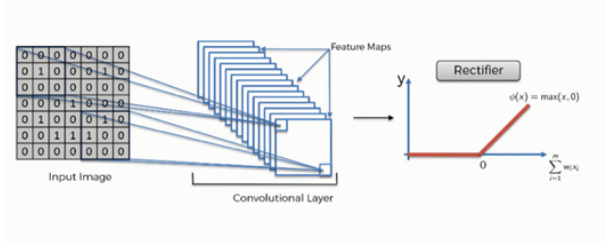
Hình 4: Mô tả Stride (Bước nhảy)

3.1.4 Padding và ReLU

Có hai phương pháp để xử lý khi kernel không phù hợp với hình ảnh đầu vào: chèn các giá trị 0 vào bốn cạnh của hình ảnh hoặc loại bỏ các phần không khớp với kernel. Hàm ReLU (Rectified Linear Unit) là một hàm phi tuyến được sử dụng phổ biến trong xây dựng mạng nơ-ron tích chập (CNN). Công thức đầu ra là:

$$f(x) = \max(0, x)$$

Trong đó, x là giá trị đầu vào. ReLU là hàm không âm, giữ nguyên các giá trị không âm và chuyển các giá trị âm thành 0.



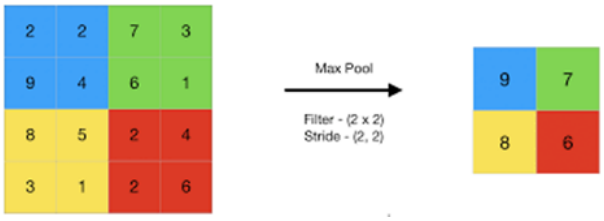
Hình 5: Mô tả quá trình padding và tính activation

ReLU được ưa chuộng trong CNN vì giải quyết tính phi tuyến của dữ liệu. Dữ liệu thường là các giá trị tuyến tính không âm, và ReLU giữ các giá trị dương không đổi trong khi đưa các giá trị âm về 0. Mặc dù các hàm tanh và sigmoid cũng có thể được sử dụng thay thế cho ReLU, nhưng ReLU vẫn được ưa chuộng nhờ hiệu suất tốt hơn trong việc xây dựng mô hình CNN.

3.1.5 Tầng Pooling

Trong xử lý ảnh bằng mạng nơ-ron tích chập (CNN), một giai đoạn quan trọng là tầng pooling được áp dụng sau tầng tích chập. Mục tiêu của tầng pooling là giảm độ phức tạp của đầu ra và giảm số lượng neuron cần tính toán. Tầng pooling thực hiện subsampling không gian, từ đó giảm kích thước mỗi feature map trong khi vẫn giữ được thông tin quan trọng nhất.

Có nhiều loại pooling như Max-Pooling, Average-Pooling, Sum-Pooling, tuy nhiên trong hầu hết các trường hợp, Max-Pooling là phương pháp phổ biến nhất. Trong Max-Pooling, giá trị lớn nhất trong vùng đầu vào của feature map được chọn làm đại diện cho vùng đó. Điều này giúp tạo ra phiên bản thu nhỏ của feature map nhưng vẫn giữ các thông tin quan trọng.



Hình 6: Mô tả quá trình Max Pooling

Tóm lại, tầng pooling trong CNN được sử dụng để giảm kích thước feature map đầu ra,đơn giản hóa mô hình và giữ lại thông tin quan trọng. Phương pháp pooling phổ biến nhất là Max-Pooling, nơi giá trị lớn nhất trong mỗi vùng đầu vào được chọn làm đại diện cho vùng đó. Do đó, chúng ta có thể thấy rằng thông qua lớp Max Pooling, số lượng neuron được giảm đáng kể so với ban đầu. Trong CNN, có nhiều Feature Map, vì vậy đối với mỗi Feature Map, Max Pooling sẽ khác nhau. Max Pooling có tác dụng tìm ra các đặc trưng quan trọng nhất trong tất cả các đặc trưng hiện có.

3.2 Mạng nơ-ron tích chập (CNN)

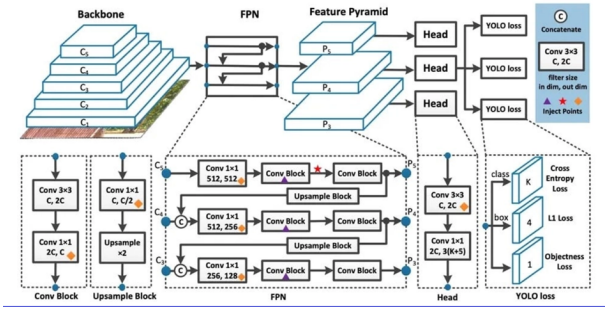
3.2.1 Ý tưởng mô hình

Trong bài toán này nhóm đã dùng 2 model:

