Exhibit 28

Object Detection in 20 Years: A Survey

Zhengxia Zou, Zhenwei Shi, *Member, IEEE*, Yuhong Guo, and Jieping Ye, *Senior Member, IEEE*

Abstract—Object detection, as of one the most fundamental and challenging problems in computer vision, has received great attention in recent years. Its development in the past two decades can be regarded as an epitome of computer vision history. If we think of today's object detection as a technical aesthetics under the power of deep learning, then turning back the clock 20 years we would witness the wisdom of cold weapon era. This paper extensively reviews 400+ papers of object detection in the light of its technical evolution, spanning over a quarter-century's time (from the 1990s to 2019). A number of topics have been covered in this paper, including the milestone detectors in history, detection datasets, metrics, fundamental building blocks of the detection system, speed up techniques, and the recent state of the art detection methods. This paper also reviews some important detection applications, such as pedestrian detection, face detection, text detection, etc, and makes an in-deep analysis of their challenges as well as technical improvements in recent years.

Index Terms—Object detection, Computer vision, Deep learning, Convolutional neural networks, Technical evolution.

arXiv:1905.05055v2 [cs.CV] 16 May 2019

1 Introduction

Object detection is an important computer vision task that deals with detecting instances of visual objects of a certain class (such as humans, animals, or cars) in digital images. The objective of object detection is to develop computational models and techniques that provide one of the most basic pieces of information needed by computer vision applications: *What objects are where?*

As one of the fundamental problems of computer vision, object detection forms the basis of many other computer vision tasks, such as instance segmentation [1–4], image captioning [5–7], object tracking [8], etc. From the application point of view, object detection can be grouped into two research topics "general object detection" and "detection applications", where the former one aims to explore the methods of detecting different types of objects under a unified framework to simulate the human vision and cognition, and the later one refers to the detection under specific application scenarios, such as pedestrian detection, face detection, text detection, etc. In recent years, the rapid development of deep learning techniques [9] has brought new blood into object detection, leading to remarkable breakthroughs and pushing it forward to a research hot-spot with unprecedented attention. Object detection has now been widely used in many real-world applications, such as autonomous driving, robot vision, video surveillance, etc. Fig. 1 shows the growing number of publications that are associated with "object detection" over the past two decades.

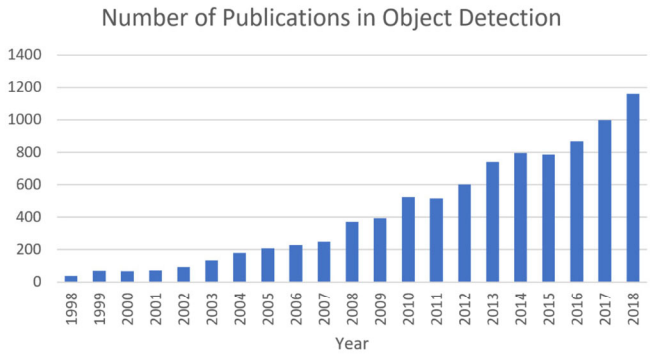


Fig. 1. The increasing number of publications in object detection from 1998 to 2018. (Data from Google scholar advanced search: *allintitle: "object detection" AND "detecting objects"*.)

- **Difference from other related reviews**

A number of reviews of general object detection have been published in recent years [24–28]. The main difference between this paper and the above reviews are summarized as follows:

1. A comprehensive review in the light of technical evolutions: This paper extensively reviews 400+ papers in the development history of object detection, spanning over a quarter-century's time (from the 1990s to 2019). Most of the previous reviews merely focus on a short historical period or on some specific detection tasks without considering the technical evolutions over their entire lifetime. Standing on the highway of the history not only helps readers build a complete knowledge hierarchy but also helps to find future directions of this fast developing field.

2. An in-depth exploration of the key technologies and the recent state of the arts: After years of development, the state of the art object detection systems have been integrated with a large number of techniques such as "multi-scale detection", "hard negative mining", "bounding box

Corresponding Author: Zhengxia Zou (zzhengxi@umich.edu) and Jieping Ye (jpye@umich.edu).
Zhengxia Zou is with the Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, MI 48109, U.S.A.
Zhenwei Shi is with the Image Processing Center, School of Astronautics, Beihang University, Beijing 100191, China, and with the State Key Laboratory of Virtual Reality Technology and Systems, School of Astronautics, Beihang University, Beijing 100191, China.
Yuhong Guo is with the School of Computer Science, Carleton University, Ottawa, HP5167, Canada, and with the DiDi Labs, Toronto, Canada.
Jieping Ye is with the Department of Computational Medicine and Bioinformatics, and the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, U.S.A., and with the DiDi AI Labs, DiDi Chuxing, Beijing, 100085, China.

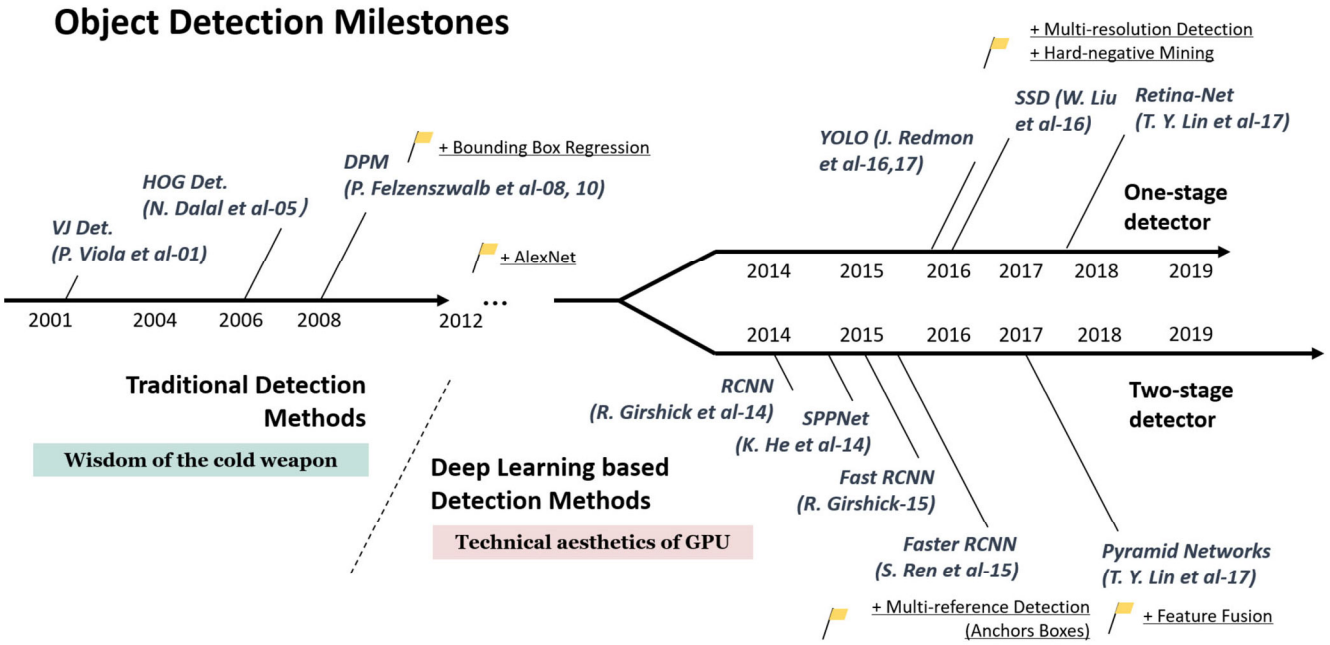


Fig. 2. A road map of object detection. Milestone detectors in this figure: VJ Det. [10, 11], HOG Det. [12], DPM [13–15], RCNN [16], SPPNet [17], Fast RCNN [18], Faster RCNN [19], YOLO [20], SSD [21], Pyramid Networks [22], Retina-Net [23].

regression", etc. However, previous reviews lack fundamental analysis to help readers understand the nature of these sophisticated techniques, e.g., "Where did they come from and how did they evolve?" "What are the pros and cons of each group of methods?" This paper makes an in-depth analysis for readers of the above concerns.

3. A comprehensive analysis of detection speed up techniques: The acceleration of object detection has long been a crucial but challenging task. This paper makes an extensive review of the speed up techniques in 20 years of object detection history at multiple levels, including "detection pipeline" (e.g., cascaded detection, feature map shared computation), "detection backbone" (e.g., network compression, lightweight network design), and "numerical computation" (e.g., integral image, vector quantization). This topic is rarely covered by previous reviews.

- **Difficulties and Challenges in Object Detection**

Despite people always asking "what are the difficulties and challenges in object detection?", actually, this question is not easy to answer and may even be over-generalized. As different detection tasks have totally different objectives and constraints, their difficulties may vary from each other. In addition to some common challenges in other computer vision tasks such as objects under different viewpoints, illuminations, and intraclass variations, the challenges in object detection include but not limited to the following aspects: object rotation and scale changes (e.g., small objects), accurate object localization, dense and occluded object detection, speed up of detection, etc. In Sections 4 and 5, we will give a more detailed analysis of these topics.

The rest of this paper is organized as follows. In Section 2, we review the 20 years' evolutionary history of object detection. Some speed up techniques in object detection will be introduced in Section 3. Some state of the art detection methods in the recent three years are summarized in Section 4. Some important detection applications will be reviewed in Section 5. In Section 6, we conclude this paper and make an analysis of the further research directions.

2 Object Detection in 20 Years

In this section, we will review the history of object detection in multiple aspects, including milestone detectors, object detection datasets, metrics, and the evolution of key techniques.

2.1 A Road Map of Object Detection

In the past two decades, it is widely accepted that the progress of object detection has generally gone through two historical periods: "traditional object detection period (before 2014)" and "deep learning based detection period (after 2014)", as shown in Fig. 2.

2.1.1 Milestones: Traditional Detectors

If we think of today's object detection as a technical aesthetics under the power of deep learning, then turning back the clock 20 years we would witness "the wisdom of cold weapon era". Most of the early object detection algorithms were built based on handcrafted features. Due to the lack of effective image representation at that time, people have no choice but to design sophisticated feature representations, and a variety of speed up skills to exhaust the usage of limited computing resources.

- Viola Jones Detectors

18 years ago, P. Viola and M. Jones achieved real-time detection of human faces for the first time without any constraints (e.g., skin color segmentation) [10, 11]. Running

on a 700MHz Pentium III CPU, the detector was tens or even hundreds of times faster than any other algorithms in its time under comparable detection accuracy. The detection algorithm, which was later referred to the "Viola-Jones (VJ) detector", was herein given by the authors' names in memory of their significant contributions.

The VJ detector follows a most straight forward way of detection, i.e., sliding windows: to go through all possible locations and scales in an image to see if any window contains a human face. Although it seems to be a very simple process, the calculation behind it was far beyond the computer's power of its time. The VJ detector has dramatically improved its detection speed by incorporating three important techniques: "integral image", "feature selection", and "detection cascades".

1) Integral image: The integral image is a computational method to speed up box filtering or convolution process. Like other object detection algorithms in its time [29–31], the Haar wavelet is used in VJ detector as the feature representation of an image. The integral image makes the computational complexity of each window in VJ detector independent of its window size.

2) Feature selection: Instead of using a set of manually selected Haar basis filters, the authors used Adaboost algorithm [32] to select a small set of features that are mostly helpful for face detection from a huge set of random features pools (about 180k-dimensional).

3) Detection cascades: A multi-stage detection paradigm (a.k.a. the "detection cascades") was introduced in VJ detector to reduce its computational overhead by spending less computations on background windows but more on face targets.

- HOG Detector

Histogram of Oriented Gradients (HOG) feature descriptor was originally proposed in 2005 by N. Dalal and B. Triggs [12]. HOG can be considered as an important improvement of the scale-invariant feature transform [33, 34] and shape contexts [35] of its time. To balance the feature invariance (including translation, scale, illumination, etc) and the nonlinearity (on discriminating different objects categories), the HOG descriptor is designed to be computed on a dense grid of uniformly spaced cells and use overlapping local contrast normalization (on "blocks") for improving accuracy. Although HOG can be used to detect a variety of object classes, it was motivated primarily by the problem of pedestrian detection. To detect objects of different sizes, the HOG detector rescales the input image for multiple times while keeping the size of a detection window unchanged. The HOG detector has long been an important foundation of many object detectors [13, 14, 36] and a large variety of computer vision applications for many years.

- Deformable Part-based Model (DPM)

DPM, as the winners of VOC-07, -08, and -09 detection challenges, was the peak of the traditional object detection methods. DPM was originally proposed by P. Felzenszwalb [13] in 2008 as an extension of the HOG detector, and then a variety of improvements have been made by R. Girshick [14, 15, 37, 38].

The DPM follows the detection philosophy of "divide and conquer", where the training can be simply considered as the learning of a proper way of decomposing an object, and the inference can be considered as an ensemble of detections on different object parts. For example, the problem of detecting a "car" can be considered as the detection of its window, body, and wheels. This part of the work, a.k.a. "star-model", was completed by P. Felzenszwalb *et al.* [13]. Later on, R. Girshick has further extended the star-model to the "mixture models" [14, 15, 37, 38] to deal with the objects in the real world under more significant variations.

A typical DPM detector consists of a root-filter and a number of part-filters. Instead of manually specifying the configurations of the part filters (e.g., size and location), a weakly supervised learning method is developed in DPM where all configurations of part filters can be learned automatically as latent variables. R. Girshick has further formulated this process as a special case of Multi-Instance learning [39], and some other important techniques such as "hard negative mining", "bounding box regression", and "context priming" are also applied for improving detection accuracy (to be introduced in Section 2.3). To speed up the detection, Girshick developed a technique for "compiling" detection models into a much faster one that implements a cascade architecture, which has achieved over 10 times acceleration without sacrificing any accuracy [14, 38].

Although today's object detectors have far surpassed DPM in terms of the detection accuracy, many of them are still deeply influenced by its valuable insights, e.g., mixture models, hard negative mining, bounding box regression, etc. In 2010, P. Felzenszwalb and R. Girshick were awarded the "lifetime achievement" by PASCAL VOC.

2.1.2 Milestones: CNN based Two-stage Detectors

As the performance of hand-crafted features became saturated, object detection has reached a plateau after 2010. R. Girshick says: "... progress has been slow during 2010-2012, with small gains obtained by building ensemble systems and employing minor variants of successful methods"[38]. In 2012, the world saw the rebirth of convolutional neural networks [40]. As a deep convolutional network is able to learn robust and high-level feature representations of an image, a natural question is whether we can bring it to object detection? R. Girshick *et al.* took the lead to break the deadlocks in 2014 by proposing the Regions with CNN features (RCNN) for object detection [16, 41]. Since then, object detection started to evolve at an unprecedented speed.

In deep learning era, object detection can be grouped into two genres: "two-stage detection" and "one-stage detection", where the former frames the detection as a "coarse-to-fine" process while the later frames it as to "complete in one step".

- RCNN

The idea behind RCNN is simple: It starts with the extraction of a set of object proposals (object candidate boxes) by selective search [42]. Then each proposal is rescaled to a fixed size image and fed into a CNN model trained on ImageNet (say, AlexNet [40]) to extract features. Finally, linear SVM classifiers are used to predict the presence of an object within each region and to recognize object categories.

RCNN yields a signicant performance boost on VOC07, with a large improvement of mean Average Precision (mAP) from 33.7% (DPM-v5 [43]) to 58.5%.

Although RCNN has made great progress, its drawbacks are obvious: the redundant feature computations on a large number of overlapped proposals (over 2000 boxes from one image) leads to an extremely slow detection speed (14s per image with GPU). Later in the same year, SPPNet [17] was proposed and has overcome this problem.

- SPPNet

In 2014, K. He *et al.* proposed Spatial Pyramid Pooling Networks (SPPNet) [17]. Previous CNN models require a fixed-size input, e.g., a 224x224 image for AlexNet [40]. The main contribution of SPPNet is the introduction of a Spatial Pyramid Pooling (SPP) layer, which enables a CNN to generate a fixed-length representation regardless of the size of image/region of interest without rescaling it. When using SPPNet for object detection, the feature maps can be computed from the entire image only once, and then fixed-length representations of arbitrary regions can be generated for training the detectors, which avoids repeatedly computing the convolutional features. SPPNet is more than 20 times faster than R-CNN without sacrificing any detection accuracy (VOC07 mAP=59.2%).

Although SPPNet has effectively improved the detection speed, there are still some drawbacks: first, the training is still multi-stage, second, SPPNet only fine-tunes its fully connected layers while simply ignores all previous layers. Later in the next year, Fast RCNN [18] was proposed and solved these problems.

- Fast RCNN

In 2015, R. Girshick proposed Fast RCNN detector [18], which is a further improvement of R-CNN and SPPNet [16, 17]. Fast RCNN enables us to simultaneously train a detector and a bounding box regressor under the same network configurations. On VOC07 dataset, Fast RCNN increased the mAP from 58.5% (RCNN) to 70.0% while with a detection speed over 200 times faster than R-CNN.

Although Fast-RCNN successfully integrates the advantages of R-CNN and SPPNet, its detection speed is still limited by the proposal detection (see Section 2.3.2 for more details). Then, a question naturally arises: "can we generate object proposals with a CNN model?" Later, Faster R-CNN [19] has answered this question.

- Faster RCNN

In 2015, S. Ren *et al.* proposed Faster RCNN detector [19, 44] shortly after the Fast RCNN. Faster RCNN is the first end-to-end, and the first near-realtime deep learning detector (COCO mAP@.5=42.7%, COCO mAP@[.5,.95]=21.9%, VOC07 mAP=73.2%, VOC12 mAP=70.4%, 17fps with ZF-Net [45]). The main contribution of Faster-RCNN is the introduction of Region Proposal Network (RPN) that enables nearly cost-free region proposals. From R-CNN to Faster RCNN, most individual blocks of an object detection system, e.g., proposal detection, feature extraction, bounding box regression, etc, have been gradually integrated into a unified, end-to-end learning framework.

Although Faster RCNN breaks through the speed bottleneck of Fast RCNN, there is still computation redundancy at subsequent detection stage. Later, a variety of improvements have been proposed, including RFCN [46] and Light head RCNN [47]. (See more details in Section 3.)

- Feature Pyramid Networks

In 2017, T.-Y. Lin *et al.* proposed Feature Pyramid Networks (FPN) [22] on basis of Faster RCNN. Before FPN, most of the deep learning based detectors run detection only on a network's top layer. Although the features in deeper layers of a CNN are beneficial for category recognition, it is not conducive to localizing objects. To this end, a top-down architecture with lateral connections is developed in FPN for building high-level semantics at all scales. Since a CNN naturally forms a feature pyramid through its forward propagation, the FPN shows great advances for detecting objects with a wide variety of scales. Using FPN in a basic Faster R-CNN system, it achieves state-of-the-art single model detection results on the MSCOCO dataset without bells and whistles (COCO mAP@.5=59.1%, COCO mAP@[.5, .95]=36.2%). FPN has now become a basic building block of many latest detectors.

2.1.3 Milestones: CNN based One-stage Detectors

- You Only Look Once (YOLO)

YOLO was proposed by R. Joseph *et al.* in 2015. It was the first one-stage detector in deep learning era [20]. YOLO is extremely fast: a fast version of YOLO runs at 155fps with VOC07 mAP=52.7%, while its enhanced version runs at 45fps with VOC07 mAP=63.4% and VOC12 mAP=57.9%. YOLO is the abbreviation of "You Only Look Once". It can be seen from its name that the authors have completely abandoned the previous detection paradigm of "proposal detection + verification". Instead, it follows a totally different philosophy: to apply a single neural network to the full image. This network divides the image into regions and predicts bounding boxes and probabilities for each region simultaneously. Later, R. Joseph has made a series of improvements on basis of YOLO and has proposed its v2 and v3 editions [48, 49], which further improve the detection accuracy while keeps a very high detection speed.

In spite of its great improvement of detection speed, YOLO suffers from a drop of the localization accuracy compared with two-stage detectors, especially for some small objects. YOLO's subsequent versions [48, 49] and the latter proposed SSD [21] has paid more attention to this problem.

- Single Shot MultiBox Detector (SSD)

SSD [21] was proposed by W. Liu *et al.* in 2015. It was the second one-stage detector in deep learning era. The main contribution of SSD is the introduction of the multi-reference and multi-resolution detection techniques (to be introduce in Section 2.3.2), which significantly improves the detection accuracy of a one-stage detector, especially for some small objects. SSD has advantages in terms of both detection speed and accuracy (VOC07 mAP=76.8%, VOC12 mAP=74.9%, COCO mAP@.5=46.5%, mAP@[.5,.95]=26.8%, a fast version runs at 59fps). The main difference between SSD and any previous detectors is that the former one detects objects of

different scales on different layers of the network, while the latter ones only run detection on their top layers.

- RetinaNet

In despite of its high speed and simplicity, the one-stage detectors have trailed the accuracy of two-stage detectors for years. T.-Y. Lin *et al.* have discovered the reasons behind and proposed RetinaNet in 2017 [23]. They claimed that the extreme foreground-background class imbalance encountered during training of dense detectors is the central cause. To this end, a new loss function named "focal loss" has been introduced in RetinaNet by reshaping the standard cross entropy loss so that detector will put more focus on hard, misclassified examples during training. Focal Loss enables the one-stage detectors to achieve comparable accuracy of two-stage detectors while maintaining very high detection speed. (COCO mAP@.5=59.1%, mAP@[.5, .95]=39.1%).

2.2 Object Detection Datasets and Metrics

Building larger datasets with less bias is critical for developing advanced computer vision algorithms. In object detection, a number of well-known datasets and benchmarks have been released in the past 10 years, including the datasets of PASCAL VOC Challenges [50, 51] (e.g., VOC2007, VOC2012), ImageNet Large Scale Visual Recognition Challenge (e.g., ILSVRC2014) [52], MS-COCO Detection Challenge [53], etc. The statistics of these datasets are given in Table 1. Fig. 4 shows some image examples of these datasets. Fig. 3 shows the improvements of detection accuracy on VOC07, VOC12 and MS-COCO datasets from 2008 to 2018.

