**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**🙠🙣🟍🙡🙢**

****

**BÁO CÁO ĐỒ ÁN MÔN MÁY HỌC**

**NHẬN DIỆN ĐEO KHẨU TRANG ĐÚNG CÁCH**

***Lớp:*** CS114.L21

***Nhóm thực hiện:*** Nhóm 11

***Thành viên nhóm:***

|  |  |
| --- | --- |
| *Hoàng Xuân Vũ* | *19522531* |
| *Nguyễn Đạt Huy Hoàng* | *19521536* |
| *Nguyễn Văn Tài* | *19522154* |

**TP. HỒ CHÍ MINH - 07/2021**

**LỜI CẢM ƠN**

Sau quá trình học tập và rèn luyện tai trường Đại học Công Nghệ Thông Tin, chúng em đã được trang bị các kiến thức cơ bản, các kỹ năng thực tế để có thể oàn thành đồ án môn học của mình.

Chúng em xin gửi lời cảm ơn chân thành đến thầy Lê Đình Duy, thầy Phạm Nguyễn Trường An đã tận tâm hướng dẫn, truyền đạt những kiến thức cũng như kinh nghiệm cho chúng em trong suốt thời gian học tập môn Máy học.

Trong quá trình làm đồ án môn học, chúng em không tránh được những sai sót. Mong nhận được sự góp ý cũng như kinh nghiệm quý báu của các các thầy để được hoàn thiện hơn và rút kinh nghiệm cho những môn học sau. Chúng em xin chân thành cảm ơn!

TP. Hồ Chí Minh, tháng 7 năm 2021.

**MỤC LỤC**

[**I.** **Giới thiệu** 4](#_Toc77059348)

[**II.** **Xác định bối cảnh yêu cầu** 4](#_Toc77059349)

[**1.** **Bài toán** 4](#_Toc77059350)

[**2.** **Xác định yêu cầu bài toán** 4](#_Toc77059351)

[**3.** **Thu thập dữ liệu** 5](#_Toc77059352)

[**4.** **Các thao tác với dữ liệu sau khi thu thập** 5](#_Toc77059353)

[**III.** **Khám phá dữ liệu** 6](#_Toc77059354)

[**IV.** **Đề xuất mô hình** 7](#_Toc77059355)

[**1.** **Mô hình sử dụng** 7](#_Toc77059356)

[**2.** **Giới thiệu mô hình YOLO** 7](#_Toc77059357)

[**3.** **Lý do chọn mô hình** 7](#_Toc77059358)

[**4.** **Cấu trúc mạng YOLOv4** 7](#_Toc77059359)

[**4.1.** **Backbone** 7](#_Toc77059360)

[**4.2.** **Neck** 8](#_Toc77059361)

[**4.3.** **Head – Bước nhận dạng** 8](#_Toc77059362)

[**4.4.** **Cấu trúc mạng Darknet53** 9](#_Toc77059363)

[**5.** **Cách mô hình hoạt động** 10](#_Toc77059364)

[**V.** **Huấn luyện mô hình** 12](#_Toc77059365)

[**VI.** **Kết quả thực nghiệm** 14](#_Toc77059366)

[**VII.** **Tinh chỉnh mô hình** 15](#_Toc77059367)

[**VIII.** **Đánh giá mô hình** 15](#_Toc77059368)

[**1.** **Đánh giá dựa trên kết quả thực nghiệm** 15](#_Toc77059369)

[**2.** **Những hạn chế của mô hình** 16](#_Toc77059370)

[**3.** **Ứng dụng vào thực tế** 16](#_Toc77059371)

[**IX.** **Tài liệu tham khảo** 17](#_Toc77059372)