- Yolov8 được pretrain để xác định khuôn mặt
- Mô hình CNN tự xây dựng và tự train

3.2.1.1 Với mô hình Yolo

Tổng quan Mô hình YOLOv8 được huấn luyện trên bộ dữ liệu COCO với khoảng 118.000 ảnh và 80 lớp đối tượng. Trong bài toán này, chúng em sử dụng phiên bản nano với: Số lượng tham số: ∼3.2 triệu Dung lượng mô hình (bao gồm trọng số): ∼6.2 MB Phiên bản nano phù hợp với các ứng dụng yêu cầu tốc độ cao và hạn chế tài nguyên.



Hình 7: Mô tả kiến trúc YOLOv8

Kiến trúc chi tiết Mô hình YOLOv8 gồm ba thành phần chính, mỗi thành phần đóng góp vào khả năng xử lý mạnh mẽ của mô hình:

1. **Backbone** Sử dụng CSPDarknet cải tiến, hiệu quả trong việc trích xuất các đặc trưng quan trọng từ hình ảnh.

2. **Neck** Sử dụng PAN-FPN (Path Aggregation Network - Feature Pyramid Network), tối ưu hóa việc trộn lẫn các đặc trưng ở nhiều tỷ lệ khác nhau. Cải thiện khả năng phát hiện đối tượng ở nhiều kích thước khác nhau.

3. **Head** Sử dụng cơ chế Anchor-free, cho phép mô hình dự đoán trực tiếp bounding box và các đặc trưng khác mà không cần các hộp neo định sẵn.

Cơ chế hoạt động YOLOv8 hoạt động bằng cách chia hình ảnh đầu vào thành các lưới (grid cells) và dự đoán đối tượng trong mỗi ô. Điểm khác biệt lớn so với các phiên bản trước là cơ chế anchor-free.

1. **Chia ảnh thành lưới** Hình ảnh được chia thành lưới N x N. Mỗi ô lưới chịu trách nhiệm dự đoán các đối tượng có tâm nằm trong ô đó.

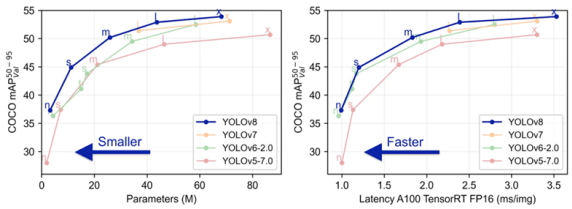
2. **Feature maps đa tầng** Sử dụng các bản đồ đặc trưng ở nhiều tỷ lệ khác nhau, giúp phát hiện cả đối tượng lớn và nhỏ hiệu quả hơn.

3. **Anchor-free Prediction** Thay vì sử dụng các anchor box được định nghĩa trước như YOLOv5, YOLOv8 dự đoán trực tiếp kích thước và vị trí bounding box, cùng với điểm tin cậy và lớp đối tượng. Điều này giúp mô hình linh hoạt hơn và giảm thiểu lỗi do chọn anchor không phù hợp.

Ưu điểm của mô hình YOLOv8

1. **Tốc độ thực (Real-time Speed)** Xử lý hình ảnh và video theo thời gian thực với độ trễ thấp. Lý tưởng cho các ứng dụng yêu cầu phản hồi nhanh.

2. **Độ chính xác cao (High Accuracy)** Đạt mAP (mean Average Precision) tốt hơn so với YOLOv5 và YOLOv7 trên nhiều bộ dữ liệu chuẩn.

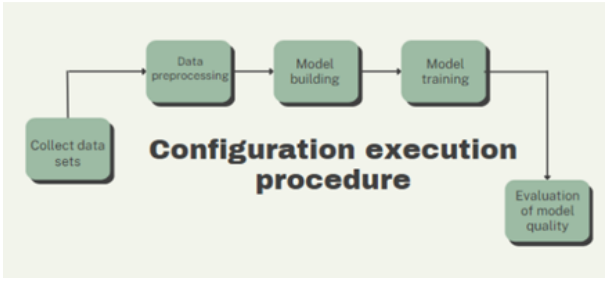


Hình 8: Bảng so sánh giữa các phiên bản YOLO

3.2.1.2 Mô hình CNN tự huấn luyện Qua nhiều lần thực nghiệm xây dựng mô hình đã từng thử cách fine tunning từ các mô hình được huấn luyện sẵn như Resnet hay Vgg tuy nhiên kết quả nhận lại không được như mong muốn. Chính vì vậy, nhóm đã thực nghiệm tự xây dựng mô hình và tự trainning và nhận được kết quả train và validaiton khá tốt . Và gọi model đó là My_model