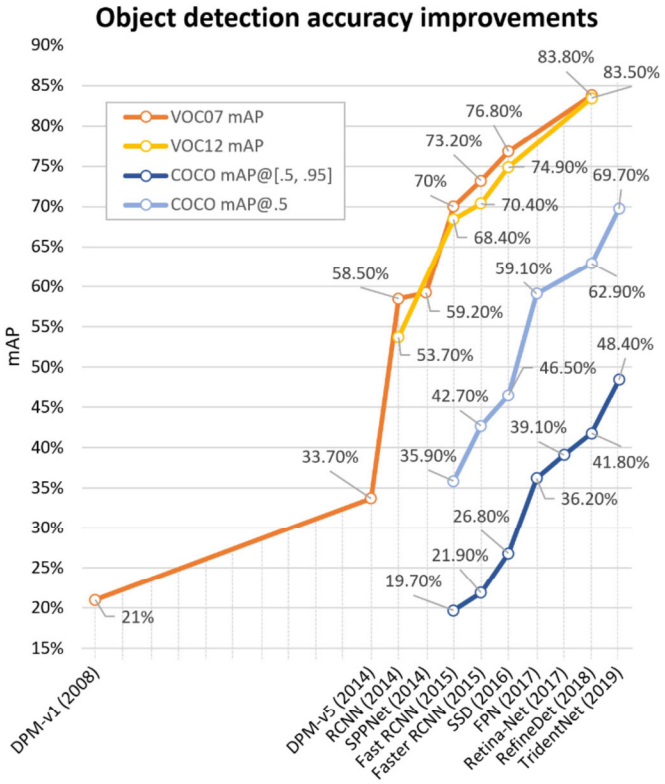


Fig. 3. The accuracy improvements of object detection on VOC07, VOC12 and MS-COCO datasets. Detectors in this figure: DPM-v1 [13], DPM-v5 [54], RCNN [16], SPPNet [17], Fast RCNN [18], Faster RCNN [19], SSD [21], FPN [22], Retina-Net [23], RefineDet [55], TridentNet[56].

- Pascal VOC

The PASCAL Visual Object Classes (VOC) Challenges[1] (from 2005 to 2012) [50, 51] was one of the most important competition in early computer vision community. There are multiple tasks in PASCAL VOC, including image classification, object detection, semantic segmentation and action detection. Two versions of Pascal-VOC are mostly used in object detection: VOC07 and VOC12, where the former consists of 5k tr. images + 12k annotated objects, and the latter consists of 11k tr. images + 27k annotated objects. 20 classes of objects that are common in life are annotated in these two datasets (Person: person; Animal: bird, cat, cow, dog, horse, sheep; Vehicle: aeroplane, bicycle, boat, bus, car, motor-bike, train; Indoor: bottle, chair, dining table, potted plant, sofa, tv/monitor). In recent years, as some larger datasets like ILSVRC and MS-COCO (to be introduced) has been released, the VOC has gradually fallen out of fashion and has now become a test-bed for most new detectors.

- ILSVRC

The ImageNet Large Scale Visual Recognition Challenge (ILSVRC)[2] [52] has pushed forward the state of the art in generic object detection. ILSVRC is organized each year from 2010 to 2017. It contains a detection challenge using ImageNet images [57]. The ILSVRC detection dataset contains 200 classes of visual objects. The number of its images/object instances is two orders of magnitude larger than VOC. For example, ILSVRC-14 contains 517k images and 534k annotated objects.

- MS-COCO

MS-COCO[3] [53] is the most challenging object detection dataset available today. The annual competition based on MS-COCO dataset has been held since 2015. It has less number of object categories than ILSVRC, but more object instances. For example, MS-COCO-17 contains 164k images and 897k annotated objects from 80 categories. Compared with VOC and ILSVRC, the biggest progress of MS-COCO is that apart from the bounding box annotations, each object is further labeled using per-instance segmentation to aid in precise localization. In addition, MS-COCO contains more small objects (whose area is smaller than 1% of the image) and more densely located objects than VOC and ILSVRC. All these features make the objects distribution in MS-COCO closer to those of the real world. Just like ImageNet in its time, MS-COCO has become the de facto standard for the object detection community.

- Open Images

The year of 2018 sees the introduction of the Open Images Detection (OID) challenge[4] [58], following MS-COCO but at an unprecedented scale. There are two tasks in

1. http://host.robots.ox.ac.uk/pascal/VOC/
2. http://image-net.org/challenges/LSVRC/
3. http://cocodataset.org/
4. https://storage.googleapis.com/openimages/web/index.html

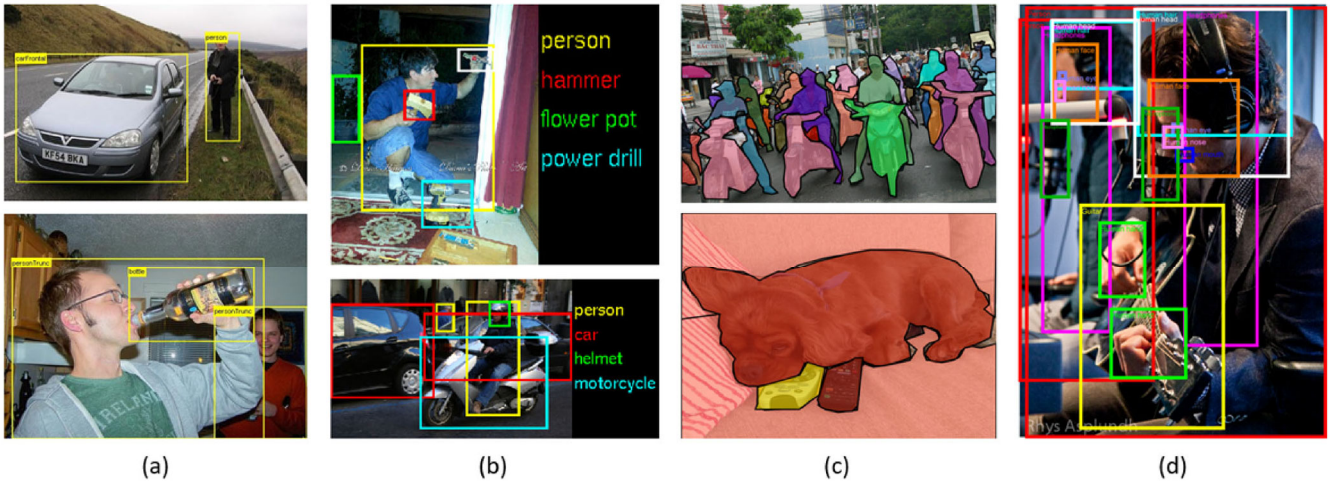


Fig. 4. Some example images and annotations in (a) PASCAL-VOC07, (b) ILSVRC, (c) MS-COCO, and (d) Open Images.

Dataset	train		validation		trainval		test	
	images	objects	images	objects	images	objects	images	objects
VOC-2007	2,501	6,301	2,510	6,307	5,011	12,608	4,952	14,976
VOC-2012	5,717	13,609	5,823	13,841	11,540	27,450	10,991	-
ILSVRC-2014	456,567	478,807	20,121	55,502	476,688	534,309	40,152	-
ILSVRC-2017	456,567	478,807	20,121	55,502	476,688	534,309	65,500	-
MS-COCO-2015	82,783	604,907	40,504	291,875	123,287	896,782	81,434	-
MS-COCO-2018	118,287	860,001	5,000	36,781	123,287	896,782	40,670	-
OID-2018	1,743,042	14,610,229	41,620	204,621	1,784,662	14,814,850	125,436	625,282

TABLE 1
Some well-known object detection datasets and their statistics.

Open Images: 1) the standard object detection, and 2) the visual relationship detection which detects paired objects in particular relations. For the object detection task, the dataset consists of 1,910k images with 15,440k annotated bounding boxes on 600 object categories.

- Datasets of Other Detection Tasks

In addition to general object detection, the past 20 years also witness the prosperity of detection applications in specific areas, such as pedestrian detection, face detection, text detection, traffic sign/light detection, and remote sensing target detection. Tables 2-6 list some of the popular datasets of these detection tasks[5]. A detailed introduction of the detection methods of these tasks can be found in Section 5.

2.2.1 Metrics

How can we evaluate the effectiveness of an object detector? This question may even have different answers at different time.

In the early time's detection community, there is no widely accepted evaluation criteria on detection performance. For example, in the early research of pedestrian detection [12], the "miss rate vs. false positives per-window (FPPW)" was usually used as a metric. However, the per-window measurement (FPPW) can be flawed and fails to predict full image performance in certain cases [59]. In 2009, the Caltech pedestrian detection benchmark was created [59, 60] and since then, the evaluation metric has changed from per-window (FPPW) to false positives per-image (FPPI).

In recent years, the most frequently used evaluation for object detection is "Average Precision (AP)", which was originally introduced in VOC2007. AP is defined as the average detection precision under different recalls, and is usually evaluated in a category specific manner. To compare performance over all object categories, the mean AP (mAP) averaged over all object categories is usually used as the final metric of performance. To measure the object localization accuracy, the Intersection over Union (IoU) is used to check whether the IoU between the predicted box and the ground truth box is greater than a predefined threshold, say, 0.5. If yes, the object will be identified as "successfully detected", otherwise will be identified as "missed". The 0.5-IoU based mAP has then become the de facto metric for object detection problems for years.

After 2014, due to the popularity of MS-COCO datasets, researchers started to pay more attention to the accuracy of the bounding box location. Instead of using a fixed IoU threshold, MS-COCO AP is averaged over multiple IoU thresholds between 0.5 (coarse localization) and 0.95 (perfect localization). This change of the metric has encouraged more accurate object localization and may be of great importance for some real-world applications (e.g., imagine there is a

5. The #Cites shows statistics as of Feb. 2019.

Dataset	Year	Description	#Cites
MIT Ped.[30]	2000	One of the first pedestrian detection datasets. Consists of ∼500 training and ∼200 testing images (built based on the LabelMe database). url: http://cbcl.mit.edu/software-datasets/PedestrianData.html	1515
INRIA [12]	2005	One of the most famous and important pedestrian detection datasets at early time. Introduced by the HOG paper [12]. url: http://pascal.inrialpes.fr/data/human/	24705
Caltech [59, 60]	2009	One of the most famous pedestrian detection datasets and benchmarks. Consists of ∼190,000 pedestrians in training set and ∼160,000 in testing set. The metric is Pascal-VOC @ 0.5 IoU. url: http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/	2026
KITTI [61]	2012	One of the most famous datasets for traffic scene analysis. Captured in Karlsruhe, Germany. Consists of ∼100,000 pedestrians (∼6,000 individuals). url: http://www.cvlibs.net/datasets/kitti/index.php	2620
CityPersons [62]	2017	Built based on CityScapes dataset [63]. Consists of ∼19,000 pedestrians in training set and ∼11,000 in testing set. Same metric with CalTech. url: https://bitbucket.org/shanshanzhang/citypersons	50
EuroCity [64]	2018	The largest pedestrian detection dataset so far. Captured from 31 cities in 12 European countries. Consists of ∼238,000 instances in ∼47,000 images. Same metric with CalTech.	1

TABLE 2
An overview of some popular pedestrian detection datasets.

Dataset	Year	Description	#Cites
FDDB [65]	2010	Consists of ∼2,800 images and ∼5,000 faces from Yahoo! With occlusions, pose changes, out-of-focus, etc. url: http://vis-www.cs.umass.edu/fddb/index.html	531
AFLW [66]	2011	Consists of ∼26,000 faces and 22,000 images from Flickr with rich facial landmark annotations. url: https://www.tugraz.at/institute/icg/research/team-bischof/lrs/downloads/aflw/	414
IJB [67]	2015	IJB-A/B/C consists of over 50,000 images and videos frames, for both recognition and detection tasks. url: https://www.nist.gov/programs-projects/face-challenges	279
WiderFace [68]	2016	One of the largest face detection dataset. Consists of ∼32,000 images and 394,000 faces with rich annotations i.e., scale, occlusion, pose, etc. url: http://mmlab.ie.cuhk.edu.hk/projects/WIDERFace/	193
UFDD [69]	2018	Consists of ∼6,000 images and ∼11,000 faces. Variations include weather-based degradation, motion blur, focus blur, etc. url: http://www.ufdd.info/	1
WildestFaces [70]	2018	With ∼68,000 video frames and ∼2,200 shots of 64 fighting celebrities in unconstrained scenarios. The dataset hasn't been released yet.	2

TABLE 3
An overview of some popular face detection datasets.

robot arm trying to grasp a spanner).

Recently, there are some further developments of the evaluation in the Open Images dataset, e.g., by considering the group-of boxes and the non-exhaustive image-level category hierarchies. Some researchers also have proposed some alternative metrics, e.g., "localization recall precision" [94]. Despite the recent changes, the VOC/COCO-based mAP is still the most frequently used evaluation metric for object detection.

2.3 Technical Evolution in Object Detection

In this section, we will introduce some important building blocks of a detection system and their technical evolutions in the past 20 years.

2.3.1 *Early Time's Dark Knowledge*

The early time's object detection (before 2000) did not follow a unified detection philosophy like sliding window detection. Detectors at that time were usually designed based on low-level and mid-level vision as follows.

- Components, shapes and edges

"Recognition-by-components", as an important cognitive theory [98], has long been the core idea of image recognition and object detection [13, 99, 100]. Some early researchers framed the object detection as a measurement of similarity between the object components, shapes and contours, including Distance Transforms [101], Shape Contexts [35], and Edgelet [102], etc. Despite promising initial results, things did not work out well on more complicated detec-

Dataset	Year	Description	#Cites
ICDAR [71]	2003	ICDAR2003 is one of the first public datasets for text detection. ICDAR 2015 and 2017 are other popular iterations of the ICDAR challenge [72, 73]. url: http://rrc.cvc.uab.es/	530
STV [74]	2010	Consists of ∼350 images and ∼720 text instances taken from Google StreetView. url: http://tc11.cvc.uab.es/datasets/SVT_1	339
MSRA-TD500 [75]	2012	Consists of ∼500 indoor/outdoor images with Chinese and English texts. url: http://www.iapr-tc11.org/mediawiki/index.php/MSRA_Text_Detection_500_Database_(MSRA-TD500)	413
IIIT5k [76]	2012	Consists of ∼1,100 images and ∼5,000 words from both streets and born-digital images. url: http://cvit.iiit.ac.in/projects/SceneTextUnderstanding/IIIT5K.html	165
Syn90k [77]	2014	A synthetic dataset with 9 million images generated from a 90,000 vocabulary of multiple fonts. url: http://www.robots.ox.ac.uk/~vgg/data/text/	246
COCOText [78]	2016	The largest text detection dataset so far. Built based on MS-COCO, Consists of ∼63,000 images and ∼173,000 text annotations. https://bgshih.github.io/cocotext/.	69

TABLE 4
An overview of some popular scene text detection datasets.

Dataset	Year	Description	#Cites
TLR [79]	2009	Captured by a moving vehicle in Paris. Consists of ∼11,000 video frames and ∼9,200 traffic light instances. url: http://www.lara.prd.fr/benchmarks/trafficlightsrecognition	164
LISA [80]	2012	One of the first traffic sign detection dataset. Consists of ∼6,600 video frames, ∼7,800 instances of 47 US signs. url: http://cvrr.ucsd.edu/LISA/lisa-traffic-sign-dataset.html	325
GTSDB [81]	2013	One of the most popular traffic signs detection dataset. Consists of ∼900 images with ∼1,200 traffic signs capture with various weather conditions during different time of a day. url: http://benchmark.ini.rub.de/?section=gtsdb&subsection=news	259
BelgianTSD [82]	2012	Consists of ∼7,300 static images, ∼120,000 video frames, and ∼11,000 traffic sign annotations of 269 types. The 3D location of each sign has been annotated. url: https://btsd.ethz.ch/shareddata/	224
TT100K [83]	2016	The largest traffic sign detection dataset so far, with ∼100,000 images (2048 x 2048) and ∼30,000 traffic sign instances of 128 classes. Each instance is annotated with class label, bounding box and pixel mask. url: http://cg.cs.tsinghua.edu.cn/traffic%2Dsign/	111
BSTL [84]	2017	The largest traffic light detection dataset. Consists of ∼5000 static images, ∼8300 video frames, and ∼24000 traffic light instances. https://hci.iwr.uni-heidelberg.de/node/6132	21

TABLE 5
An overview of some popular traffic light detection and traffic sign detection datasets.

tion problems. Therefore, machine learning based detection methods were beginning to prosper.

Machine learning based detection has gone through multiple periods, including the statistical models of appearance (before 1998), wavelet feature representations (1998-2005), and gradient-based representations (2005-2012).

Building statistical models of an object, like Eigenfaces [95, 106] as shown in Fig 5 (a), was the first wave of learning based approaches in object detection history. In 1991, M. Turk *et al.* achieved real-time face detection in a lab environment by using Eigenface decomposition [95]. Compared with the rule-based or template based approaches of its time [107, 108], a statistical model better provides holistic descriptions of an object's appearance by learning task-specific knowledge from data.

Wavelet feature transform started to dominate visual recognition and object detection since 2000. The essence of this group of methods is learning by transforming an image from pixels to a set of wavelet coefficients. Among these methods, the Haar wavelet, owing to its high computational efficiency, has been mostly used in many object detection tasks, such as general object detection [29], face detection [10, 11, 109], pedestrian detection [30, 31], etc. Fig 5 (d) shows a set of Haar wavelets basis learned by a VJ detector [10, 11] for human faces.

- Early time's CNN for object detection

The history of using CNN to detecting objects can be

Dataset	Year	Description	#Cites
TAS [85]	2008	Consists of 30 images of 729x636 pixels from Google Earth and ∼1,300 vehicles. url: http://ai.stanford.edu/~gaheitz/Research/TAS/	419
OIRDS [86]	2009	Consists for 900 images (0.08-0.3m/pixel) captured by aircraft-mounted camera and 1,800 annotated vehicle targets. url: https://sourceforge.net/projects/oirds/	32
DLR3K [87]	2013	The most frequently used datasets for small vehicle detection. Consists of 9,300 cars and 160 trucks. url: https://www.dlr.de/eoc/en/desktopdefault.aspx/tabid-5431/9230_read-42467/	68
UCAS-AOD [88]	2015	Consists of ∼900 Google Earth images, ∼2,800 vehicles and ∼3,200 airplanes. url: http://www.ucassdl.cn/resource.asp	19
VeDAI [89]	2016	Consists of ∼1,200 images (0.1-0.25m/pixel), ∼3,600 targets of 9 classes. Designed for detecting small target in remote sensing images. url: https://downloads.greyc.fr/vedai/	65
NWPU-VHR10 [90]	2016	The most frequently used remote sensing detection dataset in recent years. Consists of ∼800 images (0.08-2.0m/pixel) and ∼3,800 remote sensing targets of ten classes (e.g., airplanes, ships, baseball diamonds, tennis courts, etc). url: http://jiong.tea.ac.cn/people/JunweiHan/NWPUVHR10dataset.html	204
LEVIR [91]	2018	Consists of ∼22,000 Google Earth images and ∼10,000 independently labeled targets (airplane, ship, oil-pot). url: https://pan.baidu.com/s/1geTwAVD	15
DOTA [92]	2018	The first remote sensing detection dataset to incorporate rotated bounding boxes. Consists of ∼2,800 Google Earth images and ∼200,000 instances of 15 classes. url: https://captain-whu.github.io/DOTA/dataset.html	32
xView [93]	2018	The largest remote sensing detection dataset so far. Consists of ∼1,000,000 remote sensing targets of 60 classes (0.3m/pixel), covering1,415km^2 of land area. url: http://xviewdataset.org	10

TABLE 6
An overview of some remote sensing target detection datasets.

traced back to the 1990s [96], where Y. LeCun *et al.* have made great contributions at that time. Due to limitations in computing resources, CNN models at the time were much smaller and shallower than those of today. Despite this, the computational efficiency was still considered as one of the tough nuts to crack in early times's CNN based detection models. Y. LeCun *et al.* have made a series of improvements like "shared-weight replicated neural network" [96] and "space displacement network" [97] to reduce the computations by extending each layer of the convolutional network so as to cover the entire input image, as shown in Fig. 5 (b)-(c). In this way, the feature of any location of the entire image can be extracted by taking only one time of forward propagation of the network. This can be considered as the prototype of today's fully convolutional networks (FCN) [110, 111], which was proposed almost 20 years later. CNN also has been applied to other tasks such as face detection [112, 113] and hand tracking [114] of its time.

2.3.2 Technical Evolution of Multi-Scale Detection

Multi-scale detection of objects with "different sizes" and "different aspect ratios" is one of the main technical challenges in object detection. In the past 20 years, multi-scale detection has gone through multiple historical periods: "feature pyramids and sliding windows (before 2014)", "detection with object proposals (2010-2015)", "deep regression (2013-2016)", "multi-reference detection (after 2015)", and "multi-resolution detection (after 2016)", as shown in Fig. 6.

- Feature pyramids + sliding windows (before 2014)

With the increase of computing power after the VJ detector, researchers started to pay more attention to an intuitive way of detection by building "feature pyramid + sliding windows". From 2004 to 2014, a number of milestone detectors were built based on this detection paradigm, including the HOG detector, DPM, and even the Overfeat detector [103] of the deep learning era (winner of ILSVRC-13 localization task).

Early detection models like VJ detector and HOG detector were specifically designed to detect objects with a "fixed aspect ratio" (e.g., faces and upright pedestrians) by simply building the feature pyramid and sliding fixed size detection window on it. The detection of "various aspect ratios" was not considered at that time. To detect objects with a more complex appearance like those in PASCAL VOC, R. Girshick *et al.* began to seek better solutions outside the feature pyramid. The "mixture model" [15] was one of the best solutions at that time, by training multiple models to detect objects with different aspect ratios. Apart from this, exemplar-based detection [36, 115] provided another solution by training individual models for every object instance (exemplar) of the training set.

As objects in the modern datasets (e.g., MS-COCO) become more diversified, the mixture model or exemplar-based methods inevitably lead to more miscellaneous detection models. A question then naturally arises: is there a unified multi-scale approach to detect objects of different aspect ratios? The introduction of "object proposals" (to be introduced) has answered this question.

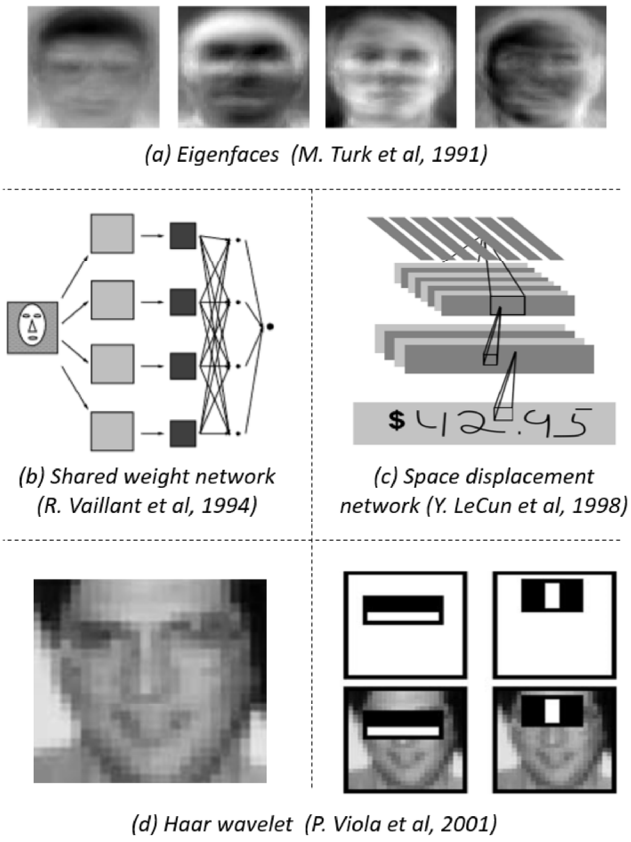


Fig. 5. Some well-known detection models of the early time: (a) Eigenfaces [95], (b) Shared weight networks [96], (c) Space displacement networks (Lenet-5) [97], (d) Haar wavelets of VJ detector [10].

