

# **BÁO CÁO ĐỒ ÁN CUỐI KỲ**

**Lớp: CS2225.CH1501**

**Môn: NHẬN DẠNG THỊ GIÁC VÀ ỨNG DỤNG**

**GV: PGS.TS Lê Đình Duy**

**Trường ĐH Công Nghệ Thông Tin, ĐHQG-HCM**



# HƯỚNG DẪN

**Các nhóm copy các slides này vào phần Bắt đầu báo cáo và Edit. KHÔNG sửa các slides của phần này**

# TÊN ĐỀ TÀI - VIẾT IN HOA

**Nguyễn Văn A**

**Link Github:** <https://github.com/ledduy610/CS2225-VRA>

**Link YouTube**

[https://www.youtube.com/watch?v=5\\_ozB0ImkYA](https://www.youtube.com/watch?v=5_ozB0ImkYA)

# Tóm tắt

- Tên đề tài
- Tóm tắt về đồ án và kết quả đạt được
- Ảnh của các thành viên của nhóm

# Hướng dẫn

- Các nhóm copy slide mẫu về tên đề tài rồi update thông tin của nhóm
- Không sửa đổi các slide mẫu - nên tạo một copy các slide này thành tập tin của chính mình, sau khi hoàn tất thì copy các slide vào file này.
- Lưu ý copy vào cuối file để tránh ghi đè lên các bạn đã nộp trước đó
- Tối đa 15 slides/đồ án

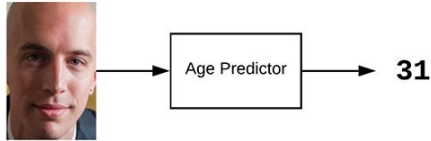
# Mô tả bài toán

- Task: Dự đoán tuổi
- Input: ảnh
- Output: vị trí của các khuôn mặt và tuổi tương ứng
- Minh hoạ (slide kế tiếp)

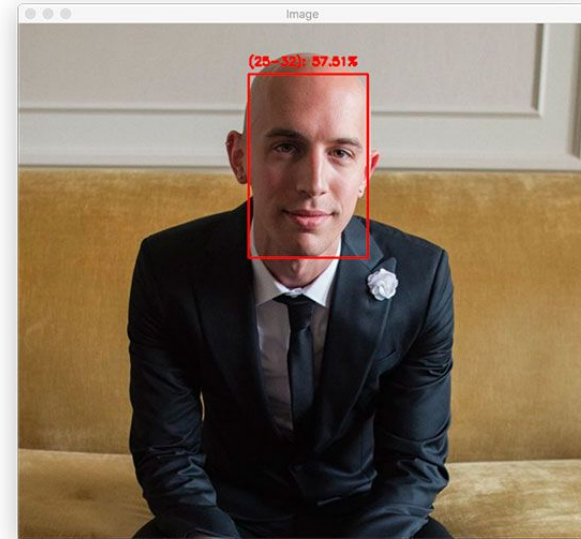
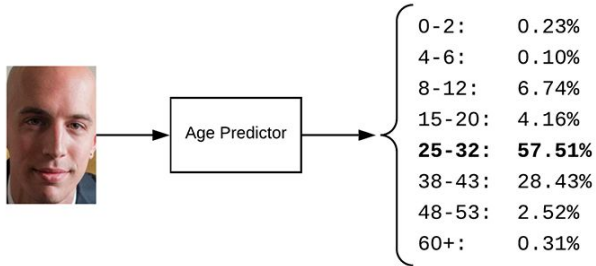
# Mô tả bài toán

- Minh họa

## Age Prediction via Regression



## Age Prediction via Classification



# Loại bài toán ML

- Regression
- Classification
- Object Detection
- Segmentation



# Dữ liệu

- Tổng số mẫu
- Số mẫu Positive
- Số mẫu Negative
- Phân chia Train-Dev-Test
- Cách thu thập
  - Tự thu thập và gán nhãn thủ công
  - Có từ nguồn khác
-

