

ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA TOÁN - CƠ - TIN HỌC



**Phát hiện và chống giả mạo
trong nhận diện khuôn mặt**

Môn học: Nhập môn Trí tuệ nhân tạo

Sinh viên

Nguyễn Khánh Huyền - 21000271

Tôn Nữ Mai Khanh - 21000410

Giảng viên

TS. Nguyễn Hải Vinh

Tháng 11, 2024

Lời cảm ơn

Chúng em xin gửi lời cảm ơn chân thành đến Tiến sĩ Nguyễn Hải Vinh, giảng viên môn học Nhập môn Trí tuệ Nhân tạo, đã tận tình hướng dẫn và tạo điều kiện thuận lợi cho chúng em trong quá trình thực hiện đề tài này. Nhờ những bài giảng đầy tâm huyết từ thầy và trợ giảng, chúng em đã có cơ hội tiếp cận và nghiên cứu sâu hơn về các phương pháp hiện đại trong lĩnh vực trí tuệ nhân tạo. Những kiến thức và kỹ năng thu nhận được từ môn học không chỉ giúp chúng em hoàn thành tốt đề tài mà còn mở ra những định hướng mới trong việc ứng dụng trí tuệ nhân tạo vào thực tiễn.

Chúng em cũng xin cảm ơn các thầy cô và bạn bè đã đồng hành, hỗ trợ và góp ý trong suốt quá trình làm việc, giúp chúng em hoàn thiện báo cáo này.

Cuối cùng, chúng em hy vọng rằng báo cáo sẽ thể hiện được nỗ lực nghiên cứu của cả nhóm và rất mong nhận được ý kiến đóng góp từ thầy để hoàn thiện thêm kiến thức trong lĩnh vực này.

Trân trọng,

Tóm tắt

Face anti-spoofing là một yếu tố then chốt để bảo vệ các hệ thống nhận diện khuôn mặt khỏi các cuộc tấn công giả mạo như ảnh in, video phát lại, mặt nạ 3D, hay deepfake. Khi công nghệ nhận diện khuôn mặt ngày càng được sử dụng rộng rãi trong các lĩnh vực như bảo mật, thanh toán và giám sát, nguy cơ bị xâm nhập và giả mạo cũng gia tăng nhanh chóng. Các tấn công giả mạo tinh vi đặt ra thách thức lớn, làm giảm hiệu quả và tính an toàn của hệ thống, khiến việc nâng cấp khả năng chống giả mạo trở nên cấp thiết. Tuy nhiên, các phương pháp và mô hình dựa trên CNN truyền thống vẫn chưa đủ mạnh để xử lý những thách thức giả mạo phức tạp hiện nay. Để giải quyết vấn đề này, nhóm chúng em đề xuất một phương pháp kết hợp hai mô hình học sâu tiên tiến. Đầu tiên, mô hình YOLOv9 được sử dụng để phát hiện và định vị khuôn mặt một cách chính xác trong khung hình. Tiếp theo, mô hình ResNet-18 tích hợp cơ chế Attention nhằm tập trung vào các đặc trưng quan trọng, giúp nâng cao khả năng phân loại khuôn mặt thật và giả. Sự kết hợp giữa hai mô hình này không chỉ cải thiện độ chính xác mà còn tăng tốc độ và hiệu quả phát hiện giả mạo. Với giải pháp này, chúng em hy vọng mang đến một công cụ đáng tin cậy để bảo vệ các hệ thống nhận diện khuôn mặt trước những thách thức giả mạo ngày càng tinh vi.

Keywords: Face anti-spoofing, YOLOv9, Resnet18, Attention

Mục lục

Lời cảm ơn	i
Tóm tắt	ii
Mục lục	iv
Danh sách hình vẽ	v
Danh sách bảng	vi
1 Giới thiệu bài toán	1
1.1 Đặt vấn đề	1
1.2 Đề xuất	1
2 Cơ sở lý thuyết	3
2.1 Mô hình YOLO	3
2.1.1 Kiến trúc YOLO	3
2.1.2 Cơ sở lý thuyết	4
2.1.3 YOLOv9	6
2.2 Mô hình ResNet18	7
2.2.1 Mạng nơron tích chập thông thường	7
2.2.2 Mạng ResNet	8
2.2.3 Mạng ResNet18	9
2.3 Cơ chế Self-Attention trong thị giác máy tính	9
3 Phương pháp đề xuất	11
3.1 Quy trình chính	11
3.2 Quy trình nhận diện khuôn mặt	12
3.3 Quy trình phân loại giả mạo	12
3.3.1 Tiền xử lý dữ liệu	13
3.3.2 Kiến trúc mạng Resnet18 kết hợp cơ chế Attention	13
4 Thí nghiệm	15
4.1 Bộ dữ liệu	15
4.1.1 Đối với quá trình huấn luyện và kiểm tra mô hình	15
4.1.2 Đối với quá trình so sánh	17
4.2 Môi trường cài đặt	17
4.3 Phương pháp đánh giá mô hình	18
4.3.1 Phương pháp đánh giá cho bài toán nhận diện khuôn mặt	18
4.3.2 Phương pháp đánh giá cho phân loại giả mạo	19

5	Thực nghiệm và đánh giá	21
5.1	Đánh giá mô hình	21
5.1.1	YOLOv9	21
5.1.2	Resnet18 + Attention	22
5.2	So sánh mô hình	24
5.2.1	Mô hình YOLOv9 và phương pháp Haar Cascade	24
5.2.2	Mô hình Resnet18+Attention và mô hình Resnet18	25
5.3	Áp dụng vào Webcam	26
6	Kết luận	28
6.1	Kết luận	28
6.2	Hướng phát triển	28
Tài Liệu Tham Khảo		30

Danh sách hình vẽ

2.1	Kiến trúc YOLO	3
2.2	Cách YOLO hoạt động [7]	4
2.3	Non-Maximum Suppression [7]	6
2.4	Kiến trúc PGI trong YOLOv9 [9]	6
2.5	Kiến trúc GELAN trong YOLOv9 [9]	7
2.6	Kiến trúc gốc Resnet18 [6]	7
2.7	Luồng CNN	8
2.8	Residual Block trong ResNet	9
2.9	Kiến trúc ResNet-18	9
2.10	Cơ chế Attention	10
3.1	Quy trình chính của phương pháp đề xuất	11
3.2	Quá trình huấn luyện YOLOv9	12
3.3	Quy trình phân loại thật-giả của mô hình Resnet18+Att [5]	12
4.1	Ví dụ 2 mẫu dữ liệu của tập huấn luyện Face Recognition	16
4.2	Ví dụ 2 mẫu thuộc 1 người trong NUAA Photograph Imposter	17
4.3	Ví dụ về một biểu đồ đường cong PR [8]	18
4.4	Ví dụ biểu đồ EER [3]	19
5.1	Đường cong PR trên tập xác minh	22
5.2	Đường cong ROC trên tập xác minh	23
5.3	Confusion Matrix của mô hình	24
5.4	Biểu đồ ROC cho hai mô hình	26
5.5	Kết quả nhận diện đúng	27
5.6	Kết quả nhận diện giả mạo (1)	27
5.7	Kết quả nhận diện giả mạo (2)	27

Danh sách bảng

4.1	Số lượng mẫu trong tập huấn luyện, xác minh và kiểm tra của tập huấn luyện Face Recognition	16
4.2	Số lượng mẫu trong tập huấn luyện và kiểm tra trong NUAA Photograph Imposter .	17
5.1	Kết quả đánh giá các độ đo của mô hình	22
5.2	Kết quả trên hai tập dữ liệu kiểm tra	25
5.3	Kết quả đánh giá trên LCC_FASD	25

Chương 1

Giới thiệu bài toán

1.1 Đặt vấn đề

Nhận diện khuôn mặt đã trở thành phương thức xác thực sinh trắc học phổ biến nhờ tính tiện lợi và khả năng thu thập dữ liệu không xâm lấn. Tuy nhiên, hệ thống này dễ bị tấn công giả mạo (spoofing), như sử dụng ảnh in, video phát lại, hoặc mặt nạ 3D, làm giảm tính đáng tin cậy. Để đảm bảo an toàn, việc phát triển các phương pháp tự động phát hiện tấn công giả mạo (PAD) là rất quan trọng.

Nhiều thuật toán PAD truyền thống dựa vào các đặc trưng thủ công, như màu sắc, kết cấu, và chuyển động, tận dụng sự suy giảm chất lượng trong quá trình tái tạo để phát hiện giả mạo. Gần đây, các mô hình CNN đã nổi lên, giúp học đặc trưng tự động và đạt hiệu suất tốt hơn trong nội bộ tập dữ liệu. Tuy nhiên, những mô hình này thường không thể tổng quát hóa tốt trên các tập dữ liệu khác hoặc các kiểu tấn công chưa từng thấy, do hiện tượng overfitting từ dữ liệu huấn luyện hạn chế.

Các nghiên cứu cho thấy rằng nhiệm vụ phụ trợ, như giám sát độ sâu (depth supervision), có thể cải thiện hiệu suất PAD. Thay vì dựa vào việc tạo bản đồ độ sâu 3D phức tạp, giám sát nhị phân theo từng điểm ảnh (deep pixel-wise binary supervision) là một hướng đi đơn giản và hiệu quả hơn, cải thiện khả năng nhận diện tấn công giả mạo mà không cần tạo thêm dữ liệu phức tạp.

1.2 Đề xuất

Trong báo cáo này, nhóm giới thiệu một khuôn khổ phát hiện tấn công giả mạo sử dụng hai mô hình mạnh mẽ là YOLO và ResNet18 kết hợp với cơ chế chú ý (attention).

Cả YOLO và ResNet18 đều là những kiến trúc mạng nơ-ron tích chập (CNN) tiên tiến, với YOLO nổi bật nhờ khả năng phát hiện đối tượng trong các bối cảnh phức tạp và ResNet18 nổi bật với khả năng học các đặc trưng sâu sắc từ các dữ liệu phức tạp. Cơ chế chú ý được áp dụng trong ResNet18 để tăng cường khả năng tập trung vào các vùng quan trọng trong hình ảnh khuôn mặt, từ đó giúp mô hình phát hiện các dấu hiệu của tấn công giả mạo với độ chính xác cao hơn. Sự kết hợp này không chỉ nâng cao khả năng phân biệt giữa khuôn mặt thật và các tấn công giả mạo, mà còn cải thiện khả năng tổng quát của hệ thống qua các bộ dữ liệu khác nhau.

Mô hình YOLO được đào tạo và đánh giá dựa trên bộ dữ liệu Face Recognition for CV (phiên bản 1), còn mô hình ResNet18 kết hợp Attention được chạy trên bộ dữ liệu NUAA Photograph Imposter. Kết quả từ các thử nghiệm này cho thấy khả năng phát hiện chính xác và tính khả thi của phương pháp khi áp dụng trong các tình huống thực tế. Các hình ảnh mẫu của các khuôn mặt trong

các bộ dữ liệu này được trình bày để minh họa sự khác biệt giữa khuôn mặt thật và các tấn công giả mạo.

