TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO CUỐI KỲ MÔN HỌC

HỌC SÂU

ĐỀ TÀI:

XÂY DỰNG ỨNG DỤNG NHẬN DIỆN KHUÔN MẶT

GVHD: TS. Nguyễn Thiên Bảo

Mã Môn Học: DLEA432085

SVTH: Nhóm 9

18110251	Lê Nguyễn Gia Bảo
18110262	Trần Nhất Duy
18110276	Võ Hồng Tiên Giang
18110367	Nguyễn Thị Thảo
18110379	Nguyễn Bùi Tiệp
18128062	Nguyễn Thị Minh Thư

TPHCM, Ngày 3 tháng 6 năm 2021

ĐH SƯ PHẠM KỸ THUẬT TP.HCM **XÃ HỘI CHỦ NGHĨA VIỆT NAM**

KHOA CNTT **Độc Lập – Tự Do – Hạnh Phúc**

PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

Họ và tên Sinh viên 1: Lê Nguyễn Gia Bảo MSSV 1: 18110251

Họ và tên Sinh viên 2: Trần Nhất Duy MSSV 2: 18110262

Họ và tên Sinh viên 3: Võ Hồng Tiên Giang MSSV 3: 18110276

Họ và tên Sinh viên 4: Nguyễn Thị Thảo MSSV 4: 18110367

Họ và tên Sinh viên 5: Nguyễn Bùi Tiệp MSSV 5: 18110379

Họ và tên Sinh viên 6: Nguyễn Thị Minh Thư MSSV 6: 18128062

Ngành: Công Nghệ Thông Tin.

Tên đề tài: Xây Dựng Ứng Dụng Nhận Diện Khuôn Mặt.

Họ và tên Giáo viên hướng dẫn: TS. Nguyễn Thiên Bảo.

NHẬN XÉT

.....

.....

.....

.....

.....

TPHCM, Ngày tháng năm 2021

Giáo viên hướng dẫn

(Ký & ghi rõ họ tên)

LỜI CẢM ƠN

Nhóm 9 chúng em xin cảm ơn Thầy Nguyễn Thiên Bảo, đã tận tình chỉ dạy, hướng dẫn, sửa sai, chia sẻ về kiến thức và kinh nghiệm đến với nhóm em cũng như cả lớp học. Bài báo cáo này là kết quả của môn học mà có sự giảng dạy của Thầy làm nền tảng để nhóm em có thể hoàn thành báo cáo một cách hoàn chỉnh và đưa ra sản phẩm tốt nhất có thể. Nhóm chúng em xin chân thành cảm ơn Thầy.

MỤC LỤC

DANH MỤC VIẾT TẮT VÀ THUẬT NGỮ

CNN – Convolution Neural Network: Mạng Neural tích chập

L1 Norm, L2 Norm: Các chuẩn tính toán trong không gian nhiều chiều.

Ảnh RGB: Ảnh hệ màu RGB (Red – Green – Blue)

Loss: Hàm mất mát, hay hàm đánh giá lỗi.

Gradient: Đạo hàm. Ở đây là đạo hàm tại tầng đó trong phép tính Lan truyền ngược.

Layer: Tầng, dùng để chỉ một đơn vị tính toán trong mạng neural.

Weight: Trọng số. Là kết quả của tính toán trong mạng neural.

DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG

LỜI GIỚI THIỆU

Việc xác thực danh tính là một nhu cầu cần thiết hiện tại. Từ sơ khai với lăn tay, đến con dấu, chữ ký. Ngày nay, với sự phát triển của khoa học công nghệ, đã có rất nhiều cách xác thực danh tính được ra đời, đem lại sự thuận tiện, tính bảo mật và độ chính xác cao hơn.

Một trong những phương pháp được sử dụng hiện nay là nhận diện khuôn mặt. Bằng việc sử dụng các đặc trưng khuôn mặt của một người, làm đặc điểm nhận dạng để định danh người đó. Công nghệ nhận diện khuôn mặt đã, đang và sẽ phát triển mạnh mẽ, tối ưu và tiện dụng hơn.

Trong bài báo cáo này, nhóm sẽ trình bày về các phương pháp cơ bản để thực hiện một hệ thống nhận diện khuôn mặt, thông qua việc sử dụng Trí tuệ nhân tạo và kiến thức môn học Học Sâu, từ đó đưa ra sản phẩm, cũng là kết quả học tập và nghiên cứu về kiến thức của môn Học Sâu.

1. CHƯƠNG 1: NHẬN DIỆN KHUÔN MẶT

1.1. NHỮNG CÁCH XÁC THỰC DANH TÍNH CON NGƯỜI

1.1.1. Nhận dạng vân tay

Sự phát triển nhanh chóng của công nghệ điện tử phối hợp, Cùng công nghệ sinh trắc học nhận dạng vân đã được ứng dụng nhiều trong các lĩnh vực sản xuất. Công nghệ vân tay giờ đây đã được rất nhiều thiết bị sử dụng .Thay cho việc nhập mật khẩu truyền thống. Nhận dạng vân tay có thể là phương pháp phức tạp nhất của tất cả công nghệ sinh trắc và được xác nhận qua nhiều ứng dụng. Nhận dạng vân tay đã chứng thực một cách đặc biệt về tính hiệu quả cao của nó và là công nghệ được đề cao xa hơn nữa trong ngành điều tra tội phạm hơn một thế kỷ.

Ưu điểm của công nghệ nhận dạng vân tay là tính chính xác cao, thời gian thực thi nhanh. Nhược điểm nhỏ của nó là sự bất tiện khi sử dụng găng tay cũng như khi tay ẩm ướt, làm giảm độ chính xác. Vì những ưu điểm này mà nhận dạng vân tay được sử dụng rộng rãi trong cuộc sống.

1.1.2. Nhận dạng mống mắt

Công nghệ nhận diện mống mắt hay còn được gọi là công nghệ cảm biến mống mắt (Iris Recognition) là phương pháp áp dụng thuật toán hình ảnh để nhận dạng một người nào đó dựa vào cấu trúc các đường vân phức tạp và duy nhất của mống mắt, thậm chí ngay cả khi họ đang đeo kính hoặc sử dụng áp tròng từ một khoảng cách nhất định. Nhiều quốc gia đã áp dụng công nghệ này để nhận diện công dân, xác thực hộ chiếu, điền thông tin xác thực qua website và giờ đây nó đã được áp dụng cho công nghệ điện thoại thông minh. Quét mống mắt thực sự được công nhận là công nghệ xác thực và bảo mật tốt nhất hiện nay. Trong khi quét võng mạc còn nhiều nhầm lẫn, thì nhận dạng mống mắt đơn giản chỉ liên quan đến việc chụp ảnh của mống mắt, hình ảnh này được sử dụng chỉ để xác thực, bảo mật.

Tính chính xác, khó giả mạo là điểm mạnh của nhận dạng mống mắt. Việc giả mạo với nhận dạng mống mắt là khó có thể thực hiện. Tuy nhiên, nhận dạng mống mắt yêu cầu thiết bị phần cứng chuyên dụng nên chưa áp dụng rộng rãi.

1.2. ĐI SÂU VÀO NHẬN DIỆN KHUÔN MẶT

1.2.1. Nhận diện khuôn mặt là gì

Công nghệ Nhận dạng khuôn mặt là một ứng dụng máy tính tự động xác định hoặc nhận dạng một người nào đó từ một bức hình ảnh kỹ thuật số hoặc một khung hình video từ một nguồn video. Một trong những cách để thực hiện điều này là so sánh các đặc điểm khuôn mặt chọn trước từ hình ảnh và một cơ sở dữ liệu về khuôn mặt.Cách thức làm việc của công nghệ này là so sánh hình ảnh khuôn mặt với những hình ảnh sẵn có trong cơ sở dữ liệu để đưa ra kết quả.

Nhận diện khuôn mặt dựa trên đặc trưng của mỗi khuôn mặt, từ đó sử dụng những đặc trưng riêng biệt đó làm cơ sở định danh một người. Những đặc điểm trên khuôn mặt như khoảng cách giữa hai mắt, hình dạng đường viền khuôn mặt, độ rộng của môi, … Các đặc điểm này được xem là duy nhất của mỗi người. Nếu đủ độ phức tạp tính toán, nhận diện khuôn mặt đảm bảo tính bảo mật và khó giả dạng, kể cả đối với một cặp song sinh.

1.2.2. Các kỹ thuật nhận diện khuôn mặt

1.2.2.1 Nhận dạng qua ảnh truyền thống – ảnh 2 chiều

Một số thuật toán nhận dạng khuôn mặt xác định các đặc điểm khuôn mặt bằng cách trích xuất các ranh giới, hoặc đặc điểm, từ một hình ảnh khuôn mặt của đối tượng. Ví dụ, một thuật toán có thể phân tích các vị trí tương đối, kích thước, và/hoặc hình dạng của mắt, mũi, gò má, và cằm. Những tính năng này sau đó được sử dụng để tìm kiếm các hình ảnh khác với các tính năng phù hợp. Các thuật toán bình thường hóa một bộ sưu tập các hình ảnh khuôn mặt và sau đó nén dữ liệu khuôn mặt, chỉ lưu dữ liệu hình ảnh nào là hữu ích cho việc nhận dạng khuôn mặt. Một hình ảnh mẫu sau đó được so sánh với các dữ liệu khuôn mặt. Một trong những hệ thống thành công sớm nhất dựa trên các kỹ thuật phù hợp với mẫu áp dụng cho một tập hợp các đặc điểm khuôn mặt nổi bật, cung cấp một dạng đại diện của khuôn mặt được nén.

Các thuật toán nhận dạng có thể được chia thành hai hướng chính, là hình học, đó là nhìn vào tính năng phân biệt, hoặc trắc quang (đo sáng), là sử dụng phương pháp thống kê

để 'chưng cất' một hình ảnh thành những giá trị và so sánh các giá trị với các mẫu để loại bỏ chênh lệch.

1.2.2.2 Nhận dạng 3 chiều

Một xu hướng mới nổi lên, tuyên bố cải thiện được độ chính xác, là nhận dạng khuôn mặt ba chiều. Kỹ thuật này sử dụng các cảm biến 3D để nắm bắt thông tin về hình dạng của khuôn mặt. Thông tin này sau đó được sử dụng để xác định các tính năng đặc biệt trên bề mặt của một khuôn mặt, chẳng hạn như các đường viền của hốc mắt, mũi và cằm.

Một lợi thế của nhận dạng khuôn mặt 3D là nó không bị ảnh hưởng bởi những thay đổi trong ánh sáng như các kỹ thuật khác. Nó cũng có thể xác định một khuôn mặt từ một loạt các góc nhìn, trong đó có góc nhìn nghiêng. Các điểm dữ liệu ba chiều từ một khuôn mặt cải thiện lớn độ chính xác cho nhận dạng khuôn mặt. Nghiên cứu 3D được tăng cường bởi sự phát triển của các bộ cảm biến tinh vi giúp nắm bắt hình ảnh chụp khuôn mặt 3D được tốt hơn. Các cảm biến hoạt động bằng cách chiếu ánh sáng có cấu trúc lên gương mặt.

1.2.2.3 Phân tích kết cấu da

Một xu hướng mới nổi sử dụng các chi tiết hình ảnh của da, được chụp trong các hình ảnh kỹ thuật số hoặc máy scan tiêu chuẩn. Kỹ thuật này được gọi là phân tích kết cấu da, đưa các đường đặc trưng, hình dạng, và các điểm nốt trên làn da của một người vào một không gian toán học.

