

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC CẦN THƠ**



**BÁO CÁO TỔNG KẾT  
ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN**

**XÂY DỰNG ỨNG DỤNG HỖ TRỢ  
CHẨN ĐOÁN BỆNH DA LIỄU  
TRÊN GƯƠNG MẶT**

**Mã số đề tài: THS2024-65**

**Chủ nhiệm đề tài: Lê Tú Như**

**Cần Thơ, 10/2024**

## DANH MỤC THUẬT NGỮ

STT	Thuật ngữ	Ý nghĩa được sử dụng trong đề tài
1	C2f	Tầng Cross Stage Partial connections with Focus, giúp giảm số lượng tham số mà vẫn giữ nguyên thông tin đặc trưng.
2	Comedones	Mụn trứng cá không viêm
3	Concat	Tầng nối (hoặc Concatenate)
4	Conv	Tầng tích chập (hoặc tầng Convolutional)
5	F1-Score	Điểm F1, là điểm đánh giá kết hợp hai chỉ số độ chính xác và độ nhạy
6	ReLU	Hàm kích hoạt sẽ trả về giá trị đầu vào nếu dương và 0 nếu âm, giúp tăng tốc độ học và giảm hiện tượng mất dần gradient.
7	SPPF	Tầng Spatial Pyramid Pooling – Fast, nơi mỗi neuron kết nối với tất cả các neuron ở tầng trước, dùng trong các bước phân loại cuối.
8	Upsample	Tăng mẫu (hoặc Upsampling)

Thông qua việc nghiên cứu đề tài này, nhóm sinh viên mong muốn học tập, nắm bắt được những kiến thức trong việc xây dựng và phát triển hệ thống, cách huấn luyện mô hình nhận diện và phân loại, cũng như các kỹ thuật liên quan đến xử lý dữ liệu và máy học. Nhóm nghiên cứu cũng hi vọng rằng đề tài này có thể trở thành một tài nguyên tham khảo có giá trị đối với những nhóm nghiên cứu khác và đóng góp đôi phần vào sự phát triển của lĩnh vực công nghệ và chăm sóc sức khỏe nói chung.

## **4. PHƯƠNG PHÁP NGHIÊN CỨU**

### **4. 1. Phương pháp nghiên cứu lý thuyết**

Về mặt lý thuyết, nhóm nghiên cứu bắt đầu bằng việc tìm hiểu chi tiết về năm loại mụn đã xác định, bao gồm nguyên nhân gây ra và phương pháp điều trị hiệu quả. Song song với đó, nhóm cũng tìm hiểu, nghiên cứu các bài báo và công trình nghiên cứu tương tự để đảm bảo các nội dung về lĩnh vực y tế và chăm sóc da được sử dụng trên đề tài là phù hợp. Sau đó nhóm tập trung vào các thuật toán máy học và học sâu, đặc biệt là mạng nơ-ron tích chập CNN, để có thể áp dụng vào việc nhận diện và phân loại mụn. Nhóm nghiên cứu cũng xác định cần phân tích các yêu cầu của đề tài, xác định các chức năng cần thiết và thiết kế kiến trúc tổng thể của ứng dụng.

### **4. 2. Phương pháp nghiên cứu thực tiễn**

Trong phần nghiên cứu thực tiễn, nhóm sinh viên thu thập hình ảnh về da mặt và mụn từ các nguồn dữ liệu đáng tin cậy. Việc sử dụng nhiều nguồn dữ liệu đòi hỏi việc tiền xử lý các hình ảnh này để chọn ra những hình ảnh phù hợp nhất với yêu cầu của bài toán. Sau khi tập dữ liệu đã đạt được mức độ hài lòng nhất định, nhóm huấn luyện các mô hình YOLOv8 và ResNet50 để nhận diện và phân loại mụn. Quá trình huấn luyện được thực hiện trên tập dữ liệu đã qua xử lý, từ đó làm cơ sở để đánh giá hiệu suất của mô hình dựa trên các chỉ số quan trọng. Dựa trên kết quả đánh giá, nhóm tinh chỉnh các tham số lần phát triển tập dữ liệu để đạt được mô hình có hiệu suất tốt nhất.

#### 4. 3. Phương pháp nghiên cứu ứng dụng

Để phát triển ứng dụng, nhóm nghiên cứu thiết lập server đám mây, nơi tích hợp mô hình nhận diện và phân loại mụn, đồng thời cung cấp các gợi ý điều trị dựa trên kết quả phân loại. Tiếp theo, nhóm sinh viên tiến hành xây dựng, cài đặt các chức năng cần thiết cũng như thiết kế giao diện thân thiện cho người dùng. Nhóm sinh viên sau đó xây dựng API để ứng dụng có thể gửi ảnh lên server và nhận kết quả phân tích. Ở giai đoạn cuối, nhóm tiến hành kiểm thử các chức năng và hiệu suất của cả ứng dụng lẫn server để đảm bảo hệ thống hoạt động ổn định, hiệu quả và đúng với những dự định của nhóm.

#### 5. NHỮNG ĐÓNG GÓP CỦA ĐỀ TÀI

Hiện nay, các hệ thống nhận diện và phân loại các bệnh về da đã và đang xuất hiện và dần dần được hoàn chỉnh hơn. Ngoài độ chính xác của các dự đoán được đưa ra của hệ thống ngày một nâng cao, thì trải nghiệm người dùng lại ít được chú ý hơn. Nhóm nghiên cứu tin rằng một vấn đề cần được chú ý nhiều hơn chính là việc các hệ thống có thể cung cấp cho người dùng dự đoán về bệnh và cả hướng điều trị thích hợp, mà không bị ảnh hưởng bởi các công ty mỹ phẩm, dược phẩm. Để có hướng điều trị thì người dùng đa phần cần phải tìm gặp các chuyên gia và cung cấp thông tin về thể trạng của bản thân, để các chuyên gia dựa vào đó mà đưa ra những phương pháp điều trị. Do đó, điểm cải tiến của đề tài nghiên cứu là có khả năng cung cấp thêm cho người sử dụng thêm phương pháp điều trị phù hợp bằng việc tăng cường tính cá nhân hoá, thông qua việc cung cấp thêm những thông tin cá nhân về thể trạng, lối sống, sinh hoạt hằng ngày,... từ đó đưa ra hướng điều trị thích hợp cho từng loại bệnh đến người dùng.

Dựa vào thành quả của đề tài trên, nhóm nghiên cứu tin rằng sản phẩm của đề tài sẽ có các tiềm năng thực tiễn đáng nhắc đến. Đầu tiên, hệ thống có thể được sử dụng để chẩn đoán các bệnh về da và đưa ra hỗ trợ điều trị cho những người dùng ở vùng sâu vùng xa hoặc khó tiếp cận dịch vụ y tế. Điều này sẽ có thể hỗ trợ người sử dụng biết được tình trạng của bệnh một cách thuận tiện hơn. Song song, ứng dụng cũng sẽ phần nào giảm tải được sự thiếu hụt về cơ sở vật chất, các chuyên gia, bác sĩ da liễu, nhờ đó cũng hạn chế được tình trạng quá tải tại các bệnh viện da liễu và các trung tâm chăm sóc

da. Cuối cùng, đề tài này, bằng việc vận dụng và kết hợp những điểm ưu việt của hai mô hình YOLOv8 và ResNet50 sẽ là cơ sở cho các nghiên cứu tương tự.

## **6. BỐ CỤC ĐỀ TÀI**

Nội dung của đề tài nghiên cứu về cơ bản được chia thành năm chương:

Chương I: Phần giới thiệu: Chương này đưa ra một bản tóm tắt về nhận định của nhóm về đề tài, sự nghiên cứu tình hình trong và ngoài nước, cũng như các nội dung như mục tiêu và phương pháp nghiên cứu của đề tài.

Chương II: Cơ sở lý thuyết: Chương này trình bày các thông tin cơ bản về những lý thuyết được nhóm nghiên cứu dựa vào để thực hiện đề tài. Nhóm sinh viên tìm hiểu lý thuyết về hai mô hình học sâu nhằm triển khai một mô hình phù hợp và tối ưu cho bài toán. Ngoài ra, chương này cũng đề cập đến các tìm hiểu của nhóm về các loại mụn trong phạm vi nghiên cứu.

Chương III: Xây dựng và kết hợp hai mô hình: Nội dung chương này bao gồm trình bày các bước liên quan đến việc thu thập thông tin, chuẩn bị tập dữ liệu và cách xây dựng, triển khai mô hình của đề tài. Nhóm sinh viên mô tả các quy trình sử dụng các thước đo hiệu suất để đào tạo, đánh giá và điều chỉnh nhằm đạt được một mô hình phù hợp và tối ưu.

Chương IV: Xây dựng server và ứng dụng chẩn đoán: Việc tạo ra server đám mây và thiết kế, cài đặt ứng dụng chẩn đoán là chủ đề chính của chương này. Nhóm sẽ trình bày cách xây dựng API để ứng dụng có thể gửi hình ảnh đến máy chủ và nhận kết quả phân tích, cũng như phản hồi cùng gợi ý điều trị. Chương này cũng bao gồm các kết quả đánh giá và kiểm thử cho ứng dụng.

Chương V: Kết luận và đánh giá: Cuối cùng, nhóm nghiên cứu sẽ tổng hợp các kết quả đã thực hiện được trong chương này và đưa ra các nhận định khách quan về đề tài. Nhóm sinh viên cũng sẽ nêu những hạn chế hiện tại và đưa ra khuyến nghị đối với hướng phát triển trong tương lai.

## CHƯƠNG II: CƠ SỞ LÝ THUYẾT

### 1. TỔNG QUAN VỀ MÔ HÌNH YOLO

#### 1. 1. Ý tưởng triển khai

Các mô hình học sâu ngày nay đã phát triển khả năng học đến mức độ cao cấp, phục vụ nhiệm vụ nhận diện các đặc điểm cho nhiều bài toán khác nhau. Nhìn chung, các mô hình có xu hướng sử dụng nhiều thuật toán trong các giai đoạn được chia sẵn. Cho đến năm 2016 tại CVPR, mô hình YOLO của Joseph Redmon và cộng sự được công bố, giới thiệu lần đầu tiên một giải pháp học sâu với hiệu quả đáng kinh ngạc.

YOLO, hay còn được viết tắt cho cụm từ “You Only Look Once” – “Bạn Chỉ Nhìn Một Lần” thể hiện khả năng phát hiện đối tượng chỉ trong một giai đoạn để xử lý hình ảnh một lần duy nhất và đưa ra các kết quả tương quan. Tác giả phát minh ra mô hình này để khắc phục các cách tiếp cận trước đó của bài toán phát hiện đối tượng, đồng thời mô phỏng quá trình nhận diện vật thể của con người.

Con người chỉ cần nhìn lướt qua một hình ảnh là có thể ngay lập tức nhận biết được các đối tượng, vị trí và cách chúng tương tác lẫn nhau. Điều này thể hiện rõ trong hệ thống thị giác của con người có thể phản ứng với tốc độ cao, cho phép việc thực hiện các tác vụ phức tạp như điều khiển phương tiện mà không cần để tâm quá nhiều. Nếu có thể “ứng dụng” khả năng này của con người vào mô hình máy học, điều này sẽ tạo điều kiện cho các mô hình học sâu truyền đạt thông tin cảnh quan theo thời gian thực cho người dùng nhanh hơn và lược bỏ sự cần thiết của những cảm biến chuyên dụng.

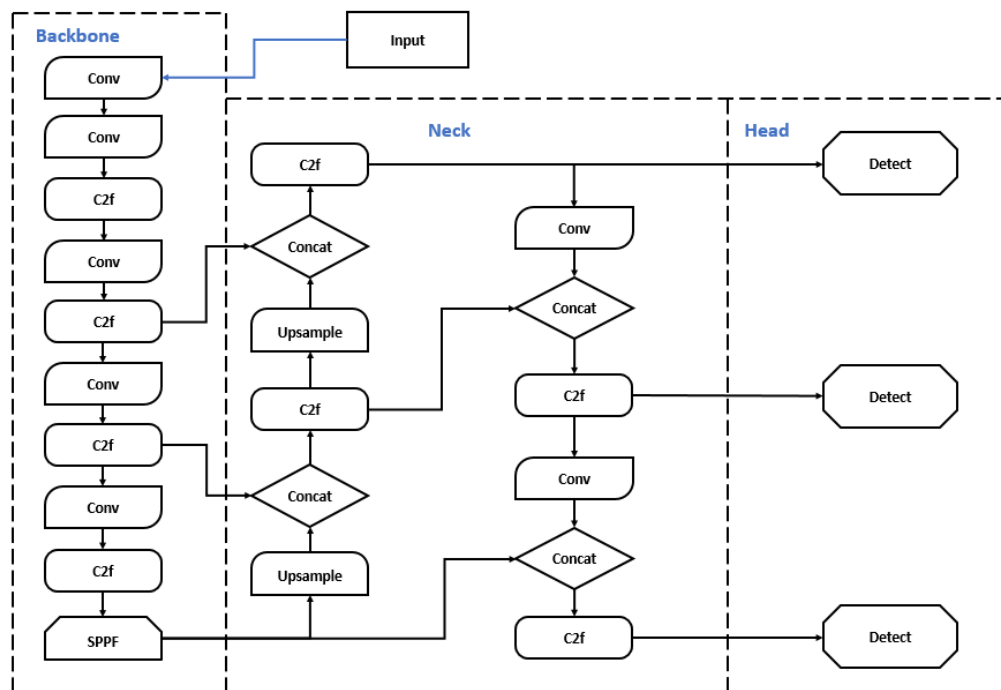
Nhóm nghiên cứu đại diện bởi Joseph Redmon đã thành công ứng dụng điều này vào YOLO, với sự nhận định rằng YOLO có “khả năng suy nghĩ toàn cục đối với toàn hình ảnh, cũng như tất cả các đối tượng được xuất hiện”. Đầu tiên, hình ảnh đầu vào sẽ được chia thành một lưới  $S \times S$ . Nếu điểm trung tâm của một đối tượng nằm trong một ô lưới, ô lưới đó sẽ chịu trách nhiệm phát hiện đối tượng đó. Lúc này, mỗi ô trong lưới sẽ dự đoán bounding boxes  $B$ , bao gồm các giá trị: tọa độ  $(x, y)$  đại diện cho trung tâm của khung so với giới hạn của ô lưới;  $w$  và  $h$  là chiều rộng và chiều cao của đối tượng so với

hình; cuối cùng là điểm tin cậy phản ánh mức độ tin tưởng của mô hình đối với khả năng chứa đối tượng và độ chính xác của khung [13].