Chương 2

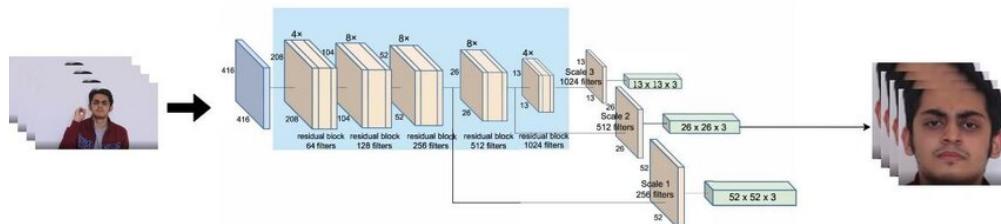
Cơ sở lý thuyết

Hệ thống được đề xuất sẽ gồm 2 phần chính:

- Sử dụng mô hình YOLO để nhận diện khuôn mặt
- Với output từ mô hình YOLO, sử dụng ResNet18 với Attention để phát hiện giả mạo

2.1 Mô hình YOLO

2.1.1 Kiến trúc YOLO



Hình 2.1: Kiến trúc YOLO

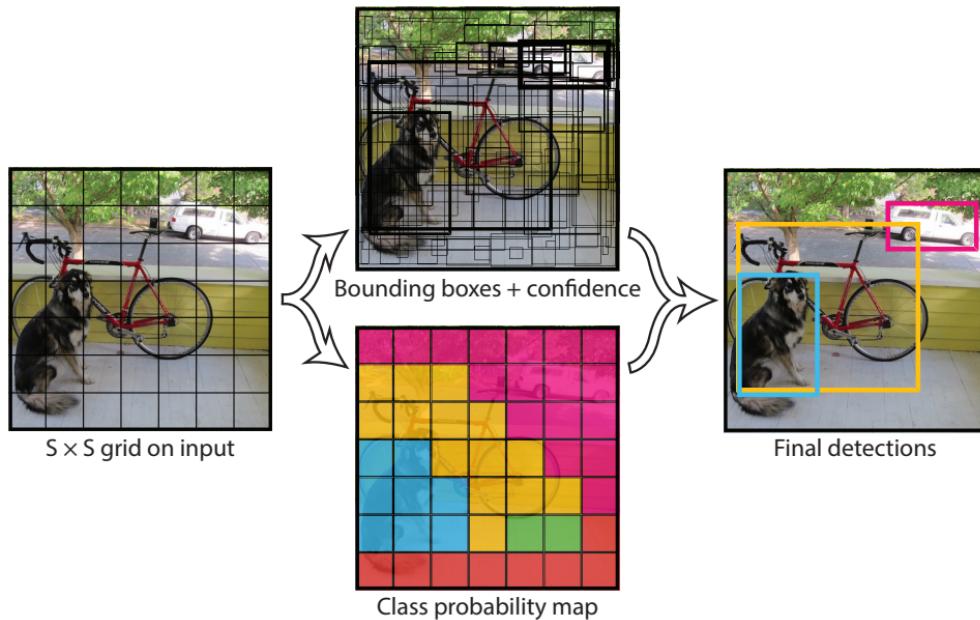
YOLO (You Only Look Once) [7] là một kiến trúc mạng nơ-ron tích chập (CNN) tiên tiến, được thiết kế để phát hiện và nhận diện đối tượng trong hình ảnh với tốc độ nhanh và độ chính xác cao. Trong nhận diện khuôn mặt, YOLO sử dụng một mạng CNN duy nhất để xử lý toàn bộ hình ảnh đầu vào. Quá trình này bắt đầu bằng việc mạng trích xuất các đặc trưng quan trọng từ hình ảnh, chẳng hạn như các chi tiết liên quan đến cấu trúc khuôn mặt. Tiếp theo, hình ảnh được chia thành một lưới với kích thước cố định (ví dụ: 13x13 hoặc 19x19), trong đó mỗi ô lưới chịu trách nhiệm dự đoán các hộp giới hạn (bounding boxes) và xác suất hiện diện của khuôn mặt trong khu vực đó.

YOLO dự đoán nhiều hộp giới hạn cho mỗi ô lưới, với mỗi hộp được đặc trưng bởi tọa độ tâm, chiều rộng, chiều cao, và độ tin cậy. Đồng thời, hệ thống cũng tính toán xác suất để đối tượng trong mỗi hộp thuộc về một lớp cụ thể, chẳng hạn như "khuôn mặt". Sau đó, quá trình hậu xử lý được thực hiện nhằm loại bỏ các hộp có độ tin cậy thấp và áp dụng kỹ thuật Non-Maximum Suppression (NMS) để giảm trùng lặp, chỉ giữ lại các hộp có độ tin cậy cao nhất đại diện cho mỗi khuôn mặt.

Cuối cùng, các hộp giới hạn được gán nhãn "khuôn mặt" và hiển thị trực tiếp trên hình ảnh hoặc video. Kết quả bao gồm vị trí và kích thước của các khuôn mặt được phát hiện.

2.1.2 Cơ sở lý thuyết

Bounding Box và Cách YOLO dự đoán



Hình 2.2: Cách YOLO hoạt động [7]

Trong YOLO, ảnh đầu vào được chia thành một lưới $S \times S$ (ví dụ, 7×7 cho YOLO ban đầu). Mỗi ô lưới này có nhiệm vụ phát hiện các đối tượng trong phạm vi của nó. Mỗi ô lưới dự đoán B hộp giới hạn (bounding boxes) và cho mỗi hộp dự đoán, mô hình tính toán một số thông số:

- x và y : Tọa độ của tâm hộp giới hạn, được tính theo tỷ lệ với kích thước của ô lưới.
- w và h : Chiều rộng và chiều cao của hộp giới hạn, cũng tính theo tỷ lệ với kích thước ô lưới.
- **Confidence score:** Mức độ tin cậy rằng hộp giới hạn chứa một đối tượng. Đây là một giá trị trong khoảng từ 0 đến 1, được tính theo công thức:

$$C = P(\text{Object}) \times \text{IoU}_{\text{pred}, \text{truth}}$$

Trong đó, $\text{IoU}_{\text{pred}, \text{truth}}$ là chỉ số giao nhau trên toàn bộ (Intersection over Union, IoU) giữa hộp dự đoán và hộp thật.

- **Class Probabilities:** Mỗi ô lưới cũng dự đoán xác suất cho các lớp đối tượng khác nhau (ví dụ: người, xe, động vật,...).

Intersection over Union (IoU)

IoU là chỉ số đo lường mức độ trùng khớp giữa hai hộp giới hạn. Đối với mỗi hộp giới hạn dự đoán, ta tính IoU với hộp giới hạn thực tế để xác định độ chính xác của dự đoán. Công thức tính IoU giữa hai hộp giới hạn A và B là:

$$\text{IoU}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Trong đó:

- $|A \cap B|$ là diện tích giao nhau của hai hộp.
- $|A \cup B|$ là diện tích hợp nhất của hai hộp.

IoU được sử dụng để đánh giá việc một hộp giới hạn có đúng với đối tượng hay không, và là cơ sở để xác định lỗi khi tính toán hàm loss.

Hàm Loss trong YOLO

Hàm loss trong YOLO bao gồm ba thành phần chính: *loss về tọa độ*, *loss về xác suất*, và *loss về phân loại*.

Loss về tọa độ:

Đây là sự sai lệch giữa các tọa độ x, y, w, h của hộp dự đoán và hộp thật. Công thức tính loss cho các tọa độ là:

$$\text{Loss}_{\text{coord}} = \sum_i \lambda_{\text{coord}} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (w_i - \hat{w}_i)^2 + (h_i - \hat{h}_i)^2]$$

Trong đó $\hat{x}_i, \hat{y}_i, \hat{w}_i, \hat{h}_i$ là các giá trị thực tế của các tọa độ, và λ_{coord} là một hệ số trọng cho việc tối ưu hóa.

Loss về xác suất:

Phần này xử lý sai số giữa độ tin cậy dự đoán và giá trị thực tế (1 nếu có đối tượng, 0 nếu không có đối tượng trong hộp). Công thức tính là:

$$\text{Loss}_{\text{conf}} = \sum_i [C_i - \hat{C}_i]^2$$

Trong đó C_i là độ tin cậy dự đoán và \hat{C}_i là độ tin cậy thực tế.

Loss về phân loại:

Đo lường sự sai lệch giữa xác suất phân loại dự đoán và xác suất thực tế cho các lớp đối tượng. Hàm loss này sử dụng **Cross-Entropy Loss**:

$$\text{Loss}_{\text{class}} = - \sum_{c=1}^C \hat{p}_c \log(p_c)$$

Trong đó \hat{p}_c là xác suất phân loại thực tế của lớp c , và p_c là xác suất dự đoán cho lớp đó.

Tổng hợp tất cả các thành phần lại, hàm loss tổng quát của YOLO là:

$$\text{Loss} = \text{Loss}_{\text{coord}} + \text{Loss}_{\text{conf}} + \text{Loss}_{\text{class}}$$

Kết quả đầu ra của YOLO

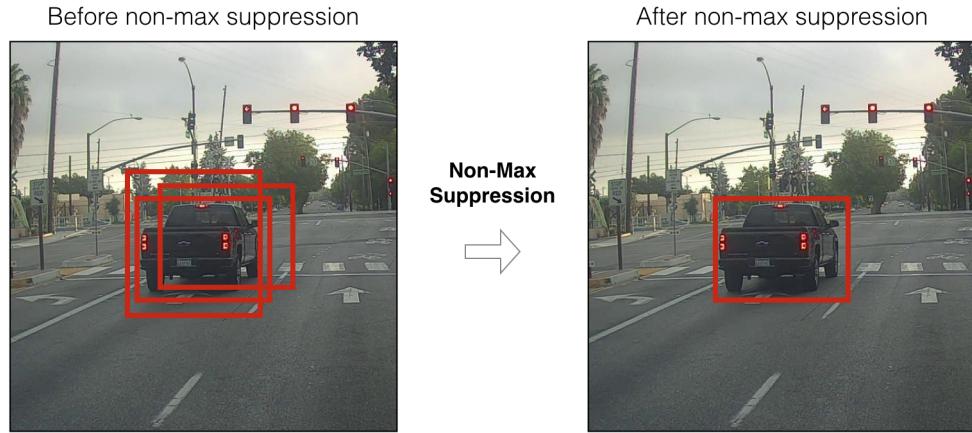
Đầu ra của YOLO cho mỗi ô lưới là một vector chứa các giá trị sau:

- B hộp giới hạn: Mỗi hộp bao gồm 5 giá trị: x, y, w, h và độ tin cậy C .
- Xác suất phân loại: Cho mỗi lớp đối tượng, YOLO dự đoán một xác suất phân loại.

Mô hình YOLO học cách tối ưu hóa các tham số này để giảm thiểu hàm loss tổng quát và đảm bảo phát hiện đối tượng nhanh chóng với độ chính xác cao.

Non-Maximum Suppression (NMS)

Sau khi YOLO dự đoán các bounding boxes, để giảm thiểu sự trùng lặp giữa các hộp giới hạn (duplicate detections), **Non-Maximum Suppression (NMS)** được áp dụng. NMS loại bỏ các hộp có sự chồng lấp cao (IoU cao) và chỉ giữ lại hộp có độ tin cậy cao nhất.

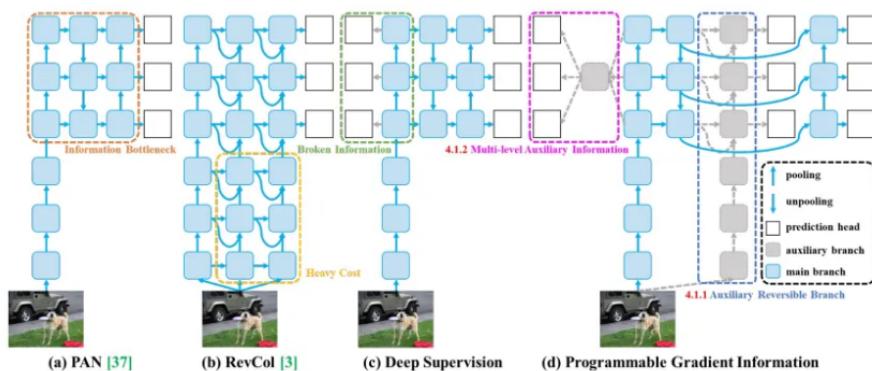


Hình 2.3: Non-Maximum Suppression [7]

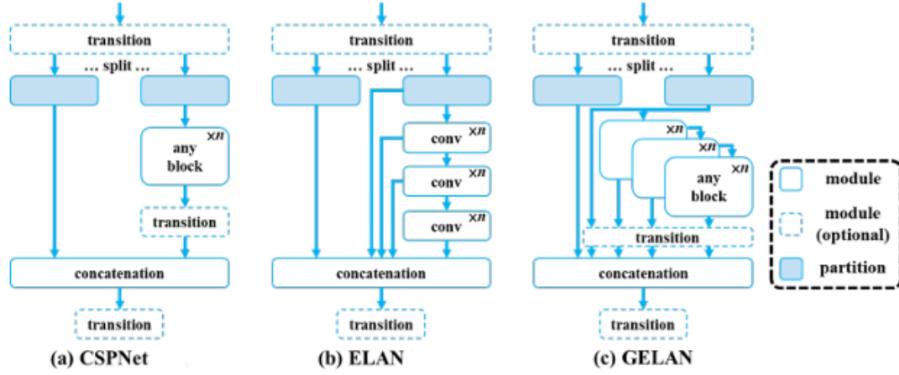
2.1.3 YOLOv9

YOLOv9 được phát hành vào tháng 2 năm 2024 như một bước tiến lớn sau thành công của YOLOv8. Những đổi mới chính bao gồm Programmable Gradient Information (PGI) và Generalized Efficient Layer Aggregation Network (GELAN), cả hai đều cải thiện đáng kể việc trích xuất đặc trưng, luồng gradient, và hiệu suất của mạng. [9]

Khác với YOLOv8, vốn tập trung tối ưu hóa tốc độ và độ chính xác thông qua CSPNet và PANet nâng cao, YOLOv9 giới thiệu hai yếu tố kiến trúc mới: Programmable Gradient Information (PGI) (Hình 2.4) và Generalized Efficient Layer Aggregation Network (GELAN) (Hình 2.5). Những cải tiến này nhằm đến vấn đề cốt lõi về mất mát thông tin khi dữ liệu đi qua các lớp của mạng, từ đó cải thiện độ ổn định của gradient và độ chính xác của dự đoán.



