

HỆ THỐNG GIÁM SÁT AN TOÀN LÁI XE

DRIVING SAFETY MONITORING SYSTEM

Lê Anh Tuấn¹, Trần Thanh Hùng^{2,*}

TÓM TẮT

Hiện nay tình trạng giao thông nước ta rất phức tạp, với sự tham gia của nhiều loại phương tiện giao thông, trong đó phương tiện ô tô ngày càng phổ biến. Ngoài việc đảm bảo an toàn cho người lái xe và hành khách, ô tô cũng là một trong những nguyên nhân gây ra tai nạn hàng đầu và đều là những vụ tai nạn giao thông nghiêm trọng. Theo nghiên cứu của tổ chức an toàn giao thông quốc gia, phần lớn các vụ tai nạn giao thông ô tô đều xuất phát từ rượu, bia hoặc tài xế buồn ngủ, giảm sự tập trung chú ý trong quá trình lái xe. Xuất phát từ nguyên nhân đó, chúng tôi đã nghiên cứu xây dựng ứng dụng hỗ trợ an toàn lái xe bằng công nghệ AI, nghiên cứu các thuật toán giúp phát hiện tình trạng mất tập trung của người lái xe. Tiếp theo, chúng tôi xây dựng mô hình cho bài toán an toàn lái xe bằng các thuật toán học sâu (Deep Learning) và sử dụng dữ liệu đã thu thập được để huấn luyện mô hình đã xây dựng. Kết quả huấn luyện mạng được sử dụng để xây dựng một hệ thống cảnh báo an toàn lái xe. Đây là một nghiên cứu mang tính thực tiễn, có thể áp dụng vào thực tế hiện nay.

Từ khóa: AI, học máy, học sâu, giám sát tài xế.

ABSTRACT

Currently, the traffic situation in our country is very complicated, with the participation of many types of transport, in which cars are increasingly popular. In addition to ensuring the safety of drivers and passengers, cars are also one of the leading causes of accidents and are all serious traffic accidents. According to a study by the National Traffic Safety Organization, the majority of car accidents are caused by alcohol, beer or drowsy drivers, reducing attention during driving. Stemming from that reason, we have researched and built an application to support driving safety using AI technology, researched algorithms to help detect driver's distraction. Next, we build a model for the driving safety problem using Deep Learning algorithms and use the collected data to train the built model. The results of the network training are used to build a driver safety warning system. This is a practical study that can be applied to current practice.

Keywords: AI, machine learning, deep learning, driver monitoring.

¹Lớp CNTT 03 - Khóa 13, Khoa CNTT, Trường Đại học Công nghiệp Hà Nội

²Khoa CNTT, Trường Đại học Công nghiệp Hà Nội

*Email: hungttbk@gmail.com

1. GIỚI THIỆU

An toàn giao thông luôn là một trong những vấn đề nóng và được cả xã hội quan tâm bởi những hậu quả mà nó để lại rất nặng nề và lâu dài không chỉ đối với gia đình của nạn nhân mà còn ảnh hưởng đến toàn xã hội. Theo số liệu được công bố bởi Phó chủ tịch chuyên trách Ủy ban An toàn giao thông (ATGT) Quốc gia Khuất Việt Hùng [1] tại Hội nghị trực tuyến sơ kết công tác đảm bảo trật tự, an toàn giao thông năm 2021, triển khai phương hướng, nhiệm vụ

năm 2022 sáng 6/1, tính từ ngày 15/12/2020 đến 14/12/2021, toàn quốc xảy ra 11.495 vụ tai nạn giao thông (TNGT), làm chết 5.799 người, bị thương 8.018 người. Mặc dù số vụ tai nạn cũng như số người thiệt mạng đã giảm đáng kể so với năm 2020, nhưng những con số kể trên vẫn cho chúng ta thấy việc đảm bảo trật tự, an toàn giao thông vẫn còn những tồn tại, hạn chế nhất định. Nguyên nhân của tình trạng này phần lớn là do sự thiếu ý thức, kỹ năng và trách nhiệm của một bộ phận không nhỏ những người tham gia giao thông. Cùng với đó là những hạn chế về mức độ ứng dụng công nghệ thông tin, chậm xây dựng và quan tâm đối với vấn đề này.

