HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO BÀI TẬP LỚN

HỌC PHẦN: LẬP TRÌNH VỚI PYTHON

ĐỀ TÀI: TẠO EMOJI VÀ PHÁT HIỆN GIỚI TÍNH DỰA TRÊN KHUÔN MẶT

Lớp: INT13162-01

Nhóm: 01

Thành viên nhóm:

Nguyễn Trung Anh	**B20DCAT009**
Phạm Tuấn Đức	**B20DCAT049**
Vũ Ngọc Khánh	**B20DCAT105**
Hoàng Tuấn Minh	**B20DCAT122**
Đỗ Bá Đức Toàn	**B20DCAT161**

Giảng viên hướng dẫn: Vũ Minh Mạnh

Hà Nội - 4/2023

Mục lục

I. Giới thiệu bài toán

Thời đại 4.0, thời đại mà công nghệ số lên ngôi. Đất nước chúng ta cũng đang dần công nghệ hóa, hiện đại hóa đất nước. Rất nhiều ứng dụng của AI đã được đưa vào áp dụng để có thể phục vụ được nhu cầu của con người. Và một trong những ứng dụng của AI được sử dụng nhiều nhất có lẽ là bài toán nhận diện khuôn mặt. Bài toán nhận diện khuôn mặt được ứng dụng rộng rãi trong nhiều lĩnh vực, từ bảo mật, quản lý chấm công, giao thông, du lịch, phân loại hình ảnh, tìm kiếm hình ảnh trên Internet và cả giải trí.

Xuất phát từ nhu cầu trên, nhóm chúng em muốn phát triển bài tập lớn của mình theo bài toán nhận dạng khuôn mặt với 2 phần chính. Phần thứ nhất là tạo ra một emoji theo cảm xúc của khuôn mặt, có thể áp dụng vào các nền tảng giải trí của con người như tiktok. Phần thứ 2 là nhận dạng giới tính của con người để áp dụng vào các hệ thống bảo mật như khóa thông minh và chấm công.

Bài toán của chúng em sử dụng chủ yếu phương pháp học sâu (Deep Learning) và thuật toán CNN

II. Ngôn Ngữ Lập Trình

Python là ngôn ngữ mã nguồn mở, được phát triển và duy trì bởi một cộng đồng lớn các lập trình viên trên toàn thế giới. Python có thư viện và framework rất phong phú, giúp cho việc lập trình trở nên nhanh chóng và dễ dàng hơn. Ngoài ra, Python còn có cú pháp đơn giản, dễ hiểu, giúp cho các lập trình viên mới bắt đầu học lập trình dễ dàng tiếp cận.

Python là ngôn ngữ lập trình đa năng, có thể được sử dụng để xây dựng nhiều loại ứng dụng khác nhau, từ các ứng dụng desktop đơn giản đến các ứng dụng web phức tạp, các trò chơi, máy học, trí tuệ nhân tạo, xử lý ngôn ngữ tự nhiên, phân tích dữ liệu và nhiều ứng dụng khác.

Những thư viện nhóm em dùng để phát triển ứng dụng:

- Tkinter: Tkinter là một thư viện giao diện đồ họa (GUI) phổ biến được sử dụng trong Python. Tkinter cung cấp các widget đồ họa để tạo ra giao diện cho ứng dụng, bao gồm các nút, hộp văn bản, menu, hộp chọn và nhiều hơn nữa. Tkinter cũng hỗ trợ định dạng, phông chữ, màu sắc, kích thước và định dạng hình ảnh. Tkinter cho phép tạo ra các ứng dụng đồ họa đơn giản và trực quan trên nền tảng máy tính.
- PIL: PIL (Python Imaging Library) là một thư viện Python sử dụng để xử lý hình ảnh. Thư viện này cho phép các thao tác xử lý ảnh phổ biến như cắt, chỉnh sửa kích thước, chuyển đổi định dạng, đóng dấu,

tạo hiệu ứng, v.v. PIL được phát triển bởi Fredrik Lundh và được phát hành theo giấy phép mã nguồn mở (open source). Tuy nhiên, thư viện này đã ngừng phát triển kể từ năm 2011 và được thay thế bằng thư viện Pillow, một bản fork của PIL với các cải tiến và bổ sung mới

