Object Detection

Mohammadreza Ahmadi Teshnizi
Sajad Sabzi

NAME	SUMMARY	USAGE	YEAR	STAGE
R-CNN (Region-based Convolutional Neural Network)	R-CNN was one of the early methods for object detection using deep learning. It proposed region-based convolutional networks for object localization and classification.	Object detection and localization	2014	TWO
Faster R-CNN	Faster R-CNN introduced the concept of Region Proposal Networks (RPNs) to efficiently generate region proposals. It improved the speed of R-CNN.	Object detection ،often used for real-time applications	2015	TWO
Mask R-CNN	An extension of Faster R-CNN that also includes instance segmentation ،providing pixel-level object masks in addition to object detection.	Object detection and instance segmentation ،commonly used in scenarios where pixel-level accuracy is required.	2017	TWO
Sparse R-CNN	parse R-CNN is designed to improve the efficiency of object detection models by focusing on sparse regions of interest (RoIs) in an image.	Object detection with a focus on efficiency	2020	TWO
YOLO (You Only Look Once)	YOLO introduced the concept of one-stage object detection by simultaneously predicting bounding boxes and class probabilities for multiple objects in a single pass.	Real-time object detection ،commonly used in applications like autonomous driving.	2016	ONE
YOLO.v7	YOLO.v7 is the newest YOLO algorithm surpasses all previous object detection models and YOLO versions in both speed and accuracy. It requires several times cheaper hardware than other neural networks and can be trained much faster on small datasets without any pre-trained weights.	Real-time object detection with improved speed and accuracy.	2022	ONE

YOLOX	YOLOX is an improvement upon YOLO with a focus on speed and accuracy. It utilizes various techniques to achieve better performance.	Real-time object detection with improved speed and accuracy.	2021	ONE
SSD (Single Shot MultiBox Detector)	SSD is another one-stage object detection method that directly predicts object bounding boxes and class scores at multiple scales within a single network.	Object detection with a focus on speed and versatility	2016	ONE
SqueezeDet	SqueezeDet is a real-time object detection system that combines deep learning with network compression techniques to achieve high-speed detection.	Real-time object detection ،often in resource-constrained environments.	2017	ONE
RetinaNet	RetinaNet introduced the focal loss to address the class imbalance problem in object detection. It combines high accuracy with faster inference times.	Object detection with a focus on handling unbalanced datasets.	2017	ONE
MobileNet	MobileNet is a family of efficient neural network architectures designed for mobile and embedded devices ، which can be used for object detection tasks.	Object detection in mobile and resource-constrained environments.	2017	ONE/TWO
EfficientDet	EfficientDet is an efficient object detection model that achieves high accuracy by optimizing model architecture and scaling strategies.	Object detection with a balance of accuracy and efficiency.	2019	ONE/TWO
CenterNet	CenterNet is a one-stage object detection method that focuses on detecting object centers and regressing bounding boxes directly from center points.	Object detection with a focus on simplicity and efficiency	2019	ONE
PANet (Path Aggregation Network)	PANet is not a standalone object detection method but a module designed to improve the feature fusion and information flow in detectors like Faster R-CNN and Mask R-CNN.	Enhancing feature representation in object detection	2018	TWO
DETR (Data-Efficient Transformer for Object Detection)	DETR is a transformer-based object detection method that casts object detection as a set prediction problem and uses attention mechanisms for object detection tasks.	Object detection with a focus on handling various object counts and sizes efficiently	2020	ONE

[1]. YOLOv7: The Fastest Object Detection Algorithm (2023) - viso.ai
[2]. Rich feature hierarchies for accurate object detection and semantic segmentation. by Ross Girshick et al. (2014).
[3]. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. by Shaoqing Ren et al. (2015).
[4]. Mask R-CNN. by Kaiming He et al. (2017)
[5]. Sparse R-CNN: End-to-End Object Detection with Learnable Proposals. by Peize Sun et al. (2020).
[6]. You Only Look Once: Unified ‹Real-Time Object Detection. by Joseph Redmon et al. (2016).
[7]. YOLOX: Exceeding YOLO Series in 2021. by Xiaolong Wang et al. (2021).
[8]. SSD: Single Shot MultiBox Detector. by Wei Liu et al. (2016).
[9]. SqueezeDet: Unified ‹Small ‹Low Power Fully Convolutional Neural Networks for Real-Time Object Detection for Autonomous Driving. by Bichen Wu et al. (2017).
[10].Focal Loss for Dense Object Detection. by Tsung-Yi Lin et al. (2017).
[11].MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. by Andrew G. Howard et al. (2017).
[12].EfficientDet: Scalable and Efficient Object Detection. by Mingxing Tan et al. (2019).
[13]."Objects as Points. by Xingyu Chen et al. (2019)
[14].Path Aggregation Network for Instance Segmentation. by He et al. (2018).
[15].End-to-End Object Detection with Transformers. by Nicolas Carion et al. (2020).
[16].YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. By Chien-Yao Wang et al. (2022).

GitHub Repository (R-CNN): https://github.com/rbgirshick/rcnn

GitHub Repository (Faster R-CNN): https://github.com/ShaoqingRen/faster_rcnn

GitHub Repository (Mask R-CNN): https://github.com/matterport/Mask_RCNN

GitHub Repository (YOLOv4): https://github.com/AlexeyAB/darknet

GitHub Repository (YOLOX): https://github.com/Megvii-BaseDetection/YOLOX

GitHub Repository (YOLOv7): https://github.com/WongKinYiu/yolov7

GitHub Repository (SSD): https://github.com/weiliu89/caffe/tree/ssd

GitHub Repository (RetinaNet): https://github.com/fizyr/keras-retinanet

GitHub Repository (MobileNet): https://github.com/tensorflow/models/blob/master/research/slim/nets/mobilenet_v1.md

GitHub Repository (EfficientDet): https://github.com/google/automl/tree/master/efficientdet

GitHub Repository (DETR): https://github.com/facebookresearch/detectron2