Trước tình trạng trên, chúng tôi đã xây dựng một ứng dụng cảnh báo mất tập trung cho người lái xe ô tô sử dụng thuật toán học sâu (Deep Learning) và có thể triển khai trên các thiết bị phần cứng đơn giản và đưa vào sử dụng thực tế. Ứng dụng bao gồm việc kết hợp nhiều bài toán trí tuệ nhân tạo và thuật toán xử lý ảnh khác nhau như phát hiện khuôn mặt (Face Detection), phát hiện các đặc điểm trên khuôn mặt (Facial Landmarks Detection), nhận diện buồn ngủ (Drowsiness Detection), ước lượng góc quay của đầu (Head Pose Estimation). Hệ thống sẽ ước lượng và dự đoán được tình trạng vật lý của người lái xe, từ đó đưa ra những cảnh báo trong các trường hợp có thể gây nguy hiểm cho người lái xe cũng như những người tham gia giao thông khác. Bên cạnh đó, bằng việc sử dụng các kiến trúc mạng học sâu tiên tiến, các mô hình, bài toán đều có tốc độ nhanh và có thể tinh chỉnh và triển khai lên các thiết bị phần cứng cơ bản trong tương lai.

Trong bài báo này, chúng tôi đề xuất sử dụng kiến trúc mạng nơ-ron tích chập MobileNetV2 [10] và ShuffleNetV2 [18], một kiến trúc mạng nhẹ và nhanh, có thể chạy trên các thiết bị di động làm tiền đề giải quyết hầu hết các bài toán về phát hiện trong hệ thống. Phần 2 của bài báo sẽ giới thiệu các nghiên cứu liên quan, phần 3 là giới thiệu phương pháp sử dụng, phần 4 là kết quả thực nghiệm. Phần cuối cùng là kết luận cũng như trích dẫn các tài liệu tham khảo.

2. CÁC NGHIÊN CỨU LIÊN QUAN

Hệ thống giám sát lái xe đã và đang được phát triển bởi nhiều nhà nghiên cứu và các công ty từ nhiều nơi trên thế giới bởi tính cấp thiết và ứng dụng cao. Có rất nhiều bài toán được ứng dụng trong các hệ thống này, tuy nhiên hầu hết đều phát triển dựa trên hai bài toán cơ bản là phát hiện đồ vật (Object Detection) và ước lượng tư thế của đồ vật (Pose Estimation).

2.1. Phát hiện đồ vật (Object Detection)

Cho tới thời điểm hiện tại, các thuật toán phát hiện đồ vật vẫn là một trong những thuật toán quan trọng nhất trong phân ngành thị giác máy tính (Computer Vision) bởi nó là tiền đề để giải quyết rất nhiều bài toán thực tế khác nhau. Các thuật toán phát hiện đồ vật thường được chia thành hai nhóm chính là *one-stage object detection* và *two-stage object detection*.

- Two-stage object detection: nhóm các thuật toán (R-CNN [2], Fast-RCNN [3], Faster-RCNN [4],...) chia việc phát hiện đồ vật ra làm 2 bước. Ở bước một, mô hình sẽ xử lý ảnh và trích xuất ra các vùng có khả năng chứa vật thể từ bức ảnh. Ví dụ, với Faster-RCNN [4] thì trong bước 1 (stage-1), mô hình sẽ đưa ra 1 sub-network gọi là RPN (Region Proposal Network) với nhiệm vụ trích xuất các vùng trên ảnh có khả năng chứa đối tượng dựa vào các anchor. Sau khi đã thu được các vùng đặc trưng từ RPN, mô hình Faster-RCNN sẽ thực hiện tiếp bước 2 là phân loại đối tượng và xác định vị trí.