- Detection with object proposals (2010-2015)

Object proposals refer to a group of class-agnostic candidate boxes that likely to contain any objects. It was first time applied in object detection in 2010 [116]. Detection with object proposals helps to avoid the exhaustive sliding window search across an image.

An object proposal detection algorithm should meet the following three requirements: 1) high recall rate, 2) high localization accuracy, and 3) on basis of the first two requirements, to improve precision and reduce processing time. Modern proposal detection methods can be divided into three categories: 1) segmentation grouping approaches [42, 117–119], 2) window scoring approaches [116, 120–122], and 3) neural network based approaches [123–128]. We refer readers to the following papers for a comprehensive review of these methods [129, 130].

Early time's proposal detection methods followed a bottom-up detection philosophy [116, 120] and were deeply affected by visual saliency detection. Later, researchers started to move to low-level vision (e.g., edge detection) and more careful handcrafted skills to improve the localization of candidate boxes [42, 117–119, 122, 131]. After 2014, with the popularity of deep CNN in visual recognition, the top-down, learning-based approaches began to show more advantages in this problem [19, 121, 123, 124]. Since then, the object proposal detection has evolved from the bottom-up vision to "overfitting to a specific set of object classes", and the distinction between detectors and proposal generators is becoming blurred [132].

As "object proposal" has revolutionized the sliding window detection and has quickly dominated the deep learning based detectors, in 2014-2015, many researchers began to ask the following questions: what is the main role of the object proposals in detection? Is it for improving accuracy, or simply for detection speed up? To answer this question, some researchers have tried to weaken the role of the proposals [133] or simply perform sliding window detection on CNN features [134–138], but none of them obtained satisfactory results. The proposal detection has soon slipped out of sight after the rise of one-stage detectors and "deep regression" techniques (to be introduced).

- Deep regression (2013-2016)

In recent years, as the increase of GPU's computing power, the way people deal with multi-scale detection has become more and more straight forward and brute-force. The idea of using the deep regression to solve multi-scale problems is very simple, i.e., to directly predict the coordinates of a bounding box based on the deep learning features [20, 104]. The advantage of this approach is that it is simple and easy to implement while the disadvantage is the localization may not be accurate enough especially for some small objects. "Multi-reference detection" (to be introduced) has latter solved this problem.

- Multi-reference/-resolution detection (after 2015)

Multi-reference detection is the most popular framework for multi-scale object detection [19, 21, 44, 48]. Its main idea is to pre-define a set of reference boxes (a.k.a. anchor boxes) with different sizes and aspect-ratios at different locations of an image, and then predict the detection box based on these references.

A typical loss of each predefined anchor box consists of two parts: 1) a cross-entropy loss for category recognition and 2) an L1/L2 regression loss for object localization. A general form of the loss function can be written as follows:

$$L(p, p^*, t, t^*) = L_{cls.}(p, p^*) + \beta I(t) L_{loc.}(t, t^*)$$
$$I(t) = \begin{cases} 1 & \text{IOU}\{a, a^*\} > \eta \\ 0 & \text{else} \end{cases} \quad (1)$$

where t and t^* are the locations of predicted and ground-truth bounding box, p and p^* are their category probabilities. IOU$\{a, a^*\}$ is the IOU between the anchor a and its ground-truth a^*. η is an IOU threshold, say, 0.5. If an anchor that does not cover any objects, its localization loss does not count in the final loss.

Another popular technique in the last two years is multi-resolution detection [21, 22, 55, 105], i.e. by detecting objects of different scales at different layers of the network. Since a CNN naturally forms a feature pyramid during its forward propagation, it is easier to detect larger objects in deeper layers and smaller ones in shallower layers. Multi-reference and multi-resolution detection have now become two basic building blocks in the state of the art object detection systems.

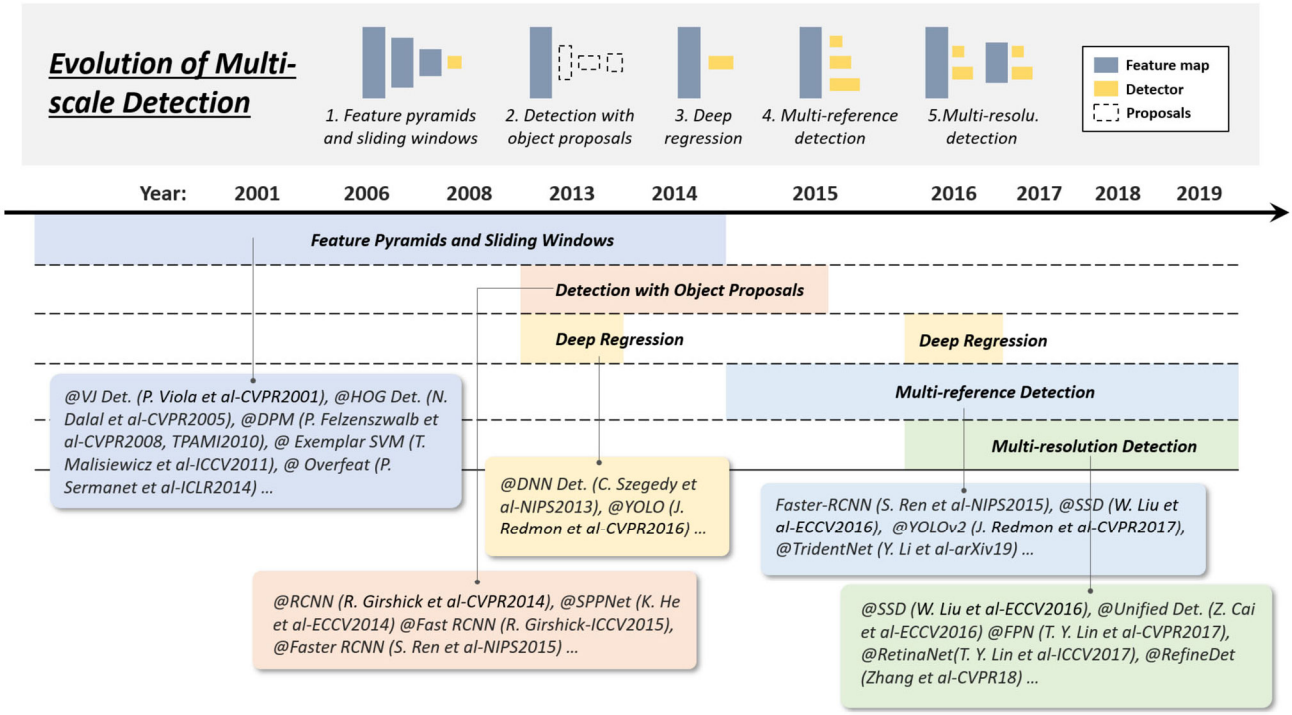


Fig. 6. Evolution of multi-scale detection techniques in object detection from 2001 to 2019: 1) feature pyramids and sliding windows, 2) detection with object proposals, 3) deep regression, 4) multi-reference detection, and 5) multi-resolution detection. Detectors in this figure: VJ Det. [10], HOG Det. [12], DPM [13, 15], Exemplar SVM [36], Overfeat [103], RCNN [16], SPPNet [17], Fast RCNN [18], Faster RCNN [19], DNN Det. [104], YOLO [20], YOLO-v2 [48], SSD [21], Unified Det. [105], FPN [22], RetinaNet [23], RefineDet [55], TridentNet [56].

2.3.3 Technical Evolution of Bounding Box Regression

The Bounding Box (BB) regression is an important technique in object detection. It aims to refine the location of a predicted bounding box based on the initial proposal or the anchor box. In the past 20 years, the evolution of BB regression has gone through three historical periods: "without BB regression (before 2008)", "from BB to BB (2008-2013)", and "from feature to BB (after 2013)". Fig. 7 shows the evolutions of bounding box regression.

- Without BB regression (before 2008)

Most of the early detection methods such as VJ detector and HOG detector do not use BB regression, and usually directly consider the sliding window as the detection result. To obtain accurate locations of an object, researchers have no choice but to build very dense pyramid and slide the detector densely on each location.

- From BB to BB (2008-2013)

The first time that BB regression was introduced to an object detection system was in DPM [15]. The BB regression at that time usually acted as a post-processing block, thus it is optional. As the goal in the PASCAL VOC is to predict single bounding box for each object, the simplest way for a DPM to generate final detection should be directly using its root filter locations. Later, R. Girshick *et al.* introduced a more complex way to predict a bounding box based on the complete configuration of an object hypothesis and formulate this process as a linear least-squares regression problem [15]. This method yields noticeable improvements of the detection under PASCAL criteria.

- From features to BB (after 2013)

After the introduction of Faster RCNN in 2015, BB regression no longer serves as an individual post-processing block but has been integrated with the detector and trained in an end-to-end fashion. At the same time, BB regression has evolved to predicting BB directly based on CNN features. In order to get more robust prediction, the smooth-L1 function [19] is commonly used,

$$L(t) = \begin{cases} 5t^2 & |t| \leqslant 0.1 \\ |t| - 0.05 & \text{else,} \end{cases} \tag{2}$$

or the root-square function [20],

$$L(x, x^*) = (\sqrt{x} - \sqrt{x^*})^2, \tag{3}$$

as their regression loss, which are more robust to the outliers than the least square loss used in DPM. Some researchers also choose to normalize the coordinates to get more robust results [18, 19, 21, 23].

2.3.4 Technical Evolution of Context Priming

Visual objects are usually embedded in a typical context with the surrounding environments. Our brain takes advantage of the associations among objects and environments to facilitate visual perception and cognition [160]. Context priming has long been used to improve detection. There are three common approaches in its evolutionary history: 1) detection with local context, 2) detection with global context, and 3) context interactives, as shown in Fig. 8.

- Detection with local context

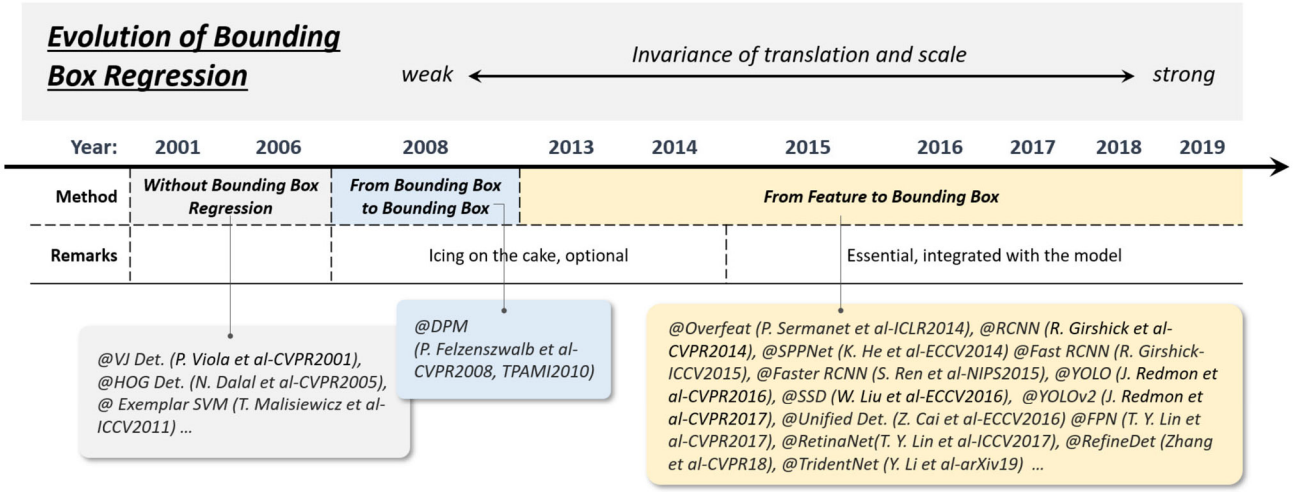


Fig. 7. Evolution of bounding box regression techniques in object detection from 2001 to 2019. Detectors in this figure: VJ Det. [10], HOG Det. [12], Exemplar SVM [36], DPM [13, 15], Overfeat [103], RCNN [16], SPPNet [17], Fast RCNN [18], Faster RCNN [19], YOLO [20], SSD [21], YOLO-v2 [48], Unified Det. [105], FPN [22], RetinaNet [23], RefineDet [55], TridentNet [56].

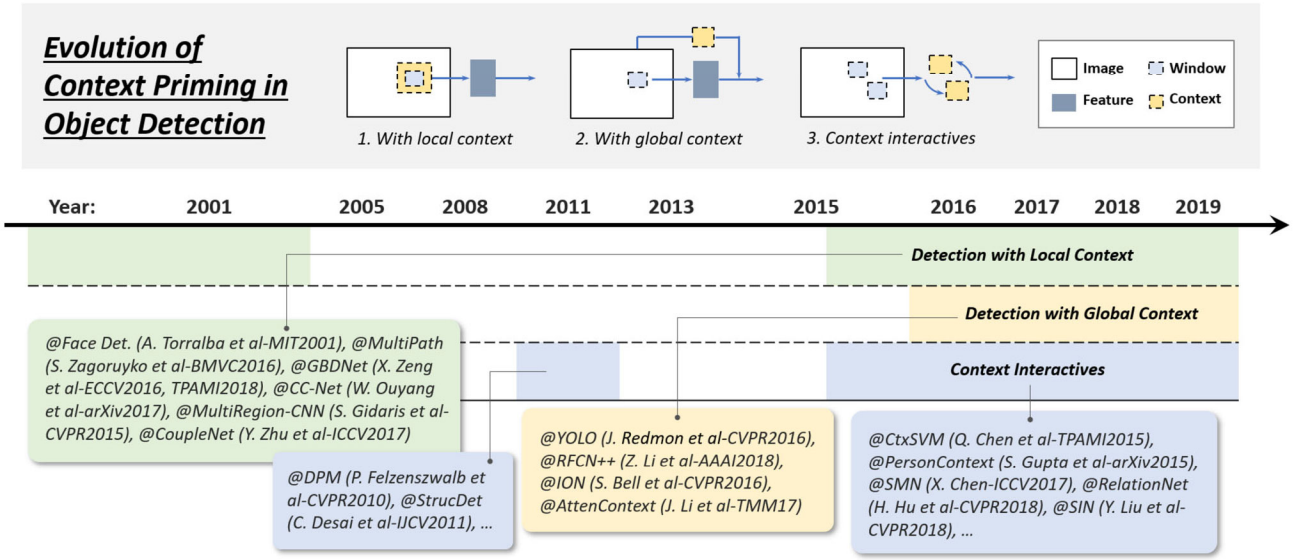


Fig. 8. Evolution of context priming in object detection from 2001 to 2019: 1) detection with local context, 2) detection with global context, 3) detection with context interactives. Detectors in this figure: Face Det. [139], MultiPath [140], GBDNet [141, 142], CC-Net [143], MultiRegion-CNN [144], CoupleNet [145], DPM [14, 15], StructDet [146], YOLO [20], RFCN++ [147], ION [148], AttenContext [149], CtxSVM [150], PersonContext [151], SMN [152], RetinaNet [23], SIN [153].

Local context refers to the visual information in the area that surrounds the object to detect. It has long been acknowledged that local context helps improve object detection. At early 2000s, Sinha and Torralba [139] found that inclusion of local contextual regions such as the facial bounding contour substantially improves face detection performance. Dalal and Triggs also found that incorporating a small amount of background information improves the accuracy of pedestrian detection [12]. Recent deep learning based detectors can also be improved with local context by simply enlarging the networks' receptive field or the size of object proposals [140–145, 161].

- Detection with global context

Global context exploits scene configuration as an additional source of information for object detection. For early time's object detectors, a common way of integrating global context is to integrate a statistical summary of the elements that comprise the scene, like Gist [160]. For modern deep learning based detectors, there are two methods to integrate global context. The first way is to take advantage of large receptive field (even larger than the input image) [20] or global pooling operation of a CNN feature [147]. The second way is to think of the global context as a kind of sequential information and to learn it with the recurrent neural networks [148, 149].

- Context interactive

Context interactive refers to the piece of information that conveys by the interactions of visual elements, such as the constraints and dependencies. For most object detectors, object instances are detected and recognized individually

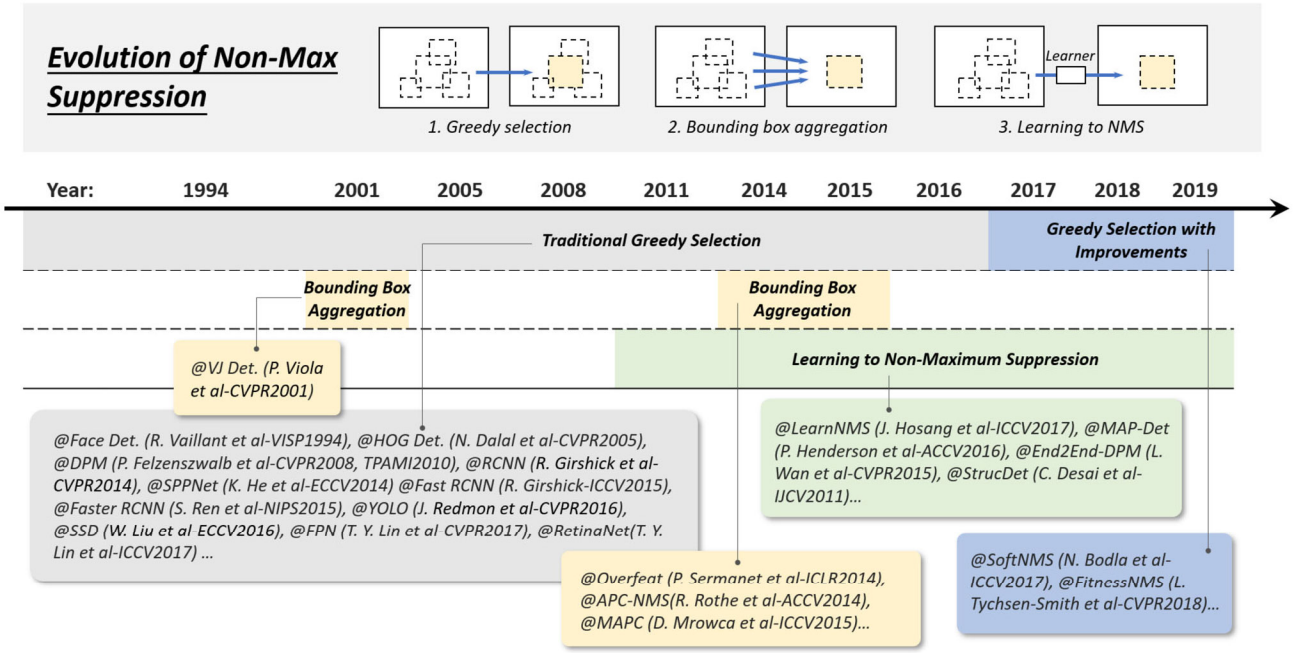


Fig. 9. Evolution of non-max suppression (NMS) techniques in object detection from 1994 to 2019: 1) Greedy selection, 2) Bounding box aggregation, and 3) Learn to NMS. Detectors in this figure: VJ Det. [10], Face Det. [96], HOG Det. [12], DPM [13, 15], RCNN [16], SPPNet [17], Fast RCNN [18], Faster RCNN [19], YOLO [20], SSD [21], FPN [22], RetinaNet [23], LearnNMS [154], MAP-Det [155], End2End-DPM [136], StrucDet [146], Overfeat [103], APC-NMS [156], MAPC [157], SoftNMS [158], FitnessNMS [159].

without exploiting their relations. Some recent researches have suggested that modern object detectors can be improved by considering context interactives. Some recent improvements can be grouped into two categories, where the first one is to explore the relationship between individual objects [15, 146, 150, 152, 162], and the second one is to explore modeling the dependencies between objects and scenes [151, 151, 153].

2.3.5 Technical Evolution of Non-Maximum Suppression

Non-maximum suppression (NMS) is an important group of techniques in object detection. As the neighboring windows usually have similar detection scores, the non-maximum suppression is herein used as a post-processing step to remove the replicated bounding boxes and obtain the final detection result. At early times of object detection, NMS was not always integrated [30]. This is because the desired output of an object detection system was not entirely clear at that time. During the past 20 years, NMS has been gradually developed into the following three groups of methods: 1) greedy selection, 2) bounding box aggregation, and 3) learning to NMS, as shown in Fig. 9.

- Greedy selection

Greedy selection is an old fashioned but the most popular way to perform NMS in object detection. The idea behind this process is simple and intuitive: for a set of overlapped detections, the bounding box with the maximum detection score is selected while its neighboring boxes are removed according to a predefined overlap threshold (say, 0.5). The above processing is iteratively performed in a greedy manner.

Although greedy selection has now become the de facto method for NMS, it still has some space for improvement, as shown in Fig 11. First of all, the top-scoring box may not be the best fit. Second, it may suppress nearby objects. Finally, it does not suppress false positives. In recent years, in spite of the fact that some manual modifications have been recently made to improve its performance [158, 159, 163] (see Section 4.4 for more details), to our best knowledge, the greedy selection still performs as the strongest baseline for today's object detection.

- BB aggregation

BB aggregation is another group of techniques for NMS [10, 103, 156, 157] with the idea of combining or clustering multiple overlapped bounding boxes into one final detection. The advantage of this type of method is that it takes full consideration of object relationships and their spatial layout. There are some well-known detectors using this method, such as the VJ detector [10] and the Overfeat [103].

- Learning to NMS

A recent group of NMS improvements that have recently received much attention is learning to NMS [136, 146, 154, 155]. The main idea of such group of methods is to think of NMS as a filter to re-score all raw detections and to train the NMS as part of a network in an end-to-end fashion. These methods have shown promising results on improving occlusion and dense object detection over traditional hand-crafted NMS methods.

