|  |
| --- |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO**  **TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH**  **KHOA CÔNG NGHỆ THÔNG TIN**  Description: C:\Documents and Settings\Administrator\Desktop\logo dai hoc_khong nen.png  **TIỂU LUẬN MÔN HỌC**  **CHUYÊN ĐỀ CHUYÊN SÂU KHDL**  **BÀI TOÁN PHÂN TÍCH NHẬN DIỆN KHUÔN MẶT**  Giảng viên giảng dạy: ThS. Vương Xuân Chí  Sinh viên thực hiện: Trần Thanh Tùng  MSSV : 2100009418  Chuyên ngành : KHOA HỌC DỮ LIỆU  Môn học : CHUYÊN ĐỀ CHUYÊN SÂU KHOA HỌC DỮ LIỆU  Khóa : 202    **Tp.HCM, ngày 25 tháng 7 Năm 2024** |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO**  **TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH**  **KHOA CÔNG NGHỆ THÔNG TIN**  Description: C:\Documents and Settings\Administrator\Desktop\logo dai hoc_khong nen.png  **TIỂU LUẬN MÔN HỌC**  **CHUYÊN NGÀNH KHOA HỌC DỮ LIỆU**  **Phân tich Bài toán phát hiện khuôn mặt**  Giảng viên giảng dạy: ThS. Vương Xuân Chí  Sinh viên thực hiện: Trần Thanh Tùng  MSSV : 2100009418  Chuyên ngành : KHOA HỌC DỮ LIỆU  Môn học : DEEP LEARNING TRONG KHOA HỌC DỮ LIỆU  Khóa :    **Tp.HCM, ngày 25 tháng 7 Năm 2024** |

|  |  |
| --- | --- |
| Trường Đại học Nguyễn Tất Thành  **Khoa Công Nghệ Thông Tin**  🙜 🙜 🙝 🙝 | CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  **Độc lập – Tự do – Hạnh phúc**  🙜 🙜 🙝 🙝 |

**NHIỆM VỤ ĐỒ ÁN**

Họ và tên: MSSV:

Chuyên ngành: KHOA HỌC DỮ LIỆU Lớp:

Email: SĐT : 0369273641

Tên đề tài: DỰ ĐOÁN PHÂN LỚP

Giảng viên giảng dạy:

Thời gian thực hiện:13/12/2022 đến 26/12/2022.

Nhiệm vụ/nội dung (mô tả chi tiết nội dung, yêu cầu, phương pháp…):

Nội dung và yêu cầu đã được thông qua bộ môn.

*TP.HCM, ngày 25 tháng 07 năm 2024*

|  |  |
| --- | --- |
| **TRƯỞNG BỘ MÔN**  *(Ký và ghi rõ họ tên)* | **GIÁO VIÊN HƯỚNG DẪN**  *(Ký và ghi rõ họ tên)* |

**LỜI MỞ ĐẦU**

Trong thời đại công nghệ số, công nghệ nhận dạng khuôn mặt đã trở thành một trong những lĩnh vực nghiên cứu và ứng dụng quan trọng . Nhận dạng khuôn mặt không chỉ đóng vai trò quan trọng trong việc bảo mật và giám sát, mà còn có ứng dụng rộng rãi trong các lĩnh vực như y tế,giáo dục,doanh nghiệp đặc biệt trong lĩnh vực ngân hàng . Với sự phát triển vượt bậc của trí tuệ nhân tạo và học máy, việc phân tích và phát hiện khuôn mặt đã đạt được nhiều tiến bộ vượt bậc và quan trọng hơn, mở ra nhiều triển vọng mới trong việc áp dụng công nghệ này vào đời sống thực tiễn.

Đồ án này nhằm nghiên cứu và phân tích bài toán phát hiện khuôn mặt, từ đó đưa ra các giải pháp kỹ thuật và thuật toán hiệu quả để nhận diện khuôn mặt một cách chính xác và nhanh chóng. Trong quá trình thực hiện, đồ án sẽ sử dụng các phương pháp và thuật toán như mạng nơ-ron,CNN để thực hiện nhằm tối ưu nhằm nâng cao hiệu suất và độ chính xác của hệ thống phát hiện khuôn mặt.

**LỜI CẢM ƠN**

Em xin gửi lời cảm ơn chân thành nhất đến thầy Chí đã dành cho em sự quan tâm, giúp đỡ và truyền đạt những kiến thức quý báu trong suốt thời gian học tập tại trường.

Nhờ có những bài giảng tận tâm, nhiệt tình của thầy, em đã có cơ hội được tiếp cận với những kiến thức mới, bổ sung và nâng cao vốn hiểu biết của mình. Đặc biệt, trong quá trình thực hiện đề tài phân tích bài toán phát hiện khuôn mặt, em đã nhận được sự hướng dẫn, chỉ bảo tận tình của thầy. Thầy đã giúp em định hướng đề tài, giải đáp những thắc mắc và góp ý cho sản phẩm của em được hoàn thiện hơn.

Em biết rằng, bài báo cáo của em còn nhiều hạn chế và thiếu sót. Em rất mong nhận được những ý kiến đóng góp quý báu của quý thầy cô để em có thể hoàn thiện hơn về kiến thức và kỹ năng của mình.

Một lần nữa, em xin chân thành cảm ơn quý thầy cô đã luôn đồng hành và giúp đỡ em trong suốt thời gian qua. Em sẽ cố gắng học tập và rèn luyện thật tốt để xứng đáng với sự tin tưởng và kỳ vọng của thầy.

Sinh viên thực hiện

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH  **TRUNG TÂM KHẢO THÍ** | **KỲ THI KẾT THÚC HỌC PHẦN**  **HỌC KỲ I NĂM HỌC 2022 - 2023** |

**PHIẾU CHẤM THI TIỂU LUẬN/ĐỒ ÁN**

BM-ChT-11

Môn thi: Deep learning trong khoa học dữ liệu Lớp học phần:

Nhóm sinh viên thực hiện :

1. Tham gia đóng góp:

2. Tham gia đóng góp:

3. Tham gia đóng góp:

4. Tham gia đóng góp:

5. Tham gia đóng góp:

6. Tham gia đóng góp:

7. Tham gia đóng góp:

8. Tham gia đóng góp:

Ngày thi: Phòng thi:

Đề tài tiểu luận/báo cáo của sinh viên : DỰ ĐOÁN PHÂN LỚP

Phần đánh giá của giảng viên (căn cứ trên thang rubrics của môn học):

|  |  |  |  |
| --- | --- | --- | --- |
| **Tiêu chí (theo CĐR HP)** | **Đánh giá của GV** | **Điểm tối đa** | **Điểm đạt được** |
| Cấu trúc của báo cáo | Gồm 3 chương | 1 |  |
| Nội dung |  |  |  |
| * Các nội dung thành phần | Chương 2 và 3 | 5 |  |
| * Lập luận | Chương 1 | 2 |  |
| * Kết luận |  | 1 |  |
| Trình bày | Theo chuẩn format luận văn font chữ 13, canh trái, phải, … | 1 |  |
| **TỔNG ĐIỂM** |  | **10** |  |

|  |  |
| --- | --- |
|  | **Giảng viên chấm thi**  *(ký, ghi rõ họ tên)* |

**NHẬN XÉT CỦA GIẢNG VIÊN GIẢNG DẠY**

*Tp.HCM, Ngày 28 tháng 12 năm 2022*

**Giảng viên giảng dạy**

(Ký tên và ghi rõ họ tên)

**MỤC LỤC**

[CHƯƠNG 1: GIỚI THIỆU 1](#_Toc172192875)

[1.Giới thiệu đề tài. 1](#_Toc172192876)

[1.2. Lý do chọn đề tài 1](#_Toc172192877)

[1.4. Phương pháp đề tài. 1](#_Toc172192878)

[1.5. Đối tượng và phạm vi nghiên cứu. 2](#_Toc172192879)

[CHƯƠNG 2 ỨNG DỤNG VÀ THUẬT TOÁN 3](#_Toc172192880)

[2.1 Giới thiệu bài toán. 3](#_Toc172192881)

[2.1.2 Tổng quan về mạng nơ-ron 4](#_Toc172192882)

[2.1.3 Tổng quan về YOLO. 7](#_Toc172192883)

[CHƯƠNG 3: THỰC NGHIỆM 9](#_Toc172192884)

[3.1 Công cụ sử dụng 9](#_Toc172192885)

[3.2. Xây dựng ứng dụng và giải thích 9](#_Toc172192886)