Dựa trên ý tưởng triển khai này, YOLO trở thành một giải pháp tối ưu và hiệu quả đối với những bài toán đòi hỏi tốc độ trong phát hiện đối tượng. Việc hợp nhất các thành phần riêng biệt của việc phát hiện đối tượng vào trong một mạng nơ-ron duy nhất chính là điểm sáng nhất của mô hình. Bằng cách này, YOLO có khả năng sử dụng các đặc trưng từ toàn bộ hình ảnh để dự đoán từng khung bao, đồng thời dự đoán tất cả các khung bao cho tất cả các lớp đối tượng trong một hình ảnh cùng một lúc. YOLO không chỉ thể hiện được khả năng của mình khi được công nhận trong các cuộc thi quốc tế như COCO (Common Objects in Context) và PASCAL VOC (Visual Object Classes), mà còn được đánh giá cao bởi các chuyên gia trong lĩnh vực máy học. Biến thể YOLOv8 được lựa chọn sử dụng trong đề tài này vì sự tương thích về phần cứng và độ phù hợp với tập dữ liệu.

## 1. 2. Khái quát về cách hoạt động

Về cơ bản, YOLO được chia thành ba thành phần chính: backbone, neck, và head.



Hình 1. Sơ lược cấu trúc của mô hình YOLO.

Hình ảnh khi được đưa vào mô hình sẽ được bộ phận backbone xử lý. Backbone là một mạng nơ-ron tích chập, có nhiệm vụ trích xuất đặc trưng của hình. Tại đây sử dụng các lớp như sau:

- Conv: Các lớp Conv được sử dụng để trích xuất các đặc trưng từ dữ liệu đầu vào bằng cách áp dụng phép tích chập. Mỗi lớp Conv có nhiệm vụ biến đổi thông tin từ các lớp trước đó để tạo ra biểu diễn các đặc trưng cấp cao hơn.
- C2f: Đây là lớp sẽ chuyển đổi đầu ra từ các lớp Conv trước đó thành đầu vào cho các lớp fully connected.
- SPPF: SPPF là lớp pooling được thực hiện trên các tỷ lệ không gian khác nhau, kết hợp với các phép tích chập để tăng khả năng biểu diễn không gian của mô hình.

Sau khi thu được output là feature maps từ backbone, các output này sẽ được kết hợp và xử lý tại neck, cụ thể sẽ sử dụng kỹ thuật PANet (Path Aggregation Network) đối với YOLOv8. PANet tạo điều kiện để mô hình nắm bắt các đặc trưng trong nhiều tỉ lệ một cách hiệu quả bằng việc hỗ trợ luồng thông tin trên các độ phân giải không gian khác nhau. Neck sử dụng thêm hai lớp Concat và Upsample bên cạnh hai lớp Conv và C2f:

- Concat: Lớp này lấy feature maps và kết hợp chúng lại với nhau theo channel-wise. Điều này giúp mô hình tối ưu thông tin chi tiết từ các lớp trước đó, từ đó cải thiện khả năng nhận diện.
- Upsample: Lớp này đóng vai trò đồng bộ các feature maps về mặt kích thước. Điều này sẽ giúp việc duy trì thông tin qua các lần thay đổi kích thước, góp phần trong hiệu suất kết hợp của những lớp khác.

Bộ phận cuối cùng của mô hình là head sẽ chịu trách nhiệm đưa ra dự đoán. YOLOv8 sử dụng nhiều mô hình phát hiện để dự đoán về class, bounding boxes và điểm tin cậy cho mỗi ô lưới trong feature maps. Những dự đoán này sau đó được tổng hợp để có được các phát hiện cuối cùng.



Như vậy, có thể tóm gọn cách hoạt động của mô hình YOLOv8 như sau: Tại backbone sẽ trích xuất feature maps từ nông đến sâu. Các output này sẽ được kết hợp, tối ưu hóa tại neck nhằm hỗ trợ mô hình có cái nhìn vừa tổng quát vừa chi tiết về hình ảnh. Cuối cùng, head sẽ là bộ phận chuyển đổi feature maps từ neck thành các dự đoán cụ thể về các đối tượng trong hình ảnh.

### **1. 3. Ưu điểm và nhược điểm**

YOLO khi ra mắt không chỉ gặt hái được nhiều thành công, mà đến thời điểm hiện tại đã không ngừng phát triển và trở thành một trong những dòng mô hình có độ tin cậy cao. Mô hình đã nhận được sự đánh giá cao và phản hồi tích cực từ cộng đồng nghiên cứu và các nhà phát triển trong lĩnh vực thị giác máy tính, chứng tỏ được những ưu điểm nổi trội và to lớn của YOLO.

Thành công lớn nhất của YOLO có thể nói là việc phổ biến hóa việc kết hợp việc dự đoán vị trí và xác suất của các đối tượng trong một bức ảnh chỉ bằng một lần thực thi mạng nơ-ron, sử dụng phương pháp tách biệt so với những loại mô hình đi trước. Khả năng phát hiện các đối tượng trong một bức ảnh toàn cảnh một cách hiệu quả, kết hợp việc có tốc độ xử lý ảnh cao, YOLO là sự lựa chọn tối ưu trong đa số những vấn đề cần phát hiện đối tượng.

Mặt khác, YOLO vẫn tồn tại những yếu điểm cần phải cân nhắc. Kiến trúc của YOLO được xây dựng để đạt mức độ tương thích cao cho các bài toán phát hiện đối tượng, đồng nghĩa với việc YOLO sẽ cần học cả đặc trưng và vị trí của các đối tượng trong toàn bộ bố cục của hình ảnh. Như vậy, nếu ứng dụng YOLO vào các vấn đề cần phân loại đối tượng sẽ gây lãng phí tài nguyên, đồng thời khả năng học đặc trưng của YOLO sẽ không tốt bằng những mô hình khác có mức độ phù hợp cao hơn.

Không thể phủ nhận những điểm cộng sáng giá của YOLO đối với lĩnh vực máy học, nhưng nếu chỉ sử dụng YOLO cho bài toán phân loại các loại mìn có những đặc điểm tương đồng nhau sẽ ảnh hưởng đến kết quả dự đoán cuối cùng. Vì lý do này, một mô

hình khác đã được đưa vào bài toán nhằm bổ trợ nhược điểm này của YOLO, đó chính là mô hình ResNet.

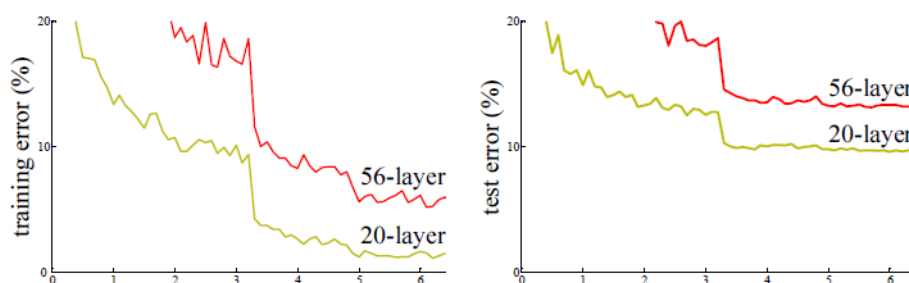
## 2. TỔNG QUAN VỀ MÔ HÌNH RESNET

### 2.1. Ý tưởng triển khai

Để tối ưu hóa việc trích xuất đặc điểm của hình ảnh qua các lớp, những mô hình tiên tiến ngày nay sở hữu một lượng lớn các lớp nhằm đáp ứng kì vọng. Việc này dẫn đến một vấn đề quan trọng: Khả năng dự đoán có tỉ lệ thuận với số lượng lớp của mô hình hay không?

Để trả lời câu hỏi trên, nhiều công trình đã được thực hiện để xác định mối quan hệ đóng vai trò quyết định này. [14] khẳng định “việc tăng số lượng lớp dẫn đến tăng lỗi huấn luyện là nguyên nhân gây ra sự suy giảm này, chứ không phải do overfitting”. Nói cách khác, thêm số lượng lớp vào mô hình nhưng không thể tối ưu hóa quá trình huấn luyện sẽ khiến lỗi huấn luyện cao hơn, dẫn đến khả năng dự đoán sẽ kém hơn.

Bài báo cáo [15] đã kiểm chứng bằng thực nghiệm đối với vấn đề này. Dựa trên lý thuyết, các nhà nghiên cứu nhận định rằng, mô hình học sâu sâu có một giải pháp theo cấu trúc: các tầng sâu hơn là ánh xạ định danh, trong khi các tầng còn lại là các bản sao của mô hình nông hơn đã học được. Theo giải pháp này, một mô hình sâu không nên cho kết quả lỗi huấn luyện lớn hơn so với một mô hình nông hơn. Tuy nhiên, việc kiểm tra thực nghiệm lại xác định rằng lý thuyết này không đúng với mô hình học sâu.



Hình 2. Lỗi huấn luyện (bên trái) và lỗi kiểm tra (bên phải) trên bộ dữ liệu CIFAR-10 với cấu trúc mạng đơn giản gồm 20 lớp và 56 lớp. Mạng sâu hơn có lỗi huấn luyện cao hơn, và do đó lỗi kiểm tra cũng cao hơn. (Nguồn: [15])

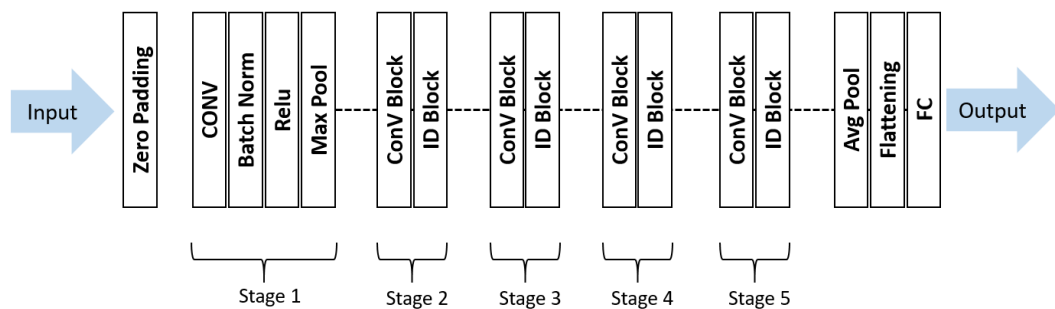
Để giải quyết vấn đề suy giảm trong mô hình học sâu, các nhà nghiên cứu đã đề xuất giải pháp “Deep residual learning framework”, được đề cập lần đầu trong [15].

Gọi  $H(x)$  là một ánh xạ cơ bản cần được xây dựng bởi một số lớp xếp chồng lên nhau, với  $x$  là đầu vào của lớp đầu tiên. [15] lập luận, giả thiết rằng nhiều lớp phi tuyến có thể tiệm cận xấp xỉ các hàm phức tạp, thì tương đương với việc giả thiết rằng chúng có thể tiệm cận xấp xỉ các hàm dư, tức  $H(x)-x$  (giả sử rằng đầu vào và đầu ra có cùng kích thước). Như vậy, thay vì đặt mục tiêu có thể xây dựng các lớp để trực tiếp có được  $H(x)$ , chúng ta cho phép các lớp này xây dựng một hàm dư  $F(x) := H(x)-x$ . Từ đó, hàm ban đầu sẽ trở thành  $F(x)+x$ . Mặc dù cả hai hình thức này đều có thể tiệm cận xấp xỉ các hàm mong muốn, khả năng học vẫn có thể bị ảnh hưởng.

Dựa trên ý tưởng này, mô hình ResNet đã được triển khai và đóng vai trò là một cột mốc quan trọng trong sự phát triển các mô hình học sâu nói chung. Việc áp dụng hàm dư giúp ResNet có khả năng phát triển về độ sâu nhưng vẫn giữ được khả năng huấn luyện tốt, khắc phục được vấn đề suy giảm của các mô hình học sâu trước đó. Điều này đồng nghĩa với việc các gradient có khả năng truyền ngược tốt hơn, cho phép việc học các đặc trưng phức tạp của dữ liệu hiệu quả hơn. Bên cạnh giành được chiến thắng trong ILSVRC 2015, các biến thể của ResNet được ứng dụng trong nhiều lĩnh vực và sử dụng thành công trong nhiều công trình nghiên cứu khoa học. ResNet50 được lựa chọn sử dụng trong đề này vì sự tương thích về phần cứng và độ phù hợp với tập dữ liệu.

## 2. 2. Khái quát về cách hoạt động

Mô hình ResNet50 về cơ bản hoạt động như sau:



Hình 3. Sơ bộ cấu trúc của ResNet50.