Các thử nghiệm đã chỉ ra rằng với việc bổ sung các phân tích cấu trúc của da, hiệu quả trong việc nhận ra khuôn mặt có thể tăng 20-25 phần trăm.

1.2.3. Ưu điểm, nhược điểm của nhận dạng khuôn mặt

Các ưu điểm của phương pháp nhận diện khuôn mặt:

- Không cần phải trực tiếp tiếp xúc với thiết bị để xác thực (các kỹ thuật xác thực sinh trắc học dựa trên tiếp xúc khác như máy quét dấu vân tay, có thể không hoạt động chính xác nếu có vết bẩn trên tay của một người).
- Cải thiện mức độ bảo mật.
- Yêu cầu xử lý ít hơn so với các kỹ thuật xác thực sinh trắc học khác.
- Dễ dàng tích hợp với các tính năng bảo mật hiện có.

- Độ chính xác được cải thiện theo thời gian.
- Có thể được sử dụng để giúp tự động hóa việc xác thực.

Ưu điểm thì không thể phủ nhận. Công nghệ nhận diện khuôn mặt vẫn có nhược điểm đó là rất khó cam kết độ chính xác.

Để có được hình ảnh đối chiếu, công nghệ nhận dạng khuôn mặt yêu cầu khách hàng phải quay ít nhất 35 độ về phía camera và không sử dụng khẩu trang, mũ, nón.... Yêu cầu này rất khó, bởi chúng ta không thể bắt ép khách hàng phải làm theo yêu cầu, điều này sẽ khiến họ cảm thấy khó chịu. Chính vì vậy, rất khó để cam kết độ chính xác về thông tin khi sử dụng camera đếm người.

1.2.4. Những ứng dụng của nhận diện khuôn mặt

Một số ứng dụng đã tiếp cận đến cuộc sống chúng ta mà ta có thể dễ dàng thấy được:

- Bảo mật trên thiết bị di động.
- Mạng xã hội (chẳng hạn như Facebook, để gắn thẻ các cá nhân trong ảnh).
- Bảo mật, vì các đơn vị an ninh có thể sử dụng nhận dạng khuôn mặt để vào tòa nhà.
- Tiếp thị, các nhà tiếp thị có thể sử dụng nhận dạng khuôn mặt để xác định độ tuổi, giới tính và dân tộc để nhắm mục tiêu tới đối tượng cụ thể.

Cụ thể trong từng hoàn cảnh cụ thể:

- Ứng dụng trong doanh nghiệp, trường học:
    - Chấm công trong công ty.
    - Quản lý điểm danh trường học.
- Các thiết bị cá nhân:
    - Nhận diện khuôn mặt để mở khóa thiết bị.
    - Trong chụp ảnh.

1.2.5. Các loại nhận diện khuôn mặt

Khi có một tập ảnh khuôn mặt, bài toán nhận diện khuôn mặt có hai hướng chính:

1.2.5.1 So khớp

So sánh một ảnh đầu vào và một ảnh dự đoán: Nhận diện khuôn mặt từ video hoặc hình ảnh được lưu lại bởi camera , sau đó so khớp với hình ảnh có sẵn trong cơ sở dữ liệu khuôn mặt đã được lưu trước đó. Hướng giải quyết của phương pháp so khớp chính là trích xuất đặc trưng và so sánh.

Việc so khớp sẽ không yêu cầu biết được thông tin người dùng. Chỉ cần dự đoán hai người là giống nhau hay khác nhau mà không cần biết tên hay những đặc trưng được lưu.

So sánh áp dụng trong hệ thống các bãi xe, trong mở khóa điện thoại, hay trong việc xác nhận người dùng trong ngân hàng trong giao dịch. Khi sử dụng, hệ thống chỉ so sánh ảnh trước đó, từ đó không gây tốn tài nguyên trong huấn luyện, phù hợp với hệ thống khuôn mặt lớn.

1.2.5.2 Định danh

Đầu vào là tên người và hình ảnh của người đó , sau đó nó sẽ được hệ thống lưu lại đặc điểm , đến khi camera nhận dạng 1 lần nữa thì sẽ phân loại được người đó. Hướng giải quyết của phương pháp nhận diện tên người chính là trích xuất đặc trưng và phân loại.

Việc này sẽ yêu cầu phải có thông tin đầu vào cụ thể. Hơn nữa việc này tiêu tốn tài nguyên trong việc huấn luyện. Yêu cầu phần cứng để thực hiện việc này.

Trung Quốc đã triển khai 200 triệu camera công cộng, nhằm giám sát 1,4 tỷ dân của nước này, áp dụng định danh con người nhằm dùng cho hệ thống tín nhiệm công dân của Trung Quốc.

1.3. VẤN ĐỀ VỀ TÍNH RIÊNG TƯ VÀ CÁ NHÂN

Các tổ chức quyền công dân, và các nhà vận động quyền riêng tư như EFF (Electronic Frontier Foundation) và ACLU (American Civil Liberties Union) bày tỏ lo ngại rằng sự riêng tư đang được tổn hại bằng cách sử dụng các công nghệ giám sát. Một số người sợ rằng nó có thể dẫn đến một "xã hội giám sát toàn diện" với chính phủ và các cơ quan khác có khả năng biết nơi ở và hoạt động của tất cả các công dân suốt ngày đêm. Những kiến thức này đã được, đang được, và có thể tiếp tục được triển khai để ngăn chặn các sự xâm phạm quyền công dân của các chính sách nhà nước hoặc từ các công ty, tập đoàn.

Nhiều cơ cấu quyền lực tập trung với khả năng giám sát đã lạm dụng đặc quyền truy cập của họ để duy trì sự kiểm soát của bộ máy chính trị và kinh tế, và để ngăn chặn những cải cách dân túy.

Nhận dạng khuôn mặt có thể được sử dụng không chỉ để xác định một cá nhân, mà còn để tìm ra dữ liệu cá nhân khác liên quan đến một cá nhân - ví dụ như hình ảnh có tính cá nhân, blog bài viết, hồ sơ mạng xã hội, hành vi trên mạng, đi lại, … - tất cả thông qua đặc điểm khuôn mặt của họ. Hơn nữa, các cá nhân có khả năng hạn chế để tránh hoặc ngăn chặn việc theo dõi nhận dạng khuôn mặt, trừ khi họ che giấu khuôn mặt của mình. Điều này về cơ bản là thay đổi động lực của sự riêng tư hàng ngày bằng cách cho phép bất kỳ nhà tiếp thị, cơ quan chính phủ, hay người lạ ngẫu nhiên thu thập bí mật danh tính và thông tin cá nhân liên quan bất kỳ của cá nhân nào bởi các hệ thống nhận dạng khuôn mặt.

Phương tiện truyền thông các trang web xã hội như Facebook có số lượng rất lớn các bức ảnh của người dân, chú thích bằng tên. Điều này đại diện cho một cơ sở dữ liệu mà có thể được sử dụng (hoặc bị lạm dụng) bởi các chính phủ cho các mục đích nhận dạng khuôn mặt.

Trong tháng 7 năm 2012, một cuộc điều trần được tổ chức trước Tiểu ban về Riêng tư, Công nghệ và Luật của Ủy ban Tư pháp, Thượng viện Hoa Kỳ, để giải quyết các vấn đề xung quanh những công nghệ nhận dạng khuôn mặt để bảo vệ cho sự riêng tư và tự do dân sự.

1.4. GIẢ MẠO KHUÔN MẶT

Với thời đại 4.0 hiện nay, việc ứng dụng công nghệ vào đời sống là rất lớn và nhận diện khuôn mặt cũng là một công nghệ không ngoại lệ. Vì sự phát triển khá rộng khắp của nó dẫn đến nhiều người luôn tìm cách khai thác những điểm yếu để thực hiện những hành động xấu với tài sản của người dùng bảo mật bằng nhận diện khuôn mặt. Và điểm yếu phổ biến nhất đó là người xấu có thể dùng video hoặc hình ảnh của bạn trên mạng xã hội và dùng nó để hệ thống nhận dạng nhận nhầm lẫn đó là bạn. Vì thế vấn đề giải quyết việc giả mạo là vô cùng cần thiết. Nhóm đề xuất ba cách như sau: Sử dùng phần cứng, dùng phản ứng người dùng và huấn luyện mô hình phát hiện giả mạo.

1.4.1. Sử dụng phần cứng

Các thiết bị phần cứng ở đây có thể là: Camera 3D, cảm biến, đèn flash, … Các thiết bị này sẽ phát hiện ra đó có phải là khuôn mặt hay là hình ảnh giả mạo bằng việc vẽ sơ đồ khuôn mặt 3 chiều, hay tính toán khoảng cách, bức xạ, …

Ưu điểm của phương pháp này là tính chính xác cao nhất và khó có thể giả mạo.

Tuy nhiên, nhược điểm của phương pháp này nằm ở việc đầu tư thiết bị phần cứng. Việc sử dụng thêm các thiết bị phần cứng có thể làm tăng chi phí triển khai gây khó tiếp cận hơn.

1.4.2. Dựa vào phản ứng người dùng

Việc bắt buộc người dùng làm theo những hành động ngẫu nhiên, nhằm tránh sự giả mạo bằng hình ảnh tĩnh hoặc video trước đó. Phương pháp này không cần sử dụng thêm các thiết bị phần cứng khác. Đây đang là phương pháp được sử dụng nhiều nhất, và chi phí bỏ ra là thấp nhất.

Khuyết điểm nằm ở việc phức tạp thao tác người dùng, trải nghiệm người dùng không tốt. Nếu việc xác định phản ứng không tốt, người dùng sẽ gặp rắc rối trong vấn đề nhận diện.

1.4.3. Huấn luyện mô hình phát hiện giả mạo

Đây cũng là một mô hình được sử dụng rộng rãi. Độ chính xác ở mức chấp nhận và khả năng triển khai trên hệ thống đơn giản. Bên cạnh đó, trải nghiệm người dùng vẫn được đảm bảo.

Ưu điểm của phương pháp là đơn giản, không tốn chi phí, không gây khó chịu.

Nhược điểm còn tồn tại như còn phụ thuộc vào điều kiện môi trường, chất lượng phần cứng cũng như các tác nhận khác.

1.4.4. Tổng kết

Việc phát hiện giả mạo là vấn đề đáng chú ý trong nhận diện, và phát hiện giả mạo cần chấp nhận những hệ lụy không mong muốn. Việc phát hiện giả mạo vẫn có xác suất rủi ro cao do sự tinh vi của kẻ giả mạo.

2. CHƯƠNG 2: THUẬT TOÁN NHẬN DIỆN KHUÔN MẶT

Từ những phương pháp nhận diện khuôn mặt được nêu ở trên, chương này sẽ nêu ra những thuật toán tương ứng với những phương pháp trên, hoặc những thuật toán có hiệu suất cao. Và cũng sẽ trình bày thuật toán chính sử dụng trong dự án.

2.1. CÁC THUẬT TOÁN NHẬN DIỆN KHUÔN MẶT

Các thuật toán nhận diện khuôn mặt càng được phát triển và tối ưu hiệu năng. Dưới đây là những thuật toán đơn giản, là nền móng cho việc phát triển nhận diện khuôn mặt.

