ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

BÁO CÁO ĐỒ ÁN

PHÁP CHỨNG KỸ THUẬT SỐ – NT334.P11.ANTT

Đề tài: Exploring Different Deepfake Detection Model

Giảng viên hướng dẫn:

Lê Đức Thịnh

Sinh viên thực hiện:

21520435 - Nguyễn Thế Sơn

21520747 - Nguyễn Viết Dũng

21520840 - Lê Quang Hiển

21521195 - Trần Lê Minh Ngọc

Thành phố Hồ Chí Minh, tháng 6 năm 2024

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

BÁO CÁO ĐỒ ÁN

PHÁP CHỨNG KỸ THUẬT SỐ – NT334.P11.ANTT

Đề tài: Exploring Different Deepfake Detection Model

Giảng viên hướng dẫn:

Lê Đức Thịnh

Sinh viên thực hiện:

21520435 - Nguyễn Thế Sơn

21520747 - Nguyễn Viết Dũng

21520840 - Lê Quang Hiển

21521195 - Trần Lê Minh Ngọc

Thành phố Hồ Chí Minh, tháng 6 năm 2024

LỜI CẢM ƠN

Trước hết, nhóm chúng em xin gửi lời cảm ơn sâu sắc đến thầy Lê Đức Thịnh đã tạo điều kiện, giúp chúng em học tập và có được những kiến thức cơ bản làm tiền đề giúp chúng em hoàn thành được dự án này. Nhờ sự hướng dẫn tận tình và chu đáo của thầy, nhóm chúng em đã học hỏi được nhiều kinh nghiệm và hoàn thành thuận lợi, đúng tiến độ cho dự án của mình.

Trong quá trình thực hiện đồ án, nhóm chúng em luôn giữ một tinh thần cầu tiến, học hỏi và cải thiện từ những sai lầm, tham khảo từ nhiều nguồn tài liệu khác nhau và luôn mong tạo ra được sản phẩm chất lượng nhất có thể. Tuy nhiên, do vốn kiến thức còn hạn chế trong quá trình trau dồi từng ngày, nhóm chúng em không thể tránh được những sai sót, vì vậy chúng em mong rằng thầy sẽ đưa ra nhận xét một cách chân thành để chúng em học hỏi thêm kinh nghiệm nhằm mục đích phục vụ tốt các dự án khác trong tương lai. Xin chân thành cảm ơn thầy!

Nhóm thực hiện

NHẬN XÉT CỦA GIẢNG VIÊN

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

……, ngày…….....tháng……năm 2024

Người nhận xét

(Ký tên và ghi rõ họ tên)

MỤC LỤC

Contents

DANH MỤC HÌNH ẢNH

CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

1.1. Giới thiệu vấn đề

An toàn thông tin là một trong những vấn đề cốt lõi quan trọng đối với mọi tổ chức và cá nhân trong thế giới kỹ thuật số ngày nay. Trong xu thế chuyển đổi số diễn ra mạnh mẽ, dữ liệu trên hệ thống gia tăng mạnh, nguy cơ tấn công mạng, đặc biệt là tấn công lừa đảo ngày càng tăng cao. Deepfake là một trong những vấn đề nổi cộm trong lĩnh vực an ninh mạng và truyền thông hiện nay, gây ra những tác động đáng lo ngại đối với cá nhân, tổ chức và cả cộng đồng xã hội. Với sự phát triển nhanh chóng của trí tuệ nhân tạo và các công nghệ tiên tiến như Machine Learning, Deep Learning, Reinforcement Learning,…, Deepfake ngày càng trở nên tinh vi, cho phép tạo ra các nội dung giả mạo, như hình ảnh, video hoặc âm thanh khó mà có thể phân biệt với thực tế. Những nội dung giả mạo này đã và đang được sử dụng vào nhiều mục đích khác nhau, từ thao túng dư luận, tấn công danh tiếng, đến gian lận tài chính hoặc lan truyền thông tin sai lệch. Tuy nhiên, thực tế, các công cụ và dữ liệu phục vụ cho việc tạo ra Deepfake hiện nay ngày càng dễ tiếp cận, khiến vấn đề trở nên khó kiểm soát hơn bao giờ hết.

Hiện nay, có một số công nghệ Deepfake phổ biến đang được sử dụng để tạo ra các nội dung giả mạo. Một trong những công nghệ chủ yếu là Generative Adversarial Networks (GANs), một mô hình học sâu giúp tạo ra các video hoặc hình ảnh cực kỳ chân thực, khó phân biệt với thực tế. Bên cạnh đó, Autoencoders cũng được sử dụng để mã hóa và giải mã hình ảnh, từ đó tái tạo lại các video Deepfake. Hay một công nghệ khác là Face Swap, cho phép thay thế khuôn mặt của người này bằng khuôn mặt của người khác trong video hoặc hình ảnh. Những công nghệ này sẽ được giải thích kỹ ở **Chương 2**. Những công nghệ này ngày càng trở nên tinh vi và dễ tiếp cận, tạo ra nhiều thách thức trong việc phát hiện và ngăn chặn Deepfake.

“Theo dự báo của VSEC, các vụ tấn công mạng bằng AI, trong đó có lừa đảo Deepfake để giả mạo khuôn mặt, giọng nói sẽ gia tăng trong năm 2024. Bên cạnh

việc gọi điện giả danh nhân viên ngân hàng, công an như trước kia, các hacker đang ngày một tinh vi hơn khi sử dụng Deepfake để giả dạng người khác, hoặc ghép khuôn mặt kẻ lừa đảo vào hình ảnh bộ quân phục.

Tại Việt Nam, các vụ tấn công bằng AI để tạo ra nội dung giả mạo như video, giọng nói, hình ảnh, tin nhắn nhằm lừa đảo người dùng đang tăng lên. Nhiều người dân đã bị tấn công bằng hình thức này và sau đó trở thành nạn nhân, bị chiếm đoạt tiền, quyền sử dụng tài khoản ngân hàng và thông tin cá nhân bởi những kẻ lừa đảo." – trích Báo VietNamNet.

Một thách thức quan trọng nhưng thường bị đánh giá thấp trong lĩnh vực phát hiện và ứng phó sự cố an ninh mạng là sự thiếu hụt một khung chuẩn hóa, thống nhất và toàn diện để phân tích và đánh giá các phương pháp phát hiện Deepfake hiện tại. Điều này dẫn đến sự không nhất quán trong việc đánh giá, so sánh hiệu suất không công bằng, và có thể dẫn đến các kết quả không đáng tin cậy. Cụ thể, hiện vẫn chưa có sự đồng nhất trong việc phân loại mối đe dọa, xử lý dữ liệu và áp dụng các chiến lược đánh giá. Để khắc phục những hạn chế này, nhóm xin đề xuất một giải pháp toàn diện nhằm chuẩn hóa quy trình, tối ưu hóa hiệu suất và thúc đẩy tính minh bạch trong nghiên cứu, góp phần hỗ trợ các nỗ lực phát triển và đổi mới trong lĩnh vực ngày càng quan trọng này - DeepfakeBench.

1.2. Mục tiêu đồ án

Trong bối cảnh nêu trên, việc nghiên cứu và phát triển các giải pháp nhằm phát hiện và ngăn chặn Deepfake là một nhiệm vụ cấp bách. Trong bài báo cáo này, nhóm xin đề xuất DeepfakeBench – một nền tảng chuẩn hóa toàn diện, hỗ trợ đánh giá các phương pháp phát hiện Deepfake hiện đại, góp phần nâng cao hiệu quả bảo vệ thông tin và đảm bảo sự tin cậy trong môi trường số. Những đóng góp chính của công trình này gồm ba phần: **1) Mã nguồn mô-đun có thể mở rộng**: Mã nguồn của nhóm tác giả bao gồm ba mô-đun chính. Mô-đun xử lý dữ liệu cung cấp một hệ thống quản lý dữ liệu thống nhất để đảm bảo tính nhất quán cho tất cả các đầu vào phát hiện, giúp giảm bớt công việc xử lý và đánh giá dữ liệu mất nhiều thời gian. Mô-đun huấn luyện cung cấp một framework để triển khai các thuật toán phát hiện

tiên tiến, tạo điều kiện thuận lợi cho việc so sánh trực tiếp giữa các thuật toán phát hiện khác nhau. Mô-đun đánh giá và phân tích cung cấp nhiều chỉ số đánh giá phổ biến và công cụ phân tích phong phú để hỗ trợ các đánh giá và phân tích sâu hơn. **2) Đánh giá toàn diện**: Nhóm tác giả đã đánh giá 15 bộ phát hiện tiên tiến với 9 bộ dữ liệu Deepfake dưới nhiều cách thức đánh giá khác nhau, cung cấp một đánh giá toàn diện về hiệu suất của từng bộ phát hiện. **3) Phân tích sâu rộng và những cái nhìn mới:** Nhóm tác giả cung cấp các phân tích toàn diện từ nhiều góc độ, không chỉ phân tích tác động của các thuật toán hiện có mà còn khám phá những cái nhìn mới để khơi gợi những công nghệ sáng tạo.

Mục tiêu của đồ án này là nghiên cứu và đánh giá bộ tiêu chuẩn DeepfakeBench trong việc phát hiện và ngăn chặn Deepfake. Nhóm sẽ xem xét các yếu tố quan trọng của DeepfakeBench, bao gồm hệ thống quản lý dữ liệu thống nhất, các phương pháp phát hiện Deepfake tiên tiến, cũng như các chỉ số và giao thức đánh giá chuẩn hóa. Thông qua việc nghiên cứu bộ tiêu chuẩn này, nhóm hy vọng có thể hiểu rõ hơn về cách thức các công nghệ Deepfake hiện đại được phát hiện và đánh giá, đồng thời đề xuất các cải tiến hoặc ứng dụng phù hợp cho các hệ thống bảo mật trong việc đối phó với các mối đe dọa từ Deepfake. Đồ án cũng sẽ tập trung vào việc phân tích hiệu quả của các phương pháp và công cụ hiện tại trong việc phát hiện Deepfake, từ đó đưa ra những kết luận và khuyến nghị có giá trị cho việc nghiên cứu và phát triển trong tương lai.

