AI VIET NAM – COURSE 2024

Phát hiện Đối tượng Sử dụng Học Sâu, Mạng Nơ-ron Tích Chập (CNN) và Biến Đổi Hình Ảnh (Vision Transformers): Tổng quan

Ngày 30 tháng 3 năm 2024

Ngày công bố:	13/04/2023
Tác giả:	Ayoub Benali Amjoud và Mustapha Amrouch
Nguồn:	Chương trình Giải thưởng Xuất sắc Nghiên cứu 2019–2022 theo Khoản tài trợ 5UIZ2019
Từ khóa:	Phát hiện đối tượng, học sâu, khảo sát, mạng nơ-ron tích chập, biến đổi hình ảnh, mạng nơ-ron
Người tóm tắt:	Nhi Ng Thảo

1 Mở đầu:

- Tầm quan trọng và các phương pháp trong việc nhận dạng đối tượng trong lĩnh vực thị giác máy tính, cũng như chỉ ra các ứng dụng phổ biến của Object Detection trong nhiều lĩnh vực khác nhau, bao gồm: xe tự hành, giám sát an ninh, hình ảnh y tế và cả robotic.
- Phác thảo quá trình phát triển và nghiên cứu về nhận dạng đối tượng, tập trung vào các giai đoạn chuyển đổi từ các mô hình truyền thống sang các phương pháp dựa trên học sâu, đặc biệt là vào năm 2014.
- Đề cập đến sự ảnh hưởng đáng kể của Deep Learning và Convolutional Neural Networks (CNNs) trong việc cải thiện nhận dạng đối tượng, mở ra những tiến bộ đáng kể trong các tác vụ như phân loại. Điều này làm nổi bật vai trò quan trọng của mô hình học sâu trong việc thúc đẩy nhận dạng đối tượng và hiểu ngữ cảnh của hình ảnh và video.

2 Sơ lược về bài báo:

- **DATASET & EVALUATION METRICS**
  - DATASET **PASCALVOC**: Bộ dữ liệu gồm 10.000 hình ảnh cho việc phát hiện vật thể, với 20 loại đối tượng khác nhau. **MS -COCO**: Bộ dữ liệu gồm hơn 200.000 hình ảnh và 80 loại đối tượng. **ILSRVC**: Bộ dữ liệu bao gồm hơn 1 triệu hình ảnh được gắn nhãn với khoảng 600 đối tượng. **OPEN IMAGE**: Bộ dữ liệu có khoảng 9,3 triệu hình ảnh được gắn nhãn với khoảng 600 đối tượng.
  - EVALUATION METRICS

– **Intersection over Union (IoU)**: Đo lường chất lượng phát hiện bằng cách tính toán sự khác biệt giữa bounding box thực tế và dự đoán.

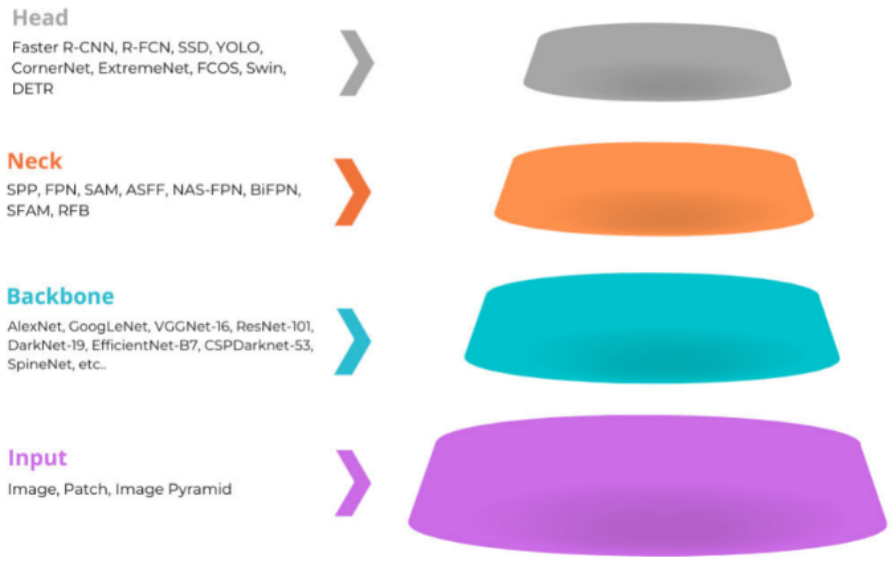
$$\text{IoU} = \frac{\text{Area of Intersection}}{\text{Area of Union}} \tag{1}$$

Mean Average precision (mAP): Độ chính xác trung bình của tất cả các lớp. **Mean Average Recall (mAR)**: Giá trị trung bình của recall cho tất cả các lớp.

Ref	Dataset	Evaluation metric
[53]	The PASCAL VOC Challenge	mAP
[56]	The COCO Object Detection Challenge	mAP, mAR

Hình 1: Tổng quan về các phương pháp, bộ dữ liệu và evaluation metrics

- **Input:** một bức anh, patch, hoặc một số lượng lớn hình ảnh, video. **Backbone:** có thể là một CNN giống VGG, ResNet,.. **The neck:** Top backbone **Head:** chia ra 2 loại: dự đoán dày đặt và dự đoán thưa thớt.



Hình 2: Những thành phần trong mô hinh OD

3 Backbone

Đóng vai trò quan trọng trong việc trích xuất đặc trưng từ ảnh trước khi chuyển sang các bước tiếp theo, chẳng hạn như định vị đối tượng. Nhiều mô hình CNN thường được sử dụng trong phát hiện đối tượng, bao gồm các mô hình được huấn luyện sẵn như VGGNet, ResNet và EfficientNet.

- **AlexNet**

  - Gồm 8 lớp: 5 lớp tích chập, 2 lớp kết nối đầy đủ (FC) và 1 lớp đầu ra softmax với 1000 lớp.
  - Sử dụng hàm kích hoạt ReLU và các lớp chuẩn hóa cục bộ.
- **VGGNet**
  - Có 5 lớp tích chập tiếp theo là 3 lớp FC.
  - Sử dụng bộ lọc tích chập nhỏ (3x3) và có nhiều lớp (16-19 lớp).
- **ResNet**
  - Nổi tiếng vì khả năng huấn luyện mạng sâu mà không gặp vấn đề mất gradient.
  - Giới thiệu các kết nối dư (residual connection), cho phép mạng học các ánh xạ dư thay vì các ánh xạ gốc.
  - Sử dụng các khối dư chứa nhiều lớp tích chập và chuẩn hóa theo batch.
- **Inception-ResNet**
  - Tích hợp các kết nối dư vào kiến trúc Inception để cải thiện luồng gradient và cho phép huấn luyện mạng sâu hơn.
  - Sử dụng các khối Inception để trích xuất đặc trưng ở các kích thước khác nhau, kết hợp với các kết nối dư.
- **EfficientNet**
  - Sử dụng kỹ thuật thu phóng hợp chất (compound scaling) để cân bằng kích thước mô hình với độ chính xác và hiệu quả tính toán.
  - Dùng các khối MBConv, kết hợp các lớp tích chập theo chiều sâu và theo điểm.
- **GoogLeNet**
  - Phát triển bởi Google, có 22 lớp với 27 lớp pooling, được tổ chức thành 9 khối Inception.
  - Sử dụng các khối Inception để trích xuất đặc trưng ở các kích thước khác nhau, tiếp theo là pooling trung bình toàn cục.
- **Các backbone khác:**
  - CSP-ResNeXt: Áp dụng phương pháp CSP (Cross Stage Partial) cho ResNeXt.
  - DenseNet: Tích hợp các kết nối dày đặc giữa các lớp thông qua Dense Block.
  - SENet: Sử dụng các khối squeeze-and-excitation để hiệu chỉnh lại các map đặc trưng một cách thích ứng.
  - Hourglass: Sử dụng xử lý lặp lại từ dưới lên và từ trên xuống để nắm bắt các đặc trưng chi tiết và toàn diện.
  - SpineNet: Có kiến trúc backbone thưa thớt với các đặc trưng trung gian được sắp xếp theo tỷ lệ.
  - CSP-Darknet: Sử dụng chiến lược CSPNet để phân vùng map đặc trưng.
  - ConvNeXt: Được xây dựng hoàn toàn từ các khối ConvNet tiêu chuẩn.

4 Data Augmentation

Quá trình huấn luyện, các mô hình sử dụng nhiều chiến lược học tập khác nhau như tinh chỉnh (fine-tuning), tăng cường dữ liệu (data augmentation), học thác đổ (cascade learning) và lấy mẫu mất cân bằng (imbalance sampling). Các chiến lược này giúp mô hình hoạt động hiệu quả để cải thiện độ chính xác và thời gian thực thi cho cả tác vụ định vị và phân loại.

- **COLOR SPACE**

  - Chọn một kênh màu duy nhất và thêm ma trận giá trị 0 vào các kênh còn lại.
  - Thay đổi giá trị ma trận RGB để điều chỉnh độ tương phản, độ sáng và độ bão hòa.
- **Rotation**
  - Xoay ảnh sang trái hoặc phải trong phạm vi góc từ 1° đến 359°.
  - Cần lưu ý góc xoay để tránh làm mất giá trị nhãn của ảnh.
- **Translation**
  - Dịch chuyển ảnh lên, xuống, trái hoặc phải.
  - Lấp đầy phần trống sau khi dịch chuyển bằng giá trị hằng số (0 hoặc 255) hoặc nhiễu ngẫu nhiên.
- **Cropping**
  - Cắt phần trung tâm của ảnh.
  - Cắt ngẫu nhiên để giảm kích thước ảnh.
- **Kernel filters**
  - Áp dụng bộ lọc làm mờ Gaussian để tạo ảnh mờ.
  - Sử dụng bộ lọc cạnh dọc hoặc ngang để tạo ảnh sắc nét hơn.
- **Random erasing**
  - Chọn ngẫu nhiên một vùng trong ảnh và che khuất nó bằng giá trị pixel trung bình (0 hoặc 255).
  - Giúp mô hình tránh che khuất và khớp quá mức.

