

Sơ lược về Object Detection

Lê Đức Minh

Tháng 4, 2023

1 Giới thiệu

Bài toán xác định đồ vật (Object Detection) là 1 trong những bài toán lâu đời trong lĩnh vực thị giác máy tính (Computer Vision). Computer Vision là một lĩnh vực bao gồm các phương pháp thu nhận, phân tích và nhận dạng các hình ảnh, phát hiện các đối tượng, tạo ảnh, siêu phân giải hình ảnh và nhiều hơn vậy. Với những cải tiến của công nghệ trong những năm gần đây, Object Detection thu hút sự chú ý trong giới công nghệ, đây trở thành lĩnh vực hoạt động sôi nổi với nhiều ứng dụng có tính thực tiễn cao.

Object Detection (OD) là bài toán về khả năng nhận diện được đối tượng trong ảnh. Các ví dụ điển hình của OD bao gồm, xác định xe trong môi trường xung quanh được sử dụng trong công nghệ xe tự hành¹, nhận diện khuôn mặt (việc mở khóa bằng khuôn mặt của các điện thoại thông minh sử dụng cơ chế này để nhận diện chủ điện thoại), trong các lĩnh vực khác như quân sự, ngân hàng, các hệ thống kiểm duyệt, khi đã xác định được đối tượng cần nhận diện, đều có thể sử dụng công nghệ này²³.

Mục đích của bài viết này nhằm giới thiệu về Object Detection. Các mục sau sẽ gồm:

- Chương 2: Cơ chế hoạt động. Chương này giới thiệu về cách thức hoạt động của OD, song song đó, sẽ giới thiệu về cách học cổ điển (sử dụng feature extraction) và cách học hiện đại (sử dụng deep learning) cho các bài toán về OD.
- Chương 3: Một số mô hình nổi tiếng. Chương này cung cấp kiến thức về một số mô hình nổi tiếng trong lĩnh vực OD.

¹<https://www.tesla.com/AI>

²<https://viso.ai/deep-learning/object-detection/#:~:text=Specific%20object%20detection%20applications%20include,%20or%20number%2Dplate%20recognition.>

³<https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/inspired/where-facial-recognition-used>

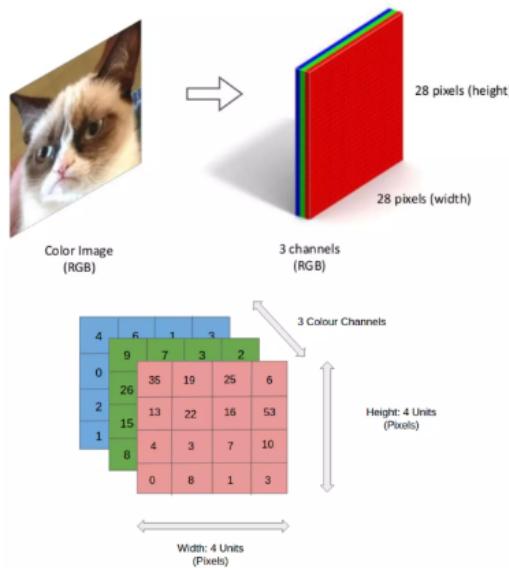
2 Cơ chế hoạt động

2.1 Khái niệm

2.1.1 Hệ màu RGB, Tensor

Hàng ngày, các bức ảnh ta được xem đều rực rỡ và đủ màu sắc. Ta có thể nhận diện rõ ràng các vật trong ảnh, màu sắc, độ sáng tối, xa gần dựa trên các thông tin về màu sắc, độ sáng tối hay tỉ lệ giữa 1 vật với các vật còn lại trong ảnh để mường tượng được độ xa gần của đối tượng. Tuy nhiên, các thuật toán thì không như ta, nó cần các số liệu, các dạng thông tin, dữ liệu mà nó có thể xử lý để cung cấp cho ta các thông tin mà ta cần. Vì vậy, cần có một biểu diễn mà ở đó, các thuật toán, có thể hiểu được ảnh.

Hệ màu RGB: RGB viết tắt của red (đỏ), green (xanh lục), blue (xanh lam), là ba màu chính của ánh sáng khi tách ra từ lăng kính. Khi trộn ba màu trên theo tỉ lệ nhất định có thể tạo thành các màu khác nhau. 1 bức ảnh thông thường sẽ gồm 3 (R, G và B) lớp ảnh với mỗi lớp sẽ chứa những tham số này để khi kết hợp 3 lớp lại với nhau, chúng sẽ tạo ra 1 bức ảnh màu như ta thường thấy.



