

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
KHOA CÔNG NGHỆ THÔNG TIN  
LỚP CỬ NHÂN TÀI NĂNG**

**NGUYỄN THÀNH AN - NGUYỄN PHÁT TÀI**

**HỆ THỐNG DỊCH VỤ TỔNG HỢP VÀ  
TÌM KIẾM TRÊN VIDEO DỰA VÀO PHÁT HIỆN  
VÀ NHẬN BIẾT MẶT NGƯỜI**

**KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN CNTT**

**TP.HCM, 2017**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
KHOA CÔNG NGHỆ THÔNG TIN  
LỚP CỬ NHÂN TÀI NĂNG**

**NGUYỄN THÀNH AN                   1312016**

**NGUYỄN PHÁT TÀI                   1312504**

**HỆ THỐNG DỊCH VỤ TỔNG HỢP VÀ  
TÌM KIẾM TRÊN VIDEO DỰA VÀO PHÁT HIỆN  
VÀ NHẬN BIẾT MẶT NGƯỜI**

**KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN TIN HỌC**

**GIÁO VIÊN HƯỚNG DẪN  
PGS.TS.TRẦN MINH TRIẾT – THS. NGUYỄN VINH TIỆP**

**NIÊN KHÓA 2013– 2017**

## **NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

## Khóa luận đáp ứng yêu cầu của LV cử nhân tin học.

TpHCM, ngày ..... tháng ..... năm 2017

## Giáo viên hướng dẫn

## **NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN**

## Khóa luận đáp ứng yêu cầu của LV cử nhân tin học.

TpHCM, ngày ..... tháng ..... năm 2017

## Giáo viên phản biện

# LỜI CẢM ƠN

Chúng em xin chân thành cảm ơn Khoa Công Nghệ Thông Tin, trường Đại Học Khoa Học Tự Nhiên, Tp.HCM đã tạo điều kiện tốt cho chúng em thực hiện đề tài này.

Chúng em xin chân thành cảm ơn Thầy Trần Minh Triết, là người đã truyền đạt cho chúng em những kiến thức quý báu trong suốt quá trình học tập và luôn tận tình hướng dẫn, chỉ bảo chúng em trong suốt thời gian thực hiện đề tài. Chúng em xin cảm ơn Thầy Nguyễn Vinh Tiệp đã có những trao đổi, những chỉ dẫn giúp chúng em giải quyết các vấn đề và hoàn thiện đề tài.

Chúng em cũng xin gửi lời cảm ơn sâu sắc đến quý Thầy Cô trong Khoa đã tận tình giảng dạy, trang bị cho chúng em những kiến thức quý báu trong những năm học vừa qua.

Chúng em xin gửi lòng biết ơn sâu sắc đến Mẹ, Ba, các anh chị và bạn bè đã ủng hộ, giúp đỡ và động viên chúng em trong những lúc khó khăn cũng như trong suốt thời gian học tập và nghiên cứu.

Mặc dù chúng em đã cố gắng hoàn thành đề tài trong phạm vi và khả năng cho phép, nhưng chắc chắn sẽ không tránh khỏi những thiếu sót, kính mong sự cảm thông và tận tình chỉ bảo của quý Thầy Cô và các bạn.

Nhóm thực hiện

Nguyễn Thành An & Nguyễn Phát Tài

# ĐỀ CƯƠNG CHI TIẾT

<p><b>Tên Đề Tài:</b> Hệ thống dịch vụ tổng hợp và tìm kiếm trên video dựa vào phát hiện và nhận biết mặt người.</p>
<p><b>Giáo viên hướng dẫn:</b> PGS.TS. Trần Minh Triết – ThS. Nguyễn Vinh Tiệp</p>
<p><b>Thời gian thực hiện:</b> Từ ngày 15/11/2016 đến ngày 15/07/2017</p>
<p><b>Sinh viên thực hiện:</b></p> <p>Nguyễn Thành An (1312016) – Nguyễn Phát Tài (1312504)</p>
<p><b>Loại đề tài:</b> Nghiên cứu lý thuyết, giải pháp kỹ thuật và xây dựng mô hình tương tác thông minh hỗ trợ xem video cho người dùng.</p>
<p><b>Nội Dung Đề Tài</b> (mô tả chi tiết nội dung đề tài, yêu cầu, phương pháp thực hiện, kết quả đạt được, ...):</p> <p>Mục tiêu của đề tài nhằm <i>nghiên cứu để phát triển một API có khả năng nhận diện 500-1000 nhân vật nổi tiếng</i> (nghệ sĩ, chính trị gia, doanh nhân,...). Đồng thời, đề tài trình bày một phương pháp tương tác thông minh mới, trong đó người dùng có thể <i>xem các video và điều hướng theo khuôn mặt của các nhân vật chính</i> xuất hiện trong video đó.</p> <p>Đề tài cũng xây dựng một <i>hệ thống truy vấn các video liên quan đến một nhân vật</i> dựa trên sự xuất hiện khuôn mặt từ đó hỗ trợ người dùng tìm kiếm nhân vật họ yêu thích.</p> <p><b>Nội dung thực hiện chi tiết bao gồm:</b></p> <ul style="list-style-type: none"><li>Nghiên cứu các công trình về phát hiện và nhận biết khuôn mặt được đề xuất và đạt độ chính xác vượt bậc trong năm 2015 – 2016.</li></ul>

- Tìm kiếm và nghiên cứu các tập dữ liệu tiêu biểu được các đề tài sử dụng cho bài toán phát hiện và nhận biết mặt người.
- Nghiên cứu chi tiết lý thuyết, cấu trúc vận hành và cài đặt mô hình SSD300 đề xuất bởi Wei Liu et al năm 2016. Từ đó đề xuất hướng tiếp cận thay đổi cho phù hợp với bài toán phát hiện khuôn mặt.
- Nghiên cứu và phân tích cấu trúc mạng nơ-ron được đề xuất bởi O. M. Parkhi, A. Vedaldi và A. Zisserman năm 2015. Đề xuất giải pháp để đạt độ chính xác cao trên một tập dữ liệu khác.
- Cài đặt lại mô hình SSD300, thay đổi kiến trúc, xây dựng tập dữ mới để huấn luyện cho bài toán phát hiện mặt người.
- Cài đặt lại mô hình VGG16 và huấn luyện với tập dữ liệu FaceScrub (đề xuất bởi H.-W. Ng và S. Winkler năm 2014).
- Xây dựng các webservice chạy trên server để hiện thực hóa đề tài thành Web APIs với hai chức năng phát hiện và nhận diện khuôn mặt.
- Xây dựng ứng dụng phân tích, chỉ mục và tìm kiếm video dựa vào phát hiện và nhận biết khuôn mặt của các nhân vật.
- Thu thập các video mẫu, chạy thực nghiệm và hoàn chỉnh ứng dụng hỗ trợ người dùng xem video trong môi trường tương tác thông minh.

**Kế Hoạch Thực Hiện:**

15/11/2016-31/12/2016: Nghiên cứu các công trình về phát hiện và nhận biết khuôn mặt được đề xuất và đạt độ chính xác vượt bậc trong năm 2015 – 2016.

01/01/2016-15/01/2017: Tìm kiếm và nghiên cứu các tập dữ liệu tiêu biểu được các đề tài sử dụng cho bài toán phát hiện và nhận biết mặt người.

16/01/2017-31/01/2017: Nghiên cứu chi tiết lý thuyết, cấu trúc vận hành và cài đặt mô hình SSD300 đề xuất bởi Wei Liu et al năm 2016. Từ đó đề xuất hướng tiếp cận thay đổi cho phù hợp với bài toán phát hiện khuôn mặt.

01/02/2017-15/02/2017: Nghiên cứu và phân tích cấu trúc mạng nơ-ron được đề xuất bởi O. M. Parkhi, A. Vedaldi và A. Zisserman năm 2015. Đề xuất giải pháp để đạt độ chính xác cao trên một tập dữ liệu khác.

16/02/2017-15/03/2017: Cài đặt lại mô hình SSD300, thay đổi kiến trúc, xây dựng tập dữ mới để huấn luyện cho bài toán phát hiện mặt người.

16/03/2017-15/04/2017: Cài đặt lại mô hình VGG16 và huấn luyện với tập dữ liệu FaceScrub (đề xuất bởi H.-W. Ng và S. Winkler năm 2014).

16/04/2017-15/05/2017: Xây dựng các webservice chạy trên server để hiện thực hóa đề tài thành Web APIs với hai chức năng phát hiện và nhận diện khuôn mặt.

16/05/2017-15/06/2017: Xây dựng ứng dụng phân tích, chỉ mục và tìm kiếm video dựa vào phát hiện và nhận biết khuôn mặt của các nhân vật.

16/06/2017-01/07/2017: Thu thập các video mẫu, chạy thử nghiệm và hoàn chỉnh ứng dụng hỗ trợ người dùng xem video trong môi trường tương tác thông minh.

<b>Xác nhận của GVHD</b>	<b>Ngày 11 tháng 11 năm 2016</b> <b>Nhóm SV Thực hiện</b>
<b>PGS.TS. Trần Minh Triết</b>	<b>Nguyễn Thành An – Nguyễn Phát Tài</b>

# MỤC LỤC

LỜI CÁM ƠN .....	iii
ĐỀ CƯƠNG CHI TIẾT.....	iv
MỤC LỤC.....	vii
DANH MỤC CÁC HÌNH.....	x
DANH MỤC CÁC BẢNG .....	xiii
TÓM TẮT ĐỀ TÀI .....	xiv
Chương 1 Mở đầu .....	1
1.1. Giới thiệu chung.....	1
1.2. Hệ thống tương tác thông minh .....	4
1.3. Lý do thực hiện đề tài .....	5
1.4. Mục tiêu đề tài.....	6
1.5. Nội dung đề tài .....	8
Chương 2 Các công trình và tập dữ liệu liên quan.....	10
2.1. Tổng quan .....	10
2.2. Các công trình tiêu biểu về phát hiện và nhận biết mặt người .....	10
2.2.1. Các công trình phát hiện mặt người.....	10
2.2.2. Các công trình nhận biết mặt người.....	16
2.3. Phát hiện vật thể bằng SSD300.....	21
2.3.1. Cấu trúc SSD300 .....	21
2.3.2. Kết quả thực nghiệm được công bố .....	24
2.4. Nhận biết mặt người bằng DNN – VGG16 .....	26
2.4.1. Cấu trúc VGG16.....	26

2.4.2. Kết quả thực nghiệm được công bố .....	28
2.5. Các tập dữ liệu liên quan.....	30
2.5.1. Khảo sát các tập dữ liệu.....	31
2.5.2. Phân tích tập dữ liệu FaceScrub .....	36
2.6. Kết luận .....	39
Chương 3 Huấn luyện mô hình phát hiện và nhận biết mặt người .....	40
3.1. Mô hình phát hiện mặt người bằng SSD300.....	40
3.1.1. Tinh chỉnh SSD300.....	40
3.1.2. Xây dựng tập dữ liệu .....	41
3.1.3. Huấn luyện và kết quả .....	41
3.2. Mô hình nhận biết mặt người bằng VGG-16 Deep features .....	43
3.2.1. Áp dụng kỹ thuật transfer learning .....	43
3.2.2. Network định danh VGG16-Deep-Feature.....	45
3.2.3. Huấn luyện và kết quả .....	46
3.3. Các công cụ và thư viện sử dụng .....	47
3.4. Kết luận .....	47
Chương 4 Các phân hệ trong hệ thống tương tác thông minh .....	48
4.1. Face Web APIs.....	48
4.1.1. Kiến trúc hệ thống .....	49
4.1.2. Đặc tả APIs .....	50
4.1.3. Demo Website .....	54
4.2. Thông tin tổng hợp từ video số dựa trên phát hiện và nhận biết mặt người .....	57

4.2.1. Dữ liệu đầu ra dạng thô của hệ thống .....	57
4.2.2. Ứng dụng Smart Video Editor – tổng hợp thông tin video .....	58
4.3. Person-based video highlight.....	62
4.3.1. Ngữ cảnh sử dụng .....	62
4.3.2. Giao diện và hệ thống chức năng .....	63
4.4. Person-based filter.....	64
4.4.1. Ngữ cảnh sử dụng .....	64
4.4.2. Thực nghiệm đánh giá .....	65
4.4.3. Giao diện ứng dụng .....	65
4.5. Các công cụ và thư viện sử dụng .....	66
4.6. Kết luận .....	67
Chương 5 Kết luận .....	68
5.1. Các kết quả đạt được .....	68
5.2. Hướng phát triển của đề tài .....	69
Tài liệu tham khảo.....	71
PHỤ LỤC .....	1
Các công trình đã công bố.....	1

## **DANH MỤC CÁC HÌNH**

Hình 1.1. Một số ví dụ về hệ thống tương tác thông minh [5].....	5
Hình 2.1 Hình ảnh minh họa phương pháp [6] .....	11
Hình 2.2. Cấu trúc mạng x-calibration-net, $x = \{12, 24, 48\}$ .....	12
Hình 2.3. Cấu trúc các mạng phân loại .....	13
Hình 2.4. Kết quả chạy trên tập FDDB (trên). Kết quả so sánh với các phương pháp khác trên tập AFW (dưới) .....	13
Hình 2.5 Kết quả minh họa cho [7].....	14
Hình 2.6. Phương pháp [7].....	14
Hình 2.7. Cấu trúc CNN của (1) .....	15
Hình 2.8. Kết quả trên tập FDDB .....	16
Hình 2.9. Kết quả trên tập PASCAL.....	16
Hình 2.10. Kiến trúc DeepID3 net1 và net2 [4]......	17
Hình 2.11. Mô hình cấu trúc (a) và hoạt động (b) của FaceNet [11].....	18
Hình 2.12. Các khuôn mặt ở nhiều độ tuổi được xử lý bởi LF-CNNs [16].....	20
Hình 2.13. Minh họa cho mô hình hoạt động của [18].....	20
Hình 2.14. Cấu trúc SSD300 [21]. .....	22
Hình 2.15. Cơ chế tính toán ở tầng phân loại ( <a href="https://deepsystems.ai">https://deepsystems.ai</a> ).....	22
Hình 2.16. Cách tạo ra các vùng mặc định ( <a href="https://deepsystems.ai">https://deepsystems.ai</a> ) .....	23
Hình 2.17 Kết quả trên tập VOC07 test .....	24
Hình 2.18. Ví dụ minh họa, 7: car, 12: dog, 13:horse, 15: person (Hình 2.17) .....	24
Hình 2.19. Kết quả trên tập VOC12 test .....	24
Hình 2.20. Ví dụ chạy trên tập COCO test-dev với mô hình SSD512 .....	25

Hình 2.21. Kết quả tổng hợp, *: được huấn luyện với phương pháp cải tiến “Data Augmentation for Small Object Accuracy”.	25
Hình 2.22. Một số ảnh mẫu trong tập FaceScrub.....	32
Hình 2.23. Tập ảnh ví dụ cho một người trong SCFace .....	32
Hình 2.24. Vài mẫu trong tập dữ liệu MUCT Landmarked.....	34
Hình 2.25. Các mẫu trong tập Bosphorus .....	35
Hình 2.26. Biểu đồ phân bố dữ liệu trong tập FaceScrub .....	36
Hình 2.27. Ví dụ về điều kiện chiếu sáng tự do trong FaceScrub.....	37
Hình 2.28. Ví dụ về lão hóa trong tập FaceScrub .....	38
Hình 2.29. Ví dụ về sự khác biệt khi dùng các camera quá khác nhau. ....	38
Hình 2.30. Ví dụ về hóa trang và trang điểm trong tập FaceScrub.....	39
Hình 3.1. Giá trị hàm lỗi trong quá trình huấn luyện.....	42
Hình 3.2. Kết quả so sánh với DMP .....	42
Hình 3.3. Kết quả so sánh với Facebook (ảnh phía trên) ngày 02/07/2017 .....	43
Hình 3.4. Vai trò của các lớp trong VGG16 network .....	44
Hình 3.5. Qui trình nhận biết mặt người tổng quát. ....	45
Hình 3.6. Kết quả huấn luyện của Deep Neural Network.....	46
Hình 3.7. Đánh kết quả đánh giá trên từng lớp .....	46
Hình 4.1. Mô hình hoạt động của Face Web APIs.....	49
Hình 4.2. Diễn viên Daniel Radcliffe vai Harry Potter trong series phim cùng tên.	53
Hình 4.3. Cấu trúc Demo Website .....	55
Hình 4.4. Ví dụ về giao diện kết quả của Detection Site (a) và Recognition Site (b)	
	56
Hình 4.5. Một tập tin đầu ra dạng JSON.....	58

Hình 4.6. Ví dụ tập tin đầu ra của Smart Video Editor (khung đỏ là các thuộc tính ứng với từng diễn viên) .....	60
Hình 4.7. Ví dụ tập “info.json” .....	61
Hình 4.8. Giao diện hoạt động của ứng dụng Smart Video Editor. ....	62
Hình 4.9. Màn hình Home sau khi load dữ liệu của Person-based Video Highlight. ....	63
Hình 4.10. Giao diện xem video theo các phân đoạn của diễn viên .....	64
Hình 4.11. Giao diện ứng dụng Person-based Filter. ....	66

## **DANH MỤC CÁC BẢNG**

Bảng 1-1. Các lĩnh vực ứng dụng phát hiện và nhận biết mặt người [1] .....	2
Bảng 2-1. Cấu trúc chi tiết network VGG16 [2].....	26
Bảng 2-2. So sánh kết quả các mô hình bằng LFW unrestricted setting [2].....	29
Bảng 2-3. So sánh kết quả các mô hình bằng Youtube Face unrestricted setting [2]. K là số lượng người dung để nhận biết trong các video. .....	29
Bảng 2-4. Một số tập dữ liệu dùng cho nhận biết mặt người.....	31
Bảng 2-5. Bảng so sánh kích thước tập CASIA-WebFace và một số tập khác [31]	33
Bảng 3-1. Cấu trúc network để xuất để phân lớp VGG16-Deep-Feature .....	45
Bảng 3-2. Các công cụ và thư viện cài đặt và huấn luyện mô hình.....	47
Bảng 4-1. Bảng phân loại Face Web APIs đã phát triển.....	48
Bảng 4-2. Nghi thức hoạt động của Face Web APIs. ....	49
Bảng 4-3. Ưu và khuyết điểm của kiến trúc hệ thống.....	50
Bảng 4-4. Địa chỉ IP của các server cung cấp Face Web APIs.....	51
Bảng 4-5. Kết quả trả về của các APIs dưới dạng JSON.....	51
Bảng 4-6. Ví dụ kết quả phát hiện và nhận biết mặt diễn viên Daniel Radcliffe. ....	53
Bảng 4-7. Cấu trúc thông tin kết quả của một frame hình. ....	57
Bảng 4-8. Cấu trúc tập tin đầu ra dạng JSON của Smart Video Editor .....	59
Bảng 4-9. Cấu trúc tập tin “info.json” .....	60
Bảng 4-10. Kết quả đánh giá hiệu suất truy vấn. ....	65
Bảng 4-11. Các công cụ và thư viện phát triển ứng dụng minh họa.....	66

## TÓM TẮT ĐỀ TÀI

Hiện nay, xem tivi, phim ảnh và các video tin tức là một hình thức giải trí phổ biến trên toàn cầu. Thế nhưng vẫn còn tồn động một số khó khăn cho người xem trong việc theo dõi, nắm bắt nhanh thông tin của video mà họ quan tâm trong điều kiện làm việc công nghiệp hạn hẹp về thời gian. Để giải quyết vấn đề đó, chúng em đã nghiên cứu bài toán phát hiện và nhận diện mặt người – một mảng lớn trong lĩnh vực thị giác máy tính – để mang đến trải nghiệm mới, một hình thức tương tác thông minh cho khán giả qua những chức năng: khái quát nội dung video theo tỷ lệ xuất hiện của các nhân vật quan trọng dựa trên nhận diện khuôn mặt, phân tích và tóm tắt các đoạn theo sự xuất hiện của diễn viên từ đó cho phép người xem truy cập nhanh đến những cảnh mà họ quan tâm và sau cùng là khả năng truy vấn các đoạn, các video liên quan đến từng nhân vật mà khán giả muốn tìm kiếm dựa trên việc nhận diện khuôn mặt.

Hệ thống ứng dụng này được đề xuất dựa trên khả năng phát hiện khuôn mặt trong các frame ảnh của video, đồng thời định danh chính xác một số lượng lớn các nhân vật, diễn viên, chính khách,... nổi tiếng trong thời gian gần đây. Do đó, chúng em đã nghiên cứu những công trình khoa học được công báo gần đây về bài toán phát hiện và nhận diện khuôn mặt để ứng dụng và tinh chỉnh cho phù hợp với mục tiêu và chức năng đề ra.

Nội dung của đề tài này tập trung vào việc *Hệ thống dịch vụ tổng hợp và tìm kiếm trên video dựa vào phát hiện và nhận biết mặt người*. Ngoài việc nghiên cứu và xây dựng hệ thống xem video thông minh này, chúng em còn mở rộng các chức năng phát hiện và nhận diện thành API dạng web để mở rộng khả năng ứng dụng về sau.

Nội dung đề tài bao gồm 5 chương:

**Chương 1:** Mở đầu

**Chương 2:** Các công trình và tập dữ liệu liên quan

**Chương 3:** Huấn luyện mô hình phát hiện và nhận biết mặt người

**Chương 4:** Các phân hệ trong hệ thống tương tác thông minh

## **Chương 5: Kết luận**

## Chương 1

### Mở đầu

*Nội dung Chương 1 trình bày tiềm năng ứng dụng của bài toán phát hiện và nhận biết mặt người, đặc biệt trong lĩnh vực giải trí. Đồng thời, nêu lên những khó khăn khi khán giả muốn nắm bắt nội dung kênh tin tức, phim ảnh mà điều kiện thời gian hạn hẹp, trong đó hướng tới sử dụng môi trường tương tác thông minh như một giải pháp. Chương 1 cũng nêu lên mục tiêu, nội dung và ý nghĩa của đề tài.*

#### 1.1. Giới thiệu chung

Phát hiện và nhận biết mặt người là các bài toán nổi tiếng trong ngành khoa học máy tính. Đây là một chủ đề trong lĩnh vực nhận dạng mẫu (pattern recognition), liên quan mật thiết với thị giác máy tính và xử lý ảnh. Vấn đề chính của phát hiện khuôn mặt là tìm ra tọa độ và kích thước của khuôn mặt trong một bức ảnh hay một frame trong video còn nhận biết mặt người thì tập trung vào phân loại một khuôn mặt mới chưa có trong cơ sở dữ liệu vào một lớp khuôn mặt đã biết hay nói cách khác đó là bài toán định danh khuôn mặt của một người.

Đầu vào của bài toán phát hiện là ảnh tĩnh hoặc một frame của video, nó đã tìm ra tọa độ và kích thước của từng khuôn mặt. Sau đó, các khuôn mặt này lại tiếp tục được dùng làm đầu vào cho bài toán nhận biết để tìm ra định danh của chúng. Vì lẽ đó, hai bài toán này có mối liên hệ chặt chẽ với nhau trong một qui trình.

Trước năm 1990, phát hiện và nhận biết mặt người chưa được ứng dụng rộng rãi vào đời sống vì khói lượng tính toán xử lý khi vận hành các mô hình này rất lớn và sức mạnh của phần cứng máy tính lúc bấy giờ chưa cho phép thực thi trong thời gian thực. Từ sau những năm 1990, với sự phát triển vượt trội của phần cứng phát hiện và nhận biết mặt người đã được đưa vào các hệ thống để phục vụ con người và từ đó đóng một vai trò không nhỏ. Bảng 1-1 thể hiện các lĩnh vực nổi bật nhất ứng dụng phát hiện và nhận biết mặt người.

**Bảng 1-1. Các lĩnh vực ứng dụng phát hiện và nhận biết mặt người [1]**

LĨNH VỰC	ỨNG DỤNG
An ninh (Security)	Kiểm soát ra vào tòa nhà, hệ thống boarding chuyến bay, xác thực email trên multimedia workstation, vào ra văn phòng.
Hệ thống tư pháp hình sự (Criminal Justice System)	Pháp y và phân tích hiện trường.
Điều tra cơ sở dữ liệu hình ảnh (Image Database Investigation)	Chứng minh thư quốc gia, đăng ký phúc lợi, cơ sở dữ liệu giấy phép lái xe cho phép tìm kiếm bằng hình ảnh, benefit recipient.
Giám sát (Surveillance)	Giám sát và truy nã người sử dụng chất gây nghiện, kiểm soát CCTV, giám sát lưới điện, kiểm tra thông tin.
Các ứng dụng thẻ thông minh (Smart Card Applications)	Face prints có thể được lưu trữ trong thẻ thông minh, mã vạch, băng từ và được xác thực bằng cách so sánh ảnh thực với các mẫu trong cơ sở dữ liệu.
Chỉ mục video (Video Indexing)	Đánh nhãn các khuôn mặt trong video.
Ứng dụng cá nhân (Civilian Applications)	Sách điện tử và thương mại điện tử.
Tương tác người-máy (Human Computer Interactions)	Game tương tác và máy tình chủ động (proactive computer).
Môi trường đa phương tiện với tương tác người-máy thích nghi (Multimedia Environment with Adaptive Human Computer Interface)	Một bộ phận của các hệ thống nhận biết ngữ cảnh hoặc phổ biến (ubiquitous), nhận diện khách hàng và gợi ý sản phẩm.