5 Anchor-based detectors

Các khung neo (anchor box) là các khung bao được định nghĩa trước với kích thước và tỷ lệ đa dạng, tương ứng với các đối tượng có thể xuất hiện trong hình ảnh. Thay vì dự đoán trực tiếp các khung bao, mạng lưới dự đoán xác suất và thuộc tính khác như nền và sự giao nhau với các khung neo. Các khung neo giúp mạng lưới phát hiện nhiều đối tượng có kích thước và vị trí khác nhau trong hình ảnh.

(a) Tạo hàng nghìn khung neo (anchor box) ứng viên mô tả tốt nhất kích thước, vị trí và hình dạng của các đối tượng.

(b) Dự đoán độ lệch cho từng khung bao quanh (bounding box).

(c) Tính toán hàm mất mát cho mỗi hộp neo dựa trên dữ liệu thực.

(d) Đối với mỗi khung neo (anchor box), tính toán Tỷ lệ Giao nhau trên Tổng diện tích (Intersection Over Union - IOU) để kiểm tra xem khung bao của đối tượng nào có IOU lớn nhất.

(e) Khi độ tin cậy lớn hơn 0.5, thông báo cho anchor box (khung neo) rằng nó cần phát hiện đối tượng có IOU cao nhất. Và đưa dự đoán này vào hàm mất mát (loss function).

(f) Như đã được đề cập, xác suất này nếu chỉ nhỏ hơn 0.5 một cách không đáng kể, chúng ta sẽ hướng dẫn hộp neo không học hỏi từ mẫu này vì dự đoán là mơ hồ; mặt khác, nếu xác suất nhỏ hơn 0.5 đáng kể, thì hộp neo có khả năng dự đoán rằng không có đối tượng nào hiện diện.

(g) Cuối cùng, bằng cách sử dụng quy trình này, chúng tôi đảm bảo rằng mô hình chỉ học được cách nhận dạng các đối tượng thực sự.

Tuy nhiên, các anchor đòi hỏi thiết kế và áp dụng cẩn thận trong các khung phát hiện vật thể.

- Trong thiết kế neo (anchor), tỷ lệ bao phủ của không gian vị trí của thể hiện (instance) là một trong những yếu tố quan trọng nhất. Để đảm bảo tỷ lệ thu hồi (recall rate) tốt, các neo được thiết kế kỹ lưỡng dựa trên các thống kê được tính toán từ tập dữ liệu huấn luyện/kiểm chứng.
- Chắc chắn rằng một số lựa chọn thiết kế dựa trên một bộ dữ liệu cụ thể có thể không áp dụng được cho các ứng dụng khác, điều này ảnh hưởng đến tính tổng quát (generality) của mô hình.
- Trong giai đoạn học, các phương pháp dựa trên neo (anchor-based) phụ thuộc vào tỷ lệ giao nhau trên liên hợp (Intersection over Union - IoU) để xác định các mẫu dương/âm, do đó làm tăng thêm tính toán và các tham số siêu (hyper-parameters) cho hệ thống phát hiện vật thể.

Các khung phát hiện vật thể dựa trên anchor (neo) thường được chia thành hai nhóm: bộ phát hiện dựa trên proposition (đề xuất) hai giai đoạn và phương pháp không proposition (miễn đề xuất) một giai đoạn.

5.1 Two-Stage Methods

5.1.1 REGION-BASED CONVOLUTIONAL NEURAL NETWORKS (R-CNN)

Được phát triển bởi R. Girshick et al. vào năm 2014, mô hình R-CNN đề xuất nhiều khung (box) trong ảnh và kiểm tra xem liệu một trong số đó có chứa vật thể hay không. Những khung này được gọi là vùng (region). Tìm kiếm chọn lọc (Selective Search) được sử dụng để trích xuất 2000 vùng ứng viên; các vùng ứng viên này được cắt và đổi kích thước để phù hợp với đầu vào của bộ trích chọn đặc trưng CNN, trích xuất ra một vector đặc trưng 4096 chiều được truyền vào một số bộ phân loại để dự đoán lớp. Các máy SVM được gán cho mỗi lớp để phân loại sự xuất hiện của vật thể.

R-CNN bao gồm ba bước đơn giản:

(a) Bằng cách đề xuất khoảng 2000 khung đối tượng (candidate boxes), sử dụng thuật toán tìm kiếm chọn lọc (selective search algorithm) để quét hình ảnh đầu vào và phát hiện các vật thể có thể xuất hiện trong ảnh.

(b) Đối với mỗi khung đối tượng dự đoán (candidate box), chúng tôi sử dụng mạng nơ-ron tích chập (CNN) để trích xuất đặc trưng.

(c) Kết quả của mỗi mạng nơ-ron tích chập (CNN) được truyền tới một SVM để phân loại và tới một bộ hồi quy tuyến tính để tinh chỉnh khung bao quanh của đối tượng.

5.2 SPATIAL PYRAMID POOLING NETWORK (SPPNet)

SPPNet triển khai một mạng CNN đặc biệt gọi là Spatial Pyramid Pooling (SPP) giữa các lớp tích chập và lớp fully connected.

Tương tự như R-CNN, SPPNet sử dụng thuật toán chọn lọc để tạo ra khoảng 2,000 đề xuất vùng cho mỗi hình ảnh. Sau đó, nó chỉ sử dụng ZFNet một lần để trích xuất đặc trưng trực tiếp từ toàn bộ hình ảnh. Tại lớp tích chập cuối cùng, các vùng đặc trưng được xác định bởi mỗi đề xuất vùng sẽ đi qua lớp SPP, sau đó là lớp fully connected.

SPPNet sử dụng SPP cho mọi đề xuất vùng để gom các đặc trưng của vùng đó từ khối đặc trưng toàn cục để tạo ra biểu diễn với độ dài cố định. SPP giải quyết vấn đề cắt hình ảnh trước khi đưa vào CNN với kích thước cố định.

Khác với R-CNN, SPPNet chỉ xử lý hình ảnh tại các lớp tích chập một lần, trong khi R-CNN xử lý hình ảnh tại các lớp tích chập ít nhất 2000 lần.

Vì vậy, SPPNet nhanh hơn và chính xác hơn nhiều so với R-CNN.

5.2.1 FAST REGION-BASED CONVOLUTIONAL NETWORK (FAST R-CNN)

So sánh Fast R-CNN với SPP-net, có thể thấy bộ phân loại SVM đã được loại bỏ, thay vào đó là một lớp hồi quy và phân loại được kết nối với mạng. VGGNet được sử dụng thay vì ZFNet, lớp gom vùng quan tâm (ROI) thay vì SPP.

Hai bổ sung chính đã cải thiện tốc độ phát hiện của nó:

- Thay vì chuyển tiếp các đề xuất vùng đến bộ trích chọn đặc trưng, phương pháp này trích chọn đặc trưng của ảnh trước khi đề xuất các vùng. Do đó, chỉ cần áp dụng một mạng nơ-ron tích chập (CNN) duy nhất cho toàn bộ ảnh thay vì 2000 mạng CNN cho 2000 vùng.
- Chuyển đổi SVM thành lớp softmax, mở rộng mạng nơ-ron thành một mô hình dự đoán thay vì xây dựng một mô hình mới.

Fast R-CNN sử dụng hàm mất mát đa nhiệm kết hợp các mất mát phân loại và hồi quy. Hàm mất mát phân loại được tính toán bằng cách sử dụng hàm mất mát log trên hai lớp. Hàm mất mát hồi quy được tính toán bằng cách sử dụng hàm mất mát L1 smooth.

5.2.2 FASTER REGION-BASED CONVOLUTIONAL NETWORK (FASTER R-CNN)

Ba thuật toán được đề cập ở trên, R-CNN, SPPNet và Fast R-CNN, đều dựa vào tìm kiếm chọn lọc để xác định các đề xuất vùng (region proposals). Tìm kiếm chọn lọc là một phương pháp chậm và tốn thời gian, ảnh hưởng đến hiệu suất của mạng và được chứng minh là nút thắt của toàn bộ quá trình.

Tác giả của Faster R-CNN đã đề xuất một khung cho việc phát hiện vật thể để thay thế thuật toán tìm kiếm chọn lọc và cho phép mạng tự phát hiện các đề xuất vùng. Họ đã phát triển một mạng đề xuất vùng (RPN) để tạo các đề xuất vùng trực tiếp, sau đó dự đoán các khung bao quanh (bounding boxes).

5.2.3 REGION-BASED FULLY CONVOLUTIONAL NETWORK (R-FCN)

Faster R-CNN vẫn chứa một số lớp fully connected riêng biệt của R-CNN, cần phải tính toán cho hàng trăm vùng đề xuất (proposal).

Region-based Fully Convolutional Network (R-FCN) là một cấu trúc kết hợp cả hai giai đoạn chính trong cùng một mô hình để tính đến cả việc phát hiện đối tượng và vị trí của nó một cách đồng thời. Nó chỉ chứa các lớp convolutional cung cấp backpropagation hoàn chỉnh cho việc huấn luyện và suy luận. Không giống như dòng R-CNN, các lớp FC sau ROI pooling đã bị loại bỏ. Sau ROI pooling, tất cả các đề xuất vùng này sẽ sử dụng cùng một bản đồ điểm (score map) để thực hiện average voting, một phép tính đơn giản. Do đó, không có lớp học nào sau lớp ROI; nói cách

khác, R-FCN nhanh hơn đáng kể so với Faster R-CNN và có mAP (mean Average Precision - độ chính xác trung bình) rất đáng tin cậy.