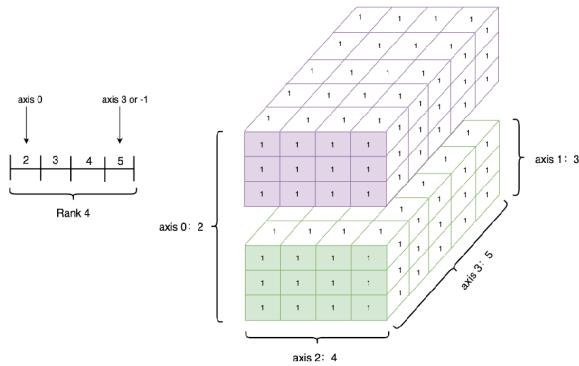
Hình 1: Hệ màu RGB. ⁴

Với mỗi một lớp ảnh, thông thường sẽ là 1 hình chữ nhật có chiều dài (height), chiều rộng (width) được định nghĩa tùy theo nhu cầu sử dụng. Ở mỗi

⁴Nguồn: <https://www.slideshare.net/BerttonEarnshaw/a-brief-survey-of-tensors>

lớp ảnh, với 1 đơn vị của lớp ảnh được gọi là pixel, có giá trị từ 0 đến 255. Ta thấy rằng, với mỗi bộ 3 số r, g, b nguyên trong khoảng [0, 255] sẽ cho ra một màu khác nhau. Do có 256 cách chọn trong nền đỏ (R), 256 cách chọn màu trong nền xanh (G), 256 cách điều chỉnh màu xanh (B), thì tổng số màu có thể tạo ra bằng hệ màu RGB là: $256 * 256 * 256 = 16777216$ màu.

Tensor: Ta đã quen với các biểu diễn toán học của vector với số chiều nhỏ như là 2 hoặc mô tả các khối hình học trong không gian 3 chiều ở cấp học trung học phổ thông và đại. Trong toán học, 1 tensor là 1 biểu diễn của các khối hình học, được miêu tả bằng đại lượng vector có hướng hoặc vô hướng, hoặc bằng các phép giữa các tensor⁵.



Hình 2: Ảnh được biểu diễn bằng tensor 3 chiều.⁶

Lấy ví dụ, 1 vector là 1 biểu diễn hình học (đường thẳng có hướng) 2 chiều, ta có thể gọi nó là 1 tensor 2 chiều, với các khối hình học 3 chiều, ta có thể gọi đó là 1 tensor ba chiều. Các hình ảnh màu gồm 3 lớp RGB cũng là 1 tensor 3 chiều⁷ (Hình 1).

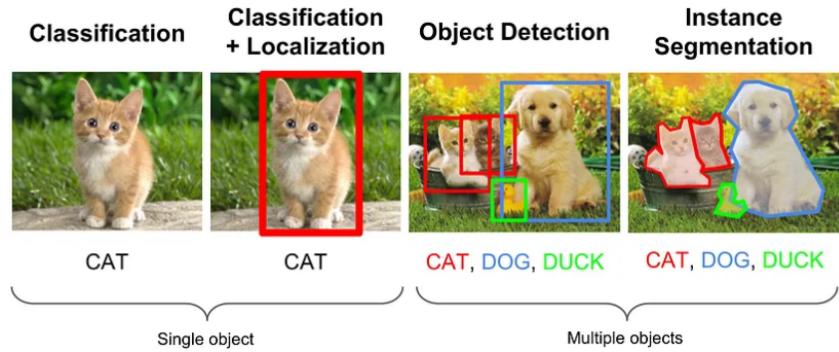
2.1.2 Object Detection

Trước khi đi sâu hơn về cách thức hoạt động cũng như các thuật toán. Ta cần có 1 hiểu biết nhất định về nhiệm vụ, yêu cầu của Object Detection, hay ít nhất rằng, có 1 sự phân biệt rõ ràng giữa nó và các nhiệm vụ khác trong lĩnh vực thị giác máy tính.

⁶Nguồn: https://www.paddlepaddle.org.cn/documentation/docs/en/guides/beginner/tensor_en.html

⁷Nguồn: <https://en.wikipedia.org/wiki/Tensor>

⁷Nguồn: https://phamduinhkhanh.github.io/deepai-book/ch_algebra/nb_appendix_algebra.html

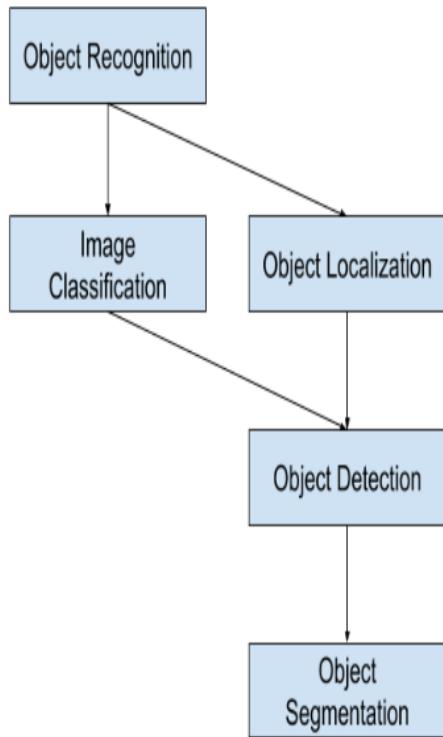


Hình 3: Phân biệt Object Detection và các bài toán khác trong lĩnh vực thị giác máy tính.⁹