- One-stage object detection: là các thuật toán (SSD [5], YOLO [6],...) không có phần trích chọn các vùng đặc trưng như RPN của Faster-RCNN. Các mô hình thuộc nhóm one-stage object detection này coi việc định vị đối tượng (object localization) như một bài toán hồi quy (regression problem) với 4 tọa độ offset, ví dụ như x, y, w, h , và cũng dựa trên các pre-define box gọi là anchor để làm việc đó. Bằng việc chỉ chạy một thuật toán một lần duy nhất, các thuật toán thuộc nhóm này thường đạt tốc độ nhanh hơn tuy nhiên độ chính xác của mô hình thường kém hơn so với các thuật toán two-stage object detection.

Mặc dù vậy, việc các thiết bị di động đang ngày càng trở nên phổ biến đã thúc đẩy các nhà nghiên cứu đi sâu tìm hiểu và cải thiện các thuật toán phát hiện đồ vật thuộc nhóm one-stage object detection để từ đó ứng dụng vào các bài toán thực tế có thể chạy được trên các thiết bị có phần cứng cơ bản. Trong bài báo này, chúng tôi sẽ sử dụng thuật toán YOLO5Face [8], được ứng dụng từ thuật toán YOLOv5 [7] cho bài toán phát hiện khuôn mặt. Thuật toán YOLOv5 [7] là phiên bản cải tiến thứ năm của họ các thuật toán phát hiện đồ vật YOLO và đã được W. Tan và đồng nghiệp [8] ứng dụng vào bài toán phát hiện khuôn mặt thông qua việc coi khuôn mặt là một đối tượng và xây dựng bài toán phát hiện khuôn mặt tương tự như một bài toán phát hiện đồ vật. Trong nghiên cứu của mình, họ đã đưa ra các mức độ mô hình với số lượng tham số khác nhau nhằm phục vụ cho các ứng dụng khác nhau. Trong bài nghiên cứu của chúng tôi, nhằm đảm bảo yêu cầu ứng dụng có thể triển khai trên các thiết bị phần cứng đơn giản và chạy với tốc độ cao, chúng tôi lựa chọn mô hình nhẹ nhất là YOLOv5n0.5. Với việc chỉ có hơn 400 nghìn tham số, mô hình YOLOv5n0.5 đảm bảo được tính nhẹ và chạy với tốc độ cao mà chúng tôi hướng đến, trong khi vẫn đạt được độ chính xác cao với 90.76% mAP (mean Average Precision) khi chạy trên bộ dữ liệu WiderFace [9].

2.2. Ước lượng tư thế của đồ vật (Pose Estimation)

Bên cạnh bài toán phát hiện, bài toán ước lượng tư thế (Pose Estimation) cũng là vấn đề khá phổ biến trong lĩnh vực xử lý ảnh khi các ứng dụng thực tế yêu cầu chúng ta xác định vị trí và hướng của một đồ vật trong không gian thực. Các bài toán cụ thể trong thực tế hầu hết sẽ liên quan đến việc xác định tư thế của con người hay cơ thể người. Trong nghiên cứu này, để phù hợp với ứng dụng chúng tôi hướng tới là giám sát người lái xe, chúng tôi đã thực hiện nghiên cứu bài toán ước lượng tư thế quay của đầu (Head Pose Estimation).

Bài toán ước lượng tư thế quay của đầu đã được nghiên cứu từ hơn 20 năm về trước. Trong những năm trở lại đây đã có những nghiên cứu mang tính đột phá về bài toán này khi ứng dụng mạng nơ-ron vào việc giải quyết bài toán này. Các nghiên cứu có thể kể đến như HyperFace [13] sử dụng một mạng nơ-ron tích chập duy nhất để dự đoán các đặc điểm khác nhau của con người như phát hiện khuôn mặt, định danh giới tính, tìm các điểm chính trên khuôn mặt và ước lượng tư thế quay của đầu. Bên cạnh đó, Chang và đồng nghiệp [19] sử dụng một mô hình mạng nơ-ron hồi quy đơn giản sử dụng các điểm chính của khuôn mặt để phát hiện tư thế 3D của đầu.