2.1.1. One – Shot Learning

One – Shot Learning thuộc nhóm thuật toán học có giám sát, là bài toán phân loại, sử dụng một hoặc một vài sample (mẫu), hoặc thậm chí là chỉ một tấm ảnh làm input đầu vào. Tên gọi của thuật toán bắt nguồn từ đặc trưng số lượng đầu vào rất ít.

Thuật toán sử dụng một kiến trúc CNN đơn giản, dùng để học và dự đoán ảnh khuôn mặt.

Thuật toán trên gặp một nhược điểm là việc phải huấn luyện lại thường xuyên khi thêm một khuôn mặt mới, việc huấn luyện phải thực hiện lại. Và nhược điểm không phù hợp với bài toán nhận diện khuôn mặt, khi số lượng output là thay đổi.

2.1.2. Similarity Learning

Similarity Learning ứng dụng nhiều trong bài toán So khớp khuôn mặt được trình bày ở trên. Thuật toán sử dụng một phép đo sai số giữa hai bức ảnh. Hay cụ thể hơn là đo khoảng cách giữa hai ma trận ảnh, hoặc giữa hai vector đặc trưng của hai ảnh.

Các phép tính toán thường được sử dụng như L1 norm, L2 norm, khoảng cách Euclid, độ tương tự Cosine. Khoảng cách tính được giữa hai ma trận sẽ so sánh với một ngưỡng (threshold). Việc xác định đúng hay sai giữa hai mặt người dựa vào việc chọn phương pháp tính vector đặc trưng, hàm tính toán khoảng cách và ngưỡng.

Similarity Learning có thể trả ra nhiều hơn một giá trị, nếu chúng vượt qua ngưỡng. Cũng như thuật toán sẽ không phụ thuộc vào số lượng output. Vì vậy sẽ không cần quá trình huấn luyện khi thêm một khuôn mặt mới.

Khi sử dụng ma trận ảnh sẽ không hiệu quả do kích thước lớn và độ phức tạp giữa các phép toán. Một bức ảnh RGB kích thước 300 x 300 pixel, sẽ tạo thành tensor với tổng giá trị là 270.000. Việc sử dụng vector đặc trưng được sử dụng ưu tiên hơn. Như vậy, cần một mô hình hiệu quả cho việc trích chọn đặc trưng từ ảnh. Khái niệm Siam Neural Network thể hiện cho những mô hình như vậy.

2.1.3. Siam Neural Network

Siam Neural Network, hay Twin Neural Network, là mô hình mạng Neural tích chập (Convolutional Neural Network) hoạt động song song giữa hai vector đầu vào khác nhau, để tính toán vector đầu ra có thể so sánh được. Siam Neural Network sử dụng kiến trúc CNN đã loại bỏ tầng output. Việc này nhằm biến hình ảnh đầu vào thành vector đặc trưng. Lợi dụng tính chất tích chập và chiều sâu của CNN, giúp lấy được những đặc trưng quan trọng của ảnh. Đầu ra của Siam Neural Network là hai vector, tương ứng biểu diễn hai ảnh đầu vào. Sau đó tính toán qua hàm mất mát (loss function) để đo lường sự khác biệt. Thông thường hàm mất mát sử dụng là L2 norm.

Do đã loại bỏ tầng output, nên việc thay đổi số lượng output không ảnh hưởng. Việc lựa chọn hàm mất mát là quan trọng, ảnh hưởng đến cách tính sai khác giữa hai ảnh đầu vào. Bên cạnh đó, một cách tính hàm mất mát được sử dụng là Triple Loss, sẽ được trình bày trong phần tiếp theo với Facenet.

2.2. FACENET

Facenet chính là một dạng Siam Neural Network, có tác dụng biểu diễn bức ảnh thành vector trong không gian n chiều (thường là 128 hoặc 512 chiều), sao cho khoảng cách giữa các vector càng nhỏ, thì mức độ trùng khớp giữa chúng càng lớn.

2.2.1. Tổng quan về Facenet và Triple Loss

Như trình bày ở trên, hàm mất mát trong Siam Neural Network chỉ đo lường khoảng cách giữa hai tấm ảnh. Mô hình chỉ học được một trong hai khả năng xảy ra giữa hai tấm ảnh, là giống nhau, hoặc khác nhau. Như vậy, ta thấy được rằng, mô hình không được học cùng lúc sự giống nhau và khác nhau trong một nhóm ảnh.

Để giải quyết vấn đề này, Facenet sử dụng Triple Loss, và đầu vào sẽ là 3 bức ảnh, trong đó gồm:

- Một ảnh cố định, kí hiệu là A (anchor).
- Một ảnh đúng, giống với ảnh cố định, kí hiệu là P (positive).
- Một ảnh sai, khác với ảnh cố định, kí hiệu là N (negative).

Mục đích của Triple Loss gồm có: tối thiểu hóa giữa A và N, và tối đa hóa giữa A và P. Có vẻ ngược, nhưng thực tế:

- Ta cần tối đa khoảng cách giữa hai ảnh giống nhau, như vậy việc huấn luyện sẽ khó khăn hơn, mô hình sẽ trở nên tốt hơn, hạn chế overfitting. Ảnh P sẽ là ảnh giống A, nhưng sẽ là ảnh khác A nhất trong các tập ảnh giống A.
- Tương tự, ta cần tìm ảnh của hai tập giá trị khác nhau nhưng lại rất giống nhau, cũng nhằm khó khăn trong việc huấn luyện. Vậy N sẽ là tập khác A nhưng sẽ giống A nhất.

Gọi d(x, y) là khoảng cách giữa hai ma trận x và y. Lần lượt ta sẽ có d(A,P) và d(A, N) là khoảng cách giữa A và P, giữa A và N. Hàm Triple Lost kì vọng giá trị đúng sẽ là

$$d(A, P) < d(A, N)$$

Điều này là hiển nhiên. Dấu " = " có thể xảy ra, khi chênh lệch khoảng cách " =0 ". Thực tế, ta chỉ cần chúng xấp xỉ bằng nhau. Để áp dụng phép bằng vào công thức trên, ta cộng một lượng α vào vế trái, với α không âm rất nhỏ. Như vậy ta được:

$$d(A, P) + \alpha < d(A, N)$$

Áp dụng L2 Norm, ta được:

$$||f(A) - f(P)||_2^2 + \alpha \leq ||f(A) - f(N)||_2^2$$

$$||f(A) - f(P)||_2^2 - ||f(A) - f(N)||_2^2 + \alpha \leq 0$$

Như vậy, hàm Triple Loss sẽ là:

$$L(A,P,N) = \sum_{i=0}^{n} ||f(A) - f(P)||_2^2 - ||f(A) - f(N)||_2^2 + \alpha$$

Trong đó n là số batch size.

Mục tiêu của hàm Triple Loss không phải là tính sai số giữa các tập ảnh với nhau, mà là sai số khi nhận nhầm P thành N, N thành P, khi đó giá trị nhầm lẫn $d(A,P)$ sẽ rất lớn, và $d(A,N)$ là rất nhỏ. Ta cần lưu giữ những giá trị này để huấn luyện mô hình. Khi đó hàm Triple Loss sẽ là:

$$L(A,P,N) = \sum_{i=0}^{n} \max\left(\left|||f(A) - f(P)||\right|_2^2 - \left|||f(A) - f(N)||\right|_2^2 + \alpha, 0\right)$$

Khi giá trị phân biệt đúng, hàm Triple Loss là nhỏ hơn 0, ta chỉ cần lưu giá trị 0. Như vậy khi áp dụng Triple Loss vào các mô hình mạng tích chập, ta có thể tạo ra các vector tốt nhất. Và đồng thời những ảnh cùng một lớp sẽ gần nhau hơn, do hàm Triple Loss = 0.

Lưu ý quan trọng trong Facenet với Triple Loss, để thực hiện ta cần A và P, nghĩa là ta phải có nhiều hơn một ảnh trong một lớp. Ảnh N được chọn từ các lớp còn lại, nên cũng cần nhiều hơn một lớp trong tập dữ liệu.

2.2.2. Tối ưu Triple Loss

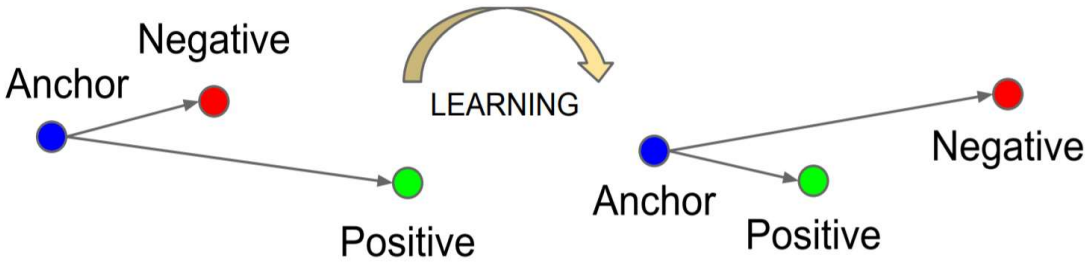
Như phân tích trên, nếu việc chọn tập ảnh A, P, N ngẫu nhiên, khả năng đúng giữa cùng một lớp, và sai khi khác lớp là rất cao. Khi đó hàm Triple Loss = 0 và việc học sẽ không còn ý nghĩa. Để mô hình học khó hơn, đồng thời tăng sự chính xác và giảm overfitting, chúng ta cần chọn tập ảnh A, P, N khó, khi đó dấu " = " trong bất đẳng thức trên dễ xãy ra. Tức là $d(A,P)$ lớn nhất và $d(A,N)$ nhỏ nhất. Trong trường hợp hoàn hảo, ta có:

$$d(A,P) = d(A,N) = 0.5.$$

Vậy việc cần làm gồm có:

- Hard Positive: Bức ảnh P có khoảng cách xa nhất với A: $argmax(d(A,P))$.
- Hard Negative: Bức ảnh N có khoảng cách gần A nhất: $argmin(d(A,P))$.

Chiến lược trên sẽ ảnh hưởng lớn đến chất lượng mô hình Facenet, việc huấn luyện hội tụ nhanh hơn, và đồng thời kết quả dự báo chuẩn xác hơn. Ngược lại, việc lựa chọn ngẫu nhiên sẽ không đem lại kết quả mong muốn.