Hình 2.4: Kiến trúc PGI trong YOLOv9 [9]



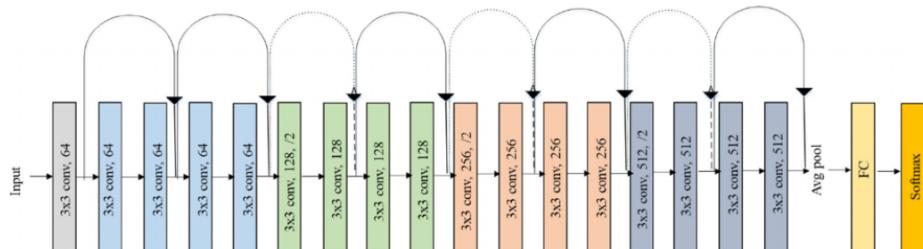
Hình 2.5: Kiến trúc GELAN trong YOLOv9 [9]

2.2 Mô hình ResNet18

ResNet (Residual Network) là một kiến trúc mạng nơ-ron sâu được thiết kế để giải quyết vấn đề biến mất gradient (vanishing gradient), thường gặp trong việc đào tạo các mạng nơ-ron sâu. Kiến trúc ResNet, đặc biệt là phiên bản ResNet18, đã được ứng dụng rộng rãi trong nhiều bài toán xử lý hình ảnh, bao gồm phát hiện giả mạo khuôn mặt (Presentation Attack Detection - PAD).

ResNet sử dụng các khối dư (Residual Blocks) để giúp mạng học các đặc trưng hiệu quả hơn. Trong mỗi khối dư, thay vì học trực tiếp ánh xạ đầu vào sang đầu ra, mạng học "phân dư" (residual mapping) giữa đầu vào và đầu ra. Phần dư này được kết hợp với đầu vào thông qua kết nối tắt (skip connection). Điều này giúp ResNet đạt được khả năng đào tạo mạng rất sâu mà không làm mất thông tin hoặc gặp khó khăn trong hội tụ.

Kiến trúc của ResNet-18 gốc [6] được minh họa trong Hình 2.6. Mạng này có tổng cộng 18 lớp (17 lớp tích chập, lớp fully-connected, và lớp softmax bổ sung để thực hiện nhiệm vụ phân loại). Các lớp tích chập sử dụng bộ lọc kích thước 3×3 , và mạng được thiết kế sao cho nếu bản đồ đặc trưng đầu ra (output feature map) có cùng kích thước, thì các lớp có cùng số lượng bộ lọc. Tuy nhiên, số lượng bộ lọc sẽ tăng gấp đôi nếu kích thước bản đồ đặc trưng đầu ra giảm một nửa. Việc giảm kích thước (downsampling) được thực hiện bởi các tầng tích chập có stride bằng 2.



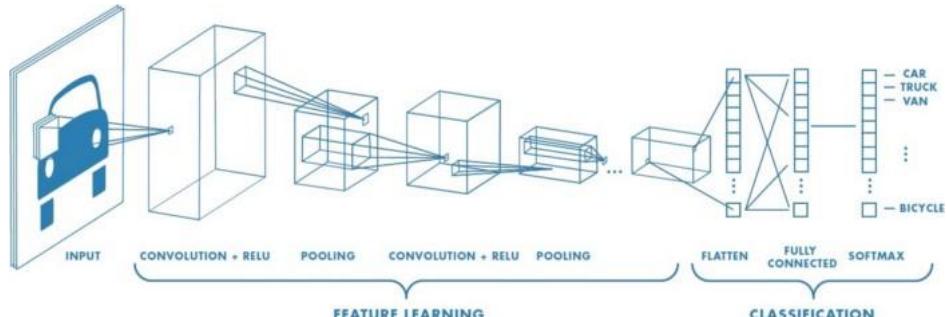
Hình 2.6: Kiến trúc gốc Resnet18 [6]

2.2.1 Mạng nơron tích chập thông thường

Trong mạng neural, mô hình mạng neural tích chập (CNN) là 1 trong những mô hình để nhận dạng và phân loại hình ảnh. Trong đó, xác định đối tượng và nhận dạng khuôn mặt là 1 trong số những lĩnh vực mà CNN được sử dụng rộng rãi.

CNN phân loại hình ảnh bằng cách lấy 1 hình ảnh đầu vào, xử lý và phân loại nó theo các hạng mục nhất định. Máy tính coi hình ảnh đầu vào là 1 mảng pixel và nó phụ thuộc vào độ phân giải của hình ảnh. Dựa trên độ phân giải hình ảnh, máy tính sẽ thấy $H \times W \times D$ (H : Chiều cao, W : Chiều rộng, D : Độ dày).

Về kỹ thuật, mô hình CNN để training và kiểm tra, mỗi hình ảnh đầu vào sẽ chuyển nó qua 1 loạt các lớp tích chập với các bộ lọc (Kernels), tổng hợp lại các lớp được kết nối đầy đủ (Full Connected) và áp dụng hàm Softmax để phân loại đối tượng có giá trị xác suất giữa 0 và 1. Hình dưới đây là toàn bộ luồng CNN để xử lý hình ảnh đầu vào và phân loại các đối tượng dựa trên giá trị.



Hình 2.7: Luồng CNN

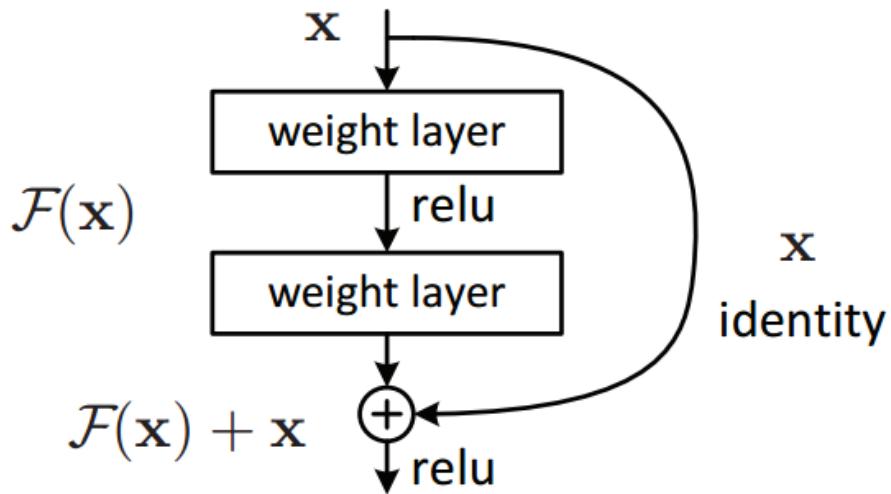
2.2.2 Mạng ResNet

Trong các mạng nơ-ron truyền thống, các lớp nối tiếp nhau theo kiểu tuyến tính (sequential) để tạo ra các kết quả cuối cùng. Các kết nối này thường có dạng sau:

- Input của lớp hiện tại là output của lớp trước đó.
- Mỗi lớp sẽ thực hiện một phép biến đổi nhất định (như convolution, activation, pooling, v.v.) và truyền kết quả tới lớp tiếp theo.

Các lớp trong mạng nơ-ron sâu có thể có nhiều tầng và sự biến đổi dần dần được thực hiện qua các lớp. Tuy nhiên, khi độ sâu của mạng tăng lên, sẽ xảy ra vấn đề Vanishing Gradient, làm cho quá trình huấn luyện trở nên khó khăn và không ổn định.

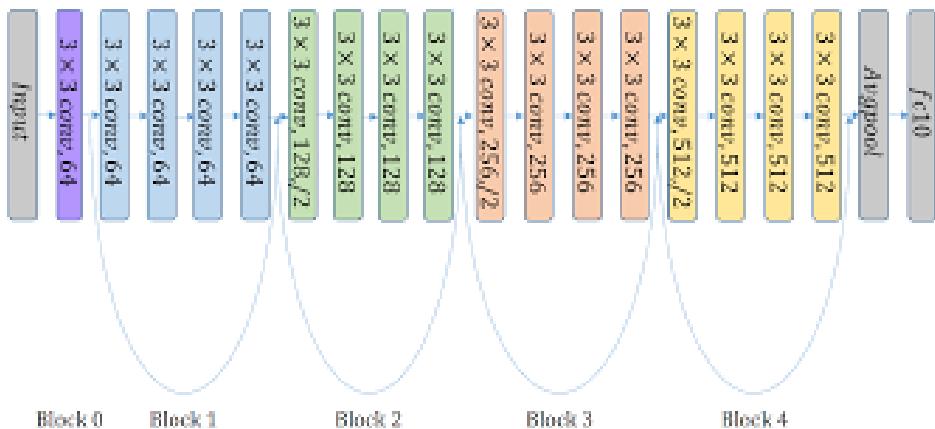
Hiện tượng Vanishing Gradient khiến các lớp ở sâu không nhận được cập nhật đủ mạnh để cải thiện trọng số, dẫn đến mô hình không hội tụ tốt. ResNet giải quyết vấn đề này bằng cách sử dụng các kết nối tắt (skip connections), bổ sung đầu vào ban đầu ($\text{input } x$) vào đầu ra của một lớp nhất định. Điều này đảm bảo rằng gradient có đường dẫn trực tiếp qua mạng, giúp mô hình học hiệu quả hơn.



Hình 2.8: Residual Block trong ResNet

2.2.3 Mạng ResNet18

ResNet-18 là một kiến trúc mạng nơ-ron sâu (deep neural network) thuộc dòng Residual Networks (ResNet), được thiết kế để giải quyết vấn đề của các mạng sâu trong việc bị mất thông tin do việc truyền qua quá nhiều lớp. ResNet-18 bao gồm 18 lớp học được huấn luyện, với các lớp còn lại là các lớp convolutional, batch normalization và ReLU.



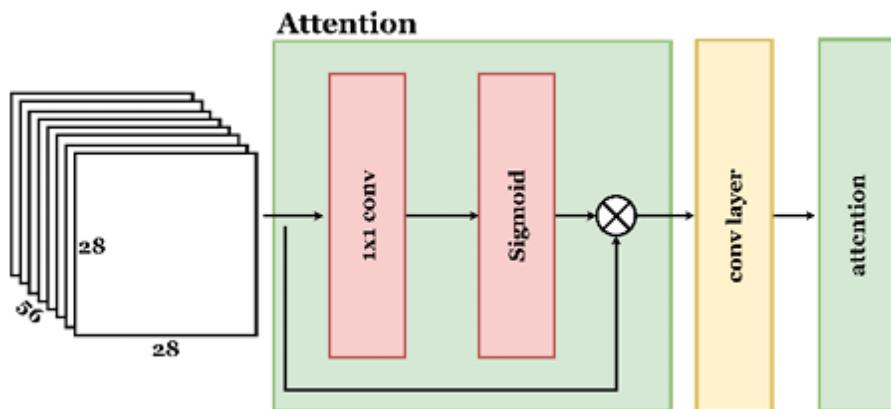
Hình 2.9: Kiến trúc ResNet-18

2.3 Cơ chế Self-Attention trong thị giác máy tính

Self-attention là một cơ chế quan trọng trong lĩnh vực thị giác máy tính (computer vision). Nó cho phép mô hình tập trung vào các phần quan trọng của hình ảnh, giúp cải thiện hiệu suất của các tác vụ như phân loại, nhận dạng và phát hiện đối tượng.