Trong thị giác máy tính, có 4 bài toán lớn, bao gồm: phân loại ảnh (Image Classification), phân loại ảnh dựa trên vị trí của đối tượng (Object Localization), nhận diện đối tượng (Object Detection) và phân đoạn ảnh (Instance Segmentation). 2 cái trước chỉ hoạt động cho 1 đối tượng trong 1 bức ảnh, 2 cái còn lại có khả năng thực hiện chức năng của nó cho nhiều đối tượng trong 1 khung hình.

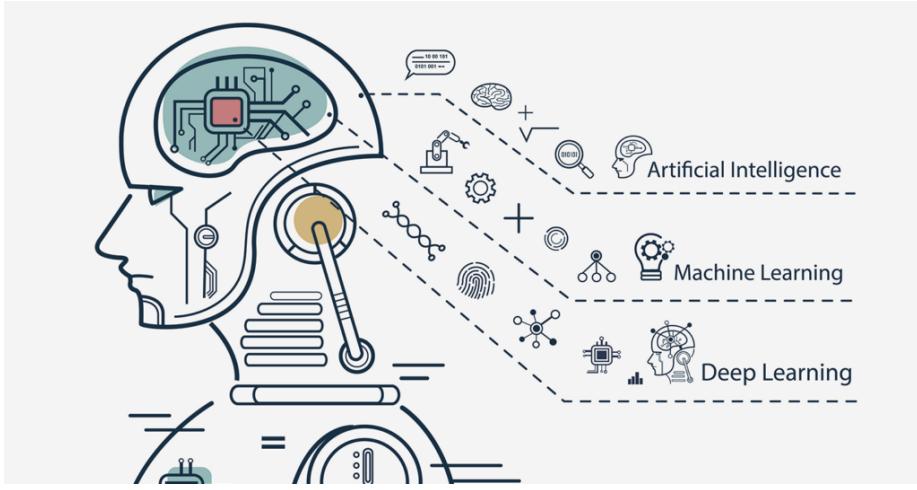
Ở hình 3, ta có thể thấy rằng, với 2 hình đầu tiên, bài toán sẽ được mô tả như sau "Bức ảnh này về đối tượng gì hoặc đối tượng đó là gì và đang ở đâu trong bức ảnh?", ở các hình còn lại, máy tính sẽ trả lời cho câu hỏi "Trong bức ảnh này gồm những đối tượng nào và những đối tượng đó ở đâu?". Ta có thể hiểu rằng, Object Detection là sự nâng cấp của sự kết hợp giữa Image Classification và Object Localization. Trong trường hợp của Instance Segmentation, nhiệm vụ này khó hơn Object Detection ở việc thay vì chỉ đóng khung đối tượng trong khung chữ nhật (bounding box), Instance Segmentation phải "gọt" cái khung ấy sao cho vừa khít với đối tượng trong ảnh, có thể được dùng để tách đối tượng đó với ảnh nền (background) hoặc tách các đối tượng khác nhau có cùng lớp.

⁹Nguồn: <https://medium.com/zylapp/review-of-deep-learning-algorithms-for-object-detection-c1f3d437b852>