2.3.6 Technical Evolution of Hard Negative Mining

The training of an object detector is essentially an imbalanced data learning problem. In the case of sliding window based detectors, the imbalance between backgrounds and objects could be as extreme as $10^4 \sim 10^5$ background

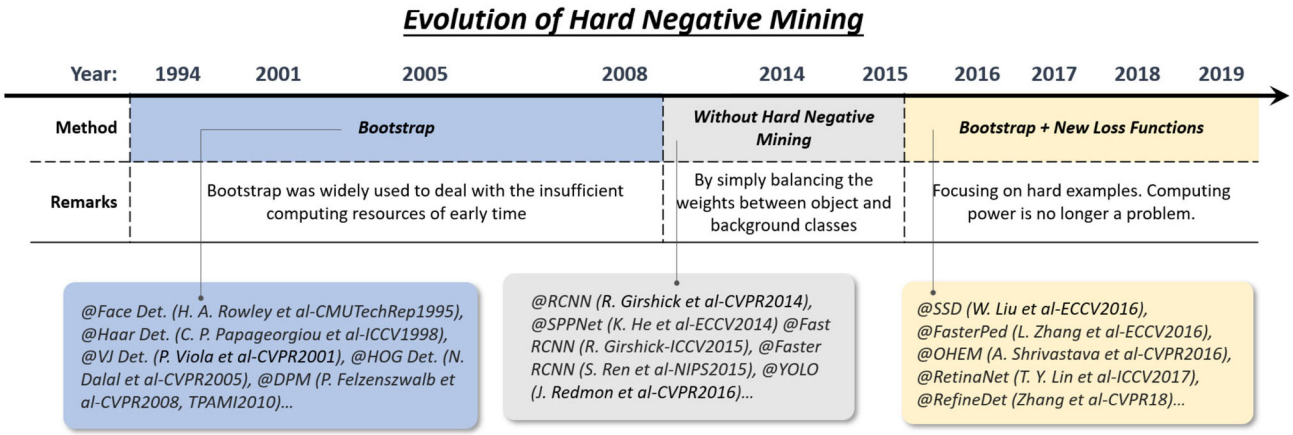


Fig. 10. Evolution of hard negative mining techniques in object detection from 1994 to 2019. Detectors in this figure: Face Det. [164], Haar Det. [29], VJ Det. [10], HOG Det. [12], DPM [13, 15], RCNN [16], SPPNet [17], Fast RCNN [18], Faster RCNN [19], YOLO [20], SSD [21], FasterPed [165], OHEM [166], RetinaNet [23], RefineDet [55].

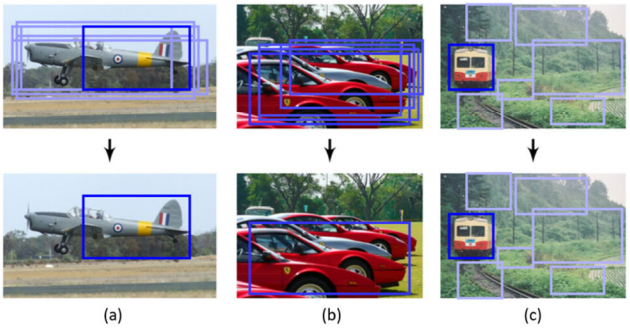


Fig. 11. Examples of possible failures when using a standard greedy selection based non-max suppression: (a) the top-scoring box may not be the best fit, (b) it may suppress nearby objects, and (c) it does not suppress false positives. Images from *R. Rothe et al. ACCV2014* [156].

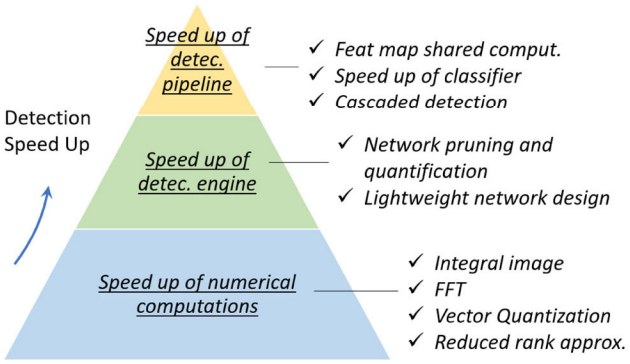


Fig. 12. An overview of the speed up techniques in object detection.

windows to every object. Modern detection datasets require the prediction of object aspect ratio, further increasing the imbalanced ratio to $10^6 \sim 10^7$ [129]. In this case, using all background data will be harmful to training as the vast number of easy negatives will overwhelm the learning process. Hard negative mining (HNM) aims to deal with the problem of imbalanced data during training. The technical evolution of HNM in object detection is shown in Fig. 10.

- Bootstrap

Bootstrap in object detection refers to a group of training techniques in which the training starts with a small part of background samples and then iteratively add new misclassified backgrounds during the training process. In early times object detectors, bootstrap was initially introduced with the purpose of reducing the training computations over millions of background samples [10, 29, 164]. Later it became a standard training technique in DPM and HOG detectors [12, 13] for solving the data imbalance problem.

- HNM in deep learning based detectors

Later in the deep learning era, due to the improvement of computing power, bootstrap was shortly discarded in object detection during 2014-2016 [16–20]. To ease the data-imbalance problem during training, detectors like Faster RCNN and YOLO simply balance the weights between the positive and negative windows. However, researchers later noticed that the weight-balancing cannot completely solve the imbalanced data problem [23]. To this end, after 2016, the bootstrap was re-introduced to deep learning based detectors [21, 165–168]. For example, in SSD [21] and OHEM [166], only the gradients of a very small part of samples (those with the largest loss values) will be back-propagated. In RefineDet [55], an "anchor refinement module" is designed to filter easy negatives. An alternative improvement is to design new loss functions [23, 169, 170], by reshaping the standard cross entropy loss so that it will put more focus on hard, misclassified examples [23].

3 Speed-Up of Detection

The acceleration of object detection has long been an important but challenging problem. In the past 20 years, the object detection community has developed sophisticated acceleration techniques. These techniques can be roughly divided into three levels of groups: "speed up of detection pipeline", "speed up of detection engine", and "speed up of numerical computation", as shown in Fig 12.

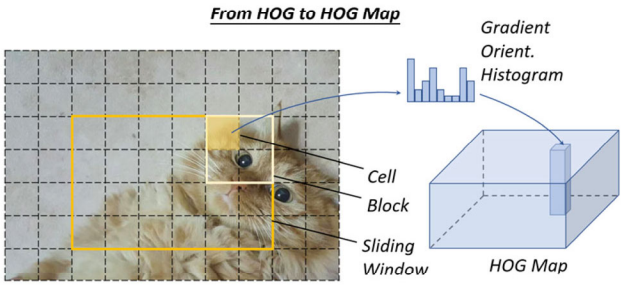


Fig. 13. An illustration of how to compute the HOG map of an image.

3.1 Feature Map Shared Computation

Among the different computational stages of an object detector, the feature extraction usually dominates the amount of computation. For a sliding window based detector, the computational redundancy starts from both positions and scales, where the former one is caused by the overlap between adjacent windows, while the later one is by the feature correlation between adjacent scales.

3.1.1 Spatial Computational Redundancy and Speed Up

The most commonly used idea to reduce the spatial computational redundancy is feature map shared computation, i.e., to compute the feature map of the whole image only once before sliding window on it. The "image pyramid" of a traditional detector herein can be considered as a "feature pyramid". For example, to speed up HOG pedestrian detector, researchers usually accumulate the "HOG map" of the whole input image, as shown in Fig. 13. However, the drawback of this method is also obvious, i.e., the feature map resolution (the minimum step size of the sliding window on this feature map) will be limited by the cell size. If a small object is located between two cells, it could be ignored by all detection windows. One solution to this problem is to build an integral feature pyramid, which will be introduced in Section 3.6.

The idea of feature map shared computation has also been extensively used in convolutional based detectors. Some related works can be traced back to the 1990s [96, 97]. Most of the CNN based detectors in recent years, e.g., SPPNet [17], Fast-RCNN [18], and Faster-RCNN [19], have applied similar ideas, which have achieved tens or even hundreds of times of acceleration.

3.1.2 Scale Computational Redundancy and Speed Up

To reduce the scale computational redundancy, the most successful way is to directly scale the features rather than the images, which has been first applied in the VJ detector [10]. However, such an approach cannot be applied directly to HOG-like features because of blurring effects. For this problem, P. Dollár *et al.* discovered the strong (log-linear) correlation between the neighbor scales of the HOG and integral channel features [171] through extensive statistical analysis. This correlation can be used to accelerate the computation of a feature pyramid [172] by approximating the feature maps of adjacent scales. Besides, building "detector pyramid" is another way to avoid scale computational redundancy, i.e., to detect objects of different scales by simply sliding multiple detectors on one feature map rather than re-scaling the image or features [173].

3.2 Speed up of Classifiers

Traditional sliding window based detectors, e.g., HOG detector and DPM, prefer using linear classifiers than nonlinear ones due to their low computational complexity. Detection with nonlinear classifiers such as kernel SVM suggests higher accuracy, but at the same time brings high computational overhead. As a standard non-parametric method, the traditional kernel method has no fixed computational complexity. When we have a very large training set, the detection speed will become extremely slow.

In object detection, there are many ways to speed up kernelized classifiers, where the "model approximation" is most commonly used [30, 174]. Since the decision boundary of a classical kernel SVM can only be determined by a small set of its training samples (support vectors), the computational complexity at the inference stage would be proportional to the number of support vectors: $O(N_{sv})$. Reduced Set Vectors [30] is an approximation method for kernel SVM, which aims to obtain an equivalent decision boundary in terms of a small number of synthetic vectors. Another way to speed up kernel SVM in object detection is to approximate its decision boundary to a piece-wise linear form so as to achieve a constant inference time [174]. The kernel method can also be accelerated with the sparse encoding methods [175].

3.3 Cascaded Detection

Cascaded detection is a commonly used technique in object detection [10, 176]. It takes a coarse to fine detection philosophy: to filter out most of the simple background windows using simple calculations, then to process those more difficult windows with complex ones. The VJ detector is a representative of cascaded detection. After that, many subsequent classical object detectors such as the HOG detector and DPM have been accelerated by using this technique [14, 38, 54, 177, 178].

In recent years, cascaded detection has also been applied to deep learning based detectors, especially for those detection tasks of "small objects in large scenes" , e.g., face detection [179, 180], pedestrian detection [165, 177, 181], etc. In addition to the algorithm acceleration, cascaded detection has been applied to solve other problems, e.g., to improve the detection of hard examples [182–184], to integrate context information [143, 185], and to improve localization accuracy [104, 125].

3.4 Network Pruning and Quantification

"Network pruning" and "network quantification" are two commonly used techniques to speed up a CNN model, where the former one refers to pruning the network structure or weight to reduce its size and the latter one refers to reducing the code-length of activations or weights.

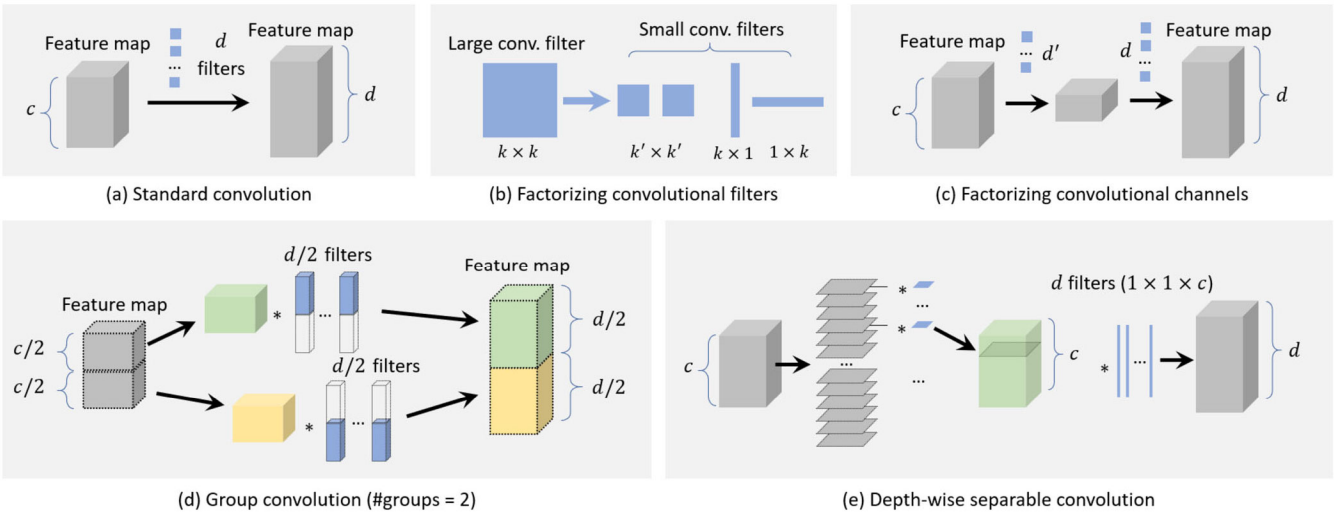


Fig. 14. An overview of speed up methods of a CNN's convolutional layer and the comparison of their computational complexity: (a) Standard convolution: $\mathcal{O}(dk^2c)$. (b) Factoring convolutional filters ($k \times k \rightarrow (k^{'} \times k^{'})^2$ or $1 \times k, k \times 1$): $\mathcal{O}(dk^{'2}c)$ or $\mathcal{O}(dkc)$. (c) Factoring convolutional channels: $\mathcal{O}(d^{'}k^2c) + \mathcal{O}(dk^2d^{'})$. (d) Group convolution (#groups=m): $\mathcal{O}(dk^2c/m)$. (e) Depth-wise separable convolution: $\mathcal{O}(ck^2) + \mathcal{O}(dc)$.

3.4.1 Network Pruning

The research of "network pruning" can be traced back to as early as the 1980s. At that time, Y. LeCun *et al.* proposed a method called "optimal brain damage" to compress the parameters of a multi-layer perceptron network [186]. In this method, the loss function of a network is approximated by taking the second-order derivatives so that to remove some unimportant weights. Following this idea, the network pruning methods in recent years usually take an iterative training and pruning process, i.e., to remove only a small group of unimportant weights after each stage of training, and to repeat those operations [187]. As traditional network pruning simply removes unimportant weights, which may result in some sparse connectivity patterns in a convolutional filter, it can not be directly applied to compress a CNN model. A simple solution to this problem is to remove the whole filters instead of the independent weights [188, 189].

3.4.2 Network Quantification

The recent works on network quantification mainly focus on network binarization, which aims to accelerate a network by quantifying its activations or weights to binary variables (say, 0/1) so that the floating-point operation is converted to AND, OR, NOT logical operations. Network binarization can significantly speed up computations and reduce the network's storage so that it can be much easier to be deployed on mobile devices. One possible implementation of the above ideas is to approximate the convolution by binary variables with the least squares method [190]. A more accurate approximation can be obtained by using linear combinations of multiple binary convolutions [191]. In addition, some researchers have further developed GPU acceleration libraries for binarized computation, which obtained more significant acceleration results [192].

3.4.3 Network Distillation

Network distillation is a general framework to compress the knowledge of a large network ("teacher net") into a small one ("student net") [193, 194]. Recently, this idea has been used in the acceleration of object detection [195, 196]. One straight forward approach of this idea is to use a teacher net to instruct the training of a (light-weight) student net so that the latter can be used for speed up detection [195]. Another approach is to make transform of the candidate regions so as to minimize the features distance between the student net and teacher net. This method makes the detection model 2 times faster while achieving a comparable accuracy [196].

3.5 Lightweight Network Design

The last group of methods to speed up a CNN based detector is to directly design a lightweight network instead of using off-the-shelf detection engines. Researchers have long been exploring the right configurations of a network so that to gain accuracy under a constrained time cost. In addition to some general designing principles like "fewer channels and more layers" [197], some other approaches have been proposed in recent years: 1) factorizing convolutions, 2) group convolution, 3) depth-wise separable convolution, 4) bottle-neck design, and 5) neural architecture search.

3.5.1 Factorizing Convolutions

Factorizing convolutions is the simplest and most straight forward way to build a lightweight CNN model. There are two groups of factorizing methods.

The first group of methods is to factorize a large convolution filter into a set of small ones in their spatial dimension [47, 147, 198], as shown in Fig. 14 (b). For example, one can factorize a 7x7 filter into three 3x3 filters, where they share the same receptive field but the later one is more efficient. Another example is to factorize a $k \times k$ filter into a $k \times 1$ filter and a $1 \times k$ filter [198, 199], which could be more efficient for very large filters, say 15x15 [199]. This idea has been recently used in object detection [200].

The second group of methods is to factorize a large group of convolutions into two small groups in their chan-

nel dimension [201, 202], as shown in Fig. 14 (c). For example, one can approximate a convolution layer with d filters and a feature map of c channels by d' filters + a nonlinear activation + another d filters ($d' < d$). In this case, the complexity $\mathcal{O}(dk^2c)$ of the original layer can be reduced to $\mathcal{O}(d'k^2c) + \mathcal{O}(dd')$.

3.5.2 Group Convolution

Group convolution aims to reduce the number of parameters in a convolution layer by dividing the feature channels into many different groups, and then convolve on each group independently [189, 203], as shown in Fig. 14 (d). If we evenly divide the feature channels into m groups, without changing other configurations, the computational complexity of the convolution will theoretically be reduced to $1/m$ of that before.

3.5.3 Depth-wise Separable Convolution

Depth-wise separable convolution, as shown in Fig. 14 (e), is a recent popular way of building lightweight convolution networks [204]. It can be viewed as a special case of the group convolution when the number of groups is set equal to the number of channels.

Suppose we have a convolutional layer with d filters and a feature map of c channels. The size of each filter is $k \times k$. For a depth-wise separable convolution, every $k \times k \times c$ filter is first to split into c slices each with the size of $k \times k \times 1$, and then the convolutions are performed individually in each channel with each slice of the filter. Finally, a number of 1x1 filters are used to make a dimension transform so that the final output should have d channels. By using depth-wise separable convolution, the computational complexity can be reduced from $\mathcal{O}(dk^2c)$ to $\mathcal{O}(ck^2) + \mathcal{O}(dc)$. This idea has been recently applied to object detection and fine-grain classification [205–207].

3.5.4 Bottle-neck Design

A bottleneck layer in a neural network contains few nodes compared to the previous layers. It can be used to learning efficient data encodings of the input with reduced dimensionality, which has been commonly used in deep autoencoders [208]. In recent years, the bottle-neck design has been widely used for designing lightweight networks [47, 209–212]. Among these methods, one common approach is to compress the input layer of a detector to reduce the amount of computation from the very beginning of the detection pipeline [209–211]. Another approach is to compress the output of the detection engine to make the feature map thinner, so as to make it more efficient for subsequent detection stages [47, 212].

3.5.5 Neural Architecture Search

More recently, there has been significant interest in designing network architectures automatically by neural architecture search (NAS) instead of relying heavily on expert experience and knowledge. NAS has been applied to large-scale image classification [213, 214], object detection [215] and image segmentation [216] tasks. NAS also shows promising results in designing lightweight networks very recently, where the constraints on the prediction accuracy and computational complexity are both considered during the searching process [217, 218].

3.6 Numerical Acceleration

In this section, we mainly introduce four important numerical acceleration methods that are frequently used in object detection: 1) speed up with the integral image, 2) speed up in the frequency domain, 3) vector quantization, and 4) reduced rank approximation.

3.6.1 Speed Up with Integral Image

The integral image is an important method in image processing. It helps to rapidly calculate summations over image sub-regions. The essence of integral image is the integral-differential separability of convolution in signal processing:

$$f(x) * g(x) = (\int f(x)dx) * (\frac{dg(x)}{dx}), \tag{4}$$

where if $dg(x)/dx$ is a sparse signal, then the convolution can be accelerated by the right part of this equation. Although the VJ detector [10] is well known for the integral image acceleration, before it was born, the integral image has already been used to speed up a CNN model [219] and achieved more than 10 times acceleration.

In addition to the above examples, integral image can also be used to speed up more general features in object detection, e.g., color histogram, gradient histogram [171, 177, 220, 221], etc. A typical example is to speed up HOG by computing integral HOG maps [177, 220]. Instead of accumulating pixel values in a traditional integral image, the integral HOG map accumulates gradient orientations in an image, as shown in Fig. 15. As the histogram of a cell can be viewed as the summation of the gradient vector in a certain region, by using the integral image, it is possible to compute a histogram in a rectangle region of an arbitrary position and size with a constant computational overhead. The integral HOG map has been used in pedestrian detection and has achieved dozens of times' acceleration without losing any accuracy [177].

Later in 2009, P. Dollár *et al.* proposed a new type of image feature called Integral Channel Features (ICF), which can be considered as a more general case of the integral image features, and has been successfully used in pedestrian detection [171]. ICF achieves state-of-the-art detection accuracy under the near realtime detection speed in its time.

3.6.2 Speed Up in Frequency Domain

Convolution is an important type of numerical operation in object detection. As the detection of a linear detector can be viewed as the window-wise inner product between the feature map and detector's weights, this process can be implemented by convolutions.

The convolution can be accelerated in many ways, where the Fourier transform is a very practical choice especially for speeding up those large filters. The theoretical basis for accelerating convolution in the frequency domain is the convolution theorem in signal processing, that is, under suitable conditions, the Fourier transform of a convolution of two signals is the point-wise product in their Fourier space:

$$I * W = F^{-1}(F(I) \odot F(W)) \tag{5}$$

where F is Fourier transform, F^{-1} is Inverse Fourier transform, I and W are the input image and filter, $*$ is the

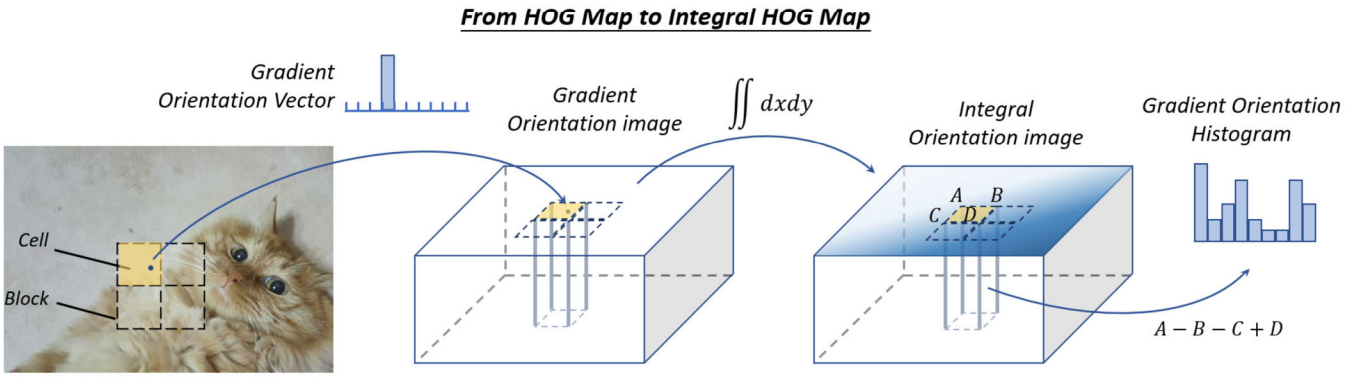


Fig. 15. An illustration of how to compute the "Integral HOG Map" [177]. With integral image techniques, we can efficiently compute the histogram feature of any location and any size with constant computational complexity.