Hàm mất mát (loss function) cho R-FCN được xác định trên mỗi RoI và là tổng của mất mát cross-entropy và mất mát hồi quy khung (bounding box regression loss). Mất mát phân loại (Lcls) và mất mát hồi quy khung hình (Lreg) được sử dụng trong online hard example mining (OHEM).

5.2.4 FEATURE PYRAMID NETWORKS (FPN)

Mặc dù FPN (Feature Pyramid Network - Mạng Kim Tháp Đặc Trưng) không phải là một công cụ phát hiện vật thể riêng lẻ, nó là một công cụ dò tìm đặc trưng hoạt động kết hợp với các bộ phận phát hiện vật thể.

So với bộ trích xuất đặc trưng được sử dụng trong một số khung như Faster R-CNN, FPN tạo ra nhiều lớp bản đồ đặc trưng hơn, bản đồ đặc trưng đa tỷ lệ và thông tin chất lượng cao hơn so với kim tháp đặc trưng tiêu chuẩn được sử dụng để phát hiện vật thể. Sử dụng FPN cho phép chúng ta phát hiện các vật thể ở các kích thước khác nhau.

5.2.5 PANET

Mạng Tích hợp Đường dẫn (PANet) là một phương pháp chủ yếu được phát triển cho segmentation, nó chèn thêm một mạng tích hợp đường dẫn hướng lên trên FPN. PANet cho phép mạng quyết định các đặc trưng nào là hữu ích.

5.2.6 TRIDENTNET

Mô hình TridentNet đề xuất một cách tiếp cận để xử lý các biến thể kích thước trong việc phát hiện vật thể dựa trên việc tạo ra các bộ lọc tính năng theo kích thước cụ thể trong mạng bằng cách sử dụng sức mạnh biểu diễn thống nhất.

Họ xây dựng một kiến trúc nhánh đa song song và áp dụng huấn luyện nhận biết tỷ lệ, trong đó mỗi nhánh chia sẻ các tham số biến đổi giống nhau nhưng với các trường tiếp nhận (receptive fields) khác nhau. Mô hình áp dụng phương pháp suy luận nhanh chỉ với một nhánh chính để cải thiện hiệu suất của mô hình mà không cần sử dụng thêm các tham số và tính toán.

5.2.7 SPINENET

SpineNet là một mô hình phân loại và phát hiện vật thể sử dụng Tìm kiếm Kiến trúc Mạng Nơ-ron (NAS) để học, khác với các kiến trúc mã hóa-giải mã truyền thống với backbone giảm tỷ lệ dẫn đến việc tạo các đặc trưng đa tỷ lệ không hiệu quả.

Phương pháp đề xuất của SpineNet có một mạng thân cố định tiếp theo là các đặc trưng trung gian được hoán đổi tỷ lệ và các kết nối xuyên tỷ lệ.

5.2.8 COPY-PASTE

Các tác giả đã áp dụng chiến lược bổ sung dữ liệu sao chép-dán (copy-paste) và chứng minh tính hiệu quả của nó đối với việc phát hiện vật thể và phân vùng thể hiện (instance segmentation). Phương pháp sao chép-dán chọn ngẫu nhiên hai ảnh và áp dụng dịch chuyển tỷ lệ ngẫu nhiên (scale jittering) và lật ngang (horizontal flip). Nó tạo ra dữ liệu mới bằng cách dán các đối tượng từ ảnh này sang ảnh khác.

Các tác giả cung cấp phương pháp tự huấn luyện Sao chép-dán, trong đó một mô hình được giám sát được huấn luyện trên dữ liệu được dán nhãn, tạo ra các nhãn giả trên dữ liệu không được dán nhãn.

5.3 One-Stage Methods

Các bộ phát hiện dựa trên neo đơn giai đoạn (one-stage anchor-based detectors) được đặc trưng chủ yếu bởi hiệu quả tính toán và thời gian chạy. Các mô hình này phân loại và hồi quy trực tiếp các hộp neo được xác định trước thay vì sử dụng các vùng quan tâm (region of interest).

Thách thức chính gặp phải trong loại bộ phát hiện này là sự mất cân bằng giữa các mẫu dương tính và âm tính. Nhiều phương pháp và cơ chế đã được triển khai để khắc phục vấn đề này, chẳng hạn như tinh chỉnh và khớp neo, huấn luyện từ đầu, hợp nhất thông tin ngữ cảnh nhiều lớp và làm giàu và căn chỉnh đặc trưng. Các nghiên cứu khác tập trung vào việc phát triển các hàm mất mát mới và kiến trúc mới.

5.3.1 YOLOv2

YOLOv2, hay YOLO9000, được công bố vào năm 2017, là một mô hình phát hiện vật thể có khả năng phát hiện hơn 9.000 loại đối tượng trong thời gian thực.

Đối với YOLOv2, vị trí được xác định bởi hàm kích hoạt logistic, do đó giảm giá trị xuống còn giữa 0 và 1, so với YOLOv1 không có giới hạn về dự đoán vị trí. YOLOv2 dự đoán nhiều khung bao quanh (bounding box) trên mỗi ô lưới. Để tính toán tổn thất cho kết quả dương thực sự (true positive), chỉ một trong số chúng phải chịu trách nhiệm cho đối tượng. Vì mục đích này, khung bao có IoU (tỷ lệ giao nhau trên hợp nhất) cao nhất với dữ liệu thực được chọn.

Hàm mất mát (loss function) của YOLOv2 bao gồm ba phần: tìm tọa độ khung bao, dự đoán điểm khung bao và dự đoán điểm lớp. Tất cả đều là các hàm mất mát lỗi bình phương trung bình (Mean-Squared Error) và được điều chỉnh bởi một số siêu tham số hoặc điểm IoU giữa dự đoán và dữ liệu thực.

5.3.2 YOLOv3

YOLOv3 sử dụng phân loại đa nhãn và lớp softmax được thay thế bằng một bộ phân loại logistic độc lập để tính toán xác suất đầu vào thuộc về một nhãn cụ thể. Thay vì sử dụng lỗi bình phương trung bình để tính toán tổn thất phân loại, YOLOv3 áp dụng tổn thất entropy nhị phân cho mọi nhãn.

YOLOv3 thực hiện dự đoán ở ba tỷ lệ, chính xác bằng cách hạ mẫu kích thước ảnh đầu vào xuống lần lượt 32, 16 và 8. Nó sử dụng tổng cộng 9 khung neo (anchor box), mỗi tỷ lệ có 3 khung.

YOLOv3 dự đoán nhiều khung bao hơn YOLOv2. Đối với cùng một ảnh 416×416, YOLOv2 có $13 \times 13 \times 5 = 845$ khung; tại mỗi ô lưới, tổng cộng 5 khung được phát hiện bằng cách sử dụng 5 anchor box, trái ngược với YOLO v3, dự đoán các khung ở ba tỷ lệ riêng biệt, tổng cộng 10.647 khung được dự đoán cho một ảnh kích thước 416×416.

Hàm tổn thất của YOLOv3 được xác định từ ba khía cạnh: lỗi vị trí của khung bao, lỗi độ tin cậy của khung bao và lỗi dự đoán phân loại giữa ground truth (sự thật mặt đất) và các khung được dự đoán.

5.3.3 SSD

- Bộ Phát Hiện Vật Thể Bằng Khung Đơn Đa Kích Thước (SSD) là một khuôn khổ phát hiện vật thể được công bố sau R-CNN và YOLO. Nó được phát triển bởi W. Liu et al. để dự đoán các khung bao và xác suất lớp trong một quy trình một lần bằng cách sử dụng kiến trúc CNN đầu cuối (end-to-end).
- SSD cho phép phát hiện nhiều đối tượng trong ảnh chỉ trong một lần thay vì hai lần cần thiết cho các phương pháp mạng đề xuất vùng được liệt kê trong phần trước. Do đó, SSD tiết kiệm thời gian đáng kể so với các phương pháp dựa trên vùng.
- SSD sử dụng lấy mẫu âm (negative sampling) để xác định các dự đoán kém. Nó áp dụng kỹ thuật loại bỏ phi cực đại (non-maximal suppression) ở cuối mô hình, giống như YOLO, để giữ lại các khung thích hợp hơn. Sau đó, phương pháp đào HardNegative Mining (HNM) được áp dụng để đảm bảo đào tạo nhanh hơn và ổn định hơn. Chúng chọn các ví dụ âm theo giá trị tin cậy cao nhất được gán cho mỗi khung mặc định và sau đó chọn các giá trị cao để đảm bảo tỷ lệ âm và dương dưới 3: 1.
- Hàm mất mát của SSD kết hợp mất mát định vị và mất mát tin cậy. Mất mát định vị là sự không khớp giữa khung sự thật cơ bản (ground truth box) và khung bao được dự đoán. SSD chỉ phạt các dự đoán từ các khớp dương. Các khớp âm có thể được bỏ qua. Mất mát tin cậy là một mất mát softmax trên nhiều lớp tin cậy (c).

5.3.4 RETINANET

Bài báo RETINANET đóng góp đáng kể nhờ hàm mất mát mới được gọi là focal loss cho phân loại, giúp tăng độ chính xác đáng kể.