Bên cạnh đó, nhóm đề xuất một mô hình sử dụng hình ảnh tự pha trộn (Self-Blended Images - SBIs) nhằm cải thiện hiệu suất phát hiện các nội dung giả mạo khuôn mặt (deepfake). Phương pháp SBIs tập trung vào việc tạo dữ liệu tổng hợp với các lỗi giả mạo phổ biến (như ranh giới pha trộn, sự không khớp màu sắc, và bất thường trong miền tần số), từ đó giúp mô hình học được các biểu diễn tổng quát và mạnh mẽ hơn trong việc nhận diện dấu vết làm giả. Nhóm kỳ vọng phương pháp SBIs không chỉ cải thiện độ chính xác phát hiện deepfake mà còn tối ưu chi phí tính toán và khả năng áp dụng trên các tập dữ liệu lớn.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Tổng quan về Deepfake

2.1.1. Deepfake là gì?

Deepfake, được tạo ra từ sự kết hợp giữa "deep learning" (học sâu) và "fake" (giả mạo), nổi bật với khả năng thao túng khuôn mặt, đã trở thành một công nghệ nổi bật có thể tạo ra các video giả mạo thông qua việc chồng ghép hình ảnh một cách liền mạch. Cụm từ này bắt đầu từ khoảng năm 2017, khi một người dùng trên Reddit tạo ra một subreddit có tên "deepfakes" và bắt đầu đăng tải các video sử dụng công nghệ hoán đổi khuôn mặt để chèn hình ảnh các ngôi sao nổi tiếng vào các video khiêu dâm sẵn có. Ngoài ra, các ví dụ deepfake được lan truyền rộng rãi còn bao gồm hình ảnh Giáo hoàng Francis mặc áo khoác phao, hình ảnh cựu Tổng thống Mỹ Donald Trump xô xát với cảnh sát, video CEO Facebook Mark Zuckerberg phát biểu về sức mạnh tối cao của công ty mình, … Tất cả những sự kiện này đều không xảy ra trong thực tế.

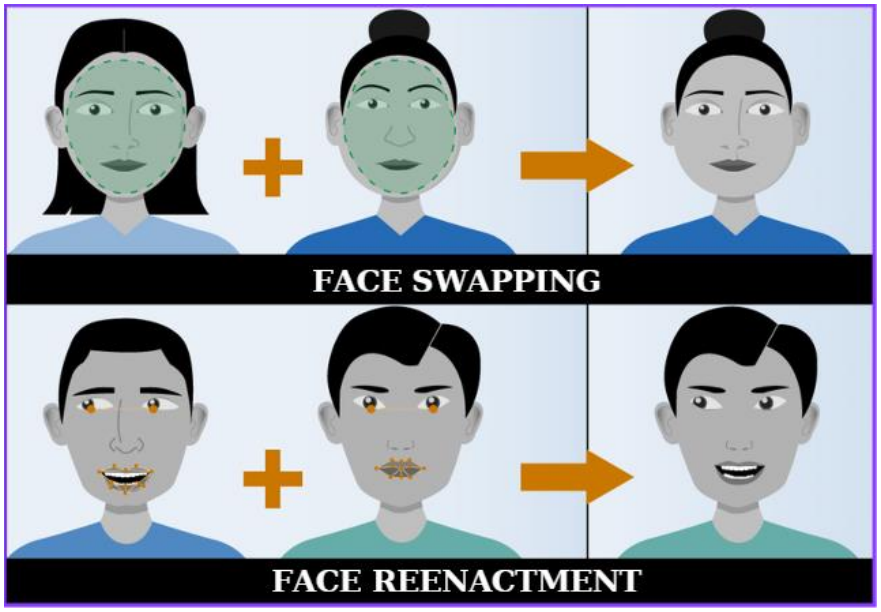
Hình 1. Hình ảnh tạo ra bởi Deepfake

Sự phổ biến của công nghệ Deepfake trong những năm gần đây có thể được giải thích nhờ vào những ứng dụng đa dạng của nó, từ giải trí và marketing đến các ứng dụng phức tạp hơn. Tuy nhiên, sự phát triển của Deepfake cũng không thiếu rủi ro. Những công cụ giúp sáng tạo và đổi mới này có thể bị lợi dụng cho mục đích xấu,

gây tổn hại đến quyền riêng tư, lan truyền thông tin sai lệch, hoặc làm suy giảm niềm tin vào truyền thông kỹ thuật số.

2.1.2. Deepfake Generation

Công nghệ tạo Deepfake, một lĩnh vực ứng dụng trí tuệ nhân tạo, tập trung vào việc chỉnh sửa khuôn mặt trong hình ảnh và video. Kể từ năm 2017, các kỹ thuật chỉnh sửa dựa trên học máy đã đạt được những bước tiến đáng kể, trong đó hai phương pháp chính nhận được nhiều sự chú ý là **hoán đổi khuôn mặt** (*Face Swapping*) và **tái hiện khuôn mặt** (*Face Reenactment*).



Hình 2. Phương pháp tạo Deepfake

Hoán đổi khuôn mặt (*Face Swapping*):

Đây là một nhánh quan trọng trong công nghệ tạo Deepfake. Phương pháp này sử dụng mô hình Autoencoder để chỉnh sửa, bao gồm hai Autoencoder với một Encoder chung và hai Decoder riêng biệt. Cụ thể:

- Encoder chung đóng vai trò chuyển đổi khuôn mặt của cả hai người (người A và người B) thành một biểu diễn nén chung. Điều này giúp hệ thống nhận diện và học được các đặc trưng cơ bản của khuôn mặt mà không bị giới hạn bởi danh tính của từng người.

- Hai Decoder riêng biệt sẽ được sử dụng để tái tạo khuôn mặt từ biểu diễn nén này. Decoder thứ nhất được huấn luyện để tái tạo lại khuôn mặt của người A, trong khi decoder thứ hai tái tạo khuôn mặt của người B.
- Quy trình hoán đổi, biểu diễn nén của người A sau khi qua Encoder sẽ được tái tạo bởi Decoder 2 (tức là decoder của người B). Điều này tạo ra khuôn mặt của người A nhưng mang hình dáng và đặc điểm của người B. Ngược lại, biểu diễn nén của người B khi qua Decoder 1 sẽ tái tạo thành khuôn mặt của người A.

Khi kết hợp, hệ thống có thể chuyển đổi khuôn mặt của người A thành khuôn mặt của người B và ngược lại, tạo ra kết quả giả mạo một cách tự nhiên và thuyết phục.

Tái hiện khuôn mặt (*Face Reenactment*):

Phương pháp này đặc trưng bởi các kỹ thuật chỉnh sửa dựa trên đồ họa, tập trung vào việc thay đổi biểu cảm hoặc chuyển động của một khuôn mặt trong hình ảnh hoặc video dựa trên dữ liệu từ một khuôn mặt khác. NeuralTextures và Face2Face thường được sử dụng trong tập dữ liệu FaceForensics++ và đều là những phương pháp tiêu chuẩn trong lĩnh vực tái hiện khuôn mặt. Trong đó Face2Face sử dụng các điểm đặc trưng trên khuôn mặt để tạo ra các biểu cảm đa dạng, còn NeuralTexture sử dụng hình ảnh được dựng từ mô hình khuôn mặt 3D để chuyển đổi biểu cảm.

Các kỹ thuật này không chỉ nâng cao chất lượng của Deepfake mà còn mở ra nhiều ứng dụng và thách thức mới trong lĩnh vực chỉnh sửa hình ảnh và video.

2.1.3. Deepfake Detection

Hiện tại, việc phát hiện Deepfake có thể được chia thành ba nhóm chính: Naive Detector (bộ phát hiện đơn giản), Spatial Detector (bộ phát hiện không gian), và Frequency Detector (bộ phát hiện tần số).

Naive detector (Bộ phát hiện đơn giản):

Naive detector là một phương pháp đơn giản nhưng hiệu quả, sử dụng mạng nơ-ron tích chập (Convolutional Neural Networks - CNNs) để phân biệt trực tiếp giữa nội dung Deepfake (giả mạo) và dữ liệu thật. Đây là một cách tiếp cận phổ biến

trong phát hiện Deepfake nhờ khả năng mạnh mẽ của CNN trong việc nhận diện các đặc trưng phức tạp của hình ảnh và video. Về nguyên lý hoạt động, mạng CNNs được thiết kế để trích xuất và học các đặc trưng trực quan như cấu trúc, kết cấu, và hình thái từ dữ liệu đầu vào. Khi áp dụng vào phát hiện Deepfake, CNN phân tích các dấu hiệu bất thường trong hình ảnh hoặc video, chẳng hạn như các lỗi làm giả, mờ viền, hoặc các sai lệch về ánh sáng và chi tiết.

Nhiều bộ phân loại nhị phân dựa trên CNN đã được đề xuất, chẳng hạn: MesoNet, Xception. Chi tiết các thuật toán sẽ được giới thiệu ở Phần 2.2.2.

Spatial detector (Bộ phát hiện không gian):

Spatial detector là một loại phương pháp phát hiện Deepfake tập trung vào việc nghiên cứu các biểu diễn không gian cụ thể trong hình ảnh hoặc video. Thay vì chỉ phân loại nội dung là thật hay giả, spatial detector đi sâu hơn để tìm hiểu và phân tích các đặc trưng cụ thể liên quan đến quá trình làm giả.

Spatial detector bao gồm một số kỹ thuật như:

- *Định vị vùng giả mạo (Forgery Localization):* thay vì chỉ xác định toàn bộ hình ảnh là giả mạo, spatial detector cố gắng xác định vị trí cụ thể trên khuôn mặt hoặc hình ảnh nơi Deepfake được áp dụng. Điều này giúp phát hiện chính xác các lỗi làm giả như vùng mắt, miệng, hoặc viền ghép nối không tự nhiên.
- *Mạng nơ-ron con nhộng (Capsule Network):* sử dụng các cấu trúc mạng đặc biệt để mô hình hóa mối quan hệ giữa các đặc trưng. Trong Deepfake, công nghệ này giúp nhận biết các đặc điểm không gian bị sai lệch do giả mạo, chẳng hạn như thay đổi ánh sáng hoặc hình dạng khuôn mặt.
- *Học tách biệt (Disentanglement Learning):* kỹ thuật này tách rời các đặc trưng quan trọng của hình ảnh thật (ví dụ: kết cấu, màu sắc) khỏi các đặc trưng giả mạo.
- *Tái dựng hình ảnh (Image Reconstruction):* là kỹ thuật sử dụng mô hình để tái tạo lại hình ảnh từ biểu diễn nén nhằm so sánh với hình ảnh gốc. Nếu hình

ảnh đầu vào là giả, quá trình tái dựng sẽ làm lộ ra các lỗi hoặc sự khác biệt bất thường giữa ảnh gốc và ảnh giả.

- *Công nghệ xóa (Erasing Technology):* một số spatial detector áp dụng kỹ thuật xóa để loại bỏ các yếu tố gây nhiễu, giúp hệ thống tập trung vào các vùng quan trọng, như vùng mặt hoặc các đặc điểm cụ thể của khuôn mặt.
- *Phát hiện lỗi ghép nối (Blending Artifacts):* lỗi ghép nối thường xuất hiện ở ranh giới giữa các vùng giả mạo và vùng thật, như viền khuôn mặt hoặc chuyển động không tự nhiên giữa các khung hình.