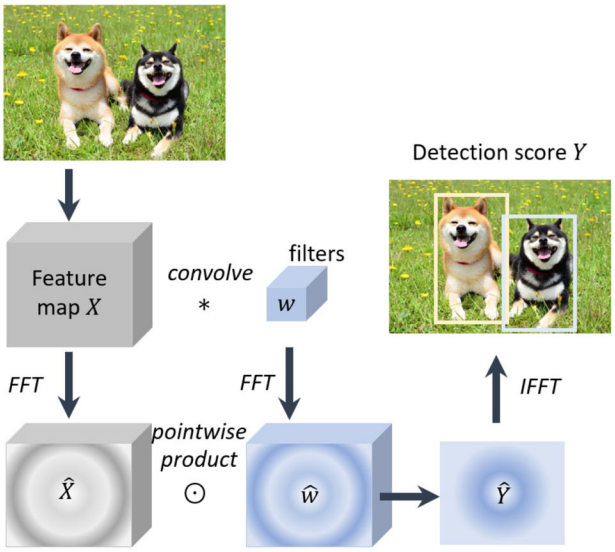


Fig. 16. An illustration of how to speed up a linear detector (e.g., HOG detector, DPM, etc) in frequency domain with fast Fourier transform and inverse fast Fourier transform [226].

convolution operation, and $\odot$ is the point-wise product. The above calculation can be accelerated by using the Fast Fourier Transform (FFT) and the Inverse Fast Fourier Transform (IFFT). FFT and IFFT have now been frequently used to speed up CNN models [222–225] and some classical linear object detectors [226], which has improved the detection speed over an order of magnitude. Fig. 16 shows a standard pipeline to speed up a linear object detector (e.g., HOG and DPM) in the frequency domain.

3.6.3 Vector Quantization

The Vector Quantization (VQ) is a classical quantization method in signal processing that aims to approximate the distribution of a large group of data by a small set of prototype vectors. It can be used for data compression and accelerating the inner product operation in object detection [227, 228]. For example, with VQ, the HOG histograms can be grouped and quantified into a set of prototype histogram vectors. Then in the detection stage, the inner production between the feature vector and detection weights can be implemented by a table-look-up operation. As there is no floating point multiplication and division in this process, the speed of a DPM and exemplar SVM detector can be accelerated over an order of magnitude [227].

3.6.4 Reduced Rank Approximation

In deep networks, the computation in a fully-connected layer is essentially a multiplication of two matrices. When the parameter matrix $\mathbf{W} \in \mathbb{R}^{u \times v}$ is large, the computing burden of a detector will be heavy. For example, in Fast RCNN detector [18] nearly half of the forward pass time is spent in computing the fully connected layers. The reduced rank approximation is a method to accelerate matrix multiplications. It aims to make a low-rank decomposition of the matrix $\mathbf{W}$:

$$\mathbf{W} \approx \mathbf{U}\mathbf{\Sigma}_t\mathbf{V}, \tag{6}$$

where $\mathbf{U}$ is a $u \times t$ matrix comprising of the first t left-singular vectors of $\mathbf{W}$, $\mathbf{\Sigma}_t$ is a $t \times t$ diagonal matrix containing the top t singular values of $\mathbf{W}$, and $\mathbf{V}$ is $v \times t$ matrix comprising of the first t right-singular vectors of $\mathbf{W}$. The above process, also known as the Truncated SVD, reduces the parameter count from uv to $t(u + v)$, which can be significant if t is much smaller than $\min(u, v)$. Truncated SVD has been used to accelerate the Fast RCNN detector [18] and achieves x2 speed up.

4 Recent Advances in Object Detection

In this section, we will review the state of the art object detection methods in recent three years.

4.1 Detection with Better Engines

In recent years, deep CNN has played a central role in many computer vision tasks. As the accuracy of a detector depends heavily on its feature extraction networks, in this paper, we refer to the backbone networks, e.g. the ResNet and VGG, as the "engine" of a detector. Fig. 17 shows the detection accuracy of three well-known detection systems: Faster RCNN [19], R-FCN [46] and SSD [21] with different choices of the engines [27].

In this section, we will introduce some of the important detection engines in deep learning era. We refer readers to the following survey for more details on this topic [229].

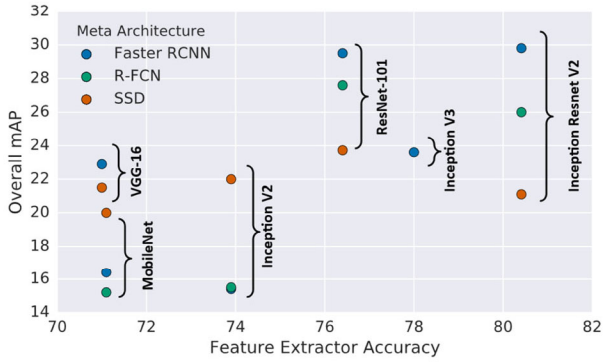


Fig. 17. A comparison of detection accuracy of three detectors: Faster RCNN [19], R-FCN [46] and SSD [21] on MS-COCO dataset with different detection engines. Image from *J. Huang et al. CVPR2017* [27].

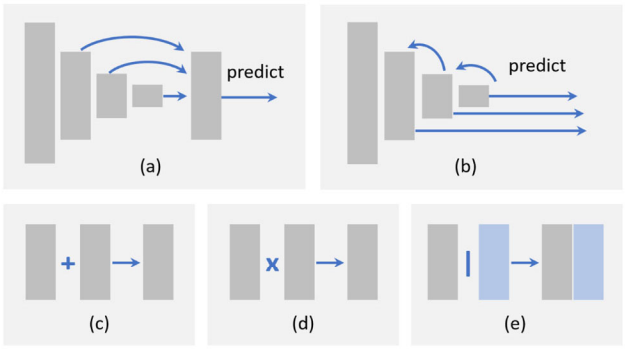


Fig. 18. An illustration of different feature fusion methods: (a) bottom-up fusion, (b) top-down fusion, (c) element-wise sum, (d) element-wise product, and (e) concatenation.

AlexNet: AlexNet [40], an eight-layer deep network, was the first CNN model that started the deep learning revolution in computer vision. AlexNet famously won the 2012 ImageNet LSVRC-2012 competition by a large margin [15.3% VS 26.2% (second place) error rates]. As of Feb. 2019, the Alexnet paper has been cited over 30,000 times.

VGG: VGG was proposed by Oxford's Visual Geometry Group (VGG) in 2014 [230]. VGG increased the model's depth to 16-19 layers and used very small (3x3) convolution filters instead of 5x5 and 7x7 those were previously used in AlexNet. VGG has achieved the state of the art performance on the ImageNet dataset of its time.

GoogLeNet: GoogLeNet, a.k.a Inception [198, 231–233], is a big family of CNN models proposed by Google Inc. since 2014. GoogLeNet increased both of a CNN's width and depth (up to 22 layers). The main contribution of the Inception family is the introduction of factorizing convolution and batch normalization.

ResNet: The Deep Residual Networks (ResNet) [234], proposed by K. He *et al.* in 2015, is a new type of convolutional network architecture that is substantially deeper (up to 152 layers) than those used previously. ResNet aims to ease the training of networks by reformulating its layers as learning residual functions with reference to the layer inputs. ResNet won multiple computer vision competitions in 2015, including ImageNet detection, ImageNet localization, COCO detection, and COCO segmentation.

DenseNet: DenseNet [235] was proposed by G. Huang and Z. Liu *et al.* in 2017. The success of ResNet suggested that the short cut connection in CNN enables us to train deeper and more accurate models. The authors embraced this observation and introduced a densely connected block, which connects each layer to every other layer in a feed-forward fashion.

SENet: Squeeze and Excitation Networks (SENet) was proposed by J. Hu and L. Shen *et al.* in 2018 [236]. Its main contribution is the integration of global pooling and shuffling to learn channel-wise importance of the feature map. SENet won the 1st place in ILSVRC 2017 classification competition.

- Object detectors with new engines

In recent three years, many of the latest engines have been applied to object detection. For example, some latest object detection models such as STDN [237], DSOD [238], TinyDSOD [207], and Pelee [209] choose DenseNet [235] as their detection engine. The Mask RCNN [4], as the state of the art model for instance segmentation, applied the next generation of ResNet: ResNeXt [239] as its detection engine. Besides, to speed up detection, the depth-wise separable convolution operation, which was introduced by Xception [204], an improved version of Incepion, has also been used in detectors such as MobileNet [205] and LightHead RCNN [47].

4.2 Detection with Better Features

The quality of feature representations is critical for object detection. In recent years, many researchers have made efforts to further improve the quality of image features on basis of some latest engines, where the most important two groups of methods are: 1) feature fusion and 2) learning high-resolution features with large receptive fields.

4.2.1 Why Feature Fusion is Important?

Invariance and equivariance are two important properties in image feature representations. Classification desires invariant feature representations since it aims at learning high-level semantic information. Object localization desires equivariant representations since it aims at discriminating position and scale changes. As object detection consists of two sub-tasks of object recognition and localization, it is crucial for a detector to learn both invariance and equivariance at the same time.

Feature fusion has been widely used in object detection in the last three years. As a CNN model consists of a series of convolutional and pooling layers, features in deeper layers will have stronger invariance but less equivariance. Although this could be beneficial to category recognition, it suffers from low localization accuracy in object detection. On the contrary, features in shallower layers is not conducive to learning semantics, but it helps object localization as it contains more information about edges and contours. Therefore, the integration of deep and shallow features in a CNN model helps improve both invariance and equivariance.

4.2.2 Feature Fusion in Different Ways

There are many ways to perform feature fusion in object detection. Here we introduce some recent methods in two aspects: 1) processing flow and 2) element-wise operation.

- Processing flow

Recent feature fusion methods in object detection can be divided into two categories: 1) bottom-up fusion, 2) top-down fusion, as shown in Fig. 18 (a)-(b). Bottom-up fusion feeds forward shallow features to deeper layers via skip connections [237, 240–242]. In comparison, top-down fusion feeds back the features of deeper layers into the shallower ones [22, 55, 243–246]. Apart from these methods, there are more complex approaches proposed recently, e.g., weaving features across different layers [247].

As the feature maps of different layers may have different sizes both in terms of their spatial and channel dimensions, one may need to accommodate the feature maps, such as by adjusting the number of channels, up-sampling low-resolution maps, or down-sampling high-resolution maps to a proper size. The easiest ways to do this is to use nearest- or bilinear-interpolation [22, 244]. Besides, fractional strided convolution (a.k.a. transpose convolution) [45, 248], is another recent popular way to resize the feature maps and adjust the number of channels. The advantage of using fractional strided convolution is that it can learn an appropriate way to perform up-sampling by itself [55, 212, 241–243, 245, 246, 249].

- Element-wise operation

From a local point of view, feature fusion can be considered as the element-wise operation between different feature maps. There are three groups of methods: 1) element-wise sum, 2) element-wise product, and 3) concatenation, as shown in Fig. 18 (c)-(e).

The element-wise sum is the easiest way to perform feature fusion. It has been frequently used in many recent object detectors [22, 55, 241, 243, 246]. The element-wise product [245, 249–251] is very similar to the element-wise sum, while the only difference is the use of multiplication instead of summation. An advantage of element-wise product is that it can be used to suppress or highlight the features within a certain area, which may further benefit small object detection [245, 250, 251]. Feature concatenation is another way of feature fusion [212, 237, 240, 244]. Its advantage is that it can be used to integrate context information of different regions [105, 144, 149, 161], while its disadvantage is the increase of the memory [235].

4.2.3 Learning High Resolution Features with Large Receptive Fields

The receptive field and feature resolution are two important characteristics of a CNN based detector, where the former one refers to the spatial range of input pixels that contribute to the calculation of a single pixel of the output, and the latter one corresponds to the down-sampling rate between the input and the feature map. A network with a larger receptive field is able to capture a larger scale of context information, while that with a smaller one may concentrate more on the local details.

As we mentioned before, the lower the feature resolution is, the harder will be to detect small objects. The most straight forward way to increase the feature resolution is to remove pooling layer or to reduce the convolution down-sampling rate. But this will cause a new problem, the receptive field will become too small due to the decreasing of output stride. In other words, this will narrow a detector's "sight" and may result in the miss detection of some large objects.

A piratical method to increase both of the receptive field and feature resolution at the same time is to introduce dilated convolution (a.k.a. atrous convolution, or convolution with holes). Dilated convolution is originally proposed in semantic segmentation tasks [252, 253]. Its main idea is to expand the convolution filter and use sparse parameters. For example, a 3x3 filter with a dilation rate of 2 will have the same receptive field as a 5x5 kernel but only have 9 parameters. Dilated convolution has now been widely used in object detection [21, 56, 254, 255], and proves to be effective for improved accuracy without any additional parameters and computational cost [56].

4.3 Beyond Sliding Window

Although object detection has evolved from using hand-crafted features to deep neural networks, the detection still follows a paradigm of "sliding window on feature maps" [137]. Recently, there are some detectors built beyond sliding windows.

- Detection as sub-region search

Sub-region search [184, 256–258] provides a new way of performing detection. One recent method is to think of detection as a path planning process that starts from initial grids and finally converges to the desired ground truth boxes [256]. Another method is to think of detection as an iterative updating process to refine the corners of a predicted bounding box [257].

- Detection as key points localization

Key points localization is an important computer vision task that has extensively broad applications, such as facial expression recognition [259], human poses identification [260], etc. As any object in an image can be uniquely determined by its upper left corner and lower right corner of the ground truth box, the detection task, therefore, can be equivalently framed as a pair-wise key points localization problem. One recent implementation of this idea is to predict a heat-map for the corners [261]. The advantage of this approach is that it can be implemented under a semantic segmentation framework, and there is no need to design multi-scale anchor boxes.

4.4 Improvements of Localization

To improve localization accuracy, there are two groups of methods in recent detectors: 1) bounding box refinement, and 2) designing new loss functions for accurate localization.

4.4.1 Bounding Box Refinement

The most intuitive way to improve localization accuracy is bounding box refinement, which can be considered as a post-processing of the detection results. Although the bounding box regression has been integrated into most of the modern object detectors, there are still some objects with unexpected scales that cannot be well captured by any of the predefined anchors. This will inevitably lead to an inaccurate prediction of their locations. For this reason, the "iterative bounding box refinement" [262–264] has been introduced recently by iteratively feeding the detection results into a BB regressor until the prediction converges to a correct location and size. However, some researchers also claimed that this method does not guarantee the monotonicity of localization accuracy [262], in other words, the BB regression may degenerate the localization if it is applied for multiple times.

4.4.2 Improving Loss Functions for Accurate Localization

In most modern detectors, object localization is considered as a coordinate regression problem. However, there are two drawbacks of this paradigm. First, the regression loss function does not correspond to the final evaluation of localization. For example, we can not guarantee that a lower regression error will always produce a higher IoU prediction, especially when the object has a very large aspect ratio. Second, the traditional bounding box regression method does not provide the confidence of localization. When there are multiple BB's overlapping with each other, this may lead to failure in non-maximum suppression (see more details in subsection 2.3.5).

The above problems can be alleviated by designing new loss functions. The most intuitive design is to directly use IoU as the localization loss function [265]. Some other researchers have further proposed an IoU-guided NMS to improve localization in both training and detection stages [163]. Besides, some researchers have also tried to improve localization under a probabilistic inference framework [266]. Different from the previous methods that directly predict the box coordinates, this method predicts the probability distribution of a bounding box location.

4.5 Learning with Segmentation

Object detection and semantic segmentation are all important tasks in computer vision. Recent researches suggest object detection can be improved by learning with semantic segmentation.

4.5.1 Why Segmentation Improves Detection?

There are three reasons why the semantic segmentation improves object detection.

- Segmentation helps category recognition

Edges and boundaries are the basic elements that constitute human visual cognition [267, 268]. In computer vision, the difference between an object (e.g., a car, a person) and a stuff (e.g., sky, water, grass) is that the former usually has a closed and well defined boundary while the latter does not. As the feature of semantic segmentation tasks well captures the boundary of an object, segmentation may be helpful for category recognition.

- Segmentation helps accurate localization

The ground-truth bounding box of an object is determined by its well-defined boundary. For some objects with a special shape (e.g., imagine a cat with a very long tail), it will be difficult to predict high IoU locations. As object boundaries can be well encoded in semantic segmentation features, learning with segmentation would be helpful for accurate object localization.

- Segmentation can be embedded as context

Objects in daily life are surrounded by different backgrounds, such as the sky, water, grass, etc, and all these elements constitute the context of an object. Integrating the context of semantic segmentation will be helpful for object detection, say, an aircraft is more likely to appear in the sky than on the water.

4.5.2 How Segmentation Improves Detection?

There are two main approaches to improve object detection by segmentation: 1) learning with enriched features and 2) learning with multi-task loss functions.

- Learning with enriched features

The simplest way is to think of the segmentation network as a fixed feature extractor and to integrate it into a detection framework as additional features [144, 269, 270]. The advantage of this approach is that it is easy to implement, while the disadvantage is that the segmentation network may bring additional calculation.

- Learning with multi-task loss functions

Another way is to introduce an additional segmentation branch on top of the original detection framework and to train this model with multi-task loss functions (segmentation loss + detection loss) [4, 269]. In most cases, the segmentation brunch will be removed at the inference stage. The advantage is the detection speed will not be affected, but the disadvantage is that the training requires pixel-level image annotations. To this end, some researchers have followed the idea of "weakly supervised learning": instead of training based on pixel-wise annotation masks, they simply train the segmentation brunch based on the bounding-box level annotations [250, 271].

4.6 Robust Detection of Rotation and Scale Changes

Object rotation and scale changes are important challenges in object detection. As the features learned by CNN are not invariant to rotation and large degree of scale changes, in recent years, many people have made efforts in this problem.

4.6.1 Rotation Robust Detection

Object rotation is very common in detection tasks such as face detection, text detection, etc. The most straight forward solution to this problem is data augmentation so that an object in any orientation can be well covered by the augmented data [88]. Another solution is to train independent

detectors for every orientation [272, 273]. Apart from these traditional approaches, recently, there are some new improvement methods.

- Rotation invariant loss functions

The idea of learning with rotation invariant loss function can be traced back to the 1990s [274]. Some recent works have introduced a constraint on the original detection loss function so that to make the features of rotated objects unchanged [275, 276].

- Rotation calibration

Another way of improving rotation invariant detection is to make geometric transformations of the objects candidates [277–279]. This will be especially helpful for multi-stage detectors, where the correlation at early stages will benefit the subsequent detections. The representative of this idea is Spatial Transformer Networks (STN) [278]. STN has now been used in rotated text detection [278] and rotated face detection [279].

- Rotation RoI Pooling

In a two-stage detector, feature pooling aims to extract a fixed length feature representation for an object proposal with any location and size by first dividing the proposal evenly into a set of grids, and then concatenating the grid features. As the grid meshing is performed in Cartesian coordinates, the features are not invariance to rotation transform. A recent improvement is to mesh the grids in polar coordinates so that the features could be robust to the rotation changes [272].

4.6.2 Scale Robust Detection

Recent improvements have been made at both training and detection stages for scale robust detection.

- Scale adaptive training

Most of the modern detectors re-scale the input image to a fixed size and back propagate the loss of the objects in all scales, as shown in Fig. 19 (a). However, a drawback of doing this is there will be a "scale imbalance" problem. Building an image pyramid during detection could alleviate this problem but not fundamentally [46, 234]. A recent improvement is Scale Normalization for Image Pyramids (SNIP) [280], which builds image pyramids at both of training and detection stages and only backpropagates the loss of some selected scales, as shown in Fig. 19 (b). Some researchers have further proposed a more efficient training strategy: SNIP with Efficient Resampling (SNIPER) [281], i.e. to crop and re-scale an image to a set of sub-regions so that to benefit from large batch training.

- Scale adaptive detection

Most of the modern detectors use the fixed configurations for detecting objects of different sizes. For example, in a typical CNN based detector, we need to carefully define the size of anchors. A drawback of doing this is the configurations cannot be adaptive to unexpected scale changes. To improve the detection of small objects, some "adaptive zoom-in" techniques are proposed in some recent detectors to adaptively enlarge the small objects into the "larger ones" [184, 258]. Another recent improvement is learning to predict the scale distribution of objects in an image, and then adaptively re-scaling the image according to the distribution [282, 283].

4.7 Training from Scratch

Most deep learning based detectors are first pre-trained on large scale datasets, say ImageNet, and then fine-tuned on specific detection tasks. People have always believed that pre-training helps to improve generalization ability and training speed and the question is, do we really need to pre-training a detector on ImageNet? In fact, there are some limitations when adopting the pre-trained networks in object detection. The first limitation is the divergence between ImageNet classification and object detection, including their loss functions and scale/category distributions. The second limitation is the domain mismatch. As images in ImageNet are RGB images while detection sometimes will be applied to depth image (RGB-D) or 3D medical images, the pre-trained knowledge can not be well transfer to these detection tasks.

In recent years, some researchers have tried to train an object detector from scratch. To speed up training and improve stability, some researchers introduce dense connection and batch normalization to accelerate the back-propagation in shallow layers [238, 284]. The recent work by K. He *et al.* [285] has further questioned the paradigm of pre-training even further by exploring the opposite regime: they reported competitive results on object detection on the COCO dataset using standard models trained from random initialization, with the sole exception of increasing the number of training iterations so the randomly initialized models may converge. Training from random initialization is also surprisingly robust even using only 10% of the training data, which indicates that ImageNet pre-training may speed up convergence, but does not necessarily provide regularization or improve final detection accuracy.

4.8 Adversarial Training

The Generative Adversarial Networks (GAN) [286], introduced by A. Goodfellow *et al.* in 2014, has received great attention in recent years. A typical GAN consists of two neural networks: a generator networks and a discriminator networks, contesting with each other in a minimax optimization framework. Typically, the generator learns to map from a latent space to a particular data distribution of interest, while the discriminator aims to discriminate between instances from the true data distribution and those produced by the generator. GAN has been widely used for many computer vision tasks such as image generation[286, 287], image style transfer [288], and image super-resolution [289]. In recent two years, GAN has also been applied to object detection, especially for improving the detection of small and occluded object.