Hình ảnh đầu tiên sẽ trải qua lớp Zero Padding. Lớp này thêm các giá trị 0 vào biên của hình ảnh, đóng vai trò không làm giảm kích thước không gian của hình ảnh khi lớp tích chập xử lý các cạnh của hình ảnh.

Giai đoạn 1 gồm 4 lớp:

- Conv: Áp dụng các bộ lọc tích chập lên hình ảnh đầu vào, tạo ra feature maps, làm nổi bật các khía cạnh (đường viền, kết cấu, màu sắc,...) của hình ảnh.
- Batch Norm: Chuẩn hóa các kích hoạt từ lớp Conv, giúp ổn định và tăng tốc quá trình huấn luyện bằng cách giảm sự thay đổi nội tại giữa các lớp.
- ReLU: Đưa vào tính phi tuyến bằng cách áp dụng hàm kích hoạt ReLU, thiết lập tất cả các giá trị âm thành 0 và giữ nguyên các giá trị dương.
- Max Pooling: Giảm kích thước của feature maps bằng cách chọn giá trị tối đa từ mỗi vùng con của feature maps, giảm số lượng tham số và giúp ngăn ngừa overfitting.

Đây cũng là các lớp nằm trong những khối Conv ở các giai đoạn còn lại, nhưng được thiết lập với các thông số khác nhau. Các khối Conv mang ý nghĩa trích xuất những đặc trưng phức tạp hơn, phát triển về độ sâu cho mô hình. Khối ID là khối Conv bổ sung bằng residual connections. Đây chính là bước khắc phục vấn đề suy giảm khi ResNet cho phép dữ liệu được truyền trực tiếp, thay vì truyền tuần tự như các mô hình học sâu trước đó.

Sau khi trải qua năm giai đoạn trích xuất và học đặc trưng, ở bước cuối cùng hình ảnh sẽ trải qua ba lớp:

- Avg Pooling: Tại đây, lớp này sẽ lấy giá trị trung bình ở mỗi vùng con của featured maps. Điều này giúp tiếp tục giảm kích thước không gian và trung bình các đặc trưng, tăng tính ổn định của mô hình.
- Flattening: Chuyển đổi feature maps 2D thành một vector 1D – kiểu dữ liệu cần thiết cho lớp tiếp theo.

- Fully Connected: Lớp này thực hiện các suy luận và phân loại cao cấp dựa trên các đặc trưng được trích xuất bởi các lớp tích chập.

Như vậy, có thể tóm gọn cách hoạt động của mô hình ResNet50 như sau: Hình ảnh sau khi được đưa vào mô hình sẽ trải qua một bước học cơ bản các đặc trưng. Các giai đoạn sau đó sẽ tập trung vào học các đặc trưng phức tạp hơn và thường xuyên được bổ sung dữ liệu nhằm giải quyết vấn đề suy giảm. Cuối cùng, mô hình sẽ tổng hợp các đặc trưng học được và ánh xạ ra các dự đoán tương ứng.

### **2.3. Ưu điểm và nhược điểm**

ResNet50 nói riêng và ResNet nói chung đều thuộc nhóm những mô hình hiệu quả và đã có những đóng góp không nhỏ trong nhiều lĩnh vực khoa học. Điều này đồng nghĩa với việc ResNet50 mang một số lượng lớn ưu điểm để có độ tin cậy cao và sự tin dùng lớn như vậy.

Điểm sáng lớn nhất của ResNet chính là có thể giải quyết vấn đề suy giảm, một vấn đề tồn đọng đáng ngại của các mô hình học sâu trước đây. Nhờ có sự đột phá trong việc huấn luyện các mạng nơron, những mô hình học sâu về sau có khả năng phát triển về độ sâu cao hơn và giảm thiểu các vấn đề liên quan đến mất mát trong truyền dữ liệu. Với chiếc chìa khóa này, các mô hình thuộc loại ResNet đều đạt những kết quả huấn luyện tốt với độ chính xác cao trong những vấn đề đòi hỏi khả năng phân loại cao.

Tuy nhiên, vẫn tồn tại những bài toán mà ResNet với kỹ thuật bổ sung residual connections không thể giải quyết được. Kiến trúc ban đầu của ResNet50 được thiết kế và tối ưu hóa cho nhiệm vụ phân loại hình ảnh, khiến cho ResNet hoạt động kém hơn khi được sử dụng cho nhiệm vụ phát hiện đối tượng. Việc giảm kích thước không gian của các đặc trưng, tuy tăng hiệu suất tính toán và giảm độ phức tạp, đã ảnh hưởng đến việc ResNet xác định vị trí chính xác của các đối tượng nhỏ.

Với bất lợi này, ResNet50 có khả năng sẽ không phát hiện được những mụn nhỏ để phân loại, khiến cho đánh giá cuối cùng cho bài toán không được tốt. Chính vì thế, nhóm

nghiên cứu đề xuất việc kết hợp giữa YOLOv8 và ResNet50 nhằm đảm bảo khả năng phát hiện và phân loại mụn của đề tài.

### **3. MÔ TẢ VÀ XÁC ĐỊNH NGUYÊN NHÂN ĐỐI VỚI TỪNG LOẠI MỤN**

Bên cạnh vấn đề phát hiện và phân loại tốt các mụn, việc xác định nguyên nhân hình thành nên từng loại mụn cũng là câu hỏi mà nhóm sinh viên cần giải quyết. Để có thể hỗ trợ tối đa cho người dùng về mặt điều trị tình trạng da, xác định đúng nguyên nhân mụn sẽ tác động tích cực đến khả năng tìm ra những điều trị phù hợp. Nhóm nghiên cứu tập trung vào những lý do tạo nên mụn phổ biến đối với lứa tuổi đây thì được xác nhận bởi các chuyên gia, các tổ chức uy tín trong lĩnh vực y học, cũng như tham khảo gợi ý điều trị từ các nghiên cứu khoa học, bài báo nghiên cứu.

#### **3.1. Mụn đầu đen và mụn đầu trắng**

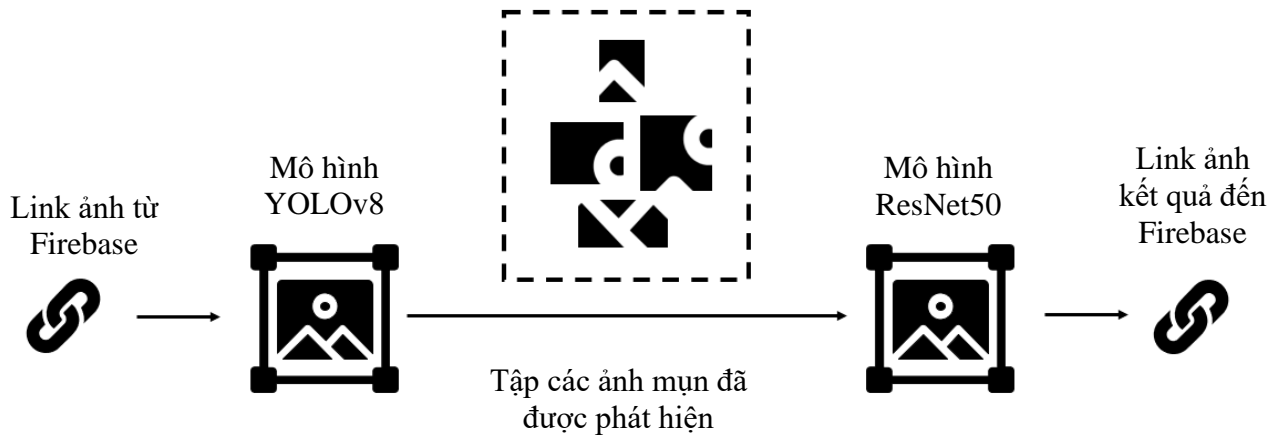
Mụn đầu đen và mụn đầu trắng có nguyên nhân hình thành giống nhau nhưng không tương đồng về sự hình thành, dẫn đến các đặc điểm nhận diện khác nhau. Nhìn chung, chúng đều được xem là mụn trứng cá và thường không xuất hiện tình trạng sưng viêm. Trong trường hợp mụn đầu đen, hay còn gọi là open comedones, các comedones cùng các lỗ chân lông dưới da sẽ mở rất lớn. Lúc này chúng bị tắc nghẽn bởi bã nhờn, đồng thời melanin sẽ bị oxy hóa và biến các lỗ chân lông thành màu đen [16]. Trong trường hợp mụn đầu trắng, hay còn gọi là closed comedones, các nang lông dưới da tuy chứa một lượng lớn vi khuẩn nhưng chỉ có một lỗ mở rất nhỏ ở đỉnh mụn. Điều này khiến không khí không thể vào được nang lông. Từ đó, vi khuẩn bên trong không bị oxy hóa và khiến mụn khi quan sát một cách trực quan sẽ có màu trắng.

#### **3.2. Mụn bọc**

Mụn bọc là một dạng mụn trứng cá nghiêm trọng, có kích thước tương đối lớn khi xuất hiện cùng những biểu hiện đau nhức, sưng đỏ ở vùng xung quanh. Mụn bọc thường không có nhân mụn rõ ràng hoặc cần thời gian để hình thành nhân mụn [17]. Về cơ bản, tình trạng mụn bọc xảy ra cũng do bít tắc lỗ chân lông, nhưng tình trạng viêm nhiễm do vi khuẩn sẽ từ trung bình đến nặng. Sự xâm nhập sâu của vi khuẩn gây nên phản ứng

## CHƯƠNG III: THIẾT KẾ GIẢI PHÁP

### 1. MÔ TẢ GIẢI PHÁP



Hình 5. Mô tả quá trình kết hợp mô hình YOLOv8 và mô hình ResNet50

Người dùng sau khi khởi động ứng dụng sẽ có thể sử dụng hình ảnh có sẵn hoặc chụp một hình mới để làm hình ảnh chẩn đoán. Hình ảnh này sau đó sẽ được gửi đến server, được lưu trữ tại Firebase, và URL ảnh sẽ được trả về frontend để chuyển tiếp cho backend. Tại đây, mô hình YOLOv8 sẽ lưu hình ảnh thông qua URL, dùng nó làm input để thực hiện tác vụ phát hiện đối tượng mụn, xác định bounding boxes của các đối tượng trên hình.

Kế tiếp, YOLOv8 sẽ cắt các vùng ảnh chứa đối tượng thành các ảnh nhỏ và gửi chúng đến mô hình ResNet50 để thực hiện tác vụ phân loại. Cuối cùng, hệ thống sẽ đánh dấu bằng các khung màu khác nhau, mỗi màu tương ứng với một lớp đối tượng lên trên ảnh input đầu tiên. Kết quả hình ảnh mới này sẽ được lưu tại Firebase, còn URL của ảnh này cùng kết quả chẩn đoán dạng JSON sẽ được gửi về cho frontend.

## 2. XÂY DỰNG TẬP DỮ LIỆU

### 2. 1. Xác định các nguồn tập dữ liệu sẽ sử dụng

Một trong những tác nhân quan trọng nhất ảnh hưởng đến hiệu suất của mô hình chính là tập dữ liệu. Xây dựng một tập dữ liệu đa dạng dữ liệu, cân bằng giữa các lớp và phù hợp với đề tài là vấn đề một tập dữ liệu duy nhất khó có thể đáp ứng đủ. Vì điều đó, nhóm nghiên cứu đã tìm hiểu và xác định tập dữ liệu của đề tài nên được tạo thành từ nhiều nguồn. Các nguồn được liệt kê sau đây bên cạnh là những tập dữ liệu con của các tập dữ liệu uy tín trên thế giới, còn là những tập dữ liệu có lượt sử dụng đông đảo trong các đề tài nghiên cứu khác.

- ImageNet: Đây là một cơ sở dữ liệu hình ảnh quy mô lớn được thiết kế để sử dụng trong nghiên cứu về nhận diện đối tượng và máy học. ImageNet bao gồm hàng triệu hình ảnh được gắn nhãn theo phân cấp của các đối tượng và được sử dụng để đánh giá trong nhiều nghiên cứu khoa học từ lớn đến nhỏ. Tổng số hình ảnh đã lấy tại Roboflow là xấp xỉ 700 hình ảnh.
- DermNet: Đây là một nguồn tài nguyên y tế hàng đầu về các bệnh về da. Tại trang web chính thức, các hình ảnh về da liễu được cung cấp với chất lượng cao, cùng với số lượng lớn tài liệu giáo dục và thông tin y học về các bệnh lý da liễu. Tổng số hình ảnh đã lấy tại Roboflow là xấp xỉ 300 hình ảnh.
- Roboflow: Đây là một nền tảng hỗ trợ cung cấp, quản lý, tiền xử lý và tăng cường dữ liệu hình ảnh cho các dự án máy học. Đây là một nguồn dữ liệu lớn và nổi tiếng trong lĩnh vực máy học nói chung và học sâu nói riêng. Tổng số hình ảnh đã lấy tại Roboflow là xấp xỉ 2,500 hình ảnh.
- Kaggle: Một nền tảng hữu ích và phổ biến về vấn đề máy học và học sâu. Nội dung được Kaggle cung cấp có thể kể đến các tập dữ liệu, mã nguồn và tài liệu học tập để thúc đẩy nghiên cứu và phát triển trong lĩnh vực máy học. Tổng số hình ảnh đã lấy tại Roboflow là xấp xỉ 1,000 hình ảnh.