Hình 4: Mối quan hệ giữa các bài toán trong Computer Vision

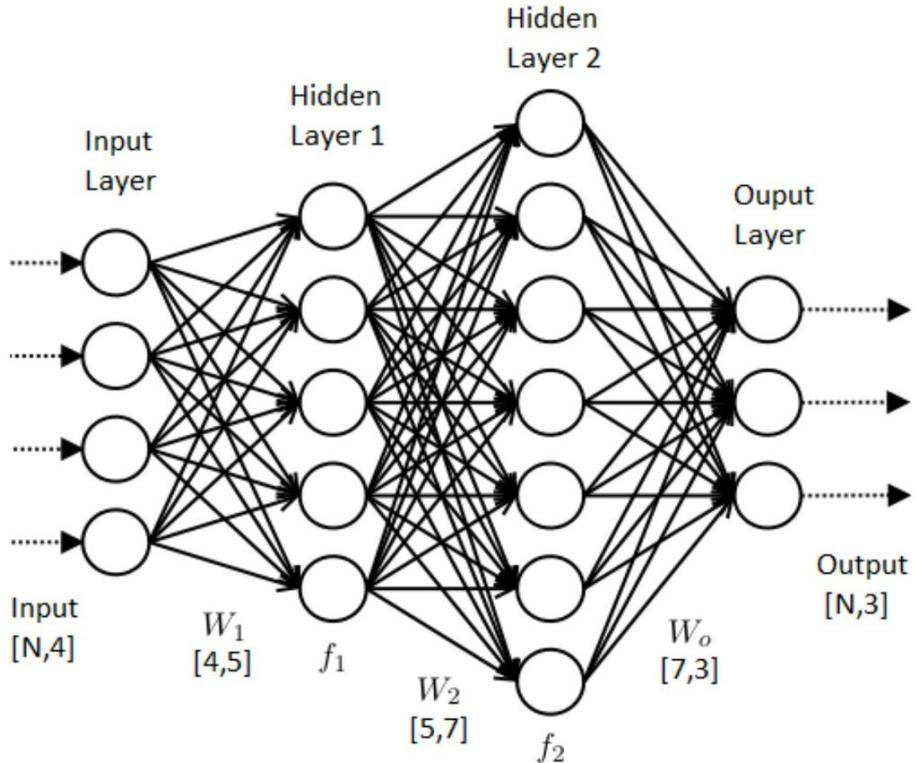
Trước khi đi sâu vào sự khác biệt giữa 2 cách học (sử dụng machine learning và deep learning), ta cần tìm hiểu đôi chút về sự khác biệt giữa 2 cách học này. Về bản chất, AI (artificial intelligence) là 1 hướng nghiên cứu sao cho máy có thể hoàn thành các tác vụ như 1 con người, nói cách khác, là 1 lĩnh vực nghiên cứu các thuật toán sao cho có thể mô phỏng trí thông minh của con người. Sự ra đời của lĩnh vực này có thể khắc phục 2 điều mà có thể được xem như là điểm yếu của con người: xử lý 1 khối lượng lớn dữ liệu cũng như chú ý vào các sự thay đổi nhỏ trong khối lượng dữ liệu đó.



Hình 5: AI-Machine Learning-Deep Learning ¹⁰

Ở cả hai cách học đều cần 1 khối lượng nhất định dữ liệu để xử lý, với khối lượng dữ liệu càng lớn, chất lượng dự đoán sẽ càng chính xác. Tuy nhiên, cả hai vẫn phải phụ thuộc vào các tiền xử lý dữ liệu, khi học các thông tin không được chắt lọc có cấu trúc (phân tích kỹ lưỡng) thì kết quả sẽ không như mong đợi, cũng giống như việc khi ta dạy trẻ em, nếu chỉ cho con trẻ tiếp thu thông tin từ nhiều nguồn nhưng không được chắt lọc kỹ càng, có khả năng rằng phần lớn thông tin đó không mang tính xây dựng, đưa trẻ ấy sẽ khó có thể "cho ra" những "hồi đáp" mà ta mong đợi ở đứa trẻ. Sự khác biệt lớn giữa 2 machine learning và deep learning; cái trước rút ra các mẫu (pattern) từ lượng thông tin đầu vào (tùy vào thuật toán được sử dụng cho máy), cái sau là sự cải tiến vượt bậc của cái trước, cách học này có thể phân tích được tính chặt chẽ (logic) của lượng dữ liệu đầu vào để đưa ra kết luận.

¹⁰Nguồn: <https://appen.com/blog/ai-vs-deep-learning-vs-machine-learning-everything-youve-ever-wanted-to-know/>



Hình 6: Kiến trúc của mạng thần kinh nhân tạo ¹¹

Kiến trúc của deep learning dựa trên hệ thống thần kinh của con người, được gọi là mạng neuron nhân tạo (Artificial Neural Network -ANN). Mạng này được thiết kế gồm 3 lớp chính, lớp đầu vào (input layer), lớp ẩn (hidden layer), lớp đầu ra (output layer) (Hình 6). Hidden layer sẽ phụ trách phần lớn khối lượng xử lý, trích xuất các đặc trưng của dữ liệu. Các xây dựng của các lớp sẽ phụ thuộc vào yêu cầu của bài toán hoặc người triển khai mô hình.

2.2 Cách học sử dụng machine learning

Trong object detection, cách học này cần nhiều thông tin phụ trợ để đưa ra kết quả có độ chính xác cao, bao gồm:

- **Feature Descriptor:** Bộ mô tả đặc trưng, là các thuật toán có khả năng biến đổi dữ liệu thành các đặc trưng (có thể kể đến các thuật toán HOG, SUFT, SHIFT).

