

Detection Toolbox

1. [MMDetection](#)
2. [MMCV](#)
3. [Detectron](#)
4. [Detectron2](#)
5. [SimpleDet](#)
6. [Maskrcnn-benchmark](#)

Paper list

Datasets

1. **[PASCAL VOC]** The PASCAL Visual Object Classes (VOC) Challenge | [\[IJCV' 10\]](#) | [\[pdf\]](#)
2. **[PASCAL VOC]** The PASCAL Visual Object Classes Challenge: A Retrospective | [\[IJCV' 15\]](#) | [\[pdf\]](#) | [\[link\]](#)
3. **[COCO]** Microsoft COCO: Common Objects in Context | [\[ECCV' 14\]](#) | [\[pdf\]](#) | [\[link\]](#)
4. **[Open Images]** The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale | [\[arXiv' 18\]](#) | [\[pdf\]](#) | [\[link\]](#)

Object Detection

two-stage

1. **[R-CNN]** Rich feature hierarchies for accurate object detection and semantic segmentation| [\[pdf\]](#) [\[official code – caffe\]](#)
2. **[Fast R-CNN]** Fast R-CNN | [\[pdf\]](#) [\[official code – caffe\]](#)
3. **[Faster R-CNN]** Faster R-CNN | [\[pdf\]](#) [\[official code – caffe\]](#) [\[code – pytorch\]](#)

one-stage

1. **[YOLO-v1]** You Only Look Once: Unified, Real-Time Object Detection [\[pdf\]](#) [\[code – tensorflow\]](#) [\[code – pytorch\]](#)
2. **[YOLO-v2]** YOLO9000: Better, Faster, Stronger [\[pdf\]](#) [\[code – pytorch\]](#)
3. **[YOLO-v3]** YOLOv3: An Incremental Improvement [\[pdf\]](#) [\[code – pytorch\]](#)
4. **[SSD]** SSD: Single Shot MultiBox Detector [\[pdf\]](#) [\[official code – caffe\]](#) [\[code – pytorch\]](#)
5. **[RetinaNet]** Focal Loss for Dense Object Detection [\[pdf\]](#) [\[official code – pytorch\]](#) [\[code – pytorch\]](#)

NMS

1. **[Soft-NMS]** Improving Object Detection With One Line of Code [\[pdf\]](#) [\[official code – caffe\]](#) [\[code – pytorch\]](#)
2. **[Softer-NMS]** Softer-NMS: Rethinking Bounding Box Regression for Accurate Object Detection [\[pdf\]](#) [\[official code – Caffe2\]](#)
3. **[KL-Loss]** Bounding Box Regression with Uncertainty for Accurate Object Detection [\[pdf\]](#) [\[official code – Caffe2\]](#)
4. Learning non-maximum suppression [\[pdf\]](#)
5. **[Relation-Network]** Relation Networks for Object Detection [\[pdf\]](#) [\[official code – mxnet\]](#)
6. **[PrPool]** Acquisition of Localization Confidence for Accurate Object Detection [\[pdf\]](#) [\[official code – pytorch\]](#)
7. **[Adaptive-NMS]** Adaptive NMS: Refining Pedestrian Detection in a Crowd [\[pdf\]](#)

Anchor-free

1. **[CornerNet]** CornerNet: Detecting Objects as Paired Keypoints [\[pdf\]](#) [\[official code – pytorch\]](#)
2. **[CornerNet-Lite]** CornerNet-Lite: Efficient Keypoint Based Object Detection [\[pdf\]](#) [\[official code – pytorch\]](#)
3. **[FSAF]** Feature Selective Anchor-Free Module for Single-Shot Object Detection [\[pdf\]](#) [\[code – tensorflow\]](#) [\[code – pytorch\]](#)
4. **[ExtremeNet]** Bottom-up Object Detection by Grouping Extreme and Center Points [\[pdf\]](#) [\[official code – pytorch\]](#)
5. **[FoveaBox]** FoveaBox: Beyond Anchor-based Object Detector [\[pdf\]](#) [\[official code – pytorch\]](#)
6. **[FCOS]** FCOS: Fully Convolutional One-Stage Object Detection [\[pdf\]](#) [\[code – pytorch\]](#)
7. **[CenterNet]** Objects as Points [\[pdf\]](#) [\[official code – pytorch\]](#)
8. AFP-Net: Realtime Anchor-Free Polyp Detection in Colonoscopy [\[pdf\]](#)
9. **[DenseBox]** Objects as Points [\[pdf\]](#) [\[code – pytorch\]](#)
10. **[RepPoints]** RepPoints: Point Set Representation for Object Detection | [\[ICCV' 19\]](#) | [\[pdf\]](#) [\[official code – pytorch\]](#)