- OpenCV (Open Source Computer Vision Library): là một thư viện mã nguồn mở dành cho xử lý ảnh và thị giác máy tính. Thư viện này được phát triển bởi Intel vào năm 1999 và hiện được duy trì bởi một nhóm các nhà phát triển đến từ nhiều quốc gia. OpenCV cung cấp các chức năng xử lý ảnh và thị giác máy tính, bao gồm các chức năng như: lọc ảnh, phát hiện đường biên, phát hiện khuôn mặt, nhận dạng vật thể, đọc và ghi file ảnh và video, và nhiều chức năng khác. OpenCV cũng được hỗ trợ trên nhiều nền tảng khác nhau như Windows, Linux, MacOS, và các thiết bị di động. OpenCV là một thư viện rất mạnh mẽ và phổ biến trong lĩnh vực thị giác máy tính và xử lý ảnh, được sử dụng rộng rãi trong nhiều ứng dụng thực tế, từ nhận dạng khuôn mặt, theo dõi đối tượng, đến xử lý ảnh y tế, tự động hóa công nghiệp, và nhiều lĩnh vực khác nữa.
- NumPy: là một thư viện phổ biến trong Python dùng để làm việc với mảng nhiều chiều (hay còn gọi là mảng đa chiều) và các phép toán trên mảng. NumPy cung cấp các hàm toán học, hàm thống kê, các hàm lô-gic, các hàm tuyến tính và hàm ngẫu nhiên để xử lý dữ liệu. Với NumPy, các phép toán trên mảng sẽ được thực hiện nhanh hơn và hiệu quả hơn so với sử dụng list trong Python. NumPy được sử dụng rộng rãi trong các lĩnh vực như khoa học dữ liệu, machine learning, xử lý ảnh và video.
- DeepFace: là một thư viện Python mã nguồn mở được sử dụng để phát hiện, nhận dạng và phân tích các đặc trưng khuôn mặt của con người trong ảnh hoặc video. Nó có khả năng thực hiện nhiều tác vụ khác nhau như phát hiện khuôn mặt, xác định độ tuổi và giới tính, phân loại tình cảm và nhận dạng danh tính. DeepFace sử dụng các mô hình học sâu để thực hiện các tác vụ này và được xây dựng trên nền tảng Keras và TensorFlow.
- Os: Thư viện os của Python cung cấp các hàm để tương tác với hệ điều hành. Thư viện này cho phép bạn thực hiện các thao tác như tạo thư mục mới, xóa thư mục, đổi tên file/thư mục, lấy danh sách các file/thư mục trong một thư mục cụ thể. Để sử dụng thư viện này bạn không cần phải cài đặt gì thêm.

III. Về thuật toán CNN:

1. CNN là gì?

CNN được viết tắt của Convolutional Neural Network hay còn được gọi là CNNS mạng nơ-ron tích chập, là một trong những mô hình Deep Learning cực kỳ tiên tiến, bởi chúng cho phép bạn xây dựng những hệ thống

có độ chính xác cao và thông minh. Nhờ khả năng đó, CNN có rất nhiều ứng dụng, đặc biệt là những bài toán cần nhận dạng vật thể (object) trong ảnh.

2. Convolutional trong CNN

Convolutional là một loại cửa sổ dạng trượt nằm trên một ma trận. Các convolutional layer sẽ chứa các parameter có khả năng tự học, qua đó sẽ điều chỉnh và tìm ra cách lấy những thông tin chính xác nhất khi không cần chọn feature. Lúc này, convolution hay tích chập đóng vai trò là nhân các phần tử thuộc ma trận. Sliding Window hay được gọi là kernel, filter hoặc feature detect là loại ma trận có kích thước nhỏ

3. Feature trong CNN

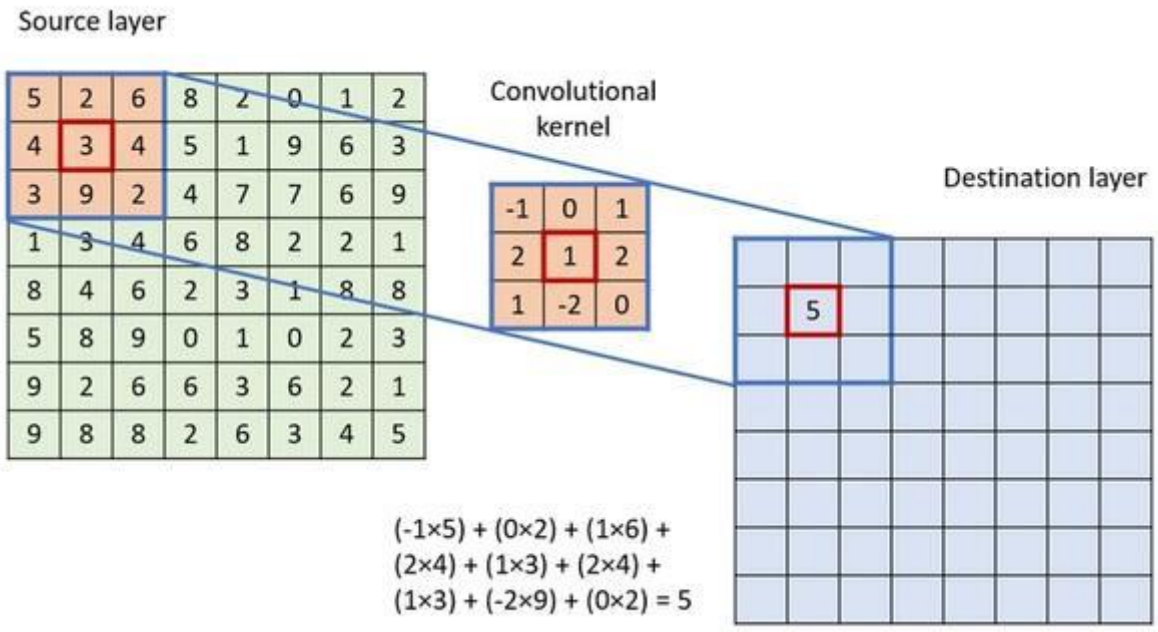
Feature là đặc trưng, mạng CNN sẽ so sánh dựa vào từng mảnh và các mảnh như vậy được gọi là feature. Thay vì phải tiến hành khớp các bức ảnh lại với nhau thì mạng CNN sẽ xác định được sự tương đồng thông qua tìm kiếm thô những đặc trưng khớp với nhau bằng hai hình ảnh tốt hơn. Một feature là mộ hình ảnh dạng mini (những mảng 2 chiều nhỏ). Những feature này đều tương ứng với một khía cạnh nào đó của hình ảnh và chúng có thể khớp lại được với nhau.