GAN has been used to enhance the detection on small objects by narrowing the representations between small and large ones [290, 291]. To improve the detection of occluded objects, one recent idea is to generate occlusion masks by using adversarial training [292]. Instead of generating

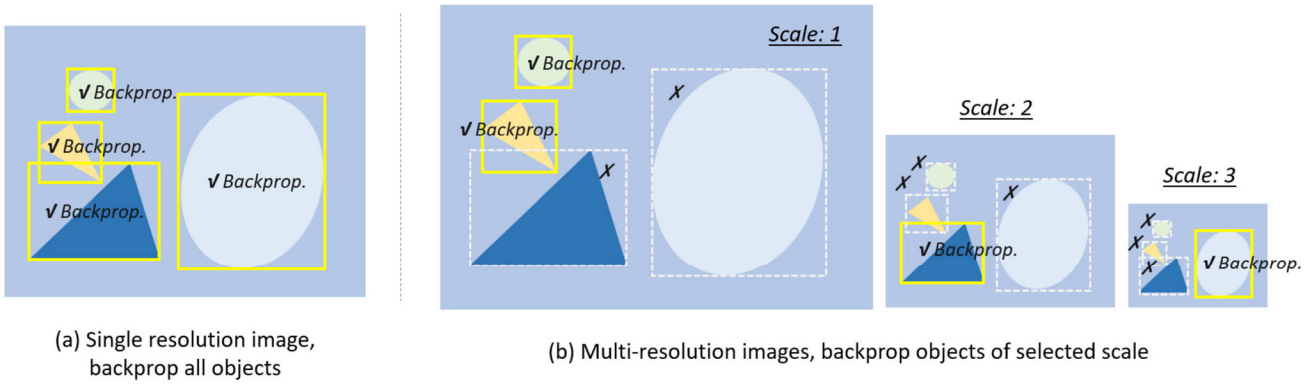


Fig. 19. Different training strategies for multi-scale object detection: (a): Training on a single resolution image, back propagate objects of all scales [17–19, 21]. (b) Training on multi-resolution images (image pyramid), back propagate objects of selected scale. If an object is too large or too small, its gradient will be discarded [56, 280, 281].

examples in pixel space, the adversarial network directly modifies the features to mimic occlusion.

In addition to these works, "adversarial attack" [293], which aims to study how to attack a detector with adversarial examples, has drawn increasing attention recently. The research on this topic is especially important for autonomous driving, as it cannot be fully trusted before guaranteeing the robustness to adversarial attacks.

4.9 Weakly Supervised Object Detection

The training of a modern object detector usually requires a large amount of manually labeled data, while the labeling process is time-consuming, expensive, and inefficient. Weakly Supervised Object Detection (WSOD) aims to solve this problem by training a detector with only image level annotations instead of bounding boxes.

Recently, multi-instance learning has been used for WSOD [294, 295]. Multi-instance learning is a group of supervised learning method [39, 296]. Instead of learning with a set of instances which are individually labeled, a multi-instance learning model receives a set of labeled bags, each containing many instances. If we consider object candidates in one image as a bag, and image-level annotation as the label, then the WSOD can be formulated as a multi-instance learning process.

Class activation mapping is another recently group of methods for WSOD [297, 298]. The research on CNN visualization has shown that the convolution layer of a CNN behaves as object detectors despite there is no supervision on the location of the object. Class activation mapping shed light on how to enable a CNN to have localization ability despite being trained on image level labels [299].

In addition to the above approaches, some other researchers considered the WSOD as a proposal ranking process by selecting the most informative regions and then training these regions with image-level annotation [300]. Another simple method for WSOD is to mask out different parts of the image. If the detection score drops sharply, then an object would be covered with high probability [301]. Besides, interactive annotation [295] takes human feedback into consideration during training so that to improve WSOD. More recently, generative adversarial training has been used for WSOD [302].

5 Applications

In this section, we will review some important detection applications in the past 20 years, including pedestrian detection, face detection, text detection, traffic sign/light detection, and remote sensing target detection.

5.1 Pedestrian Detection

Pedestrian detection, as an important object detection application, has received extensive attention in many areas such as autonomous driving, video surveillance, criminal investigation, etc. Some early time's pedestrian detection methods, such as HOG detector [12], ICF detector [171], laid a solid foundation for general object detection in terms of the feature representation [12, 171], the design of classifier [174], and the detection acceleration [177]. In recent years, some general object detection algorithms, e.g., Faster RCNN [19], have been introduced to pedestrian detection [165], and has greatly promoted the progress of this area.

5.1.1 Difficulties and Challenges

The challenges and difficulties in pedestrian detection can be summarized as follows.

Small pedestrian: Fig. 20 (a) shows some examples of the small pedestrians that are captured far from the camera. In Caltech Dataset [59, 60], 15% of the pedestrians are less than 30 pixels in height.

Hard negatives: Some backgrounds in street view images are very similar to pedestrians in their visual appearance, as shown in Fig. 20 (b).

Dense and occluded pedestrian: Fig 20 (c) shows some examples of dense and occluded pedestrians. In the Caltech Dataset [59, 60], pedestrians that haven't been occluded only account for 29% of the total pedestrian instances.

Real-time detection: The real-time pedestrian detection from HD video is crucial for some applications like autonomous driving and video surveillance.



Fig. 20. Some hard examples of pedestrian detection from Caltech dataset [59, 60]: (a) small pedestrians, (b) hard negatives, and (c) dense and occluded pedestrians.

5.1.2 Literature Review

Pedestrian detection has a very long research history [30, 31, 101]. Its development can be divided into two technical periods: 1) traditional pedestrian detection and 2) deep learning based pedestrian detection. We refer readers to the following surveys for more details on this topic [60, 303–307].

- Traditional pedestrian detection methods

Due to the limitations of computing resources, the Haar wavelet feature has been broadly used in early time's pedestrian detection [30, 31, 308]. To improve the detection of occluded pedestrians, one popular idea of that time was "detection by components" [31, 102, 220], i.e., to think of the detection as an ensemble of multiple part detectors that trained individually on different human parts, e.g. head, legs, and arms. As the increase of computing power, people started to design more complex detection models, and since 2005, gradient-based representation [12, 37, 177, 220, 309] and DPM [15, 37, 54] have become the mainstream of pedestrian detection. In 2009, by using the integral image acceleration, an effective and lightweight feature representation: the Integral Channel Features (ICF), was proposed [171]. ICF then became the new benchmark of pedestrian detection at that time [60]. In addition to the feature representation, some domain knowledge also has been considered, such as appearance constancy and shape symmetry [310] and stereo information [173, 311].

- Deep learning based pedestrian detection methods

Pedestrian detection is one of the first computer vision task that applies deep learning [312].

To improve small pedestrian detection: Although deep learning object detectors such as Fast/Faster R-CNN have shown state of the art performance for general object detection, they have limited success for detecting small pedestrians due to the low resolution of their convolutional features [165]. Some recent solutions to this problem include feature fusion [165], introducing extra high-resolution handcrafted features [313, 314], and ensembling detection results on multiple resolutions [315].

To improve hard negative detection: Some recent improvements include the integration of boosted decision tree [165], and semantics segmentation (as the context of the pedestrians) [316]. In addition, the idea of "cross-modal learning" has also been introduced to enrich the feature of hard negatives by using both RGB and infrared images [317].

To improve dense and occluded pedestrian detection: As we have mentioned in Section 2.3.2, the features in deeper layers of CNN have richer semantics but are not effective for detecting dense objects. To this end, some researchers have designed new loss function by considering the attraction of target and the repulsion of other surrounding objects [318]. Target occlusion is another problem that usually comes up with dense pedestrians. The ensemble of part detectors [319, 320] and the attention mechanism [321] are the most common ways to improve occluded pedestrian detection.

5.2 Face Detection

Face detection is one of the oldest computer vision applications [96, 164]. Early time's face detection, such as the VJ detector [10], has greatly promoted the object detection where many of its remarkable ideas are still playing important roles even in today's object detection. Face detection has now been applied in all walks of life, such as the "smile" detection in digital cameras, "face swiping" in e-commerce, facial makeup in mobile apps, etc.

5.2.1 Difficulties and Challenges

The difficulties and challenges in face detection can be summarized as follows:

Intra-class variation: Human faces may present a variety of expressions, skin colors, poses, and movements, as shown in Fig. 21 (a).

Occlusion: Faces may be partially occluded by other objects, as shown in Fig. 21 (b).

Multi-scale detection: Detecting faces in a large variety of scales, especially for some tiny faces, as shown in Fig. 21 (c).

Real-time detection: Face detection on mobile devices usually requires a CPU real-time detection speed.

5.2.2 Literature review

The research of face detection can be traced back to the early 1990s [95, 106, 108]. It then has gone through multiple historical periods: early time's face detection (before 2001), traditional face detection (2001-2015), and deep learning based face detection (2015-now). We refer readers to the following surveys for more details [323, 324].

- Early time's face detection (before 2001)

The early time's face detection algorithms can be divided into three groups: 1) Rule-based methods. This group of methods encode human knowledge of what constitutes a typical face and capture the relationships between facial elements [107, 108]. 2) Subspace analysis-based methods. This group of methods analyze the face distribution in underlying linear subspace [95, 106]. Eigenfaces is the representative of this group of methods [95]. 3) Learning based methods: To frame the face detection as a sliding window + binary classification (target vs background) process. Some commonly used models of this group include neural network [96, 164, 325] and SVM [29, 326].

- Traditional face detection (2000-2015)

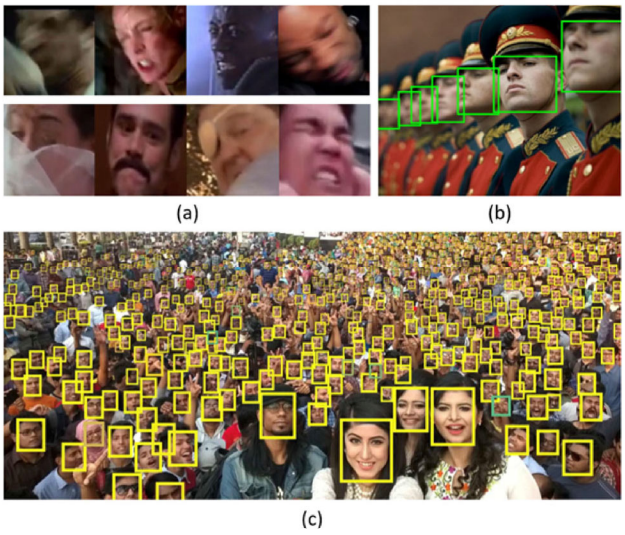


Fig. 21. Challenges in face detection: (a) Intra-class variation, image from WildestFaces Dataset [70]. (b) Face occlusion, image from UFDD Dataset [69]. (c) Multi-scale face detection. Image from *P. Hu et al. CVPR2017* [322].

There are two groups of face detectors in this period. The first group of methods are built based on boosted decision trees [10, 11, 109]. These methods are easy to compute, but usually suffer from low detection accuracy under complex scenes. The second group is based on early time's convolutional neural networks, where the shared computation of features are used to speed up detection [112, 113, 327].

- Deep learning based face detection (after 2015)

In deep learning era, most of the face detection algorithms follow the detection idea of the general object detectors such as Faster RCNN and SSD.

To speed up face detection: Cascaded detection (see more details in Section 3.3) is the most common way to speed up a face detector in deep learning era [179, 180]. Another speed up method is to predict the scale distribution of the faces in an image [283] and then run detection on some selected scales.

To improve multi-pose and occluded face detection: The idea of "face calibration" has been used to improve multi-pose face detection by estimating the calibration parameters [279] or using progressive calibration through multiple detection stages [277]. To improve occluded face detection, two methods have been proposed recently. The first one is to incorporate "attention mechanism" so that to highlight the features of underlying face targets [250]. The second one is "detection based on parts" [328], which inherits ideas from DPM.

To improve multi-scale face detection: Recent works on multi-scale face detection [322, 329–331] use similar detection strategies as those in general object detection, including multi-scale feature fusion and multi-resolution detection (see Section 2.3.2 and 4.2.2 for more details).

5.3 Text Detection

Text has long been the major information carrier of the human for thousands of years. The fundamental goal of text detection is to determine whether or not there is text in a given image, and if there is, to localize, and recognize it. Text detection has very broad applications. It helps people who are visually impaired to "read" street signs and currency [332, 333]. In geographic information systems, the detection and recognition of house numbers and street signs make it easier to build digital maps [334, 335].

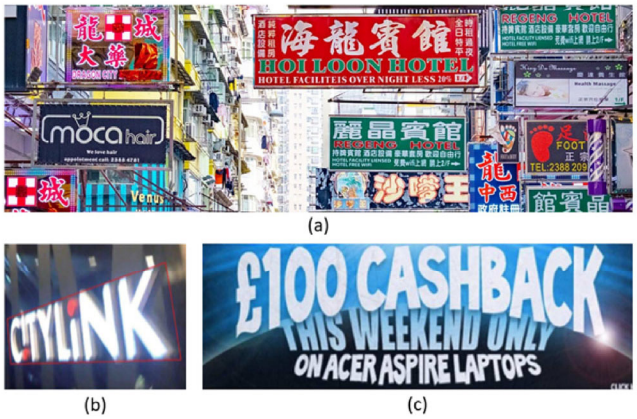


Fig. 22. Challenges in text detection and recognition: (a) Variation of fonts, colors and languages. Image from *maxpixel* (free of copyrights). (b) Text rotation and perspective distortion. Image from *Y. Liu* et al. *CVPR2017* [336]. (c) Densely arranged text localization. Image from *Y. Wu et al. ICCV2017* [337].

5.3.1 Difficulties and Challenges

The difficulties and challenges of text detection can be summarized as follows:

Different fonts and languages: Texts may have different fonts, colors, and languages, as shown in Fig. 22 (a).

Text rotation and perspective distortion: Texts may have different orientations and even may have perspective distortion, as shown in Fig. 22 (b).

Densely arranged text localization: Text lines with large aspect ratios and dense layout are difficult to localize accurately, as shown in Fig. 22 (c).

Broken and blurred characters: Broken and blurred characters are common in street view images.

5.3.2 Literature Review

Text detection consists of two related but relatively independent tasks: 1) text localization, and 2) text recognition. The existing text detection methods can be divided into two groups: "step-wise detection" and "integrated detection". We refer readers to the following survey for more details [338, 339].

- Step-wise detection vs integrated detection

Step-wise detection methods [340, 341] consist of a series of processing steps including character segmentation, candidate region verification, character grouping, and word recognition. The advantage of this group of methods is most of the background can be filtered in the coarse segmentation step, which greatly reduces the computational cost of the following process. The disadvantage is the parameters of all steps need to be set carefully, and the errors will occur and accumulate throughout each of these steps. By contrast,

integrated methods [342–345] frame the text detection as a joint probability inference problem, where the steps of character localization, grouping, and recognition are processed under a unified framework. The advantage of these methods is it avoids the cumulative error and is easy to integrate language models. The disadvantage is the inference will be computationally expensive when considering a large number of character classes and candidate windows [339].

- Traditional methods vs deep learning methods

Most of the traditional text detection methods generate text candidates in an unsupervised way, where the commonly used techniques include Maximally Stable Extremal Regions (MSER) segmentation [341] and morphological filtering [346]. Some domain knowledge, such as the symmetry of texts and the structures of strokes, also have been considered in these methods [340, 341, 347].

In recent years, researchers have paid more attention to the problem of text localization rather than recognition. Two groups of methods are proposed recently. The first group of methods frame the text detection as a special case of general object detection [251, 348–357]. These methods have a unified detection framework, but it is less effective for detecting texts with orientation or with large aspect ratio. The second group of methods frame the text detection as an image segmentation problem [336, 337, 358–360]. The advantage of these methods is there are no special restrictions for the shape and orientation of text, but the disadvantage is that it is not easy to distinguish densely arranged text lines from each other based on the segmentation result. The recent deep learning based text detection methods have proposed some solutions to the above problems.

For text rotation and perspective changes: The most common solution to this problem is to introduce additional parameters in anchor boxes and RoI pooling layer that are associated with rotation and perspective changes [351–353, 355–357].

To improve densely arranged text detection: The segmentation-based approach shows more advantages in detecting densely arranged texts. To distinguish the adjacent text lines, two groups of solutions have been proposed recently. The first one is "segment and linking", where "segment" refers to the character heatmap, and "linking" refers to the connection between two adjacent segments indicating that they belong to the same word or line of text [336, 358]. The second group is to introduce an additional corner/border detection task to help separate densely arrange texts, where a group of corners or a closed boundary corresponds to an individual line of text [337, 359, 360].

To improve broken and blurred text detection: A recent idea to deal with broken and blurred texts is to use word level [77, 361] recognition and sentence level recognition [335]. To deal with texts with different fonts, the most effective way is training with synthetic samples [77, 348].

5.4 Traffic Sign and Traffic Light Detection

With the development of self-driving technology, the automatic detection of traffic sign and traffic light has attracted great attention in recent years. Over the past decades, although the computer vision community has largely pushed towards the detection of general objects rather than fixed patterns like traffic lights and traffic signs, it would still be a mistake to believe that their recognition is not challenging.



Fig. 23. Challenges in traffic sign detection and traffic light detection: (a) Illumination changes. Image from *pxhere* (free of copyrights). (b) Motion blur. Image from GTSRB Dataset [81]. (c) Detection under bad weather. Image from *Flickr* and *Max Pixel* (free of copyrights).

5.4.1 *Difficulties and Challenges*

The challenges and difficulties of traffic sign/light detection can be summarized as follows:

Illumination changes: The detection will be particularly difficult when driving into the sun glare or at night, as shown in Fig. 23 (a).

Motion blur: The image captured by an on-board camera will become blurred due to the motion of the car, as shown in Fig. 23 (b).

Bad weather: In bad weathers, e.g., rainy and snowy days, the image quality will be affected, as shown in Fig. 23 (c).

Real-time detection: This is particularly important for autonomous driving.

5.4.2 *Literature Review*

Existing traffic sign/light detection methods can be divided into two groups: 1) traditional detection methods and 2) deep learning based detection methods. We refer readers to the following survey [80] for more details on this topic.

- Traditional detection methods

The research of vision based traffic sign/light detection can date back to as far as 20 years ago [362, 363]. As traffic sign/light has particular shape and color, the traditional detection methods are usually based on color thresholding [364–368], visual saliency detection[369], morphological filtering [79], and edge/contour analysis [370, 371]. As the above methods are merely designed based on low-level vision, they usually fail under complex environments (as is shown in Fig. 23), therefore, some researchers began to find other solutions beyond vision-based approaches, e.g., to combine GPS and digital maps in traffic light detection [372, 373]. Although "feature pyramid + sliding window"

has become a standard framework for general object detection and pedestrian detection at that time, apart from a very small number of works [374], the mainstream of traffic sign/light detection methods did not follow this paradigm until 2010 [375–377].

- Deep learning based detection methods

In deep learning era, some well-known detectors such as Faster RCNN and SSD were applied in traffic sign/light detection tasks [83, 84, 378, 379]. On basis on these detectors, some new techniques, such as the attention mechanism and adversarial training have been used to improve detection under complex traffic environments [290, 378].

5.5 Remote Sensing Target Detection

Remote sensing imaging technique has opened a door for people to better understand the earth. In recent years, as the resolution of remote sensing images has increased, remote sensing target detection (e.g., the detection of airplane, ship, oil-pot, etc), has become a research hot-spot. Remote sensing target detection has broad applications, such as military investigation, disaster rescue, and urban traffic management.

5.5.1 Difficulties and Challenges

The challenges and difficulties in remote sensing target detection are summarized as follows:

Detection in "big data": Due to the huge data volume of remote sensing images, how to quickly and accurately detect remote sensing targets remains a problem. Fig. 24 (a) shows a comparison on data volume between remote sensing images and natural images.

Occluded targets: Over 50% of the earth's surface is covered by cloud every day. Some examples of occluded targets are shown in Fig. 24 (b).

Domain adaptation: Remote sensing images captured by different sensors (e.g., with different modulates and resolutions) present a high degree of differences.

5.5.2 Literature Review

We refer readers to the following surveys for more details on this topic [90, 382].

- Traditional detection methods

Most of the traditional remote sensing target detection methods follow a two-stage detection paradigm: 1) candidate extraction and 2) target verification. In candidate extraction stage, some frequently used methods include gray value filtering based methods [383, 384], visual saliency-based methods [385–388], wavelet transform based methods [389], anomaly detection based methods [390], etc. One similarity of the above methods is they are all unsupervised methods, thus usually fail in complex environments. In target verification stage, some frequently used features include HOG [390, 391], LBP [384], SIFT [386, 388, 392], etc. Besides, there are also some other methods following the sliding window detection paradigm [391–394].

To detect targets with particular structure and shape such as oil-pots and inshore ships, some domain knowledge is used. For example, the oil-pot detection can be considered as circle/arc detection problem [395, 396]. The inshore ship detection can be considered as the detection of the foredeck and the stern [397, 398]. To improve the occluded target detection, one commonly used idea is "detection by parts" [380, 399]. To detect targets with different orientations, the "mixture model" is used by training different detectors for targets of different orientations [273].

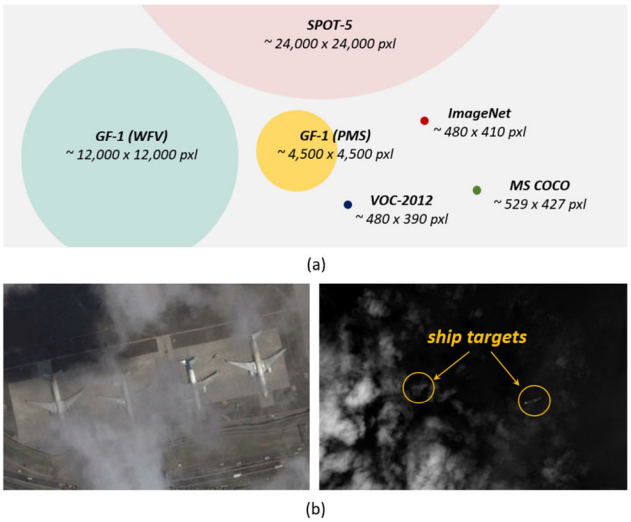


Fig. 24. Challenges in remote sensing target detection: (a) Detection in "big data": data volume comparison between a single-view remote sensing imagery and an average image size of VOC, ImageNet, and MS-COCO. (b) Targets occluded by cloud. Images from *S. Qiu et al. JSTARS2017* [380] and *Z. Zou et al. TGRS2016* [381].