1. **Giới thiệu**

Hiện nay, dịch COVID-19 đang diễn biến phức tạp không chỉ ở nước ta mà còn trên toàn thế giới. Vì vậy, giãn cách xã hội và tránh tụ tập đông người là điều cần phải thực hiện trong mùa dịch này. Tuy nhiên, nhu cầu đi siêu thị mua thực phẩm là điều không thể tránh khỏi. Để giúp việc kiểm soát được tốt hơn tại các siêu thị, nhóm chúng em đề xuất đề tài vì muốn việc theo dõi, giám sát các đối tượng không tuân thủ đúng quy tắc phòng ngừa dịch bệnh giảm được phần nào lo ngại mà không làm lây nhiễm dịch bệnh trong cộng đồng. Điều này cũng sẽ giúp các cơ quan tiết kiệm được thời gian cũng như giảm được khoảng cách tiếp xúc trong việc kiểm soát dịch bệnh.

1. **Xác định bối cảnh yêu cầu**
2. **Bài toán**

Mục tiêu: Xây dựng mô hình phát hiện người không mang khẩu trang, mang không đúng cách, mang đúng cách tại các siêu thị trong mùa dịch COVID.

1. **Xác định yêu cầu bài toán**

- Input: 1 video.

- Output: video tương ứng với input chứa các bounding box và nhãn dự đoán.

\* Các class (lớp):

- Đeo đúng: Khẩu trang ôm sát phần miệng và mũi không để hở.



- Đeo sai: Khẩu trang không ôm sát cả miệng lẫn mũi (chỉ có phần miệng nhưng chừa phân mũi hoặc đeo một bên có một bên không, …)



- Không đeo: Không có khẩu trang trên khuôn mặt.



1. **Thu thập dữ liệu**

- Dữ liệu thu thập từ camera điện thoại, video trong các siêu thị.

- Một số ảnh đi siêu thị trên internet.

- Các video bán hàng thông qua camera ghi lại.

1. **Các thao tác với dữ liệu sau khi thu thập**

- Sau khi thu thập dữ liệu, nhóm đã tiến hành tiền xử lý bằng cách cắt các frame ảnh, lọc và điều chỉnh ảnh sắc nét nhất có thể.

- Sau đó thực hiện gắn nhãn cho từng bức ảnh bằng phần mềm labelImg. Để máy tính có thể dễ dàng xử lý ta sẽ gắn nhãn 0 với những tấm ảnh không đeo khẩu trang, 1 với những tấm ảnh đeo đúng cách và 2 với những tấm ảnh đeo sai như đã mô tả trên.

- Vùng gắn nhãn làm sao cho đủ hết mặt (kể cả khẩu trang và lấy một ít tóc nhất có thể).

- Lưu các chỉ số gắn nhãn là các tọa độ nhãn đã gán bằng file txt kèm với ảnh tương ứng.

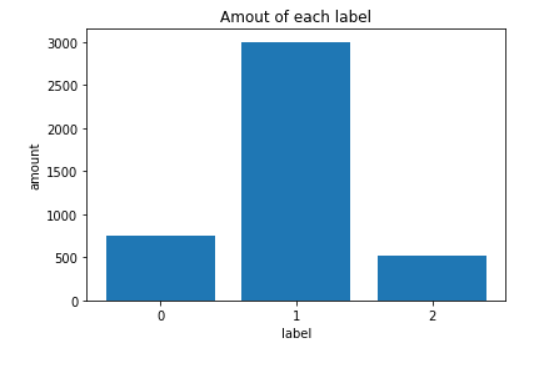
1. **Khám phá dữ liệu**

Sau khi đã thực hiện chọn lọc và thao tác xử lý trên, nhóm đã thu thập được khoảng 4276 tấm ảnh, trong đó:

- Có 749 đối tượng mang nhãn 0

- 3002 đối tượng mang nhãn 1

- 525 đối tượng mang nhãn 2.



***Nhận xét:***

Nhìn chung, dữ liệu lệch tương đối khá nhiều (các đối tượng đeo khẩu trang với các đối tượng không đeo, đeo sai). Và điều này cũng phù hợp với thực tế trong bối cảnh diễn biến phức tạp của dịch bệnh hiện nay.