# Phương pháp đánh giá

- MAP
- F1-Score

# **BẮT ĐẦU BÁO CÁO CỦA CÁC NHÓM**

**Các nhóm sử dụng hướng dẫn các nội dung  
trong các slide mẫu ở trên**

# BÁO CÁO ĐỒ ÁN CUỐI KỲ

Lớp: CS2225.CH1501

Môn: NHẬN DẠNG THỊ GIÁC VÀ ỨNG DỤNG

GV: PGS.TS Lê Đình Duy

Trường ĐH Công Nghệ Thông Tin, ĐHQG-HCM



# HƯỚNG DẪN

**Các nhóm copy các slides này vào phần Bắt đầu báo cáo và Edit. KHÔNG sửa các slides của phần này**

# NHẬN DIỆN ĐỐI TƯỢNG QUA HÌNH ẢNH VÀ VIDEO CLIP

Trần Văn San

Link Github:

<https://github.com/CS2225CH2001013/CS2225.CH2001013>

Link YouTube:

- Giới thiệu: [https://youtu.be/W9\\_efYf1DQU](https://youtu.be/W9_efYf1DQU)

- Demo các kết quả:

[https://youtube.com/playlist?list=PLI1c-nvyWcl8TZHrWjAaXO9LYzSu\\_0rJB](https://youtube.com/playlist?list=PLI1c-nvyWcl8TZHrWjAaXO9LYzSu_0rJB)

# Tóm tắt

- **Tên đề tài:** NHẬN DIỆN ĐỐI TƯỢNG TRONG ẢNH VÀ VIDEO
- **Tóm tắt về đề án và kết quả đạt được:** nhóm sử dụng công cụ YOLO 3 để nhận diện các đối tượng trong ảnh, video clip được thu thập qua mạng, tự chụp, tự quay.... Sau khi ứng dụng YOLO 3, kết quả đạt được: có thể nhận diện vị trí các đối tượng với độ chính xác 95%-100% có trong ảnh, video clip đưa vào.

# Tóm tắt

- Ảnh của các thành viên của nhóm:



**Trần Văn San**  
CH2001013



**Vũ Linh**  
CH2004008



**Cao Nguyễn Nam Hiền**  
CH2001005

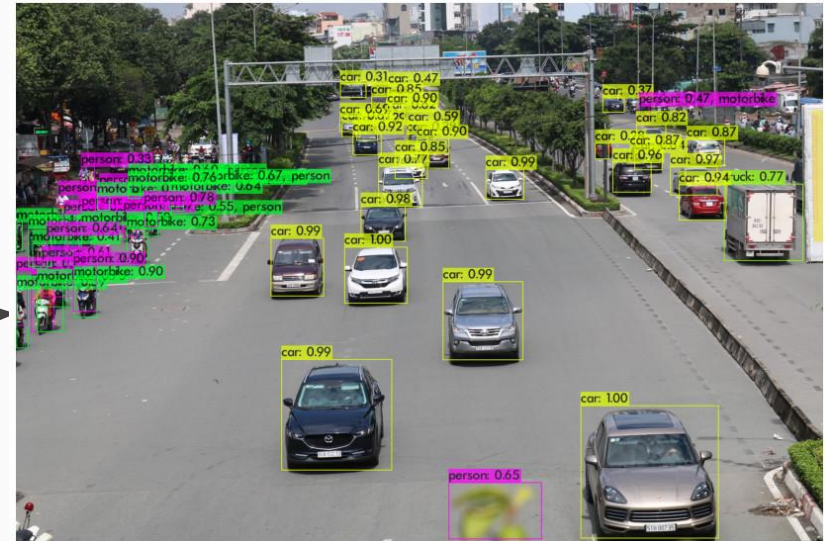


# Mô tả bài toán

- Task: Nhận diện đối tượng qua hình ảnh và video clip
- Input: ảnh, video clip
- Output: phân loại và vị trí các đối tượng có trong ảnh, video clip
- Minh hoạ (slide kế tiếp)