4. Những lớp cơ bản của mạng CNN

4.1 Convolutional

Lớp này là phần quan trọng nhất của toàn mạng CNN, nó có nhiệm vụ thực thi các tính toán. Các yếu tố quan trọng trong lớp Convolutional là: padding, stride, feature map và filter map.

- Mạng CNN sử dụng filter để áp dụng vào các vùng của ma trận hình ảnh. Các filter map là các ma trận vuông (Kích thước thường chọn là ma trận 3×3 hoặc ma trận 5×5), bên trong đó là những tham số và chúng được gọi là parameters.
- Stride tức là bạn dịch chuyển filter map theo từng pixel dựa vào các giá trị từ trái qua phải.
- Padding: Thường, giá trị viền xung quanh của ma trận hình ảnh sẽ được gán các giá trị 0 để có thể tiến hành nhân tích chập mà không làm giảm kích thước ma trận ảnh ban đầu.
- Feature map: Biểu diễn kết quả sau mỗi lần feature map quét qua ma trận ảnh đầu vào. Sau mỗi lần quét thì lớp Convolutional sẽ tiến hành tính toán.



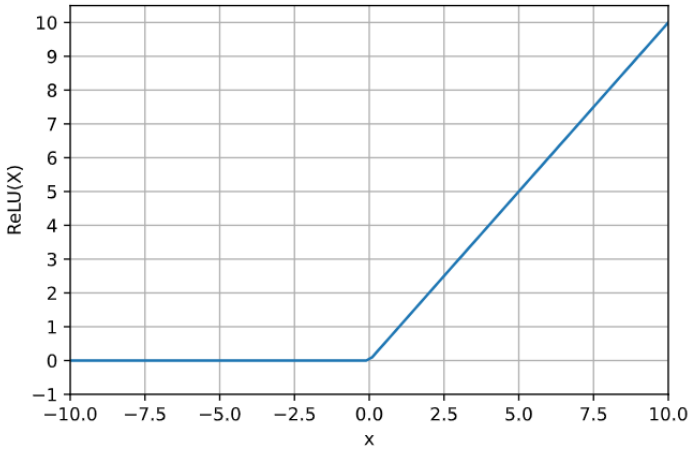
4.2 Relu

Lớp ReLU này là hàm kích hoạt trong mạng CNN, được gọi là activation function. Nó có tác dụng mô phỏng những nơ ron có tỷ lệ truyền xung qua axon. Các hàm activation khác như Leaky, Sigmoid, Leaky, Maxout,..tuy nhiên hiện nay, hàm ReLU được sử dụng phổ biến và thông dụng nhất.

Hàm này được sử dụng cho những yêu cầu huấn luyện mạng nơ ron với những ưu điểm nổi bật điển hình là hỗ trợ tính toán nhanh hơn. Trong quá trình dùng hàm ReLU, bạn cần chú ý đến việc tùy chỉnh những learning rate và dead unit. Những lớp ReLU được dùng sau khi filter map được tính và áp dụng ReLU lên các giá trị của filter map.

Công thức hàm ReLU: f(x)=max (0, x)

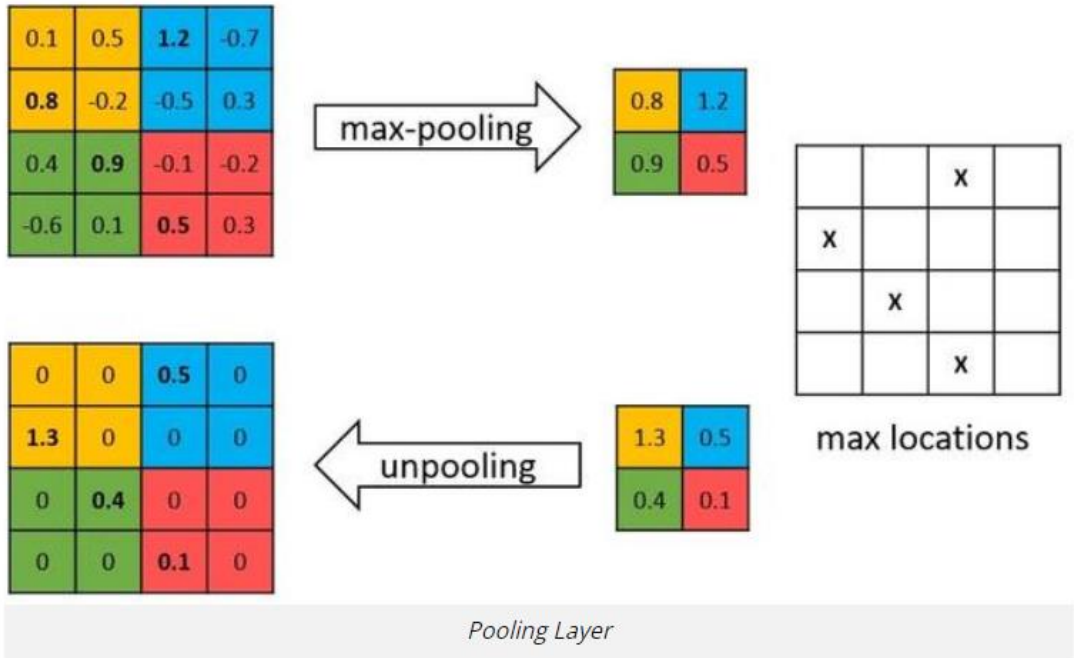
Phân tích



4.3 Pooling

Khi ma trận ảnh đầu vào có kích thước quá lớn, các lớp Pooling layer sẽ được đặt vào giữa những lớp Convolutional để làm giảm những parameters. Hiện, hai loại lớp Pooling được sử dụng phổ biến là Max pooling và Average.