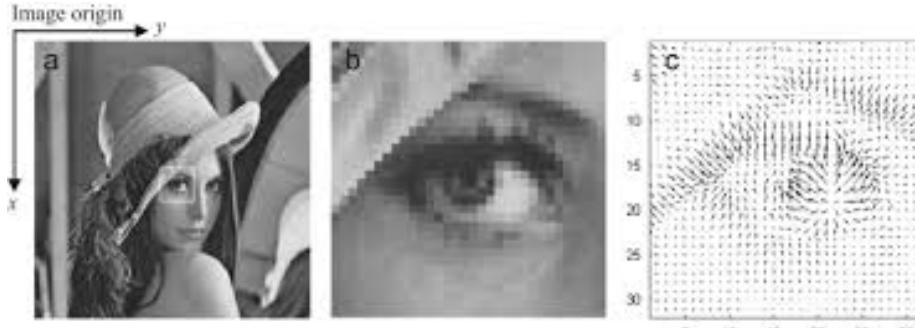
¹¹Nguồn: <https://www.datasciencecentral.com/the-artificial-neural-networks-han-dbook-part-1/>

- **Histogram:** Biểu diễn phân phối của các cường độ màu sắc theo khoảng giá trị.
- **Gradient:** Là đạo hàm của vector cường độ màu sắc giúp phát hiện hướng di chuyển của các vật trong hình ảnh, nói cách khác, thông tin này thể hiện sự biến đổi giữa các pixel trong 1 window (ví dụ, pixel 1 khác pixel 2 với giá trị là 2, giá trị gradient giữa 2 pixel là 2) (Hình 7).
- **Local cell:** Ô cục bộ (có thể hiểu là 1 pixel).
- **Local portion:** Vùng cục bộ, là 1 cùng được trích xuất từ nhiều ô vuông trên hình (có thể được gọi là 1 window).
- **Local Normalization:** Phép chuẩn hóa trên 1 vùng cục bộ, mục đích của việc này là để đồng nhất các giá trị cường độ màu sắc về chung 1 phân phối (các hàm normalization thường dùng là norm 2 và norm 1).
- **Gradient Direction**¹²: Phương Gradient, là độ lớn góc giữa vector gradient x và y giúp xác định phương thay đổi cường độ màu sắc hay chính là phương đỗ bosnh của hình ảnh.

$$\theta = \arctan\left(\frac{G_y}{G_x}\right)$$

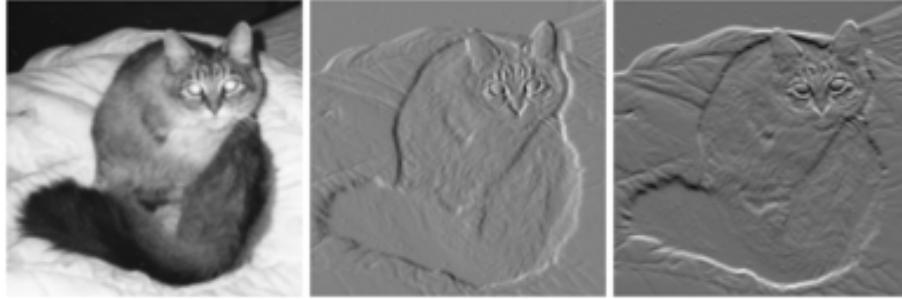
- **Gradient Magnitude:** Độ lớn gradient, giá trị độ lớn của vector gradient theo phương ngang và dọc.

$$|G| = \sqrt{G_x^2 + G_y^2}$$



Hình 7: Biểu diễn Gradient trong ảnh ¹³

¹²Hình 8 mô tả ảnh sau khi tính gradient 2 phương. Ta có thể thấy rằng, ảnh giữa được tính theo phương ngang, nhìn từ trái sang phải, ở phần rìa của con mèo bên trái được thể hiện đậm hơn, ở ảnh cuối được tính theo phương thẳng đứng (trục y), khi này, nhìn từ trên xuống dưới, phần rìa của con mèo ở phía trên được thể hiện rõ hơn.



Hình 8: Hướng gradient trong ảnh ¹⁴

Sử dụng Object Detection bằng cách học machine learning phải trải qua nhiều bước xử lý và cần thêm nhiều loại thông tin phụ trợ. Trong xử lý ảnh, thuật toán HOG ¹⁵(Histogram of oriented gradient) làm một trong những bộ mô tả đặc trưng mạnh giúp mã hóa hình ảnh thành một véc tơ đặc trưng với số chiều đủ lớn để có thể phân loại tốt các bức ảnh. Thuật toán tỏ ra khá hiệu quả khi ứng dụng tốt để phát hiện người với nhiều kích thước khác nhau. Đồng thời trong một số trường hợp phân loại ảnh, khi bộ dữ liệu có kích thước nhỏ thì những mạng nơ ron lớn như CNN có thể hoạt động không chính xác do tập ảnh huấn luyện không đủ bao quát các khả năng. Khi đó việc áp dụng những phương pháp cổ điển để trích lọc đặc trưng như HOG lại mang lại những kết quả bất ngờ mà tốt ít tài nguyên và chi phí tính toán.