Trong bài nghiên cứu này, chúng tôi ứng dụng mô hình HopeNet [11] để xây dựng một mô hình kết hợp việc huấn luyện hồi quy và phân loại để có thể phát hiện được tư thế của đầu trong không gian thực. Với việc sử dụng ResNet50 là mô-đun gốc, phương pháp của N. Ruiz và đồng nghiệp [11] đạt trung bình 6.155 MAE (Mean Absolute Error) trên cả ba góc xoay Yaw, Pitch, Roll. Tuy nhiên, để mô hình nhẹ hơn và đạt tốc độ cao hơn, chúng tôi đã sử dụng MobileNetV2 để làm mô-đun gốc (backbone) cho mô hình này. Chi tiết sẽ được trình bày ở phần III.C.

3. PHƯƠNG PHÁP

3.1. Phát hiện khuôn mặt và đặc điểm khuôn mặt

Bài toán phát hiện khuôn mặt là tiền đề cho các bài toán tiếp theo bởi các thuật toán nhận diện buồn ngủ hay ước lượng tư thế của đầu đều nhận vào khuôn mặt và các đặc điểm trên khuôn mặt để đưa ra dự đoán.

Chúng tôi đã ứng dụng lại những kỹ thuật được đưa ra trong bài báo về YOLO5Face [8] như sử dụng ShuffleNetV2 [18], một kiến trúc mạng nhẹ có tốc độ cao, để làm backbone và xây dựng mô hình phát hiện từ đó hay sử dụng hàm mất mát WingLoss nhạy cảm hơn với các lỗi nhỏ, trừng phạt sự chênh lệch giữa dự đoán của mô hình và giá trị thật, từ đó giúp mô hình hội tụ tốt hơn và dự đoán tốt hơn. Bên cạnh đó, nhóm nghiên cứu còn đưa ra một phiên bản mô hình nhỏ hơn với tên gọi là YOLOV5n0.5. Với việc giảm một nửa số node ở tất cả các lớp của mô hình, mô hình YOLOV5n0.5 chỉ chứa gần 450k tham số, nhỏ bằng ¼ so với mô hình YOLOV5n của bài nghiên cứu gốc, từ đó gia tăng đáng kể tốc độ chạy của mô hình. Đây cũng sẽ là mô hình chúng tôi ứng dụng trong nghiên cứu lần này cho bài toán phát hiện khuôn mặt.

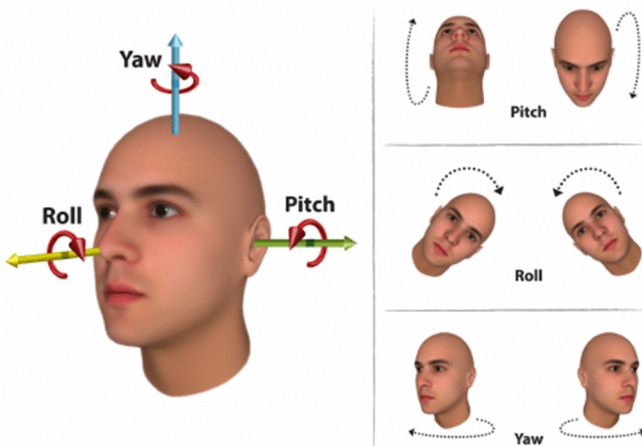
3.2. Nhận diện buồn ngủ

Dựa trên đặc tính của mô hình MobileNetV2 là một mạng nơ-ron tích chập với số lượng tham số thấp và kích thước nhỏ, có khả năng chạy được trên các thiết bị phần cứng cơ bản, chúng tôi đã lựa chọn kiến trúc mạng này làm cơ sở để huấn luyện mô hình nhận diện buồn ngủ. Bản chất của mô hình nhận diện buồn ngủ là một bài toán phân loại với việc mô hình nhận vào hình ảnh của con mắt (mắt trái hoặc mắt phải), sau quá trình tính toán mô hình sẽ đưa ra dự đoán rằng đó là đôi mắt đang nhắm hay đang mở. Chính vì vậy, chúng tôi đã áp dụng phương pháp học chuyển giao (Transfer Learning) bằng việc sử dụng các tham số của mô hình MobileNetV2 đã được huấn luyện trên bộ dữ liệu ImageNet. Việc này giúp cho quá trình huấn luyện mang nhanh hơn cũng như cho ra mô hình tốt hơn.