- *Max Pooling* có chức năng giảm kích thước của đầu vào bằng cách giữ lại giá trị lớn nhất (max value) trong mỗi vùng trượt của đầu vào.



Pooling Layer

Ví dụ, nếu ta có một ma trận đầu vào có kích thước 4x4 và một lớp max pooling với kernel size (kích thước kernel) là 2x2 và stride (độ dịch chuyển) là 2, thì max pooling sẽ tách ma trận đầu vào thành các vùng không trùng nhau có kích thước 2x2, và giữ lại giá trị lớn nhất của từng vùng đó. Kết quả thu được là một ma trận có kích thước là 2x2.

Max pooling giúp giảm kích thước của đầu vào, giúp giảm số lượng tham số cần học và giảm khả năng overfitting (quá khớp) của mô hình. Ngoài ra, nó còn giúp giảm chi phí tính toán trong quá trình huấn luyện và dự đoán.

- *Unpooling* là quá trình khôi phục kích thước của đầu vào ban đầu sau khi đã được downsample (lấy mẫu xuống). Khi chúng ta thực hiện max pooling để giảm kích thước của feature map, thông tin đã bị mất và chúng ta cần khôi phục nó. Quá trình unpooling sẽ giúp chúng ta khôi phục thông tin bị mất và đưa kích thước của feature map về giá trị ban đầu.

Một trong những cách phổ biến để thực hiện unpooling trong CNN là sử dụng Upsampling layer hoặc Deconvolution layer. Upsampling layer sẽ tăng kích thước của feature map bằng cách chèn thêm các giá trị 0 vào giữa các phần tử. Deconvolution layer sẽ thực hiện phép toán ngược lại với convolution layer để khôi phục lại kích thước feature map.

Tuy nhiên, quá trình unpooling trong CNN vẫn là một vấn đề đang được tiếp tục nghiên cứu và cải tiến để cải thiện độ chính xác và hiệu quả của mô hình.

4.4 Fully connected

Đây là lớp có nhiệm vụ đưa ra kết quả sau khi hai lớp Convolutional và Pooling đả nhận được ảnh truyền. Khi này, ta sẽ thu được một model đọc được thông tin của ảnh. Để có thể liên kế chúng cũng như cho nhiều đầu ra hơn ta sẽ sử dụng Fully connected layer.

Ngoài ra, nếu lớp này có dữ liệu hình ảnh thì lớp sẽ chuyển chúng thành các model chưa được phân chia chất lượng để tìm ra ảnh có chất lượng cao nhất.

5. Kiến trúc của CNN

Cấu trúc cơ bản của một mô hình mạng CNN thường bao gồm 3 phần chính bao gồm:

- Trường cục bộ/ Local receptive field: Lớp này sử dụng để tách lọc dữ liệu, thông tin hình ảnh để từ đó có thể lựa chọn các vùng có giá trị sử dụng hiệu quả cao nhất.
- Trọng số chia sẻ/ Shared weights and bias: Lớp này hỗ trợ làm giảm các tham số đến mức tối thiểu trong mạng CNN. Trong từng lớp convolution sẽ chứa các feature map riêng và từng feature thì sẽ có khả năng phát hiện một vài feature trong hình ảnh.
- Lớp tổng hợp/ Pooling layer: Đây là lớp cuối cùng và sử dụng để làm đơn giản các thông tin output. Tức là, sau khi tính toán xong và quét qua các layer trong mạng thì pooling layer sẽ được dùng để lược bỏ các thông tin không hữu ích. Từ đó cho ra kết quả theo kỳ vọng người dùng.

IV. Bài toán nhận dạng khuôn mặt bằng mạng nơ ron tích chập

Bài toán nhận dạng khuôn mặt bằng mạng nơ ron tích chập là 1 dạng của bài toán nhận dạng bằng mạng nơ ron tích chập vậy nên nó cũng có các đặc điểm của bài toán nhận dạng.

1. Tổng quan về nhận dạng:

Nhận dạng là một ngành thuộc lĩnh vực trí tuệ nhân tạo. Nhận dạng mẫu là khả năng phát hiện sự sắp xếp các đặc tính hoặc dữ liệu mang lại thông tin về một hệ thống hoặc tập dữ liệu nhất định. Nhận dạng mẫu chia thành nhiều lĩnh vực trong công nghệ thông tin, bao gồm phân tích dữ liệu lớn, nhận dạng sinh trắc học, bảo mật và trí tuệ nhân tạo. Nhận dạng đối tượng trong hình ảnh là một nhánh của nhận dạng mẫu. Nhận dạng đối tượng trong hình ảnh thể hiện qua các công nghệ máy tính có thể nhận ra người, động vật, vật thể hoặc các đối tượng mục tiêu khác thông qua việc sử dụng các thuật toán và khái niệm học máy.