Nhờ định vị chính xác vùng giả mạo, mô hình có khả năng phát hiện ngay cả các chi tiết tinh vi. Bên cạnh đó, việc định vị vùng giả mạo cung cấp thông tin cụ thể hơn về cách Deepfake được tạo ra, hữu ích trong các ứng dụng pháp lý hoặc điều tra.

Frequency detector (Bộ phát hiện tần số):

Frequency detector là một phương pháp phát hiện Deepfake bằng cách phân tích các đặc trưng tần số của hình ảnh hoặc video. Thay vì chỉ xem xét hình ảnh ở dạng gốc (miền không gian), phương pháp này chuyển đổi hình ảnh sang miền tần số, nơi các chi tiết bất thường do quá trình tạo Deepfake có thể dễ dàng phát hiện. Trong đó, *tần số cao:* sẽ chứa các chi tiết tinh vi như nhiễu, đường nét sắc nét hoặc các chi tiết mỏng manh, rất dễ bị tác động bởi các thuật toán chỉnh sửa như làm mịn da, làm mờ vùng biên hoặc thay đổi chi tiết nhỏ. Những thay đổi này thường không rõ ràng bằng mắt thường nhưng có thể được phát hiện bằng các phương pháp phân tích tần số, đặc biệt là qua các nhiễu tần số cao không tự nhiên xuất hiện trong quá trình tạo Deepfake. *Tần số thấp*: chứa thông tin tổng thể hơn như màu sắc và hình dạng cơ bản. Những đặc điểm này ít bị ảnh hưởng, có thể không đủ chi tiết để phát hiện các chỉnh sửa tinh vi. Do đó, các phương pháp frequency detectors thường tập trung vào việc phân tích các biến đổi ở tần số cao.

SPSL và SRM là những ví dụ khác của frequency detector. Chúng lần lượt sử dụng phân tích phổ pha (phase spectrum analysis) và nhiễu tần số cao (high-frequency noises). Các phương pháp này đại diện cho sự đa dạng trong cách tiếp

cận phát hiện Deepfake, mỗi cách có ưu và nhược điểm riêng tùy thuộc vào loại dữ liệu và ngữ cảnh sử dụng.

2.1.4. Các nghiên cứu và tiêu chuẩn liên quan đến Deepfake

Sự phát triển nhanh chóng của công nghệ Deepfake đã thúc đẩy nhiều nghiên cứu chuyên sâu, dẫn đến sự ra đời của các bài khảo sát và các bộ dữ liệu tiêu chuẩn trong lĩnh vực này, cung cấp cái nhìn chi tiết về các khía cạnh khác nhau của công nghệ deepfake. Một số tiêu chuẩn trong lĩnh vực này đã trở thành công cụ thiết yếu để cung cấp các bộ dữ liệu giả mạo thực tế như là FaceForensics++ (FF++ - một tiêu chuẩn nổi bật, cung cấp video bị chỉnh sửa chất lượng cao và nhiều loại giả mạo khác nhau) và Deepfake Detection Challenge Dataset (DFDC - giới thiệu một dải dữ liệu phong phú trong nhiều bối cảnh khác nhau)

Mặc dù các phương pháp xây dựng tiêu chuẩn trong lĩnh vực phát hiện Deepfake đã mang lại nhiều đóng góp đáng kể nhưng chúng vẫn tồn tại một số hạn chế. Các tiêu chuẩn hiện tại thường tập trung vào bộ dữ liệu của riêng mình mà không cung cấp cách tiếp cận tiêu chuẩn hóa để xử lý dữ liệu từ nhiều nguồn khác nhau, dẫn đến thiếu nhất quán và khó khăn trong việc so sánh công bằng. Bên cạnh đó, việc thiếu một framework thống nhất cho các chiến lược huấn luyện, cài đặt và gia tăng dữ liệu khiến kết quả không đồng nhất. Thêm vào đó, các công cụ phân tích thường không được cung cấp đầy đủ, hạn chế khả năng đánh giá sâu về các yếu tố ảnh hưởng.

Gần đây, một số tiêu chuẩn mới đã xuất hiện, xây dựng một tiêu chuẩn để đánh giá các bộ phát hiện trên nhiều bộ dữ liệu khác nhau, hoặc tập trung vào phát hiện hình ảnh do GAN tạo ra bằng phương pháp học liên tục. Tuy nhiên, cả hai vẫn chưa cung cấp một mã nguồn toàn diện và mở rộng, bao gồm tiền xử lý dữ liệu, cài đặt chuẩn hóa, mô-đun huấn luyện, đánh giá, và các công cụ phân tích.

Trong bối cảnh này, DeepfakeBench nổi lên như một tiêu chuẩn ngắn gọn nhưng toàn diện, với ba đóng góp chính: hệ thống quản lý dữ liệu thống nhất đảm bảo tính nhất quán, framework tích hợp để triển khai các phương pháp tiên tiến, và bộ công cụ phân tích hỗ trợ đánh giá các yếu tố liên quan.

2.2. Bộ tiêu chuẩn DeepfakeBench

2.2.1. Dataset

Bộ tiêu chuẩn (Benchmark) hiện tại bao gồm một bộ sưu tập 9 bộ dữ liệu nổi bật và được sử dụng rộng rãi trong lĩnh vực phát hiện Deepfake, bao gồm:

1) **FaceForensics++ (FF++):** một bộ dữ liệu bao gồm các tập con như FF-DF, FF-F2F, FF-FS, FF-NT, và FF-all, được tạo từ video thật và giả mạo trên YouTube. Video thật được nhân đôi và chia thành ba tập: huấn luyện (train), kiểm thử (test), và xác thực (validation). Trong khi đó, video giả được phân chia dựa trên thông tin từ các tệp JSON đi kèm.
2) **DeepFakeDetection:** một bộ dữ liệu không sự phân chia chính thức cụ thể. Trong quá trình sử dụng, dữ liệu thật và giả được nhân đôi và chia thành các tập train, test, và validation.
3) **FaceShifter:** tập trung vào các video Deepfake được tạo bằng công cụ FaceShifter, một phương pháp giả mạo tinh vi. Bộ dữ liệu được xử lý tương tự FaceForensics++, với dữ liệu thật được nhân đôi và chia thành ba tập: train, test, và validation. Dữ liệu giả được phân chia dựa trên các tệp JSON từ FF++, đảm bảo tính nhất quán trong cách sắp xếp.
4) **Celeb-DF v1/v2 (phiên bản 1 và 2):** chứa các video thật và giả mạo của người nổi tiếng. Tất cả dữ liệu thật và giả được sử dụng làm tập huấn luyện, trong khi một tập con được chọn làm tập kiểm thử dựa trên tệp văn bản do tác giả cung cấp. Đáng chú ý, tập validation được thiết lập giống hệt với tập test, giúp tập trung vào việc đánh giá hiệu suất mô hình.
5) **DFDCP (DeepFake Detection Challenge Preview):** bao gồm các video thật và video giả được tạo bằng hai phương pháp khác nhau: phương pháp A và phương pháp B. Tập huấn luyện và kiểm thử được chia dựa trên phương pháp tạo video, trong khi tập validation được thiết lập giống hệt tập kiểm thử, đảm bảo tính đơn giản và hiệu quả trong đánh giá.
6) **DFDC (DeepFake Detection Challenge):** một bộ dữ liệu lớn được Facebook cung cấp, với mục tiêu hỗ trợ nghiên cứu phát hiện Deepfake. Cách

phân chia tập dữ liệu dựa trên phương pháp tạo video, tương tự như DFDCP. Tập validation trong bộ dữ liệu này cũng được thiết lập giống hệt với tập kiểm thử, mang lại sự thuận tiện trong sử dụng.

7) **DeeperForensics-1.0:** tập trung vào việc thử nghiệm các phương pháp gây nhiễu (perturbation) trên video giả mạo. Mỗi phương pháp gây nhiễu được coi là một loại video Deepfake riêng biệt. Dữ liệu giả được phân chia thành train, test, và validation dựa trên các tệp văn bản đi kèm, trong khi dữ liệu thật được nhân đôi và chia tương tự.
8) **UADFV (University of Albany DeepFake Video Dataset):** một trong những bộ dữ liệu đầu tiên về Deepfake, chứa video thật và giả mạo. Dữ liệu thật và giả được nhân đôi ba lần để tạo thành các tập train, test, và validation. Đây là một bộ dữ liệu đơn giản nhưng đóng vai trò quan trọng trong giai đoạn đầu của nghiên cứu phát hiện Deepfake.

Thông thường, FF++ được sử dụng để huấn luyện mô hình, trong khi các bộ dữ liệu còn lại thường được sử dụng làm dữ liệu kiểm tra.

2.2.2. Detector

Deepfake Benchmark đã triển khai tổng cộng 15 thuật toán phát hiện Deepfake đã được xác lập trước đó. Việc lựa chọn các thuật toán này dựa trên ba tiêu chí chính. Thứ nhất, các phương pháp có uy tín cổ điển (ví dụ, Xception) được ưu tiên, hoặc các phương pháp tiên tiến, thường được công bố trong các hội nghị hoặc tạp chí hàng đầu về thị giác máy tính hoặc học máy gần đây. Thứ hai, bộ Benchmark phân loại các bộ phát hiện thành ba loại: bộ phát hiện cơ bản, bộ phát hiện không gian, và bộ phát hiện tần số. Hơn nữa, bộ Benchmark này đã tránh đưa các bộ phát hiện truyền thống (ví dụ, Headpose) vào vì khả năng mở rộng hạn chế đối với các bộ dữ liệu quy mô lớn, khiến chúng ít phù hợp với mục tiêu của bộ tiêu chuẩn. Thứ ba, nhóm tác giả mong muốn bao gồm những phương pháp dễ dàng triển khai và tái hiện.