## **2. 2. Xác định cấu trúc của tập dữ liệu**

Do hướng giải quyết đối với vấn đề mô hình là cùng lúc sử dụng hai loại mô hình phát hiện và phân loại, việc xây dựng kiến trúc của tập dữ liệu là câu hỏi cần cân nhắc kỹ lưỡng. Đầu tiên, mô hình YOLOv8 sẽ phát hiện những đối tượng là mụn trên hình ảnh nhận được, sau đó những hình ảnh này sẽ được gửi đến để phân loại bằng mô hình ResNet50. Từ đó, mô hình YOLOv8 cần một tập dữ liệu đơn nhãn với chỉ một lớp là mụn, cụ thể tập dữ liệu sẽ gồm một file data.yaml và hai thư mục train và val; Mỗi thư mục này sẽ gồm hai thư mục con là images chứa hình định dạng .jpg và labels định dạng .txt chứa nhãn của hình. Mặt khác, mô hình ResNet50 sử dụng một tập dữ liệu đa nhãn với năm lớp là mụn đầu đen, mụn đầu trắng, mụn bọc, mụn sần và mụn mủ; cụ thể, tập dữ liệu sẽ gồm hai thư mục train và val; Mỗi thư mục này sẽ gồm chứa các hình ảnh và một file data.csv; Trong đó, mỗi dòng sẽ lưu trữ tên hình ảnh và giá trị 1 tại cột đối tượng có trong hình ảnh, ngược lại giá trị là 0 nếu không chứa đối tượng đó..

Với yêu cầu như thế này, trước tiên nhóm nghiên cứu sử dụng các tập dữ liệu đa nhãn có kiến trúc gồm hai thư mục con là images chứa hình định dạng .jpg và labels định dạng .txt chứa nhãn của hình. Sau quá trình tinh chỉnh, nhóm nghiên cứu sẽ sao chép các hình ảnh đến thư mục của tập dữ liệu dành cho ResNet50. Đồng thời, các tập tin nhãn cũng sẽ được đọc để lưu trữ thông tin các đối tượng mà hình ảnh chứa trong tập tin data.csv.

## **2. 3. Tinh chỉnh các tập dữ liệu trước khi sử dụng**

Việc sử dụng tài nguyên từ nhiều nguồn khác nhau đòi hỏi một quá trình kiểm tra kỹ lưỡng nhằm đảm bảo sự thống nhất giữa dữ liệu, sự tương đồng cho quá trình huấn luyện mô hình, và nhiều yếu tố quan trọng khác. Một vấn đề phát sinh đó chính là, có những tập dữ liệu sẽ bao gồm những lớp không nằm trong phạm vi nghiên cứu, định dạng nhãn không theo định dạng mà nhóm nghiên cứu xác định từ trước, hoặc ID đánh dấu các lớp giữa các tập dữ liệu khác nhau. Do đó, đối với mỗi tập dữ liệu trước khi được thêm vào tập dữ liệu nghiên cứu, đều sẽ phải trải qua các bước như sau:

### 2. 2. 1. Loại bỏ những tập dữ liệu có định dạng nhãn không phù hợp

Định dạng bounding box là định dạng phổ biến khi huấn luyện các mô hình phát hiện đối tượng, cũng là định dạng được sử dụng thống nhất trong tập dữ liệu của đề tài nghiên cứu. Các nhãn theo định dạng bounding box sẽ cần có năm tham số, lần lượt từ trái sang phải là: ID của lớp, tọa độ x trung tâm theo chiều rộng, tọa độ y trung tâm theo chiều cao, chiều rộng và chiều cao của hình ảnh. Để tránh xung đột trong quá trình huấn luyện, các tập dữ liệu sẽ được kiểm tra định dạng nhãn trước khi chuyển sang bước tiếp theo.

### 2. 2. 2. Loại bỏ những hình ảnh chứa lớp không nằm trong phạm vi nghiên cứu

Mỗi tập dữ liệu sẽ được tối ưu hóa nhằm phù hợp một bài toán nhất định, đồng nghĩa các tập dữ liệu trước khi tinh chỉnh sẽ chứa nhiều lớp không nằm phạm vi nghiên cứu. Để loại bỏ các lớp không cần thiết, nhóm nghiên cứu sẽ sử dụng file data.yaml trong tập dữ liệu để xác định những lớp nào cần loại bỏ. Tất cả những hình ảnh chứa đối tượng không thể sử dụng sẽ bị xóa khỏi tập dữ liệu cùng nhãn.

### 2. 2. 3. Thống nhất cách ghi nhận về các lớp

Mỗi tập dữ liệu sẽ có một file data.yaml phục vụ cho việc lưu thông tin dữ liệu, bao gồm tên lớp và kí hiệu của lớp đó trong nhãn. Việc lưu trữ này sẽ thường là một mảng tên các lớp của tập dữ liệu, kí hiệu của các lớp sẽ tương ứng với vị trí trong mảng. Ví dụ, nếu phần tử 0 của mảng names là blackhead, có nghĩa là những đối tượng nào có ID là 0 thì là blackhead.

Từ việc các tập dữ liệu sẽ có số lượng lớp khác nhau, lưu trữ các lớp khác nhau cũng như vị trí của lớp trong mảng đều khác nhau, cần có sự thống nhất về mặt kí hiệu. Nhóm đã thống nhất thứ tự của các lớp trong mảng names của file data.yaml như sau: blackhead, nodule, papule, pustule, whitehead. Từ đó, đối chiếu với file data.yaml trong tập dữ liệu gốc, nhóm sẽ đổi kí hiệu của các lớp này theo quy ước trên.

### 2. 2. 4. Giải quyết vấn đề mất cân bằng dữ liệu

Sự cân bằng giữa các lớp trong tập dữ liệu đóng vai trò quan trọng đối với hiệu suất của mô hình. Nếu sự cân bằng không được đảm bảo, tình trạng underfitting hoặc

overfitting sẽ dễ xảy ra. Cho nên nhóm nghiên cứu sau khi chuyển dữ liệu của các tập dữ liệu đã tinh chỉnh đến tập dữ liệu của đề tài, sẽ thực hiện bước kiểm tra số đối tượng thuộc về các lớp. Nếu xuất hiện sự chênh lệch lớn hơn 10% giữa các lớp, nhóm sẽ thực hiện một bước điều chỉnh bổ sung nhằm đảm bảo sự cân bằng.

### 2. 3. Tổng kết về tập dữ liệu được sử dụng trong đề tài

Sau quá trình tinh chỉnh và điều chỉnh, tập dữ liệu mà nhóm nghiên cứu sử dụng khoảng 4,500 hình ảnh, trong đó 85% được sử dụng cho bước training, 10% sử dụng cho bước validating và 5% sử dụng cho bước testing đối với cả hai loại mô hình.

Tên đối tượng	Mụn đầu đen	Mụn đầu trắng	Mụn bọc	Mụn sần	Mụn mủ
Số lượng	3100	3097	3156	3065	3080

*Bảng 4. Bảng chi tiết số lượng đối tượng trong tập dữ liệu*

Sau khi đã hoàn thành bước tinh chỉnh tập dữ liệu, nhóm sẽ sao chép các ảnh và tạo tập tin data.csv để tạo tập dữ liệu cho mô hình ResNet50 để thực hiện tác vụ phân loại như đã đề cập. Mặt khác, nhóm cũng sẽ đổi các nhãn về một lớp (acne) và tạo tập dữ liệu cho mô hình YOLOv8 nhằm thực hiện tác vụ phát hiện.

## 3. TRIỂN KHAI GIẢI PHÁP

Để mô hình có được hiệu suất tốt nhất, quá trình tinh chỉnh là giai đoạn chìa khóa để biết được nhóm những siêu tham số nào có thể sử dụng được. Nhóm trước tiên sử dụng những siêu tham số sau đây làm tiền đề huấn luyện cho mô hình phát hiện YOLOv8: epoch là 20, batch size là 32, img size là 640, learning rate là 0.0001, learning rate final là 0.01, optimizer là AdamW, weight decay là 0.00175, và momentum là 0.9. Kết quả đầu tiên cho thấy YOLOv8 có kết quả Accuracy là 69% và 50% cho Precision. Đây là một kết quả trung bình khá, đủ để nhóm sử dụng để có thể tinh chỉnh độc lập những

tham số: lr0, lrf, weight decay và momentum. Kết thúc giai đoạn kiểm thử, nhóm sinh viên sàng lọc 5 trường hợp tốt nhất để huấn luyện mô hình.

Đối với mô hình ResNet50, trong quá trình nghiên cứu cho thấy mô hình phân loại có mức hiệu suất tốt trong khoản thời gian ngắn và ít hao tốn tài nguyên hơn so với YOLOv8. Nhóm số liệu được áp dụng cho mô hình ResNet50 đầu tiên là epoch 20, lr0 0.0001, batch size 32, beta1 0,9, beta2 0.99, weight decay là 0.002. Kết quả của lần huấn luyện đầu tiên này trên tập test là 79% về độ chính xác. Từ đó, nhóm tiếp tục tinh chỉnh các siêu tham số khác để tìm được nhóm số liệu tối ưu nhất.

#### 4. ĐÁNH GIÁ KẾT QUẢ MÔ HÌNH KẾT HỢP DỰA TRÊN CÁC THÔNG SỐ

Trong mỗi quá trình huấn luyện và kiểm thử các số liệu cho mô hình YOLOv8 và ResNet50, nhóm sinh viên chọn ra năm trường hợp siêu tham số. Bảng dưới đây trình bày các số liệu đã được nhóm tinh chỉnh để cho ra kết quả tốt nhất.

Trường hợp	lr0	lrf	weight decay	momentum
1	0.0001	0.01	0.00175	0.91
2	0.0001	0.02	0.002	0.91
3	0.00075	0.01	0.002	0.9
4	0.0001	0.02	0.0018	0.91
5	0.00075	0.02	0.00175	0.9

*Bảng 5. Bảng chi tiết các số liệu đã kiểm thử đối với những lần huấn luyện tốt nhất của mô hình YOLOv8*

Trường hợp	lr	weight decay	beta(1,2)
1	0.0001	0.002	(0.1, 0.8)
2	0.00005	0.001	(0.9, 0.999)
3	0.0001	0.001	(0.1, 0.999)
4	0.000075	0.00025	(0.8, 0.999)
5	0.00006	0.00175	(0.9, 0.99)

*Bảng 6. Bảng chi tiết các số liệu đã kiểm thử đối với những lần huấn luyện tốt nhất của mô hình ResNet50*

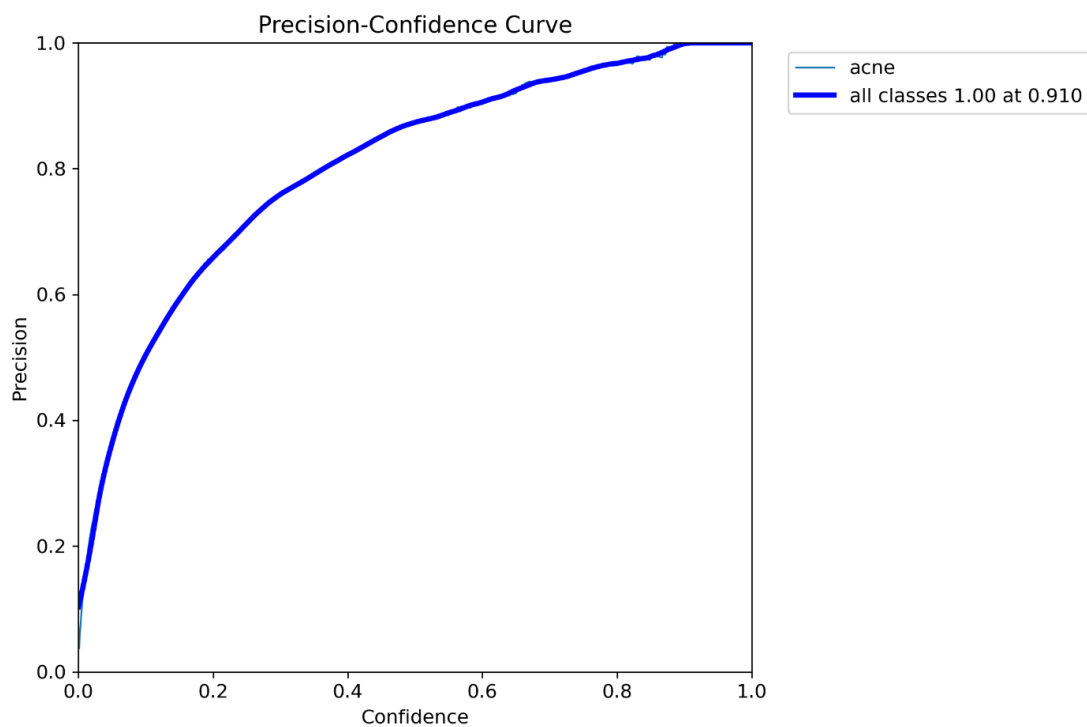
#### **4. 1. Kết quả huấn luyện mô hình YOLOv8**