Bên cạnh đó, bởi ảnh đầu vào là hình ảnh mắt đã được cắt sát và bài toán phân loại cũng chỉ có 2 lớp ("mở mắt" hoặc "nhắm mắt") nên chúng tôi đã thực hiện bỏ bớt $\frac{1}{2}$ số lượng lớp trong kiến trúc mô hình MobileNetV2 để cho ra một mô hình nhẹ hơn với chỉ hơn 400k tham số.

3.3. Phát hiện mắt tập trung

Tương tự như các nghiên cứu đi trước, tư thế của đầu được biểu diễn dưới dạng các góc Euler (Yaw, Pitch, Roll) và bài toán ước lượng tư thế của đầu tập trung vào việc ước lượng được các góc Euler này để từ đó xác định được vị trí và tư thế của đầu trong không gian 3 chiều.



Hình 1. Mô hình biểu thị góc xoay của đầu trong không gian 3 chiều

Tương tự như bài toán nhận diện buồn ngủ, ở bài toán này chúng tôi cũng sử dụng mô hình MobileNetV2 làm backbone bởi tốc độ cao và khả năng triển khai trên các thiết bị di động một cách hiệu quả. Sau khi đi qua backbone, các đặc tính được trích xuất ra sẽ đi qua 3 nhánh khác nhau và được dự đoán là các góc Yaw, Pitch, Roll.

Mô hình này kết hợp huấn luyện hồi quy và phân loại với nhau, cho nên hàm mất mát cần đảm bảo trừng phạt các sai sót theo cả 2 hướng. Vì vậy, hàm mất mát MultiLoss đã được giới thiệu trong bài báo của HopeNet [11]. Việc sử dụng hàm loss trừng phạt theo cả 2 hướng đã giúp mô hình đạt được hiệu quả tốt hơn và dự đoán chính xác hơn.

4. THỰC NGHIỆM

4.1. Cơ sở dữ liệu

Hiện nay có rất nhiều bộ dữ liệu về huấn luyện các tác vụ khác nhau liên quan đến mặt người. Đối với bài toán phát hiện khuôn mặt có thể kể đến các bộ WiderFace [9], AFLW2000 [35] hay các bộ BIWI [13] hay 300W-LP [14], 300W-LPA [15].

Bởi khuôn mặt được dự đoán ra từ mô hình phát hiện khuôn mặt sẽ được đưa vào mô hình ước lượng tư thế của đầu nên cả hai mô hình này chúng tôi đều huấn luyện với bộ dữ liệu 300W-LP [14] và 300W-LPA [15]. Tuy nhiên, để phù hợp với tình hình dịch bệnh hiện tại khi mà việc đeo khẩu trang đã trở nên thân thuộc trên khuôn mặt, chúng tôi đã sử dụng các hàm tính toán để thêm khẩu trang vào các bộ dữ liệu trên. Đối với bài toán phát hiện khuôn mặt, chúng tôi sử dụng thêm bộ WiderFace để tăng tính khái quát và sự thực tế trong các bức ảnh.



Hình 2. Một số hình ảnh của trong bộ dữ liệu 300W-LPA trước và sau khi được thêm khẩu trang

4.2. Huấn luyện

Cả ba mô hình đều được huấn luyện trên Google Colab và thư viện học sâu mã nguồn mở Pytorch cùng với các thư viện tính toán liên quan.

Đối với mô hình phát hiện khuôn mặt, kích thước ảnh đầu vào là 320x320, tốc độ học ban đầu là 0,003 và sẽ giảm đi 0.8 lần sau mỗi 20 chu kỳ. Mô hình được huấn luyện trên 300 chu kỳ. Hàm tối ưu SGD (Stochastic Gradient Descent) được sử dụng với tham số quán tính (momentum) là 0,8 được sử dụng ở 3 chu kỳ khởi động và được tăng lên 0,937 ở những chu kỳ sau.