Ngày nay, xem phim ảnh, các video tin tức hay kênh truyền hình là một hình thức giải trí phổ biến trong xã hội. Thế nhưng với nhịp sống công nghiệp và quỹ thời gian vô cùng hạn hẹp thì để theo dõi xuyên suốt những thông tin, diễn viên hay nhân vật mình yêu thích là điều rất khó khăn. Lấy một ví dụ cụ thể, đối với một tập phim, nếu khán giả có thể biết được nhân vật mà họ hâm mộ có xuất hiện thường xuyên không và xuất hiện ở những phân cảnh nào thì sẽ thật tiện lợi để tập trung vào các đoạn này và lướt nhanh hoặc bỏ qua các đoạn khác, từ đó tiết kiệm được rất nhiều thời gian. Với cách thức xem phim ảnh hiện nay thì không giải quyết được vấn đề này. Chính vì thế, để mang đến những trải nghiệm tốt hơn, cần phải xây dựng một hệ thống tương tác thông minh giữa khán giả và các thiết bị trình chiếu (tivi, máy vi tính, smart phone,...), trong đó cần đảm bảo các chức năng sau. Thứ nhất, duy trì khả năng xem phim ảnh và video vốn có. Thứ hai ngoài các thông tin cơ bản đó, cần cung cấp thêm các nội dung quan trọng có liên quan như các nhân vật, diễn viên xuất hiện, tỷ lệ thời gian của từng người cũng như thời điểm phân cảnh mà họ xuất hiện để khán giả có thể chuyển nhanh đến phân cảnh được quan tâm. Thứ ba cho phép tìm kiếm các phân đoạn liên quan đến một nhân vật hay diễn viên được yêu thích trong cơ sở dữ liệu để khán giả có thể xem nhanh các phân đoạn cùng nói về một chủ đề hay con người cụ thể mà không phải xem qua tất cả nội dung video.

Để làm được điều đó thì cần giải quyết thật tốt hai bài toán phát hiện và nhận biết mặt người. Trước đây hai bài toán này được giải quyết chủ yếu dựa trên hướng tiếp cận sử dụng hand-designed features (các đặc trưng được con người tạo ra, ví dụ SIFT, SURF, HAAR, ...). Trong những năm gần đây sự phát triển nhanh chóng của lĩnh vực Deep Learning mang đến một hướng tiếp cận mới và đạt những thành công vượt trội cho phát hiện và nhận biết mặt người đó là feature-learning. Trong đó, Convolutional Neural Network là một công cụ đắc lực và hiệu quả mang lại độ chính xác vượt khả năng của con người trong nhận diện. Một số công trình tiêu biểu có thể kể đến là [2], [3], [4].

## 1.2. Hệ thống tương tác thông minh

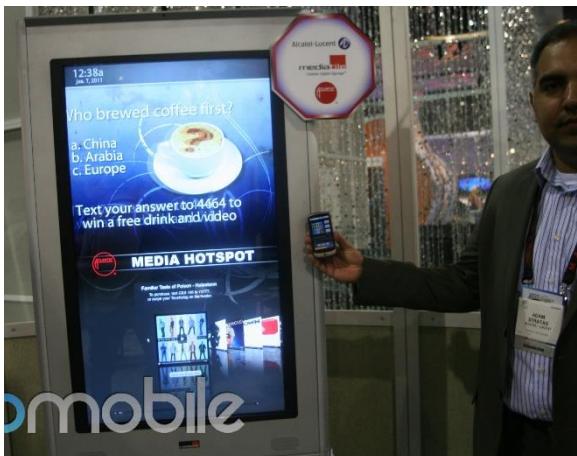
Việc nghiên cứu phát triển những hệ thống và môi trường tương tác thông minh hiện nay đang được chú trọng đầu tư phát triển. Mục đích chính của xu hướng công nghệ này là giúp cho người dùng có thể giao tiếp, điều khiển và tương tác một cách tự nhiên nhất với ứng dụng hay thiết bị đang sử dụng. Ý tưởng này đã xuất hiện từ rất lâu và dễ dàng được bắt gặp trong các phim về khoa học viễn tưởng. Trong đó, những vật dụng bình thường như bàn làm việc, cửa sổ, ... trở thành những công cụ trình chiếu và cho phép người sử dụng tương tác dễ dàng như chạm hay điều khiển bằng giọng nói và những thao tác kéo thả trên mặt kính hay những màn hình lơ lửng giữa không gian trở thành nét đặc trưng của thể loại phim này. Với xu hướng công nghệ hiện tại thì những tính đó đang dần trở thành hiện thực. Với cùng một vật dụng là cửa sổ, Nokia đưa ra ý tưởng cho việc thể hiện nội dung tin nhắn (Hình 1.1a) và Microsoft sử dụng nó để trình chiếu các thông tin kinh doanh, báo cáo doanh số (Hình 1.1b).



(a) Ý tưởng hiển thị tin nhắn trên kính cửa sổ của Nokia



(b) Ý tưởng hiển thị thông tin kinh doanh trên kính cửa sổ của Microsoft



(c) Liên kết giữa điện thoại và kiosk thông minh của Alcatel-Lucent



(d) Bàn cảm ứng Microsoft PixelSense của Microsoft

### Hình 1.1. Một số ví dụ về hệ thống tương tác thông minh [5]

Kiosk thông minh (Hình 1.1c) tích hợp công nghệ ng Connect và 4G LTE của Alcatel-Lucent giới thiệu tại CES 2011 giúp cho một chiếc điện thoại thông minh có thể kết nối và hiển thị thông tin mà người dùng quan tâm trên màn hình của kiosk, đồng thời có thể xem và mua hàng tại kiosk. Microsoft PixelSense (còn gọi là Microsoft Surface) của hãng Microsoft, cho phép đồng thời nhiều người tương tác bằng cách chạm hay đặt các vật thể lên trên màn hình và chia sẻ các nội dung số với nhiều thiết bị di động cùng lúc (Hình 1.1d).

Với ý tưởng thay đổi thói quen và cách thức xem tin tức, phim ảnh của người dùng, trong đề tài này chúng em hướng tới xây dựng một hệ thống tương tác thông minh hỗ trợ chức năng xem video, đồng thời cung cấp thêm các thông tin liên quan đến nội dung bằng cách tổng hợp dựa trên phát hiện và nhận biết mặt người. Hệ thống cho phép người dùng tương tác một cách tự nhiên để xem các thông tin liên quan một cách phù hợp với ngữ cảnh.

### 1.3. Lý do thực hiện đề tài

Các công trình nghiên cứu về phát hiện và nhận biết mặt người hiện nay đã đạt đến độ chính xác rất cao, vượt qua khả năng của con người. Trong xu thế phát triển của các hệ thống và môi trường tương tác thông minh, thì vai trò của các công trình trên

càng nêu cao trong rất nhiều các lĩnh vực liên quan: kinh doanh, giải trí, an ninh,... Với mục tiêu nghiên cứu tìm hiểu cấu trúc và mô hình hoạt động của các thuật toán phát hiện và nhận biết mặt người tiêu biểu hiện nay, nhóm sinh viên tập trung vào phân tích, cài đặt và vận hành các mô hình để hiểu rõ hơn nguyên lý, đồng thời tinh chỉnh cho phù hợp với nhu cầu ứng dụng trong các nội dung liên quan đến đề tài tổng hợp thông tin các video số.

Sau khi hiểu rõ được cấu trúc cài đặt và vận hành của các mô hình phát hiện và nhận biết mặt người, chúng em tập trung vào xây dựng hệ thống ứng dụng, trong đó đưa ra một hướng tiếp cận thông minh cho việc cập nhật tin tức và xem phim ảnh giải trí thông qua video số. Bài toán đặt ra cho hệ thống ứng dụng này là giúp người dùng có thể xem thêm các thông tin liên quan đến nhân vật, diễn viên xuất hiện trong các đoạn video thông qua cách tương tác tự nhiên và hợp ngữ cảnh nhất; bên cạnh đó là giúp họ xem nhanh được những phần nội dung quan trọng, được quan tâm từ đó nắm bắt các chi tiết chính yếu, tiết kiệm thời gian trong thời buổi công nghiệp và bận rộn.

Bên cạnh đó, chúng em cũng xây dựng một hệ thống API cho hai chức năng chính là phát hiện mặt người trong ảnh tĩnh và định danh khuôn mặt của một số lượng lớn các diễn viên, nghệ sĩ nổi tiếng quốc tế.

#### **1.4. Mục tiêu đề tài**

Mục tiêu của đề tài nhằm nghiên cứu để xây dựng hệ thống dịch vụ tổng hợp và tìm kiếm trên video dựa vào phát hiện và nhận biết mặt người. Trong đó, chú trọng phân tích và tinh chỉnh các mô hình có đạt chính xác vượt bậc trong thời gian gần đây, mà kiến trúc chủ yếu dựa trên convolutional neural network nói chung và neural network nói riêng. Các mô hình phát hiện và nhận biết mặt người này được huấn luyện trên các tập dữ liệu tiêu biểu thế nhưng để đưa vào ứng dụng trong một môi trường cụ thể thì đòi hỏi có sự điều chỉnh cho phù hợp. Với mục tiêu nghiên cứu nắm vững lý thuyết kiến trúc và phát triển hệ thống ứng dụng trực quan, nhóm sinh viên thực hiện đề tài mang đến sự đóng góp trong việc hiện thực hóa và sử dụng các mô hình như giải pháp thuật toán.

Trong quá trình nghiên cứu để nắm rõ lý thuyết, nhóm sinh viên đã trình bày lại chi tiết các thông số của từng mô hình cụ thể được sử dụng cho việc phát hiện và nhận biết mặt người trong nội dung Chương 2, khảo sát các công trình và tập dữ liệu liên quan đến đề tài và báo cáo lại như một tư liệu tham khảo cho các công trình về sau. Bên cạnh đó, việc thay đổi, tạo lập tập dữ liệu mới và cách huấn luyện trong đó sử dụng kỹ thuật transfer learning để kế thừa thông tin từ mô hình nguyên bản cũng được mô tả chi tiết trong Chương 3 như một hướng tiếp cận để tham khảo.

Về mặt hiện thực hóa, nhóm sinh viên đã xây dựng một hệ thống Web APIs hoàn chỉnh cho việc phát hiện và nhận biết mặt người, trong đó tập trung vào một lượng lớn các diễn viên và nghệ sĩ nổi tiếng của thế giới. Hệ thống này được cung cấp để phục vụ cho các ứng dụng trong nội bộ đề tài cũng như định hướng cho sự phát triển về sau hoặc hỗ trợ cho các đề tài nghiên cứu, ứng dụng khác. Thông tin chi tiết và các sử dụng được mô tả chi tiết trong phần 4.1 của luận văn.

Bên cạnh đó, đề tài còn xây dựng một hệ thống tương tác thông minh và tìm kiếm video dựa trên lý thuyết phát hiện và nhận biết mặt người đã nghiên cứu (phần 4.2). Hệ thống này cung cấp hai phân hệ: thứ nhất là tóm tắt nội dung video, các phân đoạn của các nhân vật chính từ đó mô tả và chỉ mục chi tiết các phân đoạn để người dùng có thể xem nhanh và nắm bắt cốt truyện dựa trên luồng xuất hiện của nhân vật đó, thứ hai là tìm kiếm các phân đoạn liên quan đến nhân vật được quan tâm. Các phân hệ này mang đến cho người dùng một trải nghiệm thú vị và khả năng tương tác thông minh khi xem video giải trí. Hơn thế nữa là sự tiết kiệm thời gian trong thời đại công nghiệp bận rộn.

### **Nội dung thực hiện chi tiết của khóa luận bao gồm:**

- Nghiên cứu các công trình về phát hiện và nhận biết khuôn mặt được đề xuất và đạt độ chính xác vượt bậc trong năm 2015 – 2016.
- Nghiên cứu, cài đặt lại và tinh chỉnh thuật toán SSD300 được đề xuất bởi Wei Liu et al năm 2016 để phát hiện khuôn mặt.

- Nghiên cứu và phân tích cấu trúc mạng no-ron được đề xuất bởi Karen và Andrew năm 2015 để chỉnh sửa và huấn luyện lại trên tập dữ liệu FaceScrub đề xuất bởi Hong-Wei Ng và Stefan Winkler năm 2014.
- Xây dựng server và hiện thực hóa thành Web API với hai chức năng phát hiện và nhận diện khuôn mặt.
- Nghiên cứu và kỹ thuật tracking sử dụng DeepMatching đề xuất trong công trình của Philippe Weinzaepfel et al năm 2013.
- Xây dựng ứng dụng phân tích, chỉ mục và tìm kiếm video dựa vào khuôn mặt của các nhân vật.
- Thu thập các video mẫu, chạy thử nghiệm và xây dựng ứng dụng hỗ trợ người dùng xem video trong môi trường tương tác thông minh.

## 1.5. Nội dung đề tài

Nội dung đề tài bao gồm 5 chương và nội dung chính từng chương như sau:

### **Chương 1: Mở đầu**

Trình bày tiềm năng ứng dụng của bài toán phát hiện và nhận biết mặt người, đặc biệt trong lĩnh vực giải trí. Đồng thời, nêu lên những khó khăn khi khán giả muốn nắm bắt nội dung kênh tin tức, phim ảnh mà điều kiện thời gian hạn hẹp, trong đó hướng tới sử dụng môi trường tương tác thông minh như một giải pháp. Chương 1 cũng nêu lên mục tiêu, nội dung và ý nghĩa của đề tài.

### **Chương 2: Các công trình và tập dữ liệu liên quan**

Giới thiệu một số công trình phát hiện và nhận biết mặt người tiêu biểu trong các năm gần đây. Trong đó tập trung vào hai mô hình: SSD300 dùng cho phát hiện và VGG16 dùng cho nhận biết, vì đây là hai mô hình chính được kế thừa để thực hiện đề tài. Ngoài ra chương này còn khảo sát một số tập dữ liệu liên quan và phân tích tập dữ liệu FaceScrub – được dùng để thí nghiệm với mô hình nhận biết mặt người của đề tài.

### **Chương 3: Huấn luyện mô hình phát hiện và nhận biết mặt người**

Trình bày các hướng tiếp cận và giải pháp mà nhóm thực hiện để tài triển khai trên mô hình phát hiện và nhận biết mặt người được lựa chọn (SSD300 và VGG16). Các tinh chỉnh đối với mô hình cũ, để xuất cấu trúc mới được trình bày cụ thể theo sau sự phân tích các mô hình này. Bên cạnh đó, quá trình và chiến lược huấn luyện cũng như việc xây dựng mới một số tập dữ liệu phù hợp được mô tả chi tiết để chứng minh tính hiệu quả của hướng tiếp cận và giải pháp mà nhóm đề ra.

### **Chương 4: Các phân hệ trong hệ thống tương tác thông minh**

Trình bày thành phần trong hệ thống tương tác thông minh dựa trên tổng hợp thông tin bằng phát hiện và nhận biết mặt người, bao gồm: các Face Web APIs, Person-based video highlight, Character-based filter. Trong mỗi phần tương ứng, nhóm thực hiện để tài trình bày chi tiết kiến trúc hệ thống, ngữ cảnh sử dụng và các chức năng được cung cấp cũng như hướng dẫn sử dụng và demo tương ứng.

### **Chương 5: Kết luận**

Trình bày các kết quả đạt được và hướng phát triển của đề tài.

## Chương 2

# Các công trình và tập dữ liệu liên quan

*Nội dung Chương 2 giới thiệu một số công trình phát hiện và nhận biết mặt người tiêu biểu trong các năm gần đây. Trong đó tập trung vào hai mô hình: SSD300 dùng cho phát hiện và VGG16 dùng cho nhận biết, vì đây là hai mô hình chính được kế thừa để thực hiện đề tài. Ngoài ra chương này còn khảo sát một số tập dữ liệu liên quan và phân tích tập dữ liệu FaceScrub – được dùng để thí nghiệm với mô hình nhận biết mặt người của đề tài.*

### 2.1. Tổng quan

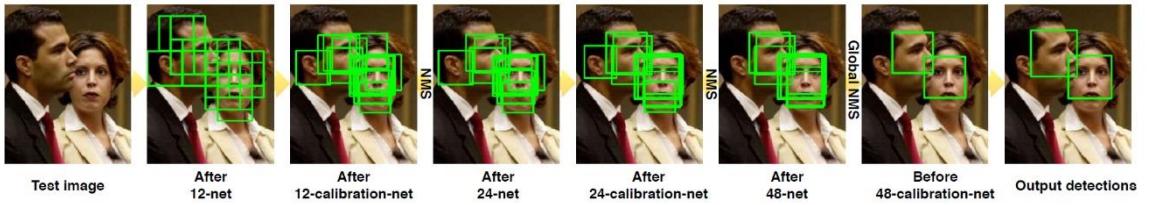
Trong những năm gần đây, lĩnh vực phát hiện và nhận biết mặt người có nhiều bước phát triển vượt bậc, trong đó tiêu biểu là các mô hình đạt độ chính xác cao, vượt qua cả giới hạn của con người. Những thành tựu này có được là sự đóng góp rất lớn từ các đề tài nghiên cứu trong lĩnh vực Deep Learning mà nổi bật là các mạng neural network nói chung và convolutional neural network nói riêng. Bên cạnh đó, sự phát triển của hệ thống thiết bị ghi hình và công nghệ thu thập lưu trữ dữ liệu cũng đóng góp một phần quan trọng trong việc cung cấp các tập mẫu lớn cho việc huấn luyện các mô hình trên. Phản tiếp theo trình bày một số công trình nổi bật trong phát hiện và nhận biết mặt người (do độ dài giới hạn nên luận văn chỉ tập trung giới thiệu một số công trình tiêu biểu trong các năm gần đây).

### 2.2. Các công trình tiêu biểu về phát hiện và nhận biết mặt người

#### 2.2.1. Các công trình phát hiện mặt người

Phát hiện mặt người là một bài toán tiêu biểu trong lĩnh vực thị giác máy tính. Các phương pháp hiện nay đều dễ dàng chỉ ra được các khuôn mặt chính diện trong một bức ảnh. Các nghiên cứu hiện nay tập trung vào giải quyết các trường hợp phức tạp hơn ví dụ như góc mặt thay đổi, cảm xúc, điều kiện chiếu sáng có thể tạo ra sự đa

dạng về hình dạng khuôn mặt, [6]. Năm 2015, Haoxiang Li et al. [6] đề xuất phương pháp phát hiện mặt người dựa trên mạng nơ-ron, có kết quả đáng chú ý trên hai tập thí nghiệm AFW và FDDB về độ chính xác và đặc biệt là tốc độ xử lý vượt các phương pháp hiện thời. Cấu trúc mạng [6] áp dụng gồm 6 bước là: 12-net, 12-calibration-net, 24-net, 24-calibration-net, 48-net, 48-calibration-net,

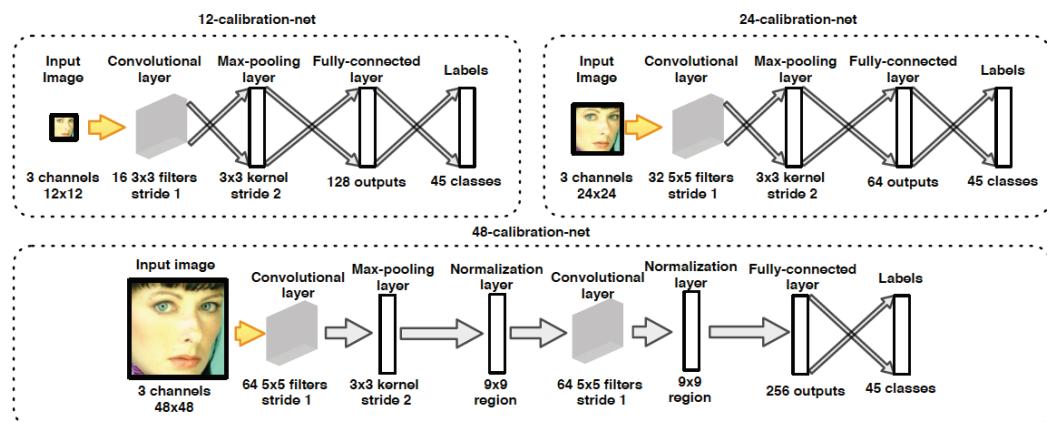


**Hình 2.1 Hình ảnh minh họa phương pháp [6]**

Cụ thể, 12-net là một mạng phân loại nhị phân đơn giản để có thể loại các trường hợp không phải khuôn mặt một cách nhanh chóng. Với ảnh kích thước  $W \times H$  và khoảng cách giữa hai cửa sổ trượt là 4 pixel cho cửa sổ kích thước  $12 \times 12$  thì số lượng cửa sổ cần phân loại là  $\left(\frac{W-12}{4} + 1\right) \times \left(\frac{H-12}{4} + 1\right)$ . Sau khi phân loại, các cửa sổ chưa khuôn mặt được tính toán ở 12-calibration-net. Mạng này có cấu trúc cạn (Hình 2.2), đầu vào được điều chỉnh với là  $N$  mẫu đại diện bằng  $\{(x_i, y_j, s_k)\}^{i,j,k=1:N}$ . Với cửa sổ  $(x, y, w, h)$  (góc trên trái  $(x, y)$  và kích thước  $(w, h)$ ) tập các cửa sổ điều chỉnh được xác định bởi:

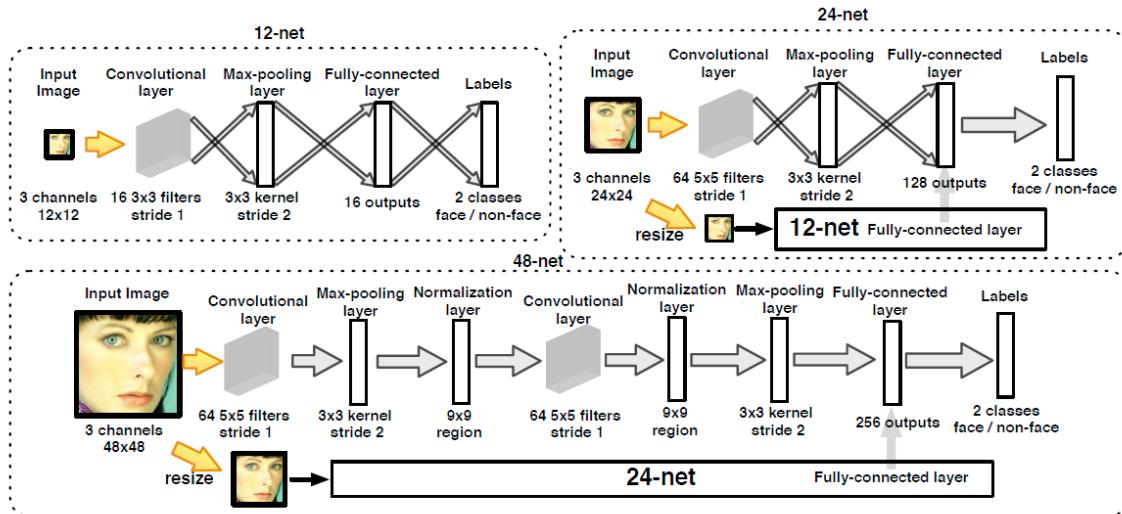
$$\left\{ \left( x - \frac{x_i w}{s_k}, y - \frac{y_j h}{s_k}, \frac{w}{s_k}, \frac{h}{s_k} \right) \right\}^{i,j,k=1:N}$$

Kết quả của bước trên là  $[c_1, c_2, \dots, c_N]$  đặc trưng cho khả năng là mặt người của từng cửa sổ mẫu.