1. **Đề xuất mô hình**
2. **Mô hình sử dụng**

Mô hình YOLOv4

1. **Giới thiệu mô hình YOLO**

YOLO (You Only Look Once) là một mô hình mạng CNN trong việc phát hiện, nhận dạng, phân loại đối tượng. YOLO được tạo ra từ việc kết hợp giữa các convolutional layers và các connected layers. Trong đó các các convolutional layers sẽ trích xuất ra các feature của ảnh còn full-connected layers sẽ dự đoán ra xác suất và tọa độ của đối tượng.

YOLO có nhiều phiên bản là YOLOv1, YOLOv2, YOLOv3 và YOLOv4. Ngoài ra, còn có YOLOv5 nhưng vẫn chưa phải là phiên bản chính thức được chấp nhận. Hơn nữa, cũng có nhiều tranh cãi về tính hiệu quả của mô hình này. Tính tới thời điểm hiện tại, YOLOv4 là phiên bản tốt nhất sau nhiều lần cải tiến.

1. **Lý do chọn mô hình**

YOLO có nhiều phiên bản nhưng nhóm chọn mô hình YOLOv4 bởi vì độ chính xác cao cũng như thời gian chạy tối ưu. Không những vậy, YOLOv4 không cần có máy phải có cấu hình mạnh như phiên bản YOLOv3. Chính vì vậy mà YOLOv4 là phiên bản tốt nhất cho tới thời điểm hiện tại sau khi qua nhiều lần cải tiến và được áp dụng rộng rãi trong cái bài toán phân loại và nhận diện hiện nay.

1. **Cấu trúc mạng YOLOv4**

**-** Gồm 4 phần:

* Backbone
* Neck
* Dense Prediction
* Sparse Prediction
  1. **Backbone**

- Mạng có tác dụng trích xuất đặc trưng. Trong bài báo gốc của tác giả đã xem xét sử dụng các Backbone: CSPRestNext50, CSPDarknet53, EfficientNet-B3.

- Qua quá trình thử nghiệm tác giả đã chọn CSPDarknet53.

- Cấu trúc CSPDarknet53: Bao gồm CSP (Cross-Stage-Partial connection) và Darknet53.

* 1. **Neck**

Có nhiệm vụ trộn và kết hợp các đặc trưng (Features map) đã học được thông qua quá trình trích xuất đặc trưng (Backbone) và quá trình nhận dạng (YOLOv4 gọi là Dense Prediction).

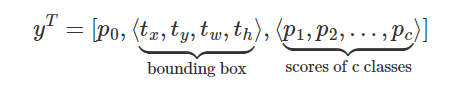
* 1. **Head – Bước nhận dạng**

- YOLOv4 sử dụng phân head là các hộp neo (anchor box) và nhận dạng với ảnh có kích thước khác nhau.

- Các ảnh có kích thước khác nhau sẽ được scale rồi đưa vào mạng. Thông thường thì ảnh sẽ được scale về kích thước 416 x 416 hoặc 608 x 608. Trong đề tài này, chúng em sử dungj kích thước 608 x 608.

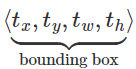
- Để tìm được bounding box cho vật thể YOLO sẽ cần anchor box để làm cơ sở. Anchor box là một bounding box và là cơ sở để xác định bounding box bao quanh vật thể dựa trên các phép dịch tâm và scale kích thước chiều dài, rộng. Mỗi loại anchor box sẽ phù hợp để tìm ra bounding box cho một loại vật thể đặc trưng.