Instance Segmentation

1. **[Mask R-CNN]** Mask R-CNN [\[pdf\]](#) [\[official code – caffe\]](#) [\[code – pytorch\]](#)
2. **[SDS]** Simultaneous Detection and Segmentation [\[pdf\]](#)
3. **[InstanceCut]** InstanceCut: from Edges to Instances with MultiCut [\[pdf\]](#) [\[official code\]](#)
4. **[InstanceFCN]** Instance-sensitive Fully Convolutional Networks [\[pdf\]](#)

5. **[FCIS]** Fully Convolutional Instance-aware Semantic Segmentation [\[pdf\]](#) [\[official code – mxnet\]](#)
6. **[PolarMask]** PolarMask: Single Shot Instance Segmentation with Polar Representation [\[pdf\]](#) [\[official code – pytorch\]](#)

Video Object Detection

1. **[FGFA]** Flow-Guided Feature Aggregation for Video Object Detection [\[pdf\]](#) [\[official code – mxnet\]](#)
2. **[TCN]** Object Detection from Video Tubelets with Convolutional Neural Networks [\[pdf\]](#) [\[official code – caffe\]](#)
3. **[Seq-NMS]** Seq-NMS for Video Object Detection [\[pdf\]](#)

Others

1. **[Libra R-CNN]** Libra R-CNN: Balanced Learning for Object Detection [\[pdf\]](#) [\[official code – pytorch\]](#)
2. **[Cascade R-CNN]** Cascade R-CNN: Delving into High Quality Object Detection [\[pdf\]](#) [\[official code – caffe\]](#) [\[code – pytorch\]](#)
3. **[FPN]** Feature Pyramid Networks for Object Detection [\[pdf\]](#) [\[code – pytorch\]](#)
4. **[M2Det]** M2Det: A Single-Shot Object Detector based on Multi-Level Feature Pyramid Network [\[pdf\]](#) [\[official code – pytorch\]](#)
5. **[OHEM]** Training Region-based Object Detectors with Online Hard Example Mining [\[pdf\]](#) [\[official code – caffe\]](#) [\[code – pytorch\]](#)
6. **[GIoU]** Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression [\[pdf\]](#) [\[code – pytorch\]](#) [\[code – tensorflow\]](#)
7. **[GA-RPN]** Region Proposal by Guided Anchoring [\[pdf\]](#) [\[official code – pytorch\]](#)
8. Multi-Oriented Scene Text Detection via Corner Localization and Region Segmentation [\[pdf\]](#) [\[official code – pytorch\]](#)
9. Precise Detection in Densely Packed Scenes [\[pdf\]](#) [\[official code – tensorflow\]](#)
10. ScratchDet: Training Single-Shot Object Detectors from Scratch [\[pdf\]](#) [\[official code – caffe\]](#)
11. **[ELASTIC]** ELASTIC: Improving CNNs with Dynamic Scaling Policies [\[pdf\]](#) [\[official code – pytorch\]](#)
12. **[FractalNet]** FractalNet: Ultra-Deep Neural Networks without Residuals [\[pdf\]](#) [\[code – pytorch\]](#) [\[code – tensorflow\]](#)
13. Attention Is All You Need [\[pdf\]](#)
14. **[GHM]** Gradient Harmonized Single-stage Detector [\[pdf\]](#) [\[official code – pytorch\]](#)

Acknowledgements

- Significant amounts of content are based on the [deep learning object detection](#)