[3.1.1 Chuẩn bị, xáo trộn vị trí file, chia tập tin 10](#_Toc172192887)

[3.1.2 Đọc file csv để đọc các thông gắn nhãn của hình anh train 11](#_Toc172192888)

[3.1.3 Lấy label cho các tập 12](#_Toc172192889)

[3.1.4 Chuyển đổi label cho phù hợp với model 13](#_Toc172192890)

[3.1.5 Tạo yaml 14](#_Toc172192891)

[3.1.6 Tải thư viện YOLO 14](#_Toc172192892)

[3.1.6 Train model 15](#_Toc172192893)

[3.1.6 Kiểm tra tính chính xác 16](#_Toc172192894)

[3.1.7 Xuất model train 17](#_Toc172192895)

[3.3 Đánh giá hiệu suất 17](#_Toc172192896)

[3.4 Ứng dụng mô hình vào camera của máy tính 20](#_Toc172192897)

[KẾT LUẬN 23](#_Toc172192898)

[Kết quả đạt được. 23](#_Toc172192899)

[Hạn chế và hướng phát triển. 23](#_Toc172192900)

[Hạn chế: 23](#_Toc172192901)

[Hướng phát triển: 23](#_Toc172192902)

[TÀI LIỆU THAM KHẢO 24](#_Toc172192903)

**DANH MỤC HÌNH**

[Hình 1:Mô hình CNN 3](#_Toc172238586)

[Hình 2: Cấu trúc CNN 5](#_Toc172238587)

[Hình 3: Cấu trúc YOLO 7](#_Toc172238588)

[Hình 4: Cách thức nhận dạng YOLO 7](#_Toc172238589)

[Hình 5: Phương thức split\_data 10](#_Toc172238590)

[Hình 6: Làm sạch dữ liệu 11](#_Toc172238591)

[Hình 7: Lấy nhãn cho hình ảnh 12](#_Toc172238592)

[Hình 8: Phương thức convert\_to\_YOLO 13](#_Toc172238593)

[Hình 9: Tạo file YAML chứa hình ảnh 14](#_Toc172238594)

[Hình 10: Thư viện YOLO 14](#_Toc172238595)

[Hình 11: Thực hiện training model 15](#_Toc172238596)

[Hình 12: Kết quả training 15](#_Toc172238597)

[Hình 13: Phương thức drawbox 16](#_Toc172238598)

[Hình 14: Kết quả drawbox 16](#_Toc172238599)

[Hình 15: Truyền file train vào drive 17](#_Toc172238600)

[Hình 16: Ma trận nhầm lẫn và trực quan hóa 18](#_Toc172238601)

[Hình 17: Độ tin cậy của f1 19](#_Toc172238602)

[Hình 18: Tập hợp các biểu đồ học máy 19](#_Toc172238603)

[Hình 19: Kết quả training 20](#_Toc172238604)

[Hình 20: Phân biệt đối tượng trong webcam 21](#_Toc172238605)

[Hình 21: Kết quả thử nghiệm bằng OPENCV 22](#_Toc172238606)

**KÍ HIỆU CÁC CỤM TỪ VIẾT TẮT**

|  |  |
| --- | --- |
| Chữ viết tắt | Ý nghĩa |
| ANN | Artificial Neural Network |
| RNN | Recurrent Neural Network |
| YOLO | You Only Look Once |
| CNN | Convolutional Neural Network |
| RCNN | Region-Based Convolutional Neural Networks |
| YAML | Ain't Markup Language |

# CHƯƠNG 1: GIỚI THIỆU

## 1.Giới thiệu đề tài.

Công nghệ trí tuệ nhân tạo (AI) và học sâu (deep learning) có khả năng vượt trội trong việc giải quyết các bài toán phức tạp, đặc biệt là trong lĩnh vực xử lý ảnh và nhận dạng hình ảnh. Bài toán này tập trung ở việc nhận diện và xác định danh tính.

Phân tích khuôn mặt bằng deep learning giúp tăng cường tính chính xác và hiệu quả của các hệ thống nhận dạng khuôn mặt. Các mô hình học sâu như YOLO sẽ được trong bài toán này vì nó giúp tăng hiệu quả trong việc nhận diện các đặc trưng phức tạp từ ảnh khuôn mặt, giúp cải thiện đáng kể độ chính xác của việc phân tíchvà nó có thể độ chính xác nhất và thời gian training ngắn nhất.

## 1.2. Lý do chọn đề tài

Nhận diện khuôn mặt được ứng dụng rộng rãi trong nhiều lĩnh vực như an ninh, giám sát, y tế, nhân chủng học và thương mại điện tử, mang lại tiện ích và an toàn cho xã hội. Bài toán nhận diện khuôn mặt ở việc nhận diện danh tính. Đồ án này được thực hiện giúp nhận diện khuôn mặt một cách dễ dàng và nhanh chóng đáp ứng yêu cầu của xã hội hiện đại trong việc nâng cao an ninh và dịch vụ khách hàng.

**1.3. Mục tiêu đề tài.**

Mục tiêu của đề tài "Phân Tích Nhận Diện Khuôn Mặt Bằng Deep Learning" là nâng cao độ chính xác và hiệu quả của các mô hình nhận diện khuôn mặt. Đề tài nhằm giải quyết các khó khăn kỹ thuật như điều kiện ánh sáng kém và góc nhìn khác nhau, đóng góp vào sự phát triển của trí tuệ nhân tạo và học sâu. Đồng thời, đề tài hướng tới các ứng dụng thực tiễn trong tăng cường bảo mật và an toàn.

## 1.4. Phương pháp đề tài.

**1. Nghiên cứu các yếu tố ảnh hưởng đến bài toán nhận diện khuôn mặt**:

Nhận diện cơ bản cho yêu cầu nhận ra các yếu tố trên khuôn mặt của con người như:

* Khuôn mặt có mắt, mũi, miệng, môi
* Nếu có khi đối tượng cười, hay có nói thì sẽ có thêm nụ cười, hàm răng, lưỡi

Các yếu tố khác như:

* Góc độ máy ảnh
* Độ sáng của máy ảnh
* Độ to nhỏ của tiêu cự máy ảnh

**2. Phân tích kỹ thuật.**

Sử dụng Phát hiện khuôn mặt dựa trên học sâu, dùng mô hình YOLO để phát hiện và nhận diện khuôn mặt

**Tuyển chọn dữ liệu**: Dữ liệu là nền tảng để xây dựng mô hình phát hiện khuôn mặt. Dữ liệu cần được tuyển chọn kỹ lưỡng, đảm bảo tính đầy đủ, chính xác vẽ hình ảnh và gắn đúng nhãn. Nên lựa chọn bộ dữ liệu: **https://www.kaggle.com/datasets/sbaghbidi/human-faces-object-detection/data**

**Xây dựng mô hình**: Sử dụng model YOLO v8 để trainnign các hình ảnh khuôn mặt

**Đánh giá hiệu quả của mô hình**: Sau khi xây dựng mô hình nhân diên, cần phải đánh giá hiệu quả của mô hình bằng cách sử dụng dữ liệu thực tế.

## 1.5. Đối tượng và phạm vi nghiên cứu.

**Đối tượng nghiên cứu**

Đối tượng nghiên cứu là khuôn mặt của con người và ứng dụng nó cho an ninh, bảo mật, xác thực giao dịch trong kinh tế.

**Phạm vi nghiên cứu**

Phạm vi nghiên cứu của nhận diện khuôn mặt bao gồm:

* Bộ dữ liệu: tập chung vào bộ dữ liệu hình ảnh khuôn mặt con người đã xác định nhãn
* Ứng dụng: sữ dụng model train được ứng dụng vào thực tế camera trên máy tính thông qua thư viên OpenCV

# CHƯƠNG 2 ỨNG DỤNG VÀ THUẬT TOÁN

## 2.1 Giới thiệu bài toán.

Bài toán phát hiện khuôn mặt bằng Deep Learning xác định và phát hiện khuôn mặt từ ảnh, video hoặc camera. Phương pháp chính là sử dụng mạng nơ-ron tích chập (CNN) để trích xuất đặc trưng. YOLO giúp phát hiện nhanh và chính xác. Thách thức gồm điều kiện ánh sáng và tốc độ xử lý, nhưng Deep Learning mang lại nhiều tiến bộ nhờ khả năng học sâu và hiệu quả cao

**2.1.1 Thuật toán CNN**

Mạng nơ-ron tích chập (Convolutional Neural Network - CNN) là một loại mạng nơ-ron sâu được thiết kế đặc biệt để xử lý dữ liệu dạng lưới, chẳng hạn như hình ảnh. CNN mô phỏng cách thức hoạt động của hệ thần kinh thị giác của động vật, giúp tự động học và trích xuất các đặc trưng từ dữ liệu đầu vào.