RetinaNet sử dụng hàm mất mát focal loss để giải quyết vấn đề mất cân bằng giữa các lớp trong quá trình huấn luyện. Hàm mất mát focal loss của RetinaNet giảm trọng số cho các ví dụ được phân loại chính xác, tập trung huấn luyện vào một tập hợp nhỏ các ví dụ khó và ngăn không cho nhiều trường hợp âm tính dễ dàng áp đảo bộ phát hiện trong quá trình huấn luyện.

5.3.5 MEGDET

MegDet là một mô hình giải quyết nhiệm vụ phát hiện vật thể từ khía cạnh kích thước mini-batch. Thay vì sử dụng kích thước mini-batch 16 như thông thường, tác giả đề xuất sử dụng kích thước lớn hơn là 256 trong quá trình huấn luyện. Để huấn luyện toàn bộ mạng trong thời gian phù hợp, họ sử dụng chuẩn hóa batch đa GPU với 128 GPU và chính sách tỷ lệ học tăng dần (warmup learning rate).

5.3.6 EFFICIENTDET

EfficientDet là một mô hình phát hiện vật thể dựa trên backbone EfficientNet được tiền huấn luyện [60], mạng đặc trưng lưỡng hướng có trọng số và kỹ thuật nhân rộng hợp chất được cá nhân hóa.

Mạng đặc trưng lưỡng hướng sử dụng các đặc trưng cấp độ 3 đến 7 từ EfficientNet và áp dụng hợp nhất đặc trưng lưỡng hướng theo hướng lên trên (top-down) và hướng xuống dưới (bottom-up). Trọng số của mạng lớp và mạng khung được chia sẻ giữa tất cả các cấp độ đặc trưng. EfficientDet sử dụng hàm focal loss cho object detection.

5.3.7 PAA

PAA, một mô hình dựa trên kỹ thuật mới để gán các điểm neo (anchor) dựa trên tối ưu hóa theo xác suất của phân bố xác suất, là viết tắt của cụm từ "gán neo theo xác suất" (probabilistic anchor assignment).

Mô hình này bao gồm việc tính điểm cho các điểm neo và xác định các mẫu dương tính và âm tính theo cách xác suất so với việc gán thách thức IoU theo kinh nghiệm, điều này khiến quá trình huấn luyện trở nên khó khăn và tốn thời gian hơn. Các tác giả đề xuất một phương pháp bỏ phiếu điểm cho hậu xử lý trong việc phát hiện đối tượng mật độ cao.

5.3.8 YOLOv5

YOLOv5 tập trung vào tốc độ suy luận và độ chính xác, sử dụng các mô hình phát hiện đối tượng được chia tỷ lệ kép được huấn luyện trên bộ dữ liệu COCO để kết hợp mô hình và Tăng cường Thời gian Kiểm tra.

YOLOv5 sử dụng một mạng nơ-ron tích chập (CNN) làm xương sống có tên CSPDarknet để tạo các đặc điểm hình ảnh. Các đặc điểm này được kết hợp trong phần cổ của mô hình, sử dụng một biến thể của PANet (Mạng tổng hợp đường dẫn), và được gửi đến phần đầu. Sau đó, phần đầu của mô hình sẽ giải thích các đặc điểm kết hợp để dự đoán lớp của một hình ảnh. Nó cũng sử dụng các khối dư và các khối dày đặc để cho phép luồng thông tin đến các lớp sâu nhất. Kiến trúc bao gồm ba phần: backbone, neck và head.

5.3.9 YOLOv7

YOLOv7 là một thuật toán nhận dạng vật thể thời gian thực nhanh hơn và chính xác hơn. Giống như Scaled YOLOv4, YOLOv7 không sử dụng các backbone được đào tạo sẵn trên ImageNet. Các weights của YOLOv7 được đào tạo bằng bộ dữ liệu COCO của Microsoft và không sử dụng bất kỳ bộ dữ liệu hoặc weights được đào tạo sẵn nào khác. Bài báo chính thức trình bày cách kiến trúc được cải tiến này vượt qua tất cả các phiên bản YOLO trước đó và tất cả các mô hình nhận dạng vật thể khác về tốc độ và độ chính xác. YOLOv7 cải thiện tốc độ và độ chính xác bằng cách giới thiệu một số cải cách về kiến trúc.

6 Các bộ phát hiện không neo

6.1 YOLOv1

YOLO có một cách tiếp cận khác đối với việc phát hiện vật thể. Nó chụp toàn bộ hình ảnh trong một lần duy nhất. Sau đó, nó dự đoán cả tọa độ của các khung bao cho hồi quy và xác suất lớp chỉ với một mạng lưới trong một đánh giá. Do đó, tên của nó là YOLO; bạn chỉ nhìn một lần. Sức mạnh của mô hình YOLO đảm bảo các dự đoán theo thời gian thực.

Hình ảnh đầu vào được chia thành một lưới các ô SxS để thực hiện phát hiện. Một ô lưới đơn được cho là dự đoán mọi đối tượng duy nhất trong hình ảnh và đây là nơi tâm của đối tượng rơi vào. Mỗi ô sẽ dự đoán B khung bao tiềm năng với mỗi giá trị xác suất lớp C của khung bao, với tổng cộng SxSxB khung bao.

Một thủ tục loại bỏ phi tối đa được áp dụng cho tất cả các ô còn lại, loại bỏ tất cả các phát hiện trùng lặp có thể và giữ lại các đối tượng chính xác nhất.

Hàm mất mát YOLOv1 được chia thành ba phần: phần chịu trách nhiệm tìm tọa độ khung bao, dự đoán điểm khung bao và dự đoán lớp. Hàm mất mát cuối cùng là tổng của ba phần này.

6.2 CORNERNET

CornerNet là một mô hình nhận dạng đối tượng sử dụng các điểm ảnh đặc trưng (keypoint) để xác định khung bao của đối tượng. Kỹ thuật này giúp cho mô hình không cần sử dụng các anchor box truyền thống thường thấy trong các bộ nhận dạng object khác.

Ngoài ra, các tác giả còn đề xuất một loại lớp pooling mới có tên là corner pooling, có mục đích định vị các góc của đối tượng một cách hiệu quả.

CornerNet sử dụng kỹ thuật nhúng kết hợp (associative embedding), trong đó mạng dự đoán các nhúng có độ tương đồng cao cho các điểm ảnh đặc trưng thuộc cùng một đối tượng và sử dụng hàm mất mát tương tự như triplet loss. Bên cạnh đó, bài báo còn đề xuất một biến thể mới của hàm mất mát focal loss, giúp điều chỉnh trọng số của từng anchor box một cách linh hoạt.

6.3 EXTREMENET

ExtremeNet sử dụng phương pháp tiếp cận từ dưới lên để nhận dạng đối tượng. Chúng sử dụng một mạng ước lượng điểm ảnh chuẩn để xác định điểm trung tâm của đối tượng và bốn điểm cực của nó: trên cùng, phải nhất, trái và dưới cùng. Bốn điểm cực kỳ quan trọng này được sử dụng làm hộp bao quanh đối tượng theo cách hoàn toàn hình học.

6.4 REPPOINTS

RepPoints là viết tắt của các điểm đại diện, một kỹ thuật biểu diễn đối tượng như một tập hợp các điểm mẫu. Vì các hộp bao quanh truyền thống cung cấp định vị và trích xuất thô, RepPoints sử dụng các điểm để định vị và nhận dạng đối tượng.

Kỹ thuật reppoint không sử dụng neo để lấy mẫu không gian của các hộp bao quanh. Thay vào đó, nó học cách xử lý tự động các mục tiêu nhận dạng và định vị mặt đất bằng cách giới hạn phạm vi không gian trong một đối tượng và xác định các khu vực cục bộ có liên quan về mặt ngữ nghĩa.

Các tác giả đề xuất mô hình phát hiện đối tượng RPDet dựa trên biểu diễn RepPoints kết hợp với tích chập biến dạng. Bài báo RepPoints mô tả hai tập RepPoints, một tập được điều khiển bởi riêng hàm mất khoảng cách điểm và tập còn lại được điều khiển bởi sự kết hợp của hàm mất khoảng cách điểm và hàm mất tâm.

6.5 FSAF

Bài báo đề xuất mô-đun FSAF (Feature Selective Anchor-Free - Chọn lọc Đặc trưng Không neo) để giải quyết hai vấn đề thường gặp trong bộ nhận dạng one-shot dựa trên neo với kim tự tháp đặc trưng: lựa chọn đặc trưng theo kinh nghiệm và lấy mẫu neo dựa trên trùng lặp.

Trong khi huấn luyện các nhánh không neo đa cấp, mô-đun FSAF áp dụng lựa chọn đặc trưng trực tuyến trong khi huấn luyện các nhánh này, cải thiện các đường baseline với chi phí suy luận nhỏ. Mỗi thể hiện được liên kết với cấp độ đặc trưng phù hợp để tối ưu hóa mạng. Mô hình mã hóa các thể hiện này theo phương pháp không neo để học các tham số cho phân loại và hồi quy.

6.6 Fully Convolutional One-Stage Detection (FCOS)

FCOS dựa trên kỹ thuật trên từng pixel để phát hiện đối tượng, tránh tất cả các siêu tham số và độ phức tạp của việc chồng chéo trong quá trình huấn luyện. FCOS sử dụng Non-maximum suppression (NMS) để xử lý hậu kỳ và lọc các khung hình giới hạn, giúp cải thiện độ chính xác.

Các tác giả sử dụng FCOS làm mạng đề xuất vùng trong bộ phát hiện đối tượng hai giai đoạn, chẳng hạn như Faster R-CNN. Hàm mất mát được sử dụng trong FCOS kết hợp ba mất mát: mất mát tiêu điểm cho phân loại, mất mát IoU cho hồi quy và mất mát tâm.