Một số khái niệm liên quan:

- Nhận dạng đối tượng (object recognition) là một thuật ngữ chung để mô tả một tập hợp các cách xử lý trong thị giác máy tính có liên quan đến việc xác định các đối tượng trong ảnh kỹ thuật số.
- Phân loại hình ảnh (image classification) liên quan đến việc dự đoán, phân loại các lớp thông tin của một đối tượng trong một hình ảnh.
- Khoanh vùng đối tượng (object localization) là việc xác định vị trí của một hoặc nhiều đối tượng trong một hình ảnh bằng hình chữ nhật xung quanh phạm vi của đối tượng bằng hộp chứa (bounding box).
- Phát hiện đối tượng (object detection) kết hợp cả hai nhiệm vụ nhận dạng đối tượng và khoanh vùng đối tượng. Phát hiện đối tượng là việc khoanh vùng và phân loại một hoặc nhiều đối tượng khác nhau trong một hình ảnh.

Nhận dạng khuôn mặt là khả năng nhận diện, lưu trữ, so sánh và phân tích các mẫu dựa trên đường nét khuôn mặt để nhận dạng người từ hình ảnh hoặc video. Hệ thống thường sử dụng các công nghệ để thực hiện nhận dạng khuôn mặt như sử dụng sinh trắc học để ánh xạ các đặc điểm khuôn mặt từ ảnh hoặc video. Hệ thống so sánh thông tin này với một cơ sở dữ liệu đã lưu trữ về các khuôn mặt để tìm ra một kết quả khớp chính xác

2. Cách thức hoạt động:

Phương pháp bao gồm các bước khác nhau để thực hiện nhận diện khuôn mặt tự động.

Đầu tiên là thực hiện phát hiện khuôn mặt để khoanh vùng khuôn mặt trong từng khung hình ảnh và video.

Sau đó, dữ liệu ở bước trên được liên kết với các khuôn mặt được phát hiện với danh tính chung trên các video và căn chỉnh các khuôn mặt thành tọa độ bằng cách sử dụng các mốc được phát hiện. Cuối cùng, hệ thống thực hiện xác minh khuôn mặt để tính toán độ tương tự giữa một cặp hình ảnh / video.

V. Mô hình nhận diện khuôn mặt DeepFace

DeepFace là một mô hình mạng neural sâu được phát triển bởi Facebook AI Research (FAIR) vào năm 2014 để giải quyết bài toán nhận diện khuôn mặt. Mô hình này được huấn luyện trên một tập dữ liệu lớn các ảnh khuôn mặt để học cách trích xuất các đặc trưng quan trọng của khuôn mặt, từ đó có thể phân loại và nhận diện khuôn mặt với độ chính xác cao.

DeepFace sử dụng nhiều cơ sở dữ liệu khác nhau để huấn luyện và thực hiện các tác vụ nhận dạng khuôn mặt, bao gồm VGGFace, FaceNet, và OpenFace. Những cơ sở dữ liệu này chứa hình ảnh khuôn mặt của nhiều người, được sử dụng để huấn luyện các mô hình Deep Learning và đánh giá hiệu suất của các mô hình này trong các tác vụ nhận dạng khuôn mặt.

DeepFace sử dụng một kiến trúc mạng neural sâu với hơn 120 triệu tham số, bao gồm nhiều lớp tích chập và lớp pooling để trích xuất các đặc trưng ảnh khuôn mặt và một mạng neural đầy đủ kết nối để phân loại khuôn mặt. Kết quả thực nghiệm cho thấy DeepFace đạt độ chính xác nhận diện khuôn mặt trên tập

dữ liệu Facescrub lên tới 97,35%, và trên tập dữ liệu YTF (YouTube Faces) là 91,4%.

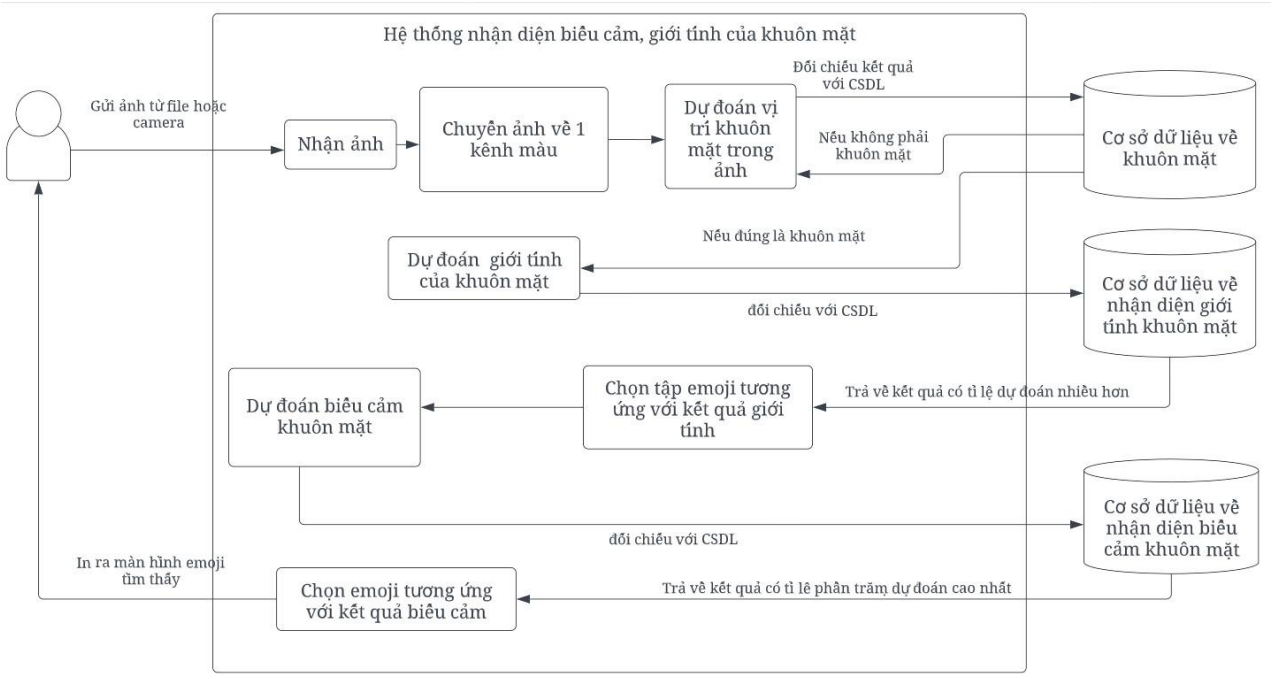
Phương pháp	Độ chính xác
Joint Bayesian	0.9242 ±0.0108
Tom-vs-Pete	0.9330 ±0.0128
High-dim LBP	0.9517 ±0.0113
TL Joint Bayesian	0.9633 ±0.0108
DeepFace-single	**0.9592** ±0.0029
DeepFace-single	**0.9700** ±0.0028
DeepFace-ensemble	**0.9715** ±0.0027
DeepFace-ensemble	**0.9735** ±0.0025
Human, cropped	0.9753