##### **4. 1. 1. So sánh kết quả huấn luyện YOLOv8 giữa các trường hợp**

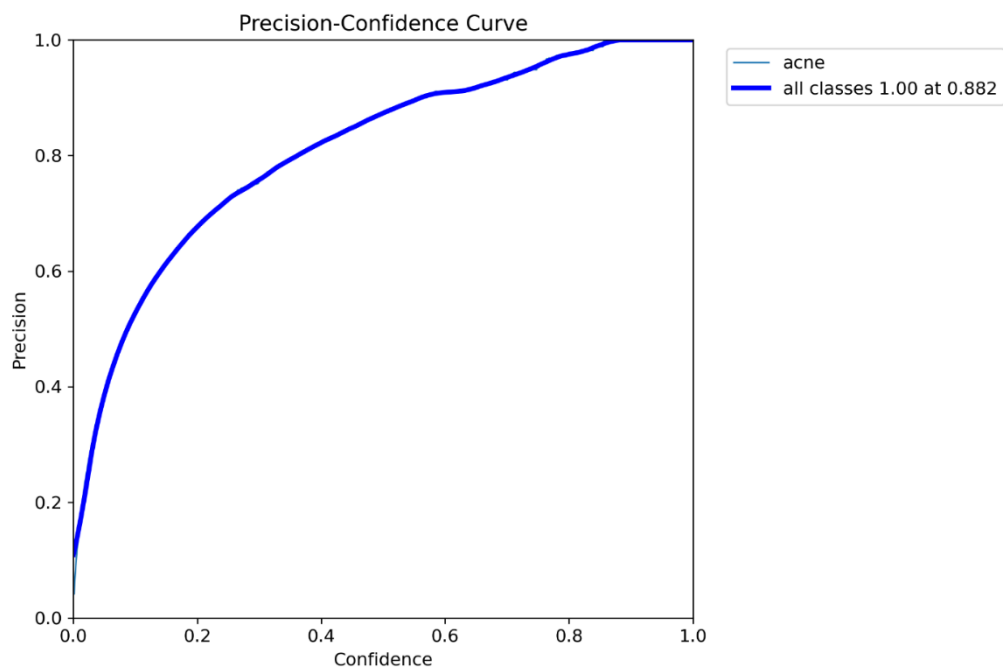
Lần thử	Precision	Recall	mAP50	mAP50-95
1	0.744	0.515	0.586	0.328
2	0.709	0.528	0.59	0.328
<b>3</b>	<b>0.785</b>	<b>0.503</b>	<b>0.595</b>	<b>0.347</b>
4	0.727	0.515	0.588	0.332
5	0.714	0.549	0.609	0.355

*Bảng 7. So sánh kết quả huấn luyện YOLOv8 giữa các trường hợp*

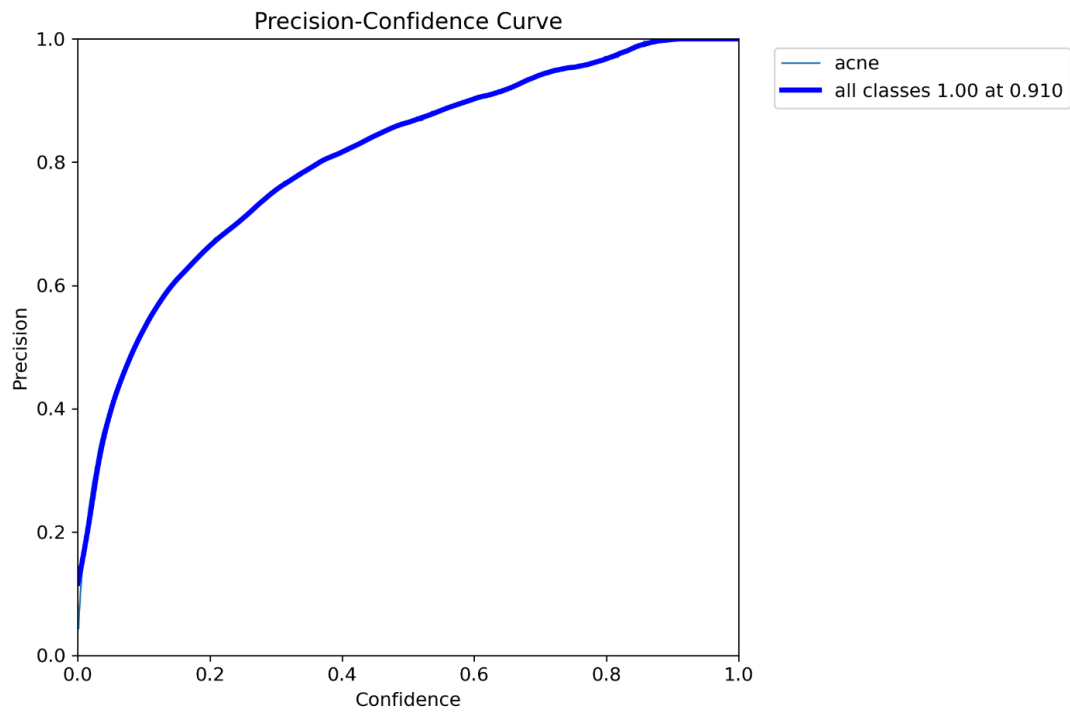
#### 4. 1. 2. So sánh dựa trên Precision-Confidence Curve



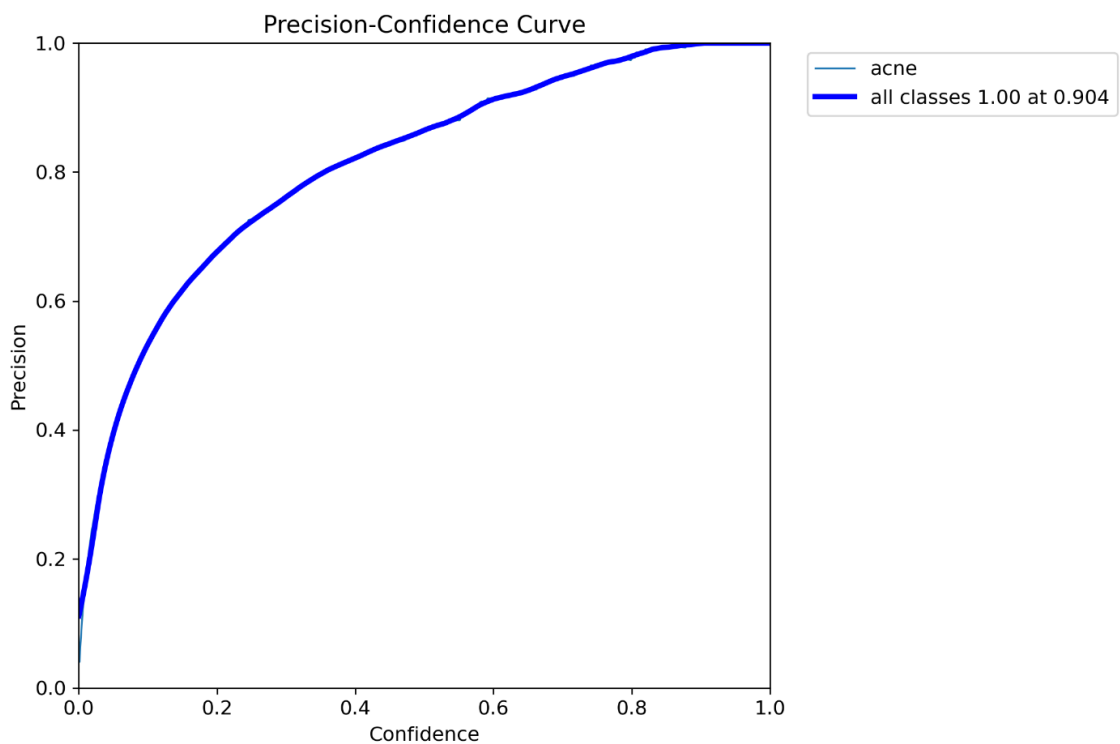
Hình 6. Precision-Confidence Curve của trường hợp YOLOv8 số 1



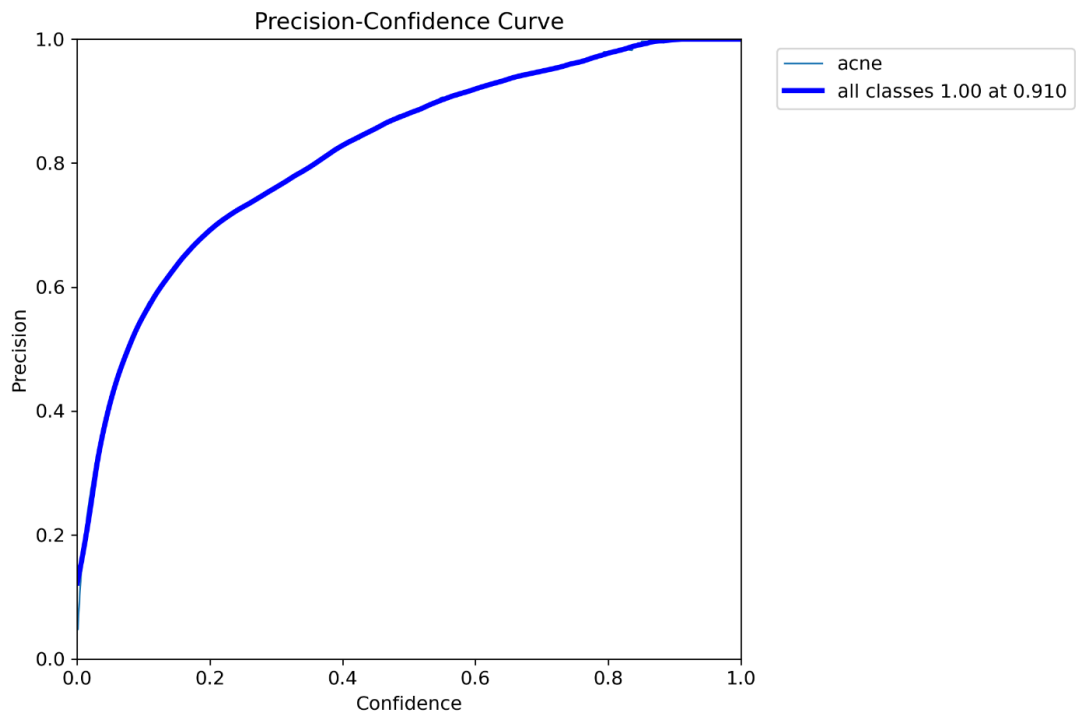
Hình 7. Precision-Confidence Curve của trường hợp YOLOv8 số 2



Hình 8. Precision-Confidence Curve của trường hợp YOLOv8 số 3



Hình 9. Precision-Confidence Curve của trường hợp YOLOv8 số 4



Hình 10. Precision-Confidence Curve của trường hợp YOLOv8 số 5

## 4. 2. Kết quả huấn luyện mô hình ResNet50

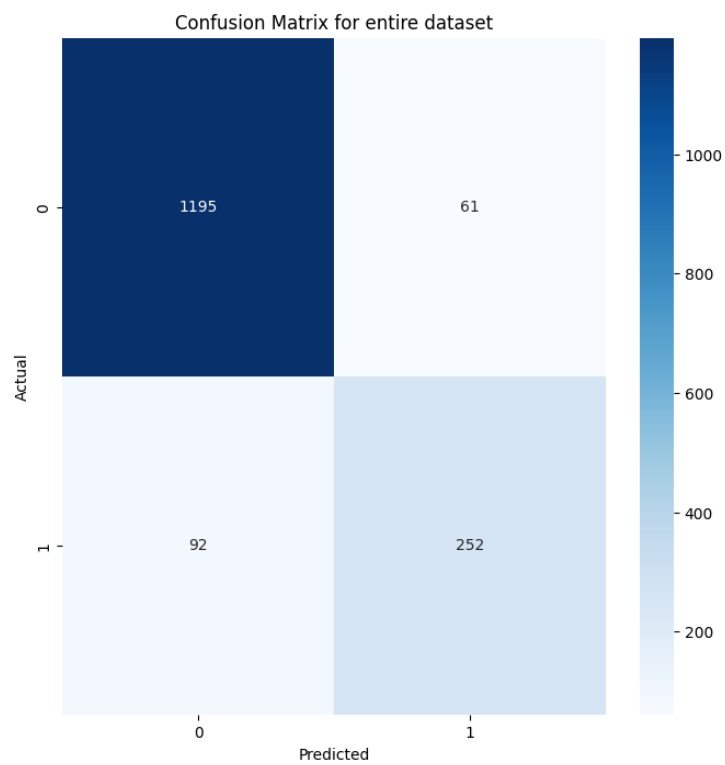
### 4. 2. 1. So sánh dựa trên Accuracy

Lần thử	Train Accuracy	Val Accuracy	Test Accuracy
<b>1</b>	<b>1</b>	<b>0.903</b>	<b>0.908</b>
2	0.997	0.886	0.8876
3	0.995	0.878	0.904
4	0.996	0.893	0.901
5	0.996	0.885	0.884

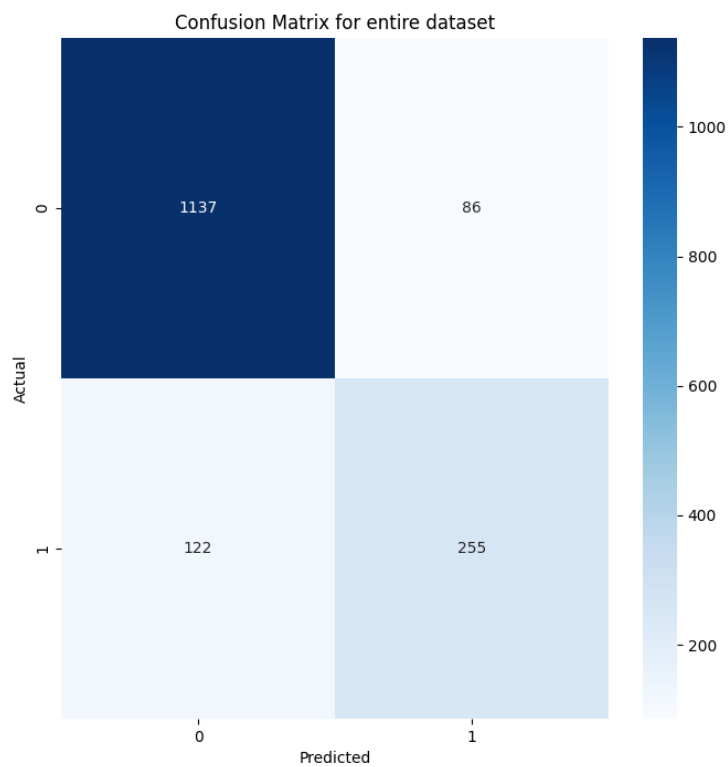
Bảng 8. So sánh kết quả huấn luyện ResNet50 giữa các trường hợp