Hình 1:Mô hình CNN

**Công Thức**

**1. Lớp Tích Chập (Convolutional Layer)**

Lớp tích chập thực hiện phép tích chập giữa bộ lọc (filter) và đầu vào.

**Công thức Tích Chập:**

(I∗K) (i, j) = ∑m∑n I(i+m, j+n) ⋅K(m, n)

* I: Ảnh đầu vào (input image)
* K: Bộ lọc (kernel/filter)
* i, j: Chỉ số vị trí trên bản đồ đầu ra (output feature map)
* m, n: Chỉ số vị trí trên bộ lọc

**2. Lớp Gộp (Pooling Layer)**

Lớp gộp giảm kích thước của bản đồ đặc trưng bằng cách lấy giá trị tối đa (max pooling) hoặc trung bình (average pooling) trên một vùng nhỏ của bản đồ đặc trưng.

**Công thức Gộp Tối Đa (Max Pooling):**

P (i, j) =max {I (x, y) ∣x∈Ri, y∈Rj}

* P (i, j): Giá trị sau khi gộp tại vị trí (i, j)
* Ri, Rj​: Vùng gộp tại vị trí (i, j)

**3. Lớp Kết Nối Đầy Đủ (Fully Connected Layer)**

Các lớp kết nối đầy đủ kết nối mọi nơ-ron trong lớp trước đó với mỗi nơ-ron trong lớp hiện tại.

**Công thức:**

y=f(W⋅x+b)

* W: Ma trận trọng số
* x: Đầu vào
* b: Hệ số bias
* f: Hàm kích hoạt (Activation function), chẳng hạn như ReLU hoặc Sigmoid

**Ứng Dụng thực tế**

**1. Thị Giác Máy Tính**

* Phân loại hình ảnh (Image Classification): Nhận diện đối tượng trong hình ảnh, ví dụ như xác định một hình ảnh chứa chó hay mèo.
* Nhận dạng đối tượng (Object Detection): Xác định và phân loại nhiều đối tượng trong một hình ảnh, ví dụ như phát hiện xe cộ và người đi bộ trên đường phố.
* Phân đoạn hình ảnh (Image Segmentation): Phân chia hình ảnh thành các vùng khác nhau tương ứng với các đối tượng khác nhau.

**2. Y Tế**

* Phân tích hình ảnh y khoa: Chẩn đoán bệnh từ ảnh chụp X-quang, MRI, CT scan, ví dụ như phát hiện khối u trong hình ảnh y tế.
* Phát hiện bệnh lý: Tự động phát hiện các dấu hiệu bệnh lý từ hình ảnh y khoa.

**3. Xe Tự Lái**

* Nhận diện làn đường: Xác định và theo dõi làn đường trên đường phố.
* Phát hiện chướng ngại vật: Nhận dạng và tránh các chướng ngại vật trên đường.

### 2.1.2 Tổng quan về mạng nơ-ron

Mạng nơ-ron nhân tạo (Artificial Neural Network - ANN) là một mô hình toán học được thiết kế để mô phỏng cách thức hoạt động của hệ thần kinh con người. Mạng nơ-ron bao gồm các đơn vị tính toán đơn giản, gọi là nơ-ron, được sắp xếp thành các lớp và liên kết với nhau thông qua các kết nối có trọng số. Mạng nơ-ron có khả năng học từ dữ liệu và tổng quát hóa từ các mẫu học để giải quyết các bài toán phức tạp.



Hình 2: Cấu trúc CNN

**Công Thức**

**1. Nơ-ron Đơn (Perceptron)**

Nơ-ron đơn, hay perceptron, là đơn vị cơ bản của một mạng nơ-ron. Nó nhận các đầu vào, nhân chúng với trọng số tương ứng, cộng thêm một giá trị bù (bias), và sau đó áp dụng một hàm kích hoạt để tạo ra đầu ra.

**Công thức Perceptron:**

* wi​: Trọng số của đầu vào thứ i
* xi​: Đầu vào thứ i
* b: Giá trị bù (bias)
* f: Hàm kích hoạt
* y: Đầu ra

**2. Lan Truyền Tiến (Forward Propagation)**

Trong quá trình lan truyền tiến, đầu vào được chuyển qua các lớp của mạng để tính toán đầu ra cuối cùng.

**Công thức Lan Truyền Tiến:**

* a(l): Đầu ra của lớp thứ l
* W(l): Ma trận trọng số của lớp thứ l
* b(l): Vector bias của lớp thứ l
* f: Hàm kích hoạt

**3. Lan Truyền Ngược (Backpropagation)**

Lan truyền ngược là thuật toán học của mạng nơ-ron, sử dụng để cập nhật các trọng số và giá trị bù dựa trên sai số giữa đầu ra dự đoán và đầu ra thực tế.

**Công thức Cập Nhật Trọng Số:**

* η: Tốc độ học (learning rate)
* L: Hàm mất mát (loss function)
* wij​: Trọng số giữa nơ-ron i và j

**Ứng Dụng**

**1. Phân Loại và Nhận Diện**

* Nhận diện hình ảnh: Phân loại đối tượng trong ảnh (ví dụ: nhận diện chữ viết tay, khuôn mặt).
* Nhận diện giọng nói: Chuyển đổi giọng nói thành văn bản.

**2. Dự Báo**

* Dự báo thời gian: Dự đoán giá cổ phiếu, nhu cầu năng lượng, thời tiết.
* Dự đoán xu hướng: Phân tích dữ liệu để dự báo xu hướng tiêu dùng, hành vi khách hàng.

**3. Xử Lý Ngôn Ngữ Tự Nhiên (NLP)**

* Dịch máy: Dịch văn bản từ ngôn ngữ này sang ngôn ngữ khác.
* Phân tích cảm xúc: Xác định cảm xúc từ văn bản (ví dụ: phân tích phản hồi khách hàng).

**4. Tự Động Hóa và Điều Khiển**

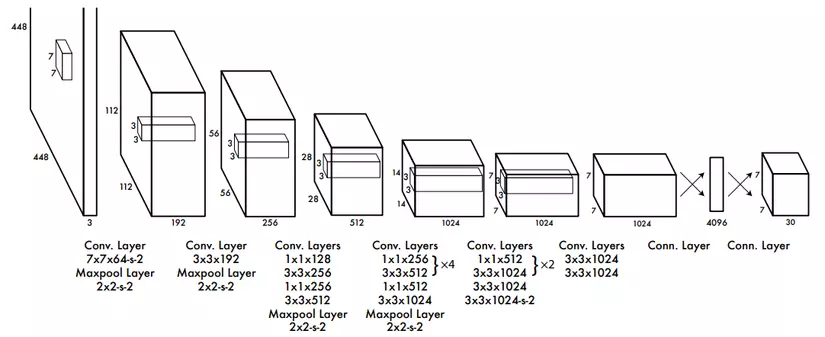
* Điều khiển robot: Lập trình robot thực hiện các nhiệm vụ phức tạp.
* Tự động hóa quá trình: Tối ưu hóa các quy trình sản xuất và quản lý.

**5. Y Tế**

* **Chẩn đoán bệnh:** Phân tích hình ảnh y khoa để chẩn đoán bệnh (ví dụ: phát hiện khối u từ ảnh MRI).
* **Dự đoán bệnh**: Sử dụng dữ liệu bệnh án để dự đoán nguy cơ mắc bệnh.

### 2.1.3 Tổng quan về YOLO.

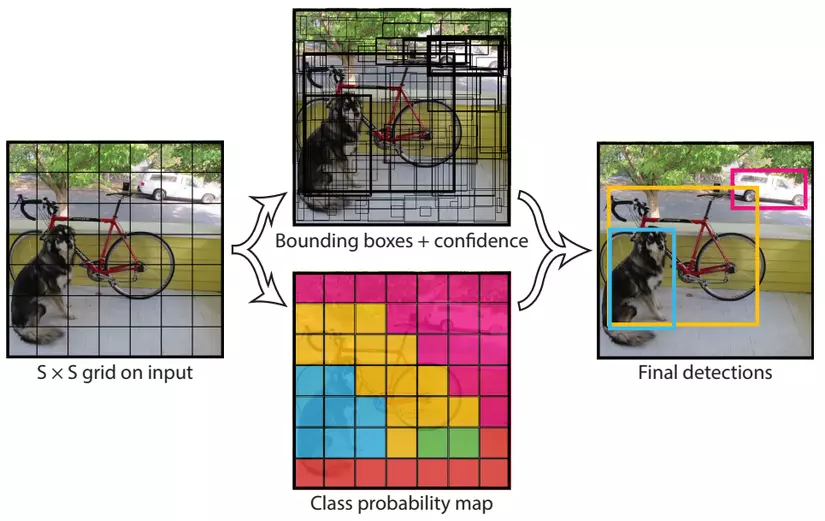
Yolo là một mô hình mạng CNN cho việc phát hiện, nhận dạng, phân loại đối tượng. Yolo được tạo ra từ việc kết hợp giữa các convolutional layers và connected layers.Trong đóp các convolutional layers sẽ trích xuất ra các feature của ảnh, còn full-connected layers sẽ dự đoán ra xác suất đó và tọa độ của đối tượng.