15 thuật toán phát hiện deepfake đã được triển khai trong DeepfakeBench:

1) **Meso4:** một phương pháp phát hiện Deepfake dựa trên mạng nơ-ron tích chập (CNN), tập trung vào các đặc tính trung bình (mesoscopic) của hình ảnh. Biến thể Meso4 sử dụng các lớp tích chập thông thường, giúp tối ưu hóa việc nhận diện các đặc điểm của Deepfake trong video.
2) **MesoIncep:** một biến thể của Meso4, được xây dựng dựa trên các module Inception tiên tiến. Tương tự Meso4, MesoIncep cũng sử dụng kiến trúc CNN đặc biệt và được triển khai trong kho lưu trữ MesoNet. Module Inception giúp mô hình phân tích và phát hiện các đặc điểm tinh vi hơn trong dữ liệu giả mạo.
3) **CNN-Aug:** một thuật toán phát hiện hình ảnh Deepfake được tạo bởi GANs, sử dụng mạng ResNet-34 cùng các kỹ thuật tăng cường dữ liệu phổ biến như nén JPEG và làm mờ Gaussian. Việc áp dụng các kỹ thuật này giúp mô hình tăng cường khả năng tổng quát hóa, đặc biệt trong môi trường dữ liệu đa dạng. Các thiết lập cụ thể về tăng cường được phân tích chi tiết trong báo cáo.
4) **EfficientNet-B4:** một mô hình dựa trên kiến trúc EfficientNet, thường được sử dụng làm nền tảng trong các phương pháp phát hiện Deepfake hiện đại. Khi triển khai trong DeepfakeBench, EfficientNet-B4 cho phép so sánh hiệu suất giữa các kiến trúc cơ bản khác nhau, đồng thời đánh giá mức cải tiến mà kiến trúc này mang lại trong việc phát hiện hình ảnh hoặc video giả mạo.
5) **Xception** là một trong những phương pháp phát hiện Deepfake hiệu quả, được xây dựng dựa trên kiến trúc **XceptionNet** – một mô hình học sâu được tối ưu hóa cho xử lý hình ảnh. Thuật toán này đặc biệt nổi bật nhờ khả năng khai thác các đặc điểm không gian (spatial features) và đặc điểm tạm thời (temporal features) trong dữ liệu video, giúp phát hiện những bất thường hoặc dấu hiệu giả mạo do Deepfake tạo ra. **Xception** được huấn luyện trên bộ dữ liệu FaceForensics++. Thuật toán này có ba biến thể: Xception-raw (dành cho video gốc), Xception-c23 (video nén H.264 với mức nén trung bình), và Xception-c40 (video nén H.264 với mức nén cao). Sự đa dạng này

giúp Xception thích ứng tốt với các định dạng và chất lượng dữ liệu khác nhau.

6) **Capsule:** một thuật toán sử dụng cấu trúc capsule dựa trên mạng VGG19 để phát hiện Deepfake. Mạng capsule này được huấn luyện ban đầu trên bộ dữ liệu FaceForensics++. Cấu trúc capsule giúp mô hình phát hiện các mối quan hệ không gian và hướng giữa các phần của hình ảnh, từ đó cải thiện khả năng phân loại Deepfake. Mô hình này sử dụng VGG19 như một kiến trúc cơ bản (backbone) để phân tích đặc điểm của hình ảnh.
7) **DSP-FWA:** một phương pháp phát hiện video Deepfake sử dụng mạng ResNet-50 để phát hiện các sai sót biến dạng khuôn mặt do thao tác thay đổi kích thước và nội suy trong các thuật toán tạo Deepfake cơ bản. Mô hình này được huấn luyện trên các hình ảnh khuôn mặt tự thu thập. Trong nghiên cứu ban đầu, DSP-FWA cải tiến thuật toán FWA bằng cách thêm module spatial pyramid pooling (SPP) để xử lý tốt hơn sự thay đổi độ phân giải của khuôn mặt. Tuy nhiên, trong DeepfakeBench, module SPP không được sử dụng để giữ nguyên kiến trúc chung cho tất cả các bộ phát hiện.
8) **Face X-ray:** sử dụng các dấu vết pha trộn trong các video Deepfake để cải thiện khả năng tổng quát trong việc phát hiện các video giả mạo chưa từng thấy. Thuật toán này sử dụng mạng HRNet được huấn luyện với các hình ảnh pha trộn và các video giả mạo từ bộ dữ liệu FaceForensics++.
9) **FFD:** áp dụng cơ chế chú ý (attention mechanism) để phát hiện và định vị các khu vực bị chỉnh sửa trong video. Mô hình này sử dụng hai loại lớp chú ý: mô hình biểu hiện chỉnh sửa và hồi quy trực tiếp, giúp mạng tập trung vào các khu vực quan trọng trong hình ảnh. Bên cạnh đó, ba loại hàm mất mát (loss functions) được sử dụng để giám sát quá trình học.
10) **CORE:** một thuật toán phát hiện Deepfake có tính chất đặc biệt là điều chỉnh sự nhất quán của các đại diện khác nhau trong quá trình học. Mỗi đại diện được thu thập từ các kỹ thuật tăng cường dữ liệu khác nhau, sau đó khoảng cách cosine giữa các đại diện này được điều chỉnh để tăng cường sự nhất

quán. CORE sử dụng Xception làm backbone để phát hiện Deepfake thông qua việc điều chỉnh các đại diện và cải thiện sự chính xác trong phân loại giả mạo.

11) **RECCE:** một thuật toán xây dựng đồ thị trên các đặc trưng của encoder và decoder theo cách đa quy mô. Thuật toán này sử dụng sự khác biệt trong quá trình tái tạo (reconstruction) như những dấu vết giả mạo trên đầu ra của đồ thị, giúp điều hướng đến đại diện cuối cùng được đưa vào bộ phân loại để phát hiện giả mạo. Quy trình tối ưu hóa từ đầu đến cuối cho cả việc tái tạo và học phân loại giúp cải thiện hiệu quả phát hiện Deepfake.
12) **UCF:** UCF giới thiệu một khung phân rã đa nhiệm nhằm giải quyết hai thách thức chính gây ra vấn đề tổng quát trong phát hiện Deepfake: quá khớp với các đặc trưng không liên quan và quá khớp với kết cấu đặc trưng của phương pháp cụ thể. Bằng cách phát hiện các đặc trưng chung, framework này giúp cải thiện khả năng tổng quát của mô hình. Thuật toán này sử dụng Xception làm kiến trúc cơ bản.
13) **F3Net:** sử dụng mạng hai nhánh với sự chú ý chéo (cross-attention) để học các manh mối nhận dạng các đặc trưng của Deepfake thông qua hai nhánh: FAD và LFS. Mô-đun FAD phân chia hình ảnh đầu vào theo miền tần số dựa trên các dải tần học được và thể hiện hình ảnh qua các thành phần nhận dạng tần số, từ đó học các mô hình giả mạo qua phân giải ảnh theo tần số. Mô-đun LFS trích xuất thống kê tần số cục bộ để mô tả sự khác biệt giữa khuôn mặt thật và giả, giúp khai thác các bất thường trong ảnh giả mạo ở mỗi dải tần.
14) **SPSL:** SPSL kết hợp hình ảnh không gian và phổ pha (phase spectrum) để phát hiện các sai sót do quá trình tăng kích thước (up-sampling) trong video Deepfake, từ đó cải thiện khả năng tổng quát. Nghiên cứu này lý thuyết phân tích tính hợp lý khi sử dụng phổ pha và nhận thấy rằng thông tin kết cấu cục bộ quan trọng hơn so với thông tin ngữ nghĩa cấp cao trong việc phát hiện giả mạo khuôn mặt.

15) **SRM:** SRM khai thác các đặc trưng tiếng ồn tần số cao và kết hợp hai đại diện khác nhau từ miền RGB và miền tần số để cải thiện khả năng tổng quát. Thuật toán này sử dụng Xception làm kiến trúc cơ bản để phát hiện Deepfake thông qua việc xử lý các đại diện khác nhau và khai thác những bất thường trong ảnh giả mạo.

2.2.3. Code base

Deepfakebench, như minh họa trong *Hình 3*, hệ thống này bao gồm ba mô-đun chính: Mô-đun xử lý dữ liệu, Mô-đun huấn luyện, và Mô-đun đánh giá và phân tích.

Data Processing Module (Mô-đun xử lý dữ liệu)

Mô-đun xử lý dữ liệu được thiết kế với hai phần chính để tự động hóa quy trình: Tiền xử lý dữ liệu và Sắp xếp dữ liệu.

Tiểu mô-đun Tiền xử lý dữ liệu cung cấp các giải pháp tối ưu, cho phép người dùng tùy chỉnh các bước xử lý thông qua tệp cấu hình YAML. Nhóm nghiên cứu đã phát triển một tập lệnh tiền xử lý chung, tích hợp các thao tác quan trọng như: trích xuất khung hình từ video, cắt và căn chỉnh khuôn mặt, cắt mặt nạ, tạo các điểm đặc trưng của khuôn mặt.

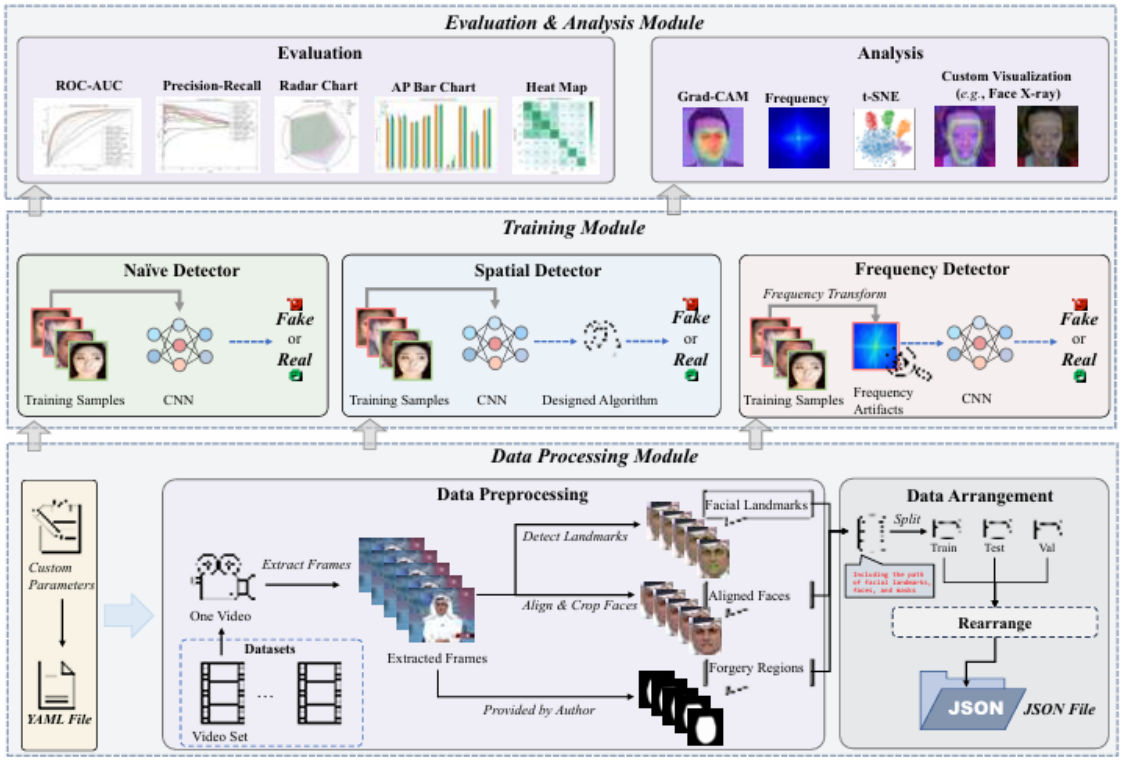
Tiểu mô-đun Sắp xếp dữ liệu giúp việc quản lý dữ liệu trở nên thuận tiện hơn nhờ sử dụng các tệp JSON riêng biệt cho từng bộ dữ liệu. Người dùng có thể chạy tập lệnh sắp xếp để tạo một tệp JSON thống nhất. Tệp này sẽ cung cấp quyền truy cập vào các tập dữ liệu huấn luyện, kiểm tra và xác thực, cùng với các thông tin chi tiết như khung hình, điểm đặc trưng, mặt nạ và các yếu tố khác liên quan đến dữ liệu.