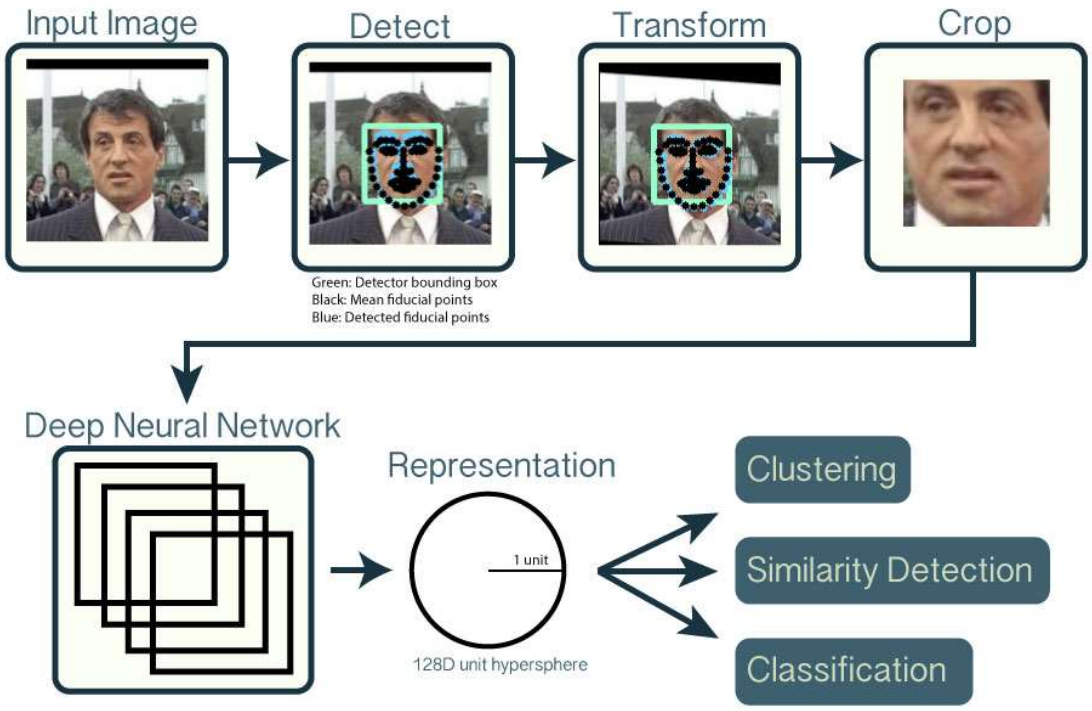


Hình 2.1 Triple Loss trong mô hình FaceNet

2.3. OPENFACE

OpenFace là một mô hình nhận dạng khuôn mặt học sâu, được dựa trên bài báo khoa học của FaceNet. OpenFace là một thư viện mã nguồn mở, được phát triển bởi cộng đồng với hiệu suất và độ chính xác cao, được sử dụng rộng rãi trong những năm gần đây. OpenFace được triển khai trên Pytorch, tính toán trên CPU hoặc GPU và hướng đến khả năng triển khai trên các thiết bị di động. OpenFace tập trung vào tính toán tập trung vào khuôn mặt. Vì vậy bạn có thể huấn luyện mô hình với độ chính xác cao với đầu vào ít và chi phí thực hiện thấp.

Kiến trúc của OpenFace:



Hình 2.2 Kiến trúc OpenFace

Trước khi vào mô hình mạng Neural Sâu (Deep Neural Network), giá trị đầu vào được xử lý và đầu ra là ảnh có kích thước nhất định, với khuôn mặt rõ ràng nhất. Cụ thể:

2.3.1. Xác định khuôn mặt

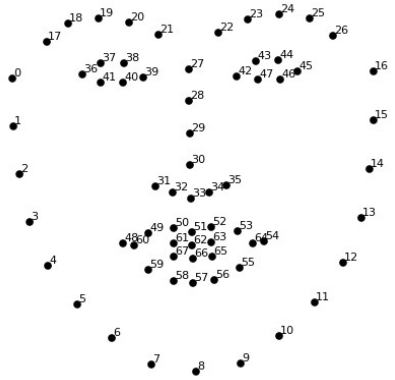
Việc chỉ tập trung vào khuôn mặt, hơn là ảnh tổng thể, sẽ làm tăng độ chính xác của mô hình và tăng tốc độ huấn luyện.

Các thuật toán xác định khuôn mặt như Histogram of Oriented Gradients (HOG - tạm dịch là biểu đồ của hướng dốc), MTCNN (Multi-task Cascaded Convolutional Networks), hoặc sự hỗ trợ của các thư viện tính toán như OpenCV với Haar Cascade, Dlib, Face_recognition. Mỗi thuật toán có cách xác định khác nhau và kết quả trả về cũng là khác nhau. Các thuật toán này sẽ được trình bày trong những phần tiếp theo.

2.3.2. Căn chỉnh khuôn mặt

Những góc chụp khác nhau sẽ là sai số của những ảnh cùng một người. Hơn nữa việc những đặc trưng nằm ở những vị trí khác nhau, ảnh hưởng đến tốc độ học của các mô hình sau.

Mục tiêu của bước này là sẽ cố gắng biến bức ảnh, về một hướng chung nhất (thường là hướng chính diện). Việc căn chỉnh mặt, mũi, môi, viền khuôn mặt về vị trí gần đúng nhất, được thực hiện thông qua thuật ngữ Facial Landmark, là việc lấy 68 điểm trên khuôn mặt, là đặc trưng của một khuôn mặt.



Hình 2.3 Facial Landmark 68 points

Chúng ta sử dụng những phép tính xoay ảnh, phóng to, thu nhỏ hay cắt ảnh, để mắt, môi và những điểm nằm chính diện bức ảnh. Ở đây thuật toán sẽ không sử dụng những phép biến đổi 3D, có thể khiến ảnh bị méo.

2.3.3. Tạo ảnh đầu ra

Từ giá trị đầu ra của bước trước, ta sẽ thu nhỏ, phóng to, và đưa ảnh về kích thước nhất định. Việc kích thước nhất định sẽ chuẩn hóa đầu vào của mô hình. Như vậy từ ảnh người, ta sẽ thu được khuôn mặt người tốt nhất.

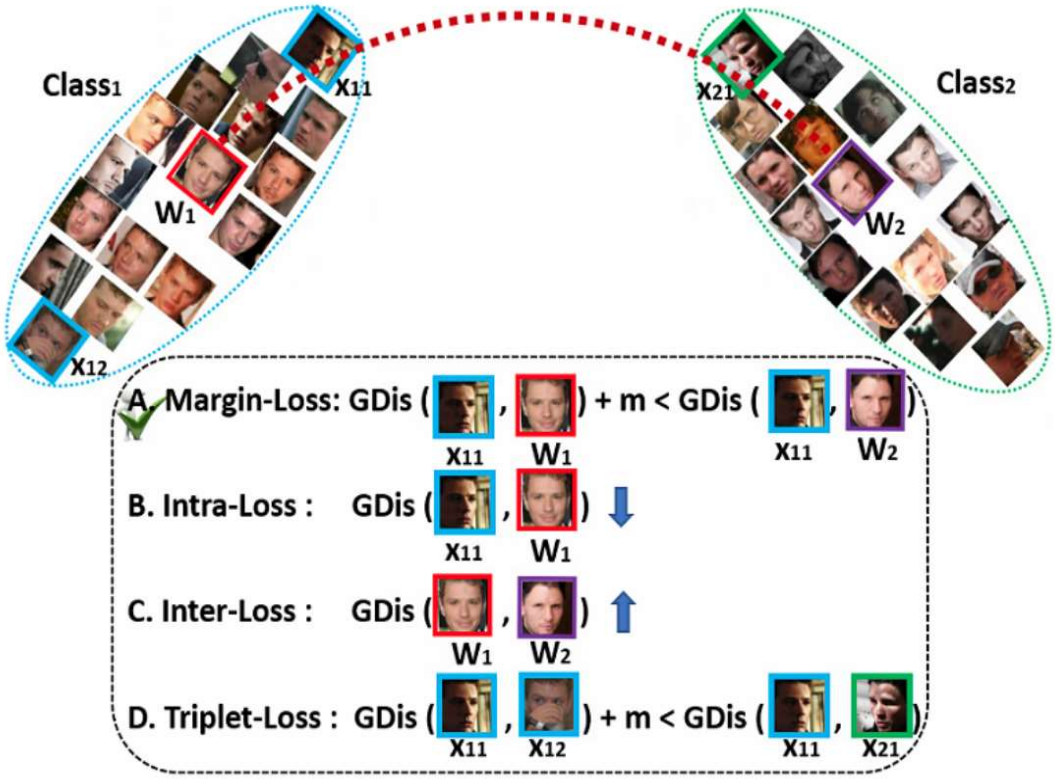
2.3.4. Sử dụng trích xuất đặc trưng

Sử dụng mô hình FaceNet được trình bày ở phần trên. Lưu ý những ràng buộc của FaceNet về số lớp và số ảnh tối thiểu trong một lớp.

2.3.5. Tối ưu theo yêu cầu bài toán

Với từng yêu cầu như so khớp khuôn mặt, định danh hay phân loại khuôn mặt, sẽ có những thuật toán cụ thể. Trong yêu cầu dự án là nhận diện khuôn mặt theo phương pháp định danh, các phép tính phân loại và định danh khuôn mặt sẽ được trình bày ở những phần tiếp theo.

2.4. ArcFace



Hình 2.4 Mô tả thuật toán ArcFace

ArcFace phân loại theo các lớp:

- Intra Loss : Giảm độ sai khác giữa các thành phần cùng lớp, tức Intra Loss thấp.
- Inter Loss : Tăng độ sai khác giữa các thành phần khác lớp, tức Inter Loss cao.
- Margin Loss: Khoảng cách giữa ảnh trong cùng lớp sẽ nhỏ hơn giữa các lớp với nhau.
- Triplet Loss: Khoảng cách giữa các ảnh cùng nhãn sẽ nhỏ hơn giữa ảnh đó và các ảnh trong lớp.

Hàm mất mát Additive Angular Margin Loss, với phép tính softmax, với những hàm Loss phía trên, sẽ đánh giá độ chính xác theo nhãn và theo lớp.

$$L_1 = -\frac{1}{N}\sum_{i=1}^{N} log\frac{e^{W_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_{y_i}^T x_i + b_{y_i}}}$$

Trong đó x_i là đặc trưng (feature) của mẫu (sample) thứ I, thuộc lớp (class) thứ y_i.

ArcFace có độ chính xác cao nhất hiện tại, với việc phân chia các thành các lớp khác nhau, cũng như hàm tối ưu được tính toán chi tiết và đặc trưng hơn. Thử nghiệm thực tế, ArcFace hiệu quả với MegaFace Challenge, khi tập ảnh đầu vào là rất lớn. Các bộ dữ liệu được sử dụng: CASIA, VGGFace2, MS1MV2 and DeepGlint-Face bao gồm MS1M-DeepGlint và Asian-DeepGlint) được sử dụng làm dữ liệu huấn luyện, giúp so sánh công bằng với các phương thức khác. Ngoài ra cũng sử dụng một số bộ dữ liệu khác như LFW, CFP-FP, AgeDB-30, CPLFW, CALFW, YTF, MegaFace, IJB-B, IJB-C, Trillion-Pairs, iQIYI-VID.

3. CHƯƠNG 3: CÁC MÔ HÌNH SỬ DỤNG TRONG DỰ ÁN

Thuật toán sử dụng trong dự án gồm luồng hoạt động như sau:

- MTCNN: cắt ảnh đầu vào thành ảnh khuôn mặt, kích thước 160 x 160 pixel.
- Trích xuất vector đặc trưng với ResNet V1, vector đầu ra kích thước (512, 1).
- SVM để phân loại vector đặc trưng và đưa ra dự đoán.

Ưu điểm của mô hình:

- MTCNN có ưu điểm lấy trọn khuôn mặt với kích thước vuông. Độ chính xác và độ bao quát cao hơn các mô hình khác.
- ResNetV1 là mạng multi Branches, kiến trúc đa nhánh, có độ chính xác cao.
- SVM: Support Vector Machine, một phương pháp phân loại hiệu quả trong bài toán phân loại lớp dữ liệu.
- Thời gian tính toán nhanh, đạt 7.5 giây cho việc thêm mới (lấy 50 khung hình) và 0.1 giây cho một lần kiểm thử.
- Lưu trữ thông tin đơn giản với và tiết kiệm bộ nhớ. Việc không dùng CSDL với hệ thống nhỏ giúp đơn giản quá trình triển khai.