- Deep learning based detection methods

After the great success of RCNN in 2014, deep CNN has been soon applied to remote sensing target detection [275, 276, 400, 401]. The general object detection framework like Faster RCNN and SSD have attracted increasing attention in remote sensing community [91, 167, 381, 402–405].

Due to the huge different between a remote sensing image and an everyday image, some investigations have been made on the effectiveness of deep CNN features for remote sensing images [406–408]. People discovered that in spite of its great success, the deep CNN is no better than traditional methods for spectral data [406]. To detect targets with different orientations, some researchers have improved the ROI Pooling layer for better rotation invariance [272, 409]. To improve domain adaptation, some researchers formulated the detection from a Bayesian view that at the detection stage, the model is adaptively updated based on the distribution of test images [91]. In addition, the attention mechanisms and feature fusion strategy also have been used to improve small target detection [410, 411].

6 Conclusion and Future Directions

Remarkable achievements have been made in object detection over the past 20 years. This paper not only extensively reviews some milestone detectors (e.g. VJ detector, HOG detector, DPM, Faster-RCNN, YOLO, SSD, etc), key technologies, speed up methods, detection applications, datasets, and metrics in its 20 years of history, but also discusses the

challenges currently met by the community, and how these detectors can be further extended and improved.

The future research of object detection may focus but is not limited to the following aspects:

Lightweight object detection: To speed up the detection algorithm so that it can run smoothly on mobile devices. Some important applications include mobile augmented reality, smart cameras, face verification, etc. Although a great effort has been made in recent years, the speed gap between a machine and human eyes still remains large, especially for detecting some small objects.

Detection meets AutoML: Recent deep learning based detectors are becoming more and more sophisticated and heavily relies on experiences. A future direction is to reduce human intervention when designing the detection model (e.g., how to design the engine and how to set anchor boxes) by using neural architecture search. AutoML could be the future of object detection.

Detection meets domain adaptation: The training process of any target detector can be essentially considered as a likelihood estimation process under the assumption of independent and identically distributed (i.i.d.) data. Object detection with non-i.i.d. data, especially for some real-world applications, still remains a challenge. GAN has shown promising results in domain adaptation and may be of great help to object detection in the future.

Weakly supervised detection: The training of a deep learning based detector usually relies on a large amount of well-annotated images. The annotation process is time-consuming, expensive, and inefficient. Developing weakly supervised detection techniques where the detectors are only trained with image-level annotations, or partially with bounding box annotations is of great importance for reducing labor costs and improving detection flexibility.

Small object detection: Detecting small objects in large scenes has long been a challenge. Some potential application of this research direction includes counting the population of wild animals with remote sensing images and detecting the state of some important military targets. Some further directions may include the integration of the visual attention mechanisms and the design of high resolution lightweight networks.

Detection in videos: Real-time object detection/tracking in HD videos is of great importance for video surveillance and autonomous driving. Traditional object detectors are usually designed under for image-wise detection, while simply ignores the correlations between videos frames. Improving detection by exploring the spatial and temporal correlation is an important research direction.

Detection with information fusion: Object detection with multiple sources/modalities of data, e.g., RGB-D image, 3d point cloud, LIDAR, etc, is of great importance for autonomous driving and drone applications. Some open questions include: how to immigrate well-trained detectors to different modalities of data, how to make information fusion to improve detection, etc.

Standing on the highway of technical evolutions, we believe this paper will help readers to build a big picture of object detection and to find future directions of this fast-moving research field.

REFERENCES

[1] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *European Conference on Computer Vision*. Springer, 2014, pp. 297–312.

[2] ——, "Hypercolumns for object segmentation and fine-grained localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 447–456.

[3] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3150–3158.

[4] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2980–2988.

[5] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3128–3137.

[6] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.

[7] Q. Wu, C. Shen, P. Wang, A. Dick, and A. van den Hengel, "Image captioning and visual question answering based on attributes and external knowledge," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 6, pp. 1367–1381, 2018.

[8] K. Kang, H. Li, J. Yan, X. Zeng, B. Yang, T. Xiao, C. Zhang, Z. Wang, R. Wang, X. Wang *et al.*, "T-cnn: Tubelets with convolutional neural networks for object detection from videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 10, pp. 2896–2907, 2018.

[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[10] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1. IEEE, 2001, pp. I–I.

[11] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[13] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[14] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*. IEEE, 2010, pp. 2241–2248.

[15] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual

recognition," in *European conference on computer vision*. Springer, 2014, pp. 346–361.
[18] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.
[19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.
[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.
[21] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.
[22] T.-Y. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection." in *CVPR*, vol. 1, no. 2, 2017, p. 4.
[23] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE transactions on pattern analysis and machine intelligence*, 2018.
[24] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *arXiv preprint arXiv:1809.02165*, 2018.
[25] S. Agarwal, J. O. D. Terrail, and F. Jurie, "Recent advances in object detection in the age of deep convolutional neural networks," *arXiv preprint arXiv:1809.03193*, 2018.
[26] A. Andreopoulos and J. K. Tsotsos, "50 years of object recognition: Directions forward," *Computer vision and image understanding*, vol. 117, no. 8, pp. 827–891, 2013.
[27] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama *et al.*, "Speed/accuracy trade-offs for modern convolutional object detectors," in *IEEE CVPR*, vol. 4, 2017.
[28] K. Grauman and B. Leibe, "Visual object recognition (synthesis lectures on artificial intelligence and machine learning)," *Morgan & Claypool*, 2011.
[29] C. P. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *Computer vision, 1998. sixth international conference on*. IEEE, 1998, pp. 555–562.
[30] C. Papageorgiou and T. Poggio, "A trainable system for object detection," *International journal of computer vision*, vol. 38, no. 1, pp. 15–33, 2000.
[31] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-based object detection in images by components," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 4, pp. 349–361, 2001.
[32] Y. Freund, R. Schapire, and N. Abe, "A short introduction to boosting," *Journal-Japanese Society For Artificial Intelligence*, vol. 14, no. 771-780, p. 1612, 1999.
[33] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157.
[34] ——, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
[35] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," CALIFORNIA UNIV SAN DIEGO LA JOLLA DEPT OF COMPUTER SCIENCE AND ENGINEERING, Tech. Rep., 2002.
[36] T. Malisiewicz, A. Gupta, and A. A. Efros, "Ensemble of exemplar-svms for object detection and beyond," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 89–96.
[37] R. B. Girshick, P. F. Felzenszwalb, and D. A. Mcallester, "Object detection with grammar models," in *Advances in Neural Information Processing Systems*, 2011, pp. 442–450.
[38] R. B. Girshick, *From rigid templates to grammars: Object detection with structured models*. Citeseer, 2012.
[39] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Advances in neural information processing systems*, 2003, pp. 577–584.
[40] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
[41] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 142–158, 2016.
[42] K. E. Van de Sande, J. R. Uijlings, T. Gevers, and A. W. Smeulders, "Segmentation as selective search for object recognition," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1879–1886.
[43] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester, "Discriminatively trained deformable part models, release 5," http://people.cs.uchicago.edu/ rbg/latent-release5/.
[44] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 6, pp. 1137–1149, 2017.
[45] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European conference on computer vision*. Springer, 2014, pp. 818–833.
[46] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *Advances in neural information processing systems*, 2016, pp. 379–387.
[47] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, and J. Sun, "Light-head r-cnn: In defense of two-stage object detector," *arXiv preprint arXiv:1711.07264*, 2017.
[48] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," *arXiv preprint*, 2017.
[49] ——, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.
[50] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.
[51] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International journal of computer vision*, vol. 111, no. 1, pp. 98–136, 2015.
[52] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.
[53] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.
[54] M. A. Sadeghi and D. Forsyth, "30hz object detection with dpm v5," in *European Conference on Computer Vision*. Springer, 2014, pp. 65–79.
[55] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Single-shot refinement neural network for object detection," in *IEEE CVPR*, 2018.
[56] Y. Li, Y. Chen, N. Wang, and Z. Zhang, "Scale-aware trident networks for object detection," *arXiv preprint arXiv:1901.01892*, 2019.
[57] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. Ieee, 2009, pp. 248–255.
[58] I. Krasin and T. e. a. Duerig, "Openimages: A

public dataset for large-scale multi-label and multi-class image classification." *Dataset available from https://storage.googleapis.com/openimages/web/index.html*, 2017.

[59] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 304–311.

[60] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 4, pp. 743–761, 2012.

[61] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3354–3361.

[62] S. Zhang, R. Benenson, and B. Schiele, "Citypersons: A diverse dataset for pedestrian detection," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, no. 2, 2017, p. 3.

[63] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[64] M. Braun, S. Krebs, F. Flohr, and D. M. Gavrila, "The eurocity persons dataset: A novel benchmark for object detection," *arXiv preprint arXiv:1805.07193*, 2018.

[65] V. Jain and E. Learned-Miller, "Fddb: A benchmark for face detection in unconstrained settings," Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, Tech. Rep., 2010.

[66] M. Koestinger, P. Wohlhart, P. M. Roth, and H. Bischof, "Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2144–2151.

[67] B. F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, and A. K. Jain, "Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1931–1939.

[68] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "Wider face: A face detection benchmark," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5525–5533.

[69] H. Nada, V. A. Sindagi, H. Zhang, and V. M. Patel, "Pushing the limits of unconstrained face detection: a challenge dataset and baseline results," *arXiv preprint arXiv:1804.10275*, 2018.

[70] M. K. Yucel, Y. C. Bilge, O. Oguz, N. Ikizler-Cinbis, P. Duygulu, and R. G. Cinbis, "Wildest faces: Face detection and recognition in violent settings," *arXiv preprint arXiv:1805.07566*, 2018.

[71] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, "Icdar 2003 robust reading competitions," in *null*. IEEE, 2003, p. 682.

[72] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu *et al.*, "Icdar 2015 competition on robust reading," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 2015, pp. 1156–1160.

[73] B. Shi, C. Yao, M. Liao, M. Yang, P. Xu, L. Cui, S. Belongie, S. Lu, and X. Bai, "Icdar2017 competition on reading chinese text in the wild (rctw-17)," in *Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on*, vol. 1. IEEE, 2017, pp. 1429–1434.

[74] K. Wang and S. Belongie, "Word spotting in the wild," in *European Conference on Computer Vision*. Springer, 2010, pp. 591–604.

[75] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1083–1090.

[76] A. Mishra, K. Alahari, and C. Jawahar, "Scene text recognition using higher order language priors," in *BMVC-British Machine Vision Conference*. BMVA, 2012.

[77] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," *arXiv preprint arXiv:1406.2227*, 2014.

[78] A. Veit, T. Matera, L. Neumann, J. Matas, and S. Belongie, "Coco-text: Dataset and benchmark for text detection and recognition in natural images," *arXiv preprint arXiv:1601.07140*, 2016.

[79] R. De Charette and F. Nashashibi, "Real time visual traffic lights recognition based on spot light detection and adaptive traffic lights templates," in *Intelligent Vehicles Symposium, 2009 IEEE*. IEEE, 2009, pp. 358–363.

[80] A. Møgelmose, M. M. Trivedi, and T. B. Moeslund, "Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey." *IEEE Trans. Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1484–1497, 2012.

[81] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The german traffic sign detection benchmark," in *Neural Networks (IJCNN), The 2013 International Joint Conference on*. IEEE, 2013, pp. 1–8.

[82] R. Timofte, K. Zimmermann, and L. Van Gool, "Multi-view traffic sign detection, recognition, and 3d localisation," *Machine vision and applications*, vol. 25, no. 3, pp. 633–647, 2014.

[83] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2110–2118.

[84] K. Behrendt, L. Novak, and R. Botros, "A deep learning approach to traffic lights: Detection, tracking, and classification," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1370–1377.

[85] G. Heitz and D. Koller, "Learning spatial context: Using stuff to find things," in *European conference on computer vision*. Springer, 2008, pp. 30–43.

[86] F. Tanner, B. Colder, C. Pullen, D. Heagy, M. Eppolito, V. Carlan, C. Oertel, and P. Sallee, "Overhead imagery research data setan annotated data library & tools to aid in the development of computer vision algorithms," in *2009 IEEE Applied Imagery Pattern Recognition Workshop (AIPR 2009)*. IEEE, 2009, pp. 1–8.

[87] K. Liu and G. Mattyus, "Fast multiclass vehicle detection on aerial images." *IEEE Geosci. Remote Sensing Lett.*, vol. 12, no. 9, pp. 1938–1942, 2015.

[88] H. Zhu, X. Chen, W. Dai, K. Fu, Q. Ye, and J. Jiao, "Orientation robust object detection in aerial images using deep convolutional neural network," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 3735–3739.

[89] S. Razakarivony and F. Jurie, "Vehicle detection in aerial imagery: A small target detection benchmark," *Journal of Visual Communication and Image Representation*, vol. 34, pp. 187–203, 2016.

[90] G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 117, pp. 11–28, 2016.

[91] Z. Zou and Z. Shi, "Random access memories: A new paradigm for target detection in high resolution aerial remote sensing images," *IEEE Transactions on Image Pro-*

cessing, vol. 27, no. 3, pp. 1100–1111, 2018.

[92] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "Dota: A large-scale dataset for object detection in aerial images," in *Proc. CVPR*, 2018.

[93] D. Lam, R. Kuzma, K. McGee, S. Dooley, M. Laielli, M. Klaric, Y. Bulatov, and B. McCord, "xview: Objects in context in overhead imagery," *arXiv preprint arXiv:1802.07856*, 2018.

[94] K. Oksuz, B. C. Cam, E. Akbas, and S. Kalkan, "Localization recall precision (lrp): A new performance metric for object detection," in *European Conference on Computer Vision (ECCV)*, vol. 6, 2018.

[95] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of cognitive neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[96] R. Vaillant, C. Monrocq, and Y. Le Cun, "Original approach for the localisation of objects in images," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 141, no. 4, pp. 245–250, 1994.

[97] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[98] I. Biederman, "Recognition-by-components: a theory of human image understanding." *Psychological review*, vol. 94, no. 2, p. 115, 1987.

[99] M. A. Fischler and R. A. Elschlager, "The representation and matching of pictorial structures," *IEEE Transactions on computers*, vol. 100, no. 1, pp. 67–92, 1973.

[100] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *International journal of computer vision*, vol. 77, no. 1-3, pp. 259–289, 2008.

[101] D. M. Gavrila and V. Philomin, "Real-time object detection for" smart" vehicles," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 1. IEEE, 1999, pp. 87–93.

[102] B. Wu and R. Nevatia, "Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors," in *null*. IEEE, 2005, pp. 90–97.

[103] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229*, 2013.

[104] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Advances in neural information processing systems*, 2013, pp. 2553–2561.

[105] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *European Conference on Computer Vision*. Springer, 2016, pp. 354–370.

[106] A. Pentland, B. Moghaddam, T. Starner *et al.*, "View-based and modular eigenspaces for face recognition," 1994.

[107] G. Yang and T. S. Huang, "Human face detection in a complex background," *Pattern recognition*, vol. 27, no. 1, pp. 53–63, 1994.

[108] I. Craw, D. Tock, and A. Bennett, "Finding face features," in *European Conference on Computer Vision*. Springer, 1992, pp. 92–96.

[109] R. Xiao, L. Zhu, and H.-J. Zhang, "Boosting chain learning for object detection," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 709–715.

[110] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[111] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *arXiv preprint arXiv:1412.7062*, 2014.

[112] C. Garcia and M. Delakis, "A neural architecture for fast and robust face detection," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 2. IEEE, 2002, pp. 44–47.

[113] M. Osadchy, M. L. Miller, and Y. L. Cun, "Synergistic face detection and pose estimation with energy-based models," in *Advances in Neural Information Processing Systems*, 2005, pp. 1017–1024.

[114] S. J. Nowlan and J. C. Platt, "A convolutional neural network hand tracker," *Advances in neural information processing systems*, pp. 901–908, 1995.

[115] T. Malisiewicz, *Exemplar-based representations for object detection, association and beyond*. Carnegie Mellon University, 2011.

[116] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?" in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 73–80.

[117] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.

[118] J. Carreira and C. Sminchisescu, "Constrained parametric min-cuts for automatic object segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 3241–3248.

[119] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 328–335.

[120] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2189–2202, 2012.

[121] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "Bing: Binarized normed gradients for objectness estimation at 300fps," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 3286–3293.

[122] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *European conference on computer vision*. Springer, 2014, pp. 391–405.

[123] C. Szegedy, S. Reed, D. Erhan, D. Anguelov, and S. Ioffe, "Scalable, high-quality object detection," *arXiv preprint arXiv:1412.1441*, 2014.

[124] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2147–2154.

[125] A. Ghodrati, A. Diba, M. Pedersoli, T. Tuytelaars, and L. Van Gool, "Deepproposal: Hunting objects by cascading deep convolutional layers," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2578–2586.

[126] W. Kuo, B. Hariharan, and J. Malik, "Deepbox: Learning objectness with convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2479–2487.

[127] S. Gidaris and N. Komodakis, "Attend refine repeat: Active box proposal generation via in-out localization," *arXiv preprint arXiv:1606.04446*, 2016.

[128] H. Li, Y. Liu, W. Ouyang, and X. Wang, "Zoom out-and-in network with recursive training for object proposal," *arXiv preprint arXiv:1702.05711*, 2017.

[129] J. Hosang, R. Benenson, P. Dollár, and B. Schiele, "What makes for effective detection proposals?" *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 4, pp. 814–830, 2016.

[130] J. Hosang, R. Benenson, and B. Schiele, "How

good are detection proposals, really?" *arXiv preprint arXiv:1406.6962*, 2014.
[131] J. Carreira and C. Sminchisescu, "Cpmc: Automatic object segmentation using constrained parametric min-cuts," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 7, pp. 1312–1328, 2011.
[132] N. Chavali, H. Agrawal, A. Mahendru, and D. Batra, "Object-proposal evaluation protocol is' gameable'," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 835–844.
[133] K. Lenc and A. Vedaldi, "R-cnn minus r," *arXiv preprint arXiv:1506.06981*, 2015.
[134] P.-A. Savalle, S. Tsogkas, G. Papandreou, and I. Kokkinos, "Deformable part models with cnn features," in *European Conference on Computer Vision, Parts and Attributes Workshop*, 2014.
[135] N. Zhang, J. Donahue, R. Girshick, and T. Darrell, "Part-based r-cnns for fine-grained category detection," in *European conference on computer vision*. Springer, 2014, pp. 834–849.
[136] L. Wan, D. Eigen, and R. Fergus, "End-to-end integration of a convolution network, deformable parts model and non-maximum suppression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 851–859.
[137] R. Girshick, F. Iandola, T. Darrell, and J. Malik, "Deformable part models are convolutional neural networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2015, pp. 437–446.
[138] B. Li, T. Wu, S. Shao, L. Zhang, and R. Chu, "Object detection via end-to-end integration of aspect ratio and context aware part-based models and fully convolutional networks," *arXiv preprint arXiv:1612.00534*, 2016.
[139] A. Torralba and P. Sinha, "Detecting faces in impoverished images," MASSACHUSETTS INST OF TECH CAMBRIDGE ARTIFICIAL INTELLIGENCE LAB, Tech. Rep., 2001.
[140] S. Zagoruyko, A. Lerer, T.-Y. Lin, P. O. Pinheiro, S. Gross, S. Chintala, and P. Dollár, "A multipath network for object detection," *arXiv preprint arXiv:1604.02135*, 2016.
[141] X. Zeng, W. Ouyang, B. Yang, J. Yan, and X. Wang, "Gated bi-directional cnn for object detection," in *European Conference on Computer Vision*. Springer, 2016, pp. 354–369.
[142] X. Zeng, W. Ouyang, J. Yan, H. Li, T. Xiao, K. Wang, Y. Liu, Y. Zhou, B. Yang, Z. Wang *et al.*, "Crafting gbd-net for object detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 9, pp. 2109–2123, 2018.
[143] W. Ouyang, K. Wang, X. Zhu, and X. Wang, "Learning chained deep features and classifiers for cascade in object detection," *arXiv preprint arXiv:1702.07054*, 2017.
[144] S. Gidaris and N. Komodakis, "Object detection via a multi-region and semantic segmentation-aware cnn model," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1134–1142.
[145] Y. Zhu, C. Zhao, J. Wang, X. Zhao, Y. Wu, H. Lu *et al.*, "Couplenet: Coupling global structure with local parts for object detection," in *Proc. of Intl Conf. on Computer Vision (ICCV)*, vol. 2, 2017.
[146] C. Desai, D. Ramanan, and C. C. Fowlkes, "Discriminative models for multi-class object layout," *International journal of computer vision*, vol. 95, no. 1, pp. 1–12, 2011.
[147] Z. Li, Y. Chen, G. Yu, and Y. Deng, "R-fcn++: Towards accurate region-based fully convolutional networks for object detection." in *AAAI*, 2018.
[148] S. Bell, C. Lawrence Zitnick, K. Bala, and R. Girshick, "Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2874–2883.
[149] J. Li, Y. Wei, X. Liang, J. Dong, T. Xu, J. Feng, and S. Yan, "Attentive contexts for object detection," *IEEE Transactions on Multimedia*, vol. 19, no. 5, pp. 944–954, 2017.
[150] Q. Chen, Z. Song, J. Dong, Z. Huang, Y. Hua, and S. Yan, "Contextualizing object detection and classification," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 1, pp. 13–27, 2015.
[151] S. Gupta, B. Hariharan, and J. Malik, "Exploring person context and local scene context for object detection," *arXiv preprint arXiv:1511.08177*, 2015.
[152] X. Chen and A. Gupta, "Spatial memory for context reasoning in object detection," *arXiv preprint arXiv:1704.04224*, 2017.
[153] Y. Liu, R. Wang, S. Shan, and X. Chen, "Structure inference net: Object detection using scene-level context and instance-level relationships," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6985–6994.
[154] J. H. Hosang, R. Benenson, and B. Schiele, "Learning non-maximum suppression." in *CVPR*, 2017, pp. 6469–6477.
[155] P. Henderson and V. Ferrari, "End-to-end training of object class detectors for mean average precision," in *Asian Conference on Computer Vision*. Springer, 2016, pp. 198–213.
[156] R. Rothe, M. Guillaumin, and L. Van Gool, "Non-maximum suppression for object detection by passing messages between windows," in *Asian Conference on Computer Vision*. Springer, 2014, pp. 290–306.
[157] D. Mrowca, M. Rohrbach, J. Hoffman, R. Hu, K. Saenko, and T. Darrell, "Spatial semantic regularisation for large scale object detection," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2003–2011.
[158] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-nmsimproving object detection with one line of code," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5562–5570.
[159] L. Tychsen-Smith and L. Petersson, "Improving object localization with fitness nms and bounded iou loss," *arXiv preprint arXiv:1711.00164*, 2017.
[160] S. K. Divvala, D. Hoiem, J. H. Hays, A. A. Efros, and M. Hebert, "An empirical study of context in object detection," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1271–1278.
[161] C. Chen, M.-Y. Liu, O. Tuzel, and J. Xiao, "R-cnn for small object detection," in *Asian conference on computer vision*. Springer, 2016, pp. 214–230.
[162] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei, "Relation networks for object detection," in *Computer Vision and Pattern Recognition (CVPR)*, vol. 2, no. 3, 2018.
[163] B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang, "Acquisition of localization confidence for accurate object detection," in *Proceedings of the European Conference on Computer Vision, Munich, Germany*, 2018, pp. 8–14.
[164] H. A. Rowley, S. Baluja, and T. Kanade, "Human face detection in visual scenes," in *Advances in Neural Information Processing Systems*, 1996, pp. 875–881.
[165] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster r-cnn doing well for pedestrian detection?" in *European Conference on Computer Vision*. Springer, 2016, pp. 443–457.
[166] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 761–769.
[167] T. Tang, S. Zhou, Z. Deng, H. Zou, and L. Lei, "Vehicle detection in aerial images based on region convolutional neural networks and hard negative example mining," *Sensors*, vol. 17, no. 2, p. 336, 2017.
[168] X. Sun, P. Wu, and S. C. Hoi, "Face detection using deep