Training Module (Mô-đun huấn luyện)

Mô-đun huấn luyện hiện hỗ trợ 15 bộ phát hiện được phân loại thành ba nhóm chính: bộ phát hiện cơ bản, bộ phát hiện không gian, và bộ phát hiện tần số. Bộ phát hiện cơ bản sử dụng các kiến trúc mạng CNN để trực tiếp phát hiện giả mạo mà không cần các đặc trưng thủ công bổ sung. Bộ phát hiện không gian mở rộng từ mạng CNN của bộ phát hiện cơ bản bằng cách kết hợp các thuật toán thủ công để phát hiện deepfake. Trong khi đó, bộ phát hiện tần số tập trung khai thác thông tin

từ miền tần số để nhận diện các đặc điểm giả mạo. Mỗi bộ phát hiện được tích hợp trong hệ thống đều được quản lý hiệu quả với tệp cấu hình YAML riêng, cho phép người dùng tùy chỉnh dễ dàng các tham số như kích thước lô hoặc tốc độ học. Quy trình huấn luyện và đánh giá được thực hiện trên một trình huấn luyện thống nhất, tự động ghi lại chỉ số và giá trị mất mát, đồng thời hỗ trợ ghi nhận và trực quan hóa dữ liệu mà không cần thao tác thủ công.

Đối với Training Module của DeepfakeBench, nhóm trình bày chi tiết một vài detector ở phần bên dưới.



Hình 3. Cấu trúc chung của các module trong DeepfakeBench

Evaluation and Analysis Module (Mô-đun đánh giá và phân tích)

Mô-đun đánh giá sử dụng bốn chỉ số đánh giá phổ biến: độ chính xác (ACC), diện tích dưới đường cong ROC (AUC), độ chính xác trung bình (AP) và tỷ lệ lỗi bằng nhau (EER). Tuy nhiên, một điểm đáng chú ý là có sự không nhất quán trong cách sử dụng các chỉ số này trong cộng đồng, khi một số phương pháp đánh giá ở

mức khung hình, trong khi số khác ở mức video, dẫn đến sự so sánh không công bằng. Trong đó:

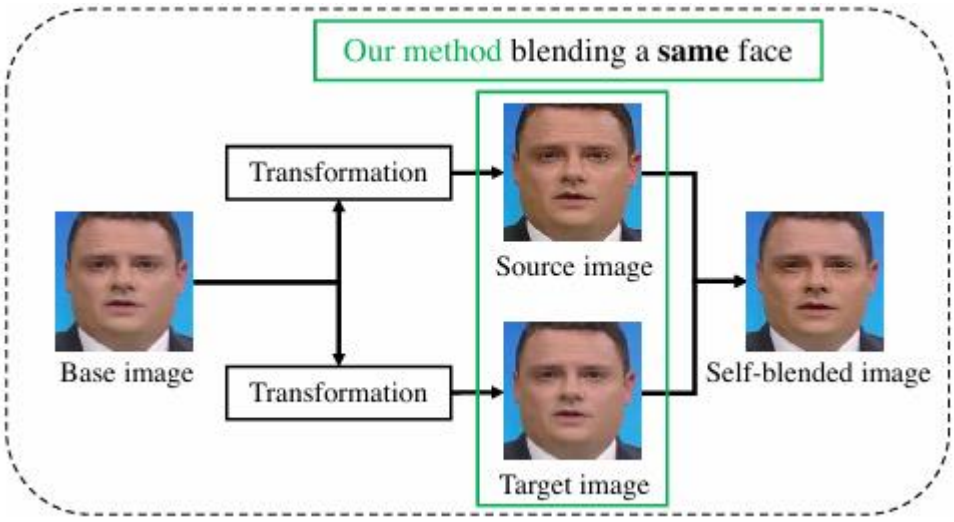
- Đánh giá ở mức khung hình nghĩa là hiệu suất của hệ thống được đo lường và tính toán dựa trên từng khung hình riêng lẻ trong một video.
- Đánh giá ở mức video nghĩa là hiệu suất của hệ thống được đo lường dựa trên toàn bộ video như một thực thể duy nhất, thay vì xem xét từng khung hình riêng lẻ.

Hệ thống của DeepfakeBench hiện áp dụng đánh giá ở mức khung hình để xây dựng cơ sở so sánh công bằng giữa các bộ phát hiện.

2.3. Self-Blended Images (SBIs) – Hình ảnh tự pha trộn

2.3.1. Self-Blended Images là gì?

Self-Blended Images (SBIs) là một loại dữ liệu huấn luyện tổng hợp mới, được thiết kế để hỗ trợ phát hiện các Deepfake một cách hiệu quả hơn. SBIs được tạo ra bằng cách pha trộn các hình ảnh giả nguồn (pseudo source) và mục tiêu (pseudo target) từ một hình ảnh gốc duy nhất. Quy trình này tái tạo các lỗi giả mạo phổ biến như ranh giới pha trộn và sự không đồng nhất về thống kê giữa các khu vực nguồn và mục tiêu trong hình ảnh. Mô tả sơ lược ở *Hình 4*.

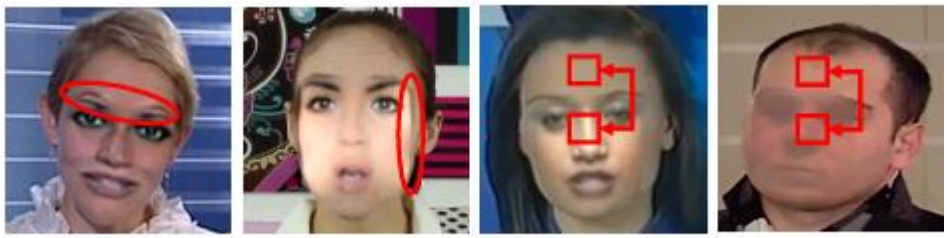


Hình 4. Quá trình tạo Self-Blended Images cơ bản

SBIs được phát triển dựa trên nhận định rằng khi các kỹ thuật deepfake ngày càng tiên tiến, hình ảnh giả do GAN tạo ra sẽ càng giống hình ảnh gốc về mặt đặc điểm khuôn mặt và thống kê pixel. Điểm độc đáo của SBIs nằm ở việc tạo ra các mẫu giả mạo tổng quát và khó nhận diện. Điều này giúp mô hình học được các biểu diễn mạnh mẽ và tổng quát, thay vì chỉ tập trung vào các đặc điểm giả mạo đặc thù của từng phương pháp thao tác. Kết quả là các mô hình huấn luyện với SBIs có khả năng phát hiện tốt hơn các thao tác chưa từng gặp trước đây.

Ý tưởng chính là tạo ra các mẫu giả mạo khó nhận diện, chứa các dấu vết giả mạo khuôn mặt phổ biến, từ đó khuyến khích các mô hình học được các biểu diễn tổng quát và mạnh mẽ hơn cho việc phát hiện giả mạo khuôn mặt. Để thực hiện điều này, nhóm sẽ phân tích các khuôn mặt bị chỉnh sửa và xác định bốn loại lỗi giả mạo điển hình (*Hình 5*):

- *Sự không nhất quán của đặc trưng nguồn (Source Feature Inconsistencies):* các điểm đặc trưng của khuôn mặt trong hình ảnh nguồn và mục tiêu không đồng nhất, dẫn đến sự không phù hợp về các đặc điểm như khuôn mặt, mắt, miệng, tạo ra các lỗi nhận diện khi so sánh các phần khác nhau của khuôn mặt.
- *Ranh giới pha trộn (Blending Boundaries):* những khu vực mà các hình ảnh nguồn và mục tiêu được kết hợp với nhau, tạo ra các ranh giới không tự nhiên hoặc mờ nhạt giữa các phần của khuôn mặt, gây ra sự bất đồng bộ rõ rệt.
- *Lỗi màu sắc (Color Inconsistencies):* sự không khớp màu sắc giữa các phần của hình ảnh gốc và hình ảnh giả, có thể làm cho màu sắc của khuôn mặt hoặc các khu vực khác trong ảnh không đồng đều, gây ra sự nhận diện giả mạo.
- *Bất thường thống kê trong miền tần số (Statistical Anomalies in Frequency Domain):* các bất thường này xảy ra trong các đặc điểm tần số của hình ảnh, nơi các mô hình phân tích tần số không thể xử lý đúng các thay đổi bất thường giữa các phần của hình ảnh giả và gốc.



Hình 5. Bốn lỗi giả mạo điển hình

2.3.2. Bộ tạo nguồn-mục tiêu (Source-Target Generator - STG)

Dựa trên một hình ảnh đầu vào **I**, Bộ tạo nguồn-mục tiêu (STG) khởi tạo hai hình ảnh giả: hình ảnh nguồn giả (pseudo source) và hình ảnh mục tiêu giả (pseudo target), bằng cách sao chép chính hình ảnh **I**. Sau khi khởi tạo, STG tiếp tục tạo ra sự không nhất quán thống kê giữa hình ảnh nguồn và mục tiêu bằng cách ngẫu nhiên áp dụng một số phép biến đổi hình ảnh lên một trong hai hình ảnh này. Cụ thể, STG thực hiện các biến đổi màu sắc và biến đổi tần số.

Đối với biến đổi màu sắc, STG sẽ ngẫu nhiên thay đổi giá trị của các kênh màu RGB, cũng như các thông số khác như độ sắc thái (hue), độ bão hòa (saturation), độ sáng (value), độ sáng tổng thể (brightness) và độ tương phản (contrast) của hình ảnh đầu vào. Những thay đổi này tạo ra sự biến động về màu sắc giữa hình ảnh nguồn và mục tiêu, tạo ra các bất thường có thể được phát hiện trong quá trình huấn luyện mô hình phát hiện giả mạo.