Nhược điểm của mô hình:

- Tính toán CPU và GPU cao.
- Kích thước model lớn.

3.1. MULTI TASK CASCADED CONVOLUTIONAL NETWORKS

MTCNN, Multi-task Cascaded Convolutional Networks sử dụng để nhận diện mặt và cắt khuôn mặt.

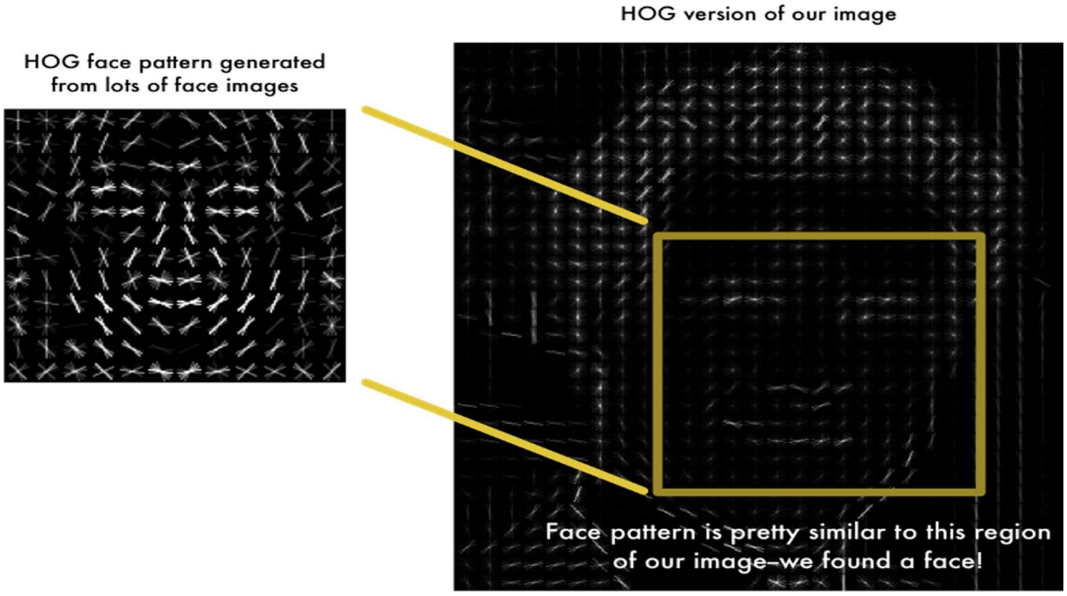
- Đầu vào: Ảnh khuôn mặt chỉ chứa một khuôn mặt.
- Đầu ra: Ảnh khuôn mặt đã cắt, kích thước 160 x 160 pixel.

3.1.1. So sánh MTCNN và các thuật toán tương tự

3.1.1.1 Histogram of Oriented Gradients (HOG)

Đây là phương pháp đầu tiên ra đời, phát triển năm 2005. Đầu vào sẽ là ảnh xám, vì chúng ta không cần ảnh màu trong nhận diện vị trí khuôn mặt.

Phương pháp chính là tìm ra sự thay đổi sáng tối giữa các điểm ảnh và các điểm xung quanh giữa chúng. Việc này sẽ tạo ra những tạo vector gradients (độ dốc) và thể hiện cho hướng của dòng ánh sáng, từ sáng sang tối. Bằng cách vẽ và dùng cửa sổ trượt (sliding windows) ta sẽ có tập hợp những viền sáng.



Hình 3.1 Histogram of Oriented Gradients

3.1.1.2 OpenCV Haar Cascade

Ưu điểm của Haar Cascade là sự thuận tiện do tích hợp vào OpenCV. Đầu vào yêu cầu là một ảnh xám. Thuật toán sử dụng những ma trận viền, ma trận đường thẳng và ma trận chéo, làm ma trận đặc trưng trong trích xuất đặc trưng. Kích thước ma trận 24x24.

Nhược điểm của Haar Cascade là khó với ảnh khuất, ảnh bị nghiêng mặt. Độ chính xác trên tập bao quát không cao, không thích hợp trong nhiều trường hợp

```
import cv2
classifier = cv2.CascadeClassifier('models/haarcascade_frontalface2.xml')
img = cv2.imread('test.jpg')
faces = classifier.detectMultiScale(img)# result
#to draw faces on image
for result in faces:
    x, y, w, h = result
    x1, y1 = x + w, y + h
    cv2.rectangle(img, (x, y), (x1, y1), (0, 0, 255), 2)
```

3.1.1.3 Dlib Frontal Face Detector

Dlib là một thư viện Machine Learning viết bằng C++. Dlib.get_frontal_face_detector() cũng sử dụng ảnh xám đầu vào và trả về bộ bounding box chứa giá trị khung viền đầu ra.

Thuật toán dựa trên HOG được trình bày ở trên, cộng với SVM ở tầng đầu ra để xác định đâu là bộ bbox tốt nhất.

```
import dlib
import cv2
detector = dlib.get_frontal_face_detector()
img = cv2.imread('test.jpg')
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
faces = detector(gray, 1) # result
#to draw faces on image
for result in faces:
    x = result.left()
    y = result.top()
    x1 = result.right()
    y1 = result.bottom()
    cv2.rectangle(img, (x, y), (x1, y1), (0, 0, 255), 2)
```

3.1.2. So sánh và đánh giá

3.1.2.1 Tốc độ

Vi xử lý được sử dụng là Intel Core i5 Gen 7. Kích thước ảnh đầu vào 640 x 360 pixel. Ta có tốc độ xử lý như sau:

Bảng 3.1 So sánh tốc độ giữa các thuật toán xác định khuôn mặt

Thuật toán	Haar Cascade	Dlib	MTCNN
FPS	9.25	5.41	7.92

3.1.2.2 Đánh giá

- Haar Cascade: tỉ lệ đúng không cao.
    - Việc xử lý quay mặt và nghiêng mặt không khả dụng.
    - Viền vẽ ra sẽ rộng chiều rộng hơn các thuật toán khác.
    - Không tốt trong vùng thiếu sáng và ngược sáng.
- Dlib: tỉ lệ đúng chấp nhận nhưng hiệu năng không tốt
    - Nếu khuôn mặt nhỏ hơn 80 x 80 pixel, Dlib sẽ không nhận diện được.

- Xử lý được ảnh nghiêng và ảnh quay.
- Việc khó cài đặt trên window.

- MTCNN: kết quả tốt
  - Đúng trên nhiều trường hợp: nghiêng, quay, điều kiện ánh sáng khác nhau.
  - Tốc độ xử lý tốt.

Vậy MTCNN là gì và tại sao lại có kết quả tốt như vậy.

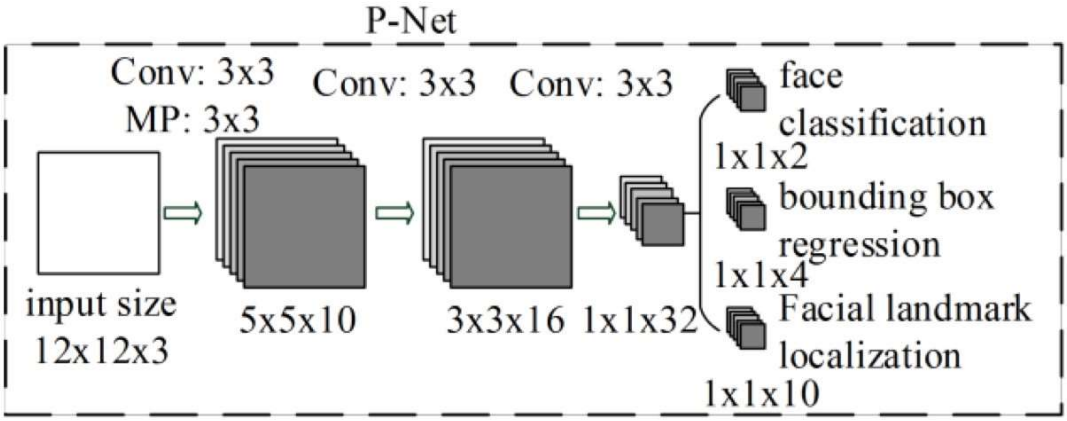
3.1.3. Kiến trúc MTCNN

MTCNN, viết tắt của Multi-task Cascaded Convolutional Networks. Nó là bao gồm ba mạng CNN xếp chồng và đồng thời hoạt động khi detect khuôn mặt. Mỗi mạng có cấu trúc khác nhau và đảm nhiệm vai trò khác nhau trong task. Đầu ra của MTCNN là vị trí khuôn mặt và các điểm trên mặt như: mắt, mũi, miệng…

Ba mạng trong MTCNN theo thức tự là P-Net, R-Net và O-Net.

3.1.3.1 The Proposal Network (P-Net)

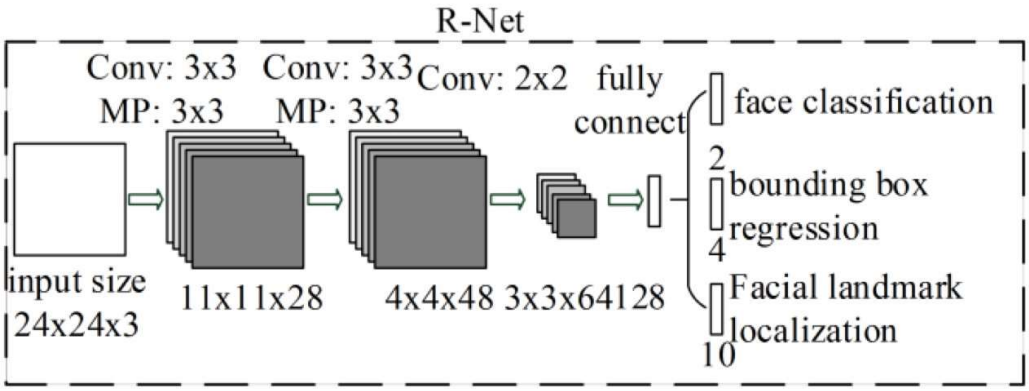
P-Net là một mạng FCN – Fully convolutional network. Khác với CNN, FCN không sử dụng Dense, thay vào đó, giá trị trả ra là các ma trận tích chập. Nhiệm vụ của mạng này là xác định các cửa sổ (window) bao gồm mặt người và những bounding box của nó. Mạng này tuy tốc độ nhanh nhưng thiếu chính xác.



Hình 3.2 P-Net Architecture

3.1.3.2 The Refine Network (R-Net)

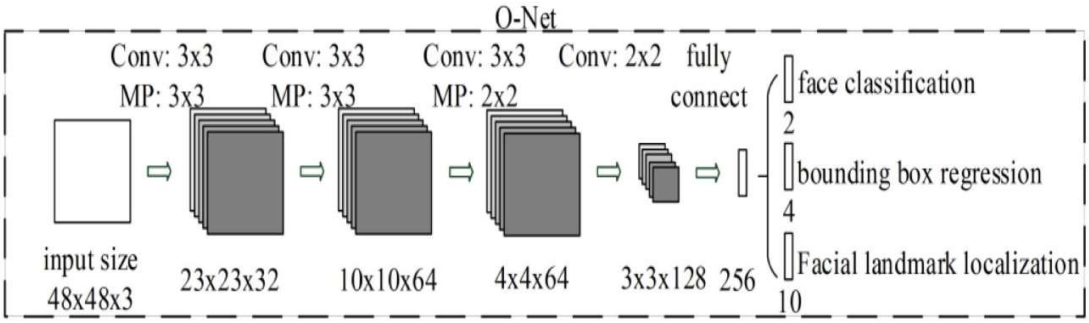
Tất cả những dự đoán của P-Net được chuyển vào R-Net. Ở đây, R-Net là một mạng CNN, sử dụng Dense tại các tầng cuối. R-Net sẽ giảm bớt số dự đoán của P-Net, và hiệu chỉnh giá trị bounding box của các dự đoán này.