Cơ chế self-attention hoạt động bằng cách tính toán sự tương quan giữa các phần khác nhau của hình ảnh. Điều này giúp mô hình hiểu được mối quan hệ giữa các phần của hình ảnh và tập trung vào những phần quan trọng nhất. Một ví dụ điển hình của việc áp dụng self-attention trong

thị giác máy tính là mô hình Vision Transformer (ViT), trong đó self-attention được sử dụng để xử lý các patch của hình ảnh thay vì các pixel riêng lẻ.[4]



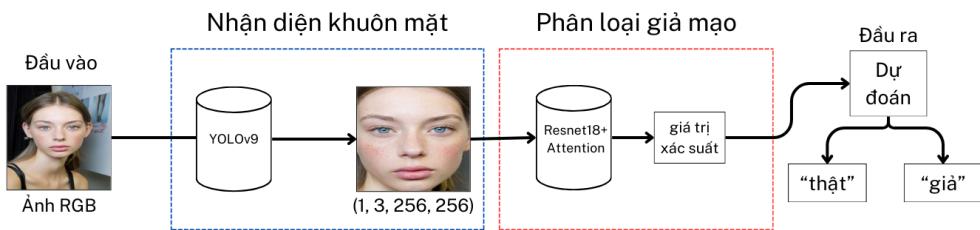
Hình 2.10: Cơ chế Attention

Chương 3

Phương pháp đề xuất

3.1 Quy trình chính

Quy trình tổng thể do chính nhóm em tự đề xuất với ý tưởng dựa trên bài báo nghiên cứu khoa học DeePixBis[2].



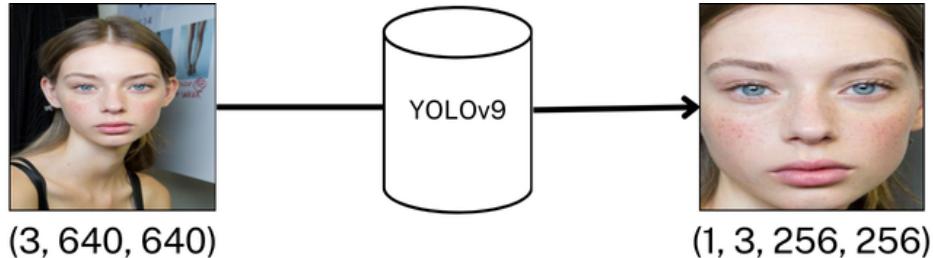
Hình 3.1: Quy trình chính của phương pháp đề xuất

Nhóm sử dụng mô hình YOLOv9[9] nhằm nhận diện khuôn mặt. Sau khi nhận diện được khuôn mặt, kết quả đầu ra từ mô hình YOLOv9 bao gồm các bounding box – các hộp giới hạn thể hiện vị trí chính xác của khuôn mặt trong khung hình. Nhờ các bounding box này, nhóm đã trích xuất khu vực chứa khuôn mặt, loại bỏ các chi tiết không cần thiết như nền, vật thể xung quanh hoặc các yếu tố gây nhiễu trong khung hình. Điều này giúp tập trung toàn bộ quá trình xử lý vào vùng mặt, giảm độ phức tạp tính toán và tăng tính chính xác của bước phân loại tiếp theo.

Sau khi khuôn mặt được trích xuất, vùng này được đưa vào mô hình ResNet18 kết hợp với cơ chế Attention [5] để tiến hành phân loại. ResNet18, với cấu trúc mạng sâu và khả năng học các đặc trưng phức tạp, đóng vai trò làm nền tảng để phân tích và trích xuất các đặc trưng quan trọng từ khu vực khuôn mặt. Để tăng cường khả năng của mô hình, tác giả đã tích hợp thêm cơ chế Attention, giúp mô hình tập trung vào các vùng hoặc đặc trưng quan trọng hơn trên khuôn mặt. Attention giúp mô hình bỏ qua các chi tiết ít liên quan và tập trung vào những đặc trưng quyết định, chẳng hạn như kết cấu da, độ sáng hoặc các đặc điểm khác biệt để phân biệt giữa khuôn mặt thật và giả mạo.

Kết quả đầu ra của mô hình ResNet18 kết hợp Attention là một xác suất, biểu thị khả năng liệu khuôn mặt đó là "thật" hay "giả mạo". Xác suất này sau đó được so sánh với một ngưỡng xác định trước. Nếu xác suất cao hơn ngưỡng, mô hình dự đoán đó là khuôn mặt thật. Ngược lại, nếu thấp hơn ngưỡng, mô hình sẽ xác định rằng đó là một cuộc tấn công giả mạo, chẳng hạn như ảnh in, video phát lại, hoặc mặt nạ 3D. Quy trình này đảm bảo rằng hệ thống có thể đưa ra kết luận cuối cùng một cách đáng tin cậy và chính xác.

3.2 Quy trình nhận diện khuôn mặt



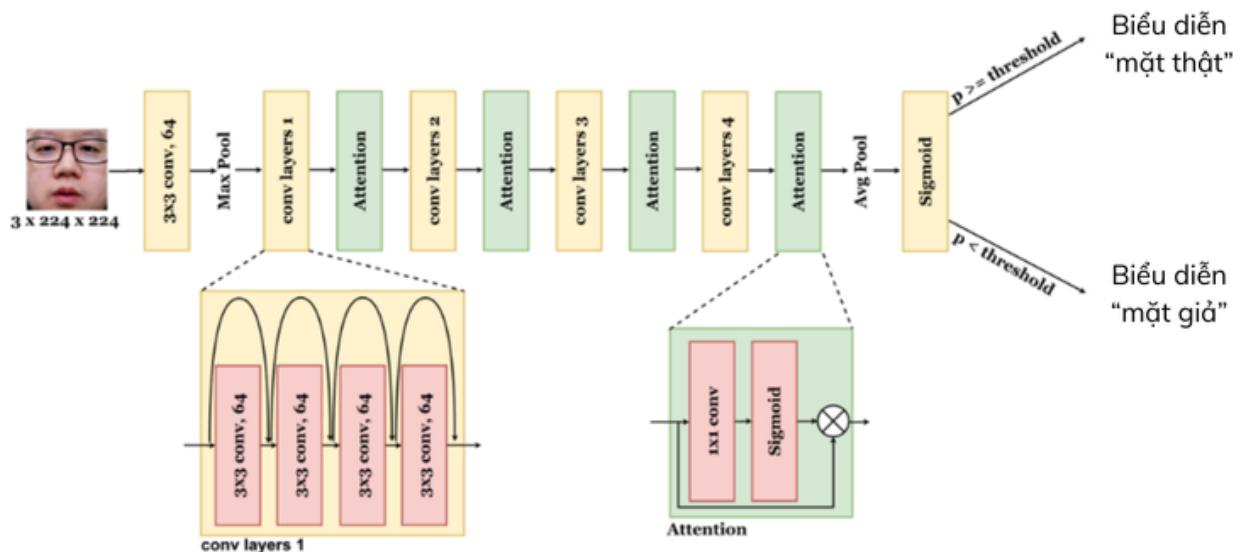
Hình 3.2: Quá trình huấn luyện YOLOv9

Trong nghiên cứu này, nhóm đã sử dụng thư viện Ultralytics, một công cụ mạnh mẽ và tiện lợi dành cho các bài toán thị giác máy tính, để khai thác trọng số pre-trained của mô hình YOLOv9. Trọng số pre-trained được sử dụng là yolov9c.pt. Đây là trọng số được huấn luyện sẵn trên tập dữ liệu COCO, một tập dữ liệu lớn và phổ biến cho các bài toán nhận diện vật thể.

Sau khi sử dụng YOLOv9 để phát hiện và cắt khuôn mặt từ ảnh đầu vào, nhóm đã chuẩn hóa kích thước ảnh đầu ra về 256x256 pixel. Kích thước này được lựa chọn phù hợp với yêu cầu đầu vào của mô hình Resnet18+Attention, nhằm đảm bảo sự tương thích và hiệu quả trong quá trình huấn luyện và phân loại.

3.3 Quy trình phân loại giả mạo

Quy trình được lấy trong bài [5], tác giả bài viết lấy ý tưởng dựa trên bài báo khoa học [1]



Hình 3.3: Quy trình phân loại thật-giả của mô hình Resnet18+Att [5]

3.3.1 Tiền xử lý dữ liệu

Trước khi đưa vào huấn luyện qua các lớp trong kiến trúc mô hình đề xuất như trên (Hình 3.3), nhóm sẽ thực hiện tiền xử lý hình ảnh đầu vào, đảm bảo dữ liệu đạt định dạng và chất lượng phù hợp với yêu cầu của các mô hình.

Đầu tiên, phần trung tâm của hình ảnh đầu vào được cắt ra với kích thước cố định là 224×224 . Việc cắt này giúp loại bỏ các vùng rìa không quan trọng và chuẩn hóa kích thước đầu vào. Lựa chọn cắt trung tâm giúp tập trung vào phần quan trọng nhất của hình ảnh, thường chứa các đặc trưng chính cần phân tích.

Tiếp theo, hình ảnh được chuyển đổi sang tensor, định dạng mà PyTorch yêu cầu cho các phép tính học sâu. Trong bước này, giá trị pixel của hình ảnh, vốn nằm trong khoảng [0,255], được chia tỷ lệ về khoảng [0,1] để cải thiện tính ổn định số học và tăng tốc độ hội tụ trong quá trình tối ưu hóa. Đồng thời, thứ tự các kênh màu được chuyển từ định dạng thông thường [H,W,C] (chiều cao, chiều rộng, kênh) sang [C,H,W] (kênh, chiều cao, chiều rộng), định dạng tiêu chuẩn của PyTorch.

Cuối cùng, các giá trị pixel được chuẩn hóa dựa trên giá trị trung bình và độ lệch chuẩn của tập dữ liệu ImageNet. Cụ thể, mỗi kênh màu (R, G, B) được chuẩn hóa bằng cách trừ đi giá trị trung bình tương ứng ([0.485,0.456,0.406]) và chia cho độ lệch chuẩn tương ứng ([0.229,0.224,0.225]). Việc chuẩn hóa này giúp đưa dữ liệu về phạm vi giá trị thống nhất, giảm thiểu sự khác biệt giữa các kênh màu và giúp mô hình tập trung vào các đặc trưng quan trọng hơn thay vì bị ảnh hưởng bởi giá trị tuyệt đối của pixel.

3.3.2 Kiến trúc mạng Resnet18 kết hợp cơ chế Attention

Quy trình 3.3 định nghĩa kiến trúc mạng ResNet18 và cơ chế attention tùy chỉnh để phân loại hình ảnh. Có tất cả ba phần chính: các khối cơ bản (Basic Blocks), cơ chế Self-Attention và kiến trúc tổng thể ResNet18.

Khối cơ bản không có attention

Các khối này được hiển thị trong "conv layers 1", "conv layers 2", "conv layers 3", và "conv layers 4" trên hình 3.3

Khối cơ bản này được thiết kế dựa trên kiến trúc truyền thống của ResNet. Nó bao gồm hai lớp tích chập (Convolutional layers), mỗi lớp có kernel kích thước 3×3 . Các lớp tích chập này được thiết kế với padding để giữ nguyên kích thước không gian của ảnh đầu ra. Sau mỗi lớp tích chập, dữ liệu được chuẩn hóa bằng Batch Normalization, giúp cải thiện tính ổn định của mạng và tăng tốc độ hội tụ trong quá trình huấn luyện. Ngoài ra, hàm kích hoạt ReLU được sử dụng để thêm tính phi tuyến, giúp mô hình học được các mối quan hệ phức tạp hơn từ dữ liệu.

Một thành phần quan trọng trong khối này là cơ chế shortcut connection. Shortcut đảm bảo đầu vào của khối được cộng trực tiếp với đầu ra, điều này giúp giảm thiểu vấn đề biến mất gradient khi độ sâu của mạng tăng. Nếu số lượng kênh đầu vào và đầu ra khác nhau hoặc kích thước không gian thay đổi (do sử dụng stride > 1), một lớp tích chập bổ sung với kernel 1×1 được sử dụng để điều chỉnh đầu vào sao cho phù hợp với đầu ra. Khối cơ bản này không có cơ chế attention, và đầu ra cuối cùng là đặc trưng đã được chuyển đổi thông qua tích chập, chuẩn hóa và shortcut.

Khối cơ bản có Attention

Cơ chế Attention được biểu diễn rõ ràng 3.3 trong khối ở phía sau mỗi lớp tích chập (từ "conv layers 1" đến "conv layers 4").

Các bước xử lý vẫn bao gồm hai lớp tích chập, chuẩn hóa và shortcut, nhưng sau khi hoàn tất các bước cơ bản, đầu ra sẽ được đưa qua một module Self-Attention. Self-Attention giúp mạng tập trung hơn vào các vùng quan trọng trong đặc trưng không gian của ảnh đầu ra.