2.3 Cách học sử dụng deep learning

Ở cách học này, phần lớn việc rút trích các đặc trưng của ảnh sẽ được thực hiện tự động bởi máy (thuật toán). Mạng thần kinh được sử dụng rộng rãi để giải quyết tác vụ này là mạng tích chập (Convolution Neural Networks - CNN). Nhận diện đối tượng thực hiện xác định vị trí hiện diện của đối tượng trong bounding box và nhãn của đối tượng trong ảnh, đầu vào và đầu ra bao gồm:

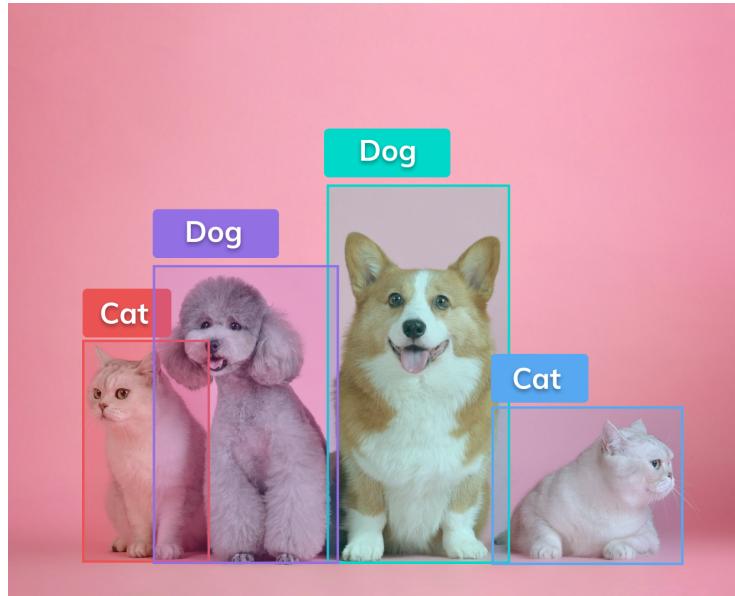
- **Input:** Một hình ảnh có một đối tượng hoặc nhiều đối tượng.
- **Output:** Một hoặc nhiều bounding box và nhãn cho mỗi bounding box.

¹³Nguồn: <https://www.sciencedirect.com/science/article/abs/pii/S003132031400131>

9

¹⁴Nguồn: https://en.wikipedia.org/wiki/Image_gradient

¹⁵Tham khảo tại đây: <https://phamdinhkhanh.github.io/2019/11/22/HOG.html>



Hình 9: Nhận diện đồ vật ¹⁶

3 Một số mô hình nổi tiếng

Có nhiều thuật toán và mô hình lớn sử dụng mạng học sâu để giải quyết tác vụ nhận diện đối tượng trong ảnh, trong đó, có 2 họ lớn nhất (gồm nhiều thuật toán khác nhau) nhưng cái lõi có nét tương đồng hoặc là sự cải tiến của cái trước nó) bao gồm họ R-CNN và họ YOLO¹⁷.

Nhìn chung, cả 2 họ trên là sự đánh đổi lẫn nhau, họ R-CNN đạt được độ chính xác cao hơn, tuy nhiên, thời gian xử lý lại là nhược điểm chí mạng của nó; trong khi đó, học YOLO có tốc độ xử lý ảnh cực cao, có thể sử dụng để xử lý ảnh trong thời gian thực, tuy nhiên, độ chính xác sẽ không được bằng như R-CNN.

3.1 Các mô hình họ nhà R-CNN

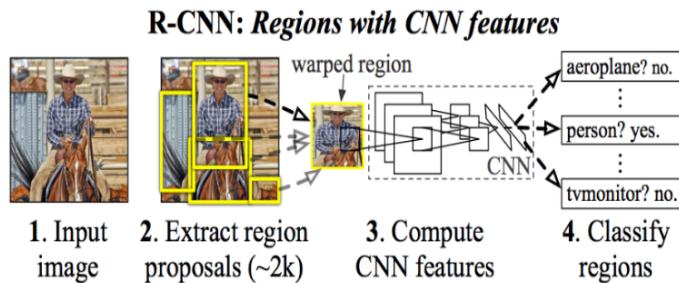
R-CNN được giới thiệu lần đầu vào 2014 bởi Ross Girshick và các cộng sự ở UC Berkeley một trong những trung tâm nghiên cứu AI hàng đầu thế giới trong bài báo Rich feature hierarchies for accurate object detection and semantic segmentation.