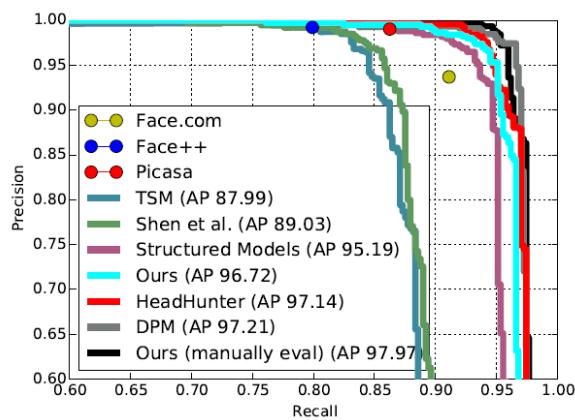
**Hình 2.2. Cấu trúc mạng x-calibration-net, x = {12, 24, 48}**

Non maximum suppression (NMS) được dùng để loại trừ các cửa sổ bị chồng lên nhau. Những cửa sổ còn lại được tách ra và thay đổi kích thước thành 24x24 để đưa vào 24-net. Qua 24-net sẽ giảm bớt 90% không gian tìm kiếm, như ở 12-net, thì các cửa sổ còn lại sẽ được điều chỉnh bởi 24-calibration-net và lại áp dụng NMS. Cuối cùng là 48-net sử dụng các cửa sổ còn lại như một ảnh 48x48 để ước lượng khả năng chứa khuôn mặt tương tự như 24-net. Kết quả cuối cùng được điều chỉnh bởi 48-calibration-net. Nói về NMS, thuật toán này lựa chọn các cửa sổ có khả năng cao và dựa vào các tỉ số trùng lặp giữa các cửa sổ mà loại bỏ cửa sổ có khả năng thấp hơn, tỉ số trùng lặp được so sánh với một ngưỡng gán trước. Các mạng phân loại có cấu trúc mạng nơ-ron xoắn (CNN) được huấn luyện để phân loại nhiều lớp (Hình 2.3).



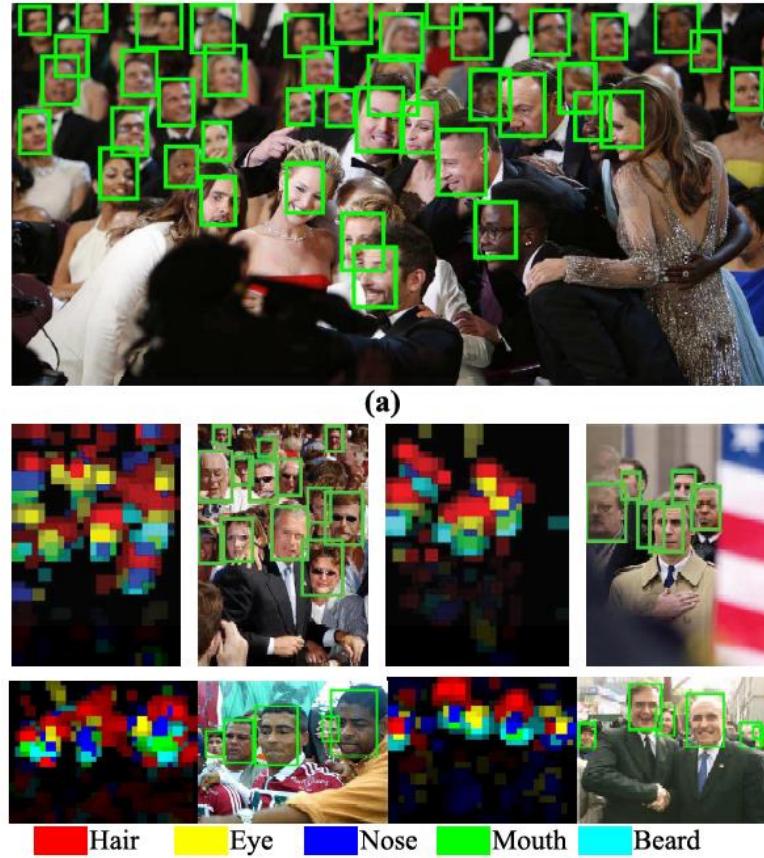
**Hình 2.3. Cấu trúc các mạng phân loại**

Tuy trên cả hai tập AFW và FDDB, về độ chính xác, [6] có thể so sánh được với phần còn lại tuy nhiên về tốc độ xử lý thì [6] thể hiện là phương pháp tốt nhất lúc bấy giờ với kết quả chạy trên GPU là 10ms cho một ảnh.

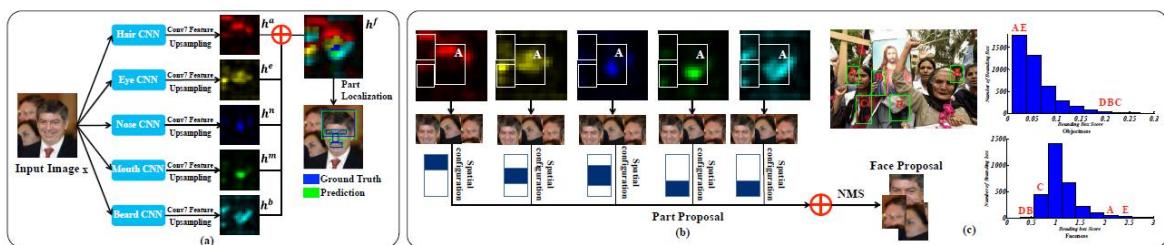


**Hình 2.4. Kết quả chạy trên tập FDDB (trên). Kết quả so sánh với các phương pháp khác trên tập AFW (dưới)**

Cùng năm 2015, [7] đề xuất phương pháp dựa vào các đặc trưng trên khuôn mặt như mắt, mũi, tóc ... và vị trí tương đối của chúng để xây dựng nên một mô hình Deep Learning giải quyết bài toán phát hiện mặt người. Phương pháp này đạt hiệu quả tốt hơn hẳn so với các công trình trước đó.

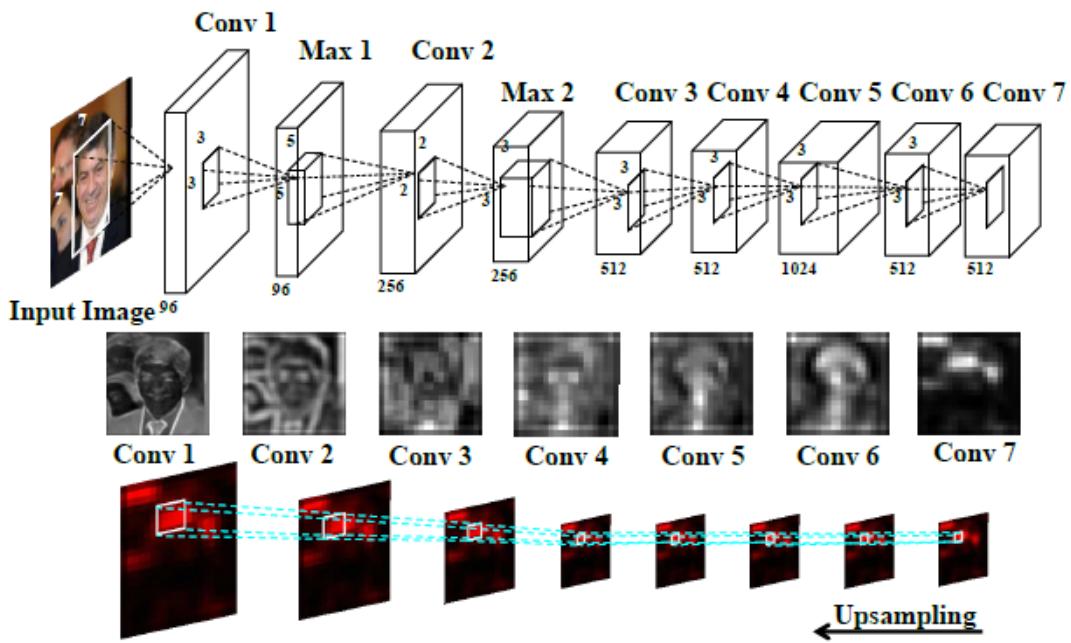


Hình 2.5 Kết quả minh họa cho [7]



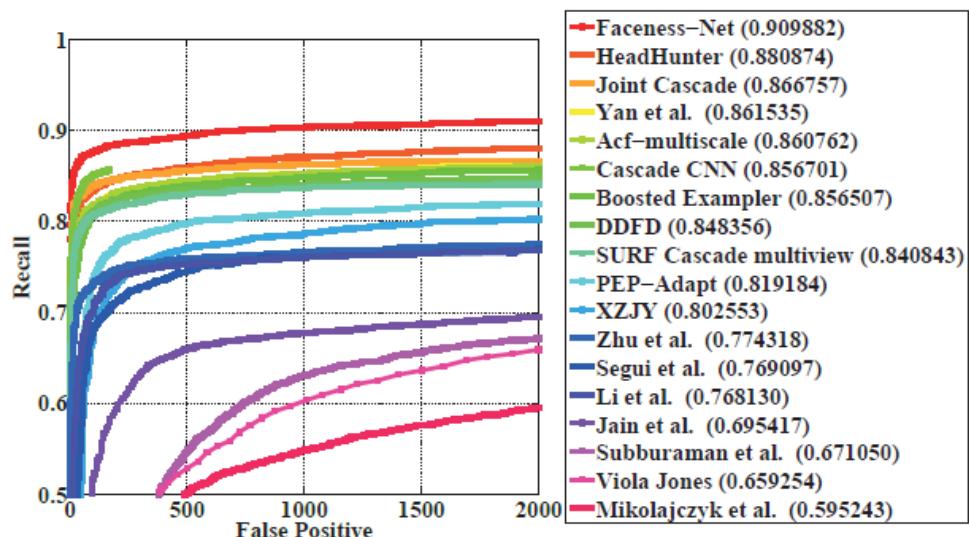
Hình 2.6. Phương pháp [7]

[7] xây dựng một mạng gọi là Faceness-net, gồm ba phần là: Rút trích ma bộ phân (Partness Maps Extraction - 1); Xếp hạng các khả năng dựa vào các bộ phận (Ranking Windows by faceness measure - 2); và Phát hiện khuôn mặt (Face detection – 3). Ảnh x đi vào (1) sẽ được xử lý bởi năm CNN, (Hình 2.6). Các mạng này có cùng cấu trúc dựa trên ý tưởng của AlexNet. (Hình 2.7) và sử dụng chung bộ nhớ để tăng hiệu năng tính toán.

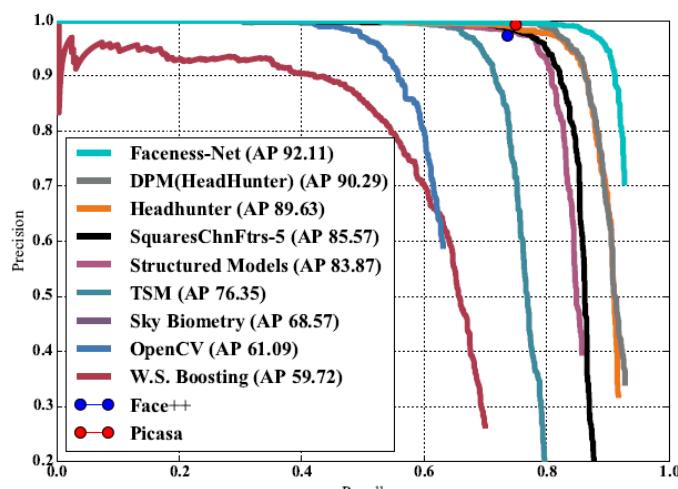


**Hình 2.7. Cấu trúc CNN của (1)**

Kết quả của mỗi CNN này thể hiện vùng chứa bộ phận tương ứng, Sau đó các kết quả này sẽ được tổng hợp lại thành một ma trận duy nhất thể hiện vùng chứa khuôn mặt, làm đầu vào cho các bước sau. Ké tiếp, dùng các phương pháp để xuất vùng vật thể tồn tại. tạo ra tập các cửa sổ. Xếp hạng khả năng là mặt người của chung dựa vào các bộ phận trên khuôn mặt được rút trích từ bước đầu tiên, kết quả hiển thị ở phần dưới Hình 2.6 – b. Dựa vào kết quả xếp hạng chọn ra các cửa sổ chứa khuôn mặt nằm trong top 50 và có khả năng lớn hơn một ngưỡng đặt trước. Kết quả thí nghiệm cho thấy, Faceness là phương pháp tốt nhất lúc bấy giờ về độ chính xác.



Hình 2.8. Kết quả trên tập FDDB

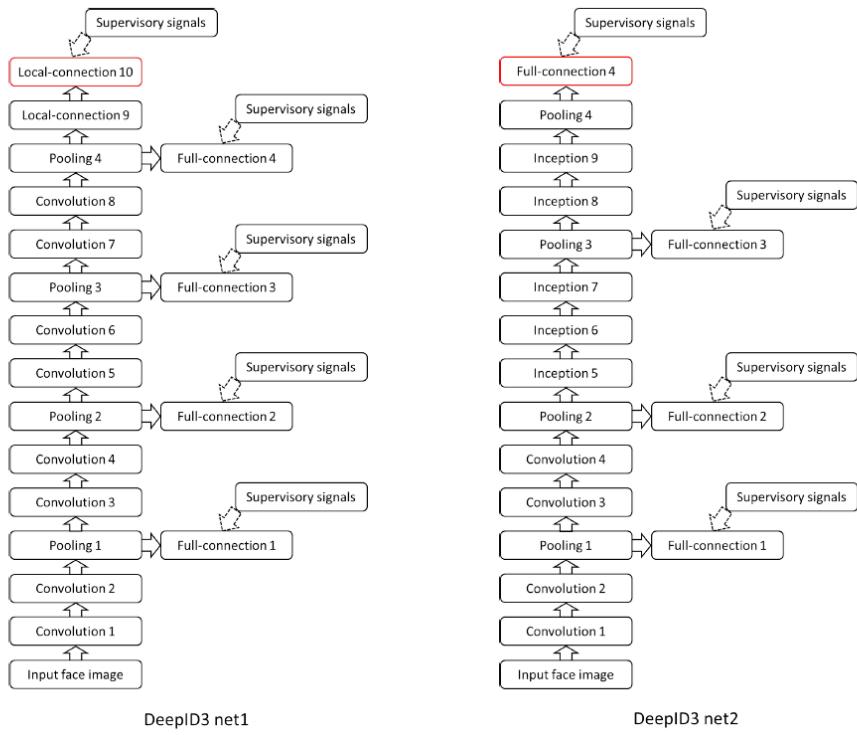


Hình 2.9. Kết quả trên tập PASCAL

### 2.2.2. Các công trình nhận biết mặt người

Đầu tiên phải kể đến công trình [4] với kiến trúc network được gọi là DeepID3. Với mục đích khảo sát sự hiệu quả của các neural network sâu (very deep neural network) trong việc nhận biết mặt người, Yi Sun et al đã tạo ra hai kiến trúc mới bằng cách xây dựng lại các lớp convolution và inception kết chồng lên nhau được đề xuất trong VGG net [8] và GoogLeNet [9]. Điểm đặc biệt trong mô hình này là các tín hiệu kết hợp nhận biết và xác minh khuôn mặt có giám sát (joint face identification-verification

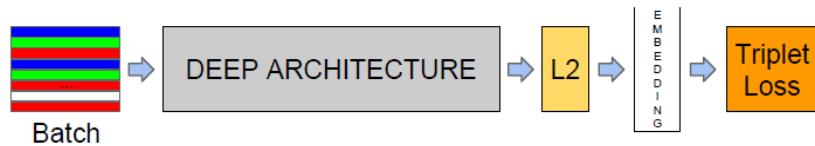
supervisory signal) được thêm vào ngay sau các lớp rút trích đặc trưng cuối cùng trong suốt quá trình huấn luyện mô hình. Hình 2.10 minh họa hai cấu trúc DeepID3 net1 và net2, trong đó mũi tên liền nét thể hiện hướng forward-propagation, các mũi tên nghiêng chỉ ra các lớp mà tại đó tín hiệu nhận biết và xác nhận khuôn mặt có giám sát được thêm vào, lớp rút trích đặc trưng cuối cùng trong khung màu đỏ phục vụ cho việc nhận biết khuôn mặt. Với sự cải tiến này DeepID3 đạt độ chính xác cao trên tập dữ liệu LFW [10] với 99.53% cho xác nhận khuôn mặt và 96% cho rank-1 face recognition.



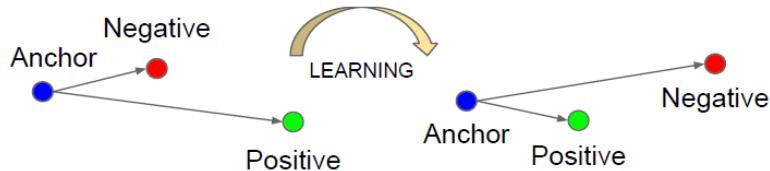
**Hình 2.10. Kiến trúc DeepID3 net1 và net2 [4].**

Nhóm tác giả [11] đưa ra một hệ thống gọi là FaceNet để học một ánh xạ từ các ảnh khuôn mặt vào một không gian Euclidean chặt chẽ mà ở đó khoảng cách tượng trưng trực tiếp cho sự tương đồng giữa các khuôn mặt. Minh họa cho cấu trúc và hoạt động của mô hình được thể hiện trong Hình 2.11. Với hướng tiếp cận này, các bài toán về nhận biết, xác minh và gom nhóm có thể được cài đặt dễ dàng với FaceNet embeddings (như là các vector đặc trưng). Phương pháp này sử dụng deep convolutional neural network được huấn luyện rồi để tự tối ưu ánh xạ hơn là tầng thắt

cỗ chai trung gian (intermediate bottleneck layer) như trong các hướng tiếp cận trước đây. Điểm nổi bật của mô hình là tính hiệu quả lớn trong việc thể hiện khuôn mặt: nhóm tác giả đạt hiệu suất vượt trội mà chỉ sử dụng 128 bytes cho một khuôn mặt. Kết quả thực nghiệm trên tập LFW [10] là 99.63% và YTF [12] là 95.12%, trong đó giảm đi 30% tỷ lệ lỗi trên cả hai tập dữ liệu so với kết quả tốt nhất được công bố trong [8].



#### - Mô hình cấu trúc của FaceNet



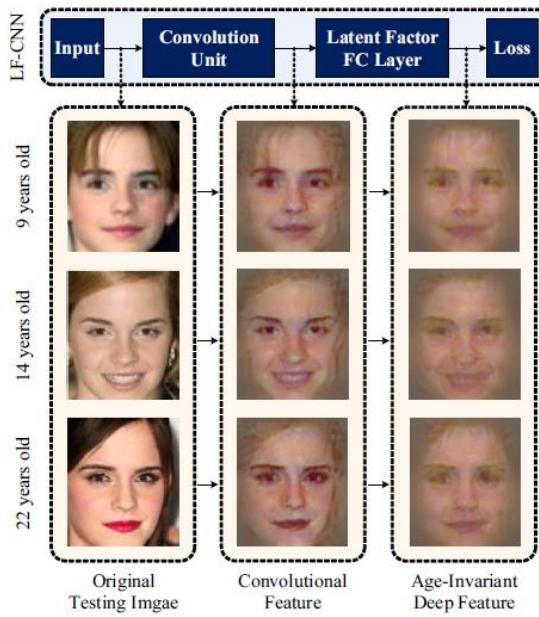
#### - Minh họa hoạt động của FaceNet

**Hình 2.11. Mô hình cấu trúc (a) và hoạt động (b) của FaceNet [11]**

Yi Sun et al đề xuất một hướng tiếp cận mới cho việc huấn luyện dữ liệu trong công trình [13]. Kiến trúc được sử dụng trong công trình này là convolutional neural network kế thừa từ baseline high-performance VGG-like deep neural network [8]. Điểm đặc trưng của mô hình mạng này là dữ liệu được huấn luyện theo chu kỳ. Mỗi lần một lớp bổ sung sẽ được sparsify và toàn bộ mô hình sẽ được huấn luyện lại với các tham số đã có từ chu kỳ trước. Độ chính xác của mô hình gốc đạt được trên tập LFW [10] là 98.95%. Với kiến trúc mới, Sparsifying Neural Network tăng độ chính xác trên cùng tập dữ liệu lên 99.30% và giảm error rate 33% trong khi chỉ giữ lại 12% tham số từ mô hình gốc. Điều này tăng tính khả dụng trên các thiết bị cấu hình thấp như mobile.

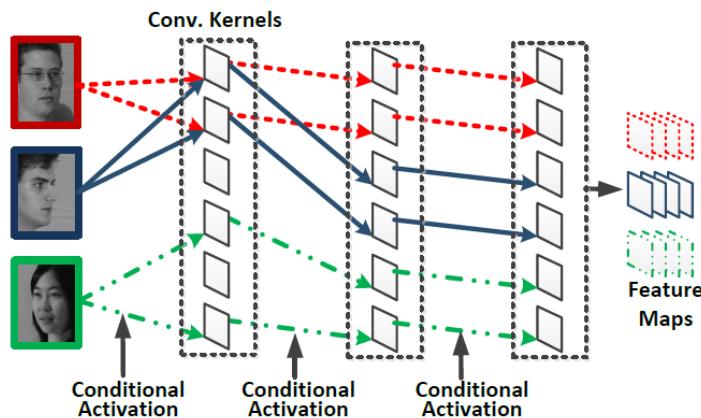
Trong [14], Iacopo Masi et al sử dụng deep convolutional neural networks để giải quyết vấn đề sự sai khác tư thế của khuôn mặt bằng cách sử dụng các mô hình xác định đa tư thế (multiple pose-specific models) và các ảnh khuôn mặt được phát sinh (render). Ý tưởng chính được sử dụng là huấn luyện theo các mô hình riêng ứng với từng tư thế của khuôn mặt, sau đó tổng hợp (fusion) các kết quả lại. Mô hình này được sử dụng cho cả xác nhận và định danh khuôn mặt. Trên tập [15], thực nghiệm đạt độ chính xác  $0.895 \pm 0.006$  (FAR = 0.01) cho xác nhận khuôn mặt và  $0.862 \pm 0.0009$  (Rank-1) cho nhận biết mặt người.

Hầu hết các phương pháp nhận diện đều dựa trên những đặc trưng cố định tại một độ tuổi của khuôn mặt dù là dùng hand-designed feature hay feature-learning. Các vấn đề về lão hóa và biến đổi khuôn mặt là một thách thức lớn. [16] là một công trình sử dụng neural network để giải quyết bài toán này. Trong đó, nhóm tác giả phát triển một latent variable model gọi là latent identity analysis (LIA) kết hợp với CNN để tìm ra các đặc trưng bất biến theo quá trình lão hóa bằng cách huấn luyện theo cặp các tham số của CNNs và LIA. Hình 2.12 minh họa một hoạt động ví dụ được xử lý qua các giai đoạn trong network. Giải pháp này đạt độ chính xác đáng ghi nhận: 97.51% trên tập MORPH Album2 [17] và 99.50% trên tập LFW [10].



**Hình 2.12. Các khuôn mặt ở nhiều độ tuổi được xử lý bởi LF-CNNs [16].**

Với cùng tư tưởng sử dụng convolutional neural network nhưng mỗi nhóm tác giả lại có cách cải biến và thiết kế cho phù hợp với từng bài toán cụ thể. Một ví dụ khác là [18]. Để xử lý cho các vấn đề của ảnh khuôn mặt như tư thế khuôn mặt, điều kiện chiếu sáng và che khuất, các tác giả đề xuất một cấu trúc convolutional neural network, trong đó sử dụng nhiều bộ kernel khác nhau và sẽ được kích hoạt tùy vào những điều kiện nhất định phụ thuộc vào ảnh đầu vào. Nói cách khác, các mẫu được xử lý bằng các kernel được kích hoạt động tùy thuộc vào dữ liệu. Tập hợp các kernel được kích hoạt xuyên suốt các lớp định hướng luồng tính toán theo từng mẫu. Mô hình hoạt động của [18] được mô tả trong Hình 2.13. Phương pháp này đạt độ chính xác 73.54% trên tập Multi-PIE [19].