- Ouput của YOLOv4 là một vector bao gồm các thành phần:



Trong đó:

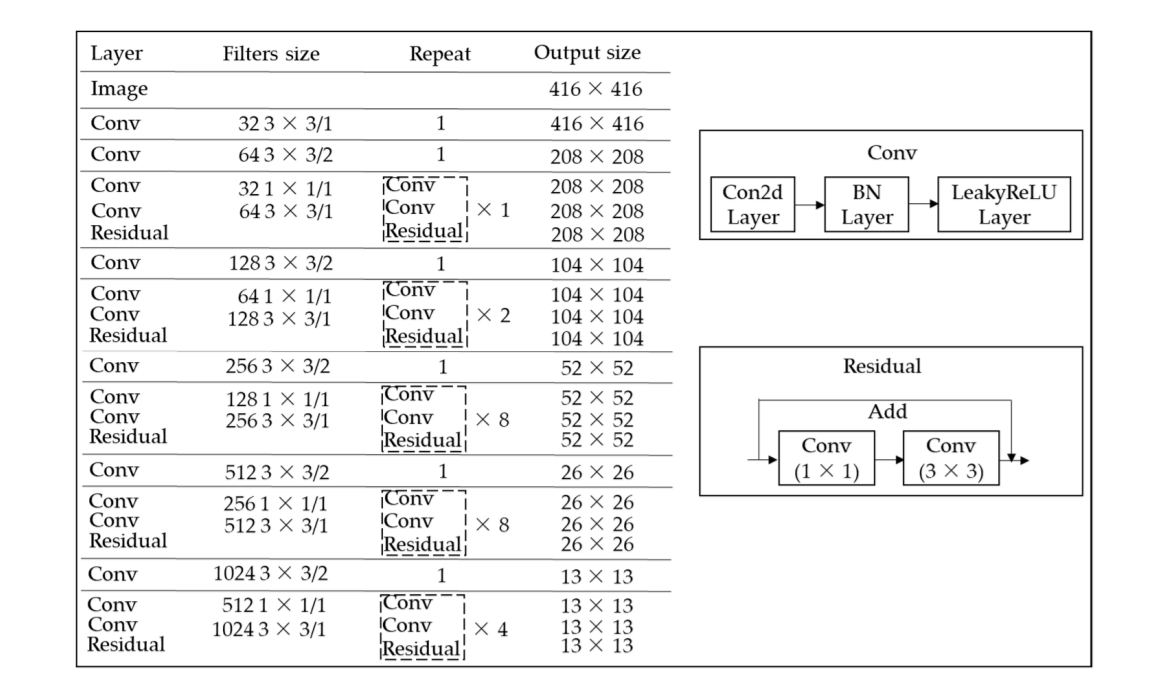
* Po là xác suất dự báo vật thể xuất hiện trong bounding box



là tọa độ tâm và là chiều rộng và chiều dài của bounding box.

* Scores of classes: Là vector phân phối xác suất cúa các class.
* Như vậy, số lượng theo công thức: . Trong đó, n là số lượng class, k là số lượng anchor/cell.
  1. **Cấu trúc mạng Darknet53**

Đây là một trong những cấu trúc đặc trưng của YOLOv4. Mạng này gồm 53 convolutional layers kết nối liên tiếp (bao gồm cả fully connect layer nhưng không gồm residual layer(resnet), mỗi layer được theo sau bởi một batch normalization và một activation Leaky Relu. Một vấn đề xảy ra khi xây dựng mạng CNN với nhiều lớp chập sẽ xảy ra hiện tượng Vanishing Gradient dẫn tới quá trình học tập không tốt. Và thêm residual layer là để giải quyết vấn đề này. Để giảm kích thước của output sau mỗi convolution layer, tác giả down sample bằng các filter với kích thước là 2. Mẹo này có tác dụng giảm thiểu số lượng tham số cho mô hình.



Các bức ảnh đưa vào mô hình sẽ được điều chỉnh kích thước cho phù hợp với mô hình và đưa vào để huấn luyện. Thông thường thì kích thước được điều chỉnh về 416 x 416 hoặc 608 x 608. Trong đồ án này nhóm chúng em sử dụng kích thước đầu vào là 608 x 608. Sau khi đi qua các layer convolutional thì kích thước giảm dần theo cấp số nhân là 2. Cuối cùng ta thu được một feature map có kích thước tương đối nhỏ để dự đoán vật thể trên từng ô của feature map. Tùy vào kích thước đầu vào sẽ tạo ra các feature map tương ứng. Với đầu vào 416 x 416 thì feature map có các kích thước là 13 x 13, 26 x 26 và 52 x 52. Với đầu vào là 608 x 608 sẽ tạo ra feature map là 19 x 19, 38 x 38, 72 x 72.