Kiến trúc của R-CNN gồm 3 thành phần đó là:

¹⁶Nguồn: <https://www.v7labs.com/blog/object-detection-guide>

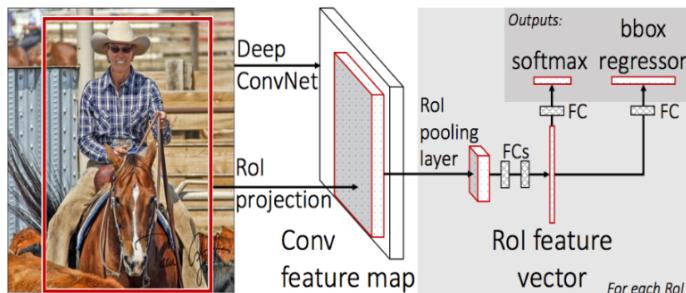
¹⁷Tham khảo tại đây: <https://phamdinhkhanh.github.io/2019/09/29/OverviewObjectDetection.html>

- **Vùng đề xuất hình ảnh (Region proposal):** Có tác dụng tạo và trích xuất các vùng đề xuất chứa vật thể được bao bởi các bounding box.
- **Trích lọc đặc trưng (Feature Extractor):** Trích xuất các đặc trưng giúp nhận diện hình ảnh từ các region proposal thông qua các mạng deep convolutional neural network.
- **Phân loại (Classifier):** Dựa vào input là các features ở phần trước để phân loại hình ảnh chứa trong region proposal về đúng nhãn.



Hình 10: Mô hình RCNN được đề xuất năm 2014.

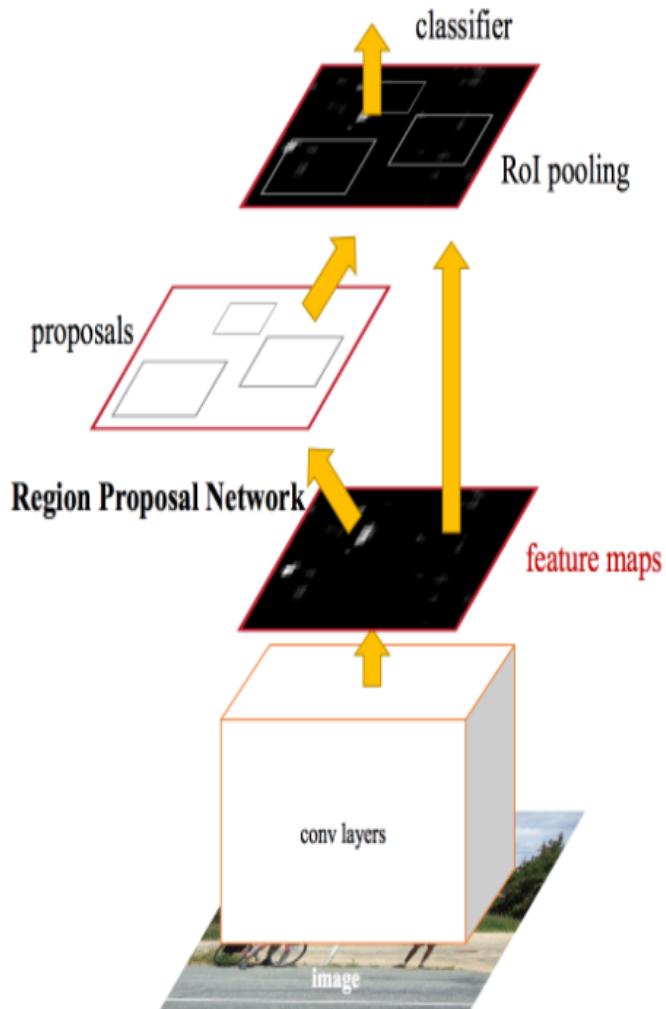
Các cải tiến về sau giúp cho R-CNN thực hiện tác vụ nhanh hơn và chính xác hơn. Ở Fast R-CNN (2015), sau khi nhận thấy R-CNN phải huấn luyện qua một pipeline gồm quá nhiều bước (giai đoạn chuẩn bị, sử dụng 3 mô hình cho quá trình huấn luyện), mô hình này sử dụng single model thay vì pipeline. Với cải tiến về sau, Faster R-CNN (2016), mang lại độ chính xác cao nhất đạt được trên cả hai nhiệm vụ phát hiện và nhận dạng đối tượng tại các cuộc thi ILSVRC-2015 và MS COCO-2015.



Hình 11: Mô hình Fast R-CNN đề xuất năm 2015.

Faster R-CNN được thiết kế để đề xuất và tinh chỉnh các region proposals như là một phần của quá trình huấn luyện, được gọi là mạng đề xuất khu vực

(Region Proposal Network), hoặc RPN. Các vùng này sau đó được sử dụng cùng với mô hình Fast R-CNN trong một thiết kế mô hình duy nhất. Những cải tiến này vừa làm giảm số lượng region proposal vừa tăng tốc hoạt động trong thời gian thử nghiệm mô hình lên gần thời gian thực với hiệu suất tốt nhất. Tốc độ là 5fps trên một GPU.