**Hình 2.13. Minh họat cho mô hình hoạt động của [18].**

Với mục đích tạo ra các đặc trưng bất biến với các phép biến đổi phức mà có thể mô hình hóa cục bộ thành đơn nhất phục vụ cho bài toán nhận biết mặt người và ước tính tư thế khuôn mặt, nhóm tác giả [20] đề xuất một phương pháp gọi là “bells and whistles free”. Bằng cách sử dụng một phương pháp đơn giản hoạt động trên các điểm ảnh thô, [20] đạt kết quả vượt trội trên Multi-PIE database protocol [19] (75.75%), LFW [10] unsupervised protocol (91.54%) và LFW [10] image-restricted, label-free outside data protocol (88.67%). Trong đó, đề tài mang đến ba đóng góp quan trọng nhất. Thứ nhất đề xuất một hướng tiếp cận đơn giản để học các đặc trưng phi tuyến

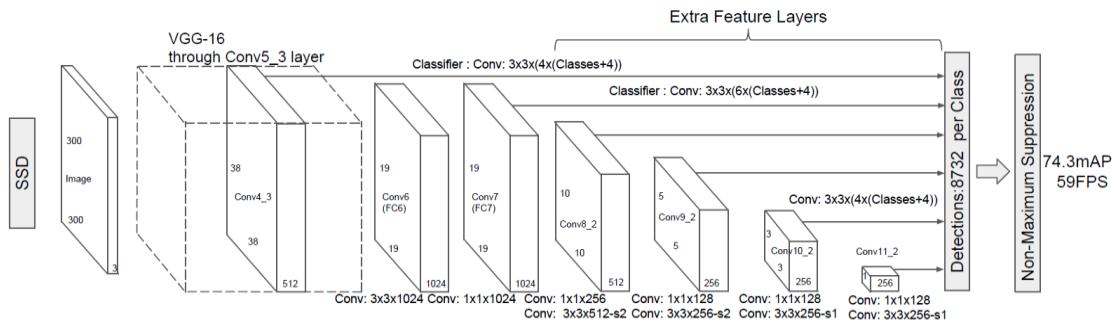
phân biệt bất biến với các phép biến đổi đơn nhất, mở rộng phạm vi lý thuyết gần đây về bất biến với đặc trưng phân biệt và kernelized. Thứ hai là đưa ra một hướng tiếp cận đơn giản dense-landmark-free tạo ra một framework có khả năng nhận biết mặt người open-set pose-invariant và ước tính tư thế khuôn mặt đồng thời. Thứ ba là đề xuất một hướng tiếp cận nối tiếp để tạo ra bất biến các đa biến đổi nội nhóm (multi sub-groups transformations) từ đó có được một framework landmark-free hoàn chỉnh cho nhận biết khuôn mặt và ước tính tư thế bất biến với các phép biến đổi.

### 2.3. Phát hiện vật thể bằng SSD300

Hiện nay, các hệ thống phát hiện vật thể đều tiếp cận theo hướng chọn ra các bao chữ nhật; tính toán, rút trích đặc trưng rồi phân loại các bao đó để cho ra kết quả cuối cùng. Các phương pháp này đều dựa trên kiến trúc mạng Fast R-CNN và hiện là hướng tiếp cận tối ưu dựa vào kết quả chạy thí nghiệm trên các tập dữ liệu nổi tiếng trong lĩnh vực này như PASCAL VOC, COCO, ILSVRC. Mặc dù có độ chính xác cao nhưng các hệ thống này chưa thể cho ra kết quả theo thời gian thực. Ngay cả Fast R-CNN nhanh nhất cũng chỉ đạt ở ngưỡng 7 fps. Một số khác có thời gian chạy tốt, đổi lại độ chính xác sẽ giảm đáng kể. SSD (Single Shot multibox Detector), được đề xuất vào tháng 12 năm 2015, để giải quyết bài toán đề độ chính xác và tốc độ cho bài toán phát hiện vật thể. Cụ thể bằng thí nghiệm, kết quả cho thấy có sự cải tiến vượt bậc về tốc độ nhưng vẫn đảm bảo độ chính xác cao (59 fps với mAP 74.3% trên tập giám định VOC2007, so với Faster R-CNN 7 fps với mAP 73.2% hay YOLO 45 fps với mAP 63.4%) [21].

#### 2.3.1. Cấu trúc SSD300

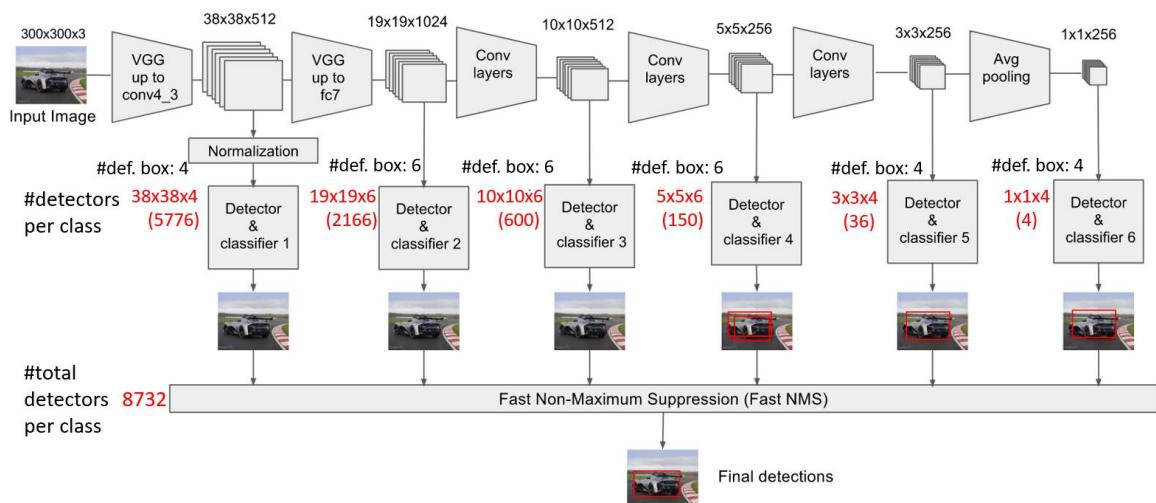
SSD được xây dựng dựa trên mô hình feed-forward convolutional network. Có cấu trúc gồm tầng đầu là mạng cơ sở, là toàn bộ mạng VGG-16 từ đầu đến lớp “fc7 – fully-connected” (trước lớp phân loại). Sau khi hình ảnh đi qua mạng cơ sở, đầu ra được phân loại bằng tầng thứ hai, trả về tập các vùng chữ nhật và trọng số cho từng vật thể trong vùng đó. Sau đó tập này được xử lý cho ra kết quả tìm thấy cuối cùng bằng thuật toán “non-maximum suppression” [21] (Xem Hình 2.14, Hình 2.15).



**Hình 2.14. Cấu trúc SSD300 [21].**

Các tính chất của cấu trúc mạng SSD [21]:

- Dự đoán theo vùng (Convolutional predictors for detection): Mỗi một bộ phân loại không chỉ tính toán dựa trên một ô riêng biệt mà dữa trên kết quả tổng hợp với các ô xung quanh sử dụng bộ lọc xoắn (convolutional filter). Vì thế mỗi lớp đặc trưng tạo số lượng cố định bộ phân loại. Với lớp có kích thước  $m \times n \times p$  sau khi đi qua bộ lọc  $3 \times 3 \times p$  sẽ cho  $m \times n$  bộ phân loại.
- Bản đặc trưng đa tỉ lệ (Multi-scale feature maps for detection): Bộ phân loại được thêm vào dần dần từ lớp đặc trưng cục bộ “Conv5\_5” đến “Conv11\_2” (Hình 2.14), với kích thước của các lớp giảm dần. Điều này làm cho bản đặc trưng đa dạng về mối quan giữa các vùng lân cận trong ảnh.



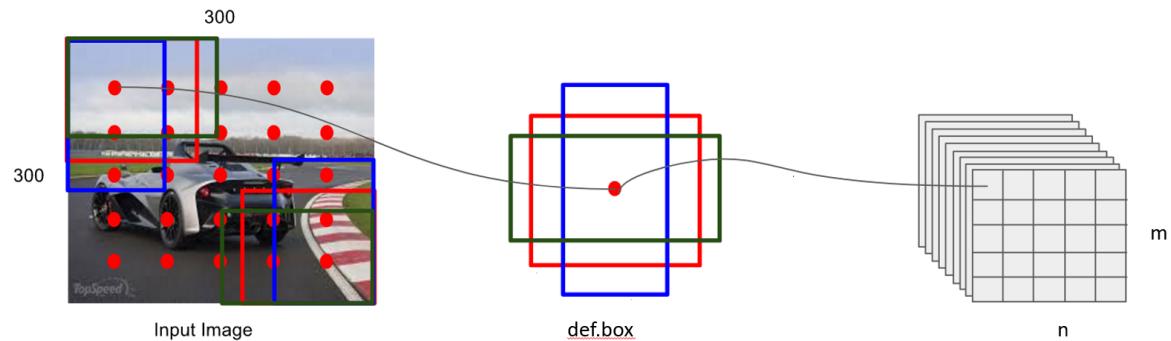
**Hình 2.15. Cơ chế tính toán ở tầng phân loại (<https://deepsystems.ai>)**

- Vùng mặc định (default box - **def. box**) và tỉ lệ (aspect ratios - **asp. ratio**): Với bản đặt trưng, có kích thước  $m \times n$ , được chia thành  $m \times n$  ô, mỗi ô chứa  $\#def. box$  vùng mặc định có kích thước chuẩn là **size** (Hình 2.16). Ta được:

- $def. box = \left\{ (size, size), \left( \frac{size}{\sqrt{asp.ratio}}, \frac{size}{\sqrt{asp.ratio}} \right), \dots \right\}$
- $\#total_{def.box} = m \times n \times \#def. box$

Mỗi vùng mặc định ứng với một bộ phân loại, cho ra kết quả dự đoán bao gồm hình dạng của vùng bao vật thể cùng xác suất mà vật thể trong vùng này có thể là một trong các loại khảo sát (gồm  $\#class$  vật thể) có kích thước  $\#def. box \times (\#class + 4)$ .

- Ví dụ, “classifier 4” có  $5 \times 5 \times 6 = 150$  bộ phân loại và kích thước là  $150 \times (\#class + 4)$  (Hình 2.15).



**Hình 2.16. Cách tạo ra các vùng mặc định (<https://deepsystems.ai>)**

Sau khi tổng hợp được 8732 bộ phân loại cho mỗi loại, kết quả này được tính toán bằng thuật toán “non-maximum suppression” để đưa ra kết quả cuối cùng là tập các vùng chưa vật thể đã được gán nhãn.

### 2.3.2. Kết quả thực nghiệm được công bố

Mô hình SSD gồm hai phiên bản về kích thước ảnh đầu vào là SSD300 (300x300) và SSD512 (512x512). Qua thí nghiệm trên tập VOC07 test cho thấy SSD đạt kết quả tốt hơn Fast [22] và Faster [23] và mọi vật thể.

Method	data	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Fast [6]	07	66.9	74.5	78.3	69.2	53.2	36.6	77.3	78.2	82.0	40.7	72.7	67.9	79.6	79.2	73.0	69.0	30.1	65.4	70.2	75.8	65.8
Fast [6]	07+12	70.0	77.0	78.1	69.3	59.4	38.3	81.6	78.6	86.7	42.8	78.8	68.9	84.7	82.0	76.6	69.9	31.8	70.1	74.8	80.4	70.4
Faster [2]	07	69.9	70.0	80.6	70.1	57.3	49.9	78.2	80.4	82.0	52.2	75.3	67.2	80.3	79.8	75.0	76.3	39.1	68.3	67.3	81.1	67.6
Faster [2]	07+12	73.2	76.5	79.0	70.9	65.5	52.1	83.1	84.7	86.4	52.0	81.9	65.7	84.8	84.6	77.5	76.7	38.8	73.6	73.9	83.0	72.6
Faster [2]	07+12+COCO	78.8	84.3	82.0	77.7	68.9	65.7	88.1	88.4	88.9	63.6	86.3	70.8	85.9	87.6	80.1	82.3	53.6	80.4	75.8	86.6	78.9
SSD300	07	68.0	73.4	77.5	64.1	59.0	38.9	75.2	80.8	78.5	46.0	67.8	69.2	76.6	82.1	77.0	72.5	41.2	64.2	69.1	78.0	68.5
SSD300	07+12	74.3	75.5	80.2	72.3	66.3	47.6	83.0	84.2	86.1	54.7	78.3	73.9	84.5	85.3	82.6	76.2	48.6	73.9	76.0	83.4	74.0
SSD300	07+12+COCO	79.6	80.9	86.3	79.0	76.2	57.6	87.3	88.2	88.6	60.5	85.4	76.7	87.5	89.2	84.5	81.4	55.0	81.9	81.5	85.9	78.9
SSD512	07	71.6	75.1	81.4	69.8	60.8	46.3	82.6	84.7	84.1	48.5	75.0	67.4	82.3	83.9	79.4	76.6	44.9	69.9	69.1	78.1	71.8
SSD512	07+12	76.8	82.4	84.7	78.4	73.8	53.2	86.2	87.5	86.0	57.8	83.1	70.2	84.9	85.2	83.9	79.7	50.3	77.9	73.9	82.5	75.3
SSD512	07+12+COCO	<b>81.6</b>	<b>86.6</b>	<b>88.3</b>	<b>82.4</b>	<b>76.0</b>	<b>66.3</b>	<b>88.6</b>	<b>88.9</b>	<b>89.1</b>	<b>65.1</b>	<b>88.4</b>	<b>73.6</b>	<b>86.5</b>	<b>88.9</b>	<b>85.3</b>	<b>84.6</b>	<b>59.1</b>	<b>85.0</b>	<b>80.4</b>	<b>87.4</b>	<b>81.2</b>

Hình 2.17 Kết quả trên tập VOC07 test

Một số kết quả chạy trên VOC07.

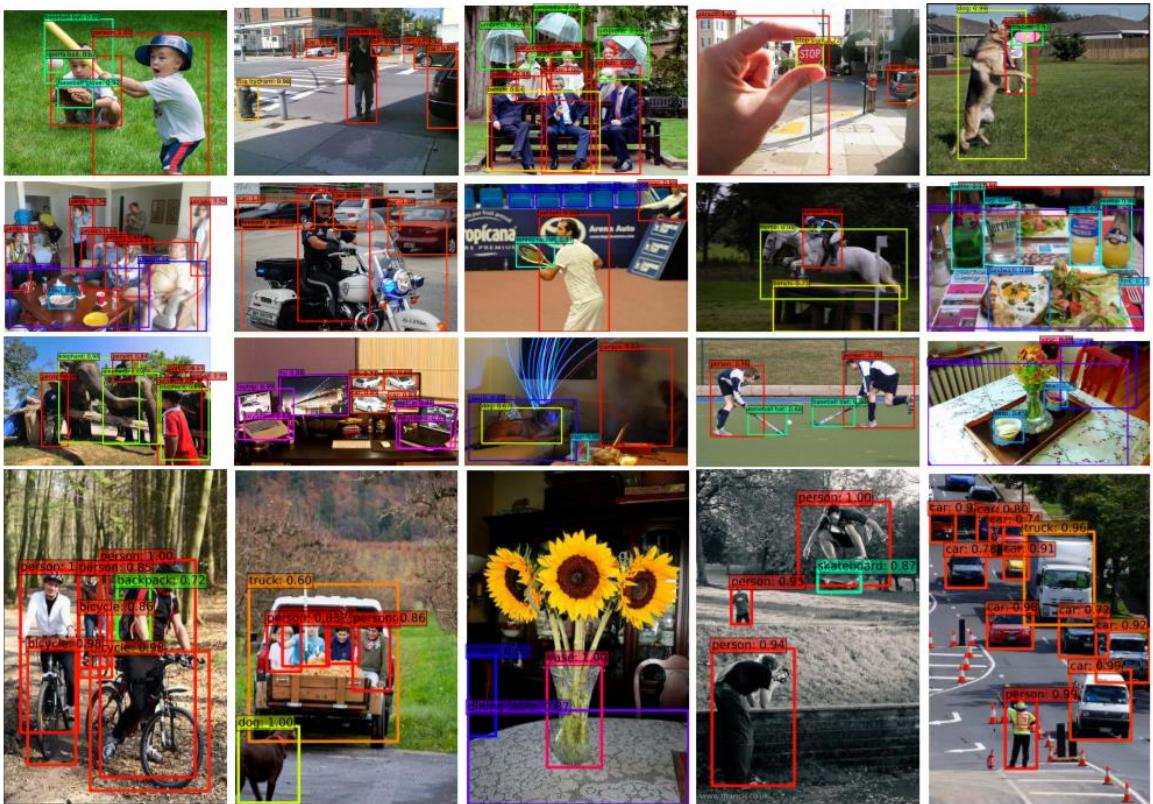


Hình 2.18. Ví dụ minh họa, 7: car, 12: dog, 13:horse, 15: person (Hình 2.17)

SSD tiếp tục chiến thắng trên tập VOC12 test, trong khi ảnh đầu vào của Fast và Faster R-CNN có kích thước thấp nhất là 600 vẫn có kết quả thấp hơn so với SSD300 và SSD512, còn YOLO [24] là 448x448 có kết quả thấp hơn nhiều so với SSD300.

Method	data	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Fast[6]	07++12	68.4	82.3	78.4	70.8	52.3	38.7	77.8	71.6	89.3	44.2	73.0	55.0	87.5	80.5	80.8	72.0	35.1	68.3	65.7	80.4	64.2
Faster[2]	07++12	70.4	84.9	79.8	74.3	53.9	49.8	77.5	75.9	88.5	45.6	77.1	55.3	86.9	81.7	80.9	79.6	40.1	72.6	60.9	81.2	61.5
Faster[2]	07++12+COCO	75.9	87.4	83.6	76.8	62.9	59.6	81.9	82.0	91.3	54.9	82.6	59.0	89.0	85.5	84.7	84.1	52.2	78.9	65.5	85.4	70.2
YOLO[5]	07++12	57.9	77.0	67.2	57.7	38.3	22.7	68.3	55.9	81.4	36.2	60.8	48.5	77.2	72.3	71.3	63.5	28.9	52.2	54.8	73.9	50.8
SSD300	07++12	72.4	85.6	80.1	70.5	57.6	46.2	79.4	76.1	89.2	53.0	77.0	60.8	87.0	83.1	82.3	79.4	45.9	75.9	69.5	81.9	67.5
SSD300	07++12+COCO	77.5	90.2	83.3	76.3	63.0	53.6	83.8	82.8	92.0	59.7	82.7	63.5	89.3	87.6	85.9	84.3	52.6	82.5	<b>74.1</b>	<b>88.4</b>	74.2
SSD512	07++12	74.9	87.4	82.3	75.8	59.0	52.6	81.7	81.5	90.0	55.4	79.0	59.8	88.4	84.3	84.7	83.3	50.2	78.0	66.3	86.3	72.0
SSD512	07++12+COCO	<b>80.0</b>	<b>90.7</b>	<b>86.8</b>	<b>80.5</b>	<b>67.8</b>	<b>60.8</b>	<b>86.3</b>	<b>85.5</b>	<b>93.5</b>	<b>63.2</b>	<b>85.7</b>	<b>64.4</b>	<b>90.9</b>	<b>89.0</b>	<b>88.9</b>	<b>86.8</b>	<b>57.2</b>	<b>85.1</b>	<b>72.8</b>	<b>88.4</b>	<b>75.9</b>

Hình 2.19. Kết quả trên tập VOC12 test



**Hình 2.20. Ví dụ chạy trên tập COCO test-dev với mô hình SSD512**

Kết quả tổng kết trên các tập được công bố trong [21].

Method	VOC2007 test		VOC2012 test		COCO test-dev2015 trainval35k		
	07+12	07+12+COCO	07++12	07++12+COCO	0.5:0.95	0.5	0.75
SSD300	74.3	79.6	72.4	77.5	23.2	41.2	23.4
SSD512	76.8	81.6	74.9	80.0	26.8	46.5	27.8
SSD300*	77.2	81.2	75.8	79.3	25.1	43.1	25.8
SSD512*	<b>79.8</b>	<b>83.2</b>	<b>78.5</b>	<b>82.2</b>	<b>28.8</b>	<b>48.5</b>	<b>30.3</b>

**Hình 2.21. Kết quả tổng hợp, \*: được huấn luyện với phương pháp cải tiến “Data Augmentation for Small Object Accuracy”.**

Dựa vào các kết quả công bố trên, nhóm em quyết định chọn mô hình này áp dụng từ bài toán phát hiện vật thể sang bài toán phát hiện mặt người với hy vọng sẽ cải thiện được về mặt tốc độ để có thể ứng dụng vào real-time.

## 2.4. Nhận biết mặt người bằng DNN – VGG16

Các năm gần đây, sự phát triển của mạng neural network đã mang lại những bước tiến vượt bậc trong lĩnh vực thị giác máy tính, đặc biệt là các bài toán về detection, segmentation, classification,... Bài toán nhận biết mặt người cũng đạt được độ chính xác cao, vượt qua khả năng nhận biết của người. Năm 2015, Omkar M. Parkhi, Andrea Vedaldi và Andrew Zisserman thực hiện công trình nghiên cứu [2], trong đó đề xuất mô hình nhận biết mặt người sử dụng Deep Convolutional Neural Network. Mô hình này đã trở thành baseline và kiến trúc tiêu biểu được tham chiếu bởi gần 400 (Google Scholar) công trình nghiên cứu khác năm 2015.

### 2.4.1. Cấu trúc VGG16

Bảng 2-1. Cấu trúc chi tiết network VGG16 [2]

Layer	Type	Name	Support	Filt dim	Num filts	Stride	Pad
0	input	-	-	-	-	-	-
1	conv	conv1-1	3	3	64	1	1
2	relu	relu1-1	1	-	-	1	0
3	conv	conv1-2	3	64	64	1	1
4	relu	relu1-2	1	-	-	1	0
5	mpool	pool1	2	-	-	2	0
6	conv	conv2-1	3	64	128	1	1
7	relu	relu2-1	1	-	-	1	0
8	conv	conv2-2	3	128	128	1	1
9	relu	relu2-2	1	-	-	1	0
10	mpool	pool2	2	-	-	2	0
11	conv	conv3-1	3	128	256	1	1

12	relu	relu3-1	1	-	-	1	0
13	conv	conv3-2	3	256	256	1	1
14	relu	relu3-2	1	-	-	1	0
15	conv	conv3-3	3	256	256	1	1
16	relu	relu3-3	1	-	-	1	0
17	mpool	pool3	2	-	-	2	0
18	conv	conv4-1	3	256	512	1	1
19	relu	relu4-1	1	-	-	1	0
20	conv	conv4-2	3	512	512	1	1
21	relu	relu4-2	1	-	-	1	0
22	conv	conv4-3	3	512	512	1	1
23	relu	relu4-3	1	-	-	1	0
24	mpool	pool4	2	-	-	2	0
25	conv	conv5-1	3	512	512	1	1
26	relu	relu5-1	1	-	-	1	0
27	conv	conv5-2	3	512	512	1	1
28	relu	relu5-2	1	-	-	1	0
29	conv	conv5-3	3	512	512	1	1
30	relu	relu5-3	1	-	-	1	0
31	mpool	pool5	2	-	-	2	0
32	conv	fc6	7	512	4096	1	0
33	relu	relu6	1	-	-	1	0

34	conv	fc7	1	4096	4096	1	0
35	relu	relu7	1	-	-	1	0
36	conv	fc8	1	4096	2622	1	0
37	softmax	prob	1	-	-	1	0

Chi tiết về cấu trúc của Deep Convolutional Neural Network – VGG16 được thể hiện trong Bảng 2-1. Network bao gồm 11 khối, trong đó mỗi khối bao gồm một toán tử tuyến tính (linear operator) và theo sau đó là một hay nhiều toán tử phi tuyến (non-linearity) như ReLU hay max pooling. Tám khối đầu tiên là convolutional, sử dụng bank of filters. Ba khối cuối được gọi là Fully Connected, về bản chất thì các khối này tương tự như các lớp convolutional nhưng kích thước filter thì bằng với kích thước của dữ liệu đầu vào để mà từ đó các filter “quan sát” được toàn bộ ảnh.

Theo sau tất cả các lớp convolutional là lớp rectification (ReLU) tương tự như trong [25]. Tuy nhiên, không hoàn toàn giống như [25] mà tương tự [26], VGG16 không sử dụng Local Response Normalisation operator. Trong các lớp fully connected sau cùng thì hai lớp đầu tiên có 4096 chiều và lớp cuối có  $N = 2622$  hoặc  $L = 1024$  chiều tùy thuộc vào hàm lỗi (loss functions) được sử dụng để tối ưu hoặc mục đích dự đoán cho  $N$  classes (N-way class prediction). Vector kết quả sau đó được đưa vào lớp softmax để tính toán xác suất hậu nghiệm (posterior probabilities).

Ảnh đầu vào của network này có kích thước  $224 \times 224$  và đã trừ đi ảnh khuôn mặt trung bình (tính toán từ tập dữ liệu).

#### **2.4.2. Kết quả thực nghiệm được công bố**

##### **2.4.2.1. Các tập dữ liệu và cách đánh giá**

###### **- Tập dữ liệu Labeled Faces in the Wild (LFW) [10]**

Tập dữ liệu Labeled Faces in the Wild (được mô tả chi tiết ở 2.5.1) bao gồm 13,233 ảnh của 5,749 người và đây là một tập dữ liệu chuẩn để huấn luyện và đánh giá các

thuật toán về xác nhận khuôn mặt (face verification). Nhóm tác giả [1] dùng hình thức đánh giá tiêu chuẩn “unrestricted setting” sử dụng thêm dữ liệu ngoài để huấn luyện và chọn Equal Error Rate (EER) để làm độ đo. Độ đo này được định nghĩa như là độ lỗi tại điểm trên đường cong ROC (Receiver operating characteristic) mà tại đó tỷ lệ true positive và false negative là bằng nhau.