Sau đó, STG áp dụng các biến đổi tần số, chẳng hạn như giảm độ phân giải (downsample) hoặc làm sắc nét (sharpen) hình ảnh đầu vào. Các biến đổi tần số này tác động đến các chi tiết cấu trúc và kết cấu của hình ảnh, giúp tạo ra các bất thường trong miền tần số, qua đó làm tăng độ khó trong việc phát hiện các hình ảnh giả mạo. Thông qua các biến đổi này, STG tạo ra các hình ảnh giả mạo có sự không nhất quán về màu sắc và cấu trúc, giúp huấn luyện các mô hình phát hiện deepfake trở nên tổng quát và mạnh mẽ hơn.

2.3.3. Bộ tạo mặt nạ (Mask Generator - MG)

MG tạo ra một hình ảnh mask màu xám để pha trộn các hình ảnh nguồn và mục tiêu. Quá trình này được thực hiện qua các bước sau:

Dự đoán vùng khuôn mặt và tạo mặt nạ

Đầu tiên, MG sử dụng một bộ phát hiện đặc điểm khuôn mặt (landmark detector) để xác định vùng khuôn mặt trong hình ảnh đầu vào. Sau đó, từ các đặc điểm khuôn mặt đã được phát hiện, MG tính toán một convex hull (bao lồi) để tạo ra một mặt nạ. Convex hull là một hình dạng bao quanh tất cả các điểm đặc trưng khuôn mặt, giúp xác định chính xác khu vực mà mặt nạ sẽ bao phủ.

Biến dạng mặt nạ

Sau khi tạo mặt nạ, MG tiếp tục biến dạng mặt nạ này bằng cách sử dụng phép biến đổi đặc điểm khuôn mặt. Mục tiêu của việc này là làm cho mặt nạ trở nên phức tạp hơn, giống như các thay đổi tự nhiên của khuôn mặt trong hình ảnh giả mạo.

Tăng sự đa dạng của mask

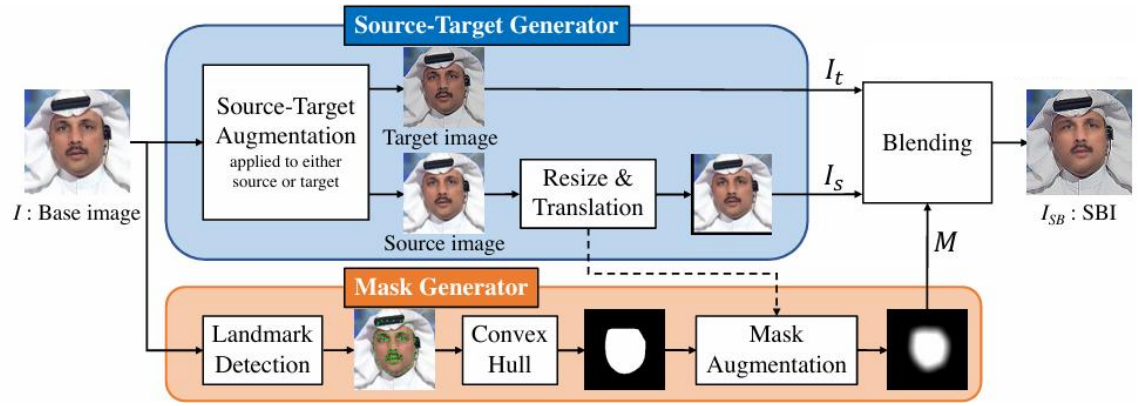
Để tạo ra nhiều loại mặt nạ khác nhau, MG thay đổi hình dạng của mặt nạ và tỷ lệ pha trộn một cách ngẫu nhiên. Cụ thể: *Biến dạng đàn hồi*: mặt nạ được biến dạng theo một cách đàn hồi, giúp làm cho các khu vực của mặt nạ trở nên mềm mại và linh hoạt hơn. Điều này giúp mặt nạ trở nên khó nhận diện và phù hợp với nhiều kiểu khuôn mặt; *Làm mượt mặt nạ*: MG áp dụng hai bộ lọc Gaussian với các tham số khác nhau để làm mượt mặt nạ. Quá trình này giúp làm cho các biên của mặt nạ trở nên mềm mại và tự nhiên hơn; *Xói mòn và giãn nở*: Nếu kích thước hạt nhân của bộ lọc Gaussian đầu tiên lớn hơn bộ lọc thứ hai, mặt nạ sẽ bị xói mòn (thu hẹp lại). Ngược lại, nếu kích thước hạt nhân của bộ lọc đầu tiên nhỏ hơn, mặt nạ sẽ bị giãn nở (mở rộng ra).

Thay đổi tỷ lệ pha trộn

Cuối cùng, MG thay đổi tỷ lệ pha trộn của hình ảnh nguồn vào. Điều này được thực hiện bằng cách nhân mặt nạ với một hằng số $r \in (0,1]$. Các giá trị của r được lấy ngẫu nhiên từ các giá trị $\{0.25,0.5,0.75,1\}$. Tỷ lệ pha trộn này quyết định mức

độ ảnh hưởng của mặt nạ lên hình ảnh nguồn, từ đó tạo ra những sự pha trộn khác nhau giữa nguồn và mục tiêu.

Tổng quan lại, một hình ảnh cơ sở **I** được đưa vào **Bộ tạo nguồn-mục tiêu (STG)** và **Bộ tạo mặt nạ (MG).** STG tạo ra các hình ảnh giả nguồn và mục tiêu từ hình ảnh cơ sở bằng cách sử dụng một số phép biến đổi hình ảnh, trong khi MG tạo ra một mặt nạ pha trộn từ các đặc điểm khuôn mặt và biến dạng nó để tăng sự đa dạng của mặt nạ. Cuối cùng, các hình ảnh nguồn và mục tiêu được pha trộn với mặt nạ.



Hình 6. Quá trình tạo ra Self-Blended Images chi tiết

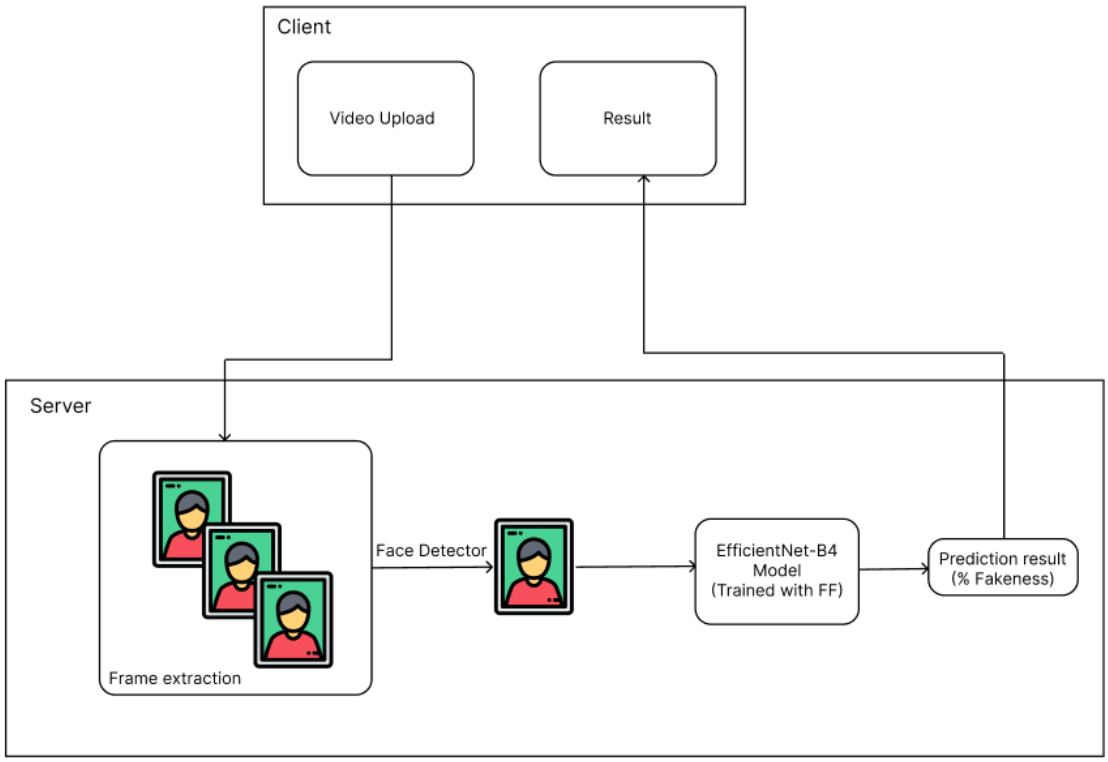
CHƯƠNG 3: HIỆN THỰC HỆ THỐNG

3.1. Giới thiệu thành phần hệ thống:

Hệ thống được thiết kế nhằm phát hiện video deepfake, dựa trên phương pháp Self-Blended Images (SBIs) được giới thiệu trong bài báo "Detecting Deepfakes with Self-Blended Images". Tận dụng mô hình EfficientNet-B4 đã được huấn luyện trên các đặc điểm của hình ảnh giả mạo và phương pháp Self-Blended Images (SBIs), cùng với quy trình xử lý video và phân tích khung hình để dự đoán mức độ giả mạo (% Fakeness) của video đầu vào.

Hệ thống bao gồm hai thành phần chính:

- Client: Cung cấp giao diện cho người dùng để tải lên video và nhận kết quả dự đoán.
- Server: Xử lý dữ liệu, trích xuất đặc trưng từ video, áp dụng mô hình EfficientNet-B4 được huấn luyện trên phương pháp SBIs và trả về kết quả cho người dùng.