3.3 Quy trình thực hiện

Quy trình huấn luyện được thực hiện theo các bước sau



Hình 9: Sơ đồ quy trình thực hiện

3.3.1 Data collection

Mô hình được huấn luyện trên bộ data FER2013 với 7 loại cảm xúc chính là surprise, fear, angry, neutral, sad, disgust. Tuy nhiên ứng với bài toán này, chúng em chỉ dự đoán trên 5 loại cảm xúc chính angry, happy, sad, surprise, neutral được thống kê như sau:

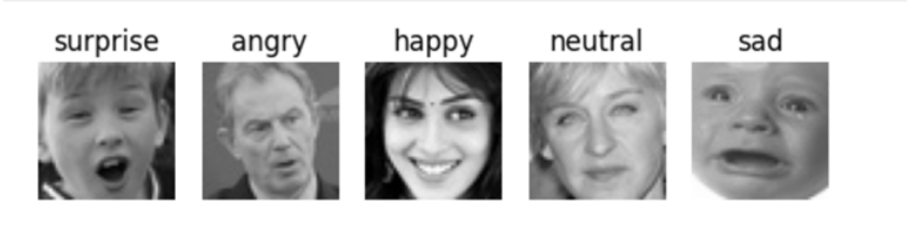
Đối với bộ train:

Loại cảm xúc	**Số lượng**
Angry	3995
Happy	7215
Sad	4830
Surprise	3171
Neutral	4965

Đối với bộ validation:

Loại cảm xúc	**Số lượng**
Angry	958
Happy	1774
Sad	1274
Surprise	831
Neutral	1233

Bộ ảnh là ảnh gray, có kích thước (48,48)



Hình 10: Ảnh mẫu từ bộ dữ liệu FER2013

3.3.2 Data preprocessing

1. RandomHorizontalFlip: Lật ngang ảnh ngẫu nhiên để tăng cường dữ liệu, giúp mô hình học tốt hơn với các biến thể.
2. RandomRotation(10): Xoay ảnh $\pm 10^\circ$ để mô hình kháng biến dạng và cải thiện khả năng tổng quát hóa.
3. ToTensor: Chuyển ảnh sang tensor PyTorch, chuẩn hóa giá trị pixel về [0, 1], thuận tiện cho quá trình huấn luyện.
4. Normalize((0.5,), (0.5,)): Chuẩn hóa pixel về [-1, 1], giúp mô hình hội tụ nhanh và ổn định hơn.

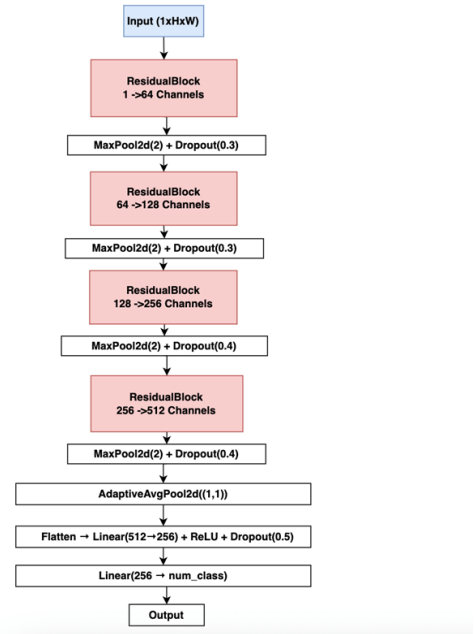
Tác dụng tổng thể: Tăng cường dữ liệu, giảm overfitting và chuẩn hóa ảnh để cải thiện hiệu suất mô hình.

3.3.3 Model building

Mô hình My_model tổng thể gồm khoảng 5 triệu tham số. Trọng số mô hình sau khi huấn luyện là 20Mb

Kiến trúc mô hình

- ResidualBlock: Kết hợp 2 Conv2d + BN + ReLU + SEBlock + shortcut.
- SEBlock: Thêm khả năng mô hình "chú ý" đến các kênh quan trọng.
- MaxPool + Dropout: Giảm kích thước feature map và hạn chế overfitting.
- AdaptiveAvgPool + Flatten + FC: Chuyển đổi feature map sang vector trước khi dự đoán lớp.