#### - **Tập dữ liệu YouTube Faces (YTF) [12]**

Tập dữ liệu Youtube Faces (mô tả chi tiết ở 2.5.1) bao gồm 3,425 video của 1,595 người thu thập từ Youtube, trong đó mỗi người có trung bình 2 video. Đây được xem là một tập dữ liệu tiêu chuẩn cho xác nhận mặt người (face verification) trong video. Độ lỗi EER cũng được sử dụng để đánh giá trên Youtube Faces tương tự như 2.4.2.1a.

#### **2.4.2.2. Kết quả**

**Bảng 2-2. So sánh kết quả các mô hình bằng LFW unrestricted setting [2]**

#	Phương pháp	Số ảnh	Số network	Độ chính xác
1	Fisher Vector Faces [27]	-	-	93.10
2	DeepFace [3]	4M	3	97.35
3	Fusion [28]	500M	5	98.37
4	DeepID-2,3		200	99.47
5	FaceNet [11]	200M	1	98.87
6	FaceNet [11] + Alignment	200M	1	99.63
7	VGG16 [2]	2.6M	1	98.95

**Bảng 2-3. So sánh kết quả các mô hình bằng Youtube Face unrestricted setting [2]. K là số lượng người dung để nhận biết trong các video.**

#	Phương pháp	Số ảnh	Số network	100% - EER	Độ chính xác
1	Video Fisher Vector Faces	-	-	87.7	83.8
2	DeepFace [3]	4M	1	91.4	91.4
3	DeepID-2,2+,3		200	-	93.2
4	FaceNet [11] + Alignment	200M	1	-	95.1
5	VGG16 [2] (K=100)	2.6M	1	92.8	91.6
6	VGG16 [2] (K=100) + Embedding Learning	2.6M	1	97.4	97.3

Bảng 2-2 và Bảng 2-3 so sánh độ chính xác của VGG16 với các mô hình đạt kết quả cao nhất trên tập dữ liệu [10] và trên [12]. Điểm nổi bật network trong [2] là đạt được độ chính xác xấp xỉ các mô hình hàng đầu (mặc dù chưa vượt qua được) nhưng số lượng dữ liệu cần sử dụng ít hơn một cách đáng kể và cấu trúc network cũng đơn giản hơn rất nhiều (chỉ sử dụng 1 network).

## 2.5. Các tập dữ liệu liên quan

Trong hai thập kỷ qua, cùng với sự phát triển của lĩnh vực Machine Learning nói chung và Face Recognition nói riêng, có rất nhiều tập dữ liệu mới được tạo ra nhằm mục đích phục vụ cho khoa học và thương mại. Các tập dữ liệu ban đầu có kích thước nhỏ, với các định dạng đơn giản như grayscale hay RGB có độ phân giải chưa cao (còn bị ảnh hưởng nhiều bởi các yếu tố về nhiễu, ánh sáng, tương phản,...). Trong các năm gần đây, với sự phát triển cao của các thiết bị thu nhận hình ảnh cùng với nhu cầu từ các đề tài nghiên cứu cũng như ứng dụng từ giới công nghiệp, rất nhiều tập dữ liệu “không lồ” được tạo ra. Trong đó, số lượng ảnh thu thập tăng lên đáng kể từ hàng trăm ngàn đến hàng triệu. Chất lượng ảnh cũng được cải thiện rõ rệt nhờ cấu tạo tiên tiến của hệ thống camera, có nhiều định dạng ảnh mới ra đời như RGB-D,

các mô hình ba chiều,... Phản tiếp theo tập trung giới thiệu các tập dữ liệu điển hình cho bài toán nhận biết mặt người và giới thiệu tập dữ liệu được chọn để thực hiện đề tài. Bảng 2-4 liệt kê một số tập dữ liệu tiêu biểu dùng cho nhận biết mặt người.

**Bảng 2-4. Một số tập dữ liệu dùng cho nhận biết mặt người**

STT	Tên	Số người	Số mẫu	Năm
1	FaceScrub [29]	530	106,863	2014
2	SCFace [30]	130	4160	2011
3	YouTube Faces [12]	1,595	3,425	2011
4	CASIA-WebFace [31]	10,575	494,414	2011
5	The MUCT Landmarked [32]	276	3755	2010
6	Bosphorus [33]	105	4666	2008
7	CMU Multi-PIE Face [19]	337	750,000	2008
8	Labeled Faces in the Wild [10]	1680	13,000	2007

### **2.5.1. Khảo sát các tập dữ liệu**

#### **2.5.1.1. FaceScrub**

FaceScrub được tạo ra với mục đích cung cấp một tập dữ liệu lớn phục vụ cho nghiên cứu nhận biết mặt người. Nhóm tác giả [29] phát triển một hệ thống phát hiện các khuôn mặt trong ảnh trả về từ việc tìm kiếm ảnh các nhân vật nổi tiếng trên internet, trong đó hệ thống tự động lọc bỏ các ảnh không thuộc về người đang được tìm kiếm.

Tập dữ liệu bao gồm 106,863 ảnh màu của 530 diễn viên nổi tiếng, trong đó mỗi người có trung bình 200 mẫu. Mỗi ảnh được chụp trong môi trường tự nhiên và không có điều kiện ràng buộc nào. Thông tin về tên và giới tính của 265 nam và 265 nữ diễn viên được cung cấp đầy đủ.

Hình 2.22 thể hiện ví dụ về một số khuôn mặt trong tập FaceScrub với sự đa dạng về điều kiện chụp, góc nhìn, độ sáng và rất nhiều thông số camera.



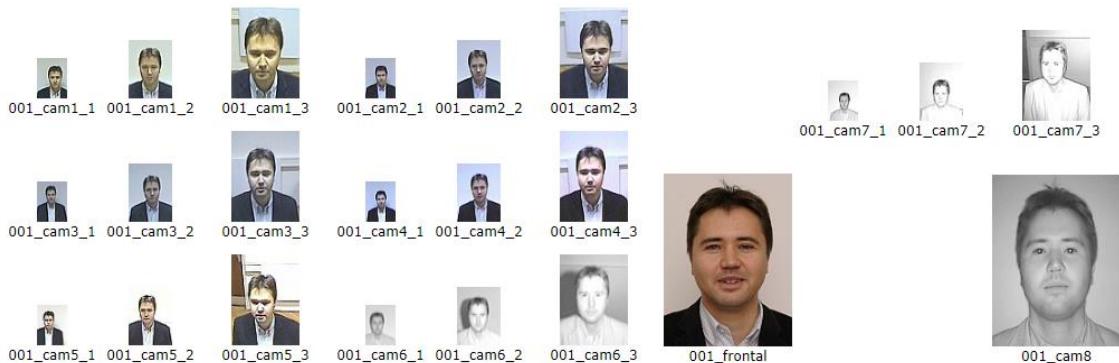
**Hình 2.22. Một số ảnh mẫu trong tập FaceScrub**

(<http://vintage.winklerbros.net/facescrub.html>)

#### 2.5.1.2. SCFace

SCFace là tập dữ liệu chứa các ảnh tinh được thu thập trong môi trường tự do từ năm camera giám sát. Tập dữ liệu gồm 4160 ảnh của 130 người. Ảnh từ các camera với điều kiện khác nhau mô phỏng theo điều kiện thế giới thực cho rất có ích cho các nghiên cứu, kiểm tra hiệu năng của các thuật toán nhận biết mặt người.

Các ảnh ví dụ về ảnh được chụp từ các camera khác nhau được thể hiện trong Hình 2.23.



**Hình 2.23. Tập ảnh ví dụ cho một người trong SCFace**

(<http://www.scface.org/>)

#### 2.5.1.3. Youtube Faces

Youtube faces dataset được thiết kế nhằm phục vụ cho việc nghiên cứu các vấn đề nhận biết mặt người trong video tự do. Tập dữ liệu chứa 3,425 của 1,595 người và

được download từ Youtube. Trung bình mỗi người có 2.15 video. Độ dài các video trung bình là 181.3 frame, video ngắn nhất có 48 frame và dài nhất là 6,070 ms.

Nhóm tác giả [12] cung cấp các bộ test chuẩn để đánh giá hiệu suất của các kỹ thuật video pair-matching. Bảng mô tả (descriptor) cho sự xuất hiện của các khuôn mặt được cung cấp sử dụng các bảng mô tả chuẩn.

#### **2.5.1.4. CASIA-WebFace**

Tập dữ liệu đóng một vai trò quan trọng trong các nghiên cứu về nhận biết mặt người, vì thế nhóm tác giả [31] đề xuất một phương pháp bán tự động để thu thập ảnh từ internet và thành lập một tập dữ liệu mới. CASIA bao gồm 494,414 ảnh tĩnh của 10,575 người và trở thành tập dữ liệu cao thứ hai, chỉ nhỏ hơn tập không được công khai của Facebook lúc công bố.

**Bảng 2-5. Bảng so sánh kích thước tập CASIA-WebFace và một số tập khác [31]**

Tập dữ liệu	Số người	Số ảnh	Công khai
LFW [10]	5,749	13,233	+
WDRef [34]	2,995	99,773	+ (chỉ đặc trưng)
CelebFaces [35]	10,177	202,599	+
SFC [3]	4,030	4,400,000	-
CACD [36]	2,000	163,446	+ (một phần chú thích)
CASIA-WebFace [31]	10,575	494,414	+

Bảng 2-5 so sánh kích thước tập CASIA-WebFace (số lượng người và số mẫu) với một số tập dữ liệu lớn.

#### **2.5.1.5. The MUCT Landmarked**

Tập dữ liệu MUCT được tạo ra để cung cấp các mẫu mặt người đa dạng về độ sáng, tuổi và dân tộc với 3755 khuôn mặt của 76 người. Một vài ví dụ mẫu được thể hiện trong Hình 2.24.



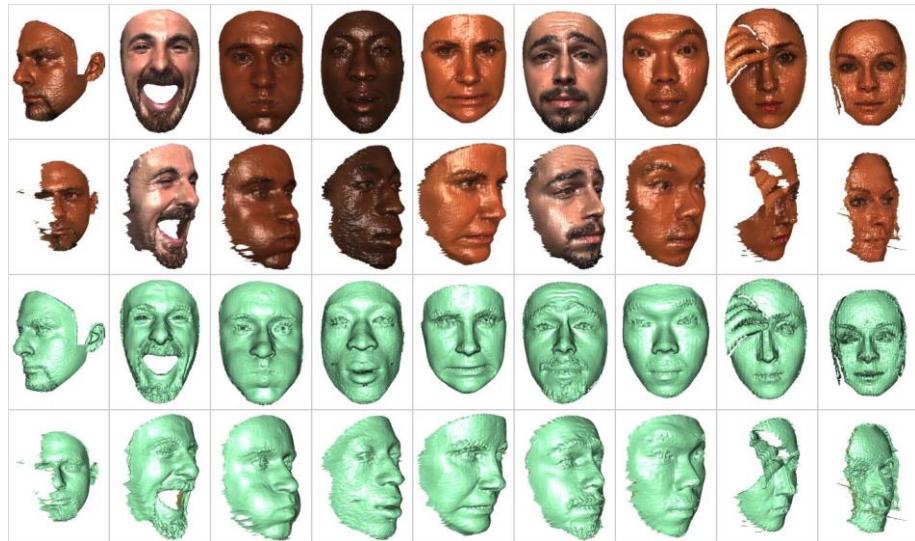
**Hình 2.24. Vài mẫu trong tập dữ liệu MUCT Landmarked**

(<http://www.milbo.org/muct/>)

#### 2.5.1.6. *Bosphorus*

Bosphorus là tập dữ liệu phụ vụ cho nghiên cứu về các bài toán xử lý mặt người 2D và 3D, bao gồm: nhận biết cảm xúc, phát hiện cử chỉ trên mặt (facial action unit detection), ước lượng cường độ đơn vị cử chỉ trên mặt (facial action unit intensity estimation), nhận biết mặt người trong điều kiện bất lợi (face recognition under adverse conditions), mô hình hóa khuôn mặt biến dạng được, tái cấu trúc khuôn mặt ba chiều.

Có tất cả 4666 khuôn mặt của 105 người trong tập dữ liệu với ba đặc điểm chính: đa dạng về biểu cảm (có tới 35 trạng thái cho mỗi người, FACS scoring – bao gồm cường độ và mã bắt đối xứng cho mỗi AU, một phần ba tập dữ liệu là các diễn viên chuyên nghiệp), tư thế khuôn mặt có hệ thống (bao gồm 13 kiểu nghiêng và xoay), có rất nhiều loại che khuất (râu, tóc, tay, mắt kính) (Hình 2.25).



**Hình 2.25. Các mẫu trong tập Bosphorus**

(<http://bosphorus.ee.boun.edu.tr/default.aspx>)

#### 2.5.1.7. *CMU Multi-PIE Face*

Năm 2000, tập dữ liệu PIE database [37] được thu thập để phục vụ nghiên cứu về nhận biết khuôn mặt, trong đó tập trung vào hai yếu tố cản trở việc nhận biết là tư thế và điều kiện chiếu sáng. Tuy đóng vai trò hiệu quả cho công nghệ tài nhưng tập PIE vẫn còn hạn chế ở một số mặt như sau: số lượng cá thể ít, thu thập tại cùng một lần (a single recording session), không đa dạng về biểu cảm. Vì thế, tập Multi-PIE được ra đời, phát triển từ tập dữ liệu cũ để giải quyết các vấn đề hiện hữu.

Tập CMU Multi-PIE Face chứa hơn 750,000 ảnh thu thập từ 337 người với 15 góc nhìn, 19 điều kiện chiếu sáng trong bốn phiên khác nhau. Điều này làm nên sự đa dạng lớn cho tập dữ liệu.

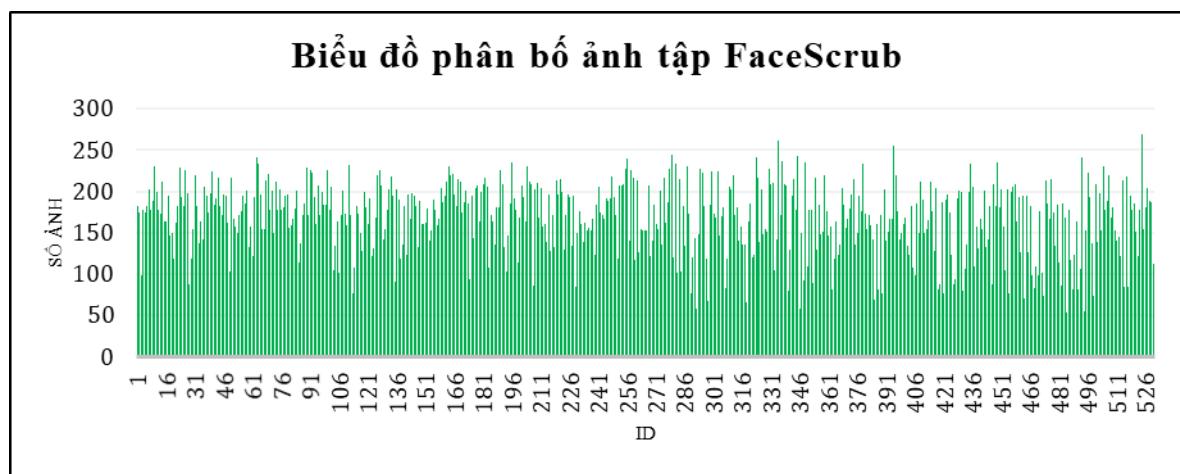
#### 2.5.1.8. *Labeled Faces in the Wild*

Labeled Faces in the Wild là tập dữ liệu được thiết kế cho bài toán nhận biết mặt người trong điều kiện tự do, bao gồm 13,233 ảnh thu thập từ internet của 5,749 người. Trong đó 1,680 người có nhiều hơn một ảnh. Mỗi ảnh được đặt tên theo người trong hình và điểm chung của các khuôn mặt trong tập dữ liệu này đó là đều được phát hiện bởi Viola-Jones face detector.

Đến nay, có tất cả bốn tập LFW bao gồm một tập gốc và ba tập được căn chỉnh. Ba tập dữ liệu mới là “funneled images” (ICCV 2007), LFW-a (được căn chỉnh bằng một thuật toán chưa công bố) và “deep funneled images” (NIPS 2012). Trong số đó, LFW-a và “deep funneled images” được dùng hiệu quả hơn cho các thuật toán xác nhận mặt người (face verification) hơn các tập còn lại (ICCV 2007).

### **2.5.2. Phân tích tập dữ liệu FaceScrub**

Tập dữ liệu FaceScrub nguyên bản gồm 106,863 ảnh màu của 530 diễn viên nổi tiếng và được tác giả cung cấp theo dạng các URL để tải ảnh về từ internet. Nhóm thực hiện đề tài đã xây dựng crawler tự động để thu thập. Do một số URL không hợp lệ hoặc dữ liệu không còn tồn tại tại thời điểm tải nên nhóm thực hiện đề tài chỉ thu được tổng cộng 89295 mẫu, với phân bố được thống kê trong Hình 2.26.



**Hình 2.26. Biểu đồ phân bố dữ liệu trong tập FaceScrub**

Trong đó, người có số lượng ảnh cao nhất là 269, thấp nhất là 53 và trung bình là 168 ảnh/người.

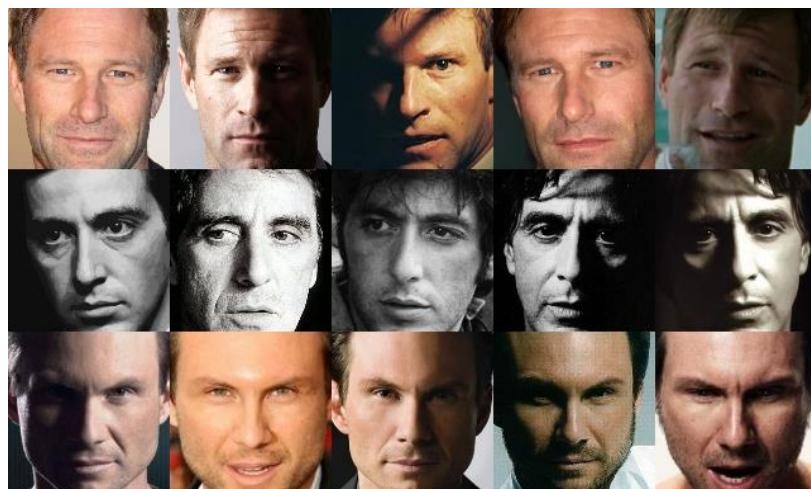
Đây là tập dữ liệu được tìm kiếm từ internet theo tên các diễn viên trong đó, do vậy tồn tại một số vấn đề về chất lượng dữ liệu và điều đó gây ảnh hưởng khá lớn đến độ chính xác của mô hình nhận biết mặt người.

#### **a) Số lượng ảnh của mỗi người**

Số lượng ảnh giữa các lớp có sự khác biệt khá lớn ( $\pm 42$  ảnh). Điều này làm cho những người có số lượng ảnh lớn dễ lẩn át kết quả phân lớp những người có ít ảnh hơn trong tập.

### b) *Điều kiện chiếu sáng*

Điều kiện chiếu sáng tự do và đang dạng trong các ảnh mẫu làm cho rất nhiều khuôn mặt bị che khuất và do đó mắt đi rất nhiều đặc điểm về màu sắc, hình dáng, đường nét,... giúp ích cho việc phát hiện và nhận biết khuôn mặt (Hình 2.27).



**Hình 2.27. Ví dụ về điều kiện chiếu sáng tự do trong FaceScrub**

### c) *Sự lão hóa khuôn mặt*

Các ảnh được thu thập từ internet và không có ràng buộc nào đảm bảo các ảnh của một người sẽ cùng thuộc một giai đoạn tuổi tác của họ. Chính vì thế, sự lão hóa và biến dạng khuôn mặt làm thay đổi đặc điểm nhận dạng rất nhiều. Điều này là một vấn đề lớn và đang được rất nhiều đề tài nghiên cứu quan tâm trong lĩnh vực nhận biết mặt người (Hình 2.28).



**Hình 2.28. Ví dụ về lão hóa trong tập FaceScrub**

*d) Thông số camera*

Thông số camera là một điều đặc biệt quan trọng trong việc quyết định chất lượng hình ảnh. Các thông số tiêu biểu có thể kể đến là góc chụp, hệ màu, độ phân giải,... Sự khác biệt giữa các ảnh do sử dụng các camera quá khác nhau về chất lượng và thông số gây ra những biến thể khó cho việc định danh nhân vật (Hình 2.29).



**Hình 2.29. Ví dụ về sự khác biệt khi dùng các camera quá khác nhau.**

*e) Trang điểm và hóa trang*

Do đặc thù tập dữ liệu FaceScrub bao gồm các diễn viên và nghệ sĩ nên hóa trang, trang điểm là việc hết sức đa dạng. Tùy thuộc vào phim, vào nhân vật trong truyện hay môi trường mà các diễn viên này có sự thay đổi khá lớn, đơn cử là các trường

hợp biến hóa thành quái vật, siêu nhân,... Lúc này ảnh khuôn mặt thay đổi lớn thách thức cả bài toán phát hiện khuôn mặt (Hình 2.30).



**Hình 2.30. Ví dụ về hóa trang và trang điểm trong tập FaceScrub.**

## 2.6. Kết luận

Chương này, chúng em trình bày một số khả năng về các công trình tiên tiến trong lĩnh vực phát hiện và nhận biết mặt người. Trong đó tập trung vào hai mô hình tiêu biểu là [21] cho phát hiện và [2] cho nhận biết mặt người. Đồng thời giới thiệu các tập dữ liệu liên quan, phân tích tập dữ liệu FaceScrub [29] – tập dữ liệu tiêu biểu được chọn để thực hiện các thí nghiệm về sau. Đây là cơ sở cho việc tinh chỉnh cũng như huấn luyện các mô hình ở Chương 2.

## Chương 3

# Huấn luyện mô hình phát hiện và nhận biết mặt người

*Nội dung Chương 3 trình bày các hướng tiếp cận và giải pháp mà nhóm thực hiện để tài triển khai trên mô hình phát hiện và nhận biết mặt người được lựa chọn (SSD300 và VGG16). Các tinh chỉnh đối với mô hình cũ, để xuất cấu trúc mới được trình bày cụ thể sau sự phân tích các mô hình này. Bên cạnh đó, quá trình và chiến lược huấn luyện cũng như việc xây dựng mới một số tập dữ liệu phù hợp được mô tả chi tiết để chứng minh tính hiệu quả của hướng tiếp cận và giải pháp mà nhóm đề ra.*

### 3.1. Mô hình phát hiện mặt người bằng SSD300

SSD không sử dụng chọn lại các đặc trưng giống Fast R-CNN nên không thể phát hiện được các vật thể có kích thước nhỏ mà mặt người xuất hiện trong ảnh thực thì thường chiếm diện tích nhỏ. Ngoài ra, dữ liệu trong tập huấn luyện thì diện tích mặt người lớn và kích thước của vật thể cũng là một giá trị quyết định việc có tìm ra được của vật thể đó trong bức ảnh hay không. Vì thế khó có thể huấn luyện với gương mặt lớn và đòi hỏi phải tìm được gương mặt nhỏ hơn nhiều lần, vì dữ liệu huấn luyện có sẵn chỉ chứa một người trên một ảnh và dữ liệu thực tế thì không như thế. Để giải quyết vấn đề trên, nhóm em đề xuất một số thay đổi để áp dụng mô hình SSD vào bài toán phát hiện mặt người.

#### 3.1.1. Tinh chỉnh SSD300

Thay vì tìm khuôn mặt trên cả ảnh, nhóm chỉ quan tâm đến những vùng « khả thi », vùng chứa người. Cụ thể, nhóm dùng mô hình SSD300<sup>1</sup> với bộ tham số được huấn luyện trên tập COCO + VOC2007+VOC2012, phương pháp cân bằng giữa độ chính xác và tốc độ [21]. Sau khi được các vùng « khả thi », nhóm lại áp dụng SDD300,

---

<sup>1</sup> Mã nguồn được lấy từ trang: <https://github.com/balancap/SSD-Tensorflow>

được huấn luyện lại với tập dữ liệu khuôn mặt (Xem 3.1.2), tạo ra danh sách các ứng cử có thể là mặt người. Danh sách này được tính toán dựa trên tỉ lệ giao, trùng lặp lên nhau và trọng số để quyết định kết quả cuối cùng. Tuy nhiên nếu ở bước tìm ra các vùng chứa người thực hiện ra kết quả không tốt sẽ ảnh hưởng đến bước tìm khuôn mặt. Các trường hợp ảnh có quá nhiều người sẽ làm cho thuật toán chạy chậm; hoặc kích thước người nhỏ làm cho SSD khó có thể phát hiện ra.