Hình 3: Cấu trúc YOLO

Cách YOLO hoạt động:

Đầu vào của mô hình là một ảnh, mô hình sẽ nhận dạng ảnh đó có đối tượng nào hay không, sau đó sẽ xác định tọa độ của đối tượng trong bức ảnh. ẢNh đầu vào được chia thành thành 𝑆×𝑆 ô thường thì sẽ là 3×3, 7×7, 9×9... việc chia ô này có ảnh hưởng tới việc mô hình phát hiện đối tượng, mình xin trình bày ở phần sau.



Hình 4: Cách thức nhận dạng YOLO

**Công Thức**

Localization Loss:

Trong đó,

* S2: Số lượng ô lưới.
* B: Số lượng hộp dự đoán trên mỗi ô.
* ​: Chỉ báo có giá trị 1 nếu ô iii chứa đối tượng.
* (xi​, yi​, wi​, hi​): Tọa độ và kích thước của hộp thực tế.
* ​): Tọa độ và kích thước của hộp dự đoán.

**Confidence Loss:**

Trong đó, C là điểm tin cậy dự đoán và thực tế, ​ là hệ số để giảm trọng số của các hộp không chứa đối tượng.

**Classification Loss:**

Trong đó, p(c) là xác suất của lớp dự đoán và thực tế.

Kết hợp cả ba thành phần này, hàm mất mát tổng thể của YOLO là:

Loss=Localization Loss+Confidence Loss+Classification Loss

**2.1.3 Tổng quan về OpenCV**

OpenCV (Open Source Computer Vision Library) là một thư viện mã nguồn mở được sử dụng rộng rãi trong lĩnh vực thị giác máy tính và xử lý hình ảnh. Thư viện này hỗ trợ nhiều ngôn ngữ lập trình và cung cấp nhiều công cụ mạnh mẽ để phân tích và xử lý hình ảnh.

**1. Xử Lý Ảnh Cơ Bản:**

* **Chuyển đổi không gian màu**: Chuyển đổi giữa các không gian màu khác nhau như RGB, HSV, Grayscale.
* **Biến đổi hình học**: Thay đổi kích thước, xoay, cắt và phóng đại ảnh.
* **Lọc ảnh**: Áp dụng các bộ lọc để làm mờ, tăng cường cạnh, loại bỏ nhiễu.

**2. Phát Hiện và Nhận Dạng Đối Tượng:**

* **Phát hiện cạnh**: Sử dụng các thuật toán như Canny, Sobel để phát hiện cạnh trong hình ảnh.
* **Phát hiện khuôn mặt**: Sử dụng mô hình Haar Cascade để phát hiện khuôn mặt.
* **Nhận dạng đối tượng**: Sử dụng các mô hình như HOG, SIFT, SURF để nhận dạng các đặc trưng trong hình ảnh.

**3. Xử Lý Video:**

* **Đọc và ghi video:** Đọc khung hình từ video hoặc camera và ghi lại video.
* **Phân tích khung hình:** Phân tích và xử lý từng khung hình trong video.
* **Theo dõi đối tượng:** Theo dõi chuyển động của các đối tượng trong video.

4. **Thị Giác Máy Tính Cao Cấp:**

* **Nhận diện và ghép đặc trưng:** Sử dụng các thuật toán như ORB, BRIEF để nhận diện và ghép đặc trưng.
* **Xây dựng panorama:** Ghép nhiều ảnh lại với nhau để tạo thành một ảnh toàn cảnh.
* **Phát hiện và nhận dạng văn bản:** Sử dụng OCR để nhận dạng văn bản trong hình ảnh.

**5. Học Máy và Học Sâu:**

* **Các thuật toán học máy**: Sử dụng các mô hình như KNN, SVM, Decision Trees để phân loại và nhận dạng.
* **Mô hình học sâu**: Tích hợp với các framework học sâu như TensorFlow, PyTorch để triển khai các mô hình AI tiên tiến.

# CHƯƠNG 3: THỰC NGHIỆM

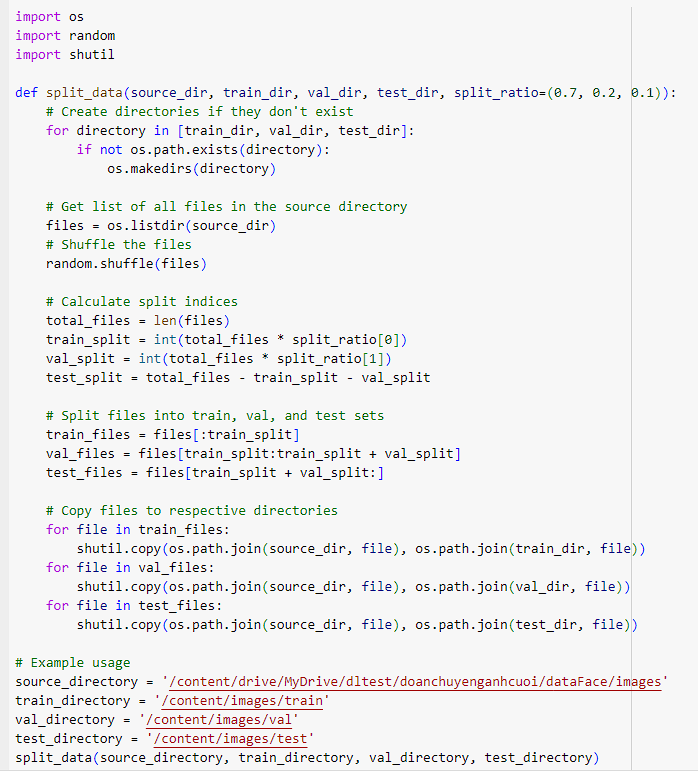
## 3.1 Công cụ sử dụng

Các công cụ dùng trong dự án này là Colab và Pycharm

Trong đó, Colab có sẳn T4GPU giúp giảm tải thời gian trainning của mô hình sử dụng. Pycharm sẽ được tải về các thư viện opencv-python tương tác với camera của máy tính để ứng dụng mô hình train

## 3.2. Xây dựng ứng dụng và giải thích

### 3.1.1 Chuẩn bị, xáo trộn vị trí file, chia tập tin

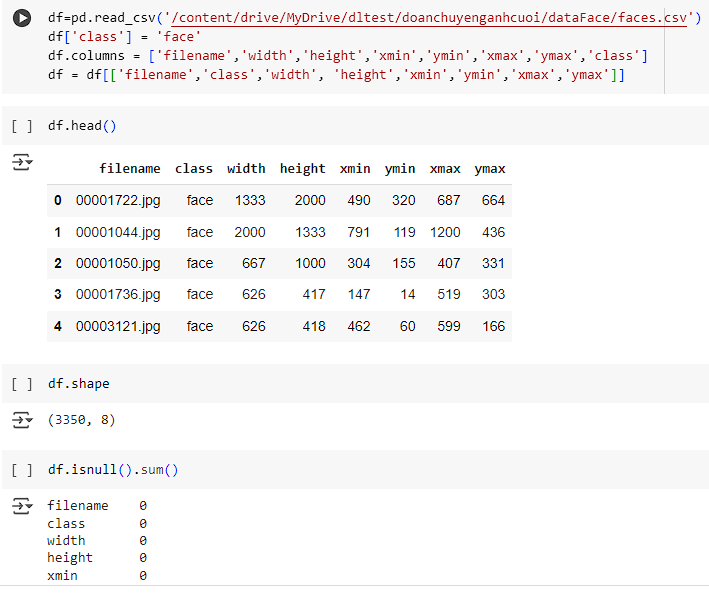


Hình 5: Phương thức split\_data

Xây dựng phương thức split\_data lấy đường dẫn của file chứa hình ảnh và chia các tập theo tỉ lệ train 70%, val 20%, test 10%. nó sẽ tạo thư mục chứa tập file train, test, val. n danh sách các tệp để đảm bảo các tệp được phân chia ngẫu nhiên và tính toán đúng tỉ lệ để chia tập file theo 70:20:10.