Hình 3.3 R-Net Architecture

3.1.3.3 The Output Network (O-Net)

Mạng O-Net tương tự R-Net, nhưng mục tiêu của O-Net là mô tả khuôn mặt chi tiết hơn, và giá trị trả ra là giá trị bounding box của khuôn mặt. Các mạng này lọc được các bounding box chính xác nhờ tận dụng vào độ sâu của kiến trúc mạng.



Hình 3.4 O-Net Architecture

3.1.4. Biến thể Fast MTCNN

Việc xử lý cắt ảnh từ video sẽ là thử thách khi lượng ảnh truyền vào là liên tục. Đôi khi sự thay đổi giữa hai hay các khung hình liên tiếp là không nhiều.

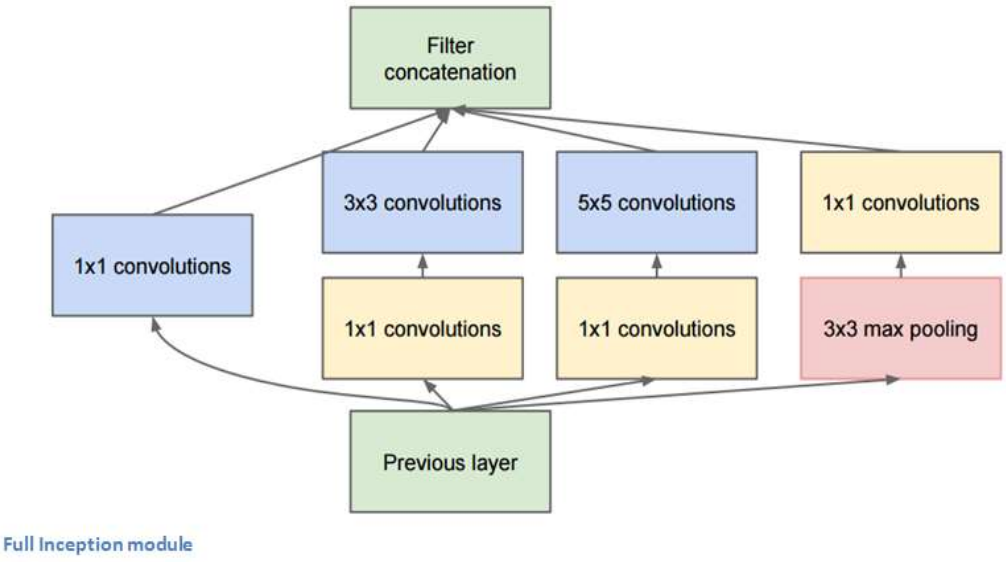
Kỹ thuật được áp dụng là sử dụng bước nhảy (stride): sẽ nhận diện mỗi N khung hình. Ví dụ ta có 9 khung hình từ 0 đến 8, và stride = 3, khi đó ta sẽ lấy những khung hình 0, 3 và 6. Các khung hình 1 và 2 sẽ tính toán dựa trên 0, và xem như là sai số rất ít. Giả sử video chất lượng 30FPS, vậy khoảng cách giữa hai frame liên tiếp sẽ là 1/30 giây. Như vậy trong khoảng thời gian hai frame bỏ qua xấp xỉ 0.07 giây. Nếu khuôn mặt không thay đổi nhanh hơn mức thời gian trên, ta sẽ thu được kết quả tốt. Ngược lại, nếu chuyển động là đủ nhanh, hình ảnh sẽ bị nhòe, và việc bỏ hai khung hình này vẫn không ảnh hưởng đến hiệu suất, còn giúp thuật toán nhanh hơn.

Hơn nữa, khoảng thời gian bỏ qua khung hình sẽ giúp việc tính toán mượt mà, không nghẽn CPU, hiệu suất vẫn đảm bảo.

3.2. INCEPTION RESNET

3.2.1. Inception Module

GoogLeNet (2014) đã đề xuất một khái niệm mới được gọi là kiến trúc khởi động. Kiến trúc khởi động là một mạng trong mạng. Phần ban đầu của kiến trúc giống như một mạng lưới truyền thống và được gọi là gốc. Phần quan trọng của mạng là một lớp trung gian, được gọi là một mô-đun khởi động (Inception Module). Một ví dụ về mô-đun khởi động được minh họa như sau.



Hình 3.5 Inception Module

Ý tưởng cơ bản của mô-đun khởi động là thông tin quan trọng trong các hình ảnh có sẵn ở các mức chi tiết khác nhau. Nếu chúng ta sử dụng một bộ lọc lớn, chúng ta có thể nắm bắt thông tin trong một khu vực lớn hơn có chứa biến thể hạn chế; nếu chúng ta sử dụng một bộ lọc nhỏ hơn, chúng ta có thể nắm bắt thông tin chi tiết trong một khu vực nhỏ hơn. Trong khi một giải pháp sẽ là kết hợp với nhau nhiều bộ lọc nhỏ, điều này sẽ lãng phí các thông số và chiều sâu khi nó đủ để sử dụng các mẫu rộng hơn trong một khu vực rộng hơn. Vấn đề là chúng ta không biết trước mức chi tiết nào phù hợp với từng vùng của hình ảnh. Tại sao không cung cấp cho mạng thần kinh sự linh hoạt để mô hình hóa hình ảnh ở các mức chi tiết khác nhau?

Điều này đạt được với một môđun khởi động, mà xoay với ba kích thước bộ lọc khác nhau song song. Các kích thước bộ lọc này là 1 × 1, 3 × 3, và 5 × 5. Một đường ống hoàn toàn tuần tự của các bộ lọc có cùng kích thước là không hiệu quả khi người ta phải đối mặt với các đối tượng có quy mô khác nhau trong các hình ảnh khác nhau. Vì tất cả các bộ lọc trên lớp khởi động đều có thể học được, mạng nơ-ron có thể quyết định những bộ lọc nào sẽ ảnh hưởng nhiều nhất đến đầu ra. Bằng cách chọn các bộ lọc có kích thước khác nhau dọc theo các đường dẫn khác nhau, các vùng khác nhau được thể hiện ở mức độ chi tiết khác nhau.

Một quan sát là mô-đun khởi động dẫn đến một số tính toán không hiệu quả do số lượng lớn các convolutions có kích thước khác nhau. Quan sát kiến trúc của mô-đun khởi động, trong đó convolutions 1 × 1 được sử dụng để đầu tiên giảm độ sâu của bản đồ đặc trưng. Điều này là do số lượng bộ lọc convolutions 1 × 1 là một yếu tố khiêm tốn, nhỏ hơn độ sâu của khối lượng đầu vào. Ví dụ, trước tiên, người ta có thể giảm độ sâu đầu vào từ 256 đến 64 bằng cách sử dụng 64 bộ lọc 1 × 1 khác nhau. Các convolutions 1 × 1 bổ sung này được gọi là các hoạt động nút cổ chai của mô-đun khởi động. Ban đầu giảm độ sâu của bản đồ tính năng (với giá trị 1 × 1 convolutions) tiết kiệm tính toán hiệu quả với các convolutions lớn hơn vì độ sâu giảm ở các lớp sau khi áp dụng các nút thắt cổ chai. Người ta có thể xem các convolutions 1 × 1 như là một loại giảm kích thước có giám sát trước khi áp dụng các bộ lọc không gian lớn hơn. Việc giảm kích thước được giám sát bởi vì các tham số trong các bộ lọc nút cổ chai được học trong quá trình truyền ngược. Các nút cổ

chai cũng giúp giảm độ sâu pooling layer. Bí quyết của các lớp nút cổ chai cũng được sử dụng trong một số kiến trúc khác, nơi nó rất hữu ích cho việc nâng cao hiệu quả và độ sâu đầu ra.

Tóm lại, mô-đun khởi động, hay Inception, là một kỹ thuật sử dụng các lớp tích chập khác nhau, nhằm khai thác tối đa đặc trưng của ảnh.

3.2.2. ResNet

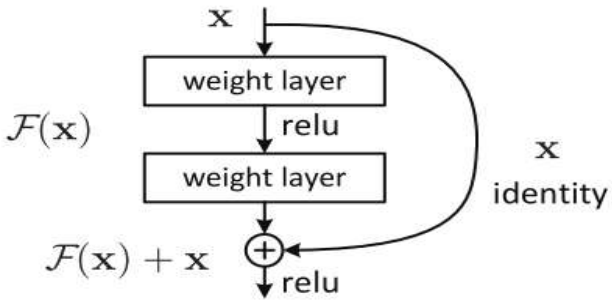
Vấn đề chính khi huấn luyện những mạng sâu như vậy chính là việc luồng gradient giữa các layer sẽ bị chặn bởi một lượng lớn các phép tính ở các layer sâu có thể làm tăng hoặc giảm kích thước của gradient. Bùng nổ đạo hàm hay triệt tiêu đạo hàm (Gradient Vanishing) là khó tránh khỏi với mạng sâu. Các lớp ở cuối sẽ không được cập nhật nhiều, dẫn đến quá trình huấn luyện lớn.

Giải thích thêm về Vanishing Gradient:

Trước hết thì Backpropagation Algorithm là một kỹ thuật thường được sử dụng trong quá trình training. Ý tưởng chung của thuật toán lá sẽ đi từ output layer đến input layer và tính toán gradient của cost function tương ứng cho từng parameter (weight) của mạng. Gradient Descent sau đó được sử dụng để cập nhật các parameter đó. Toàn bộ quá trình trên sẽ được lặp đi lặp lại cho tới khi mà các parameter của network được hội tụ. Thông thường chúng ta sẽ có một hyperparameter (số Epoch - số lần mà training set được duyệt qua một lần và weights được cập nhật) định nghĩa cho số lượng vòng lặp để thực hiện quá trình này. Nếu số lượng vòng lặp quá nhỏ thì ta gặp phải trường hợp mạng có thể sẽ không cho ra kết quả tốt và ngược lại thời gian training sẽ lâu nếu số lượng vòng lặp quá lớn.

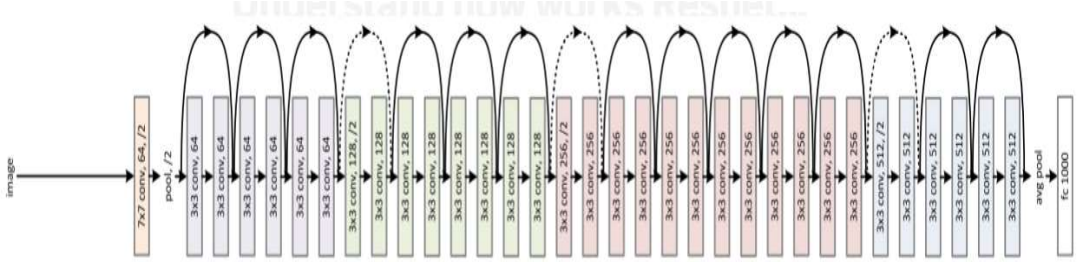
Tuy nhiên, trong thực tế Gradients thường sẽ có giá trị nhỏ dần khi đi xuống các layer thấp hơn. Dẫn đến kết quả là các cập nhật thực hiện bởi Gradient Descent không làm thay đổi nhiều weights của các layer đó và làm chúng không thể hội tụ và mạng sẽ không thu được kết quả tốt. Hiện tượng như vậy gọi là Vanishing Gradients.