### **3.1.2. Xây dựng tập dữ liệu**

Tập dữ liệu huấn luyện được xây dựng từ hai tập dữ liệu về mặt người là Facescrub [29] và ALFW [38]. Đối với tập Facescrub, nhóm em đã loại bỏ các ảnh không phải mặt người bằng DMP<sup>2</sup> trong vòng 5 giờ, do đây là phương pháp chạy tốt đối với những khuôn mặt lớn, rõ (đặc tính của ảnh trong hai tập dữ liệu nêu trên) và có code trên matlab. Với tập ALFW được lưu trữ theo cấu trúc FDDB [39], bao gồm các ảnh có xác định vùng chứa khuôn mặt, như theo elip, chữ nhật hoặc đa giác lồi và được lưu dưới định dạng sqlite3. Nhóm chỉ lấy các ảnh có mặt người được khoanh vùng chữ nhật. Kết quả nhóm thu được 89K Facescrub + 25K ALFW mặt người có cấu trúc PASCAL VOC [40].

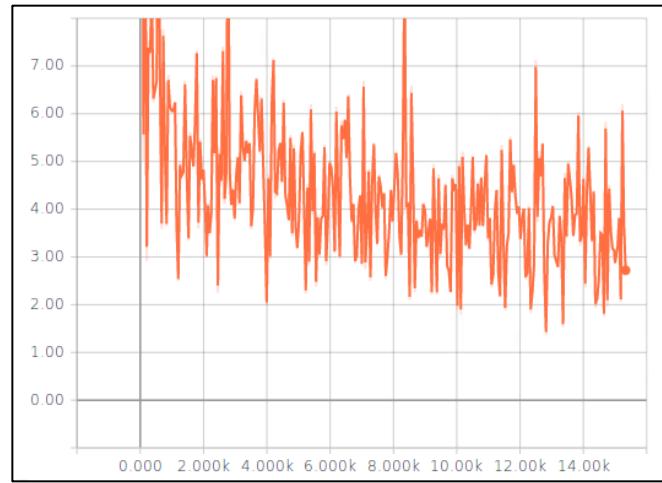
### **3.1.3. Huấn luyện và kết quả**

Nhóm huấn luyện tập dữ liệu trên, khoảng 114K ảnh, bằng mô hình SSD, chạy trên GPU GTX1070 8GB, sử dụng bộ trọng số của VGG16 chạy trong 6 giờ số lần lặp là 15000. Kết quả kiểm tra thực nghiệm bằng các ảnh trên Facebook, Google, so sánh với phương pháp DMP.

---

<sup>2</sup> DMP: phương pháp phát hiện mặt người, code:

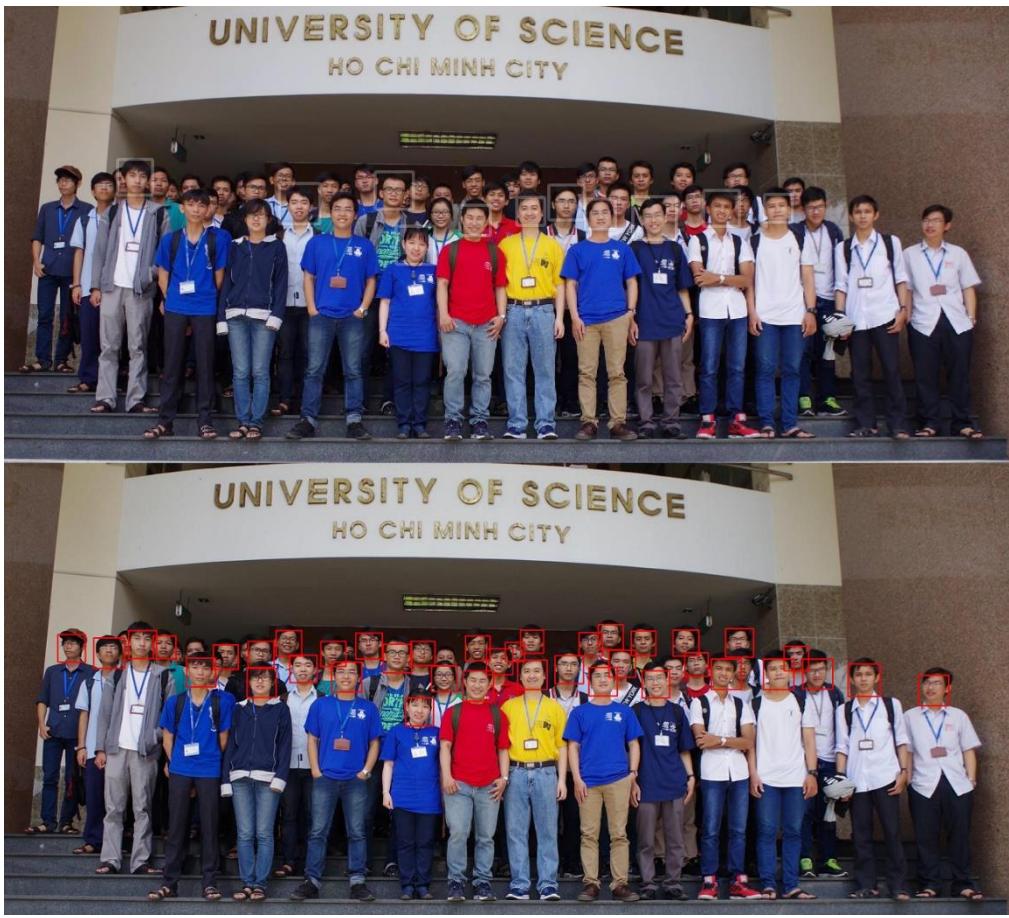
[www.robots.ox.ac.uk/~vgg/software/vgg\\_face/vgg\\_face\\_matconvnet.tar.gz](http://www.robots.ox.ac.uk/~vgg/software/vgg_face/vgg_face_matconvnet.tar.gz)



**Hình 3.1.** Giá trị hàm lỗi trong quá trình huấn luyện.



**Hình 3.2.** Kết quả so sánh với DMP



**Hình 3.3. Kết quả so sánh với Facebook (ảnh phía trên) ngày 02/07/2017**

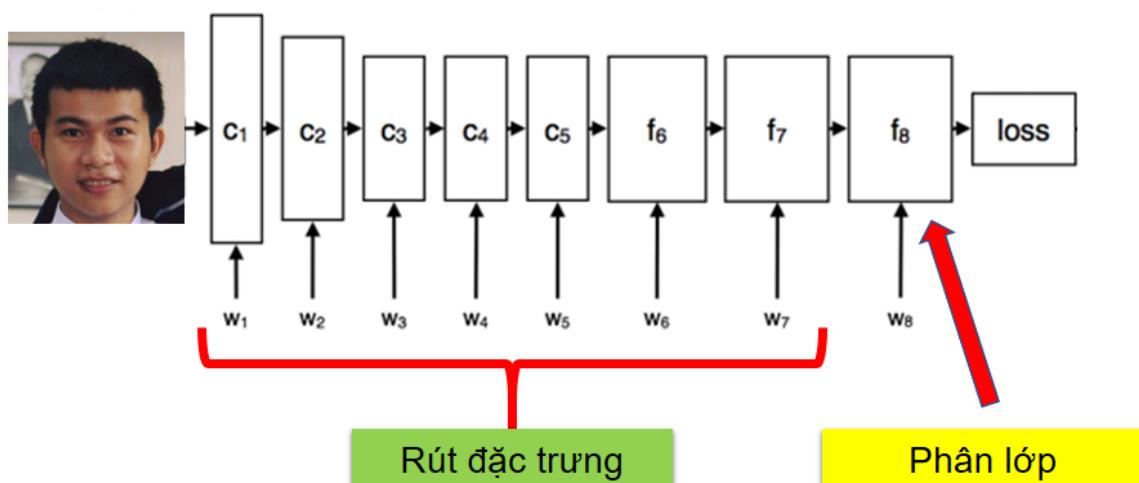
### **3.2. Mô hình nhận biết mặt người bằng VGG-16 Deep features**

#### **3.2.1. Áp dụng kỹ thuật transfer learning**

[2] đã đạt kết quả vượt bậc trên hai tập dữ liệu là LFW [10] và YTF [12] và điều đó chứng minh kiến trúc network này rất phù hợp cho bài toán nhận biết mặt người. Thế nhưng để sử dụng lại cấu trúc này trên một tập dữ liệu hoàn toàn mới thì đòi hỏi một quá trình huấn luyện lâu và tốn nhiều chi phí. Chính vì thế nhóm thực hiện đề tài sử dụng kỹ thuật transfer learning để huấn luyện và đáp ứng yêu cầu trên tập dữ liệu FaceScrub [29].

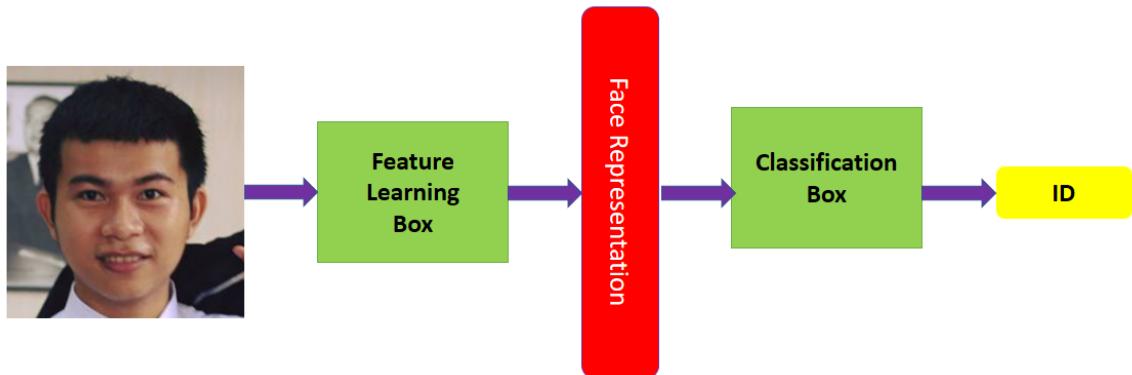
Nếu phân tích cấu trúc network của VGG16 [2] thì có thể chia các lớp thành hai nhóm chủ yếu là các lớp học đặc trưng và các lớp định danh/ phân loại. Nhóm học đặc trưng (feature learning) bao gồm các lớp convolution và hai lớp Fully Connected đầu tiên.

Tầng Fully Connected cuối cùng đảm nhiệm vai trò định danh cho các đặc trưng ảnh được rút ra. Hình 3.4 trình bày vai trò các lớp trong cấu trúc VGG16 network: các khối  $C_x$  tượng trưng cho các lớp convolution theo sau là ReLU nằm giữa các tầng max-pooling, các khối  $f_x$  tương trưng cho các tầng Fully Connected.



**Hình 3.4. Vai trò của các lớp trong VGG16 network**

Nhìn một cách tổng quát thì bài toán nhận biết khuôn mặt theo hướng tiếp cận neural network có thể được chia làm các giai đoạn như sau: từ ảnh đầu vào → hộp đen học đặc trưng → các biểu diễn ảnh theo một chiều không gian khác → hộp đen phân lớp → định danh (Hình 3.5). VGG16 Net [2] đã được huấn luyện để có bộ trọng số tốt cho việc biến đổi ảnh đầu vào thành một cách biểu diễn khác chặt chẽ và cô đọng hơn rất nhiều trong đó làm nổi bật các đặc trưng của ảnh, chính vì thế nhóm thực hiện đề tài đã dùng khối feature learning này rút ra các deep feature trên tập dữ liệu FaceScrub [29] rồi tiến hành phân lớp định danh lại bằng một cấu trúc network khác được trình bày trong phần 3.2.2. Deep feature được rút ra sau tầng Fully Connected thứ hai ( $f_7$ ).



**Hình 3.5. Qui trình nhận biết mặt người tổng quát.**

### 3.2.2. Network định danh VGG16-Deep-Feature

Sau khi rút các deep feature theo cách được trình bày ở phần 3.2.1, nhóm thực hiện đề tài thiết kế một Deep Neural Network mới để phân lớp cho các đặc trưng này. Network mới có cấu trúc đơn giản chỉ bao gồm các lớp Fully Connected phù hợp với việc định danh cho khuôn mặt – tương ứng với vector đặc trưng đầu vào. Cấu trúc network được mô tả chi tiết trong Bảng 3-1. Kích thước deep feature đưa vào là 25,088 – được giữ nguyên so với VGG16, kích thước đầu ra là 530 tương ứng với số lượng diễn viên trong tập FaceScrub [29].

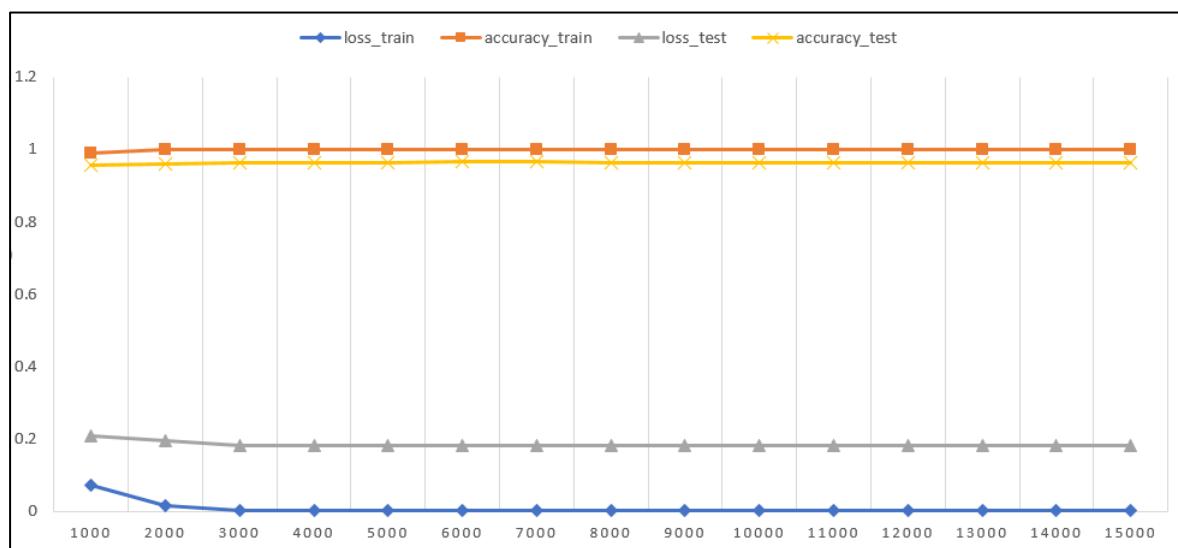
**Bảng 3-1. Cấu trúc network đề xuất để phân lớp VGG16-Deep-Feature**

Layer	Type	Name	#. nodes	Input Size
0	input	-	-	-
1	Fully Connected	fc1	4096	25088
2	Fully Connected	fc2	2048	4096
3	Fully Connected	Prob	530	2048

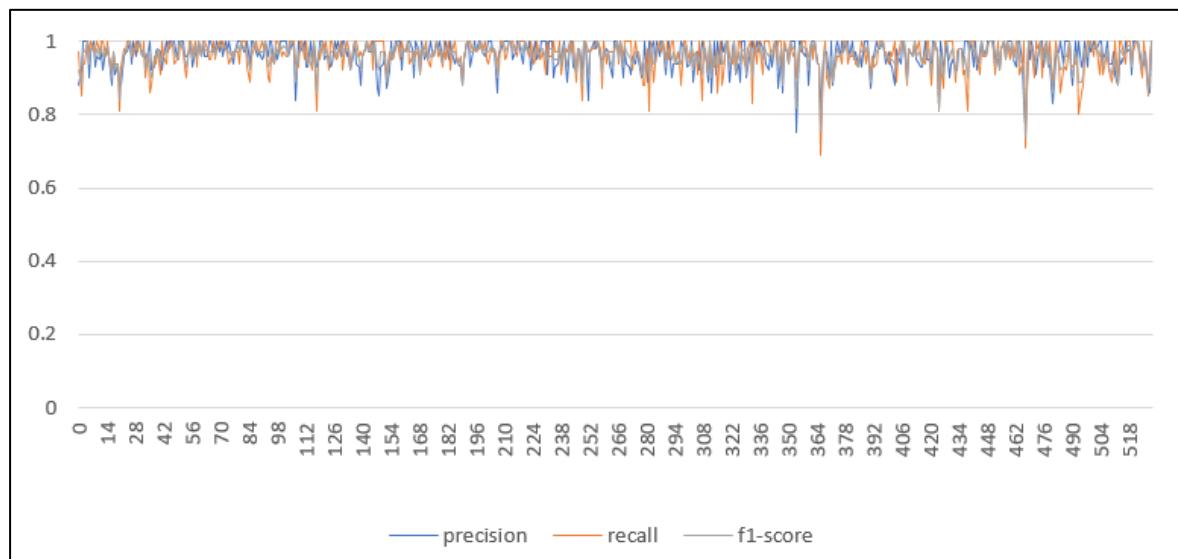
### 3.2.3. Huấn luyện và kết quả

Dữ liệu huấn luyện và kiểm tra là tập Facescrub [29], được chia theo tỉ lệ 8 : 2 tương ứng.

Trước khi huấn luyện, các ảnh đầu vào sẽ được rút đặc trưng bằng VGG16. Sau đó, nhóm sẽ huấn luyện tập đặc trưng này bằng Deep Neural Network. Quá trình huấn luyện nhanh sau 15 epoch, batch có kích thước là 1024 (85s/epoch, 1000step/epoch), sử dụng GPU GTX 1070.



Hình 3.6. Kết quả huấn luyện của Deep Neural Network



Hình 3.7. Đánh giá kết quả đánh giá trên từng lớp

Kết quả, mô hình trên đạt kết quả độ chính xác 96.4% trên tập Facescrub test.

### 3.3. Các công cụ và thư viện sử dụng

Các công cụ và thư viện lập trình sử dụng trong việc cài đặt, xây dựng và huấn luyện mô hình được trình bày trong Bảng 3-2.

**Bảng 3-2. Các công cụ và thư viện cài đặt và huấn luyện mô hình.**

STT	Công cụ/Thư viện	Nội dung
1	Tensorflow-Python [41]	Được sử dụng làm framework chính để cài đặt và huấn luyện các mô hình phát hiện (SSD300) và nhận biết mặt người (VGG16) trong đề tài.
2	OpenCV-Python [42]	Thư viện xử lý ảnh thông dụng cho mục đích tiền xử lý dữ liệu và minh họa kết quả.
3	Caffe-Python [43]	Sử dụng để hỗ trợ cài đặt và huấn luyện mô hình phát hiện mặt người (SSD300).
4	Python [44]	Ngôn ngữ chính được sử dụng để lập trình các mô hình cũng như xây dựng các web APIs (xem phần 4.1).

### 3.4. Kết luận

Nội dung Chương 3 trình bày giải pháp mà nhóm thực hiện để tài đã làm để tinh chỉnh thuật toán SSD300 [21] cho phù hợp với yêu cầu phát hiện mặt người (mục 3.1.1), xây dựng tập dữ liệu mới dựa trên sự kế thừa các tập dữ liệu đã có (mục 3.1.2), để xuất hướng sử dụng deep feature từ VGG16 network (mục 3.2.1) cũng như thiết kế một cấu trúc mới và huấn luyện cho việc định danh các đặc trưng này (mục 3.2.2). Bên cạnh đó, quá trình huấn luyện và các kết quả đạt cũng được trình bày chi tiết và thống kê sau mỗi phần tương ứng nhằm chứng minh tính hiệu quả của các hướng tiếp cận và giải pháp mà nhóm thực hiện để tài đưa ra.

## Chương 4

# Các phân hệ trong hệ thống tương tác thông minh

*Nội dung Chương 4 trình bày thành phần trong hệ thống tương tác thông minh dựa trên tổng hợp thông tin bằng phát hiện và nhận biết mặt người, bao gồm: các Face Web APIs, Person-based video highlight, Character-based filter. Trong mỗi phần tương ứng, nhóm thực hiện đề tài trình bày chi tiết kiến trúc hệ thống, ngũ cành sử dụng và các chức năng được cung cấp cũng như hướng dẫn sử dụng và demo tương ứng.*

### 4.1. Face Web APIs

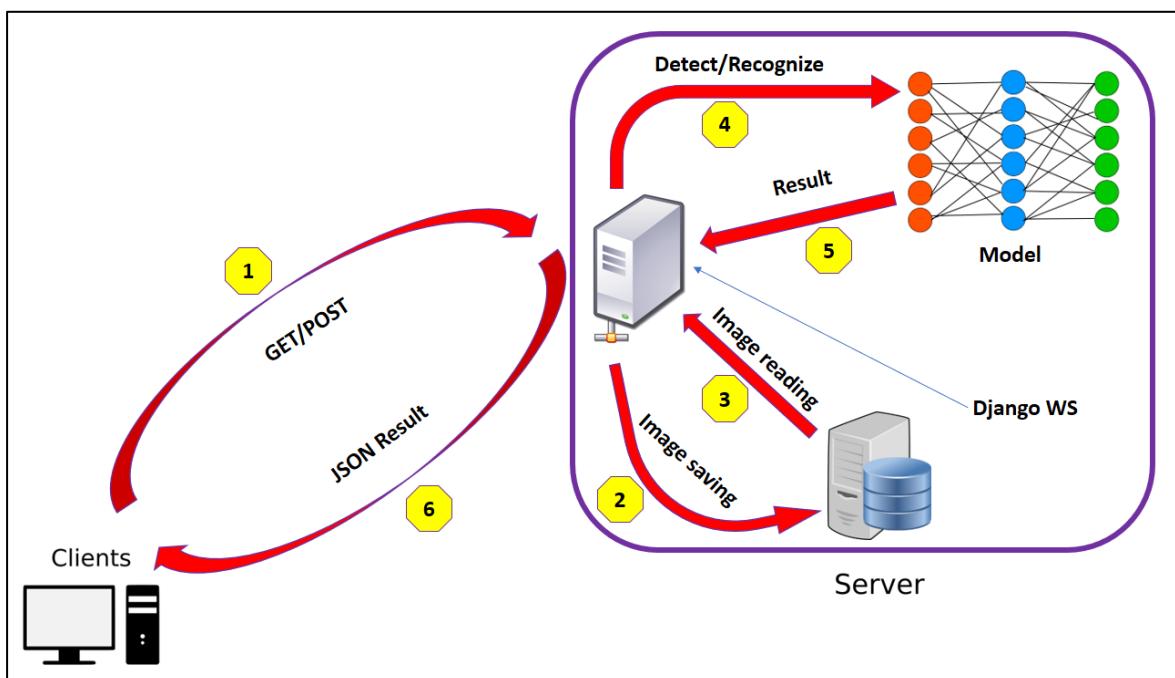
Sau khi huấn luyện hoàn chỉnh các mô hình phát hiện và nhận biết mặt người (trình bày ở Chương 3), nhóm thực hiện đề tài phát triển thành Web APIs với mục đích phục vụ cho các ứng dụng về sau và demo trực quan cho khả năng và tính hiệu quả của các network này. Các APIs được chia thành hai nhóm là phát hiện và nhận diện và được trình bày trong Bảng 4-1. Với nhóm phát hiện khuôn mặt, nhóm phát triển hai APIs sử dụng hai phương pháp khác nhau là SSD300 [21] và OpenCV sử dụng đặc trưng Frontal Haar Cascade. Nhóm nhận diện sử dụng phương pháp duy nhất là VGG16+NN (VGG16-Deep-Feature và phân lớp bằng neural network tự đề xuất). Tuy nhiên, các APIs nhận diện bao gồm hai bước là phát hiện và định danh, do đó, nhóm phát triển hai APIs nhận diện với giai đoạn phát hiện sử dụng hai giải pháp khác nhau đã đề cập.

**Bảng 4-1. Bảng phân loại Face Web APIs đã phát triển.**

Số thứ tự	API	Công dụng
1	SSD300 [21] (*)	Phát hiện
2	OpenCV + Frontal Haar Cascade (**)	Phát hiện
3	VGG16+NN phát hiện bằng (*)	Nhận diện

#### 4.1.1. Kiến trúc hệ thống

Các chứng năng phát hiện và nhận biết mặt người được nhóm sinh viên thực hiện để tài phát triển dưới dạng Web APIs hoạt động theo mô hình client-server như sau :



**Hình 4.1. Mô hình hoạt động của Face Web APIs**

Nghi thức hoạt động của hệ thống Face Web APIs được trình bày chi tiết trong Bảng 4-2.