Dùng thư viện shutil.copy để copy tập file đã chia vào folder theo đường dẫn

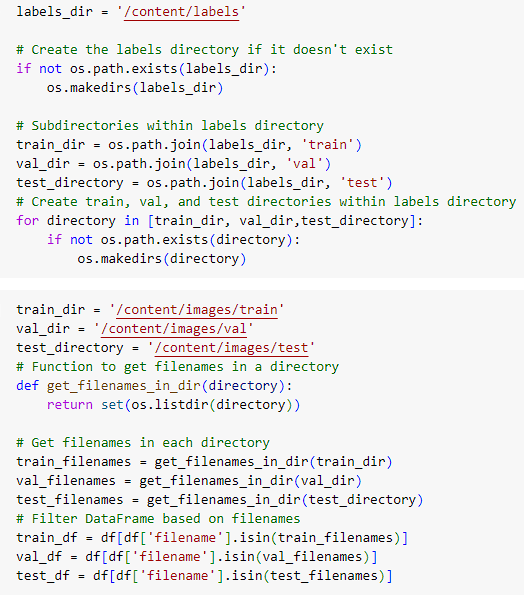
### 3.1.2 Đọc file csv để đọc các thông gắn nhãn của hình anh train



Hình 6: Làm sạch dữ liệu

Sử dụng thư viện pandas để đọc file csv của label hình ảnh và lấy các thông tin của csv như 5 dòng đầu, shape, tổng số dòng vị null

### 3.1.3 Lấy label cho các tập



Hình 7: Lấy nhãn cho hình ảnh

Tạo các thư mục để lưu nhãn của tập dữ liệu, sau đó lọc dữ liệu trong DataFrame dựa trên tên tệp trong các thư mục huấn luyện, kiểm tra, và kiểm thử.

labels\_dir: Đường dẫn đến thư mục chứa nhãn.

Kiểm tra sự tồn tại của labelsdir và tạo nếu chưa tồn tại tạo các đường dẫn cho các thư mục con train, val, test trong thư mục labelsdir.

Tạo các thư mục train, val, test trong labels\_dir nếu chúng chưa tồn tại.

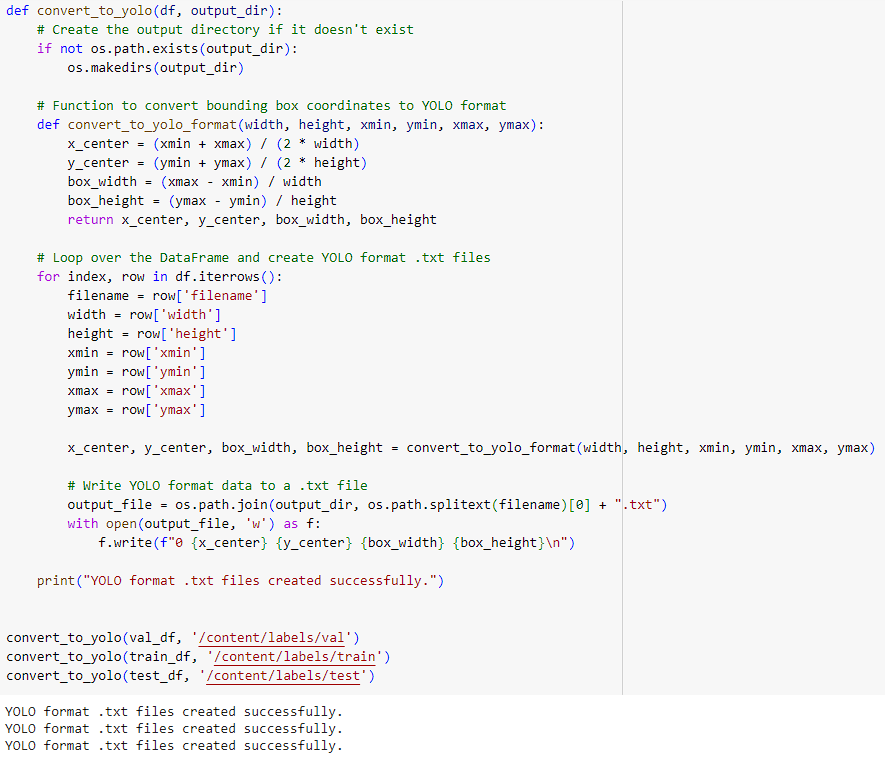
get\_filenames\_in\_dir(directory): Hàm này nhận đầu vào là một thư mục và trả về tập hợp các tên tệp trong thư mục đó.

train\_df: DataFrame chứa các hàng có tên tệp nằm trong train\_filenames.

val\_df: DataFrame chứa các hàng có tên tệp nằm trong val\_filenames.

test\_df: DataFrame chứa các hàng có tên tệp nằm trong test\_filenames.

### 3.1.4 Chuyển đổi label cho phù hợp với model



Hình 8: Phương thức convert\_to\_YOLO

Chuyển đổi các tọa độ hộp giới hạn từ định dạng của DataFrame sang định dạng YOLO và lưu chúng vào các tệp .txt tương ứng

**Tạo thư mục đích nếu chưa tồn tại**: Kiểm tra xem thư mục đích output\_dir có tồn tại hay không. Nếu chưa, tạo thư mục này.

**Hàm chuyển đổi tọa độ hộp giới hạn sang định dạng YOLO**: x\_center, y\_center: Tọa độ trung tâm của hộp giới hạn trong tỷ lệ so với chiều rộng và chiều cao của ảnh. box\_width, box\_height: Chiều rộng và chiều cao của hộp giới hạn trong tỷ lệ so với chiều rộng và chiều cao của ảnh.

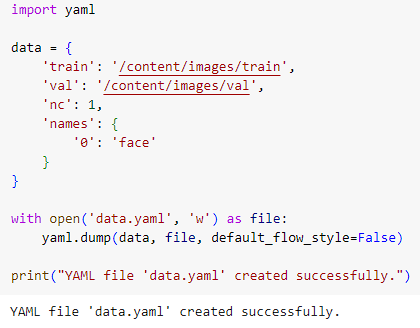
**Duyệt qua DataFrame và tạo các tệp .txt định dạng YOLO**:

Duyệt qua từng hàng trong DataFrame, lấy các giá trị cần thiết như filename, width, height, xmin, ymin, xmax, ymax.

Chuyển đổi các giá trị này sang định dạng YOLO bằng hàm convert\_to\_yolo\_format.

Ghi các giá trị này vào tệp .txt trong thư mục đích.

### 3.1.5 Tạo yaml



Hình 9: Tạo file YAML chứa hình ảnh

Tạo một tệp YAML chứa các đường dẫn đến các tập dữ liệu huấn luyện và kiểm tra, cùng với thông tin về số lượng lớp và tên các lớp

data: chứa các thông tin cấu hình cần thiết:

* 'train': Đường dẫn đến thư mục chứa các hình ảnh huấn luyện.
* 'val': Đường dẫn đến thư mục chứa các hình ảnh kiểm tra.
* 'nc': Số lượng lớp (ở đây là 1).
* 'names': Tên của các lớp, với '0' là lớp đầu tiên có tên là 'face'.