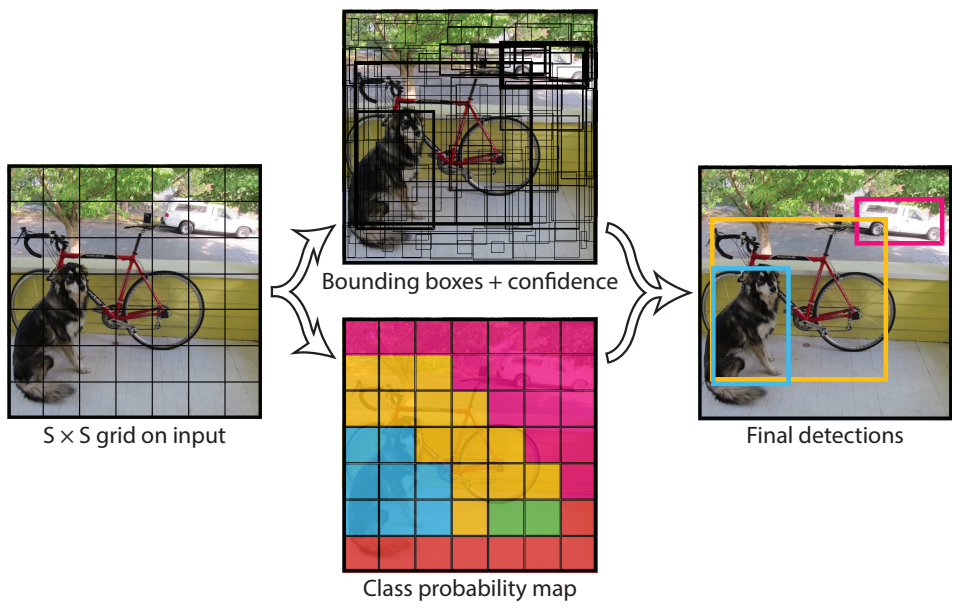
1. **Cách mô hình hoạt động**

- Đầu vào là một ảnh, mô hình sẽ nhận dạng sự xuất hiện của đối tượng trong tấm ảnh sau đó sẽ xác định tọa độ. Ảnh đầu vào được chia thành lưới các ô kích thước S x S gọi là grid cell, S thường là các con số lẻ (3, 5, 7, …) nhưng mặc định là 7. Ứng với mỗi grid cell mô hình sẽ đưa ra dự đoán Bounding Box kèm theo các tham số (x, y, w, h, confidence). Với

- (x, y) là tọa độ tâm

- h, w là chiều cao, chiều rộng

- confidence là độ tự tin của dự đoán.



- Đầu ra mô hình là một ma trận kích thước S x S x (5 x N + M), với N là số lượng Box và M là số class mỗi ô cần dự đoán.

- Mỗi bounding box gồm 5 tham số (x, y, w, h, prediction) với

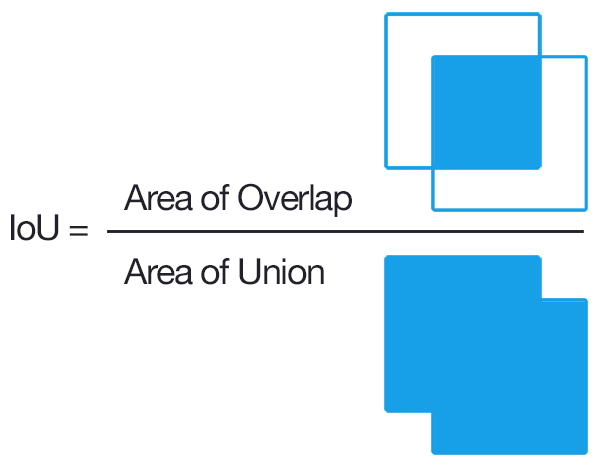
+ (x, y) là tâm tọa độ

+ h, w lần lượt là chiều cao và chiều rộng

+ prediction = pr(object) \*IoU (pred, truth)

\* Hàm tính IoU

- IoU (Intersection ovẻ Union) là độ chính xác của object detector trên tập dữ liệu cụ thể.