**Bảng 4-2. Nghi thức hoạt động của Face Web APIs.**

Giai đoạn	Client Side	Server Side
1	Client request lên server bằng giao thức GET/POST trong đó kèm theo ảnh cần xử lý (dạng URL/stream).	<i>Waiting...</i>
2	<i>Waiting...</i>	Server nhận request download ảnh từ URL/stream.

3	<i>Waiting...</i>	Server đọc ảnh lên và chuẩn bị các dữ liệu khác tương ứng.
4	<i>Waiting...</i>	Server gọi ứng dụng bên thứ ba đảm trách vô tròn vận hành model và thực hiện detect/recognize.
5	<i>Waiting...</i>	Server nhận kết quả trả về, chuẩn hóa và định dạng kết quả.
6	<i>Waiting...</i>	Server response client với kết quả xử lý dưới dạng JSON.

Triển khai hệ thống theo kiến trúc như trên mang lại một số ưu điểm và khuyết điểm đi kèm so với khi triển khai trên nội bộ máy tính (Bảng 4-3).

**Bảng 4-3. Ưu và khuyết điểm của kiến trúc hệ thống.**

STT	Khuyết điểm	Ưu điểm
1	Đòi hỏi kết nối internet để truy cập server.	Không giới hạn nền tảng phát triển ứng dụng phía client.
2	Thời gian xử lý chậm hơn do phải truyền dữ liệu lên xuống.	Server có khả năng xử lý mạnh hơn máy cá nhân, đặc biệt hiệu quả trong vận hành các model lớn.
3	Khó triển khai cho các hệ thống realtime.	Dễ dàng nâng cấp và cải tiến hệ thống.

#### **4.1.2. Đặc tả APIs**

Các APIs được phát triển trên framework Django [45] sử dụng ngôn ngữ Python 2 và được sử dụng để phát hiện hay nhận biết các khuôn mặt trong một ảnh tĩnh. Trong đó, nhóm hỗ trợ hai phương thức **GET** và **POST** với định dạng request URL sau:

- **Phương thức GET :**

`http://<IP server>:8000/<loại API>?url=<Image URL>`

- *Phương thức POST :*

`http://<IP server>:8000/recognise`

Trong đó:

- **<IP server>** : địa chỉ server cung cấp API tương ứng (Bảng 4-4).
- **<Image URL>** : URL đến ảnh mong muốn cần xử lý.
- **<loại API>** : “**detect**” hoặc “**recognise**”.
- Phương thức POST gửi ảnh lên với tên: “**image**”.

**Bảng 4-4. Địa chỉ IP của các server cung cấp Face Web APIs.**

Số thứ tự	API	IP
1	SSD300 [21] (*)	<b>128.199.90.168</b>
2	OpenCV + Frontal Haar Cascade (**)	<b>139.59.252.244</b>
3	VGG16+NN phát hiện bằng (*)	<b>128.199.205.131</b>
4	VGG16+NN phát hiện bằng (**)	<b>139.59.252.244</b>

Kết quả được trả về theo định dạng JSON và mô tả chi tiết trong Bảng 4-5.

**Bảng 4-5. Kết quả trả về của các APIs dưới dạng JSON.**

STT	Result Code	Nội dung	JSON	Loại API
1	<b>-1</b>	Request bằng phương thức không phải GET/POST	{ "code": -1 }	-
2	<b>-2</b>	Image URL lỗi	{ "code": -2 }	-

3	<b>-3</b>	Lỗi hệ thống	{ "code": -3 }	-
4	<b>0</b>	<p><b>&lt;số lượng&gt;</b>: số lượng khuôn mặt được phát hiện.</p> <p><b>&lt;x, y, width, height&gt;</b>: tọa độ theo định dạng của bounding box (top-left).</p> <p><b>&lt;URL&gt;</b>: đường dẫn đến ảnh kết quả được visualized. (*)</p>	{           "code": 0,           "num": <số lượng>,           "coordinates": [             "<x,y,wh>",             ...           ],           "url": <URL>         }	GET
5	<b>0</b>	<p><b>&lt;số lượng&gt;</b>: số lượng khuôn mặt được phát hiện.</p> <p><b>&lt;x, y, width, height&gt;</b>: tọa độ theo định dạng của bounding box (top-left).</p> <p><b>&lt;tên&gt;</b>: tên nhân vật. (**)</p> <p><b>&lt;URL&gt;</b>: đường dẫn đến ảnh kết quả được visualized. (*)</p>	{           "code": 0,           "num": <số lượng>,           "names": [             "<tên>",             ...           ],           "coordinates": [             "<x,y,wh>",             ...           ],           "url": <URL>         }	POST

(\*): truy cập theo đường dẫn với định dạng: <IP server>:8000<URL>.

(\*\*): thứ tự tên trong “names” tương ứng với thứ tự tọa độ trong “coordinates”.

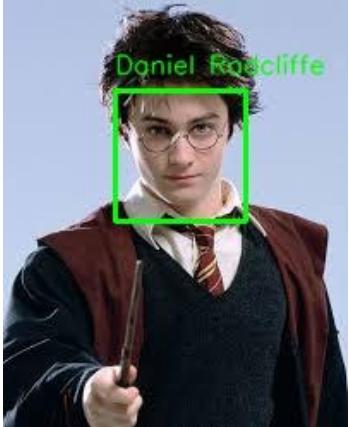
Bảng 4-6 trình bày ví dụ về kết quả xử lý ảnh của diễn viên Daniel Radcliffe vai Harry Potter trong series phim cùng tên (Hình 4.2), trong đó sử dụng thuật toán SSD300 [21] cho phát hiện và VGG16+NN-SSD300 [2] [21] cho nhận diện.



**Hình 4.2. Diễn viên Daniel Radcliffe vai Harry Potter trong series phim cùng tên.**

**Bảng 4-6. Ví dụ kết quả phát hiện và nhận biết mặt diễn viên Daniel Radcliffe.**

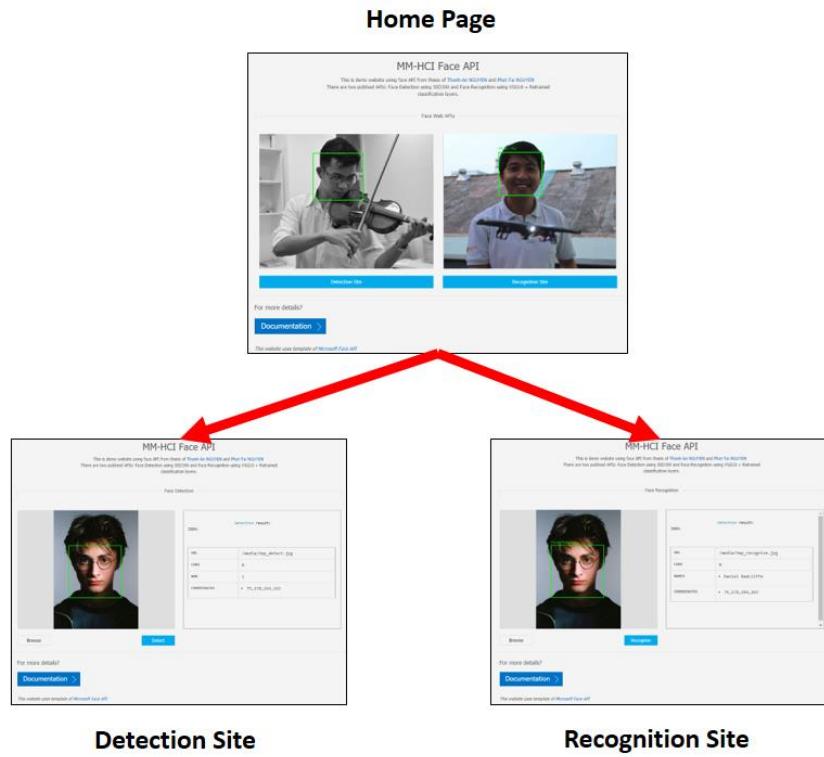
API	Kết quả	Ảnh kết quả
Phát hiện	{ "url": "\media\tmp_detect.jpg", "code": 0, "num": 1, "coordinates": [ "65,53,76,76" ] }	A photograph of Harry Potter with a yellow rectangular box drawn around his face. The text "face/0.999" is displayed above the box in yellow.

	}	
Nhận diện	{         "url": "\media\temp_recognise.jpg",         "code": 0,         "names": [             "Daniel Radcliffe"         ],         "coordinates": [             "65,53,76,76"         ]     }	

#### 4.1.3. Demo Website

Trang chủ của website có địa chỉ <http://128.199.70.20:8000/> bao gồm hai trang con tương ứng với hai loại API chính của đề tài là phát hiện và nhận biết khuôn mặt. Trang chủ có nhiệm vụ giới thiệu và cung cấp thông tin. Hai trang con cung cấp giao diện cho phép người dùng upload ảnh và thực hiện detect/recognise các khuôn mặt trong ảnh đó. Kết quả trả về được thể hiện bằng hai cách: xuất thông tin dưới dạng bảng và vẽ trên ảnh để thể hiện một cách trực quan nhất. Hình 4.3 minh họa cấu trúc cũng như giao diện các trang trong ứng dụng Demo Website.

Thuật toán được sử dụng trong Detection Site là SSD300 [21] và trong Recognition Site là VGG16+NN\_SSD300 [2], [21]. Điểm đặc biệt trong ứng dụng này là webserver nằm hoàn toàn độc lập với các server cung cấp Face APIs. Mọi thao tác xử lý tính toán đều thông qua sự liên kết và trao đổi dữ liệu giữa các server. Điều này cho phép nâng cấp và bảo trì hệ thống một cách dễ dàng.

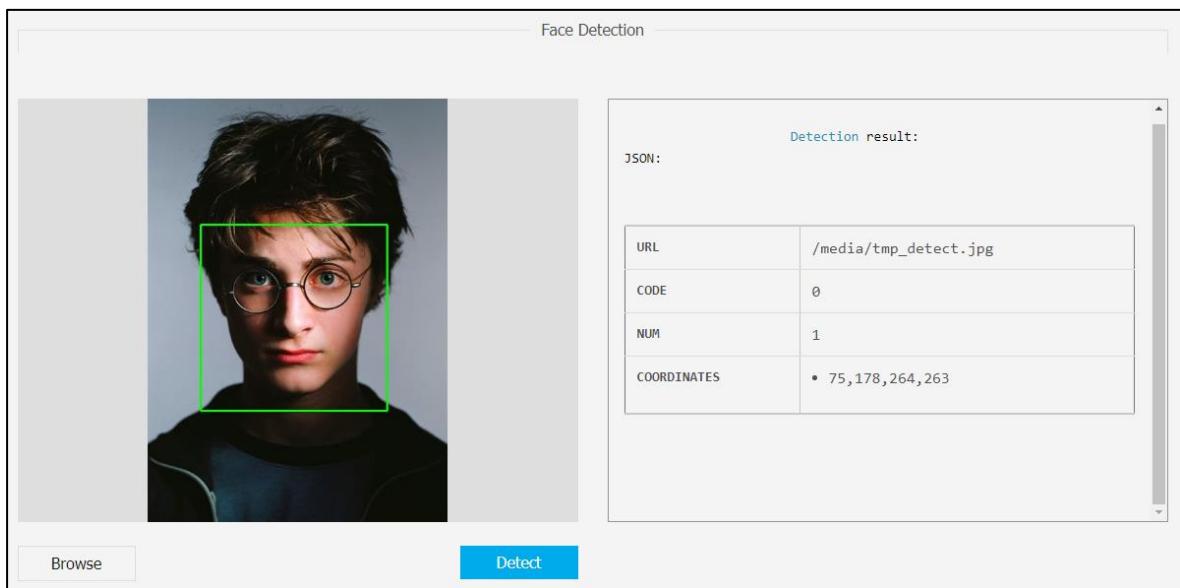


**Hình 4.3. Cấu trúc Demo Website**

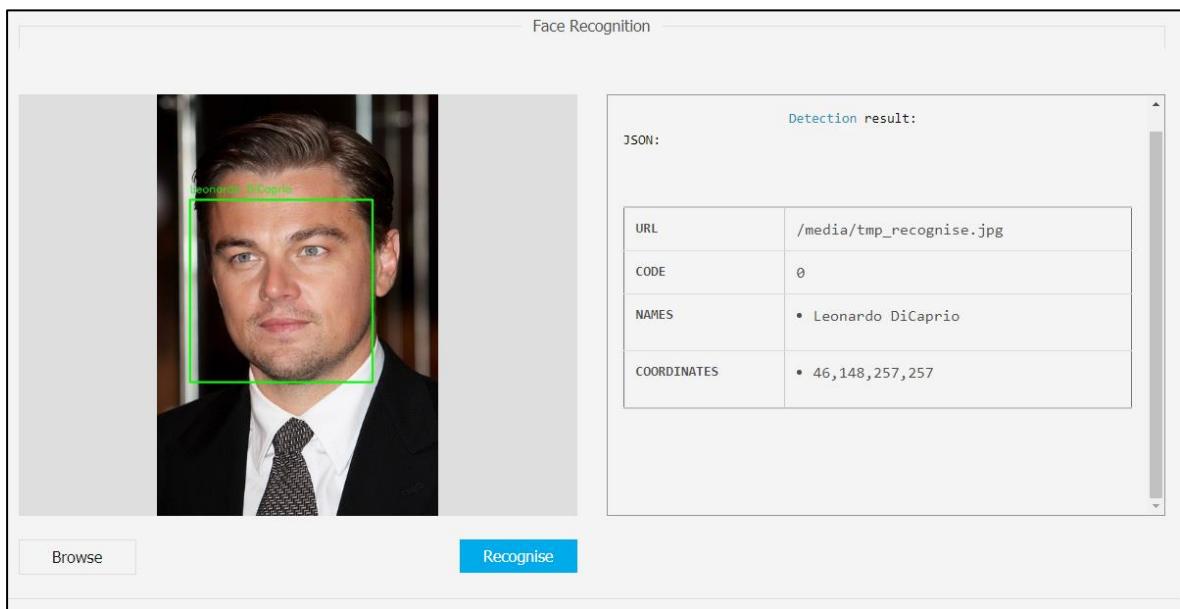
Qui trình hoạt động chi tiết của Detection Site và Recognition Site bao gồm các bước :

1. Người dùng duyệt ảnh bằng nút “Browse” và thực hiện xử lý bằng nút “Detect” hoặc “Recognise”: ảnh sẽ được upload lên server của Demo Website.
2. Server lưu trữ ảnh upload lên và tạo request đến server chuyên trách Detect hoặc Recognise bằng phương thức POST với dữ liệu gửi đi là ảnh vừa nhận
3. Server nhận kết quả trả về dạng JSON, download ảnh được visualized từ server chuyên trách, sau đó render các thông tin này để trả về cho người dùng.

Hình 4.4 trình bày hai ví dụ về giao diện kết quả của Detection Site (a) và Recognition Site (b). Hình ảnh thể hiện được vẽ khung cho từng khuôn mặt (có thêm tên ở góc trái trên nếu là nhận diện) và bên phải là bảng các thông số trả về (xem Bảng 4-5).



(a) *Detection Site*



(b) *Recognition Site*

Hình 4.4. Ví dụ về giao diện kết quả của Detection Site (a) và Recognition Site (b)

## 4.2. Thông tin tổng hợp từ video số dựa trên phát hiện và nhận biết mặt người

### 4.2.1. Dữ liệu đầu ra dạng thô của hệ thống

Đề tài của nhóm sinh viên hướng tới xử lý các video số như phim, tin tức, kênh giải trí,... có thời lượng lớn và đó là một cản trở lớn cho người xem khi không đủ thời gian.

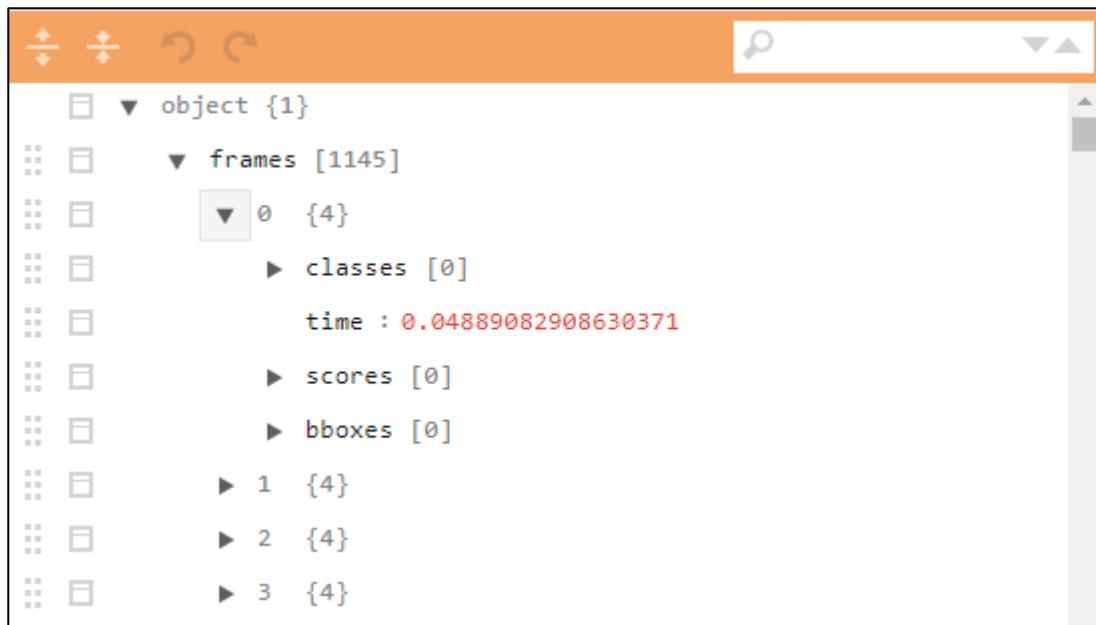
Các video này sẽ được sử dụng làm đầu vào, hệ thống sẽ phân tích và tổng hợp thông tin dựa trên phát hiện và nhận biết mặt người. Sau đó, đầu ra là thông tin chi tiết về nội dung từng frame ảnh (xem Bảng 4-7).

Bảng 4-7. Cấu trúc thông tin kết quả của một frame hình.

STT	Tên thuộc tính	Nội dung
1	classes	Mảng các bộ gồm N số. N là số lượng diễn viên được nhận diện trong video và mỗi phần tử của bộ cho biết tỷ lệ mà người đó được dự đoán.
2	scores	Mảng các số, mỗi số ứng với tỷ lệ vùng bounding box là được dự đoán là mặt người.
3	time	Tổng thời gian xử lý theo giây.
4	bboxes	Mảng các bộ gồm 4 số. Mỗi bộ là một bounding box với các phần tử lần lượt là tọa độ y_min, x_min, y_max, x_max.

Để dễ dàng xử lý và tránh quá tải bộ nhớ, các video được cắt thành từng đoạn nhỏ hơn rồi mới đưa vào hệ thống. Mỗi video nhỏ hơn đưa vào hệ thống sẽ cho ra một tập tin định dạng JSON chứa kết quả xử lý của từng frame và giữ đúng thứ tự của chúng. Hình 4.5. Một tập tin đầu ra dạng JSON. Hình 4.5 là ví dụ cho một tập tin đầu ra ứng với một video nhỏ đưa vào.

Với mục đích tổng hợp thông tin và thể hiện dưới dạng cô đọng, dễ dàng sử dụng nhất có thể, chúng em đã xây dựng một ứng dụng xử lý các tập tin đầu ra dạng thô và trình bày chi tiết trong phần tiếp theo.



The screenshot shows a JSON editor interface with an orange header bar. The main area displays a hierarchical tree of JSON data. At the top level, there is an object with one item. This item is an array named 'frames' containing 1145 elements. The first element of this array is indexed as '0' and contains four items. These items are arrays for 'classes', 'scores', and 'bboxes', and another array indexed as '1' which also contains four items. The 'time' value for the first frame is highlighted in red as '0.04889082908630371'. The interface includes standard window controls (minimize, maximize, close) and a search bar at the top right.

**Hình 4.5. Một tập tin đầu ra dạng JSON.**

(Công cụ hiển thị sử dụng: <http://jsoneditoronline.org/>)

#### **4.2.2. Ứng dụng Smart Video Editor – tổng hợp thông tin video**

Ứng dụng này được phát triển bằng ngôn ngữ C# trên nền tảng .NET framework 4.5.2 với mục đích tổng hợp thông tin từng frame rời rạc dạng thô (4.2.1) thành dạng cô đọng và dễ dàng sử dụng hơn.

Smart Video Editor được thiết kế dành cho nhà biên tập với đầu vào là dữ liệu nội dung thô của từng frame và đầu ra là thông tin chi tiết của video bao gồm các nhân vật chính, mỗi nhân vật xuất hiện ở những phân đoạn nào, ... (xem Bảng 4-8).

**Bảng 4-8. Cấu trúc tập tin đầu ra dạng JSON của Smart Video Editor**

STT	Tên thuộc tính	Nội dung
1	folder_path	Đường dẫn đến thư mục input
2	title	Tên video
3	image_path	Đường dẫn đến thư mục ảnh các diễn viên. Mỗi người có một ảnh và đặt tên theo định dạng <tên>.jpg
4	video_path	Đường dẫn đến video
5	duration	Tổng thời lượng video
6	frame_width	Chiều rộng frame ảnh
7	frame_height	Chiều cao frame ảnh
8	fps	Tốc độ video
9	num_character	Số lượng diễn viên
10	characters	Danh sách tên diễn viên
11	<Tên diễn viên>	Mảng chứa các phân đoạn. Mỗi phần tử chứa thông tin: tên diễn viên, frame bắt đầu, frame kết thúc, thời điểm bắt đầu, thời điểm kết thúc.

Hình 4.6 là một ví dụ về tập tin đầu ra của ứng dụng Smart Video Editor, trong đó phần khung màu đỏ là các thuộc tính ứng với tên diễn viên chứa các phân đoạn mà họ xuất hiện.

Đầu vào của ứng dụng là một thư mục chứa dữ liệu thô xuất ra từ quá trình phát hiện và nhận biết khuôn mặt có cấu trúc như sau:

- Thư mục “img”: chứa ảnh của các diễn viên. Tên tập tin đặt theo cú pháp  
`<tên diễn viên>.jpg`
- Thư mục “meta”: chứa các tập tin json nội dung các frame. Mỗi tập tin tương ứng với một video nhỏ được cắt ra khi xử lý và đặt tên theo cú pháp:  
`<số thứ tự (3 chữ số)>.json`

Tập tin “info.json”: gồm các thông tin được mô tả trong Bảng 4-9 (xem ví dụ Hình 4.7)

```

object {21}
  folder_path : C:\\Users\\an\\Desktop\\Fantastic_Beasts_and_Where_to_Find_Them
  title : Fantastic Beasts and Where to Find Them
  image_path : C:\\Users\\an\\Desktop\\Fantastic_Beasts_and_Where_to_Find_Them/img
  video_path : D:/001PROJECTS/Thesis/Videos/Oscar/Fantastic_Beasts_and_Where_to_Find_Them.mp4
  duration : 02:12:52
  frame_width : 1280
  frame_height : 534
  fps : 23
  num_character : 10
  characters [10]
    Eddie Redmayne [32]
    Katherine Waterston [18]
    Colin Farrell [9]

```

**Hình 4.6. Ví dụ tập tin đầu ra của Smart Video Editor (khung đỏ là các thuộc tính ứng với từng diễn viên)**

(Công cụ hiển thị sử dụng: <http://jsoneditoronline.org/>)

**Bảng 4-9. Cấu trúc tập tin “info.json”**

STT	Tên thuộc tính	Nội dung
1	title	Tên video
2	path	Đường dẫn đến video
3	duration	Tổng thời lượng video
4	frame-width	Chiều rộng frame ảnh
5	frame-height	Chiều cao frame ảnh
6	fps	Tốc độ video
7	num-meta	Số lượng file json (số lượng video được cắt nhỏ)

8	num-character	Số lượng diễn viên
9	characters	Danh sách tên diễn viên

```

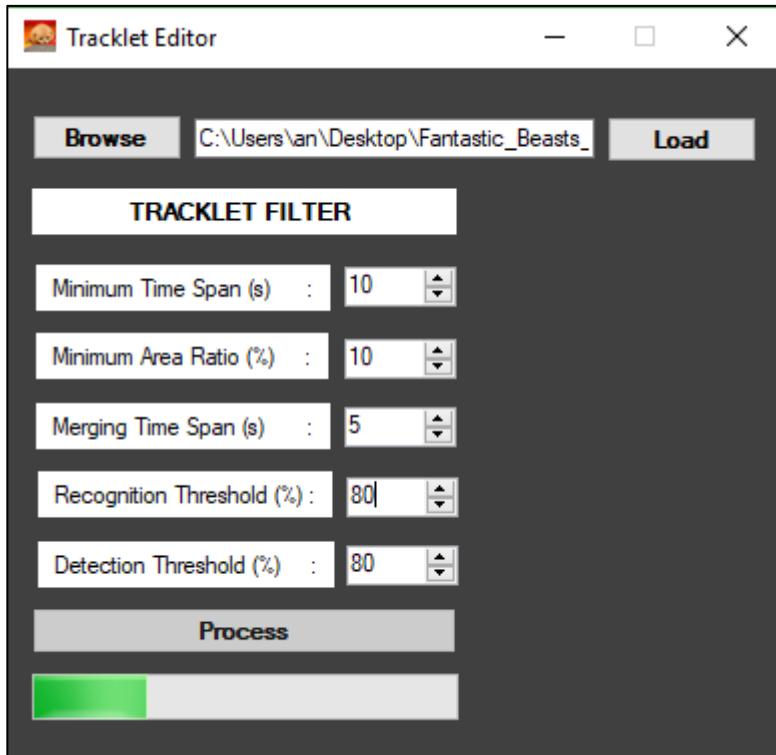
object {9}
  title : Fantastic Beasts and Where to Find Them
  path : D:/001PROJECTS/Thesis/Videos/Oscar/Fantastic_Beasts_and_Where_to_Find_Them.mp4
  duration : 02:12:52
  frame-width : 1280
  frame-height : 534
  fps : 23
  num-meta : 20
  num-character : 10
  characters [11]
    0 : Eddie Redmayne
    1 : Katherine Waterston
    2 : Colin Farrell
  
```

**Hình 4.7. Ví dụ tập “info.json”**

(Công cụ hiển thị sử dụng: <http://jsoneditoronline.org/>)

Như đã trình bày ở trên, Smart Video Editor là ứng dụng dành cho biên tập viên để tổng hợp thông tin là các phân đoạn mà các diễn viên xuất hiện. Trong quá trình tổng hợp đó, nhà biên tập có thể lựa chọn và thay đổi các thông số đặc trưng cho phù hợp với từng video cụ thể, bao gồm (xem thêm Hình 4.8):

- Thời lượng tối thiểu cho mỗi phân đoạn được phát hiện.
- Tỷ lệ diện tích tối thiểu để một khuôn mặt được tính là có xuất hiện trong frame.
- Khoảng thời gian giữa hai phân đoạn của cùng một diễn viên để gộp thành một.
- Độ chính xác tối thiểu cho thuật toán phát hiện khuôn mặt.
- Độ chính xác cho thuật toán nhận biết mặt người.