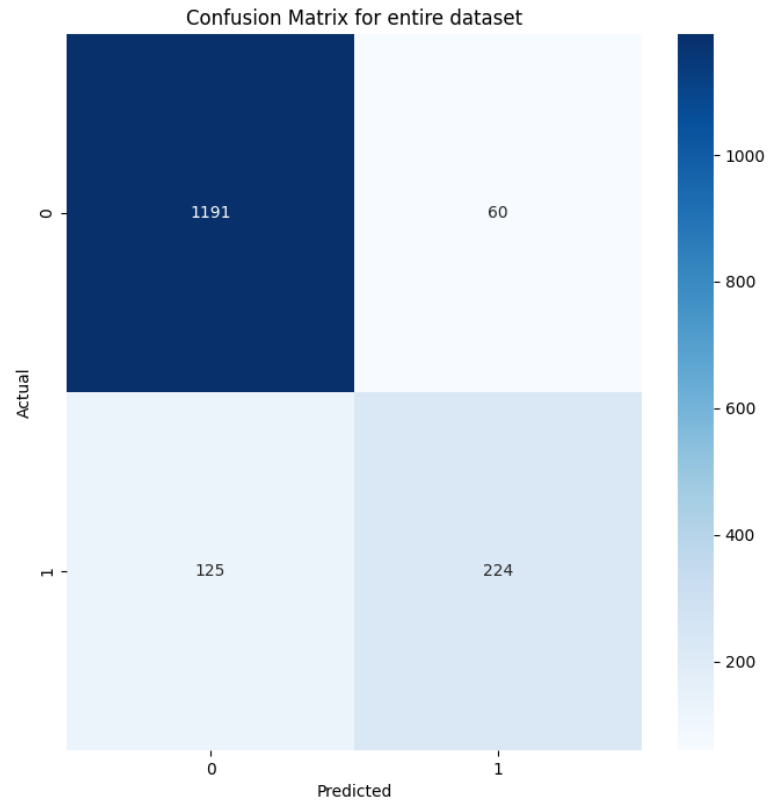
#### 4. 2. 1. So sánh dựa trên Confusion Matrix



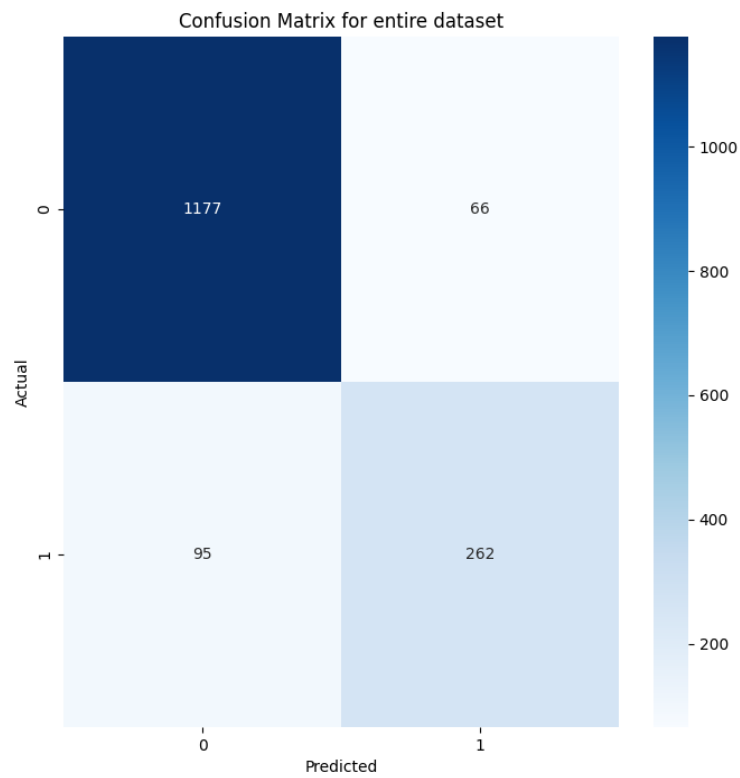
Hình 11. Confusion Matrix cho lần huấn luyện số 1 của ResNet50



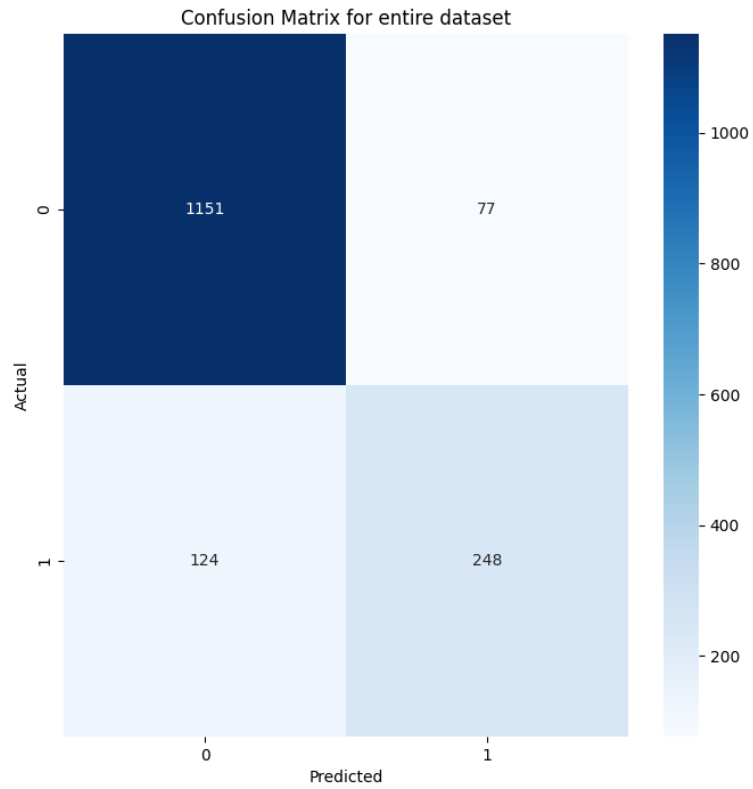
Hình 12. Confusion Matrix cho lần huấn luyện số 2 của ResNet50



Hình 13. Confusion Matrix cho lần huấn luyện số 3 của ResNet50



Hình 14. Confusion Matrix cho lần huấn luyện số 4 của ResNet50



*Hình 15. Confusion Matrix cho lần huấn luyện số 5 của ResNet50*

## 5. KẾT LUẬN VÀ ĐÁNH GIÁ GIẢI PHÁP ĐỀ XUẤT

### 5. 1. Các hạn chế khi sử dụng độc lập mô hình YOLOv8 và mô hình ResNet50

Về cơ bản, YOLOv8 có thể thực hiện nhiệm vụ phân loại đối tượng cũng như ResNet50 có thể thực hiện nhiệm vụ phát hiện đối tượng nếu có những sự điều chỉnh phù hợp. Tuy nhiên, điều này lại đặt hai mô hình vào những bài toán mà chúng không thể phát huy điểm mạnh. Điều này có thể chứng minh bằng kiểm thử với những kết quả như sau:

Mô hình	Nhóm số liệu 1	Nhóm số liệu 2	Nhóm số liệu 3	Nhóm số liệu 4	Nhóm số liệu 5
YOLOv8 (5 lớp)	0.71	0.68	0.69	0.7	0.69
YOLOv8 (1 lớp)	0.744	0.709	0.785	0.727	0.714

*Bảng 9. So sánh về độ chính xác giữa YOLOv8 (5 lớp) và YOLOv8 (1 lớp)*

Mô hình	Nhóm số liệu 1	Nhóm số liệu 2	Nhóm số liệu 3	Nhóm số liệu 4	Nhóm số liệu 5
ResNet50 (Mô hình phát hiện)	0.135	0.089	0.081	0.146	0.071
ResNet50 (Mô hình phân loại)	1	0.997	0.995	0.996	0.996

*Bảng 10. So sánh về độ chính xác giữa ResNet50 (mô hình phát hiện) và ResNet50 (mô hình phân loại)*

Nhóm nghiên cứu nhận thấy, tồn tại một giải pháp đơn giản để tận dụng ưu điểm của YOLOv8 và ResNet50 để giải quyết vấn đề. Như đã trình bày, kiến trúc tổng quan của

YOLO nói chung được thiết kế nhằm phục vụ hiệu quả cho bài toán phát hiện đối tượng, cụ thể là sự cân bằng giữa hai yếu tố khả năng học đặc trưng và tốc độ học những đặc trưng ấy. Mặc khác, các phiên bản ResNet đều có một ưu điểm là tạo ra những feature maps giàu thông tin, tập trung vào các tiểu tiết, một điều rất quan trọng trong vấn đề phát hiện và phân loại mụn. Do đó, nhóm đề xuất giải pháp sẽ sử dụng YOLOv8 làm mô hình phát hiện và ResNet50 làm mô hình phân loại. Mặc dù việc sử dụng cùng lúc hai mô hình tồn tại một số nhược điểm nhất định, nhóm sinh viên tin rằng định hướng này là câu trả lời phù hợp với đề tài nghiên cứu.

## **5. 2. Kết luận và đánh giá về xây dựng tập dữ liệu và kết hợp hai mô hình**

Một trong những yếu tố khiến việc huấn luyện mô hình thành công nằm ở sự xây dựng tập dữ liệu hợp lý. Do yêu cầu về đối tượng và phạm vi nghiên cứu của đề tài, nhóm sinh viên nhận định không thể sử dụng một tập dữ liệu có sẵn để huấn luyện mô hình mà cần thời gian để tìm hiểu và tinh chỉnh thành giải pháp tối ưu cho đề bài. Quá trình này bao gồm: tìm và xác định loại tập dữ liệu phù hợp cho từng loại mô hình, tinh chỉnh nhằm đảm bảo không có đối tượng không nằm trong phạm vi nghiên cứu cũng như sự cân bằng giữa các lớp, sự phân chia giữa các tập huấn luyện và kiểm thử. Kết quả của việc xây dựng tập dữ liệu tùy chỉnh là 4,500 ảnh, trong đó mỗi loại mụn chiếm 20%. Có thể nói tổng số đối tượng của tập dữ liệu khó có thể so sánh với nhiều tập dữ liệu khác, tuy nhiên đây là một tiền đề tốt để nhóm nghiên cứu có thể phát triển đề tài, cũng như đồng nghĩa với việc phát triển tập dữ liệu này trong tương lai sẽ có thể hỗ trợ đáng kể trong việc tối ưu hiệu quả cho các mô hình.

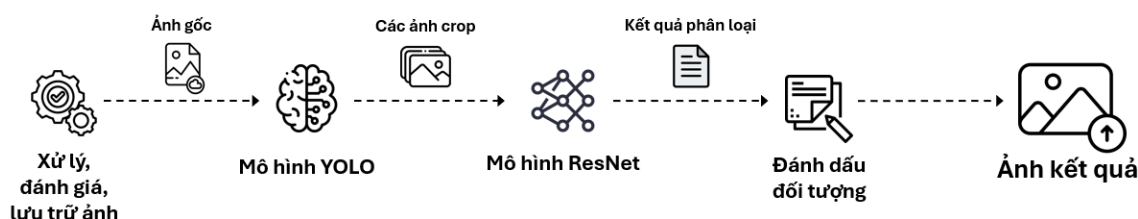
Về việc sử dụng phương pháp cho bài toán phát hiện và phân loại các loại mụn phổ biến trên gương mặt, nhóm nghiên cứu đã chọn việc sử dụng song song hai loại mô hình là mô hình phát hiện YOLOv8 và ResNet50. Nhóm cũng nhận thấy giải pháp như kết hợp YOLOv8 và ResNet50 trong một mô hình có thể là cách tiếp cận hợp lý và tiết kiệm tài nguyên hơn, nhưng điều này đòi hỏi sự nghiên cứu chuyên sâu và dài hạn hơn để có thể xác định các lớp hợp lý trong kiến trúc kết hợp, những số liệu phù hợp nhằm đảm bảo sự tương thích giữa hai loại công nghệ phân loại và phát hiện trong cùng một mô

hình. Do đó, sử dụng độc lập hai mô hình để đảm nhiệm lần lượt hai nhiệm vụ là sự lựa chọn hợp lý hơn đối với đề tài này. Nhóm đã thử nghiệm bằng việc tinh chỉnh nhiều loại siêu tham số và kiểm tra trên nhiều trường hợp để tìm ra nhóm số liệu phù hợp nhất đối với hai mô hình. Kết quả thực tế cho thấy, với các tham số learning rate là 0.00075, weight decay là 0.001, momentum là 0.9, và learning schedule là 0.002, mô hình phát hiện YOLOv8 cho ra hiệu suất tốt nhất. Đối với mô hình phân loại ResNet50, hiệu quả của mô hình sẽ tối ưu khi sử dụng các tham số learning rate là 0.0001, weight decay là 0.002, và betas là từ 0.1 đến 0.8. Khi so sánh các số liệu này với việc yêu cầu YOLOv8 hoặc ResNet50 đồng thời đảm nhiệm việc phân loại và phát hiện là không phù hợp. Từ đó, có thể khẳng định rằng, đặt đúng giải pháp cho đúng vấn đề, hay phân biệt hai nhiệm vụ cho hai mô hình là cách tiếp cận phù hợp để đưa ra kết quả tốt nhất cho đề tài nghiên cứu với sự hạn chế về tài nguyên và dữ liệu.