Cơ chế Attention hoạt động bằng cách sử dụng một lớp tích chập 1×1 để học trọng số cho từng vị trí không gian trên đặc trưng. Trọng số này được chuẩn hóa thông qua hàm Sigmoid để đảm bảo giá trị nằm trong khoảng $[0,1]$. Cuối cùng, đặc trưng đầu ra được nhân với trọng số attention để làm nổi bật các vùng quan trọng, trong khi giảm thiểu ảnh hưởng của các vùng ít liên quan.

Kiến trúc tổng thể của ResNet18 với Attention

Mô hình bắt đầu bằng một lớp tích chập 3×3 , được áp dụng lên ảnh đầu vào để trích xuất đặc trưng ban đầu, kèm theo Batch Normalization và hàm kích hoạt ReLU.

Mạng được chia thành bốn tầng chính:

- **Layer 1:** Bao gồm hai khối cơ bản với 64 kênh đầu ra. Kích thước không gian của ảnh được giữ nguyên. Sau mỗi tầng, một module Self-Attention được áp dụng để tăng cường học tập.
- **Layer 2:** Bao gồm hai khối cơ bản với 128 kênh đầu ra. Tầng đầu tiên trong layer sử dụng stride = 2 để giảm kích thước không gian.
- **Layer 3 và Layer 4:** Tương tự như Layer 2 nhưng tăng dần số lượng kênh lên 256 và 512.

Sau khi đi qua các tầng này, dữ liệu được chuyển qua một lớp Adaptive Average Pooling để giảm chiều không gian về 1×1 trên mỗi kênh. Điều này đảm bảo rằng đầu ra chứa thông tin tổng quát của toàn bộ ảnh. Dữ liệu sau đó được đưa qua một lớp fully connected để tạo ra dự đoán cuối cùng.

Trong quá trình huấn luyện, để mô hình biết cách điều chỉnh trọng số và học phân loại tốt hơn, tác giả đã sử dụng hàm mất mát là Binary Cross Entropy with Logits Loss (BCEWithLogitsLoss). Hàm mất mát BCEWithLogitsLoss kết hợp giữa:

- Sigmoid Activation: Áp dụng hàm Sigmoid lên đầu ra của mô hình để chuyển đổi giá trị đầu ra thành xác suất p (giá trị nằm trong khoảng $[0,1]$).
- Binary Cross Entropy (BCE): Tính toán sai số dựa trên dự đoán xác suất và nhãn thực tế.

Để giải quyết vấn đề gradient vanishing và tăng hiệu quả huấn luyện, tác giả sử dụng hàm tối ưu Adam với learning_rate là 0.0001, giá trị nhỏ giúp đảm bảo hội tụ ổn định.

Cuối cùng, đầu ra của mô hình sẽ là một giá trị xác suất p , biểu diễn khả năng bức ảnh đầu vào là khuôn mặt thật. Một ngưỡng sẽ được ta chọn để xác định xem đây thuộc nhãn thật hay giả mạo. Trong nghiên cứu này, nhóm chọn ngưỡng là $p = 0.5$

- Nếu $p \geq 0.5$: Mô hình dự đoán đây là khuôn mặt thật.
- Nếu $p < 0.5$: Mô hình dự đoán đây là khuôn mặt giả mạo.

Chương 4

Thí nghiệm

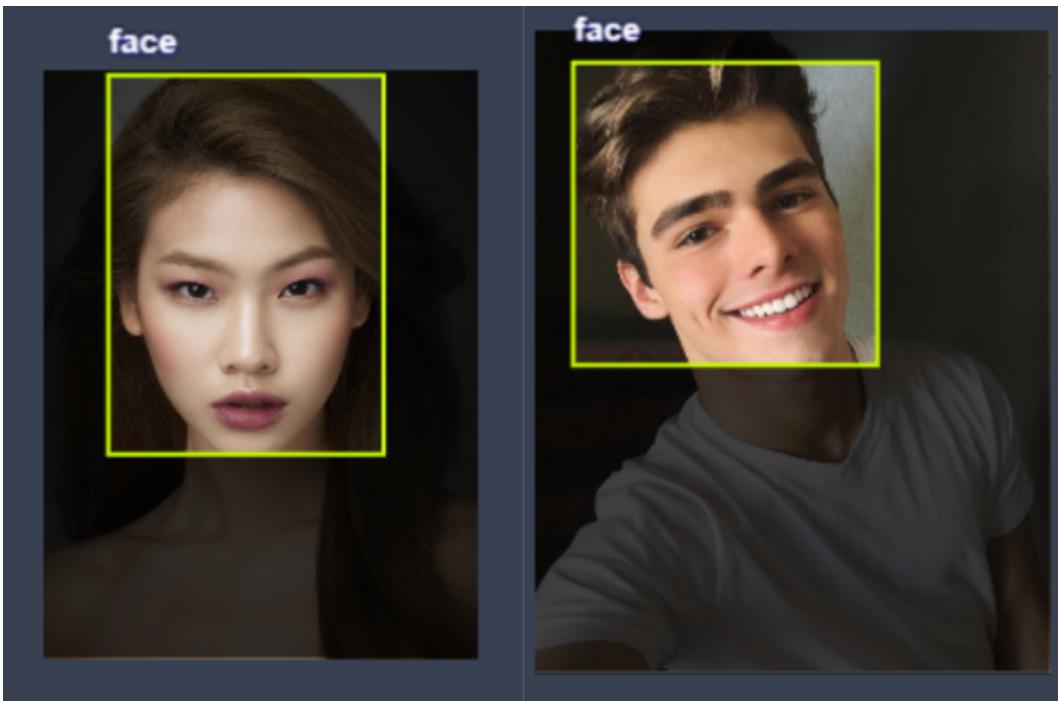
Chương này nhóm trình bày công cụ phục vụ cho việc thí nghiệm mô hình YOLOv9 và ResNet18 kết hợp SeftAttention.

4.1 Bộ dữ liệu

4.1.1 Đối với quá trình huấn luyện và kiểm tra mô hình

Trong nhiệm vụ nhận diện khuôn mặt bằng mô hình YOLOv9, nhóm đã sử dụng bộ dữ liệu hình ảnh từ Roboflow của Face Recognition for CV (phiên bản 1) của người dùng Sodiq Ismoilov, được thiết kế và gán nhãn phù hợp với bài toán nhận diện đối tượng. Tất cả các ảnh trong bộ dữ liệu được gán nhãn với một nhãn duy nhất là "face"(khuôn mặt) (Hình 4.1), đại diện cho nhiệm vụ nhận diện khuôn mặt. Để đảm bảo mô hình có khả năng hoạt động tốt trong các tình huống thực tế và tránh bị overfitting, chủ sở hữu của bộ dữ liệu đã tiến hành tăng cường dữ liệu (data augmentation) bằng cách áp dụng các phương pháp ngẫu nhiên trên Roboflow. Cụ thể:

- Lật hình ảnh theo chiều ngang (horizontal flip): Tạo ra các phiên bản đối xứng của ảnh để mô hình học được các hướng khác nhau của khuôn mặt.
- Xoay hình ảnh trong khoảng từ -15° đến $+15^\circ$: Thêm các biến thể về góc nghiêng của khuôn mặt, giúp mô hình nhận diện được các khuôn mặt không cố định vị trí.
- Điều chỉnh độ bão hòa màu (saturation): Thay đổi độ bão hòa màu của hình ảnh trong khoảng từ giảm 25% đến tăng 25% để mô hình học được các biến đổi về ánh sáng và màu sắc.



Hình 4.1: Ví dụ 2 mẫu dữ liệu của tập huấn luyện Face Recognition

Việc áp dụng các kỹ thuật tăng cường dữ liệu này giúp mô hình học được các đặc trưng khác nhau của dữ liệu trong thực tế, tăng cường khả năng khái quát hóa (generalization) của mô hình. Đồng thời, nó cũng làm giảm nguy cơ overfitting khi mô hình học được nhiều biến thể hơn, thay vì chỉ dựa vào các mẫu dữ liệu gốc.

Bộ dữ liệu cuối cùng sau khi tăng cường bao gồm tổng cộng 2400 ảnh. Tất cả các ảnh đều được chuẩn hóa về kích thước cố định là 640x640 pixel, đảm bảo tính nhất quán về đầu vào cho mô hình YOLOv9. Để đánh giá hiệu suất của mô hình một cách toàn diện, nhóm đã chia bộ dữ liệu này thành ba tập riêng biệt:

	Số lượng mẫu
Tập huấn luyện	2100
Tập xác minh	200
Tập kiểm tra	100

Bảng 4.1: Số lượng mẫu trong tập huấn luyện, xác minh và kiểm tra của tập huấn luyện Face Recognition

Trong nhiệm vụ phân loại giả mạo, nhóm sử dụng tập dữ liệu NUAA Photograph Imposter, trong đó nhóm em chọn dữ liệu đã xử lý phần khuôn mặt được cắt ra. Dữ liệu được lấy trên 15 người. Để phục vụ cho việc phân loại, bộ dữ liệu này được gán 2 nhãn là client (mặt thật) và imposter (mặt giả mạo)



(a) Mặt thật

(b) Mặt giả mạo

Hình 4.2: Ví dụ 2 mẫu thuộc 1 người trong NUAA Photograph Imposter

Dữ liệu thực hiện cho quá trình huấn luyện và kiểm tra được cho như sau:

	Mặt thật	Mặt giả mạo
Tập huấn luyện	1743	1748
Tập kiểm tra	3362	5761

Bảng 4.2: Số lượng mẫu trong tập huấn luyện và kiểm tra trong NUAA Photograph Imposter

4.1.2 Đổi với quá trình so sánh

Để đánh giá hiệu suất nhận diện khuôn mặt, nhóm em sử dụng 2 tập dữ liệu. Thứ nhất là tập kiểm tra trong tập dữ liệu Face Recognition gồm 100 ảnh. Thứ hai là tập Face Detection Dataset trên Kaggle. Dữ liệu bao gồm 16.7 nghìn hình ảnh với kích cỡ ảnh khác nhau và có thể có nhiều nhãn "face" trong cùng 1 ảnh. Nhưng trong quá trình so sánh, nhóm chỉ trích xuất ngẫu nhiên 800 mẫu dữ liệu từ tập xác minh để so sánh hiệu suất giữa YOLOv9 và Haar-Cascade.

Để đánh giá hiệu suất phân loại khuôn mặt giả mạo, nhóm em sử dụng tập dữ liệu LCC_FASD đã được xử lý và dùng trong đề tài nghiên cứu về Resnet18+Attention [5]. Trong nghiên cứu này, nhóm chọn tập dữ liệu cho kiểm tra gồm 389 mẫu dữ liệu là "mặt thật" và 3377 mẫu dữ liệu là "mặt giả mạo" để so sánh hiệu suất giữa Resnet18+Attention và Resnet18[6]

4.2 Môi trường cài đặt

Mô hình YOLOv9 được huấn luyện trên colab T4 GPU.

Mô hình Resnet18+Attention được huấn luyện trên Kaggle, CUDA T4 GPU.

4.3 Phương pháp đánh giá mô hình

4.3.1 Phương pháp đánh giá cho bài toán nhận diện khuôn mặt IOU

IOU là hàm đánh giá độ chính xác của object detector trên tập dữ liệu cụ thể [8]. IOU được tính bằng:

$$IOU = \frac{\text{Area_of_overlap}}{\text{Area_of_Union}}$$

Trong đó Area of Overlap là diện tích phần giao nhau giữa predicted bounding box (nhận dự đoán) với ground-truth bounding box (nhận thực), còn Area of Union là diện tích phần hợp giữa nhận dự đoán với nhận thực. Như vậy, IoU càng lớn thì mô hình dự đoán càng tốt, đồng nghĩa với việc phần giao lớn và phần hợp nhỏ (nhận dự đoán ra giống với nhận thực).