- Trong đó,

+ Area of Overlap là phần giao giữa predicted bounding box và ground-truth bouding box.

+ Area of Union là phần hợp giữa predicted bounding box và ground-truth bouding box.

- Nếu IoU > 0.5 thì prediction được coi là tốt.

- Với mỗi IoU, ta có thể so sánh được độ tốt của mô hình vì giá trị IoU phụ thuộc vào Precision-Recall được mô tả như hình trên

1. **Huấn luyện mô hình**

- Sau khi thu thập và khám phá dữ liệu, nhóm đã chia tập dữ liệu thành tập train và test. Trong đó, tập train chiếm 80% còn tập test chiếm 20%.

- Khi huấn luyện mô hình, ngoài các tham số đầu vào như height, width, channel, scale, … trong cấu trúc mô hình sử dụng. Một tham số quan trọng phải kể đến đó chính là tham số learning rate lr, một trong những yếu tố quyết định mô hình có tốt hay không. Nếu điều chỉnh tham số phù hợp thì sẽ cho được mô hình tốt có độ chính xác phù hợp với bài toán đề ra.

- Nhóm đã sử dụng Precision-Recall cũng như mean Average Precision (mAP) để đo độ chính xác của mô hình:

+ Precision: Độ chính xác của mô hình đưa ra (tỉ lệ mô hình đưa ra dự đoán đúng).

+ Recall: Tỉ lệ mô hình nhận diện được.

+ mAP là giá trị trung bình của các AP (AP là giá trị của mối quan hệ precision-recall). Mối quan hệ giữa precision-recall giúp mAP đánh giá được độ chính xác mô hình dự đoán.

- Để đánh giá chúng ta sẽ sửu dụng ngưỡng là threshold.

* Đối tượng nhận dạng đúng với tỷ lệ IoU, IoU > threshold: True Positive (TP).
* Đối tượng nhận dạng sai với tỷ lệ IoU, IoU <= threshold: False Positive (FP).

- Sau khi có được TP và FP chúng ta sử dụng bộ đánh giá precision và recall.



- Giả sử có N ngưỡng để tính precision và recall, với mỗi ngưỡng có cặp giá trị precision và recall tương ứng là Pi và Ri. Thì công thức AP được tính theo như sau:

- Giá trị mAP là trung bình của tất cả các lớp. Ngoài đánh giá mô hình bằng mAP thì nhóm còn đánh giá dựa vào trực quan bằng video demo để kết luận mô hình tốt hay không.

1. **Kết quả thực nghiệm**

Lần đầu tiên nhóm thực hiện, đã chọn tham số learning rate lr = 0.001. Và thu được kết quả như sau:

|  |  |
| --- | --- |
| Số vòng lặp | mAP (%) |
| 1000 | 57.21 |
| 2000 | 76.85 |
| 3000 | 78.89 |
| 4000 | 81.91 |
| 5000 | 81.45 |
| 6000 | 84.7 |

Kết quả thu được còn khá thấp ở lần thực hiện đầu tiên, nên nhóm quyết định cải thiện mô hình bằng cách điều chỉnh tham số lr để xem mô hình có tốt hơn không.

1. **Tinh chỉnh mô hình**

Như đã đề cập ở trên thì mô hình cần được cải thiện nên nhóm đã điều chỉnh tham số lr lần lượt là 0.002 và 0.003 và thấy được sự thay đổi rõ rệt sau khi điều chỉnh tham số. Kết quả thu được có vẻ tốt hơn lần đầu.