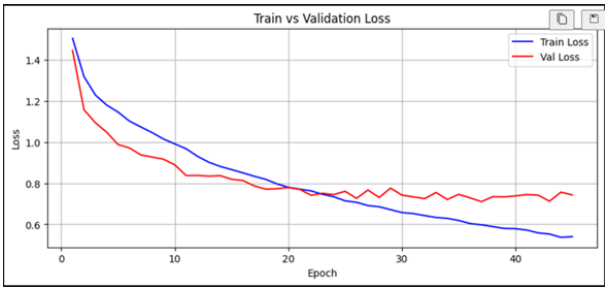
Hình 11: Kiến trúc My_model

3.3.4 Quy trình huấn luyện

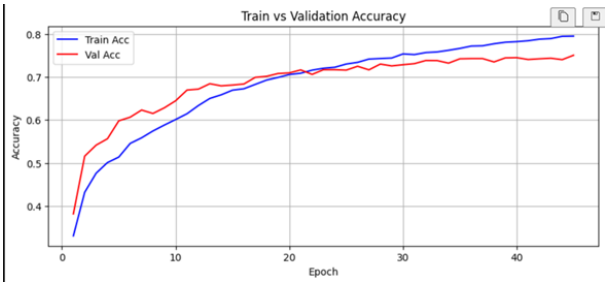
1. Chuẩn bị dữ liệu:
    - Dữ liệu được chia thành hai tập: train và validation.
    - Áp dụng train_transform (lật ngang, xoay, chuẩn hóa...) để tăng cường và chuẩn hóa dữ liệu.
    - Sử dụng DataLoader để chia dữ liệu thành từng batch (64 mẫu/batch), trộn ngẫu nhiên khi huấn luyện.
2. Cấu hình huấn luyện:
    - Thiết bị: GPU (nếu có) hoặc CPU.
    - Hàm mất mát: CrossEntropyLoss.
    - Bộ tối ưu: Adam, tốc độ học lr = 0.001.
    - Số epoch: 45.
3. Vòng lặp huấn luyện:
    - Train phase:
        § Mô hình ở chế độ train().
        § Tính toán đầu ra, loss, lan truyền ngược (backpropagation) và cập nhật trọng số.
        § Ghi lại train loss và độ chính xác (accuracy).
    - Validation phase:
        § Mô hình chuyển sang chế độ eval().
        § Không cập nhật trọng số, chỉ đánh giá loss và accuracy trên tập validation.
4. Lưu kết quả:
    - Sau mỗi epoch, lưu lại train/val loss và accuracy để theo dõi quá trình học.
    - In ra kết quả huấn luyện sau từng epoch để quan sát hiệu suất mô hình.

3.4 Kết quả huấn luyện

Sau khi hoàn thành quá trình xây dựng và huấn luyện mô hình, kết quả thu được cho thấy độ chính xác cao trong việc dự đoán giới tính vàcảm xúc từ khuôn mặt. Với mô hình dự đoán cảm xúc, mặc dù độ chính xác chưa cao bằng, nhưng vẫn đạt mức đáng kể. Cụ thể, mô hình đạt 82% độ chính xác trên tập huấn luyện và 76% trên tập kiểm định. Mặc dù mức độ này có thể cải thiện thêm, nhưng kết quả cho thấy mô hình vẫn có khả năng nhận diện cảm xúc khuôn mặt với một mức độ tin cậy nhất định



Hình 12: Biểu đồ hàm Loss trong quá trình huấn luyện



Hình 13: Biểu đồ hàm Accuracy trong quá trình huấn luyện

3.5 Kết quả đánh giá qua thực nghiệm

Tóm lại, mô hình đã đạt hiệu quả cao trong việc dự đoán giới tính và kết quả khả quan trong dự đoán cảm xúc từ khuôn mặt. Những kết quả này chứng minh tính hiệu quả của việc ứng dụng trí tuệ nhân tạo trong phân tích và nhận diện thông tin tâm lý từ khuôn mặt.

Hình 14: Kết quả đánh giá thực nghiệm

4 Tổng kết

Trong nghiên cứu này, nhóm đề xuất một mô hình học sâu (deep learning) để nhận dạng cảm xúc dựa trên hình ảnh khuôn mặt. Mô hình được xây dựng bằng các lớp mạng nơ-ron tích chập (CNN) và đã được kiểm thử nhiều lần nhằm tối ưu hóa hiệu suất. Đặc biệt,

mô hình được thiết kế để hoạt động trong thời gian thực, có khả năng nhận dạng khuôn mặt ngay khi hình ảnh được cung cấp. Trong tương lai, chúng tôi sẽ tiếp tục phát triển dự án để ứng dụng thực tế. Một phần của công việc tiếp theo là sử dụng các lớp mạng được huấn luyện sẵn (pre-trained layers) nhằm nâng cao độ chính xác và chất lượng dự đoán của mô hình. Điều này sẽ giúp cải thiện khả năng nhận dạng khuôn mặt thời gian thực và đáp ứng tốt hơn cho các ứng dụng thực tế.

Tài liệu tham khảo

[**1**] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Advances in neural information processing systems, pp. 1097-1105, 2012.

[**2**] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions", Proceedings of the IEEE conference on computer vision and patternrecognition, pp. 1–9, 2015.