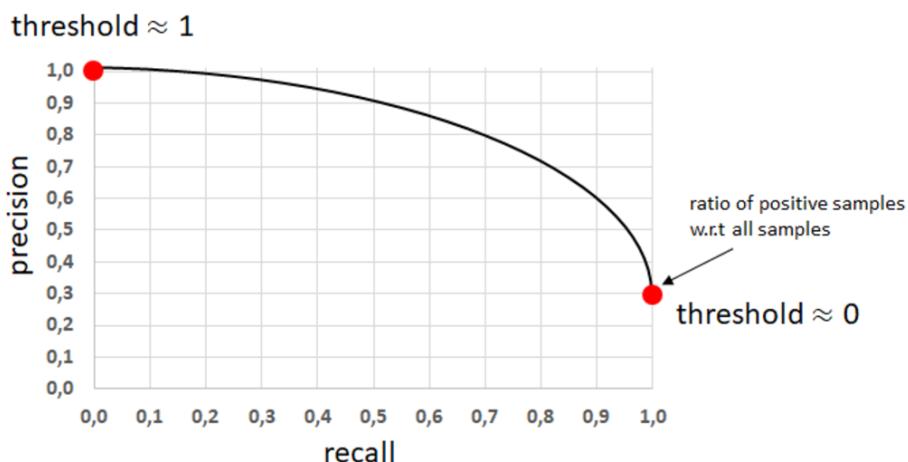
mAP (Mean Average Precision)

mAP (Mean Average Precision) là một chỉ số thường được sử dụng để đánh giá hiệu suất của các mô hình phát hiện đối tượng (object detection). Trong đó, AP (Average Precision) là một chỉ số đánh giá hiệu suất trên precision và recall.

Precision và Recall được xác định từ công thức:

$$\text{Precision} = \frac{TP}{TP+FP}, \text{Recall} = \frac{TP}{TP+FN}$$

Đường cong Precision-Recall (PR) 4.3 là biểu đồ đường cong cho thấy mối quan hệ giữa Precision và Recall khi thay đổi ngưỡng điểm dự đoán (confidence threshold). AP được tính là diện tích dưới đường cong Precision-Recall [8]. Diện tích này đại diện cho mức độ cân bằng giữa Precision và Recall trên tất cả các ngưỡng dự đoán.



Hình 4.3: Ví dụ về một biểu đồ đường cong PR [8]

Trong bài toán nhận diện khuôn mặt, kí hiệu mAP@ α được hiểu là giá trị mAP tại ngưỡng là α .

4.3.2 Phương pháp đánh giá cho phân loại giả mạo

FPR và FNR

Hai chỉ số thường được dùng trong xác minh sinh trắc học và cả phát hiện giả mạo là Tỉ lệ từ chối sai - False Rejection Rate (FRR) và Tỉ lệ chấp nhận sai - False Acceptance Rate (FAR). [3]. Hai chỉ số được xác định bằng công thức:

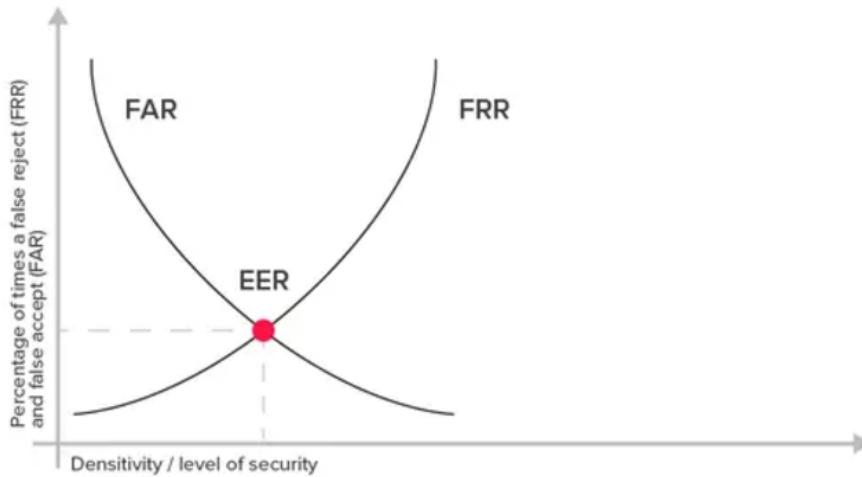
$$FAR = \frac{FP}{FP+TN}, FRR = \frac{FN}{FN+TP}$$

FPR đo lường tỷ lệ các mẫu giả bị nhận nhầm là thật, đây là tỷ lệ các trường hợp dương tính giả trên tổng số mẫu thực sự là giả. FPR thấp đồng nghĩa với việc mô hình ít nhầm lẫn các mẫu giả là thật, đảm bảo an toàn chống gian lận. Trong khi đó, FNR đo lường tỷ lệ các mẫu thật bị nhận nhầm là giả, đây là tỷ lệ các trường hợp âm tính giả trên tổng số mẫu thực sự là thật. FNR thấp nghĩa là hệ thống nhận diện được hầu hết khuôn mặt thật, đảm bảo trải nghiệm người dùng.

EER và AUC

EER (Equal Error Rate) và AUC (Area Under the Curve) đều là các chỉ số quan trọng để đánh giá hiệu quả của mô hình.

EER (Hình 4.4) là giá trị lỗi tại điểm mà FAR và FRR [3], từ đó đánh giá khả năng cân bằng của mô hình. Giá trị EER càng thấp, hiệu suất mô hình càng tốt.



Hình 4.4: Ví dụ biểu đồ EER [3]

AUC là diện tích dưới đường cong ROC, biểu thị mối quan hệ giữa TPR (True Positive Rate) và FPR (False Positive Rate) ở mọi ngưỡng quyết định. AUC cung cấp một cái nhìn toàn diện về khả năng phân biệt giữa khuôn mặt thật và giả của mô hình.

ACER

ACER (Average Classification Error Rate) là chỉ số được tính từ trung bình cộng giữa Attack Presentation Classification Error Rate (APCER) và Bona Fide Presentation Classification Error

Rate (BPCER) [2]:

$$ACER = \frac{APCER+BPCER}{2}$$

Trong đó, APCER là tỷ lệ mẫu giả bị nhận nhầm là thật, phản ánh mức độ dễ bị tấn công bởi các mẫu giả. Trong khi đó, BPCER là tỷ lệ mẫu thật bị nhận nhầm là giả, phản ánh mức độ gây phiền toái khi từ chối nhầm các mẫu thật. Công thức cho 2 chỉ số này lần lượt là:

$$APCER = \frac{FP}{FP+TN}, BPCER = \frac{FN}{FN+TP}$$

ACER cung cấp một cách tiếp cận cân bằng để đánh giá mô hình phát hiện giả mạo, đồng thời xem xét cả hai loại lỗi. Một giá trị ACER thấp cho thấy mô hình hoạt động hiệu quả trong việc phân biệt giữa khuôn mặt thật và giả.

Chương 5

Thực nghiệm và đánh giá

5.1 Đánh giá mô hình

5.1.1 YOLOv9

Quá trình huấn luyện YOLOv9 được chạy qua 50 epoch với thời gian huấn luyện là 1.435 tiếng.

Mô hình được đánh giá trên tập xác minh của dữ liệu gồm 200 mẫu dữ liệu.

Kết quả thu được:

1. Giá trị độ tin cậy (Precision): 0.994

Chỉ số Precision cao cho thấy mô hình hiếm khi dự đoán nhầm các đối tượng không phải khuôn mặt (giảm False Positives).

2. Giá trị độ nhạy (Recall): 1

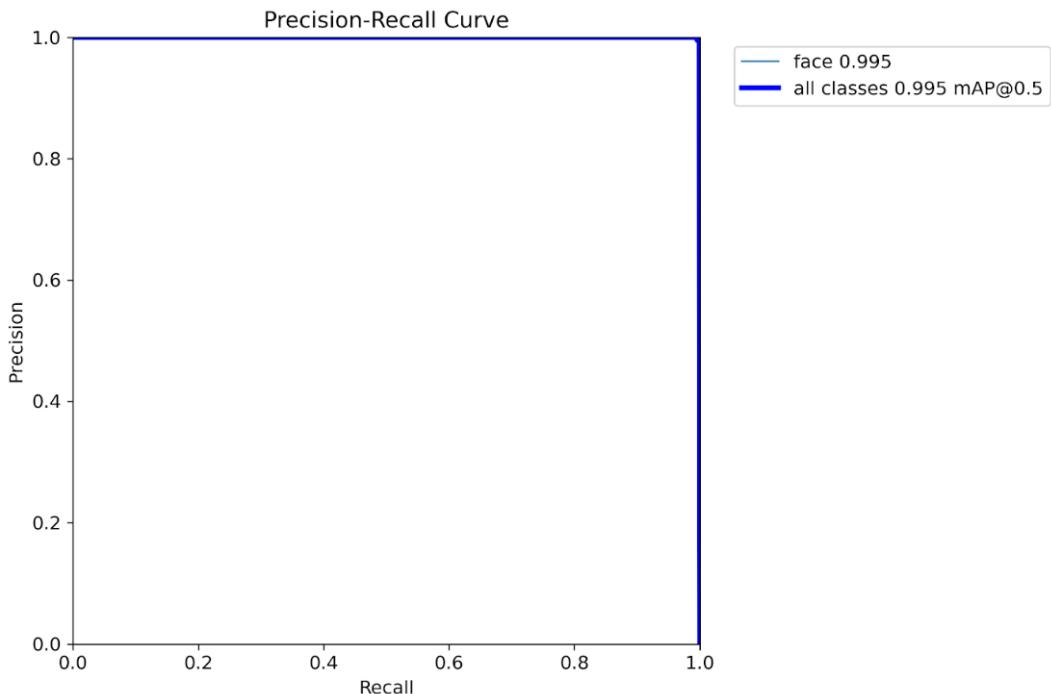
Mô hình có khả năng phát hiện tất cả khuôn mặt, không bỏ sót đối tượng (False Negatives bằng 0).

3. Giá trị mAP50, tức mAP tại ngưỡng IOU 50%: 0.995

Giá trị 0.995 cho thấy mô hình đạt độ chính xác cao ở mức yêu cầu cơ bản, rất ít sai sót trong việc xác định vị trí đối tượng.

4. Giá trị mAP50-95, tức trung bình giá trị mAP từ ngưỡng IOU 50% đến 95% với bước nhảy 5%: 0.864

Giá trị 0.864 thể hiện mô hình vẫn duy trì hiệu suất tốt, nhưng giảm dần khi ngưỡng IOU tăng cao.



Hình 5.1: Đường cong PR trên tập xác minh

Đường cong Precision - Recall (Hình 5.5) cho thấy đường cong có xu hướng gần góc trên bên phải và có nghĩa là tại các ngưỡng khác nhau thì Precision và Recall đều khá cao. Từ đó, suy ra mô hình YOLOv9 đã huấn luyện có thể cho ra kết quả dự đoán tốt.

5.1.2 Resnet18 + Attention

Nhóm thực hiện huấn luyện mô hình ResNet18+Attention với 100 epoch với thời gian huấn luyện là 6 tiếng.

Mô hình được huấn luyện bằng CUDA trên Kaggle, với tập kiểm tra gồm 9123 ảnh, gồm 5761 ảnh có nhãn giả mạo và 3362 hình ảnh có nhãn là thật.

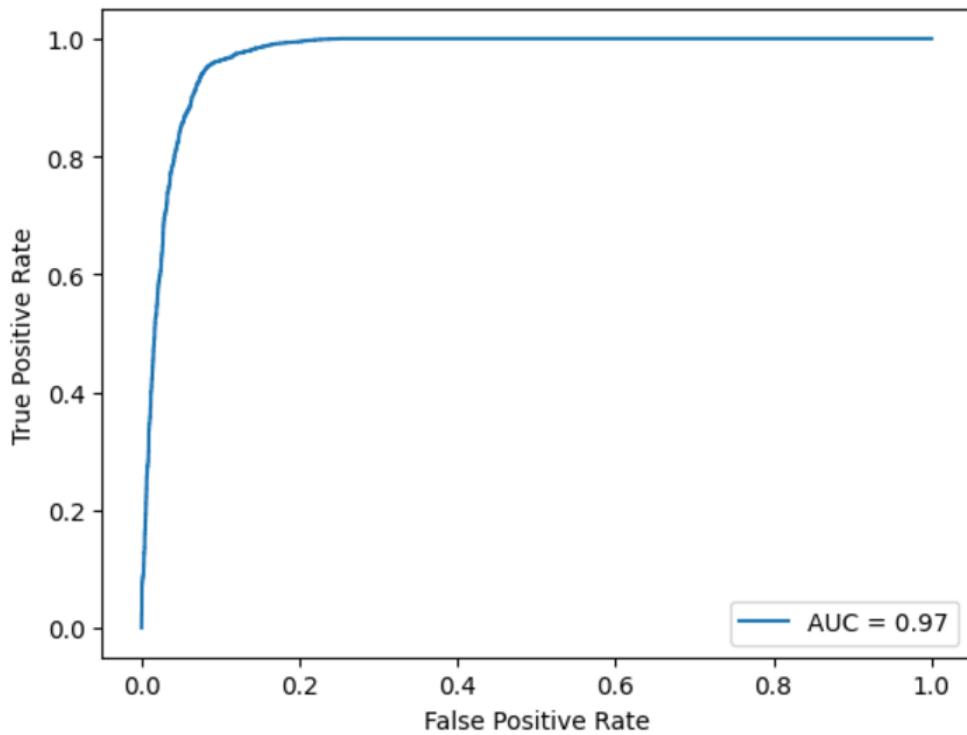
Độ đo	Kết quả
ACER	0.09596410291730659
FRR	0.006543723973825104
FAR	0.18538448186078807
ROC AUC score	0.9728075230676312
EER	0.07255684776948447
F1 score	0.8597168597168597
Accuracy	0.8805217581935767

Bảng 5.1: Kết quả đánh giá các độ đo của mô hình

- **ACER (9.6%):** Mô hình có ACER thấp, cho thấy sự cân bằng hợp lý giữa việc chấp nhận sai (FAR) và từ chối sai (FRR). Tuy nhiên, ACER vẫn có thể được cải thiện thêm để giảm thiểu sai sót trong nhận diện client và imposter.