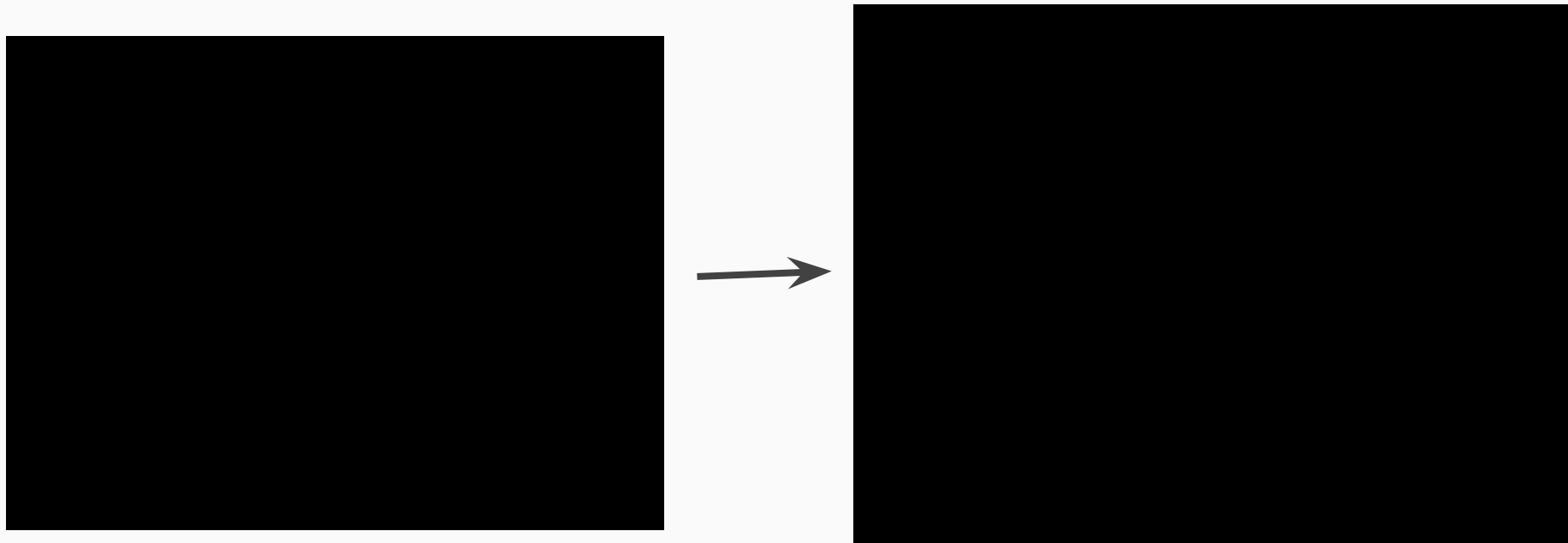
# Mô tả bài toán

- Minh họa nhận diện đối tượng qua ảnh



# Mô tả bài toán

- Minh họa nhận diện đối tượng qua video clip



# Loại bài toán ML

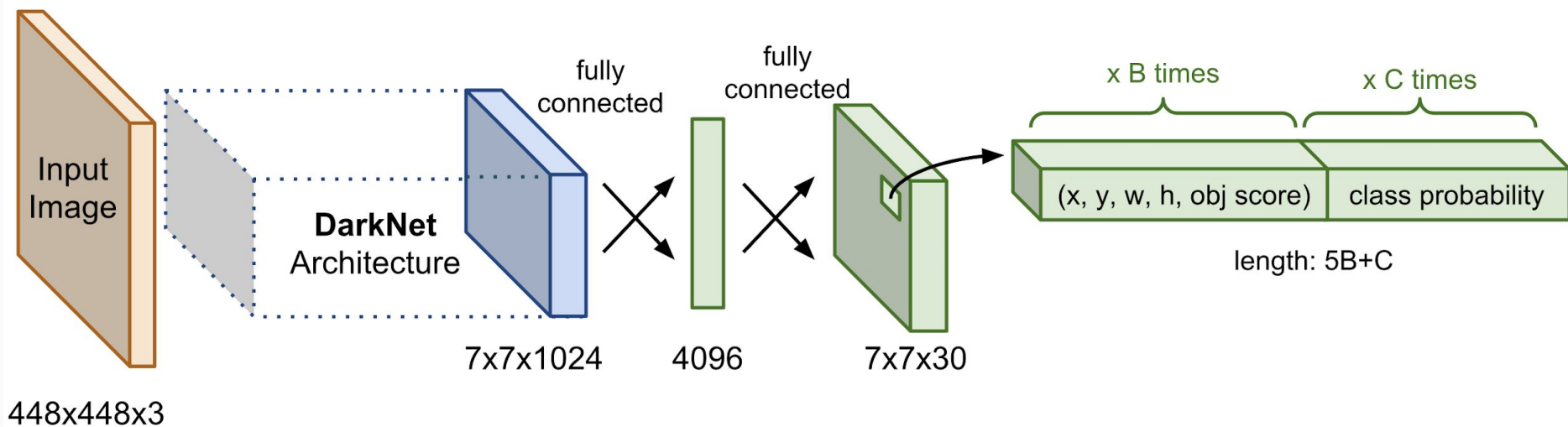
- Object Detection
- Sử dụng: YOLO 3

# Dữ liệu

- Tổng số mẫu: Bộ dữ liệu train có dung lượng: 6,2GB với 20.288 ảnh (nguồn:<http://images.cocodataset.org/zips/test2017.zip>)
- Số mẫu Positive
- Số mẫu Negative
- Phân chia Train-Dev-Test