- Với learning rate lr = 0.002

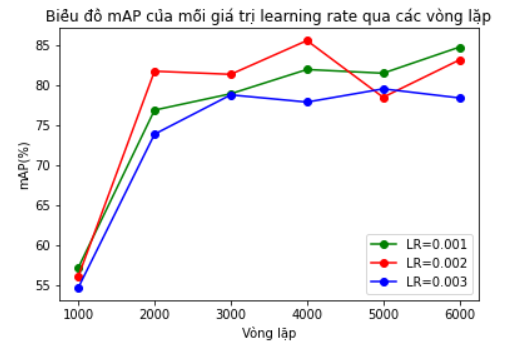
|  |  |
| --- | --- |
| Số vòng lặp | mAP (%) |
| 1000 | 56.12 |
| 2000 | 81.7 |
| 3000 | 81.31 |
| 4000 | 85.52 |
| 5000 | 78.45 |
| 6000 | 83.1 |

- Với learning rate lr = 0.003

|  |  |
| --- | --- |
| Số vòng lặp | mAP (%) |
| 1000 | 54.72 |
| 2000 | 73.85 |
| 3000 | 78.75 |
| 4000 | 77.85 |
| 5000 | 79.5 |
| 6000 | 78.37 |

1. **Đánh giá mô hình**
2. **Đánh giá dựa trên kết quả thực nghiệm**

Sau khi huấn luyện mô hình và thu được kết quả như trên, nhóm thu được kết quả tốt nhất với độ chính xác 85.52% với learning rate = 0.002. Nhìn chung, sau khi điều chỉnh tham số thì lr = 0.002 mô hình dự đoán tốt nhất.

1. **Những hạn chế của mô hình**

Mặc dù độ chính xác của mô hình tương đối cao nhưng bên cạnh đó vẫn có một vài hạn chế mà nhóm vẫn chưa giải quyết được:

- Chỉ nhận diện chính xác oử gần nhưng lại nhận diện sai những người ở phía xa.

- Chất lượng video thấp ảnh hưởng đến độ chính xác.

- Nhận diện nhầm lẫn với các biển quảng cáo mặc dù nơi đó không có người.

- Đa số người đeo khẩu trang y tế màu xanh, xám, trắng … Một số sử dụng khẩu trang khác ít phổ biến lại nhận diện sai hoặc nhận diện không được.

1. **Ứng dụng vào thực tế**

Với mô hình này thì vẫn chưa thực sự áp dụng vào giải quyết triệt để bài toán đưa ra vì dữ liệu để huấn luyện mô hình còn khá thấp cũng như một số hạn chế đã nêu trên. Để có thể sử dụng mô hình, cần phải thu thập thêm nhiều dữ liệu cũng như có những thao tác tiền xử lý tốt hơn để có thể cải thiện tốt hơn.

1. **Tài liệu tham khảo**

[1]<https://phamdinhkhanh.github.io/2020/03/09/DarknetAlgorithm.html?fbclid=IwAR19jha45MHH-49FzTsygJm8lRdJN1zdcy3HemfM8RaWO1CMUpdLNdTPTb8#2-ki%E1%BA%BFn-tr%C3%BAc-m%E1%BA%A1ng-yolo>

[2]<https://aicurious.io/posts/tim-hieu-yolo-cho-phat-hien-vat-tu-v1-den-v5/?fbclid=IwAR2FDNikgcqvg4zz1_Z4agNYyIc5I8MbCUMESSkn44CCtDmdO5eTeXKX8og>

[3]<https://devai.info/2021/02/24/series-yolo-4-tim-hieu-cau-truc-yolov1v2v3-va-v4-phan-2/?fbclid=IwAR3SLyX_IoprDPsFWR9XDHmhYoZJhHWpTxO1LQ4IIsSAtKvFOfaDQOdkQfs>