6.7 Adaptive Training Sample Selection (ATSS)

Chương trình được đề xuất có thể tự động xác định các mẫu huấn luyện dương và âm dựa trên các đặc điểm thống kê của đối tượng. Các mẫu dương và âm được sử dụng để phân loại, trong khi các mẫu âm được sử dụng để hồi quy. Kỹ thuật Lựa chọn Mẫu Huấn luyện Thích ứng (ATSS) không có các tham số siêu so với các kỹ thuật trước đó. Các tác giả cũng đề cập rằng việc lát nhiều neo trên mỗi vị trí là rất quan trọng trong quá trình phát hiện đối tượng.

Phương pháp Lựa chọn Mẫu Huấn luyện Thích ứng (ATSS) tự động chọn các mẫu dương và âm dựa trên các đặc điểm của đối tượng bằng cách sử dụng các đặc điểm thống kê để tính toán các ngưỡng động.

6.8 OTA

Các tác giả đề xuất một kỹ thuật Gán Vận Chuyển Tối ưu (Optimal Transport Assignment - OTA) dựa trên lý thuyết tối ưu hóa.

Kỹ thuật này sử dụng phương pháp vận chuyển nhãn hiệu hiệu quả về chi phí từ các đối tượng và nền thực tế tới các điểm neo (anchor) bằng cách sử dụng Lặp Sinkhorn-Knopp. Dựa trên các giá trị Tỉ lệ Giao Thoa trên Liên Kết (Intersection-over-Union - IoU) giữa các khung bao dự đoán và mỗi thực tế cơ sở, họ đưa ra một chiến lược ước tính đơn giản mới để xác định các nhãn hiệu tích cực mà mỗi thực tế cơ sở cần.

OTA có thể xử lý việc gán các điểm neo mơ hồ bằng cách gán chúng thủ công bằng các quy tắc thủ công trước khi áp dụng việc gán vận chuyển tối ưu.

Hàm mất mát Gán Vận Chuyển Tối ưu (OTA) là một thủ tục gán nhãn trong việc phát hiện đối tượng, nhằm vận chuyển nhãn hiệu từ các đối tượng thực tế và gán chúng cho các khung neo.

6.9 Dynamic Smooth Label Assignment (DSLA)

Chương trình cải thiện quá trình chuyển đổi giữa các mẫu dương và mẫu âm bằng cách cải thiện biểu diễn tâm điểm được đề xuất trong FCOS và cung cấp chiến lược nới lỏng khoảng.

Giao nhau của Liên hợp được kết hợp với nhãn mịn có giá trị từ 0 đến 1 để giám sát nhánh phân loại, được hợp nhất với nhánh ước tính chất lượng, dẫn đến mô hình không neo được đơn giản hóa hơn với chất lượng định vị tốt. IoU được dự đoán động trong quá trình huấn luyện.

DSLA cải thiện hiệu suất của các mô hình phát hiện với các thuật toán gán nhãn thích ứng và giảm thiệt hại khung giới hạn cho các mẫu dương, cho biết thêm nhiều mẫu có hộp dự đoán chất lượng cao hơn được chọn làm dương tính.

6.10 YOLOv8

YOLOv8 dựa trên thành công của các phiên bản YOLO trước đó và giới thiệu các tính năng mới, cải tiến để nâng cao hiệu suất và tính linh hoạt hơn nữa.

Nó có thể được huấn luyện trên các tập dữ liệu lớn và chạy trên các nền tảng phần cứng khác nhau, từ CPU đến GPU. Một tính năng chính của YOLOv8 là khả năng mở rộng. Nó hỗ trợ tất cả các phiên bản YOLO trước đó, giúp dễ dàng chuyển đổi giữa các phiên bản khác nhau và so sánh hiệu suất của chúng. Điều này làm cho YOLOv8 trở thành lựa chọn lý tưởng cho những người dùng muốn tận dụng công nghệ YOLO mới nhất trong khi vẫn có thể sử dụng các mô hình YOLO hiện có của họ. YOLOv8 bao gồm nhiều tính năng về kiến trúc và tiện lợi cho nhà phát triển, làm cho nó trở thành lựa chọn hấp dẫn cho nhiều tác vụ phát hiện đối tượng và phân vùng ảnh.

7 Các bộ phát hiện dựa trên Transformer

7.1 DETR

DETR (DEtection TRansformer) là mô hình phát hiện đối tượng đầu tiên dựa trên kiến trúc Transformer.

Mô hình này kết hợp một mạng nơ-ron tích chập (CNN) làm cơ sở và một kiến trúc Transformer. Mạng CNN sử dụng ResNet làm cơ sở để trích xuất đặc trưng từ ảnh, sau đó đặc trưng này được định dạng và mã hóa vị trí trước khi đưa vào kiến trúc Transformer. Kiến trúc Transformer trong DETR bao gồm một bộ mã hóa và một bộ giải mã, loại bỏ các phần tử như mô-đun tạo neo.

DETR sử dụng hàm mất mát bipartite matching để tối ưu hoá việc khớp giữa đầu ra của mô hình và dữ liệu thực tế. DETR tạo ra một số lượng dự đoán cố định, mỗi dự đoán được tính toán song song. Mô hình DETR tiếp cận phát hiện đối tượng bằng cách dự đoán trực tiếp tập hợp các đối tượng và sử dụng một hàm mất mát toàn cục dựa trên tập hợp.

7.2 SMCA

Được giới thiệu vào năm 2021, SMCA là một biến thể nhằm cải thiện hiệu suất hội tụ của DETR.

SMCA đề xuất cơ chế Co-Attention được điều chỉnh không gian (Spatially Modulated Co-Attention) để cải thiện khả năng hội tụ của DETR. Mô hình SMCA thay thế cơ chế co-attention trong bộ giải mã DETR bằng cơ chế co-attention nhận biết vị trí. Điều này giúp hạn chế ảnh hưởng của co-attention ở các vị trí không cần thiết gần với các vị trí khung hình ban đầu.

Để huấn luyện DETR từ đầu, cần khoảng 500 epochs để đạt hiệu suất tốt nhất. SMCA chỉ cần 108 epochs để huấn luyện và đạt được hiệu suất tốt hơn so với DETR gốc, đồng thời thể hiện tiềm năng trong việc xử lý thông tin toàn cầu.

7.3 ANCHOR DETR

Một mô hình phát hiện đối tượng dựa trên Transformer mới có tên là Anchor DETR được đề xuất với thiết kế truy vấn mới.

Anchor DETR sử dụng các điểm neo để giải quyết vấn đề về ý nghĩa vật lý của các truy vấn đối tượng. Điều này giúp mô hình tập trung vào các đối tượng gần các điểm neo. Mô hình này có khả năng dự đoán nhiều đối tượng tại cùng một vị trí.

Để tối ưu hóa độ phức tạp, Anchor DETR sử dụng một biến thể của cơ chế chú ý gọi là Chú ý Tách Lọc Hàng-Cột để giảm bớt chi phí bộ nhớ mà vẫn duy trì độ chính xác.

7.4 DESTR

"DESTR" đề xuất một giải pháp cho một số vấn đề trước đây của Transformer, bao gồm cơ chế self-attention và cross-attention, cùng với cách khởi tạo truy vấn nội dung của giải mã Transformer.

Tác giả đề xuất một biến thể mới được gọi là Attention Phân tách Phát hiện (Detection Split Transformer), phương pháp này chia ước tính nhúng nội dung của cơ chế attention cross thành hai phần độc lập, một phần cho phân loại và một phần cho nhúng hồi quy khung. Bằng cách này, họ cho phép mỗi cơ chế attention cross xử lý nhiệm vụ riêng của nó. Đối với việc khởi tạo truy vấn nội dung, họ sử dụng một bộ phận dò mini để học nội dung và khởi tạo nhúng vị trí của giải mã. Bộ phận này được trang bị các thành phần cho nhúng phân loại và hồi quy. Cuối cùng, để tính toán các cặp truy vấn đối tượng liền kề trong giải mã, họ bổ sung cho cơ chế attention tự bằng bối cảnh không gian của truy vấn khác trong cặp.

8 Tóm tắt

Có bốn phương pháp chính trong việc phát hiện đối tượng: phát hiện dựa trên anchor hai giai đoạn, phát hiện dựa trên anchor một giai đoạn, phát hiện không dùng anchor, và phát hiện dùng transformer.

Chúng tôi tổng quan mỗi phương pháp bằng cách đánh giá một số mô hình và phương pháp:

- Phát hiện dựa trên anchor hai giai đoạn: R-CNN, SPPNet, FAST R-CNN, FASTER R-CNN, R-FCN, FPN, PANET, TRIDENTNET, SPINENET,...
- Phát hiện dựa trên anchor một giai đoạn: YOLOv2, YOLOv3, SSD, RETINANET, MEGDET, EFFICIENTDET, PAA, YOLOv5, YOLOv7,...
- Phát hiện không dùng anchor: YOLOv1, CORNERNET, EXTREMENET, REPPOINTS, FSAF, FCOS, ATSS, OTA, DSLA, YOLOv8,...
- Phát hiện dùng transformer: DETR, SMCA, ANCHOR-DETR, DESTR,...

Nhìn chung, tương lai của việc phát hiện đối tượng sử dụng học sâu là rất sáng sủa, với nhiều tiến bộ thú vị cho nghiên cứu trong tương lai.