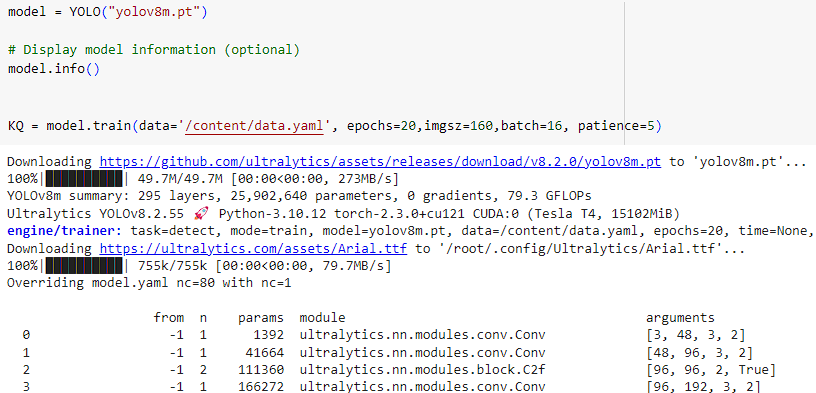
### 3.1.6 Tải thư viện YOLO



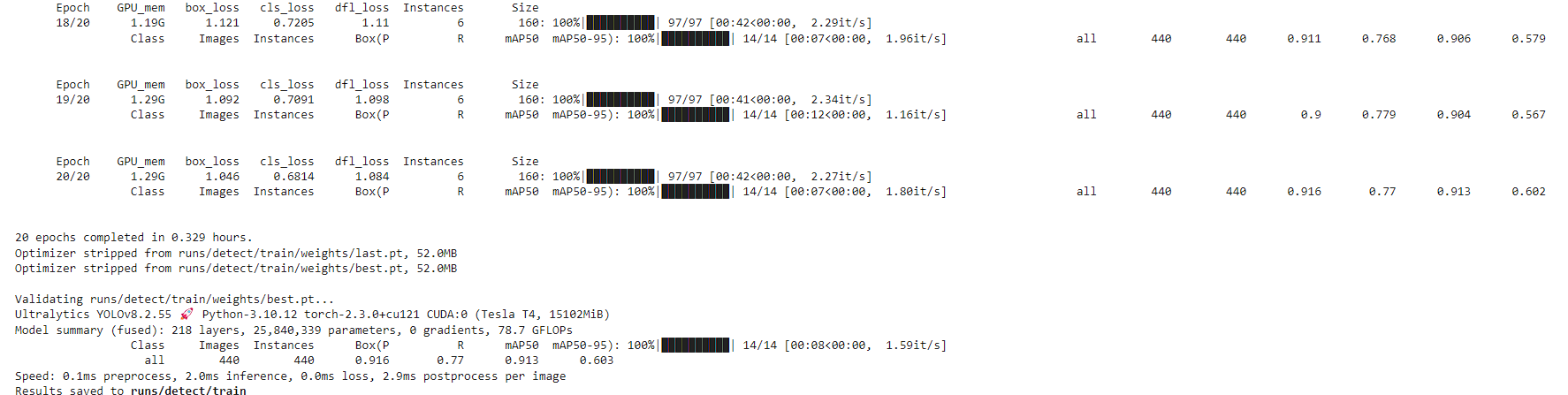
Hình 10: Thư viện YOLO

Cài đặt thư viện ultralytics và ipywidgets, sau đó kiểm tra cài đặt và sử dụng thư viện ultralytics để làm việc với mô hình YOLO. Kiểm tra cài đặt của thư viên ultralytics

### 3.1.6 Train model



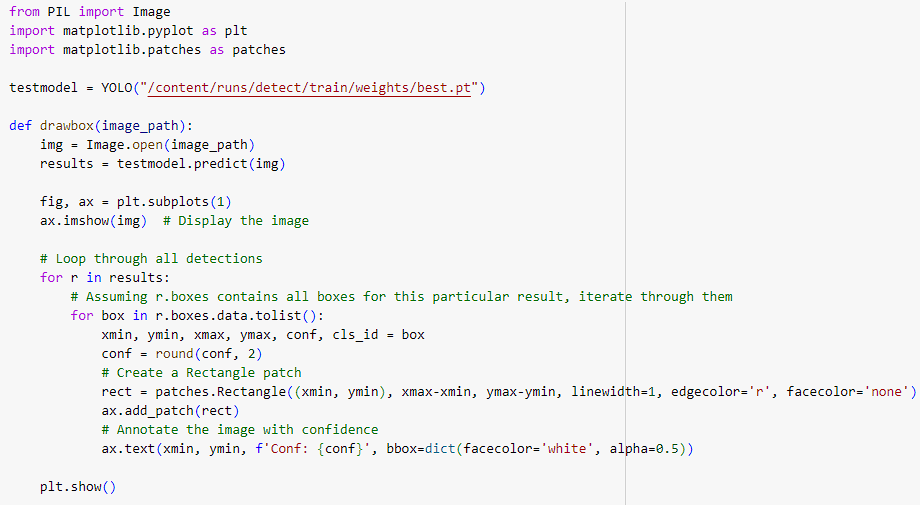
Hình 11: Thực hiện training model



Hình 12: Kết quả training

Sử dụng Yolo ver 8 để trainning với epochs = 20, imagesize =160, batch = 16, patience = 5

### 3.1.6 Kiểm tra tính chính xác



Hình 13: Phương thức drawbox

Sau khi train xong mô hình, ta ứng dụng bằng cách đưa 1 hình ảnh và dự đoán các hộp giới hạn trên một ảnh và hiển thị kết quả bằng cách vẽ các hộp này lên ảnh gốc.



Hình 14: Kết quả drawbox

### 3.1.7 Xuất model train

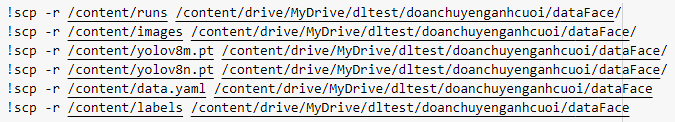
Xuất model theo các dạng để ứng dụng

1. Theo onnx



1. Theo tflite



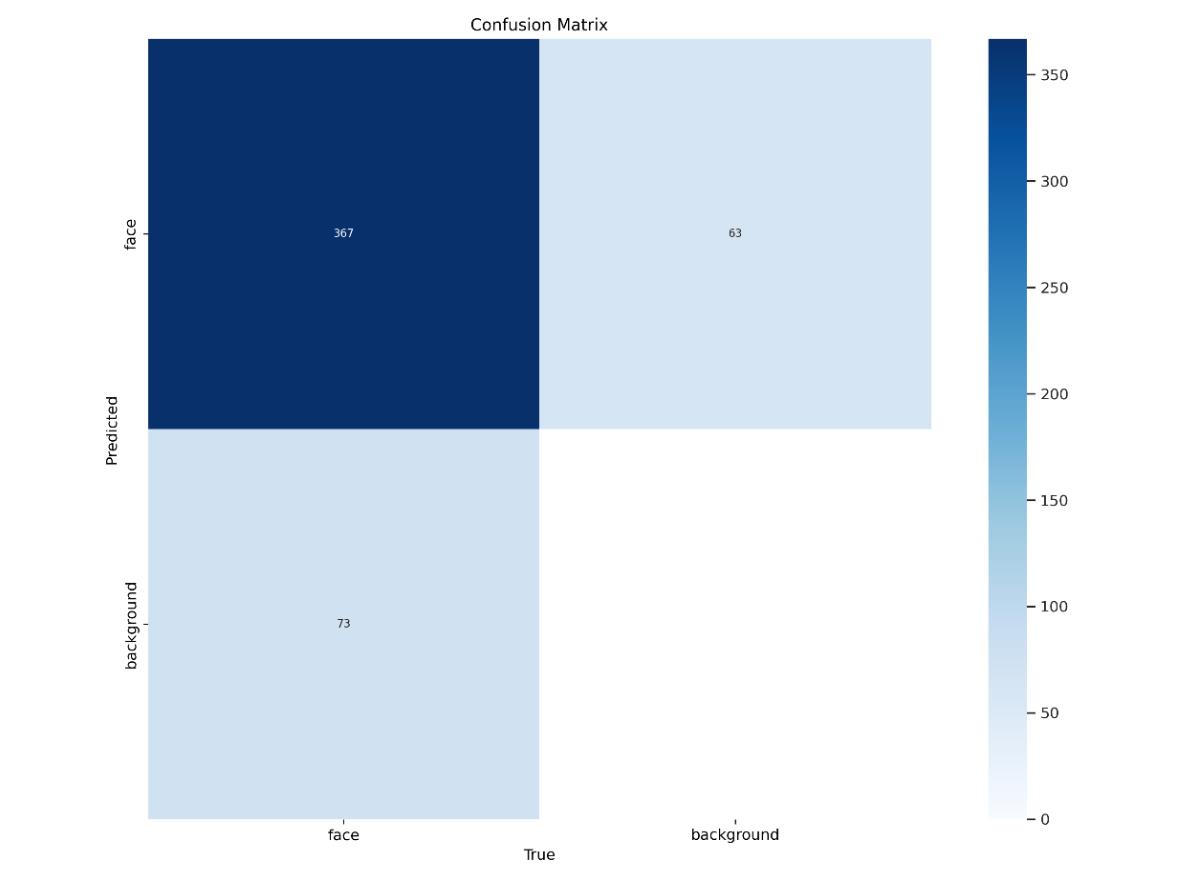


Hình 15: Truyền file train vào drive

Các lệnh bạn đưa ra sử dụng scp để sao chép thư mục và tệp từ Google Colab (/content/) sang Google Drive (/content/drive/MyDrive/dltest/doanchuyenganhcuoi/dataFace/) để lưu trữ mô hình vào drive.