3.2. Chi tiết hiện thực hệ thống:

- Phía client:
    - Video Upload:
        - Người dùng tải video lên thông qua giao diện web. Video sẽ được gửi trực tiếp đến server xử lý.
        - Video được kiểm tra định dạng trước khi chấp nhận để đảm bảo tính tương thích.
    - Kết quả: Sau khi quá trình xử lý hoàn tất, mức độ giả mạo của video dưới dạng phần trăm (% Fakeness) sẽ được hiển thị, video cũng đồng thời được phát.
- Phía server:
    - Trích xuất khung hình: Video được chuyển đổi thành chuỗi các khung hình để xử lý. Sử dụng thư viện OpenCV để trích xuất khung hình với tần số phù hợp.
    - Phát hiện khuôn mặt Face Detection: Sử dụng mô hình phát hiện khuôn mặt RetinaFace để xác định và cắt khuôn mặt trong các khung hình. Cụ thể là sử dụng mô hình Resnet-50 cho việc phát hiện khuôn mặt trên các frame ảnh. Ảnh khuôn mặt sẽ được chuẩn hoá với kích thước tối đa 2048 pixel
    - Dự đoán với EfficientNet-B4: Mô hình EfficientNet-B4, đã được huấn luyện trước trên dữ liệu FF (FaccForensics) + SBIs, được sử dụng để phân loại mức độ giả mạo của từng khuôn mặt trong khung hình.
    - Kết quả dự đoán và đánh giá: Kết quả dự đoán của từng khung hình được tổng hợp để đưa ra mức độ giả mạo cuối cùng cho toàn bộ video và là kết quả trả về cho client. Đánh giá mô hình trên một tập dữ liệu Celeb-DFv1 và cho ra kết quả là chỉ số AUC (Area Under the Curve).

3.3. Công nghệ và công cụ:

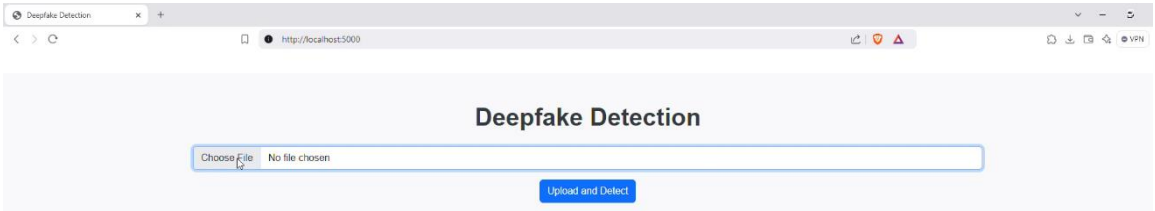
- Thư viện và framework:
    - OpenCV: Trích xuất khung hình từ video.
    - MTCNN/RetinaFace: Phát hiện và thực hiện quy trình extraction.
    - Flask: Xây dựng web
    - Torch: framework deep learning cung cấp các mô hình công cụ và thuật toán hỗ trợ xử lý hình ảnh, Tích hợp và triển khai EfficientNet-B4.
    - Numpy, Pandas,…: các thư viện chuẩn của Python cho Machine Learning và Deep Learning

CHƯƠNG 4: THỰC NGHIỆM VÀ ĐÁNH GIÁ

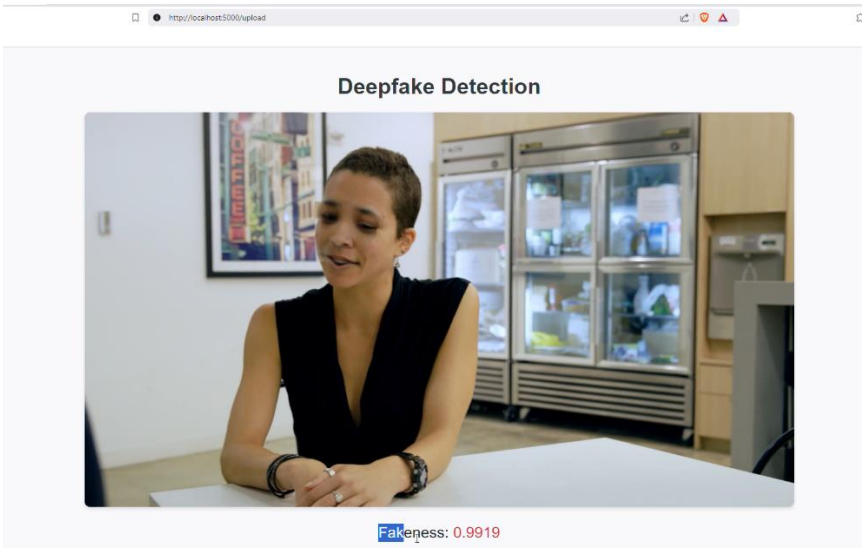
4.1. Deepfake Detection with SBIs

Hệ thống phát hiện Deepfake với SBIs được triển khai trên nền tảng website sử dụng Flask. Hệ thống cho phép người dùng tải lên video, sau đó áp dụng thuật toán phát hiện Deepfake dựa trên SBIs. Kết quả được hiển thị trực quan thông qua giao diện web xác định xem video được tải lên có phải là video dùng deepfake hay không.

- **FrontEnd**: Giao diện web đơn giản với các chức năng chính:
    - Tải lên tệp video.
    - Xem kết quả phân tích và phát hiện Deepfake.
- Giao diện được xây dựng bằng HTML, CSS để đảm bảo tính thân thiện và trực quan.



Hình 7. Giao diện khi vào trang web



Hình 8. Giao diện khi trả về kết quả nhận diện

- **BackEnd**:
  - Sử dụng Flask để xử lý yêu cầu từ giao diện web và tương tác với mô hình phát hiện Deepfake.
  - Tích hợp các thư viện cần thiết như OpenCV, Torch, và NumPy để xử lý dữ liệu đầu vào.
  - Áp dụng mô-đun SBIs để phân tích video và phát hiện các dấu hiệu Deepfake.

app.py 3 result.html index.html

app.py > upload_video

```python
import os
import subprocess
from flask import Flask, request, render_template, redirect, url_for, send_from_directory, jsonify

app = Flask(__name__)

UPLOAD_FOLDER = os.path.join(app.root_path, 'uploads')
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
os.makedirs(app.config['UPLOAD_FOLDER'], exist_ok=True)


@app.route('/')
def index():
    return render_template('index.html')

@app.route('/upload', methods=['POST'])
def upload_video():
    if 'video' not in request.files:
        return jsonify({"error": "No video file uploaded"}), 400
    file = request.files['video']
    if file.filename == '':
        return jsonify({"error": "No selected file"}), 400

    if file and file.filename.endswith('.mp4'):
        filepath = os.path.join(app.config['UPLOAD_FOLDER'], file.filename)
        file.save(filepath)

        command = f"python src/inference/inference_video.py -w weights/FFraw.tar -i {filepath}"
        output = subprocess.getoutput(command)
        print(output)
        result = output.split(': ')[1]
        print(result)
        if result:
            confidence = output.split("Fake: ")[-1].strip()
        else:
            return jsonify({"error": "Failed to process video"}), 500
        print(url_for('uploaded_file', filename=file.filename))
        #vidPath = filepath
        return render_template('result.html', video_url=url_for('uploaded_file', filename=file.filename), result=result, confidence=confidence)
    else:
        return jsonify({"error": "Invalid file type"}), 400

@app.route('/uploads/<filename>')
def uploaded_file(filename):
    return send_from_directory(app.config['UPLOAD_FOLDER'], filename)

if __name__ == '__main__':
    app.run(debug=True)
```

Hình 9. Code BackEnd sử dụng Flask

- **Mô-đun phát hiện**: Tích hợp các mô hình đã huấn luyện trên SBIs, được tham khảo trên github: https://github.com/mapooon/SelfBlendedImages

4.2. DeepFakeBench

DeepFakeBench là 1 framework tổng hợp nhiều mô hình học máy/học sâu khác nhau với mục đích giúp người sử dụng hoặc các nghiên cứu sinh có thể so sánh và đánh giá hiệu suất của từng mô hình trên các tập dữ liệu khác nhau.

Thực hiện download source code và set up theo hướng dẫn tại trang github của tác giả: https://github.com/SCLBD/DeepfakeBench?tab=readme-ov-file

Tải thêm các file lưu trữ weight mà nhóm tác giả đã huấn luyện từ trước trong phần release. Có tất cả 13 file weight:

Sau khi tải xong, nhóm sẽ tiến hành chạy thử và đánh giá các model trên bộ dữ liệu Celeb-DF-v1. Nhóm sử dụng điểm AUC (Area Under the Curve) để thực hiện đánh giá hiệu suất model

Kết quả sau khi thử nghiệm các model*:

Tên model	AUC trên từng frame ảnh	AUC trên toàn bộ video
Xception	0.7538	0.8098
UCF	0.8111	0.8607
F3Net	0.75	0.8111
CapsuleNet	0.7729	0.8281
EfficientNet B4	0.7666	0.8149
Meso4	0.4946	0.4936
Meso4 Inception	0.6935	0.7415

* Do một số giới hạn về phần cứng nên nhóm chỉ có thể triển khai 1 số model

Với phương pháp xử lý là tách 1 video ra thành nhiều frame khác nhau, việc thực hiện tính kết quả đánh giá của framework cũng chia ra thành 2 loại điểm khác nhau để việc đánh giá trở nên trực quan hơn:

- Đưa ra điểm dựa trên từng frame ảnh được tách ra: các frame ảnh được tách ra từ 1 video sẽ được đánh nhãn của video đó, ví dụ: nếu video được đánh nhãn là video thật thì các frame được tách ra video cũng được đánh nhãn là thật và ngược lại. Các frame này sau đó sẽ được đưa vào cho mô hình phán đoán và tính toán kết quả (AUC) như bình thường.
- Đưa ra điểm dựa trên việc đánh giá toàn bộ video: do phương pháp xử lý dữ liệu đặc thù của framework, việc đưa ra kết quả đánh giá của toàn bộ 1 video sẽ được thực hiện bằng các lấy toàn bộ các điểm đánh giá của các frame thuộc video đó chia cho tổng số frame trích xuất ra được từ video đó.

Chính vì cách tính điểm như vậy nên điểm số đánh giá của 2 phần này có sự cách biệt tương đối lớn nhất là trong trường hợp 1 video deepfake có chứa lẫn lộn giữ các frame ảnh thật và frame ảnh giả.

Trong ngữ cảnh thực tế, việc đánh giá 1 video có phải là deepfake hay không dựa vào từng frame của video đó như cách mà Deepfake Bench đã làm có thể cho ra kết quả sai lệch khá lớn khi thời lượng của video đó quá dài so với thời lượng xuất hiện các dấu hiệu deepfake. Tuy nhiên, đối với các mẫu video ngắn có thời lượng trung bình từ 1-3 phút, Deepfake Bench lại có hiệu suất khá tốt và có thể sử dụng làm tiêu chuẩn để so sánh và đánh giá các mô hình học máy/học sâu phát hiện deepfake khác.

Tóm lại, Deepfake Bench là một framework tập hợp rất nhiều mô hình học máy/học sâu khác nhau, có tính tùy biến cao, có thể triển khai và tích hợp với nhiều hệ thống giúp cho việc đánh giá, so sánh hiệu suất của các mô hình trong việc phát hiện deepfake.