AI VIET NAM – RESEARCH TEAM

Object Detection Using Deep Learning, CNNs and Vision Transformers: A Review

Ngày 30 tháng 3 năm 2024

Date of publication:	13/04/2023
Authors:	Ayoub Benali Amjoud and Mustapha Amrouch
Sources:	Research Excellence Awards Program 2019–2022 under Grant 5UIZ2019
Keywords:	Object detection, deep learning, review, convolutional neural networks, transformers, survey, neural networks.
Summary by:	Nhi Ng Thao

1 Introduction:

- Talk about the importance and methods of object recognition in the field of computer vision. Present and point out the wide applications of Object Detection in many different fields. Including in the fields of: autonomous vehicles, surveillance, medical imaging and even robotics.
- Outlines the development and research of object recognition, with an emphasis on the transition from traditional models to deep learning-based methods in 2014.
- Emphasizes the significant impact of Deep Learning Neural Networks and Convolutional Neural Networks in improving object recognition. Enables advances in tasks such as classification. Through this, we can see that deep learning models are playing an important role in promoting object recognition and semantic understanding of images and videos.

2 Overview:

- **DATASET & EVALUATION METRICS**
  - DATASET **PASCALVOC**: A dataset of 10,000 images for object detection, with 20 different object types. **MS -COCO**: The dataset includes more than 200,000 images and 80 object types. **ILSRVC**: The dataset includes more than 1 million labeled images with about 600 objects. **OPEN IMAGE**: The dataset has about 9.3 million labeled images with about 600 objects.
  - EVALUATION METRICS
  - **Intersection over Union (IoU)**: Measures detection quality by calculating the difference between the actual and predicted bounding box.

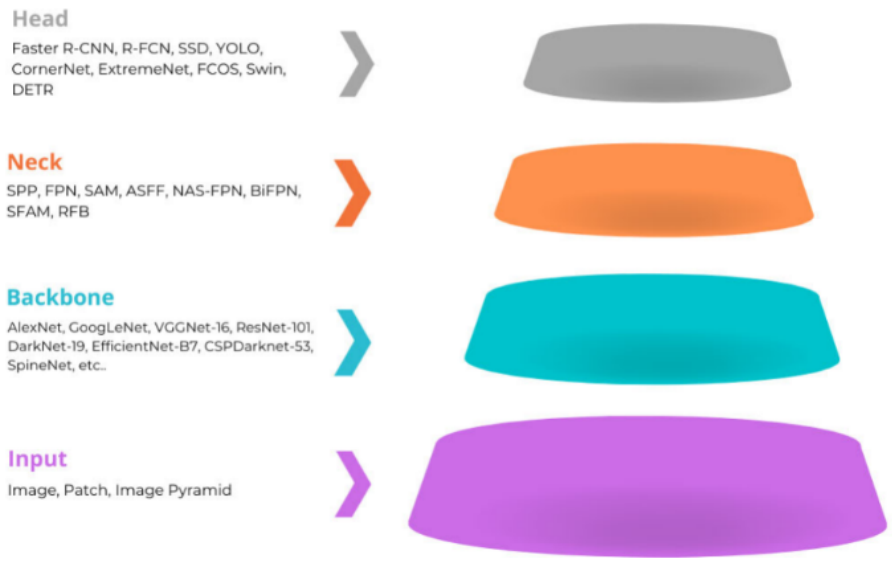
$$\text{IoU} = \frac{\text{Area of Intersection}}{\text{Area of Union}} \tag{1}$$

Mean Average precision (mAP): Average precision of all classes. **Mean Average Recall (mAR)**: The average value of recall for all classes.

Ref	Dataset	Evaluation metric
[53]	The PASCAL VOC Challenge	mAP
[56]	The COCO Object Detection Challenge	mAP, mAR

Hình 1: Overview of methods, datasets and evaluation metrics

- **Input:** a picture, patch, or a large number of images or videos. **Backbone:** can be a CNN like VGG, ResNet, etc. **The neck:** Top backbone **Head:** is divided into two types: dense prediction and sparse prediction.



Hình 2: Components in the OD model

3 Backbone

Plays an important role in extracting features from images before moving on to the next steps, such as object positioning. Many CNN models are commonly used in object detection, including pre-trained models such as VGGNet, ResNet, and EfficientNet.

- **AlexNet**
  - Includes 8 layers: 5 convolutional layers, 2 fully connected (FC) layers and 1 softmax output layer with 1000 layers.
  - Use ReLU activation function and local normalization layers.
- **VGGNet**
  - There are 5 convolutional layers followed by 3 FC layers.

  - Use a small convolution filter (3x3) and have many layers (16-19 layers).
- **ResNet**
  - Famous for its ability to train deep networks without the problem of gradient loss.
  - Introduces residual connections, allowing the network to learn residual mappings instead of original mappings.
  - Use residual blocks containing multiple convolution layers and batch normalization.
- **Inception-ResNet**
  - Integrate residual connections into the Inception architecture to improve gradient flow and enable deeper network training.
  - Use Inception blocks to extract features at different sizes, combined with residual connections.
- **EfficientNet**
  - Uses compound scaling to balance model size with accuracy and computational efficiency.
  - Using MBConv blocks, combine depth and point convolution layers.
- **GoogLeNet**
  - Developed by Google, there are 22 layers with 27 pooling layers, organized into 9 Inception blocks.
  - Use Inception blocks to extract features at different sizes, followed by global average pooling.
- **Other backbones:**
  - CSP-ResNeXt: Apply CSP (Cross Stage Partial) method to ResNeXt.
  - DenseNet: Integrates dense connections between layers via Dense Block.
  - SENet: Use squeeze-and-excitation blocks to adaptively recalibrate feature maps.
  - Hourglass: Uses bottom-up and top-down iterative processing to capture detailed and comprehensive features.
  - SpineNet: Has a sparse backbone architecture with intermediate features arranged at scale.
  - CSP-Darknet: Use CSPNet strategy to partition feature maps.
  - ConvNeXt: Built entirely from standard ConvNet blocks.

4 Data Augmentation

During training, the models use many different learning strategies such as fine-tuning, data augmentation, cascade learning and imbalance sampling. sampling). These strategies help the model perform efficiently to improve accuracy and execution time for both positioning and classification tasks.

- **COLOR SPACE**
  - Select a single color channel and add a zero value matrix to the remaining channels.
  - Change RGB matrix values to adjust contrast, brightness, and saturation.
- **Rotation**
  - Rotate the image left or right within an angle range of 1° to 359°.
  - Note the rotation angle to avoid losing the label value of the image.

- **Translation**
  - Move the image up, down, left or right.
  - Fills the empty space after the shift with a constant value (0 or 255) or random noise.
- **Cropping**
  - Crop the center part of the photo.
  - Random crop to reduce image size.
- **Kernel filters**
  - Apply a Gaussian blur filter to create a blurred image.
  - Use a vertical or horizontal edge filter to create sharper photos.
- **Random erasing**
  - Randomly select an area in the image and mask it using the average pixel value (0 or 255).
  - Helps the model avoid occlusion and overfitting.

5 Anchor-based detectors

Anchor boxes are predefined bounding boxes with diverse sizes and ratios, corresponding to objects that may appear in the image. Instead of directly predicting bounding frames, the network predicts probabilities and other attributes such as background and intersection with anchor frames. Anchor frames help the network detect multiple objects of different sizes and positions in the image.

(a) Generate thousands of candidate anchor boxes that best describe the size, position, and shape of objects.

(b) Predict the offset for each bounding box.

(c) Calculate the loss function for each anchor box based on real data.

(d) For each anchor box, calculate the Intersection Over Union (IOU) to check which object's bounding box has the largest IOU.

(e) When the confidence is greater than 0.5, inform the anchor box that it needs to detect the object with the highest IOU. And put this prediction into the loss function.

(f) As already mentioned, if this probability is only marginally less than 0.5, we will instruct the anchor box not to learn from this sample because the prediction is ambiguous; On the other hand, if the probability is significantly less than 0.5, then the anchor box is likely to predict that no object is present.

(g) Finally, by using this procedure, we ensure that the model only learns to recognize real objects.

However, anchors require careful design and application in object detection frameworks.

- In anchor design, the coverage ratio of the instance's location space is one of the most important factors. To ensure a good recall rate, the anchors are carefully designed based on statistics calculated from the training/validation data set.
- Certainly, some design choices based on a particular data set may not be applicable to other applications, which affects the generality of the model.

- During the learning phase, anchor-based methods depend on the Intersection over Union (IoU) ratio to identify positive/negative samples, thus increasing the mathematics and hyper-parameters for the object detection system.

Anchor-based object detection frameworks are generally divided into two groups: two-stage proposition-based detectors and one-stage non-proposition (proposition-free) methods.

5.1 Two-Stage Methods

5.1.1 REGION-BASED CONVOLUTIONAL NEURAL NETWORKS (R-CNN)

Developed by R. Girshick et al. in 2014, the R-CNN model proposed multiple frames (boxes) in the image and checked whether one of them contained an object. These frames are called regions. Selective Search was used to extract 2000 candidate regions; These candidate regions are cropped and resized to fit the input of the CNN feature extractor, extracting a 4096-dimensional feature vector that is passed into several classifiers for class prediction. SVMs are assigned to each class to classify the appearance of objects.

R-CNN involves three simple steps:

(a) By proposing about 2000 object boxes (candidate boxes), using a selective search algorithm to scan the input image and detect objects that may appear in the image.

(b) For each candidate box, we use a convolutional neural network (CNN) to extract features.

(c) The results of each convolutional neural network (CNN) are passed to an SVM for classification and to a linear regressor to refine the object's bounding frame.