- **FRR (0.65%)**: Giá trị FRR rất thấp, cho thấy mô hình rất nhạy và có khả năng nhận diện chính xác các mẫu client mà không bỏ sót. Đây là một yếu tố quan trọng trong các ứng dụng xác thực người dùng.
- **FAR (18.5%)**: FAR khá cao, điều này cho thấy mô hình nhận diện một số mẫu imposter là client. Để cải thiện, cần giảm thiểu việc nhận diện sai các mẫu imposter, điều này đặc biệt quan trọng trong các ứng dụng bảo mật.
- **ROC AUC score (97.3%)**: Giá trị AUC rất cao, chứng tỏ mô hình có khả năng phân biệt giữa client và imposter rất tốt, với độ chính xác gần như tuyệt đối trong việc phân loại.
- **EER (7.3%)**: EER thấp, cho thấy mô hình có sự cân bằng tốt giữa FAR và FRR. Điều này cho thấy rằng mô hình đạt được hiệu suất mạnh mẽ mà không có sự thiên lệch lớn giữa các loại lỗi.
- **F1 score (85.97%)**: F1 score cao, thể hiện sự cân bằng tốt giữa độ chính xác và độ nhạy. Mô hình đạt được sự kết hợp hợp lý của các chỉ số này và có thể cải thiện thêm.
- **Accuracy (88.05%)**: Accuracy cao, cho thấy mô hình phân loại đúng phần lớn các mẫu. Tuy nhiên, trong các bài toán phân loại không đều, accuracy có thể không phản ánh đầy đủ hiệu quả của mô hình.

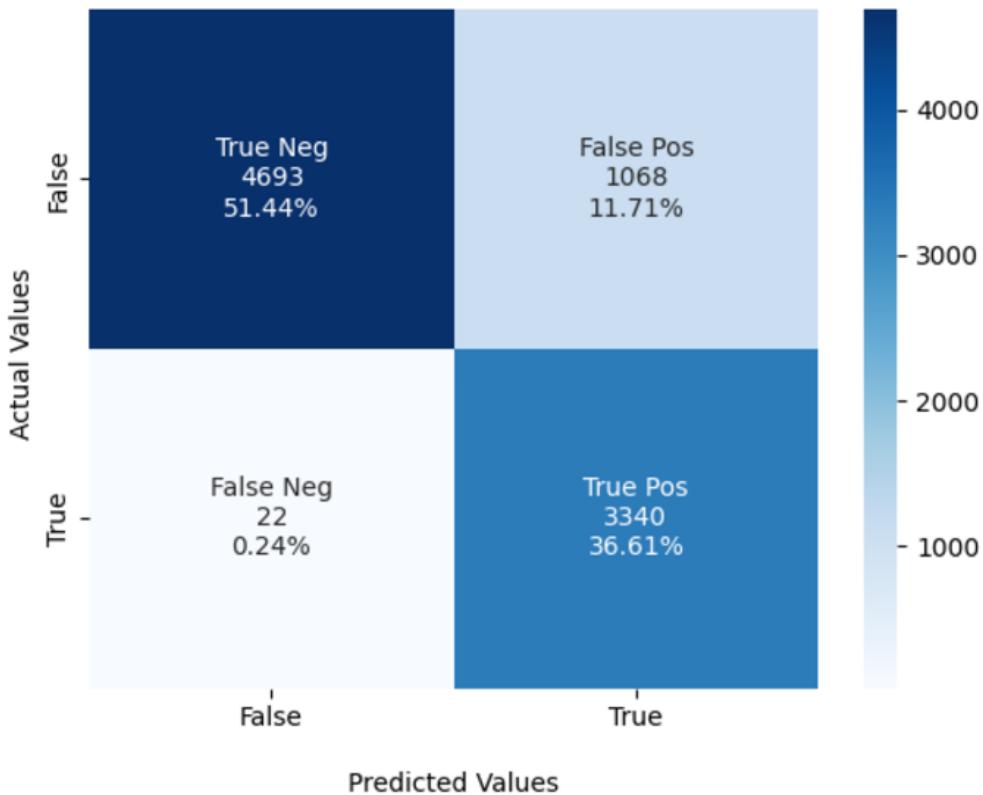
Biểu đồ ROC



Hình 5.2: Đường cong ROC trên tập xác minh

Biểu đồ này thể hiện một mô hình khá mạnh với khả năng phân loại chính xác, với giá trị AUC đạt 0.97. Tuy nhiên, một AUC cũng có thể yêu cầu kiểm tra thêm về tính chính xác trong các tình huống cụ thể của dữ liệu thực tế.

Confusion Matrix của mô hình



Hình 5.3: Confusion Matrix của mô hình

Accuracy cao, do tỷ lệ True Negatives và True Positives chiếm phần lớn trong tổng số mẫu.

False Positive Rate vẫn khá cao, nghĩa là mô hình đòi hỏi đánh giá sai các mẫu tiêu cực thành dương tính.

False Negative Rate rất thấp, điều này cho thấy mô hình có khả năng phát hiện tốt các trường hợp dương tính, giảm thiểu nguy cơ bỏ sót các mẫu quan trọng.

Nhìn chung, mô hình này hoạt động tốt với khả năng phân biệt các lớp, đặc biệt là các mẫu dương tính. Tuy nhiên, có thể cần tinh chỉnh thêm để giảm thiểu False Positives mà không làm tăng False Negatives.

5.2 So sánh mô hình

5.2.1 Mô hình YOLOv9 và phương pháp Haar Cascade

Nhóm em sẽ dựa trên chỉ số IOU để đánh giá hiệu suất. Trong đó, số box nhãn thực của tập Face Recognition là 100 còn số box nhãn thực của Face Detection là 2323.

YOLOv9 chứng tỏ ưu thế vượt trội trong nhiệm vụ nhận diện khuôn mặt, với độ chính xác IOU cao và khả năng khớp hoàn toàn với số lượng nhãn thực trên ảnh chứa duy nhất một khuôn mặt. Tuy nhiên, khi xử lý dữ liệu phức tạp hơn, đòi hỏi nhận diện đầy đủ tất cả khuôn mặt xuất hiện trong ảnh, hiệu suất của YOLOv9 vẫn còn hạn chế.

Face Recognition				Face Detection		
	Số box được dự đoán	IOU trung bình	Thời gian chạy (s)	Số box được dự đoán	IOU trung bình	Thời gian chạy (s)
YOLOv9	100	0.8917	107.87	705	0.4607	343.44
Haar Cascade	96	0.5566	46.90	1491	0.4750	541.94

Bảng 5.2: Kết quả trên hai tập dữ liệu kiểm tra

Dù vậy, điểm yếu này không ảnh hưởng đáng kể đến nhiệm vụ phát hiện giả mạo, bởi trọng tâm của bài toán chỉ nằm ở việc nhận diện chính xác khuôn mặt rõ nét trong ảnh đầu vào.

5.2.2 Mô hình Resnet18+Attention và mô hình Resnet18

Nhóm thực hiện so sánh hai mô hình trên tập dữ liệu LCC_FASD. Trong đó, mô hình Resnet18 được huấn luyện trên tập dữ liệu huấn luyện NUAA với 20 epoch trong thời gian chạy là 2,5 tiếng trên môi trường CUDA T4 GPU của Kaggle.

Trong nghiên cứu này, hai mô hình Resnet18 và Resnet18 kết hợp Attention đã được so sánh thông qua ba chỉ số quan trọng là FRR (False Rejection Rate), FAR (False Acceptance Rate), và ACER (Average Classification Error Rate). Kết quả thu được thể hiện những ưu điểm và hạn chế của từng mô hình:

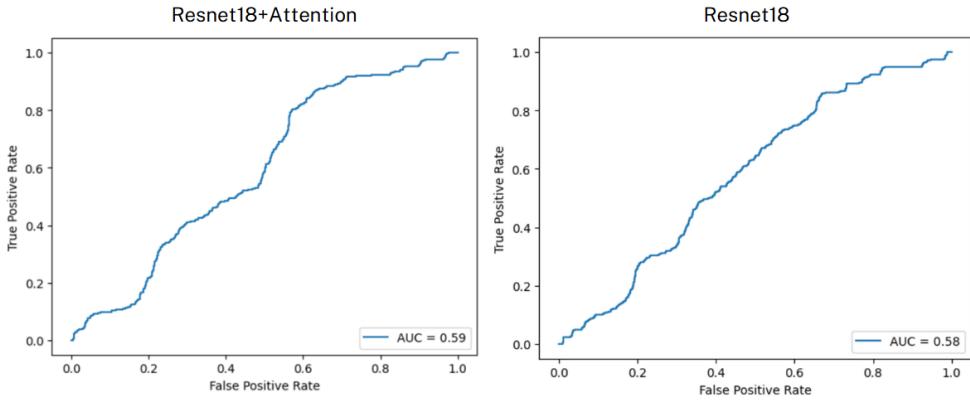
Mô hình	FRR	FAR	ACER
Resnet18+Attention	0.077	0.799	0.438
Resnet18	0.139	0.698	0.418

Bảng 5.3: Kết quả đánh giá trên LCC_FASD

Đầu tiên, xét về chỉ số FRR, mô hình Resnet18+Attention đạt kết quả tốt hơn đáng kể với giá trị 0.077, thấp hơn so với 0.139 của mô hình Resnet18. Điều này cho thấy việc tích hợp cơ chế Attention đã giúp mô hình giảm FRR, nâng cao khả năng nhận diện chính xác các mẫu thuộc lớp dương tính. Đây là một điểm mạnh nổi bật của Resnet18+Attention, đặc biệt trong các bài toán yêu cầu độ nhạy cao.

Tuy nhiên, khi xem xét chỉ số FAR, mô hình Resnet18+Attention lại có giá trị cao hơn, đạt 0.799, trong khi mô hình Resnet18 chỉ có 0.698. Điều này đồng nghĩa với việc Resnet18+Attention dễ dàng chấp nhận nhầm các mẫu thuộc lớp âm tính hơn. Đây là một hạn chế đáng kể của mô hình khi áp dụng vào các bài toán yêu cầu kiểm soát chặt chẽ độ chính xác của lớp âm tính.

Về chỉ số tổng quát ACER, mô hình Resnet18+Attention đạt giá trị 0.438, cao hơn một chút so với 0.418 của Resnet18. Điều này cho thấy hiệu suất hai mô hình là tương đương nhau.