4.3. Một số công cụ opensource để tạo DeepFake

4.3.1. deepfakes_faceswap

Deepfakes_Faceswap là một công cụ deepfake opensource với 3.1k lượt stars và 1k lượt forks trên github với những thư viện, requirements đơn giản. Chỉ cần python3, openCV3, tensorflow và keras 2 là đã có thể cài đặt được công cụ. Công cụ có sẵn file model đã được train sẵn nên chỉ cần cài đặt những thư viện cần thiết là sử dụng được.

Link của công cụ: https://github.com/joshua-wu/deepfakes_faceswap

Nhược điểm: Công cụ lần cuối update là vào 7 năm trước nên những thư viện đã lỗi thời cũng như thuật toán deepfake không còn được tối ưu, dễ bị phát hiện bởi những công cụ deepfake detector

4.3.2. roop

Công cụ roop là một công cụ mã nguồn mở được thiết kế để thực hiện deepfake dựa trên kĩ thuật face swapping trong thời gian thực với input đầu vào là các video và hình ảnh. Roop nổi bật với sự đơn giản trong cách sử dụng, cho phép người dùng hoán đổi khuôn mặt giữa các video hoặc hình ảnh chỉ với một vài bước cơ bản mà không yêu cầu kiến thức chuyên sâu về xử lý hình ảnh hoặc học máy. Công cụ này được phát triển bởi **s0md3v**, một nhà phát triển nổi tiếng trong cộng đồng mã nguồn mở với 28.7k lượt star và 7k lượt fork.

Link công cụ: https://github.com/s0md3v/roop

Ưu điểm: Giao diện đơn giản, không đòi hỏi người dùng phải có kiến thức chuyên môn về học máy hoặc xử lý hình ảnh. Hỗ trợ các lệnh CLI dễ hiểu và cài đặt nhanh chóng thông qua mã nguồn trên GitHub.

Nhược điểm: So với các công cụ hoán đổi khuôn mặt cao cấp như DeepFaceLab, chất lượng hình ảnh đầu ra của Roop thường không mượt mà hoặc tự nhiên bằng, đặc biệt khi làm việc với video dài. Phụ thuộc vào phần cứng.

4.3.3. faceswap

Faceswap là một trong những công cụ mã nguồn mở hàng đầu được sử dụng để deepfake sử dụng kĩ thuật faceswapping với input đầu vào là ảnh hoặc video. Đây là một dự án cộng đồng được phát triển bởi các thành viên nhiệt huyết nhằm cung cấp một nền tảng mạnh mẽ để xử lý hoán đổi khuôn mặt dựa trên các mô hình học sâu (Deep Learning). FaceSwap có tính linh hoạt cao và hỗ trợ nhiều tính năng tùy chỉnh, làm cho nó trở thành một công cụ phổ biến trong lĩnh vực xử lý hình ảnh và nghiên cứu về Deepfake.

Link công cụ: https://github.com/deepfakes/faceswap

Ưu điểm: Hiện tại, công cụ vẫn được duy trì và tiếp tục phát triển. Có giao diện người dùng (GUI) để người dùng dễ dàng thao tác và sử dụng. Có hướng dẫn kĩ càng và chi tiết. Chất lượng output cũng rất tốt.

Nhược điểm: Tương tự như những công cụ trên thì faceswap cũng yêu cầu tài nguyên cao. Ở công cụ này thì cần có bước train nên thời gian xử lý cho ra output lâu hơn bình thường.

4.4. Tool phát hiện deepfake

4.4.1. Deepfake Detector

Deepfake Detector là một ứng dụng phát hiện deepfake thương mại, được phát triển để chống lại các cuộc lừa đảo sử dụng deepfake nhắm đến cá nhân hoặc doanh nghiệp. Trên trang chủ của ứng dụng này khẳng định rằng có thể phân biệt giữa các video thật và giả với độ chính xác lên đến 92%. Để sử dụng được dịch vụ, người dùng cần phải đăng ký tài khoản thì mới có được kết quả.

Trang chủ của ứng dụng: https://deepfakedetector.ai

4.4.2. Deepware

Deepware là 1 công cụ khác giúp người dùng có thể scan và phát hiện video deepfake. Deepware có công ty mẹ là Zemana, chuyên nghiên cứu và phát triển các antivirus dựa trên AI, nghiên cứu về phương pháp phát hiện và tạo sinh deepfake với mục đích phòng chống các cuộc lừa đảo sử dụng deepfake. Công cụ này cho

phép người dùng đăng tải video của mình lên tự do, tuy nhiên nó mới chỉ đang ở giai đoạn beta nên có thể vẫn có sai sót.

Link website: https://scanner.deepware.ai

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết luận

Trong bối cảnh các công nghệ Deepfake ngày càng phát triển mạnh mẽ và trở thành một thách thức lớn đối với an ninh thông tin, việc nghiên cứu và triển khai các phương pháp phát hiện Deepfake hiệu quả là vô cùng cần thiết. Đồ án này tập trung vào việc xây dựng một hệ thống phát hiện Deepfake sử dụng hình ảnh tự pha trộn (Self-Blended Images - SBIs) kết hợp Flask để tạo ra website cung cấp giao diện giúp người dùng phát hiện được video là thật hay là sản phẩm của Deepfake. Ngoài ra, Đồ án cũng chạy benhmark những thuật toán detector khác để đánh giá độ hiệu quả của những Detector này.

Đồ án đã có những kết quả thu được sau:

Việc áp dụng SBIs vào hệ thống đã giúp cải thiện đáng kể khả năng phát hiện Deepfake. Kỹ thuật này mô phỏng các lỗi giả mạo phổ biến như sự không nhất quán màu sắc, ranh giới pha trộn và các bất thường trong miền tần số, giúp mô hình học được các biểu diễn tổng quát và mạnh mẽ hơn. Điều này không chỉ nâng cao độ chính xác mà còn làm tăng hiệu suất khi xử lý các trường hợp giả mạo phức tạp, góp phần làm cho hệ thống trở nên đáng tin cậy hơn trong các ứng dụng thực tế.

Các thử nghiệm ban đầu với bộ dữ liệu tiêu chuẩn như FaceForensics++ và DFDC cùng các mẫu SBIs đã cho thấy hệ thống đạt hiệu suất cao trong việc phát hiện Deepfake. Kết quả thực nghiệm khẳng định rằng phương pháp sử dụng SBIs có tiềm năng lớn trong việc phát hiện và đối phó với các nội dung giả mạo tinh vi. Đặc biệt, hệ thống hoạt động tốt ngay cả khi xử lý các hình ảnh hoặc video có độ phức tạp cao, thể hiện tính khả thi của giải pháp được đề xuất.

Mặc dù đạt được nhiều thành tựu, hệ thống vẫn còn một số hạn chế. Hiệu suất xử lý video thời gian thực chưa được tối ưu, đặc biệt với các video dài hoặc có độ

phân giải cao. Ngoài ra, mô hình chưa được thử nghiệm trên các loại dữ liệu phức tạp hơn, như âm thanh giả mạo hoặc nội dung tổng hợp từ nhiều nguồn. Những điểm hạn chế này là cơ sở cho các nghiên cứu và cải tiến trong tương lai.

5.2. Hướng phát triển

Mặc dù hệ thống phát hiện Deepfake sử dụng hình ảnh tự pha trộn (Self-Blended Images - SBIs) đã đạt được những kết quả đáng khích lệ, vẫn còn nhiều tiềm năng để mở rộng và cải tiến nhằm tăng cường hiệu quả, tính linh hoạt và khả năng ứng dụng thực tế. Trong tương lai, nhóm nghiên cứu dự định tập trung vào các hướng phát triển chính sau:

- **Tối ưu hóa hiệu suất xử lý**: Một trong những hạn chế hiện tại của hệ thống là thời gian xử lý các video dài hoặc có độ phân giải cao chưa được tối ưu. Nhóm sẽ nghiên cứu áp dụng các kỹ thuật tăng tốc xử lý như xử lý song song (parallel processing), sử dụng GPU hoặc FPGA để giảm thời gian xử lý. Điều này sẽ giúp cải thiện khả năng hoạt động thời gian thực, đáp ứng nhu cầu của các ứng dụng an ninh và giám sát trực tiếp.
- **Phát triển mô hình đa phương tiện**: Hiện tại, hệ thống chỉ tập trung vào phát hiện Deepfake trong video. Nhóm sẽ mở rộng phạm vi phát hiện sang các nội dung giả mạo đa phương tiện khác, chẳng hạn như hình ảnh, âm thanh giả mạo (audio deepfake) và văn bản tổng hợp (text-based fake content). Điều này sẽ cung cấp một giải pháp toàn diện hơn để đối phó với các dạng tấn công giả mạo phức tạp trong không gian mạng.
- **Tăng cường giao diện người dùng (UI):** Giao diện hiện tại sẽ được cải tiến để trở nên trực quan hơn, bổ sung các công cụ hiển thị như biểu đồ và báo cáo chi tiết kết quả phát hiện Deepfake. Người dùng sẽ có thể dễ dàng theo dõi các thông số quan trọng như tỷ lệ chính xác, độ tin cậy của hệ thống, và các khu vực bị nghi ngờ giả mạo trong nội dung kiểm tra.

TÀI LIỆU THAM KHẢO

STT	Tên tài liệu
1	Yan, Z., Zhang, Y., Yuan, X., Lyu, S. and Wu, B., 2023. Deepfakebench: A comprehensive benchmark of deepfake detection. *arXiv preprint arXiv:2307.01426*.
2	Shiohara, K. and Yamasaki, T., 2022. Detecting deepfakes with self-blended images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 18720-18729).
3	VietNamNet. Lừa đảo Deepfake, giả mạo khuôn mặt, giọng nói gia tăng trong năm 2024. Available at: https://vietnamnet.vn/lua-dao-deepfake-gia-mao-khuon-mat-giong-noi-gia-tang-trong-nam-2024-2254095.html (Accessed: 28 February 2024)
4	Joshua-Wu. (n.d.). *GitHub - joshua-wu/deepfakes_faceswap: from deekfakes' faceswap: https://www.reddit.com/user/deepfakes/*. GitHub. https://github.com/joshua-wu/deepfakes_faceswap
5	S0md3v. (n.d.). *GitHub - s0md3v/roop: one-click face swap*. GitHub. https://github.com/s0md3v/roop
6	Deepinsight. (n.d.). GitHub - deepinsight/insightface: State-of-the-art 2D and 3D Face Analysis Project. GitHub. https://github.com/deepinsight/insightface
7	Deepfakes. (n.d.). GitHub - deepfakes/faceswap: Deepfakes Software For All. GitHub. https://github.com/deepfakes/faceswap

HẾT