5.2 SPATIAL PYRAMID POOLING NETWORK (SPPNet)

SPPNet implements a special CNN network called Spatial Pyramid Pooling (SPP) between the convolutional layers and the fully connected layers.

Similar to R-CNN, SPPNet uses a selection algorithm to generate approximately 2,000 region proposals for each image. Then, it uses ZFNet only once to extract features directly from the entire image. At the final convolutional layer, the feature regions identified by each region proposal pass through the SPP layer, followed by the fully connected layer.

SPPNet uses SPP for every region proposal to gather the region's features from the global feature block to create a fixed-length representation. SPP solves the problem of cropping the image before feeding it into the CNN with a fixed size.

Unlike R-CNN, SPPNet only processes images at convolutional layers once, while R-CNN processes images at convolutional layers at least 2000 times.

So, SPPNet is much faster and more accurate than R-CNN.

5.2.1 FAST REGION-BASED CONVOLUTIONAL NETWORK (FAST R-CNN)

Comparing Fast R-CNN with SPP-net, it can be seen that the SVM classifier has been removed, replaced by a regression and classification layer connected to the network. VGGNet is used instead of ZFNet, region of interest (ROI) pooling layer instead of SPP.

Two major additions have improved its detection speed:

- Instead of forwarding region proposals to the feature extractor, this method extracts image features before proposing regions. Therefore, only a single convolutional neural network (CNN) needs to be applied to the entire image instead of 2000 CNN networks for 2000 regions.
- Convert the SVM to a softmax layer, extending the neural network into a prediction model instead of building a new model.

Fast R-CNN uses a multitask loss function that combines classification and regression losses. The classification loss function is computed using the log loss function on the two classes. The regression loss function is computed using the L1 smooth loss function.

5.2.2 FASTER REGION-BASED CONVOLUTIONAL NETWORK (FASTER R-CNN)

The three algorithms mentioned above, R-CNN, SPPNet and Fast R-CNN, all rely on selective search to identify region proposals. Selective search is a slow and time-consuming method that affects network performance and proves to be the bottleneck of the entire process.

The authors of Faster R-CNN proposed a framework for object detection that replaces the selective search algorithm and allows the network to detect region proposals on its own. They developed a region proposal network (RPN) to generate region proposals directly, then predict bounding boxes.

5.2.3 REGION-BASED FULLY CONVOLUTIONAL NETWORK (R-FCN)

Faster R-CNN still contains several separate fully connected layers of R-CNN, which need to be calculated for hundreds of proposal regions.

Region-based Fully Convolutional Network (R-FCN) is an architecture that combines both main stages in the same model to take into account both object detection and its location simultaneously. It contains only convolutional layers that provide complete backpropagation for training and inference. Unlike the R-CNN series, the FC layers after ROI pooling have been removed. After ROI pooling, all these regional proposals will use the same score map to perform average voting, a simple calculation. Therefore, there is no class after the ROI class; In other words, R-FCN is significantly faster than Faster R-CNN and has a very reliable mAP (mean Average Precision).

The loss function for R-FCN is defined per RoI and is the sum of cross-entropy loss and bounding box regression loss. Classification loss (Lcls) and frame regression loss (Lreg) are used in online hard example mining (OHEM).

5.2.4 FEATURE PYRAMID NETWORKS (FPN)

Although FPN (Feature Pyramid Network) is not a stand-alone object detector, it is a feature detector that works in conjunction with object detectors.

Compared to the feature extractors used in some frameworks such as Faster R-CNN, FPN generates more feature map layers, multi-scale feature maps, and higher quality information than feature pyramids. Standard characteristic used for object detection. Using FPN allows us to detect objects of different sizes.

5.2.5 PANET

Path Integration Network (PANet) is a method mainly developed for segmentation, which inserts an upstream path integration network on top of the FPN. PANet allows the network to decide which features are useful.

5.2.6 TRIDENTNET

The TridentNet model proposes an approach to handle size variations in object detection based on creating size-specific feature filters in the network using unified representation power. .

They built a multi-parallel branch architecture and applied rate-aware training, where each branch shares the same transformation parameters but with different receptive fields. The model adopts a fast inference method with only one main branch to improve the performance of the model without using additional parameters and calculations.

5.2.7 SPINENET

SpineNet is an object detection and classification model that uses Neural Network Architecture Search (NAS) for learning, different from traditional encoder-decoder architectures with a scaled-down backbone leading to the creation of Multi-scale features are not effective.

SpineNet's proposed method has a fixed trunk network followed by scale-swapped intermediate features and cross-scale connections.

5.2.8 COPY-PASTE

The authors applied the copy-paste data augmentation strategy and demonstrated its effectiveness for object detection and instance segmentation. The copy-paste method randomly selects two images and applies scale jittering and horizontal flip. It creates new data by pasting objects from one image to another.

The authors provide a Copy-paste self-training method, where a supervised model is trained on labeled data, generating pseudo-labels on the unlabeled data.

5.3 One-Stage Methods

One-stage anchor-based detectors are mainly characterized by computational efficiency and running time. These models classify and regress predefined anchor boxes directly instead of using regions of interest.

The main challenge encountered in this type of detector is the imbalance between positive and negative samples. Many methods and mechanisms have been implemented to overcome this problem, such as anchor refinement and matching, training from scratch, multi-layer context information fusion, and feature enrichment and alignment. Other research focuses on developing new loss functions and new architectures.

5.3.1 YOLOv2

YOLOv2, or YOLO9000, announced in 2017, is an object detection model capable of detecting more than 9,000 types of objects in real time.

For YOLOv2, the location is determined by the logistic activation function, thus reducing the value to between 0 and 1, compared to YOLOv1 which has no limit on location prediction. YOLOv2 predicts multiple bounding boxes per grid cell. To calculate the true positive loss, only one of them must be responsible for the object. For this purpose, the bounding frame with the highest IoU (intersection-to-union ratio) with real data is selected.

The loss function of YOLOv2 includes three parts: finding bounding box coordinates, predicting bounding point points, and predicting class points. All are Mean-Squared Error loss functions and are tuned by some hyperparameter or IoU point between prediction and real data.

5.3.2 YOLOv3

YOLOv3 uses multi-label classification and the softmax layer is replaced by an independent logistic classifier to calculate the probability that the input belongs to a particular label. Instead of using mean squared error to calculate classification loss, YOLOv3 applies a binary entropy loss to every label.

YOLOv3 performs prediction at three scales, accurately by downsampling the input image size by 32, 16 and 8 respectively. It uses a total of 9 anchor boxes, 3 frames at each scale.

YOLOv3 predicts more bounding frames than YOLOv2. For the same 416×416 image, YOLOv2 has $13 \times 13 \times 5 = 845$ frames; At each grid cell, a total of 5 frames are detected using 5 anchor boxes, in contrast to YOLO v3, which predicts frames at three separate scales, a total of 10,647 frames are predicted for an image of size 416×416.

The loss function of YOLOv3 is determined from three aspects: bounding frame position error, bounding frame reliability error, and classification prediction error between ground truth and predicted frames.

5.3.3 SSD

- Multi-Size Single Frame Object Detector (SSD) is an object detection framework published after R-CNN and YOLO. It was developed by W. Liu et al. to predict bounding boxes and class probabilities in a one-shot process using an end-to-end CNN architecture.
- SSD allows detecting multiple objects in an image in just one pass instead of the two passes required for the region proposal network methods listed in the previous section. Therefore, SSD saves significant time compared to zone-based methods.
- SSD uses negative sampling to identify poor predictions. It applies non-maximal suppression at the end of the model, like YOLO, to retain more relevant frames. Then, the HardNegative Mining (HNM) mining method is applied to ensure faster and more stable training. They select negative examples according to the highest confidence value assigned to each default frame and then select high values to ensure a negative to positive ratio below 3:1.
- SSD's loss function combines locality loss and reliability loss. Localization loss is the mismatch between the ground truth box and the predicted bounding box. SSD only penalizes predictions from positive matches. Consonants can be omitted. The trust loss is a softmax loss over multiple trust layers (c).

5.3.4 RETINANET

The RETINANET paper makes a significant contribution thanks to a new loss function called focal loss for classification, which significantly increases accuracy.

RetinaNet uses a focal loss function to solve the problem of imbalance between layers during training. RetinaNet's focal loss function downweights correctly classified examples, focusing training on a small set of difficult examples and preventing many negative cases from easily overwhelming the detector in training process.

5.3.5 MEGDET

MegDet is a model that addresses the task of object detection from a mini-batch size perspective. Instead of using the usual mini-batch size of 16, the author suggests using a larger size of 256 during training. To train the entire network in a reasonable time, they used multi-GPU batch normalization with 128 GPUs and a warmup learning rate policy.

5.3.6 EFFICIENTDET

EfficientDet is an object detection model based on the pre-trained EfficientNet backbone [60], weighted bidirectional feature network, and personalized compound replication technique.

The bidirectional feature network uses level 3 to 7 features from EfficientNet and applies bidirectional feature fusion in the top-down and bottom-up directions. The weights of the layer network and frame network are shared among all feature levels. EfficientDet uses the focal loss function for object detection.

5.3.7 PAA

PAA, a model based on a new technique for assigning anchor points based on probabilistic optimization of a probability distribution, stands for "probabilistic anchor assignment". .

This model involves scoring anchor points and identifying positive and negative samples in a probabilistic manner versus assigning the IoU challenge empirically, which makes the training process difficult and expensive. more time. The authors propose a point voting method for post-processing in high-density object detection.

5.3.8 YOLOv5

YOLOv5 focuses on inference speed and accuracy, using dual-scaled object detection models trained on the COCO dataset for model fusion and Test Time Enhancement.