- Cách thu thập
  - Tự thu thập và gán nhãn thủ công
  - Có từ nguồn khác
-

# Kiến trúc mạng YOLO

Phần đầu Darknet Architecture được gọi là base network có tác dụng trích xuất đặc trưng. Phần sau là những Extra Layers được áp dụng để phát hiện vật thể trên feature map của base network



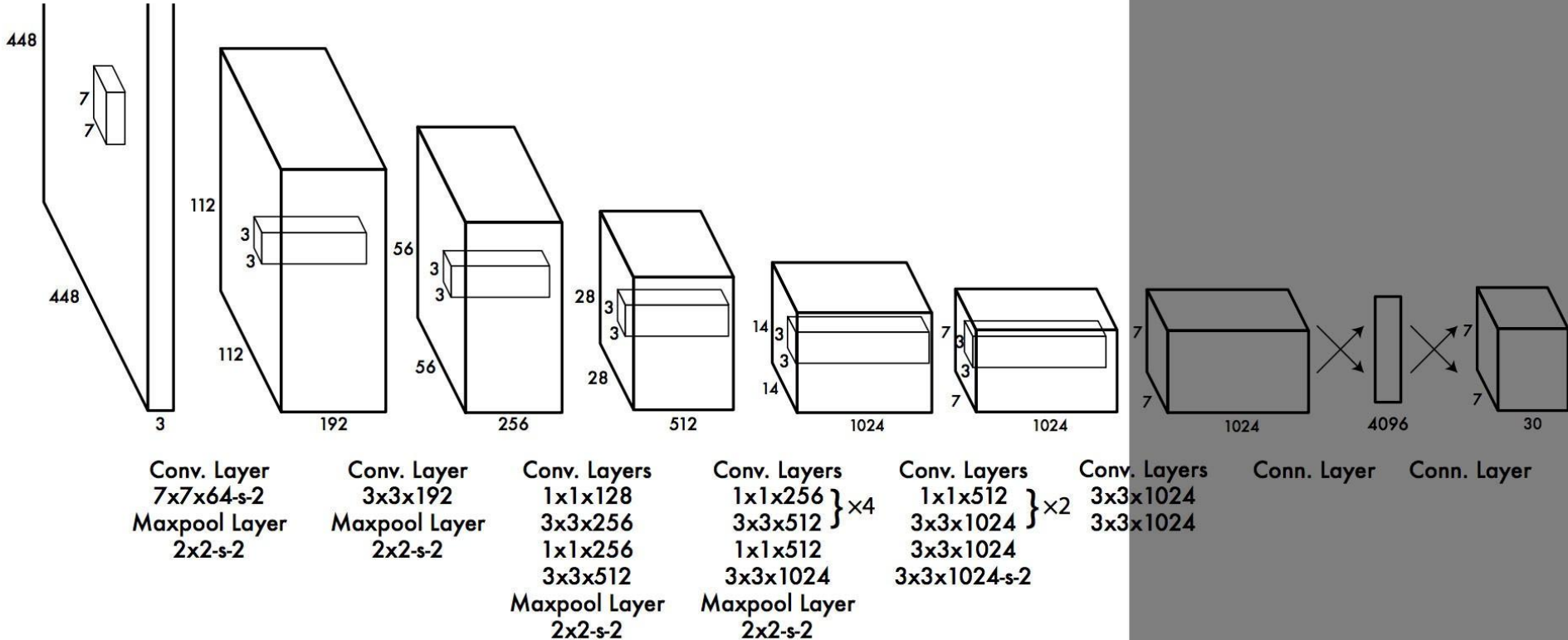
# YOLO version 2

YOLOv2 là phiên bản thứ 2 của YOLO: Có độ chính xác mAP tăng 2%, tỉ số Anchor boxes tăng từ 81% lên 88%, mAP của phân lớp tăng 5%.