## 3.3 Đánh giá hiệu suất

Sau khi hoàn thành huấn luyện, đánh giá mô hình trên tập kiểm thử để kiểm tra hiệu suất thực tế.



Hình 16: Ma trận nhầm lẫn và trực quan hóa

Ma trận nhầm lẫn là một bảng biểu giúp đánh giá hiệu suất của một mô hình phân loại học máy. Nó cho biết số lượng các trường hợp mà mô hình phân loại chính xác và số lượng các trường hợp mà mô hình phân loại sai.

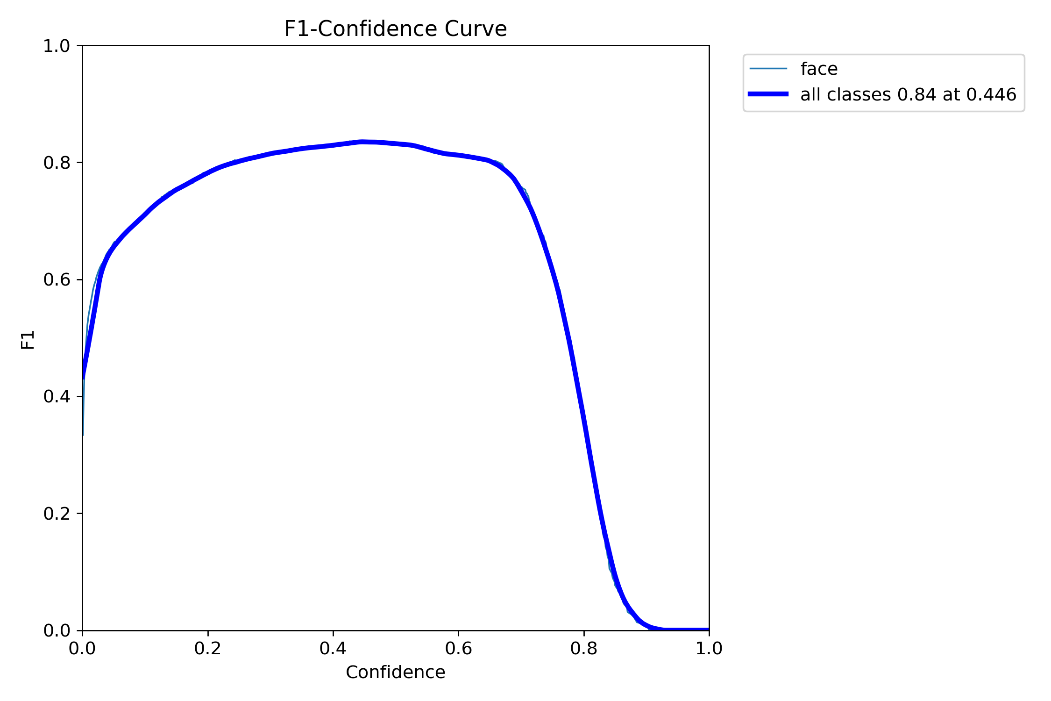
Độ chính xác = (TP + TN) / (TP + TN + FP + FN) = (367+0) / (367+0+73+63) = 0.729

Recall = TP / (TP + FN) = 367 / (367 + 63) = 0.855

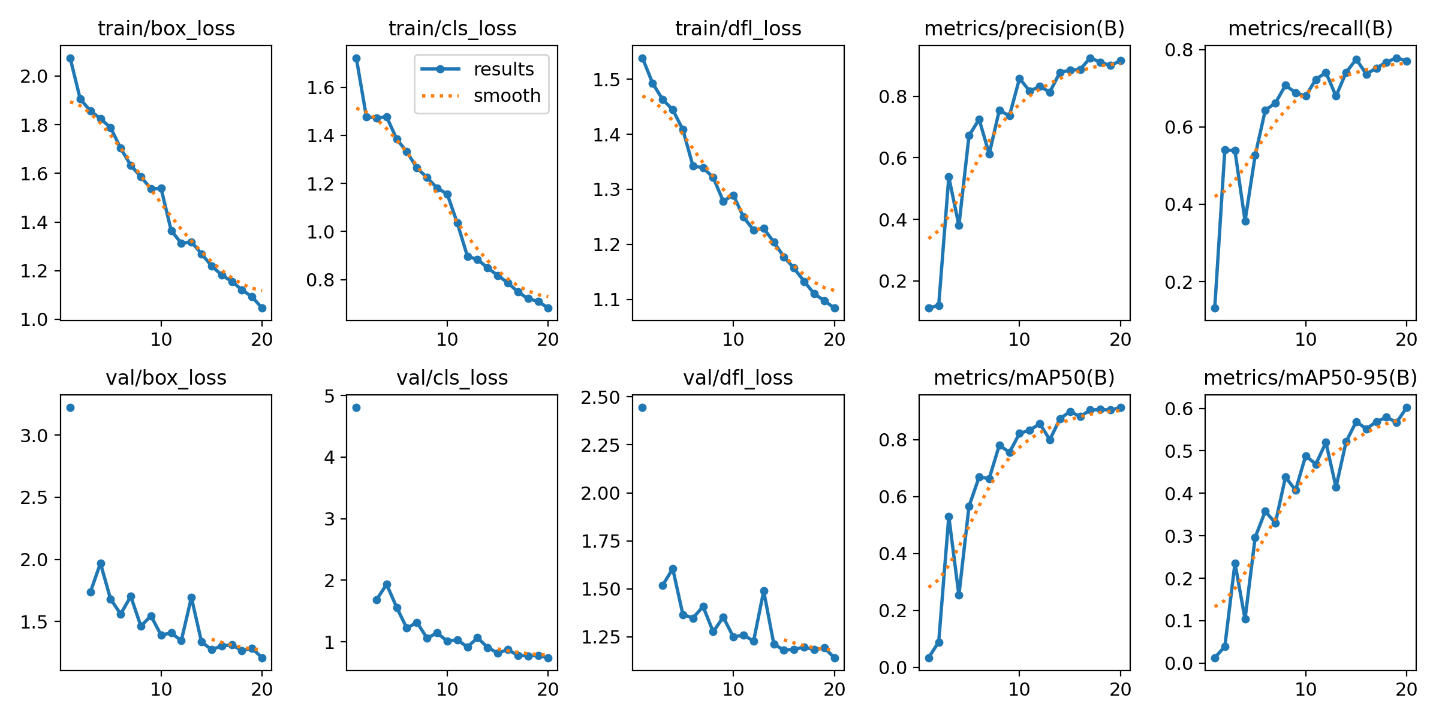
Precision = TN / (TN + FP) = 0 / (0 + 73) = 0

Tỷ lệ sai sót = (FP + FN) / (TP + TN + FP + FN) = (73 + 63) / (367 + 0 + 73 + 63) = 0.152

Mô hình có độ chính xác khá cao (84.8%), tuy nhiên độ đặc hiệu lại rất thấp (0%). Điều này có nghĩa là mô hình có xu hướng dự đoán sai các trường hợp âm tính là dương tính (73 trường hợp).