Kết quả của khi so sánh DeepFace với các công nghệ hiện đại trên bộ dữ liệu LFW

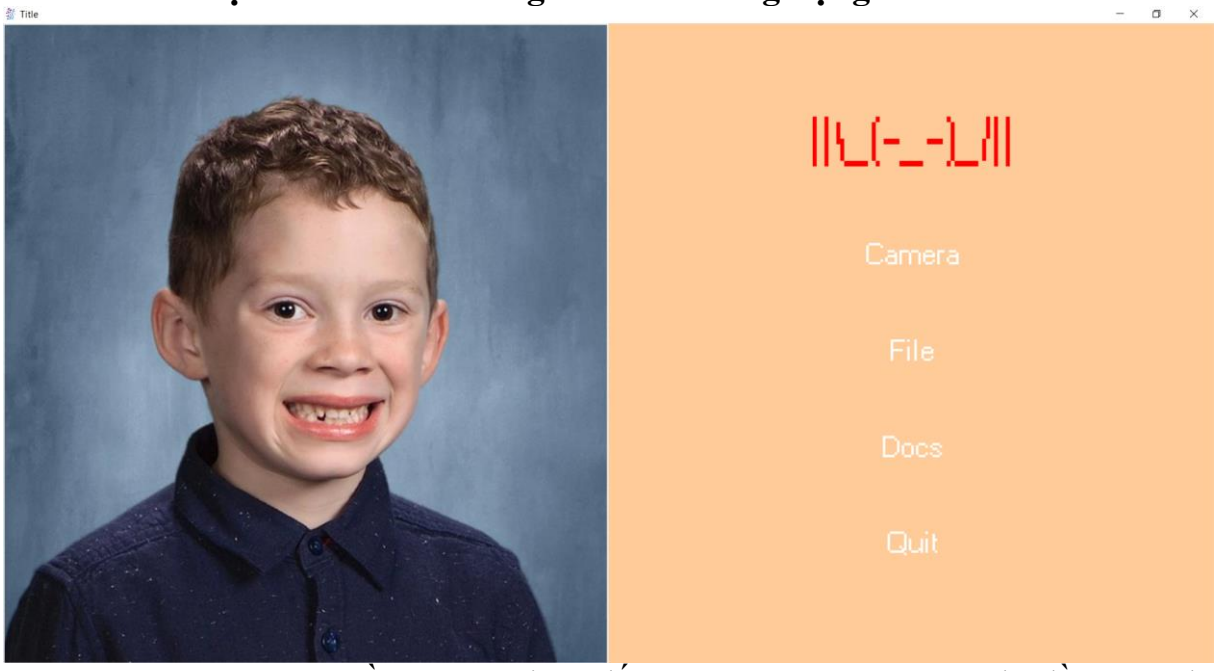
Có thể thấy, mô hình DeepFace đạt kết quả rất cao, lên đến 0.95 tới 0.97, gần ở mức tiệm cận con người.