Đối với mô hình phát hiện mắt tập trung, ảnh đầu vào có kích thước là 224x224, tốc độ học ban đầu là 0,001 và số chu kỳ huấn luyện là 50 chu kỳ. Hàm tối ưu Adam được sử dụng.

Đối với mô hình ước lượng tư thế của đầu, ảnh đầu vào có kích thước là 224x224, tốc độ học ban đầu là 0,001 và sẽ giảm đi 0,8 sau mỗi 10 chu kỳ. Mô hình được huấn luyện trên 100 chu kỳ. Hàm tối ưu là hàm Adam được sử dụng.

4.3. Đánh giá trên tập thực nghiệm

Mặc dù dữ liệu khuôn mặt cho tác vụ phát hiện khuôn mặt và ước lượng tư thế của đầu có thêm khẩu trang gây khó khăn cho quá trình huấn luyện, nhưng các mô hình của chúng tôi đều đạt những kết quả sát với kết quả được công bố trong các nghiên cứu gốc. Bên cạnh đó, tốc độ dự đoán thực nghiệm vẫn đảm bảo để có thể triển khai các bài toán này lên thiết bị di động.

Mô hình phát hiện khuôn mặt đạt 98.06% mAP trên tập dữ liệu wider_face cùng một số ảnh thêm khẩu trang, cao hơn so với các phương pháp trước đó như RetinaFace [16] (94,92%) hay DSFD [17] (94,29%).

Mô hình dự đoán buồn ngủ đạt 99,97% độ chính xác khi chạy trên tập dữ liệu thử nghiệm và đạt 20,1 fps khi chạy trên video thử nghiệm.

Mô hình ước lượng tư thế của đầu đạt 5,5907 MAE (mean square error) trung bình trên cả 3 góc yaw, pitch, roll khi chạy thử nghiệm trên bộ dữ liệu AFLW2000 [35], thấp hơn so với nghiên cứu gốc (6,155 MAE) mặc dù huấn luyện với dữ liệu có khẩu trang.

4.4. Triển khai hệ thống

Sau khi đã hoàn thiện việc huấn luyện và thử nghiệm các mô hình của các bài toán con, chúng tôi sẽ thực hiện ghép các bài toán vào hệ thống an toàn lái xe hoàn chỉnh. Các phần cứng cần có sẽ là một bộ CPU cơ bản (2 - 4 nhân) có khả năng xử lý tính toán và một camera HD. Camera sẽ nhận vào hình ảnh trực tiếp của người dùng, sau đó sẽ đưa qua các bước tiền xử lý (chuẩn hóa kênh màu, chuẩn hóa kích thước) rồi đưa vào mô hình phát hiện khuôn mặt. Mô hình này sẽ phân tích và đưa ra ảnh cắt sát khuôn mặt cùng tọa độ của 2 mắt.

Từ tọa độ của 2 mắt, chúng tôi sẽ cắt ra hình ảnh của 2 mắt để đưa vào mô hình dự đoán buồn ngủ. Khi mô hình dự đoán buồn ngủ đưa ra dự đoán là mắt của người dùng đang nhắm, bộ đếm sẽ bắt đầu tính thời gian. Khi thời gian người dùng nhắm mắt quá 2 giây, hệ thống sẽ phát hiện người dùng đang buồn ngủ và đưa ra cảnh báo âm thanh.

Hình ảnh khuôn mặt được phát hiện sẽ được đưa vào mô hình ước lượng tư thế đầu và cho ra giá trị của 3 góc xoay Yaw, Pitch, Roll. Khi giá trị Yaw lớn hơn 12° hoặc nhỏ hơn -12°, hệ thống sẽ nhận biết rằng người dùng đang không tập trung nhìn thẳng và sẽ đưa ra cảnh báo âm thanh. Với giá trị Pitch cũng sẽ tương tự bởi khi giá trị này hơn 15° hoặc nhỏ hơn -15° tức người dùng đang ngửa đầu lên hoặc cúi xuống mà không tập trung lái xe.