YOLOv2 đã thực hiện các cải tiến sau:

- Loại bỏ các lớp fully connected chịu trách nhiệm dự đoán hộp giới hạn
- Di chuyển lớp dự đoán từ cấp độ ô lên cấp độ hộp giới hạn.
- Điều chỉnh lớp convolutional và các filter, loại bỏ một lớp pooling, giảm kích thước ảnh đầu vào.

# Kiến trúc mạng YOLOv2





# YOLO version 3

YOLOv3 là mô hình phát hiện đối tượng một giai đoạn, thời gian thực, được xây dựng trên YOLOv2 với một số cải tiến. Phát hiện các đối tượng nhỏ tốt hơn nhờ phát hiện ở các lớp khác nhau giúp giải quyết được vấn đề phát hiện các vật thể nhỏ, một vấn đề mà YOLOv2 gặp phải. Tuy tốc độ xử lý giảm đi do tăng độ phức tạp của mô hình để cải thiện độ chính xác.

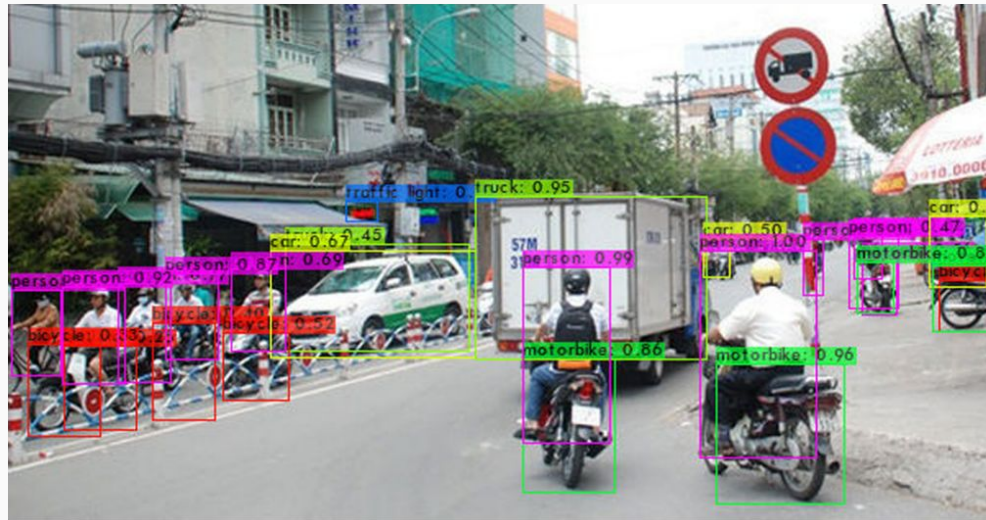
YOLOv3 sử dụng một biến thể của Darknet và thay đổi kích thước để dự đoán:

- Áp dụng filter kích thước  $1 \times 1$  trên toàn bộ features map của ba kích thước khác nhau tại ba địa điểm khác nhau trong mạng.
- Các lớp tăng kích thước được ghép nối với các lớp trước giúp bảo vệ các đặc trưng tốt giúp phát hiện các vật thể nhỏ.
- Giảm kích thước ảnh đầu vào xuống lần lượt là 32, 16 và 8.