VI. Thiết kế chức năng hệ thống

1. Mô hình tổng quan về các chức năng của hệ thống

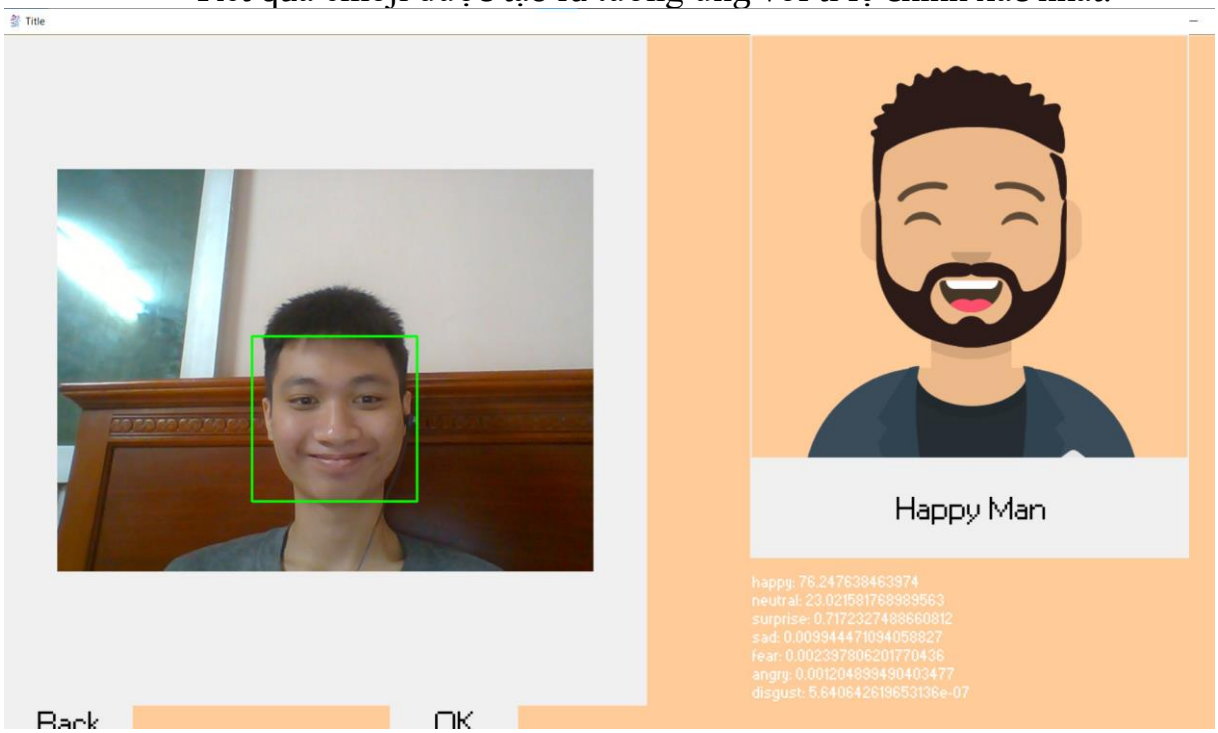


2. Giao diện và các chức năng chính của ứng dụng:

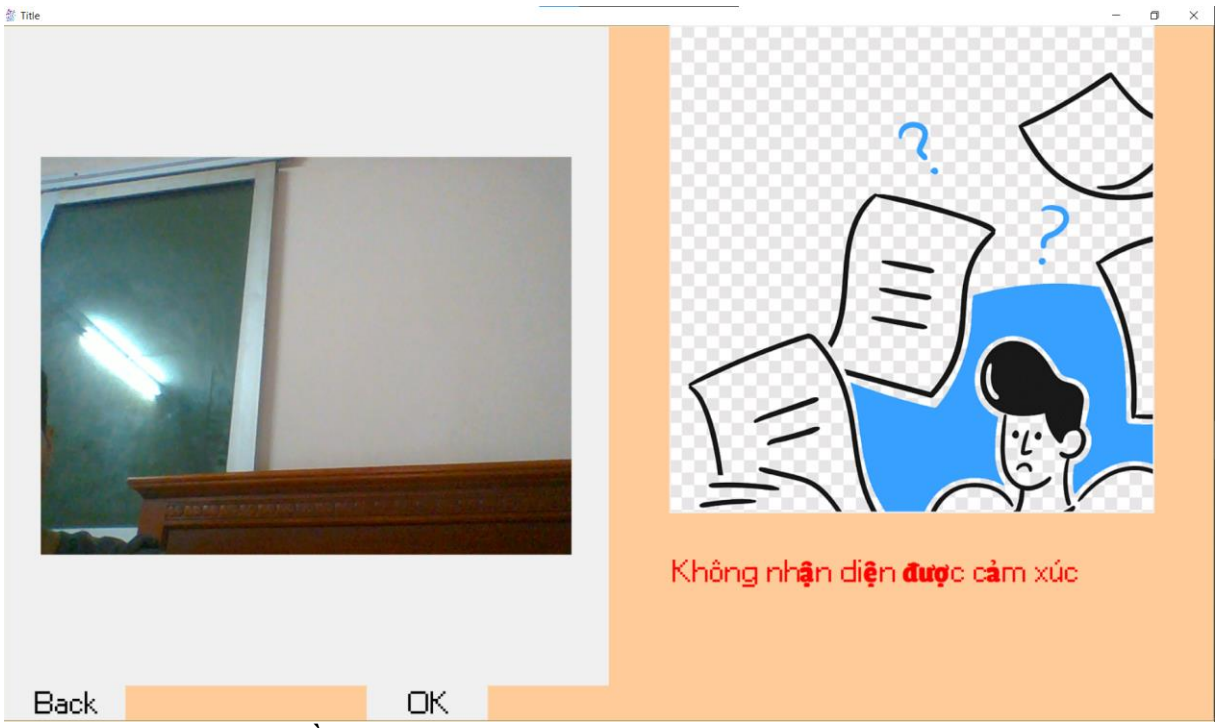


- Nút “Camera”: Đầu vào sẽ được lấy từ camera của máy tính, đầu ra sẽ là emoji tương ứng với biểu cảm và giới tính của khuôn mặt nhận vào

– Khi nhận diện được khuôn mặt ta nhận được kết quả như phía bên phải gồm:
  – Tỉ lệ phần trăm dự đoán các biểu cảm mà máy nhận dạng được.
  – Kết quả emoji được tạo ra tương ứng với tỉ lệ chính xác nhất.