5. KẾT LUẬN

Với bài nghiên cứu này, chúng tôi đã đưa ra một phương pháp kết hợp 3 bài toán con cơ bản để xây dựng thành một hệ thống an toàn lái xe thông minh áp dụng các kiến thức về trí tuệ nhân tạo và học sâu. Đây sẽ là cơ sở để các nghiên cứu về an toàn giao thông nói chung có thể được phát triển trong tương lai với những cải tiến về độ chính xác và tốc độ.

TÀI LIỆU THAM KHẢO

- [1]. Báo điện tử Đảng Cộng sản Việt Nam. Năm 2021 có 55 tỉnh, thành có số người chết do tai nạn giao thông giảm.
- [2]. R. Girshick, J. Donahue, T. Darrell, J. Malik, U. Berkeley. Rich feature hierarchies for accurate object detection and semantic segmentation Tech report (v5). <https://arxiv.org/pdf/1311.2524.pdf>.
- [3]. R. Girshick. Fast R-CNN. <https://arxiv.org/pdf/1504.08083.pdf>.
- [4]. S. Ren, K. He, R. Girshick, J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Network. <https://arxiv.org/pdf/1506.01497.pdf>
- [5]. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, A.C. Berg. SSD: Single Shot MultiBox Detector. <https://arxiv.org/pdf/1512.02325.pdf>
- [6]. J. Redmon, S. Divvala, R. Girshick, A. Farhadi. You only look once: Unified, Real-Time Object Detection. <https://arxiv.org/pdf/1506.02640.pdf>
- [7]. Yolov5. <https://github.com/ultralytics/yolov5>.
- [8]. D. Qi, W. Tan, Q. Yao, J. Liu. YOLO5Face: Why Reinventing a Face Detector. <https://arxiv.org/pdf/2105.12931.pdf>
- [9]. S. Yang, P. Luo, C. C. Loy, X. Tang. Wider face: A face detection benchmark, <http://shuoyang1213.me/WIDERFACE/index.html>.
- [10]. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L. Chen. MobileNetV2: Inverted Residuals and Linear Bottlenecks. <https://arxiv.org/pdf/1801.04381.pdf>
- [11]. N. Ruiz, E. Chong, J.M. Rehg. Fine-Grained Head Pose Estimation Without Keypoints. <https://arxiv.org/pdf/1710.00925.pdf>
- [12]. Y. Zhou, J. Gregson. WHENet: Real-time Fine-Grained Estimation for Wide Range Head Pose. <https://arxiv.org/pdf/2005.10353v2.pdf>
- [13]. G. Fanelli, M. Dantone, J. Gall, A. Fossati, L. Van Gool, 2013. Random forests for real time 3d face analysis. Int. J. Comput.Vision, 101(3):437–458, February.
- [14]. X. Zhu, Z. Lei, X. Liu, H. Shi, S. Z. Li, 2016. Face alignment across large poses: A 3d solution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 146–155.
- [15]. Prof. Hsu. 300W-LPA Database. <https://sites.google.com/view/300w-lpa-database>.
- [16]. J. Deng, J. Guo, Y. Zhou, J. Yu, I. Kotsia, S. Zafeiriou, 2020. Retinaface: Single-stage dense face localisation in the wild. CVPR.
- [17]. J. Li, Y. Wang, C. Wang, Y. Tai, J. Qian, J. Yang, C. Wang, J. Li, F. Huang, 2018. Dsfd: Dual shot face detector. ArXiv preprint 1810.102207.
- [18]. N. Ma, X. Zhang, H. Zheng, J. Sun. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. <https://arxiv.org/pdf/1807.11164.pdf>
- [19]. F.-J. Chang, A. T. Tran, T. Hassner, I. Masi, R. Nevatia, G. Medioni. 2017. Faceposenet: Making a case for landmark-free face alignment. In Computer Vision Workshop (ICCVW), IEEE International Conference on, pages 1599–1608.