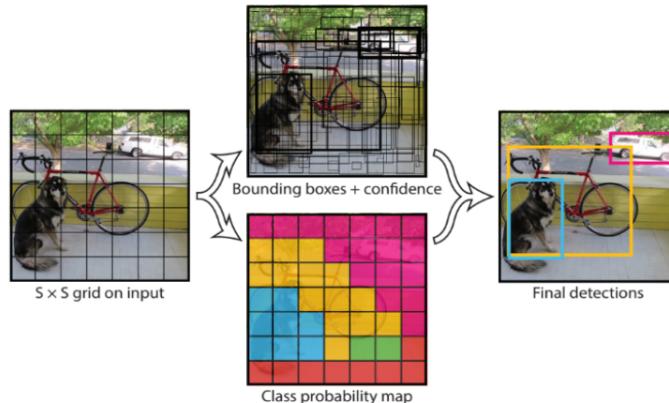


Hình 12: Mô hình Faster R-CNN đề xuất năm 2016.

3.2 Các mô hình họ nhà YOLO

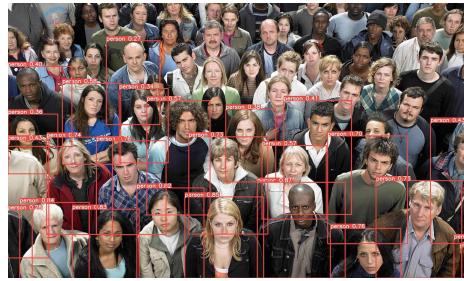
Một họ mô hình nhận dạng đối tượng phổ biến khác được gọi chung là YOLO. YOLO không phải là bạn chỉ sống một lần đâu nhé, nó có nghĩa là bạn chỉ nhìn một lần (you only look one), được phát triển bởi Joseph Redmon, và các cộng sự.

Mô hình hoạt động bằng cách trước tiên phân chia hình ảnh đầu vào thành một lưới các ô (grid of cells), trong đó mỗi ô chịu trách nhiệm dự đoán các bounding boxes nếu tâm của nó nằm trong ô. Mỗi grid cell (tức 1 ô bất kì nằm trong lưới ô) dự đoán các bounding boxes được xác định dựa trên tọa độ x, y (thông thường là tọa độ tâm, một số phiên bản là tọa độ góc trên cùng bên trái) và chiều rộng (width) và chiều cao (height) và độ tin cậy (confidence) về khả năng chứa vật thể bên trong. Ngoài ra các dự đoán nhãn cũng được thực hiện trên mỗi một bounding box.

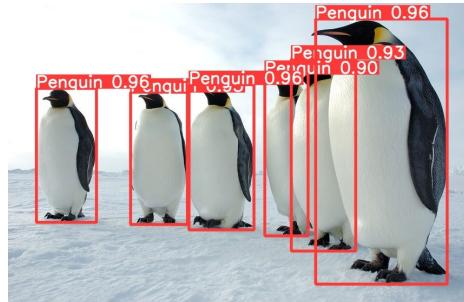


Hình 13: Kiến trúc mô hình YOLO thế hệ đầu tiên

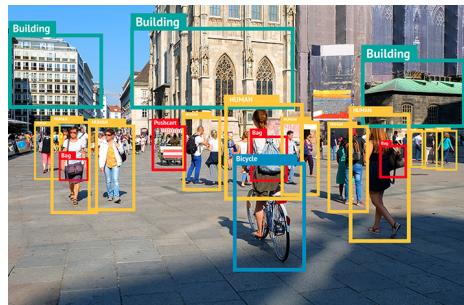
Những cải tiến về sau của YOLO giúp cho mô hình dự đoán chính xác hơn và tốc độ dự đoán nhanh hơn sau mỗi lần YOLO đề xuất các cải tiến. Hiện nay, YOLO đã có đến phiên bản 7 và 8, tốc độ xử lý và độ chính xác đều được cải thiện đến gần mức tuyệt đối.



(a)



(b)



(c)

Hình 14: YOLO có thể hoạt động trong thời gian thực và có thể nhận diện nhiều đối tượng trong 1 khung hình theo thời gian thực.¹⁹

Ngoài 2 họ được đề cập ở trên, còn có các mô hình khác như Single Shot MultiBox Detector (SSD)²⁰, RetinaNet, sau đợt dịch Covid bùng nổ (2020), sự xuất hiện của MaskFaceNet cũng mang lại kết quả hứa hẹn.

¹⁹<https://bytewarn.com/yolov5-object-detection-experiments.html>

²⁰<https://phamdinhkhanh.github.io/2019/10/05/SSDModelObjectDetection.html>,
<https://viblo.asia/p/object-detection-speed-and-accuracy-faster-r-cnn-r-fcn-ssd-fpn-retinanet-and-yolov3-ByEZkJRWKQ0>