Hình 4.8. Giao diện hoạt động của ứng dụng Smart Video Editor.

### 4.3. Person-based video highlight

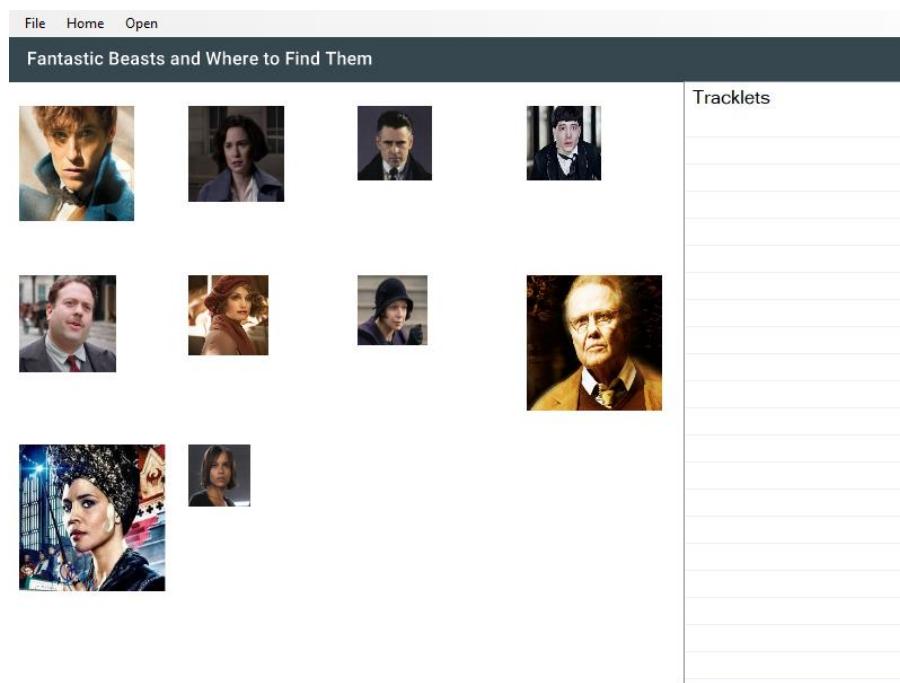
#### 4.3.1. Ngữ cảnh sử dụng

Ứng dụng Person-based video highlight được xây dựng bằng ngôn ngữ C# trên nền tảng .NET Framework 4.5.2 với mục đích mang đến một khả năng tương tác thông minh và trải nghiệm thú vị cho người xem. Bên cạnh đó ứng dụng này đặc biệt hữu ích cho những người có cuộc sống, làm việc bận rộn nhưng vẫn muốn theo dõi nội dung phim, tin tức và đặc biệt là nhân vật, diễn viên mà họ yêu thích.

Person-based video highlight sử dụng dữ liệu được output từ Smart Video Editor (4.2.2) để thể hiện trực quan một bộ phim thành dạng poster. Ứng với mỗi khuôn mặt là một dãy các phân đoạn mà họ xuất hiện giúp người xem có thể lướt nhanh qua nội dung và xem những phần chính họ mong muốn. Bằng cách tiếp cận này, ứng dụng đã tiết kiệm được rất nhiều thời gian cho người dùng và mang đến một trải nghiệm mới.

### 4.3.2. Giao diện và hệ thống chức năng

Phần 4.3.2 tập trung giới thiệu giao diện trực quan và hệ thống chức năng hỗ trợ của ứng dụng Person-based video highlight. Hình 4.9 là giao diện màn hình **Home** sau khi tải dữ liệu bằng menu **Open**.



**Hình 4.9.** Màn hình Home sau khi load dữ liệu của Person-based Video Highlight.

Khi nhấn vào ảnh của diễn viên quan tâm thì cửa sổ xem video cùng danh sách các phân đoạn tương ứng hiện lên cho người dùng lựa chọn (Hình 4.10). Về tổng thể video vẫn chạy tuần tự và chỉ khi người dùng chọn phân đoạn nào đó thì video mới chuyển đến phân đoạn ứng. Ngoài ra, có thể nhấn **Home** để quay về giao diện chính và lựa chọn nhân vật khác.



Hình 4.10. Giao diện xem video theo các phân đoạn của diễn viên

Eddie Redmayne.

#### 4.4. Person-based filter

##### 4.4.1. Ngữ cảnh sử dụng

Person-based filter là ứng dụng được phát triển bằng ngôn ngữ C# trên nền tảng .NET Framework 4.5.2 với mục tiêu hỗ trợ người dùng truy vấn các phân đoạn video liên quan đến một nhân vật mà họ yêu thích.

Các video tin tức, phim ảnh, giải trí,... sau quá trình phân tích sẽ được tổng hợp thành một cơ sở dữ liệu trong đó có thông tin về các phân đoạn mà các nhân vật tiêu biểu xuất hiện. Từ đó hệ thống cung cấp cho người xem một tính năng là truy vấn theo tên của nhân vật mà họ yêu thích để có thể xem các đoạn liên quan đó.

#### **4.4.2. Thực nghiệm đánh giá**

Để đánh giá mức độ hiệu quả của hệ thống, nhóm sinh viên đã xây dựng mô hình phát hiện và nhận biết mặt người cho các chính trị gia nổi tiếng hiện tại như : Donald Trump, Barack Obama, Hillary Clinton, Vladimir Putin, ... và 623 phân đoạn video của 7 nhân vật trong danh sách huấn luyện. Trong mỗi video có thể xuất hiện một hoặc một vài chính khách khác nhau.

Qui trình đánh giá được thực hiện bằng cách truy vấn lần lượt tên của 7 chính khách và kiểm tra kết quả trả về để tính precision và recall cho mỗi truy vấn. Kết quả đó được trình bày chi tiết trong bảng Bảng 4-10.

**Bảng 4-10. Kết quả đánh giá hiệu suất truy vấn.**

STT	Tên	Số video	Precision	Recall
1	Donald Trump	316	1.00	0.52
2	Barack Obama	173	0.94	0.88
3	Hillary Clinton	128	0.95	0.62
4	Vladimir Putin	99	0.98	0.51
5	António Guterres	59	1.00	0.98
6	Ban Ki-moon	24	0.92	1.00
7	Shinzō Abe	65	0.85	0.95

#### **4.4.3. Giao diện ứng dụng**

Hình 4.11 thể hiện giao diện ứng dụng Person-based Filter, trong đó gồm ba thành phần chính : thanh công cụ tìm kiếm, danh sách phân đoạn và màn hình chính để xem cách video. Người dùng có thể điền tên nhân vật mình yêu thích, quan tâm như : diễn viên, chính trị gia, ... vào thanh công cụ và nhấn nút tìm kiếm để có được danh sách các phân đoạn video tương ứng. Nhấn chọn một phân đoạn để xem nội dung được trình chiếu trên màn hình chí.



**Hình 4.11. Giao diện ứng dụng Person-based Filter.**

#### 4.5. Các công cụ và thư viện sử dụng

Các môi trường phát triển tích hợp, các công cụ và thư viện được sử dụng để xây dựng ứng dụng minh họa được trình bày trong bảng Bảng 4-11.

**Bảng 4-11. Các công cụ và thư viện phát triển ứng dụng minh họa**

STT	Công cụ/Thư viện	Nội dung
1	Visual Studio 2015 [46]	Công cụ được sử dụng để lập trình các ứng dụng minh họa được trình bày trong Chương 4.
2	C# + .NET 4.5.2	Ngôn ngữ và nền tảng phát triển các ứng dụng minh họa.
3	Material Skin [47]	Thư viện material skin được sử dụng để thiết kế giao diện cho các ứng dụng.
4	Json.NET [48]	Thư viện sử dụng để xử lý các tập tin JSON trong đè tài.

## 4.6. Kết luận

Trong chương này, chúng em đã trình bày những thành phần chính tạo nên hệ thống tương tác thông minh dựa trên phát hiện và nhận biết mặt người. Trong đó bao gồm hệ thống 4 web APIs và một webserver minh họa có tính năng của hai mô hình đã được cài đặt và huấn luyện với độ chính xác cao ở Chương 3. Bên cạnh, đó nhóm cũng pháp triển ứng dụng giúp biên tập viên tổng hợp thông tin từ video số dựa trên dữ liệu đầu ra của hệ thống xử lý video. Một ứng dụng minh họa cho khả năng xem video và hỗ trợ tương tác thông minh cho người xem bằng cách chỉ mục nhanh theo luồng các phân đoạn xuất hiện của một nhân vật. Không những thế, những phân đoạn mà họ xuất hiện cũng được liệt kê để người xem có thể tập trung nhanh vào những phân họ thích và tiết kiệm được khá nhiều thời gian nhưng vẫn theo kịp nội dung cốt truyện. Và cuối cùng là khả năng truy vấn các phân đoạn liên quan đến các nhân vật nào đó được yêu thích.

## Chương 5

### Kết luận

 Nội dung của Chương 5 trình bày các kết quả đạt được và hướng phát triển của đề tài.

#### 5.1. Các kết quả đạt được

Trong đề tài này, chúng em đã tìm hiểu và nắm bắt được cấu trúc cũng như nguyên lý hoạt động của neural network và cụ thể là convolutional neural network cho bài toán phát hiện và nhận biết mặt người. Sau đó thực hiện cài đặt lại, đề xuất kiến trúc và chỉnh sửa mô hình phát hiện vật thể SSD300 [21] và nhận biết mặt người VGG16 [2]. Bằng cách này nhóm đã tinh chỉnh lại mô hình SSD300 [21] cho phù hợp với việc phát hiện khuôn mặt (phần 3.1.1) cũng như tận dụng được tri thức đã huấn luyện trong [2] (phần 3.2.1) cho việc định danh trên một tập dữ liệu mới, [29]. Nhóm thực hiện xây dựng một tập dữ liệu mới dựa trên việc tổng hợp các tập nổi tiếng hiện có (3.1.2) để huấn luyện mô hình phát hiện và sử dụng tập dữ liệu [29] để huấn luyện mô hình định danh. Kết quả thực nghiệm chứng minh tính hiệu quả của hướng tiếp cận mà nhóm sinh viên đã đề xuất (phần 3.1.3 và 3.2.3).

Bên cạnh, nhóm đã xây dựng và vận hành hệ thống gồm 4 webservices và 1 webserver minh họa thực hiện chức năng phát hiện và nhận biết khuôn mặt của 530 nam nữ diễn viên trong tập FaceScrub [29] (phần 4.1). Đây là một hệ thống mở để phát triển các ứng dụng và công cụ tham khảo cho các đề tài về sau.

Nhóm sinh viên đã xây dựng một hệ thống hỗ trợ người dùng xem video và tương tác thông minh, trong đó giúp họ có thể xem nhanh các phân đoạn mà diễn viên quan tâm xuất hiện (phần 4.3). Từ đó tiết kiệm được thời gian nhưng vẫn theo dõi được nội dung video phim, tin tức,... Một chức năng khác được phát triển song song là hỗ trợ người xem truy vấn các phân đoạn, video theo tên nhân vật mà họ quan tâm (xem 4.4). Đây là một hướng tiếp cận mới vì nó dựa trên phát hiện và nhận biết mặt người thay vì tìm kiếm và xử lý tiêu đề như trước đây. Với hệ thống này, nhóm sinh viên

mong muốn mang đến cho người dùng một trải nghiệm thú vị và khả năng tương tác thông minh.

## 5.2. Hướng phát triển của đề tài

Chúng em sẽ tiếp tục tìm hiểu sâu hơn về đề tài phát hiện và nhận biết mặt người, trong đó chú trọng nghiên cứu và phân tích các kiến trúc sử dụng convolutional neural network. Trên nền tảng đó đề xuất các kiến trúc mới để giải quyết các vấn đề còn tồn động như tư thế khuôn mặt (pose), điều kiện chiếu sáng (illumination), che khuất (occlusion), lão hóa (aging), .... Tham khảo các công trình tiên tiến và các tập dữ liệu tiêu chuẩn để có công bố một công trình khoa học hoàn chỉnh.

Về mặt ứng dụng và hiện thực hóa đề tài, nhóm sinh viên sẽ nghiên cứu thêm hai lĩnh vực là xác nhận khuôn mặt (face verification) và định danh (face identification). Sau đó, phát triển thành một hệ thống Face Web APIs hoàn chỉnh với bốn nhóm modules. Hiện tại với cùng ý tưởng này thì đã có hai hệ thống APIs nổi tiếng và hiệu quả đang hoạt động của Google và Microsoft Project Oxford. Chúng em sẽ tìm hiểu và phát triển một hệ thống tương tự để hỗ trợ cho các ứng dụng trong nước và các đề tài nghiên cứu liên quan. Chức năng mà nhóm đang hướng tới là cho phép người dùng tự tạo ra tập dữ liệu cá nhân và hệ thống sẽ tự động huấn luyện để có mô hình định danh cho những người trong tập dữ liệu ấy.

Bên cạnh đó, để phục vụ tốt hơn cho người dùng, nhóm tập trung vào tối ưu hiệu suất cho các mô hình tính toán. Với các cấu trúc nhiều lớp phức tạp thì khả năng đáp ứng cho các ứng dụng realtime là không khả thi khi giao tiếp theo kiến trúc client-server qua các Web APIs. Chúng em hướng tới các giải pháp tối ưu như sau: triển khai hệ thống APIs trên các server có cấu hình mạnh và đặc biệt là sử dụng GPU để tăng tốc độ xử lý, thu nhỏ kiến trúc mô hình để tạo thành các framework offline hỗ trợ cho các ứng dụng realtime trên máy cá nhân và thiết bị di động.

Đối với hệ thống tương tác thông minh hiện tại trong việc hỗ trợ người dùng xem các video, nhóm sinh viên tiếp tục phát triển khả năng tương tác bằng cách tích hợp các thiết bị và sensor hiện đại vào hệ thống. Định hướng của nhóm là cho phép người

dùng có thể xem trên các mặt phẳng khác như cửa sổ, bàn, tường, ... thay vì màn hình LCD như hiện tại. Bên cạnh đó, sử dụng thêm các camera đặc biệt như Intel Realsense, PrimeSense hay Kinect để hỗ trợ giao tiếp thông qua cử chỉ tay. Từ các cải tiến đó, nhóm hi vọng mang lại trải nghiệm thú vị nhất cho người dùng.

## Tài liệu tham khảo

- [1] N. H. Barnouti, S. S. M. Al-Dabbagh and W. E. Matti, "Face Recognition: A Literature Review," in *International Journal of Applied Information Systems 2016 (IJAIS 2016)*, New York, 2016.
- [2] O. M. Parkhi, A. Vedaldi and A. Zisserman, "Deep Face Recognition," in *British Machine Computer Vision 2015 (BMVC 2015)*, Swansea, 2015.
- [3] Y. Taigman, M. Yang, M. Ranzato and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in *Computer Vision and Pattern Recognition 2014 (CVPR 2014)*, Columbus, 2014.
- [4] Y. Sun, D. Liang, X. Wang and X. Tang, "DeepID3: Face Recognition with Very Deep Neural Networks," in *CoRR, abs/1502.00873*, 2015.
- [5] T. PHAN-DUONG and M.-P. NGUYEN, "Chứng thực với thiết bị di động cho môi trường tương tác thông minh," Hochiminh, 2015.
- [6] H. Li, Z. Lin, X. Shen, J. Brandt and G. Hua, "A Convolutional Neural Network Cascade for Face Detection," in *CVPR*, 2015.
- [7] S. Yang, P. Luo, C. C. Loy and X. Tang, "From Facial Parts Responses to Face Detection A Deep Learning Approach," in *ICCV*, 2015.
- [8] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *CoRR abs/1409.1556*, 2014.
- [9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going deeper with convolutions," *arXiv:1409.4842*, 2014.

- [10] G. B. Huang, M. Ramesh, T. Berg and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," in *Technical Report 07-49, University of Massachusetts*, Amherst, 2007.
- [11] F. Schroff, D. Kalenichenko and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proc. CVPR*, 2015.
- [12] L. Wolf, T. Hassner and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Proc. CVPR*, 2011.
- [13] Y. Sun, X. Wang and X. Tang, "Sparsifying Neural Network Connections for Face Recognition," in *CVPR*, 2016.
- [14] I. Masi, S. Rawls, G. Medioni and P. Natarajan, "Pose-Aware Face Recognition in the Wild," in *CVPR*, 2016.
- [15] B. F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, M. Burge and A. K. Jain, "Pushing the frontiers of unconstrained face detection and recognition: IARPA Janus Benchmark-A," in *CVPR*, 2015.
- [16] Y. Wen, Z. Li and Y. Qiao, "Latent Factor Guided Convolutional Neural Networks for Age-Invariant Face Recognition," in *CVPR*, 2016.
- [17] K. R. Jr and T. Tesafaye, "Morph: A longitudinal image database of normal adult age-progression," in *FG*, 2006.
- [18] C. Xiong, X. Zhao, D. Tang, K. Jayashree, S. Yan and T.-K. Kim, "Conditional Convolutional Neural Network for Modality-aware Face Recognition," in *ICCV*, 2015.
- [19] R. Gross, I. Matthews, J. F. Cohn, T. Kanade and S. Baker, "Multi-PIE," in *Proceedings of the Eighth IEEE International Conference on Automatic Face and Gesture Recognition*, 2008.

- [20] D. K. Pal, F. Juefei-Xu and M. Savvides, "Discriminative Invariant Kernel Features: A Bells-and-Whistles-Free Approach to Unsupervised Face Recognition and Pose Estimation," in *CVPR*, 2016.
- [21] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu and A. C. Berg, "SSD: Single Shot MultiBox Detector," in *ECCV*, 2016.
- [22] Girshick, R., "Fast R-CNN," in *ICCV*, 2015.
- [23] Ren, S., He, K., Girshick, R., Sun, J, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *NIPS*, 2015.
- [24] Redmon, J., Divvala, S., Girshick, R., Farhadi, A., "You only look once: Unified, real-time object detection," in *CVPR*, 2016.
- [25] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet classification with deep convolutional," in *Conference on Neural Information Processing Systems NIPS 2012*, 2012.
- [26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations ICLR 2015*, 2015.
- [27] O. M. Parkhi, K. Simonyan, A. Vedaldi and A. Zisserman, "A compact and discriminative face track descriptor," in *Proc. CVPR*, 2014.
- [28] Y. Taigman, M. Yang, M. Ranzato and L. Wolf, "Web-scale training for face identification," in *Proc. CVPR*, 2015.
- [29] H.-W. Ng and S. Winkler, "A data-driven approach to cleaning large face datasets," in *Proc. IEEE International Conference on Image Processing*, Paris, 2014.

- [30] M. Grgic, K. Delac and S. Grgic, "SCface - surveillance cameras face database," *Multimedia Tools and Applications Journal*, vol. 51, pp. 863-879, 2011.
- [31] D. Yi, Z. Lei, S. Liao and S. Z. Li, "Learning Face Representation from Scratch," in *arXiv preprint arXiv:1411.7923.*, 2014.
- [32] S. Milborrow, J. Morkel and F. Nicolls, "The MUCT Landmarked Face Database," in *Pattern Recognition Association of South Africa*, 2010.
- [33] A. Savran, N. Alyüz, H. Dibeklioglu, O. Çeliktutan, B. Gökberk, B. Sankur and L. Akarunh, "Bosphorus Database for 3D Face Analysis," in *Biomedical Innovation and Development Conference*, 2008.
- [34] D. Chen, X. Cao, L. Wang, F. Wen and J. Sun, "Bayesian face revisited: A joint formulatio," in *ECCV*, Springer, 2012.
- [35] Z. Liu, P. Luo, X. Wang and X. Tang, "Deep Learning Face Attributes in the Wild," in *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.
- [36] B.-C. Chen, C.-S. Chen and W. H. Hsu, "Face Recognition using Cross-Age Reference Coding with Cross-Age Celebrity Dataset," in *IEEE Transactions on Multimedia*, 2015.
- [37] T. Sim, S. Baker and M. Bsat, "The CMU Pose, Illumination, and Expression (PIE) Database," in *International Conference on Automatic Face and Gesture Recognition*, 2002.
- [38] Martin Koestinger, Paul Wohlhart, Peter M. Roth and Horst Bischof, "Annotated Facial Landmarks in the Wild: A Large-scale, Real-world Database for Facial Landmark Localization," in *Proc. First IEEE International Workshop on Benchmarking Facial Image Analysis Technologies*, 2011.

- [39] Jain, Vudit, Learned-Miller, Erik, "FDDB: A Benchmark for Face Detection in Unconstrained Settings," 2010.
- [40] Everingham, M. and Eslami, S. M. A. and Van~Gool, L. and Williams, C. K. I. and Winn, J. and Zisserman, A., "The Pascal Visual Object Classes Challenge: A Retrospective," *International Journal of Computer Vision*, vol. 111, pp. 98-136, 2015.
- [41] G. B. Team, "Tensorflow," Google, [Online]. Available: <https://www.tensorflow.org/>.
- [42] "OpenCV," [Online]. Available: <http://opencv.org>.
- [43] Y. Jia and B. A. Research, "Caffe," [Online]. Available: <http://caffe.berkeleyvision.org/>.
- [44] G. v. Rossum, "Python," [Online]. Available: <https://www.python.org/>.
- [45] F. Wiles, D. Procida, J. Bennett, R. Conley, K. Love and K. W. Alger, "Django," Django Software Foundation, [Online]. Available: <https://www.djangoproject.com/>.
- [46] "Visual Studio," Microsoft, [Online]. Available: <https://www.visualstudio.com/vs/>.
- [47] I. Maes, "Material Skin Github," [Online]. Available: <https://github.com/IgnaceMaes/MaterialSkin>.
- [48] J. Newton-King, "Json.NET," [Online]. Available: <http://www.newtonsoft.com/json>.
- [49] M. Kostinger, P. Wohlhart, P. M. Roth and H. Bischof, "Annotated Facial Landmarks in the Wild A Large-scale, Real-world Database for Facial Landmark Localization," in *ICCV*, 2011.



## **PHỤ LỤC**

## Các công trình đã công bố

C1. Vinh-Tiep NGUYEN, Manh-Tien H.NGUYEN, Quoc-Huu Che, Van-Tu NINH, Tu-Khiem LE, **Thanh-An NGUYEN**, Minh-Triet TRAN, “HCMUS team at the Multimodal Person Discovery in Broadcast TV Task of MediaEval 2016”, MediaEval 2016, Hilversum, The Netherlands, 2016.