# CHƯƠNG IV: XÂY DỰNG HỆ THỐNG HỖ TRỢ CHẨN ĐOÁN BỆNH DA LIỄU TRÊN GƯƠNG MẶT

## 1. THIẾT LẬP SERVER

### 1. 1. Tổng quan



Hình 16. Sơ đồ hoạt động của server.

Quy trình hoạt động của máy chủ được thực hiện như sau:

Khi URL ảnh cần chẩn đoán được gửi đến server từ Firebase, hệ thống sẽ truy cập và tải ảnh về kho lưu trữ cục bộ. Sau khi ảnh được tải thành công, mô hình YOLO sẽ tiến hành nhận dạng các đối tượng mụn hiện diện trong ảnh. Nếu có mụn được phát hiện, hệ thống sẽ tạo ra một file label tương ứng và các ảnh crop cho từng đối tượng mụn. Tại giai đoạn này, tất cả các đối tượng mụn đều được phân loại vào một lớp duy nhất.

Tiếp theo, các ảnh crop này sẽ được xử lý để phân loại theo các lớp đã được xác định, sử dụng mô hình ResNet đã được huấn luyện trước đó. Dựa trên kết quả phân loại, hệ thống sẽ đánh dấu vị trí của từng mụn cùng với phân loại lớp của chúng lên ảnh gốc. Khi việc đánh dấu hoàn tất, ảnh đã được xử lý sẽ được lưu trữ trên Firebase, đồng thời hệ thống sẽ trả về URL của ảnh này cùng với các thông tin chẩn đoán cho frontend.

Hệ thống cũng có cơ chế xử lý khi sự cố bất ngờ xảy ra. Nếu có vấn đề ở phía server, chẳng hạn như URL không hợp lệ hoặc phát hiện lỗi trong file ảnh, hệ thống sẽ gửi đi thông báo đến frontend để hỗ trợ trong công tác xử lý sự cố, đảm bảo cho quy trình được hoạt động ổn định.

Trong quá trình xây dựng ứng dụng, nhóm đã sử dụng Pipenv, một công cụ quản lý môi trường ảo Python, giúp cho việc cài đặt và kiểm soát các thư viện dễ dàng hơn. Với Pipenv, ta có thể theo dõi các dependencies trong dự án một cách đơn giản và nhanh chóng thiết lập lại môi trường khi cần thiết.

Server của hệ thống được xây dựng bằng Django, một framework Python phổ biến, mạnh mẽ và dễ sử dụng. Django không chỉ giúp quản lý cơ sở dữ liệu, hệ thống template, xác thực người dùng và các công cụ bảo mật tích hợp sẵn giúp phát triển web nhanh chóng. Nhóm cũng tích hợp Django REST Framework (DRF), giúp tạo các API RESTful dễ dàng, cho phép frontend và backend giao tiếp với nhau thông qua các yêu cầu HTTP.

Để đảm bảo các yêu cầu từ frontend đến backend diễn ra mượt mà, server sử dụng thư viện Requests. Thư viện này hỗ trợ việc gửi và nhận dữ liệu từ server một cách dễ dàng, phù hợp cho việc tương tác với các API từ xa hoặc thu thập dữ liệu trực tuyến.

Ngoài ra, nhóm cũng tích hợp Pyrebase4, một thư viện Python giúp kết nối với các dịch vụ của Firebase, chẳng hạn như Authentication, Realtime Database, Storage, và Hosting. Điều này giúp cho việc quản lý thông tin người dùng và lưu trữ dữ liệu an toàn và hiệu quả.

Bảo mật thông tin nhạy cảm như API keys và các biến môi trường bằng cách sử dụng python-dotenv. Thư viện này giúp lưu trữ các thông tin nhạy cảm trong file `.env` riêng biệt, đảm bảo rằng những thông tin quan trọng không bị lộ ra ngoài mã nguồn.

Cuối cùng, để tích hợp các mô hình học máy vào hệ thống, nhóm đã sử dụng thư viện Ultralytics. Đây là một công cụ giúp triển khai mô hình YOLO vào dự án để xử lý các tác vụ liên quan đến nhận dạng đối tượng từ hình ảnh.



## 1.2. Thiết kế API

### 1.2.1. Tổng quan về API

API (Application Programming Interface) là một giao diện cho phép các phần mềm giao tiếp với nhau. Trong ứng dụng này, API được sử dụng để kết nối giữa Frontend (giao diện người dùng) và Backend (nơi xử lý logic và lưu trữ dữ liệu). Cụ thể, Frontend sẽ gửi yêu cầu chứa đường link ảnh đến Backend. Backend sẽ xử lý ảnh thông qua model đã được huấn luyện và trả về kết quả xử lý là một đường link ảnh mới.

### 1.2.2. Thiết kế endpoint

Endpoint của API sẽ chịu trách nhiệm tiếp nhận yêu cầu POST từ Frontend, yêu cầu này sẽ chứa đường link ảnh mà người dùng muốn chẩn đoán. Sau khi nhận được yêu cầu, phía Backend sẽ tải ảnh từ đường link đó và đưa vào model để xử lý. Khi xử lý thành công Backend ảnh sẽ được lưu trữ trên Firebase và trả về kết quả là đường link dẫn đến ảnh đã chẩn đoán cùng với những thông tin như loại mụn và số lượng từng loại.

URL: /api/detect/

Phương thức HTTP: POST

Request: JSON chứa đường link ảnh:

```
"image_url": "https://example.com/path/to/image.jpg"
```

Response: JSON chứa đường link ảnh kết quả và thông tin liên quan

```
"status": "status_of_processing",
```

```
"data": {
```

```
  "detected_image_url": "https://example.com/path/to/detected_image.jpg"
```

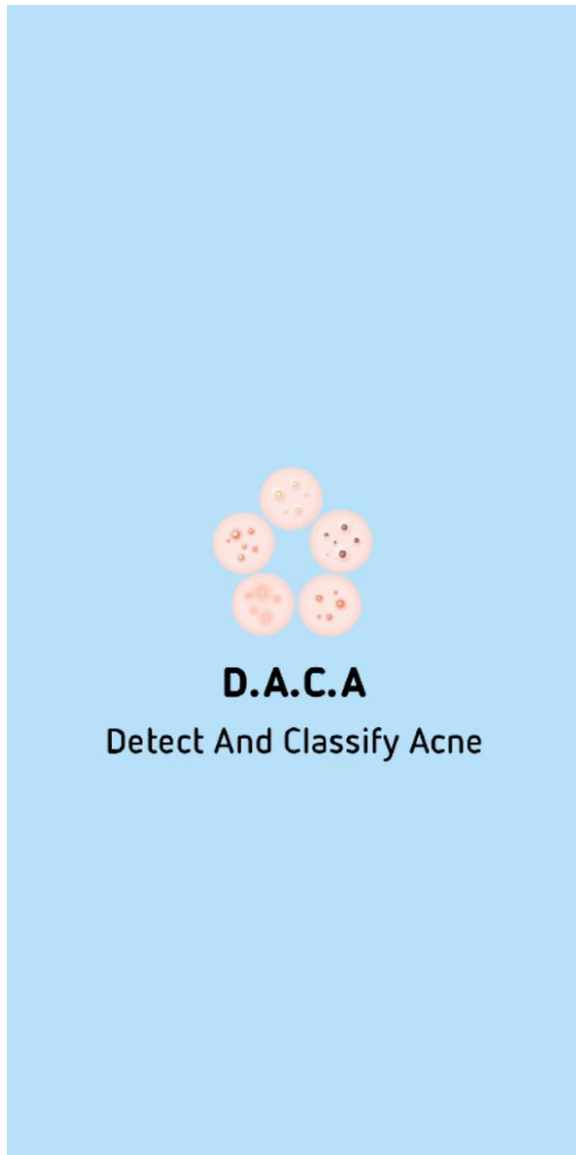
```
  "number_of_acnes": {
```

```
    "acne_type": "quantity"
```

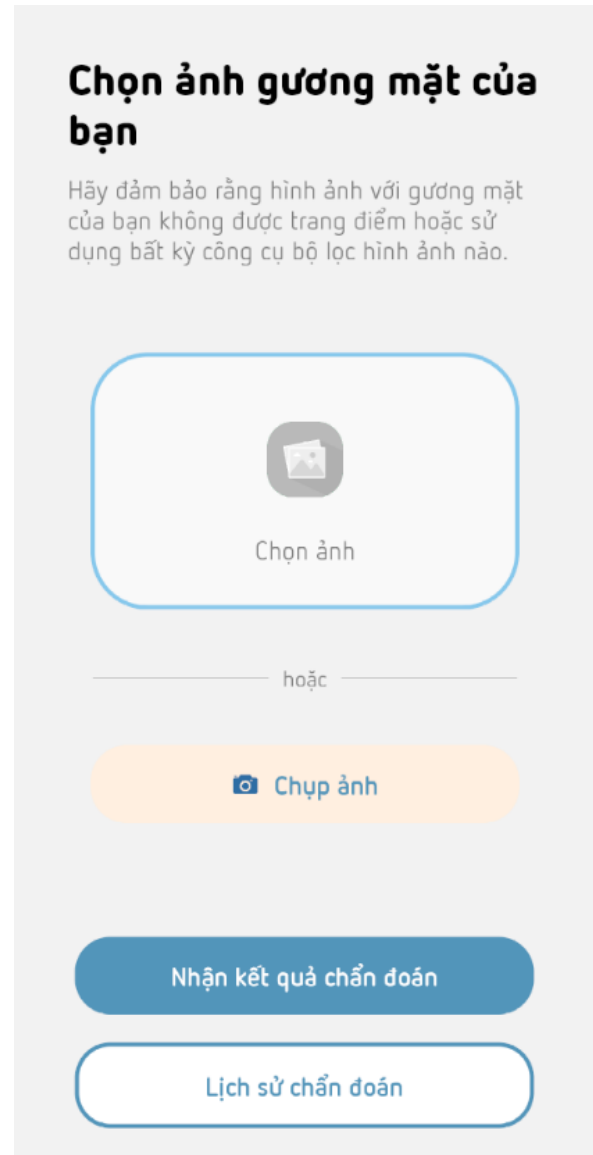
```
  }
```

```
}
```

## 2. CÀI ĐẶT GIAO DIỆN VÀ CHỨC NĂNG CHO ỨNG DỤNG

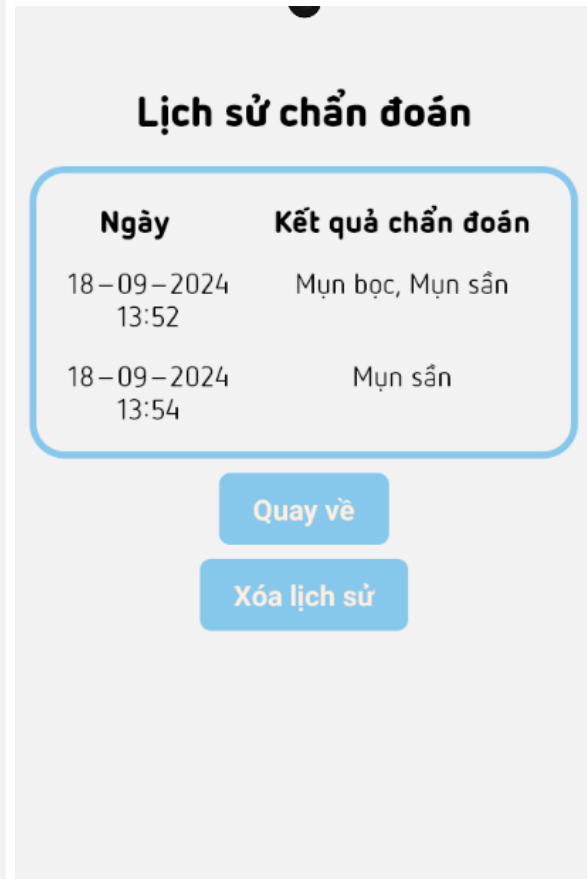
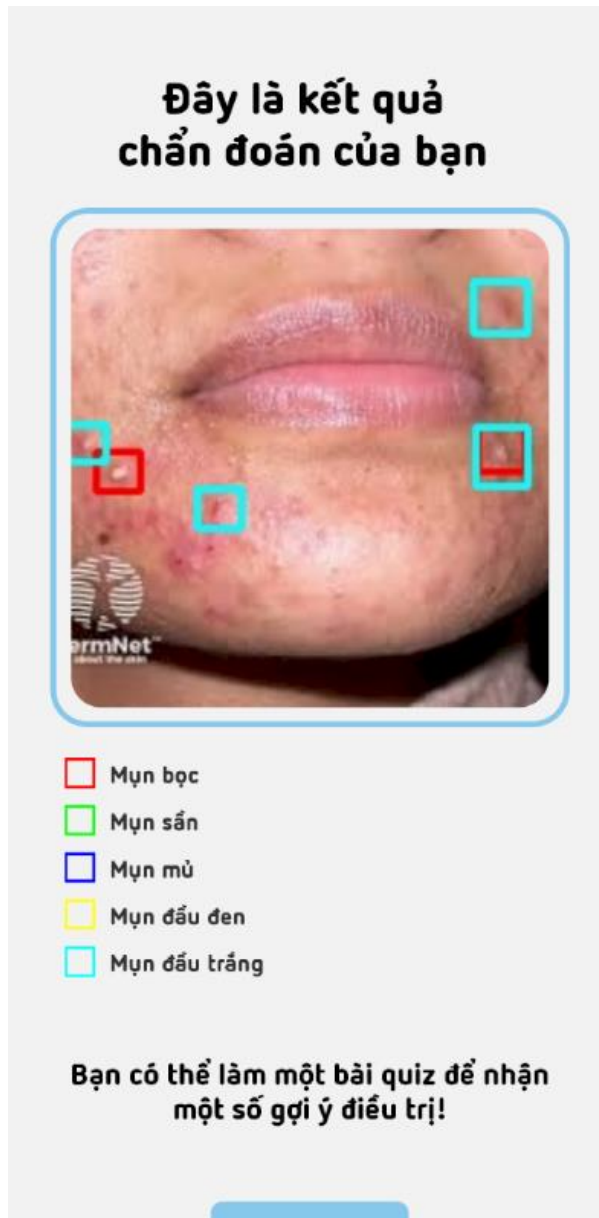


Hình 177. Logo của ứng dụng



Hình 188. Giao diện chính

Khi khởi động ứng dụng, giao diện trong hình 17 là giao diện đầu tiên người dùng nhìn thấy. Sau khi logo biến mất, giao diện chính của ứng dụng, đồng thời là hình 18, sẽ xuất hiện. Đây là nơi người dùng có thể sử dụng hình ảnh có sẵn hoặc chụp một hình mới để nhận kết quả chẩn đoán. Tại đây người dùng cũng có thể xem lại những lần chẩn đoán trước đó.