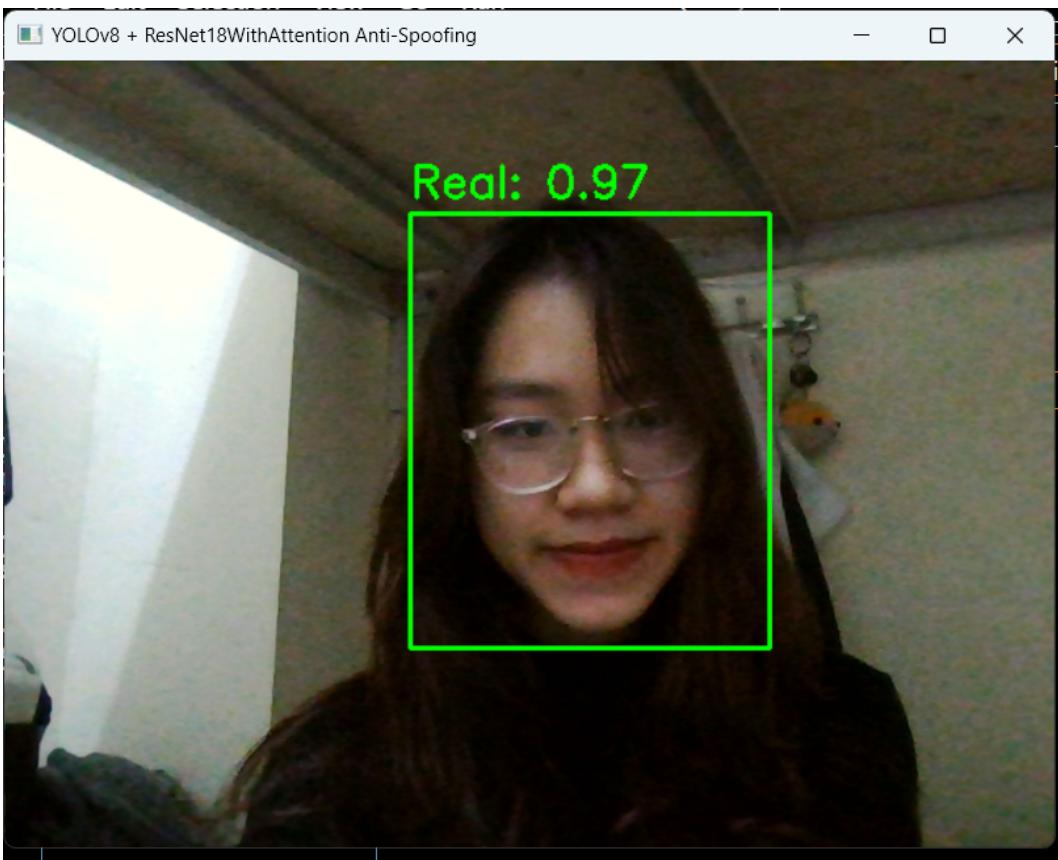
Hình 5.4: Biểu đồ ROC cho hai mô hình

Sự khác biệt về AUC (Hình 5.4) giữa hai mô hình là rất nhỏ, cho thấy việc thêm Attention chỉ cải thiện nhẹ hiệu suất tổng thể của mô hình Resnet18. Tuy nhiên, cả hai mô hình vẫn chưa đạt được hiệu suất phân loại mạnh mẽ, điều này có thể yêu cầu tối ưu hóa thêm về mặt kiến trúc hoặc dữ liệu. Trong các ứng dụng thực tế, sự cải thiện nhẹ này của Resnet18+Attention có thể hữu ích, nhưng cần cân nhắc kỹ lưỡng về chi phí tính toán bổ sung do việc sử dụng Attention.

5.3 Áp dụng vào Webcam

Hệ thống nhận diện khuôn mặt và phát hiện giả mạo được triển khai bằng cách sử dụng hai mô hình chính là YOLOv9 và ResNet18 với Attention. Mô hình YOLOv9 chịu trách nhiệm phát hiện và định vị khuôn mặt trong khung hình thu được từ webcam, trong khi mô hình ResNet18 với Attention được sử dụng để xác định tính xác thực của khuôn mặt. Đầu tiên, các khung hình từ webcam được xử lý qua YOLOv9 để phát hiện các khu vực chứa khuôn mặt với độ tin cậy trên ngưỡng 0.5. Những khu vực này sau đó được cắt và chuyển đổi sang định dạng phù hợp bằng các phép biến đổi như thay đổi kích thước, cắt trung tâm và chuẩn hóa. Tiếp theo, mô hình ResNet18 với Attention đưa ra dự đoán dưới dạng xác suất, giúp phân loại khuôn mặt là thật hoặc giả mạo.

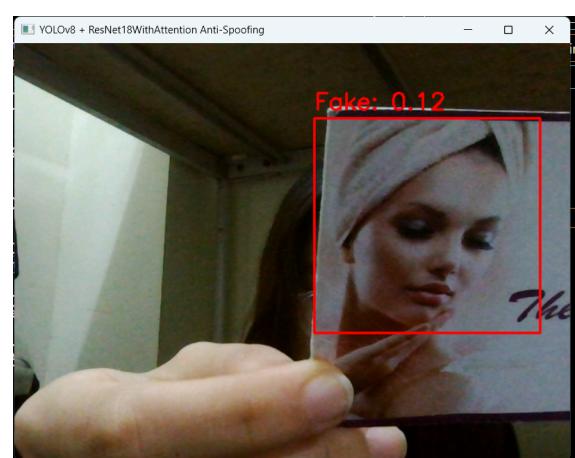
Kết quả được hiển thị trực tiếp trên khung hình webcam. Nếu khuôn mặt được xác định là thật, hệ thống hiển thị bounding box màu xanh lá cây kèm theo nhãn "Real" và giá trị xác suất tương ứng. Ngược lại, nếu khuôn mặt bị nhận diện là giả mạo, bounding box sẽ có màu đỏ cùng với nhãn "Fake" và xác suất. Toàn bộ quy trình được thực hiện trong thời gian thực, cho phép giám sát và nhận diện hiệu quả. Hệ thống sử dụng ngôn ngữ lập trình Python cùng các thư viện hỗ trợ như PyTorch, ultralytics, và OpenCV để đảm bảo tính chính xác và hiệu suất cao trong xử lý video trực tiếp.



Hình 5.5: Kết quả nhận diện đúng



Hình 5.6: Kết quả nhận diện giả mạo (1)



Hình 5.7: Kết quả nhận diện giả mạo (2)

Chương 6

Kết luận

6.1 Kết luận

Trong báo cáo này, nhóm đã đề xuất ý tưởng từ bài nghiên cứu Deepixbis [2], kết hợp hai mô hình YOLOv9 và Resnet18+Attention để giải quyết bài toán **phát hiện mặt giả mạo (face anti-spoofing)**. Mô hình YOLOv9 được sử dụng để nhận diện khuôn mặt, trong khi Resnet18+Attention đảm nhiệm vai trò phân loại các khuôn mặt thật và giả. Ý tưởng này được đánh giá là phù hợp và khả thi, nhờ vào sự kết hợp giữa khả năng nhận diện chính xác của YOLOv9 và sức mạnh học sâu của Resnet18+Attention trong việc nắm bắt các đặc trưng chi tiết. Đây là hướng tiếp cận mang tính đổi mới, giúp giải quyết bài toán trong bối cảnh các phương pháp gian lận khuôn mặt ngày càng tinh vi.

Tuy nhiên, kết quả thực nghiệm trên mô hình hiện tại cho thấy vẫn còn một số hạn chế. Hiệu suất mô hình khi thử nghiệm trực tiếp trên webcam chưa đạt được mức độ mong đợi, đặc biệt trong các tình huống thực tế phức tạp như ánh sáng thay đổi, nhiễu hình ảnh, hoặc khuôn mặt có độ phân giải thấp. Điều này cho thấy các mô hình cần được tối ưu hóa hơn nữa về mặt kiến trúc, dữ liệu và tham số huấn luyện để đạt hiệu quả cao hơn. Đồng thời, việc áp dụng mô hình trong thời gian thực cũng cần được cải thiện để đảm bảo tốc độ và tính khả thi trong các ứng dụng thực tế.

Mặc dù còn nhiều điểm cần cải tiến, nghiên cứu này đã đặt nền tảng quan trọng cho các hướng phát triển trong con đường nghiên cứu về AI của chúng em trong tương lai. Ý tưởng kết hợp giữa nhận diện và phân loại đã mở ra tiềm năng ứng dụng rộng rãi trong các lĩnh vực như bảo mật, giám sát, và xác thực danh tính.

6.2 Hướng phát triển

Qua quá trình đánh giá và so sánh kết quả giữa các mô hình, nhóm nhận thấy rằng mặc dù các mô hình đã đạt được một số kết quả khả quan, nhưng vẫn còn tồn tại những hạn chế nhất định. Những hạn chế này chủ yếu xuất phát từ việc chưa có sự huấn luyện sâu và tối ưu hóa toàn diện. Cụ thể, số lượng epoch trong quá trình huấn luyện chưa đủ lớn để mô hình học được các đặc trưng phức tạp của dữ liệu. Đồng thời, việc lựa chọn các tham số huấn luyện (hyperparameters) như learning rate, batch size, hay optimizer chưa được tối ưu hóa triệt để. Đây là những yếu tố ảnh hưởng trực tiếp đến hiệu suất của mô hình và cần được cải thiện trong tương lai.

Trong hướng phát triển nghiên cứu tiếp theo, nhóm đặt mục tiêu tập trung vào việc cải thiện và mở rộng cả về dữ liệu và phương pháp huấn luyện cho các mô hình nhằm nâng cao hiệu suất phát hiện và phân loại.

Đối với mô hình YOLOv9 nhận diện khuôn mặt, nhóm dự kiến sẽ thực hiện các bước cải tiến

nhằm nâng cao hiệu suất của mô hình. Đầu tiên, nhóm sẽ mở rộng tập dữ liệu bằng cách thu thập thêm ảnh chứa nhiều khuôn mặt trong các điều kiện thực tế đa dạng hơn, bao gồm các yếu tố như ánh sáng yếu, nhiễu ảnh, hoặc độ phức tạp của môi trường. Điều này giúp mô hình học được nhiều đặc trưng phong phú hơn và tăng khả năng tổng quát hóa. Tiếp theo, tập dữ liệu sẽ được gán nhãn một cách chi tiết và chính xác nhất để đảm bảo độ tin cậy cao trong quá trình huấn luyện. Ngoài ra, nhóm dự định sẽ tăng số lượng epoch huấn luyện lên trên 50 để mô hình có đủ thời gian học tập sâu hơn và nắm bắt các đặc trưng phức tạp của dữ liệu.

Đối với mô hình Resnet18+Attention, nhóm sẽ tập trung vào việc cải thiện chất lượng dữ liệu và tối ưu hóa quá trình huấn luyện. Cụ thể, tập dữ liệu sẽ được mở rộng để bao quát các điều kiện thực tế phức tạp, như các yếu tố ánh sáng không đồng đều hoặc nhiễu. Bên cạnh đó, nhóm sẽ thực hiện tối ưu hóa các tham số huấn luyện, bao gồm việc điều chỉnh learning rate, batch size, và áp dụng các kỹ thuật như regularization để giảm thiểu hiện tượng overfitting. Cuối cùng, nhóm cũng dự kiến tăng số lượng epoch huấn luyện nhằm cải thiện khả năng hội tụ và hiệu suất tổng thể của mô hình.

Những cải tiến này nhằm mục tiêu nâng cao hiệu quả phát hiện và phân loại của cả hai mô hình, đồng thời hướng đến khả năng ứng dụng thực tế trong các bài toán bảo mật và xác thực danh tính.

Tài liệu tham khảo

- [1] Nicolò Bonettini **and others**. *Video Face Manipulation Detection Through Ensemble of CNNs*. 2020. arXiv: 2004.07676 [cs.CV]. URL: <https://arxiv.org/abs/2004.07676>.
- [2] Anjith George **and** Sébastien Marcel. *Deep Pixel-wise Binary Supervision for Face Presentation Attack Detection*. 2019. arXiv: 1907.04047 [cs.CV]. URL: <https://arxiv.org/abs/1907.04047>.
- [3] Nguyen Thanh Huyen. “Tổng quan về Face Anti-Spoofing - Bài toán chống giả mạo khuôn mặt”. **in**.
- [4] Nguyen Mai. “Tính chất của Self-Attention và Transformer trong Computer Vision”. **in**.
- [5] Aleksandr Pikul. *A Method for Improving Presentation Attack Detection in Biometric Face Recognition Systems Using a Convolutional Neural Network with an Attention Mechanism*. https://drive.google.com/file/d/1s_JRff9DHX-ifOKN2n6RWqbtNOWxDgDC/view.
- [6] Farheen Ramzan **and others**. “A Deep Learning Approach for Automated Diagnosis and Multi-Class Classification of Alzheimer’s Disease Stages Using Resting-State fMRI and Residual Neural Networks”. **in** *Journal of Medical Systems*: 44 (december 2019). DOI: 10.1007/s10916-019-1475-2.
- [7] Joseph Redmon **and others**. *You Only Look Once: Unified, Real-Time Object Detection*. 2016. arXiv: 1506.02640 [cs.CV]. URL: <https://arxiv.org/abs/1506.02640>.
- [8] Nguyễn Chiến Thắng. “Thử tìm hiểu về mAP – đo lường Object Detection model”. **in**.
- [9] Muhammad Yaseen. *What is YOLOv9: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector*. 2024. arXiv: 2409.07813 [cs.CV]. URL: <https://arxiv.org/abs/2409.07813>.