Để giải quyết vấn đề đó, ResNet ra đời với kiến trúc mới lạ. ResNet sử dụng phương thức kết nối tắt (skip connections) giữa các layer nhằm mục đích cho phép việc sao chép giữa các layer. Một khối như vậy được gọi là một Residual Block.



Hình 3.6 Residual Block

Tư tưởng ban đầu của ResNet chỉ cộng thêm các kết nối giữa layer i và (i+r). Ví dụ nếu chúng ta chọn r = 2, sẽ chỉ có các kết nối tắt giữa các layer lẻ liên tiếp được sử dụng. Việc sử dụng kết nối tắt đã gỡ bỏ cản trở cho luồng gradient, qua đó đem lại những hệ quả quan trọng trong sự vận hành của thuật toán lan truyền ngược. Các kết nối tắt áp dụng các hàm super-highway để kích hoạt luồng gradient, cho phép nhiều đường với độ dài đa dạng cùng tồn tại từ đầu vào đến đầu ra. Trong trường hợp như vậy, những đường đi ngắn nhất thực hiện phần lớn việc học trong khi đó các đường đi dài hơn được xem như là những đóng góp bổ sung.



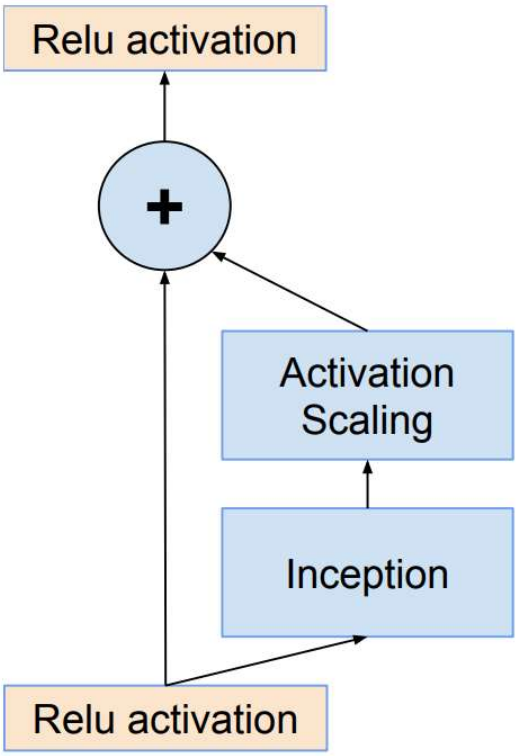
Hình 3.7 ResNet Architecture

Do đó, dù ResNet có 152 tầng layer nhưng hiệu suất của ResNet là rất cao. ResNet dành chiến thắng trong ILSVRC 2015.

3.2.3. Sự kết hợp giữa ResNet và Inception

Cả Inception và Residual Network đều là những State Of The Art CNN, với hiệu suất tốt và chi phí huấn luyện thấp. Chúng ta sẽ tìm hiểu sự kết hợp của Inception Module và Residual Block.

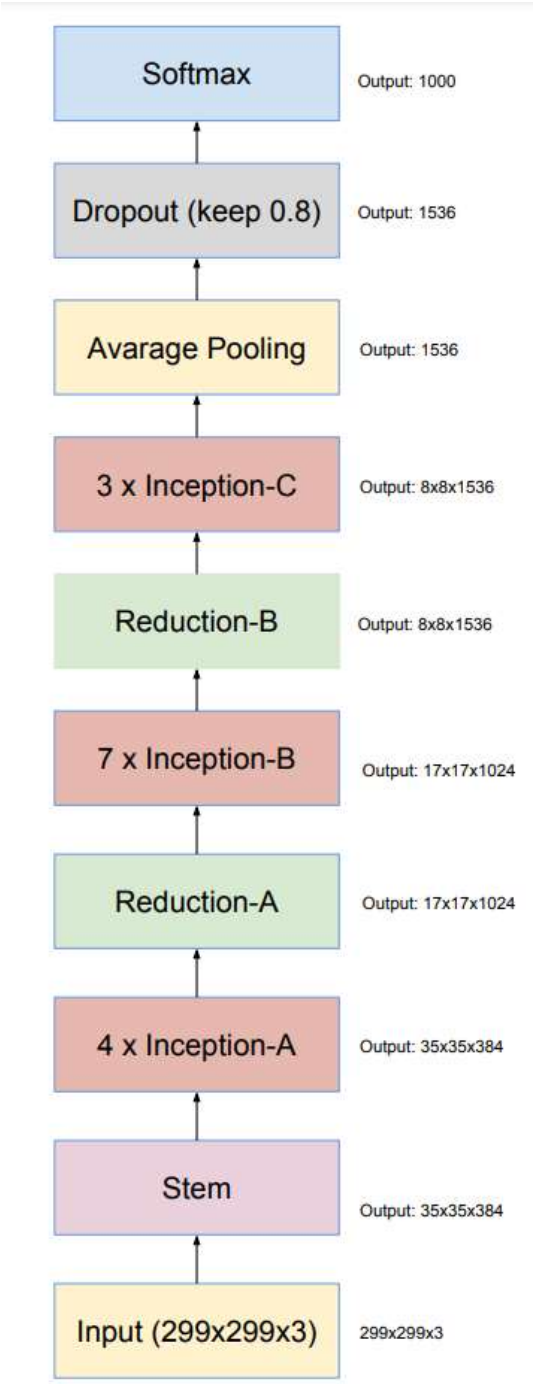
Để tận dụng được sự khai thác theo nhiều filter của Inception, và việc giảm suy giảm đạo hàm (Vanishing Gradients), một khối Residual Inception block ra đời. Kiến trúc tượng trưng của khối như sau:



Hình 3.8 Residual Inception Block

Thay vì là phương thức kết nối tắt (skip connection) trong ResNet, ta sẽ thay thế bằng khối Inception. Như vậy, tại kết nối này, ngoài việc rút ngắn đường đi, giảm mất mát đạo hàm, Inception sẽ là tầng trích xuất đặc trưng tốt khi lan truyền tiến.

Mảng Inception ResNet V1 được ra đời dựa trên khái niệm trên, với kiến trúc:



Hình 3.9 Inception ResNet V1

Chi tiết về các khối và cách thức hoạt động tại đây.

Mạng Inception ResNet V1 được đánh giá hiệu quả, hiệu năng tốt và tối ưu tính toán huấn luyện. Đây sẽ là mô hình tốt cho việc trích xuất đặc trưng khuôn mặt. Lưu ý, mạng sử dụng trong dự án sẽ không có tầng softmax ở cuối.

3.3. SUPPORT VECTOR MACHINE

Support Vector Machine, hay SVM, là bài toán học có giám sát trong Machine Learning. Đây được đánh giá là một phương pháp vô cùng hiệu qủa trong bài toán phân lớp dữ liệu.

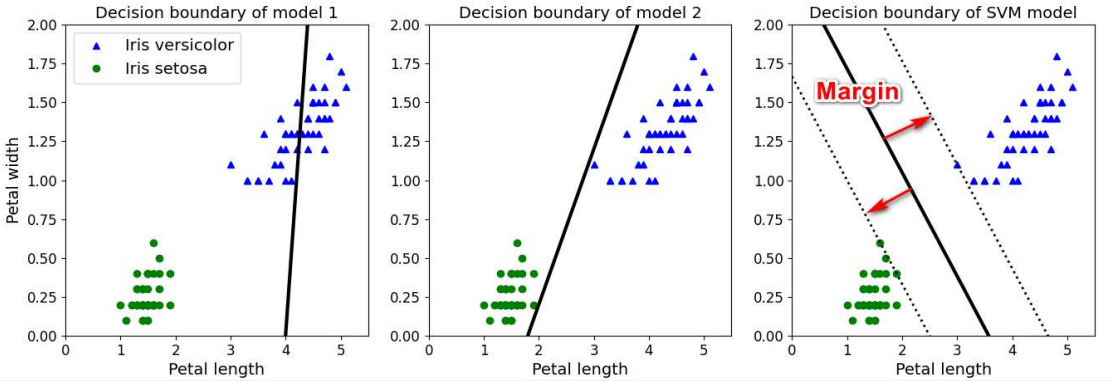
3.3.1. Bài toán

Nếu có hai tập dữ liệu cùng với nhãn cho trước, mục tiêu là xác định một điểm dữ liệu mới sẽ thuộc lớp nào. Kết quả của bài toán là một đường phân cách tuyến tích các giá trị, và sau đó xem dữ liệu mới thuộc tập dữ liệu nào.

3.3.2. Các khái niệm trong SVM

3.3.2.1 Margin

Margin là khoảng cách từ đường phân cách đến điểm gần nhất của tập dữ liệu. SVM cố gắng tìm ra đường phân cách để margin đạt giá trị lớn nhất. Vậy giá trị margin tốt nhất là khi khoảng cách margin của hai lớp đến đường phân cách là bằng nhau.

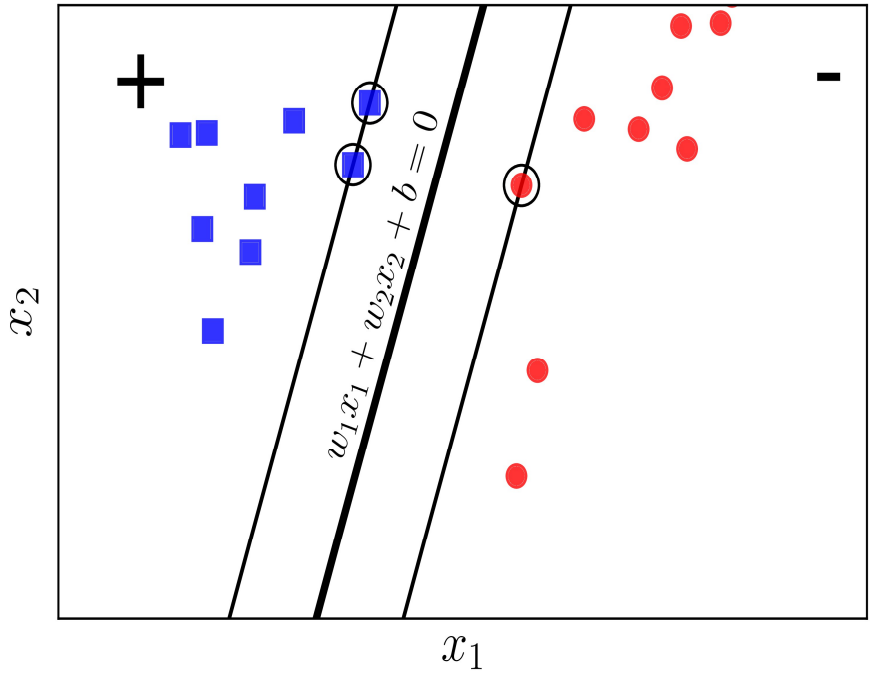


Hình 3.10 Margin trong SVM

3.3.2.2 Support Vector

Các điểm nằm trên đường nét đứt trên hình gọi là support vector, vì chúng hỗ trợ để tìm ra đường phân cách. Tên gọi SVM bắt nguồn từ đây.