Hình 1919. Giao diện khi nhận kết quả

Hình 200. Giao diện hiển thị lịch sử chẩn đoán

Kết quả gửi về là hình ảnh với các bounding boxes có khung viền màu sắc để phân loại giữa các đối tượng khác nhau. Kết quả này sẽ được hiển thị tại như hình 19 và sẽ kèm theo các chú thích để người dùng xác định cụ thể mình gặp phải mụn nào tại vị trí nào trên gương mặt. Tại đây người dùng cũng có thể thực hiện một bài khảo sát để nhận một số gợi ý điều trị cho tình trạng da liễu của mình.

Đối với chức năng xem lại lịch sử chẩn đoán, dữ liệu sẽ được lưu trên thiết bị của người dùng với các hai trường là thời gian và kết quả chẩn đoán đã thực hiện. Khi được gọi, ứng dụng sẽ sử dụng AsynStorage để lưu các phản hồi từ server về thiết bị, đối với mỗi lần thực hiện chức năng thì sẽ hiển thị dữ liệu cho người dùng dưới dạng bảng, cụ thể là cách hiển thị trong hình 20.

"Bài quiz này sẽ giúp bạn hiểu rõ hơn về tình trạng da của mình bằng cách cung cấp các gợi ý điều trị phù hợp. Hãy chọn các phương án đúng với cách sinh hoạt của bạn để nhận được lời khuyên tốt nhất!"

**Các vấn đề về hormone:**

- Có sử dụng thuốc kê đơn của bác sĩ
- Cân nặng không ổn định
- Bất đều cô kinh nguyệt/Kinh nguyệt không đều
- Căng thẳng do học tập
- Tâm trạng thất thường, dễ cáu gắt
- Khó tập trung, khó ghi nhớ thông tin

**Các vấn đề về giấc ngủ:**

- Khó ngủ
- Mệt mỏi mặc dù đã nghỉ ngơi đầy đủ

**Đối với Mụn mủ:**

- Benzol Peroxide
- Salicylic Acid
- Retinoids
- Azelaic Acid
- Corticosteroid
- Calamine
- Cortisone
- Dapsone

**Gợi ý điều trị dựa trên lựa chọn của bạn:**

**Đối với những vấn đề hormone:**

- Xem xét các phương pháp điều trị nội tiết như bổ sung kẽm hoặc vitamin B6 theo chỉ dẫn của bác sĩ.
- Thực hành các kỹ thuật thở sâu hoặc tham gia các lớp học hỗ trợ quản lý căng thẳng.
- Hạn chế sử dụng caffeine và đường để duy trì tâm trạng ổn định hơn.
- Áp dụng các bài tập trí nhớ đơn giản hoặc sử dụng ứng dụng hỗ trợ để giảm thiểu stress tinh thần.

Hình 2121. Giao diện làm khảo sát

Hình 22. Giao diện nhận kết quả khảo sát

Hình 21 là giao diện khi người dùng chọn thực hiện khảo sát. Tại đây, người dùng sẽ cần phải trả lời các câu hỏi có liên quan đến nguyên nhân gây nên mụn và có thể chọn

những phương án phù hợp với tình trạng của bản thân. Ứng dụng sẽ tính toán để xem những nguyên nhân xuất hiện với cường độ cao là những nguyên nhân chủ yếu và đề xuất một số các giải pháp phù hợp, cũng như căn cứ theo kết quả chẩn đoán mụn trước đó mà đề xuất một số hoạt chất điều trị phù hợp. Trường hợp người dùng không chọn phương án nào hoặc ứng dụng không thể tính toán được đâu là nguyên nhân chủ yếu, người dùng sẽ được yêu cầu để cung cấp thêm các nguyên nhân có liên quan. Hình 22 là hình kết quả khảo sát mà người dùng nhận được, có kèm theo một số hoạt chất phù hợp với loại mụn đã được phát hiện.

### **3. ĐÁNH GIÁ VỀ VIỆC XÂY DỰNG SERVER VÀ ỨNG DỤNG CHẨN ĐOÁN**

Việc kết hợp giữa ứng dụng chẩn đoán và hệ thống server đã chứng minh sự hiệu quả trong việc khai thác tiềm năng của cả công nghệ di động và hệ thống backend. Sự phối hợp này không chỉ đảm bảo rằng ứng dụng đáp ứng được các yêu cầu cơ bản về hiệu suất và độ tin cậy, mà còn tạo điều kiện thuận lợi cho việc xử lý dữ liệu nhanh chóng và chính xác. Thông qua Firebase, dữ liệu được lưu trữ và truy xuất một cách an toàn, đảm bảo rằng thông tin cá nhân của người dùng được bảo mật chặt chẽ. Server xây dựng bằng Node.js có thể đảm nhận việc xử lý các yêu cầu từ ứng dụng một cách hiệu quả, đảm bảo rằng người dùng nhận được kết quả chẩn đoán một cách nhanh chóng mà không gặp phải độ trễ không mong muốn. Mặt khác, khi kết hợp React Native và Node.js, sản phẩm của đề tài có khả năng mở rộng linh hoạt và dễ dàng khi cần thiết. Nhìn chung, sự kết hợp giữa server và ứng dụng đã mang lại một giải pháp mạnh mẽ, đáng tin cậy, sẵn sàng đáp ứng nhu cầu ngày càng tăng của người dùng và mở ra nhiều cơ hội phát triển trong tương lai.

Bên cạnh đó, trải nghiệm người dùng luôn cũng là một yếu tố quan trọng mà nhóm nghiên cứu quan tâm đến, cũng như đặt mục tiêu trong việc tạo ra một giao diện thân thiện và dễ sử dụng. Ứng dụng chẩn đoán cung cấp một giao diện đơn giản, trực quan mà còn đảm bảo rằng người dùng có thể dễ dàng chụp và tải ảnh lên để nhận kết quả chẩn đoán. Quá trình tương tác diễn ra mượt mà, từ việc tải ảnh đến nhận kết quả, tất cả

đều được xử lý nhanh chóng nhờ sự kết hợp hiệu quả giữa ứng dụng và server. Firebase đóng vai trò quan trọng trong việc đảm bảo dữ liệu người dùng được lưu trữ an toàn, giúp người dùng yên tâm về tính bảo mật. Điều này tạo ra một trải nghiệm người dùng liền mạch và đáng tin cậy, góp phần gia tăng sự hài lòng ở phía đầu cuối. Khả năng phản hồi nhanh chóng từ hệ thống giúp cũng là một điểm đánh giá có ý nghĩa lớn, giúp người dùng cảm thấy ứng dụng đáp ứng kịp thời các nhu cầu cá nhân . Với những đặc điểm này, trải nghiệm người dùng trong dự án không chỉ được đảm bảo về mặt kỹ thuật mà còn tạo ra giá trị thực tế, mang lại sự hài lòng và thoải mái cho người dùng trong quá trình sử dụng ứng dụng.

## CHƯƠNG V: KẾT LUẬN VÀ ĐÁNH GIÁ

### 1. ĐÁNH GIÁ TỔNG QUAN VỀ ĐỀ TÀI

Khi sức khỏe xuất hiện những biểu hiện bất thường, sẽ có vô số những nguyên nhân đòi hỏi sự xem xét kỹ lưỡng để xác định nguồn gốc của các biểu hiện ấy, bao gồm cả việc nổi mụn nói chung. Đáng nói hơn, đây là bài toán gây ra sự khó chịu thường xuyên cho nhóm đối tượng trong đối tượng đây, phần lớn vì sự không ổn định hormone trong quá trình phát triển. Việc có một công cụ hỗ trợ nhận diện các loại mụn có nhiều điểm tương đồng với nhau, có tác động tích cực đến việc thuyên giảm tình trạng mất thẩm mỹ trên da mặt. Đây cũng là mục tiêu quan trọng nhất của đề tài, và đã được dành thời gian dài nghiên cứu và kiểm tra. Sản phẩm cuối cùng của nhóm nghiên cứu là một ứng dụng có thể từ hình ảnh cung cấp của người dùng, phát hiện và phân loại năm loại mụn trứng cá thường gặp (mụn đầu đen, mụn đầu trắng, mụn bọc, mụn sần, mụn mủ). Người dùng sau đó có thể thực hiện làm một bài trắc nghiệm đơn giản để hỗ trợ việc xác định nguyên nhân chủ yếu khiến mụn phát triển, cũng như tham khảo một vài hoạt chất phù hợp cho từng tình trạng mụn. Với kết quả khả quan về riêng Accuracy của hai mô hình phát hiện YOLOv8 và mô hình phân loại ResNet50 lần lượt là 79% và 91%, nhóm nghiên cứu có niềm tin kết quả cuối cùng sẽ khiến người dùng hài lòng khi dễ dàng xác định được những vấn đề mụn đã đề cập. Nhóm sinh viên cũng tin rằng kết quả của đề tài này sẽ có thể làm một đề tài tham khảo về những phương pháp nghiên cứu, hướng giải quyết và góp phần xác định sự khả thi tích cực cho các nghiên cứu có liên quan trong tương lai.

### 2. HẠN CHẾ CỦA ĐỀ TÀI

Trong quá trình xây dựng tập dữ liệu, nhóm nghiên cứu đã gặp phải một số các thách thức. Một trong những vấn đề chính là việc tập dữ liệu còn chứa đựng nhiều mẫu có độ phân giải thấp, điều này ảnh hưởng trực tiếp đến chất lượng của mô hình. Các ảnh không đạt về chất lượng hoặc kích thước không chỉ làm giảm độ chi tiết của các đối tượng cần nhận diện, mà còn khiến cho quá trình tiền xử lý trở nên phức tạp hơn. Bên cạnh đó, nhóm cũng nhận thấy rằng một số ảnh trong tập dữ liệu bị lỗi, như bị mờ, mất góc, hoặc

chứa các yếu tố nhiễu không mong muốn. Những ảnh này, nếu không được xử lý đúng cách, sẽ làm giảm hiệu suất của mô hình.

Về phần mô hình máy học, mặc dù đã được tối ưu hóa qua nhiều giai đoạn, nhưng kết quả đạt được vẫn chưa thực sự thỏa mãn. Độ tin cậy của các đối tượng được định vị trong ảnh chưa đủ cao, và vẫn tồn tại một số sai sót trong việc nhận diện đối tượng. Điều này có thể do sự bất cân xứng trong tập dữ liệu, hoặc do các ảnh đầu vào chưa đủ chất lượng, cũng như yêu cầu về số lượng đối tượng nhiều hơn. Hơn nữa, còn một số trường hợp thiếu số mà mô hình không thể nhận diện được đối tượng trong ảnh, làm giảm hiệu suất chung của hệ thống. Đây là một vấn đề cần lưu ý khi đưa mô hình vào bối cảnh ứng dụng thực tế, đặc biệt là trong các tình huống đòi hỏi độ chính xác cao.

Trong quá trình nghiên cứu, nhiều ý tưởng mới và các chức năng tiềm năng đã nảy sinh. Những ý tưởng này có thể giúp cải thiện hiệu suất của mô hình, tăng cường khả năng nhận diện và xử lý dữ liệu. Tuy nhiên, do hạn chế về nguồn lực và thời gian, nhóm nghiên cứu vẫn chưa có khả năng thực hiện chúng. Việc triển khai các ý tưởng này đòi hỏi phải có sự đầu tư lớn về công nghệ và nhân lực, cũng như cần có sự hiểu biết sâu hơn về các kỹ thuật tiên tiến trong lĩnh vực trí tuệ nhân tạo và học máy. Dù vậy, nhóm vẫn lạc quan về tiềm năng của các ý tưởng này và tin rằng, với sự hỗ trợ và điều kiện phù hợp, chúng sẽ có thể được hiện thực hóa trong tương lai gần, đóng góp đáng kể vào sự phát triển của dự án.

### **3. HƯỚNG PHÁT TRIỂN CHO CÁC NGHIÊN CỨU TIẾP THEO**

Mặc dù ứng dụng chẩn đoán các bệnh về da dựa trên mô hình máy học đã đạt được những kết quả khả quan, không thể phủ nhận sự tồn tại của một số hạn chế có thể cải thiện trong tương lai để nâng cao độ chính xác và hiệu quả của hệ thống. Một trong những thách thức lớn là khả năng nâng cao độ chính xác của mô hình. Để cải thiện hiệu suất phát hiện và phân loại các loại mụn, cần tập trung vào việc tăng cường dữ liệu huấn luyện bằng cách thu thập và sử dụng các mẫu hình ảnh đa dạng và chất lượng cao hơn. Điều này không chỉ giúp mô hình học máy tiếp cận với nhiều tình huống khác nhau mà