Hình 17: Độ tin cậy của f1

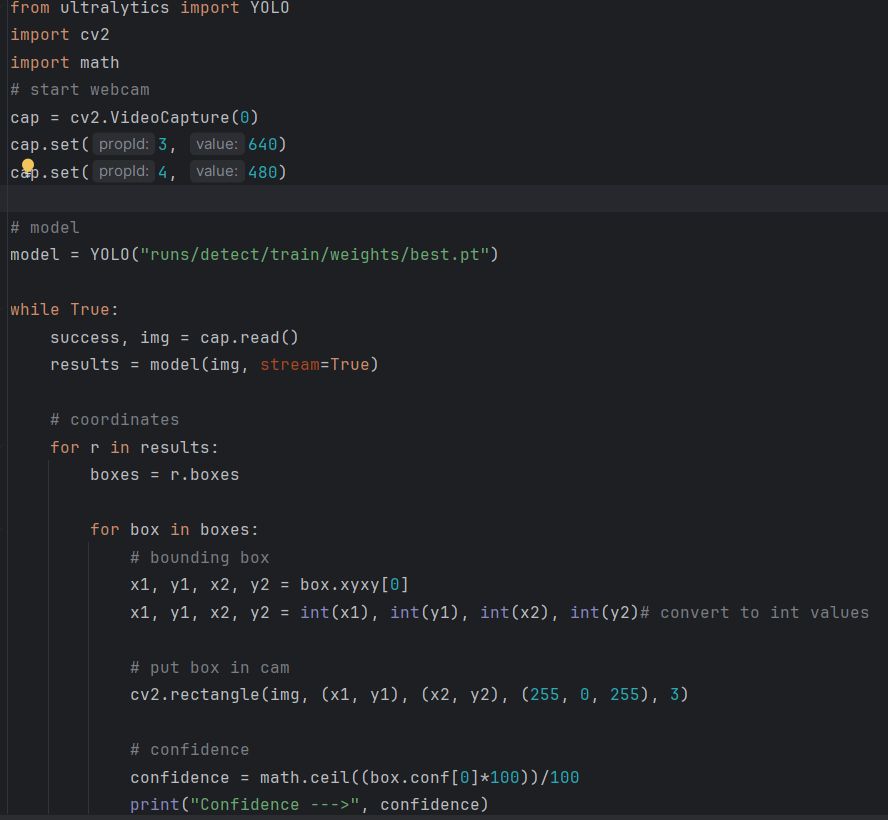


Hình 18: Tập hợp các biểu đồ học máy



Hình 19: Kết quả training

## 3.4 Ứng dụng mô hình vào camera của máy tính





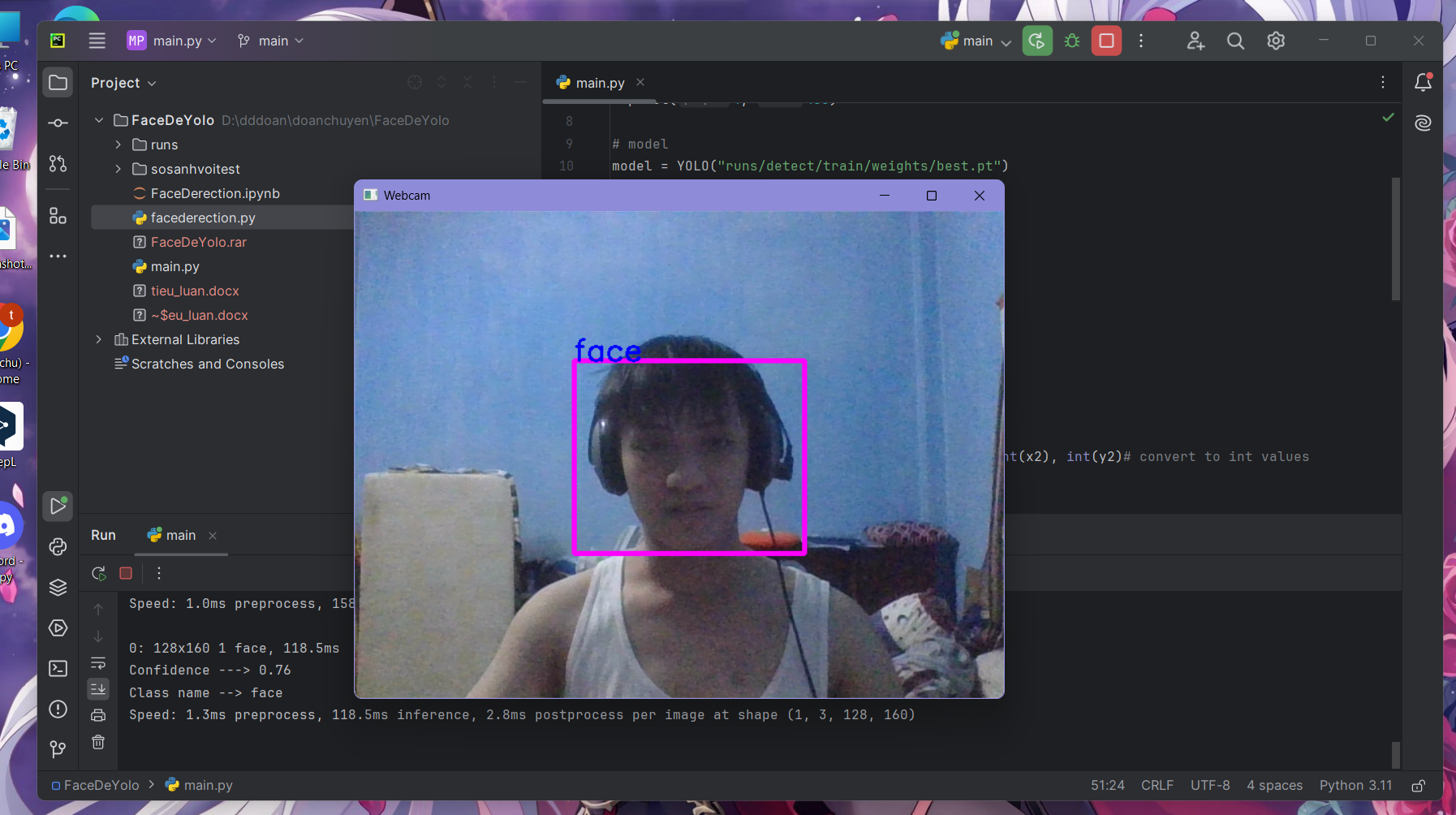
Hình 20: Phân biệt đối tượng trong webcam

Với Pycharm và các thư viện cài sẳn, nó sẽ truy cập vào camera và ứng dụng model vào hệ thống

Bằng cách sử dụng một framework web hoặc các công nghệ liên quan như

template engine và AJAX. Quá trình này giúp bạn xây dựng các ứng dụng web hiệu quả

và linh hoạt hơn.



Hình 21: Kết quả thử nghiệm bằng OPENCV

# KẾT LUẬN

## Kết quả đạt được.

Với xu hướng dân số tăng lên, việc bảo vệ an toàn là vấn đề cấp thiết.

Hệ thống này sử dụng các công nghệ tiên tiến như trí tuệ nhân tạo (AI), học sâu (Deep Learning) và cảm biến chuyển động để phát hiện khuôn mặt người. Các camera giám sát được lắp đặt trong nhà sẽ quan sát và ghi lại hình ảnh, video con người. Công nghệ AI sẽ phân tích các hình ảnh và video này để nhận diện khuôn mặt người.

Dựa trên các kết quả đánh giá từ biểu đồ sau khi huấn luyện với độ chính xác cao và độ hiệu quả từ việc thử nghiệm trên camera nên có thể ứng dụng.Tuy nhiên mô hình còn hạn chế khi gặp môi trường thiếu ánh sáng (cụ thể là ban đêm nếu không có ánh sáng) vì thế chương trình phụ thuộc khá nhiều vào độ rõ ràng của dữ liệu đầu vào.

## Hạn chế và hướng phát triển.

### Hạn chế:

* Không nhận diện khuôn mặt trong môi trường thiếu ánh sáng
* Cần phải có chất lượng camera tốt
* Không nhận diện được nhưng khuôn mặt từ xa

### Hướng phát triển:

* Tăng thêm nhiều dữ liệu khác như video, hình ảnh, tăng cường học
* Phát triển giao diện thân thiện với người dùng
* Nhận diện cảm xúc của khuôn mặt
* Ứng dụng lên đa nền tảng

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | N. T. Tuấn, DEEP LEARNING CƠ BẢN. |
| [2] | N. Đ. Tài, Tài liệu môn Mạng Neuron và giải thuật giải truyền. |
| [3] | geeksforgeeks, "Confusion Matrix in Machine Learning," 8 6 2024. [Online]. Available: https://www.geeksforgeeks.org/confusion-matrix-machine-learning/. [Accessed 8 6 2024]. |
| [4] | geeksforgeeks, "Object Detection with YOLO and OpenCV," [Online]. Available: https://www.geeksforgeeks.org/object-detection-with-yolo-and-opencv/?ref=header\_search. [Accessed 28 5 2024]. |
| [5] | OpenCV, "OpenCV modules," [Online]. Available: https://docs.opencv.org/4.x/index.html. |
| [6] | J. W. Drew Conway, "Machine Learning for Hackers," in *Machine Learning for Hackers*, 2012. |
| [7] | U. Inc., "Ultralytics YOLO Documents," YOLO v8, [Online]. Available: https://docs.ultralytics.com/vi. |
| [8] | B. T. Đức, Tài liệu giảng môn Học máy và Ứng dùng. |
| [9] | "Link code," [Online]. Available: https://github.com/olddragon381/Face\_detection\_with\_yolov8. |