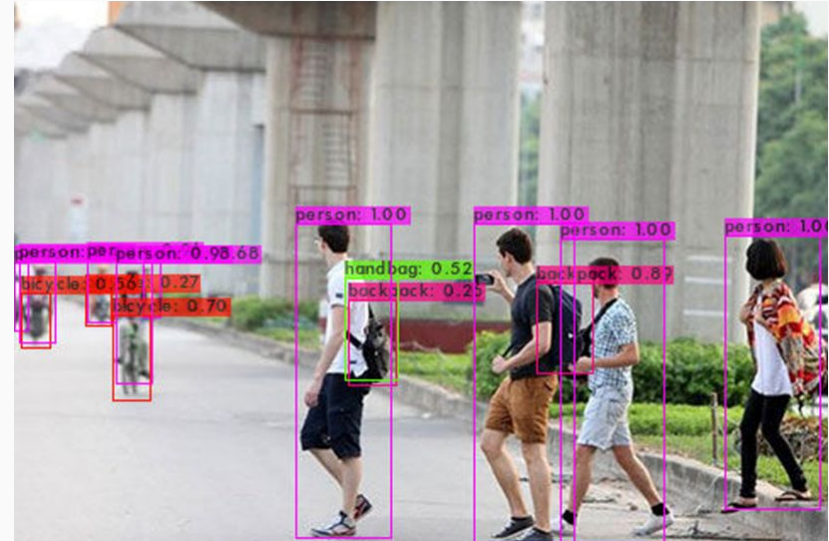
# Kiến trúc mạng YOLOv3

Type	Filters	Size	Output
Convolutional	32	$3 \times 3$	$256 \times 256$
Convolutional	64	$3 \times 3 / 2$	$128 \times 128$
Convolutional	32	$1 \times 1$	$128 \times 128$
Convolutional	64	$3 \times 3$	
Residual			
Convolutional	128	$3 \times 3 / 2$	$64 \times 64$
Convolutional	64	$1 \times 1$	$64 \times 64$
Convolutional	128	$3 \times 3$	
Residual			
Convolutional	256	$3 \times 3 / 2$	$32 \times 32$
Convolutional	128	$1 \times 1$	$32 \times 32$
Convolutional	256	$3 \times 3$	
Residual			
Convolutional	512	$3 \times 3 / 2$	$16 \times 16$
Convolutional	256	$1 \times 1$	$16 \times 16$
Convolutional	512	$3 \times 3$	
Residual			
Convolutional	1024	$3 \times 3 / 2$	$8 \times 8$
Convolutional	512	$1 \times 1$	$8 \times 8$
Convolutional	1024	$3 \times 3$	
Residual			
Avgpool		Global	
Connected		1000	
Softmax			

## Một số ảnh minh họa kết quả thực hiện



# Một số ảnh minh họa kết quả thực hiện





# Định hướng phát triển

1. Nhóm chưa tùy biến code YOLOv3 để nhận diện một số đối tượng cụ thể. Đây cũng là hướng để nhóm nghiên cứu, phát triển thêm.
2. Trong thời gian tới nhóm mong muốn nghiên cứu tìm hiểu và cài đặt YOLOv4, YOLOv5 và các mô hình SSD, R-CNN, Fast R-CNN vào việc nhận diện một số đối tượng cụ thể.
3. Bổ sung vào bộ dữ liệu test:
  - Thêm ảnh và video clip trời tối
  - Thêm vào ảnh và video clip mờ hơn

# Tài liệu tham khảo

1. Lê Đình Duy. *Phát triển ứng dụng camera thông minh*. 2018
2. Redmon, Joseph, et al. *You only look once: Unified, real-time object detection*. IEEE conference on computer vision and pattern recognition. 2016.
3. Redmon, Joseph, and Ali Farhadi. *Yolov3: An incremental improvement*. arXiv 2018.
4. Hosang, Jan Hendrik, Rodrigo Benenson, and Bernt Schiele. *Learning nonmaximum suppression*. CVPR. 2017.
5. <https://github.com/AlexeyAB/darknet/#how-to-improve-object-detection>
6. [https://github.com/ledduy610/yolov3\\_deepsort](https://github.com/ledduy610/yolov3_deepsort)
7. <http://cs231n.github.io/transfer-learning/>
8. <https://pjreddie.com/darknet/imagenet/#darknet53>
9. <http://cocodataset.org/>
10. <https://phamdinhhkhanh.github.io/2020/03/09/DarknetAlgorithm.html>
11. <https://paperswithcode.com/method/yolov3>

# **BẮT ĐẦU BÁO CÁO CỦA CÁC NHÓM**

**Các nhóm sử dụng hướng dẫn các nội dung  
trong các slide mẫu ở trên**