YOLOv5 uses a convolutional neural network (CNN) as its backbone called CSPDarknet to generate image features. These features are combined in the neck of the model, using a variant of PANet (Path Aggregation Network), and sent to the head. The first part of the model then interprets the combined features to predict the class of an image. It also uses residual blocks and dense blocks to enable information flow to the deepest layers. The architecture consists of three parts: backbone, neck and head.

5.3.9 YOLOv7

YOLOv7 is a faster and more accurate real-time object recognition algorithm. Like Scaled YOLOv4, YOLOv7 does not use pre-trained backbones on ImageNet. YOLOv7's weights are trained using Microsoft's COCO dataset and do not use any other pre-trained datasets or weights. The paper formally demonstrates how this improved architecture surpasses all previous versions of YOLO and all other object recognition models in terms of speed and accuracy. YOLOv7 improves speed and accuracy by introducing several architectural innovations.

6 Unanchored Detectors

6.1 YOLOv1

YOLO takes a different approach to object detection. It captures the entire image in a single shot. It then predicts both the coordinates of the bounding boxes for regression and the class probabilities with just one mesh in one evaluation. Hence, its name is YOLO; you only look once. The power of the YOLO model ensures real-time predictions.

The input image is divided into a grid of SxS cells to perform detection. A single grid cell is said to predict every single object in the image, and this is where the center of the object falls. Each cell predicts B potential bounding frames for each bounding box's class C probability value, for a total of SxSxB bounding frames.

A maximum phi elimination procedure is applied to all remaining cells, removing all possible duplicate detections and retaining the most accurate features.

The loss function YOLOv1 is divided into three parts: the part responsible for finding the bounding coordinates, predicting the bounding point, and predicting the class. The final loss function is the sum of these three parts.

6.2 CORNERNET

CornerNet is an object recognition model that uses characteristic pixels (keypoints) to determine the object's bounding frame. This technique eliminates the need for the model to use traditional anchor boxes commonly found in other object identifiers.

In addition, the authors also propose a new type of pooling layer called corner pooling, which aims to effectively locate the corners of objects.

CornerNet uses an associative embedding technique, in which the network predicts highly similar embeddings for feature pixels belonging to the same object and uses a loss function similar to triplet loss. Besides, the article also proposes a new variation of the focal loss function, which helps to adjust the weight of each anchor box flexibly.

6.3 EXTREMENET

ExtremeNet uses a bottom-up approach for object recognition. They use a standard pixel estimation network to determine the object's center point and its four extreme points: top, right most, left, and bottom. These four extremely important points are used as boxes that surround the object in a purely geometric manner.

6.4 REPPOINTS

RepPoints stands for representative points, a technique for representing an object as a set of sample points. Since traditional bounding boxes provide coarse positioning and extraction, RepPoints use points to locate and recognize objects.

The reppoint technique does not use anchors to sample the space of bounding boxes. Instead, it learns to automatically process recognition and ground positioning targets by limiting the spatial extent within an object and identifying semantically related local regions.

The authors propose the RPDet object detection model based on RepPoints representation combined with deformation convolution. The RepPoints paper describes two sets of RepPoints, one driven by the point distance loss function alone and the other set driven by the combination of the point distance loss function and the center loss function.

6.5 FSAF

This article proposes the FSAF (Feature Selective Anchor-Free) module to solve two common problems in anchor-based one-shot recognizers with feature pyramids: feature selection. Empirical sampling and duplicate-based anchor sampling.

While training multi-level unanchored branches, the FSAF module applies online feature selection while training these branches, improving the baselines at a small inference cost. Each instance is associated with the appropriate feature level to optimize the network. The model encodes these instances in an anchorless manner to learn parameters for classification and regression.

6.6 Fully Convolutional One-Stage Detection (FCOS)

FCOS relies on a per-pixel technique for object detection, avoiding all hyperparameters and the complexity of overlapping during training. FCOS uses Non-maximum suppression (NMS) for post-processing and filtering of limited frames, which improves accuracy.

The authors use FCOS as a region proposal network in a two-stage object detector, such as Faster R-CNN. The loss function used in FCOS combines three losses: focal loss for classification, IoU loss for regression, and centroid loss.

6.7 Adaptive Training Sample Selection (ATSS)

The proposed program can automatically identify positive and negative training samples based on the statistical characteristics of the subjects. Positive and negative samples are used for classification, while negative samples are used for regression. The Adaptive Training Sample Selection (ATSS) technique has no hyper parameters compared to previous techniques. The authors also mentioned that tiling multiple anchors per location is important in the object detection process.

The Adaptive Training Sample Selection (ATSS) method automatically selects positive and negative samples based on subject characteristics by using statistical features to calculate dynamic thresholds.

6.8 OTA

The authors propose an Optimal Transport Assignment (OTA) technique based on optimization theory.

This technique uses a cost-effective method of transporting marks from realistic objects and backgrounds to anchor points using Sinkhorn-Knopp Iteration. Based on the Intersection-over-Union (IoU) values between the predicted bounding boxes and each ground truth, they present a new simple estimation strategy to determine the labels. positive signal that each ground reality needs.

OTA can handle the assignment of ambiguous anchor points by assigning them manually using manual rules before applying the optimal transport assignment.

The Optimal Transport Assignment (OTA) loss function is a label assignment procedure in object detection that transports labels from real objects and assigns them to anchor frames.

6.9 Dynamic Smooth Label Assignment (DSLA)

The program improves the transition between positive and negative samples by improving the centroid representation proposed in FCOS and providing an interval relaxation strategy.

The intersection of the Conjugate is combined with a smooth label with a value between 0 and 1 to monitor the classification branch, which is merged with the quality estimation branch, resulting in a more simplified unanchored model with quantified quality good taste. IoU is dynamically predicted during training.

DSLA improves the performance of detection models with adaptive labeling algorithms and reduces bounding box loss for positive samples, indicating that more samples with higher quality prediction boxes are selected as positives.

6.10 YOLOv8

YOLOv8 builds on the success of previous YOLO versions and introduces new, improved features to further enhance performance and flexibility.

It can be trained on large data sets and run on different hardware platforms, from CPUs to GPUs. A key feature of YOLOv8 is extensibility. It supports all previous YOLO versions, making it easy to switch between different versions and compare their performance. This makes YOLOv8 the ideal choice for users who want to take advantage of the latest YOLO technology while still being able to use their existing YOLO models. YOLOv8 includes many architectural and developer convenience features, making it an attractive choice for many object detection and image segmentation tasks.

7 Transformer-based detectors

7.1 DETR

DETR (DEtection TRansformer) is the first object detection model based on Transformer architecture.

This model combines a convolutional neural network (CNN) as a base and a Transformer architecture. The CNN network uses ResNet as a basis to extract features from images, which are then formatted and position encoded before being fed into the Transformer architecture. The Transformer architecture in DETR consists of an encoder and a decoder, eliminating elements such as the anchor generation module.

DETR uses a bipartite matching loss function to optimize the match between model output and actual data. DETR generates a fixed number of predictions, each predicted in parallel. The DETR

model approaches object detection by directly predicting the set of objects and using an ensemble-based global loss function.

7.2 SMCA

Introduced in 2021, SMCA is a variant that improves the convergence performance of DETR.

SMCA proposes a Spatial Modulated Co-Attention mechanism to improve the convergence ability of DETR. The SMCA model replaces the co-attention mechanism in the DETR decoder with a position-aware co-attention mechanism. This helps limit the effect of co-attention on unnecessary locations close to the original frame locations.

To train DETR from scratch, about 500 epochs are needed to achieve best performance. SMCA only needs 108 epochs to train and achieves better performance than the original DETR, and shows potential in global information processing.

7.3 ANCHOR DETR

A new Transformer-based object detection model called Anchor DETR is proposed with a new query design.

Anchor DETR uses anchor points to address the physical meaning of object queries. This helps the model focus on objects near the anchor points. This model is capable of predicting multiple objects at the same location.

To optimize complexity, Anchor DETR uses a variant of the attention mechanism called Row-Column Filtered Attention to reduce memory overhead while maintaining accuracy.

7.4 DESTR

"DESTR" proposes a solution to some of Transformer's previous problems, including self-attention and cross-attention mechanisms, along with how to initialize the content query of Transformer decoding.

The author proposes a new variant called Attention Split Detection (Detection Split Transformer), which divides the content embedding estimate of the attention cross mechanism into two independent parts, one for classification and one for for frame regression embedding. This way, they allow each attention cross mechanism to handle its own task. For content query initialization, they use a mini detector to learn the content and initialize the position embedding of the decoder. This unit is equipped with components for classification and regression embeddings. Finally, to account for adjacent object query pairs in decoding, they supplement the self-attention mechanism with the spatial context of the other query in the pair.

8 Summary

There are four main methods in object detection: two-stage anchor-based detection, one-stage anchor-based detection, anchorless detection, and transformer detection.

We overview each method by evaluating several models and methods:

- Two-stage anchor-based detection: R-CNN, SPPNet, FAST R-CNN, FASTER R-CNN, R-FCN, FPN, PANET, TRIDENTNET, SPINENET,...

- One-stage anchor-based detection: YOLOv2, YOLOv3, SSD, RETINANET, MEGDET, EFFICIENTDET, PAA, YOLOv5, YOLOv7,...
- Detect unused anchors: YOLOv1, CORNERNET, EXTREMENET, REPPOINTS, FSAF, FCOS, ATSS, OTA, DSLA, YOLOv8,...
- Detection using transformer: DETR, SMCA, ANCHOR-DETR, DESTR,...

Overall, the future of object detection using deep learning is bright, with many exciting advances for future research.