3.3.2.3 Khoảng cách từ một điểm đến đường phân cách



Hình 3.11 Ví dụ về SVM

Giả sử rằng các điểm vuông xanh thuộc class 1, các điểm tròn đỏ thuộc class -1 và mặt $w^T x + b = w_1x_1 + w_2x_2 + b$ là mặt phân chia giữa hai classes. Hơn nữa, class 1 nằm về phía dương, class -1 nằm về phía âm của mặt phân chia. Nếu ngược lại, ta chỉ cần đổi dấu của w và b. Bài toán trở thành tìm hệ số của w và b.

Ta quan sát thấy một điểm quan trọng sau đây: với cặp dữ liệu (x_n, y_n) bất kỳ, khoảng cách từ điểm đó tới mặt phân chia là:

$$d = \frac{y_n(w^T x_n + b)}{||w||_2}$$

Điều này có thể dễ nhận thấy vì theo giả sử trên, y_n luôn cùng dấu với x_n. Từ đó suy ra tử số luôn là một số không âm.

Với mặt phân chia như trên, margin được tính từ khoảng cách của một điểm gần nhất tới mặt phẳng đó. Vậy margin được tính như sau:

$$margin = min_n \frac{y_n(w^T x_n + b)}{||w||_2}$$

Bài toán tối ưu lại trở thành tìm w và b sao cho margin đạt giá trị lớn nhất. Như vậy cặp giá trị w và b được tính như sau:

$$(w, b) = argmax_{(w,b)} \left\{ min_n \frac{y_n(w^T x_n + b)}{||w||_2} \right\} = argmax_{(w,b)} \left\{ \frac{1}{||w||_2} min_n y_n(w^T x_n + b) \right\}$$

3.3.2.4 Vấn đề với SVM

- Giá trị margin phụ thuộc vào support vector, vì vậy SVM bị ảnh hưởng nhiều bởi support vector.

Từ đó khái niệm "Soft margin" ra đời: cho phép một số điểm dữ liệu nằm về hai phía của margin, hoặc tệ hơn là nằm thuộc về phía bên kia của lớp khác. Mục đích để đạt được giá trị margin lớn nhất.

- Nếu không thể chia tập dữ liệu thành hai tập phân biệt, SVM sẽ không hoạt động được, do tất cả các dữ liệu phải về cùng một phía.

Sử dụng kỹ thuật polynomial, nhằm tăng chiều dữ liệu, khi đó đường phân cách sẽ là non-linear.

3.3.2.5 Kết luận

Là một kĩ thuật phân lớp khá phổ biến, SVM thể hiện được nhiều ưu điểm trong số đó có việc tính toán hiệu quả trên các tập dữ liệu lớn. Ưu điểm như:

- Xử lý trên không gian số chiều cao: SVM là một công cụ tính toán hiệu quả trong không gian chiều cao, trong đó đặc biệt áp dụng cho các bài toán phân loại văn bản và phân tích quan điểm nơi chiều có thể cực kỳ lớn.
- Tiết kiệm bộ nhớ: Do chỉ có một tập hợp con của các điểm được sử dụng trong quá trình huấn luyện và ra quyết định thực tế cho các điểm dữ liệu mới nên chỉ có những điểm cần thiết mới được lưu trữ trong bộ nhớ khi ra quyết định.

- Tính linh hoạt - phân lớp thường là phi tuyến tính. Khả năng áp dụng Kernel mới cho phép linh động giữa các phương pháp tuyến tính và phi tuyến tính từ đó khiến cho hiệu suất phân loại lớn hơn.

SVM vẫn tồn tại những nhược điểm như:

- Bài toán số chiều cao: Trong trường hợp số lượng thuộc tính (p) của tập dữ liệu lớn hơn rất nhiều so với số lượng dữ liệu (n) thì SVM cho kết quả khá tồi.
- Chưa thể hiện rõ tính xác suất: Việc phân lớp của SVM chỉ là việc cố gắng tách các đối tượng vào hai lớp được phân tách bởi siêu phẳng SVM. Điều này chưa giải thích được xác suất xuất hiện của một thành viên trong một nhóm là như thế nào. Tuy nhiên hiệu quả của việc phân lớp có thể được xác định dựa vào khái niệm margin từ điểm dữ liệu mới đến siêu phẳng phân lớp mà chúng ta đã bàn luận ở trên.

SVM là một phương pháp hiệu quả cho bài toán phân lớp dữ liệu. Nó là một công cụ đắc lực cho các bài toán về xử lý ảnh, phân loại văn bản, phân tích quan điểm. Một yếu tố làm nên hiệu quả của SVM đó là việc sử dụng Kernel function khiến cho các phương pháp chuyển không gian trở nên linh hoạt hơn.

4. CHƯƠNG 4: DỰ ÁN NHẬN DIỆN KHUÔN MẶT

4.1. MÔ TẢ DỰ ÁN

Bài toán: Nhận diện khuôn mặt và đưa ra nhãn dự đoán.

Đầu vào: Ảnh khuôn mặt hoặc thông qua camera.

Đầu ra: Nhãn của ảnh khuôn mặt.

4.2. CÔNG NGHỆ SỬ DỤNG

Các mô hình sử dụng: MTCNN, Inception Net V1, SVM.

Dataset: VGGFace2.

Mô tả dữ liệu: Dataset gồm 3,31 triệu ảnh thuộc 9131 người, trung bình 362,6 ảnh một người. Link dataset.

Python phiên bản 3.7.9.

Framework Deep learning chính: Pytorch v1.8.1.

Giao diện: Tkinter.

Lưu trữ: File .npy và file Pickle .pkl.

Việc sử dụng file lưu thay vì CSDL MySQL nhằm giảm các bước trong việc cài đặt và triển khai hệ thống.

4.3. LUỒNG HOẠT ĐỘNG:

4.3.1. Đăng kí khuôn mặt mới:

- Nhập id khuôn mặt mới.
- Ảnh được lấy sẽ qua mô hình MTCNN, trả ra ảnh mặt kích thước 160 x 160 pixel. Số lượng 50 ảnh.
- Ảnh tiếp tục vào mô hình mạng Inception ResNet V1, giá trị trả ra là vector đặc trưng (512, 1).
- Tạo biến label, và đưa vào SVM phân loại.
- Có thể đăng kí khuôn mặt mới bằng video hoặc tập ảnh.

4.3.2. Dự đoán:

- Đưa mặt vào camera và chọn chức năng “Login”.
- Ảnh được cắt bằng MTCNN và trích xuất đặc trưng bằng Inception ResNet V1.
- Dự đoán bằng SVM đã lưu từ trước.
- Ngưỡng dự đoán: 0.8

4.4. HIỆU SUẤT

4.4.1. Khi đăng ký mặt mới

- Thời gian đăng ký mặt mới: 6.5 giây/ 50 ảnh.
- CPU: 82%
- GPU: 50%

4.4.2. Dự đoán

- Thời gian dự đoán: 0.3 giây
- CPU và GPU không đáng kể
- Tỉ lệ chính xác: khoảng 85%

4.5. HƯỚNG PHÁT TRIỂN

4.5.1. Hoàn thiện ứng dụng

- Thực hiện các tính năng khác của ứng dụng.
- Sử dụng CSDL (nếu cần) trong các hệ thống lớn.
- Hạn chế phát sinh lỗi trong quá trình hoạt động

4.5.2. Phát triển ứng dụng

- Tối ưu phần mềm, tăng tính ổn định và hạn chế lỗi.
- Giảm thời gian xử lý, giảm yêu cầu phần cứng.
- Phát triển hệ thống Client – Server cho nhu cầu nhiều thiết bị cùng sử dụng hệ thống.

KẾT LUẬN

Qua những kiến thức được học trong môn học Học Sâu, như mạng CNN, mạng ANN, các kiến trúc mạng và các phép tính toán trong Học Sâu, nhóm đã nghiên cứu, và phát triển ứng dụng nhận diện khuôn mặt cơ bản. Tuy còn nhiều điểm hạn chế, nhóm sẽ tiếp tục phát triển và hoàn thiện ứng dụng.

TÀI LIỆU THAM KHẢO

https://paperswithcode.com/task/face-verification

https://towardsdatascience.com/face-detection-models-which-to-use-and-why-d263e82c302c

https://arxiv.org/pdf/1801.07698.pdf

https://arxiv.org/pdf/1503.03832v3.pdf

https://medium.com/@iselagradilla94/multi-task-cascaded-convolutional-networks-mtcnn-for-face-detection-and-facial-landmark-alignment-7c21e8007923

https://www.kaggle.com/timesler/fast-mtcnn-detector-55-fps-at-full-resolution

https://github.com/serengil/tensorflow-101/blob/master/model/inception_resnet_v1.py

https://paperswithcode.com/dataset/vggface2-1

https://arxiv.org/pdf/1602.07261.pdf

PHỤ LỤC

NORM (CHUẨN)

Trong không gian một chiều, việc đo khoảng cách giữa hai điểm đã rất quen thuộc: lấy trị tuyệt đối của hiệu giữa hai giá trị đó. Trong không gian hai chiều, tức mặt phẳng, chúng ta thường dùng khoảng cách Euclid để đo khoảng cách giữa hai điểm. Khoảng cách này chính là cái chúng ta thường nói bằng ngôn ngữ thông thường là đường chim bay. Đôi khi, để đi từ một điểm này tới một điểm kia, con người chúng ta không thể đi bằng đường chim bay được mà còn phụ thuộc vào việc đường đi nối giữa hai điểm có dạng như thế nào nữa.

Việc đo khoảng cách giữa hai điểm dữ liệu nhiều chiều, tức hai vector, là rất cần thiết trong Machine Learning. Chúng ta cần đánh giá xem điểm nào là điểm gần nhất của một điểm khác; chúng ta cũng cần đánh giá xem độ chính xác của việc ước lượng; và trong rất nhiều ví dụ khác nữa.

Và đó chính là lý do mà khái niệm norm (chuẩn) ra đời. Có nhiều loại norm khác nhau mà các bạn sẽ thấy ở dưới đây:

Để xác định khoảng cách giữa hai vector y và z, người ta thường áp dụng một hàm số lên vector hiệu x = y - z. Một hàm số được dùng để đo các vector cần có một vài tính chất đặc biệt.

L1 Norm

Giả sử vector $x = [x_1; x_2; \dots; x_n], y = [y_1; y_2; \dots; y_n]$. Ta có L1 Norm

$$||x - y|| = |x_1 - y_1| + |x_2 - y_2| + \cdots + |x_n - y_n|$$

L2 Norm

Phép tính L2 Norm được tính bằng

$$||x - y||_2 = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_n - y_n)^2}$$

BẢNG PHÂN CÔNG CÔNG VIỆC

Công Việc	**Thực Hiện**
Tìm hiểu về nhận diện khuôn mặt	Cả nhóm
Tìm hiểu về thuật toán và các thành phần liên quan	Cả nhóm
Thực hiện triển khai thuật toán cơ bản nhận diện khuôn mặt	Cả nhóm
Tối ưu thuật toán và triển khai thuật toán	Bảo, Duy
Thiết kế và triển khai giao diện	Giang, Thảo
Tìm hiểu và thực hiện phương pháp lưu data	Tiệp, Thư
Thực hiện kiểm thử	Cả nhóm
Thực hiện triển khai hệ thống phát hiện giả mạo khuôn mặt	Bảo