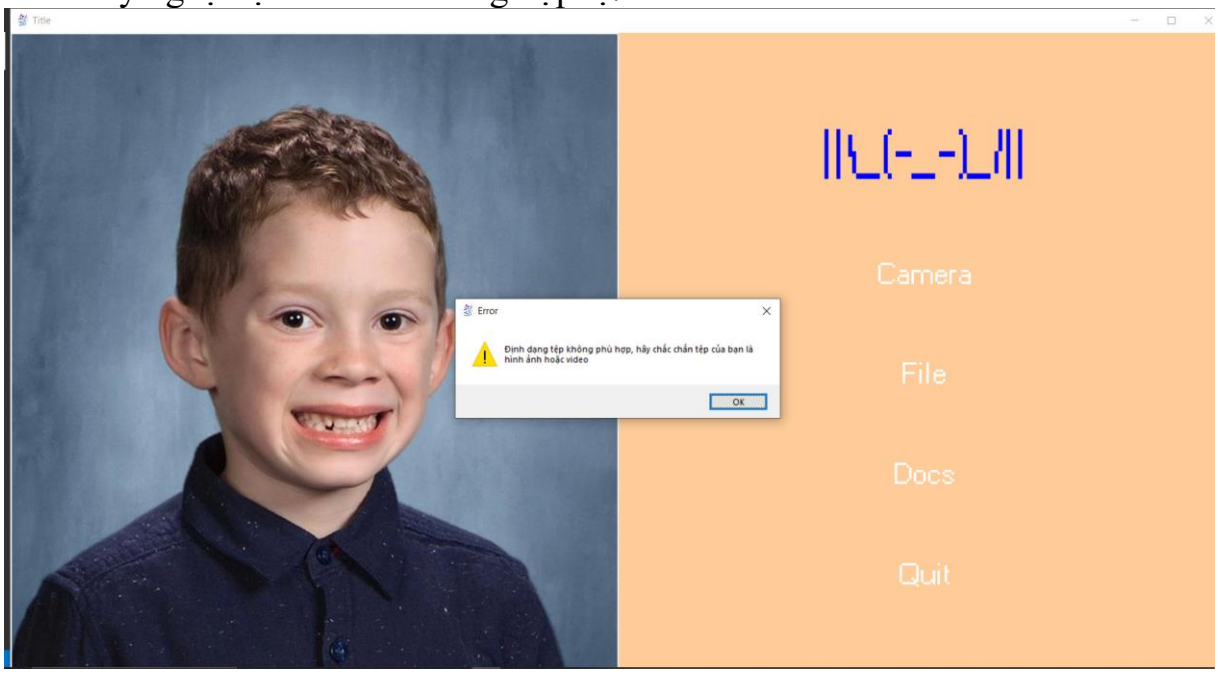


– Khi không nhận diện được khuôn mặt, kết quả trả về sẽ như sau:



- Nút “File”: Đầu vào sẽ là hình ảnh hoặc video trong máy tính của mình, đầu ra sẽ là emoji tương ứng với biểu cảm và giới tính của khuôn mặt nhận vào

– Xử lý ngoại lệ - mở file không hợp lệ:



– Khi nhận diện được khuôn mặt ở file ảnh ta thu được kết quả như khi nhận diện được khuôn mặt ở camera:

– Khi không nhận diện được khuôn mặt hoặc cảm xúc cuả con người:

- Nút “Docs”: Chuyển hướng tới tệp word là tài liệu của nhóm.
- Nút “Quit”: Thoát khỏi giao diện ứng dụng

VII. Đánh giá và định hướng phát triển trong tương lai

Ưu điểm:

- Giao diện bắt mắt, dễ sử dụng
- Tốc độ hiển thị ra các emoji tương ứng khá nhanh và chính xác
- Đầu vào đa dạng: ảnh, video và camera

Nhược điểm:

- Ứng dụng mới chỉ dừng lại ở việc tạo ra emoji tương ứng với đầu vào, chưa áp dụng vào ứng dụng cụ thể nào

Hướng phát triển trong tương lai:

- Dự định trong tương lai của nhóm em sẽ tích hợp vào các ứng dụng giải trí và bảo mật như tiktok, ứng dụng mở khóa bằng khuôn mặt hoặc chấm công.

BẢNG PHÂN CÔNG NHIỆM VỤ

STT	Tên thành viên Mã sinh viên	Nhiệm vụ	Tỷ lệ đóng góp
1	Hoàng Tuấn Minh B20DCAT122	Viết báo cáo. Xây dựng code nhận diện khuôn mặt,cảm xúc và giới tính bằng Tensor Flow.	20%
2	Vũ Ngọc Khánh B20DCAT105	Lập kế hoạch, Tìm hiểu thuật toán, Xây dựng code tìm kiếm khuôn mặt và cảm xúc bằng Deepface.	20%
3	Đỗ Bá Đức Toàn B20DCAT161	Tinh chỉnh code, code giao diện phần thực thi, nhận diện giới tính, xử lý các ngoại lệ.	25%
4	Nguyễn Trung Anh B20DCAT009	Viết báo cáo. Tìm hiểu thuật toán. Xây dựng code nhận diện khuôn mặt và cảm xúc, giới tính bằng Tensor Flow.	20%
5	Phạm Tuấn Đức B20DCAT049	Thiết kế và code giao diện.	15%

Tài Liệu Tham Khảo

https://pypi.org/project/deepface/
https://www.geeksforgeeks.org/
https://viblo.asia/p/deep-learning-tim-hieu-ve-mang-tich-chap-cnn-maGK73bOKj2