learning: An improved faster rcnn approach," *Neurocomputing*, vol. 299, pp. 42–50, 2018.
[169] J. Jin, K. Fu, and C. Zhang, "Traffic sign recognition with hinge loss trained convolutional neural networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 1991–2000, 2014.
[170] M. Zhou, M. Jing, D. Liu, Z. Xia, Z. Zou, and Z. Shi, "Multi-resolution networks for ship detection in infrared remote sensing images," *Infrared Physics & Technology*, 2018.
[171] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," 2009.
[172] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532–1545, 2014.
[173] R. Benenson, M. Mathias, R. Timofte, and L. Van Gool, "Pedestrian detection at 100 frames per second," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2903–2910.
[174] S. Maji, A. C. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.
[175] A. Vedaldi and A. Zisserman, "Sparse kernel approximations for efficient classification and detection," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2320–2327.
[176] F. Fleuret and D. Geman, "Coarse-to-fine face detection," *International Journal of computer vision*, vol. 41, no. 1-2, pp. 85–107, 2001.
[177] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2. IEEE, 2006, pp. 1491–1498.
[178] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman, "Multiple kernels for object detection," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 606–613.
[179] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5325–5334.
[180] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.
[181] Z. Cai, M. Saberian, and N. Vasconcelos, "Learning complexity-aware cascades for deep pedestrian detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3361–3369.
[182] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Craft objects from images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 6043–6051.
[183] F. Yang, W. Choi, and Y. Lin, "Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2129–2137.
[184] M. Gao, R. Yu, A. Li, V. I. Morariu, and L. S. Davis, "Dynamic zoom-in network for fast object detection in large images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
[185] W. Ouyang, K. Wang, X. Zhu, and X. Wang, "Chained cascade network for object detection," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1956–1964.
[186] Y. LeCun, J. S. Denker, and S. A. Solla, "Optimal brain damage," in *Advances in neural information processing systems*, 1990, pp. 598–605.
[187] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.
[188] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf, "Pruning filters for efficient convnets," *arXiv preprint arXiv:1608.08710*, 2016.
[189] G. Huang, S. Liu, L. van der Maaten, and K. Q. Weinberger, "Condensenet: An efficient densenet using learned group convolutions," *group*, vol. 3, no. 12, p. 11, 2017.
[190] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "Xnor-net: Imagenet classification using binary convolutional neural networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 525–542.
[191] X. Lin, C. Zhao, and W. Pan, "Towards accurate binary convolutional neural network," in *Advances in Neural Information Processing Systems*, 2017, pp. 345–353.
[192] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks," in *Advances in neural information processing systems*, 2016, pp. 4107–4115.
[193] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.
[194] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," *arXiv preprint arXiv:1412.6550*, 2014.
[195] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker, "Learning efficient object detection models with knowledge distillation," in *Advances in Neural Information Processing Systems*, 2017, pp. 742–751.
[196] Q. Li, S. Jin, and J. Yan, "Mimicking very efficient network for object detection," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 7341–7349.
[197] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5353–5360.
[198] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.
[199] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, "Large kernel mattersimprove semantic segmentation by global convolutional network," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 1743–1751.
[200] K.-H. Kim, S. Hong, B. Roh, Y. Cheon, and M. Park, "Pvanet: deep but lightweight neural networks for real-time object detection," *arXiv preprint arXiv:1608.08021*, 2016.
[201] X. Zhang, J. Zou, X. Ming, K. He, and J. Sun, "Efficient and accurate approximations of nonlinear convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1984–1992.
[202] X. Zhang, J. Zou, K. He, and J. Sun, "Accelerating very deep convolutional networks for classification and detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 1943–1955, 2016.
[203] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," 2017.
[204] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *arXiv preprint*, pp. 1610–02 357, 2017.
[205] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mo-

bile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
[206] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 2018, pp. 4510–4520.
[207] Y. Li, J. Li, W. Lin, and J. Li, "Tiny-dsod: Lightweight object detection for resource-restricted usages," *arXiv preprint arXiv:1807.11013*, 2018.
[208] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.
[209] R. J. Wang, X. Li, S. Ao, and C. X. Ling, "Pelee: A real-time object detection system on mobile devices," *arXiv preprint arXiv:1804.06882*, 2018.
[210] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016.
[211] B. Wu, F. N. Iandola, P. H. Jin, and K. Keutzer, "Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving." in *CVPR Workshops*, 2017, pp. 446–454.
[212] T. Kong, A. Yao, Y. Chen, and F. Sun, "Hypernet: Towards accurate region proposal generation and joint object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 845–853.
[213] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8697–8710.
[214] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," *arXiv preprint arXiv:1611.01578*, 2016.
[215] Y. Chen, T. Yang, X. Zhang, G. Meng, C. Pan, and J. Sun, "Detnas: Neural architecture search on object detection," *arXiv preprint arXiv:1903.10979*, 2019.
[216] C. Liu, L.-C. Chen, F. Schroff, H. Adam, W. Hua, A. Yuille, and L. Fei-Fei, "Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation," *arXiv preprint arXiv:1901.02985*, 2019.
[217] X. Chu, B. Zhang, R. Xu, and H. Ma, "Multi-objective reinforced evolution in mobile neural architecture search," *arXiv preprint arXiv:1901.01074*, 2019.
[218] C.-H. Hsu, S.-H. Chang, D.-C. Juan, J.-Y. Pan, Y.-T. Chen, W. Wei, and S.-C. Chang, "Monas: Multi-objective neural architecture search using reinforcement learning," *arXiv preprint arXiv:1806.10332*, 2018.
[219] P. Simard, L. Bottou, P. Haffner, and Y. LeCun, "Boxlets: a fast convolution algorithm for signal processing and neural networks," in *Advances in Neural Information Processing Systems*, 1999, pp. 571–577.
[220] X. Wang, T. X. Han, and S. Yan, "An hog-lbp human detector with partial occlusion handling," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 32–39.
[221] F. Porikli, "Integral histogram: A fast way to extract histograms in cartesian spaces," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 829–836.
[222] M. Mathieu, M. Henaff, and Y. LeCun, "Fast training of convolutional networks through ffts," *arXiv preprint arXiv:1312.5851*, 2013.
[223] H. Pratt, B. Williams, F. Coenen, and Y. Zheng, "Fcnn: Fourier convolutional neural networks," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2017, pp. 786–798.
[224] N. Vasilache, J. Johnson, M. Mathieu, S. Chintala, S. Piantino, and Y. LeCun, "Fast convolutional nets with fbfft: A gpu performance evaluation," *arXiv preprint arXiv:1412.7580*, 2014.
[225] O. Rippel, J. Snoek, and R. P. Adams, "Spectral representations for convolutional neural networks," in *Advances in neural information processing systems*, 2015, pp. 2449–2457.
[226] C. Dubout and F. Fleuret, "Exact acceleration of linear object detectors," in *European Conference on Computer Vision*. Springer, 2012, pp. 301–311.
[227] M. A. Sadeghi and D. Forsyth, "Fast template evaluation with vector quantization," in *Advances in neural information processing systems*, 2013, pp. 2949–2957.
[228] I. Kokkinos, "Bounding part scores for rapid detection with deformable part models," in *European Conference on Computer Vision*. Springer, 2012, pp. 41–50.
[229] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, L. Wang, G. Wang *et al.*, "Recent advances in convolutional neural networks," *arXiv preprint arXiv:1512.07108*, 2015.
[230] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[231] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.
[232] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.
[233] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning." in *AAAI*, vol. 4, 2017, p. 12.
[234] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
[235] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks." in *CVPR*, vol. 1, no. 2, 2017, p. 3.
[236] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," *arXiv preprint arXiv:1709.01507*, vol. 7, 2017.
[237] P. Zhou, B. Ni, C. Geng, J. Hu, and Y. Xu, "Scale-transferrable object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 528–537.
[238] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "Dsod: Learning deeply supervised object detectors from scratch," in *The IEEE International Conference on Computer Vision (ICCV)*, vol. 3, no. 6, 2017, p. 7.
[239] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 5987–5995.
[240] J. Jeong, H. Park, and N. Kwak, "Enhancement of ssd by concatenating feature maps for object detection," *arXiv preprint arXiv:1705.09587*, 2017.
[241] K. Lee, J. Choi, J. Jeong, and N. Kwak, "Residual features and unified prediction network for single stage detection," *arXiv preprint arXiv:1707.05031*, 2017.
[242] G. Cao, X. Xie, W. Yang, Q. Liao, G. Shi, and J. Wu, "Feature-fused ssd: fast detection for small objects," in *Ninth International Conference on Graphic and Image Processing (ICGIP 2017)*, vol. 10615. International Society for Optics and Photonics, 2018, p. 106151E.
[243] L. Zheng, C. Fu, and Y. Zhao, "Extend the shallow part of single shot multibox detector via convolutional neural network," *arXiv preprint arXiv:1801.05918*, 2018.
[244] A. Shrivastava, R. Sukthankar, J. Malik, and A. Gupta, "Beyond skip connections: Top-down modulation for ob-

ject detection," *arXiv preprint arXiv:1612.06851*, 2016.
[245] T. Kong, F. Sun, A. Yao, H. Liu, M. Lu, and Y. Chen, "Ron: Reverse connection with objectness prior networks for object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2017, p. 2.
[246] S. Woo, S. Hwang, and I. S. Kweon, "Stairnet: Top-down semantic aggregation for accurate one shot detection," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 1093–1102.
[247] Y. Chen, J. Li, B. Zhou, J. Feng, and S. Yan, "Weaving multi-scale context for single shot detector," *arXiv preprint arXiv:1712.03149*, 2017.
[248] M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2018–2025.
[249] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "Dssd: Deconvolutional single shot detector," *arXiv preprint arXiv:1701.06659*, 2017.
[250] J. Wang, Y. Yuan, and G. Yu, "Face attention network: An effective face detector for the occluded faces," *arXiv preprint arXiv:1711.07246*, 2017.
[251] P. He, W. Huang, T. He, Q. Zhu, Y. Qiao, and X. Li, "Single shot text detector with regional attention," in *The IEEE International Conference on Computer Vision (ICCV)*, vol. 6, no. 7, 2017.
[252] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.
[253] F. Yu, V. Koltun, and T. A. Funkhouser, "Dilated residual networks." in *CVPR*, vol. 2, 2017, p. 3.
[254] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, and J. Sun, "Detnet: A backbone network for object detection," *arXiv preprint arXiv:1804.06215*, 2018.
[255] S. Liu, D. Huang, and Y. Wang, "Receptive field block net for accurate and fast object detection," *arXiv preprint arXiv:1711.07767*, 2017.
[256] M. Najibi, M. Rastegari, and L. S. Davis, "G-cnn: an iterative grid based object detector," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2369–2377.
[257] D. Yoo, S. Park, J.-Y. Lee, A. S. Paek, and I. So Kweon, "Attentionnet: Aggregating weak directions for accurate object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2659–2667.
[258] Y. Lu, T. Javidi, and S. Lazebnik, "Adaptive object detection using adjacency and zoom prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2351–2359.
[259] R. Ranjan, V. M. Patel, and R. Chellappa, "Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 1, pp. 121–135, 2019.
[260] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," *arXiv preprint arXiv:1611.08050*, 2016.
[261] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," in *Proceedings of the European Conference on Computer Vision (ECCV)*, vol. 6, 2018.
[262] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, no. 2, 2018, p. 10.
[263] R. N. Rajaram, E. Ohn-Bar, and M. M. Trivedi, "Refinenet: Iterative refinement for accurate object localization," in *Intelligent Transportation Systems (ITSC), 2016 IEEE 19th International Conference on*. IEEE, 2016, pp. 1528–1533.
[264] M.-C. Roh and J.-y. Lee, "Refining faster-rcnn for accurate object detection," in *Machine Vision Applications (MVA), 2017 Fifteenth IAPR International Conference on*. IEEE, 2017, pp. 514–517.
[265] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang, "Unitbox: An advanced object detection network," in *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016, pp. 516–520.
[266] S. Gidaris and N. Komodakis, "Locnet: Improving localization accuracy for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 789–798.
[267] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, p. 607, 1996.
[268] A. J. Bell and T. J. Sejnowski, "The independent components of natural scenes are edge filters," *Vision research*, vol. 37, no. 23, pp. 3327–3338, 1997.
[269] S. Brahmbhatt, H. I. Christensen, and J. Hays, "Stuffnet: Using stuffto improve object detection," in *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*. IEEE, 2017, pp. 934–943.
[270] A. Shrivastava and A. Gupta, "Contextual priming and feedback for faster r-cnn," in *European Conference on Computer Vision*. Springer, 2016, pp. 330–348.
[271] Z. Zhang, S. Qiao, C. Xie, W. Shen, B. Wang, and A. L. Yuille, "Single-shot object detection with enriched semantics," Center for Brains, Minds and Machines (CBMM), Tech. Rep., 2018.
[272] B. Cai, Z. Jiang, H. Zhang, Y. Yao, and S. Nie, "Online exemplar-based fully convolutional network for aircraft detection in remote sensing images," *IEEE Geoscience and Remote Sensing Letters*, no. 99, pp. 1–5, 2018.
[273] G. Cheng, J. Han, P. Zhou, and L. Guo, "Multi-class geospatial object detection and geographic image classification based on collection of part detectors," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 98, pp. 119–132, 2014.
[274] P. Y. Simard, Y. A. LeCun, J. S. Denker, and B. Victorri, "Transformation invariance in pattern recognitiontangent distance and tangent propagation," in *Neural networks: tricks of the trade*. Springer, 1998, pp. 239–274.
[275] G. Cheng, P. Zhou, and J. Han, "Rifd-cnn: Rotation-invariant and fisher discriminative convolutional neural networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2884–2893.
[276] ——, "Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 12, pp. 7405–7415, 2016.
[277] X. Shi, S. Shan, M. Kan, S. Wu, and X. Chen, "Real-time rotation-invariant face detection with progressive calibration networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2295–2303.
[278] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Advances in neural information processing systems*, 2015, pp. 2017–2025.
[279] D. Chen, G. Hua, F. Wen, and J. Sun, "Supervised transformer network for efficient face detection," in *European Conference on Computer Vision*. Springer, 2016, pp. 122–138.
[280] B. Singh and L. S. Davis, "An analysis of scale invariance in object detection–snip," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3578–3587.
[281] B. Singh, M. Najibi, and L. S. Davis, "Sniper: Efficient multi-scale training," *arXiv preprint arXiv:1805.09300*, 2018.
[282] S. Qiao, W. Shen, W. Qiu, C. Liu, and A. L. Yuille,

"Scalenet: Guiding object proposal generation in supermarkets and beyond." in *ICCV*, 2017, pp. 1809–1818.

[283] Z. Hao, Y. Liu, H. Qin, J. Yan, X. Li, and X. Hu, "Scale-aware face detection," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 3, 2017.

[284] R. Zhu, S. Zhang, X. Wang, L. Wen, H. Shi, L. Bo, and T. Mei, "Scratchdet: Exploring to train single-shot object detectors from scratch," *arXiv preprint arXiv:1810.08425*, 2018.

[285] K. He, R. Girshick, and P. Dollár, "Rethinking imagenet pre-training," *arXiv preprint arXiv:1811.08883*, 2018.

[286] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[287] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[288] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," *arXiv preprint*, 2017.

[289] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. P. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network." in *CVPR*, vol. 2, no. 3, 2017, p. 4.

[290] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *IEEE CVPR*, 2017.

[291] Y. Bai, Y. Zhang, M. Ding, and B. Ghanem, "Sod-mtgan: Small object detection via multi-task generative adversarial network," *Computer Vision-ECCV*, pp. 8–14, 2018.

[292] X. Wang, A. Shrivastava, and A. Gupta, "A-fast-rcnn: Hard positive generation via adversary for object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[293] S.-T. Chen, C. Cornelius, J. Martin, and D. H. Chau, "Robust physical adversarial attack on faster r-cnn object detector," *arXiv preprint arXiv:1804.05810*, 2018.

[294] R. G. Cinbis, J. Verbeek, and C. Schmid, "Weakly supervised object localization with multi-fold multiple instance learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 1, pp. 189–203, 2017.

[295] D. P. Papadopoulos, J. R. Uijlings, F. Keller, and V. Ferrari, "We don't need no bounding-boxes: Training object class detectors using only human verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 854–863.

[296] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial intelligence*, vol. 89, no. 1-2, pp. 31–71, 1997.

[297] Y. Zhu, Y. Zhou, Q. Ye, Q. Qiu, and J. Jiao, "Soft proposal networks for weakly supervised object localization," in *Proc. IEEE Int. Conf. Comput. Vis.(ICCV)*, 2017, pp. 1841–1850.

[298] A. Diba, V. Sharma, A. M. Pazandeh, H. Pirsiavash, and L. Van Gool, "Weakly supervised cascaded convolutional networks." in *CVPR*, vol. 1, no. 2, 2017, p. 8.

[299] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2921–2929.

[300] H. Bilen and A. Vedaldi, "Weakly supervised deep detection networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2846–2854.

[301] L. Bazzani, A. Bergamo, D. Anguelov, and L. Torresani, "Self-taught object localization with deep networks," in *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*. IEEE, 2016, pp. 1–9.

[302] Y. Shen, R. Ji, S. Zhang, W. Zuo, and Y. Wang, "Generative adversarial learning towards fast weakly supervised detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5764–5773.

[303] M. Enzweiler and D. M. Gavrila, "Monocular pedestrian detection: Survey and experiments," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 12, pp. 2179–2195, 2008.

[304] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 7, pp. 1239–1258, 2010.

[305] R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Ten years of pedestrian detection, what have we learned?" in *European Conference on Computer Vision*. Springer, 2014, pp. 613–627.

[306] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, "How far are we from solving pedestrian detection?" in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1259–1267.

[307] ——, "Towards reaching human performance in pedestrian detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 973–986, 2018.

[308] P. Viola, M. J. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," *International Journal of Computer Vision*, vol. 63, no. 2, pp. 153–161, 2005.

[309] P. Sabzmeydani and G. Mori, "Detecting pedestrians by learning shapelet features," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8.

[310] J. Cao, Y. Pang, and X. Li, "Pedestrian detection inspired by appearance constancy and shape symmetry," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1316–1324.

[311] R. Benenson, R. Timofte, and L. Van Gool, "Stixels estimation without depth map computation," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2010–2017.

[312] J. Hosang, M. Omran, R. Benenson, and B. Schiele, "Taking a deeper look at pedestrians," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4073–4082.

[313] J. Cao, Y. Pang, and X. Li, "Learning multilayer channel features for pedestrian detection," *IEEE transactions on image processing*, vol. 26, no. 7, pp. 3210–3220, 2017.

[314] J. Mao, T. Xiao, Y. Jiang, and Z. Cao, "What can help pedestrian detection?" in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 6034–6043.

[315] Q. Hu, P. Wang, C. Shen, A. van den Hengel, and F. Porikli, "Pushing the limits of deep cnns for pedestrian detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 6, pp. 1358–1368, 2018.

[316] Y. Tian, P. Luo, X. Wang, and X. Tang, "Pedestrian detection aided by deep learning semantic tasks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5079–5087.

[317] D. Xu, W. Ouyang, E. Ricci, X. Wang, and N. Sebe, "Learning cross-modal deep representations for robust pedestrian detection," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[318] X. Wang, T. Xiao, Y. Jiang, S. Shao, J. Sun, and C. Shen, "Repulsion loss: Detecting pedestrians in a crowd," *arXiv preprint arXiv:1711.07752*, 2017.

[319] Y. Tian, P. Luo, X. Wang, and X. Tang, "Deep learning strong parts for pedestrian detection," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1904–1912.

[320] W. Ouyang, H. Zhou, H. Li, Q. Li, J. Yan, and X. Wang, "Jointly learning deep features, deformable parts, occlusion and classification for pedestrian detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 8, pp. 1874–1887, 2018.

[321] S. Zhang, J. Yang, and B. Schiele, "Occluded pedestrian detection through guided attention in cnns," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6995–7003.

[322] P. Hu and D. Ramanan, "Finding tiny faces," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 1522–1530.

[323] M.-H. Yang, D. J. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 24, no. 1, pp. 34–58, 2002.

[324] S. Zafeiriou, C. Zhang, and Z. Zhang, "A survey on face detection in the wild: past, present and future," *Computer Vision and Image Understanding*, vol. 138, pp. 1–24, 2015.

[325] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 20, no. 1, pp. 23–38, 1998.

[326] E. Osuna, R. Freund, and F. Girosit, "Training support vector machines: an application to face detection," in *Computer vision and pattern recognition, 1997. Proceedings., 1997 IEEE computer society conference on*. IEEE, 1997, pp. 130–136.

[327] M. Osadchy, Y. L. Cun, and M. L. Miller, "Synergistic face detection and pose estimation with energy-based models," *Journal of Machine Learning Research*, vol. 8, no. May, pp. 1197–1215, 2007.

[328] S. Yang, P. Luo, C. C. Loy, and X. Tang, "Faceness-net: Face detection through deep facial part responses," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 8, pp. 1845–1859, 2018.

[329] S. Yang, Y. Xiong, C. C. Loy, and X. Tang, "Face detection through scale-friendly deep convolutional networks," *arXiv preprint arXiv:1706.02863*, 2017.

[330] M. Najibi, P. Samangouei, R. Chellappa, and L. S. Davis, "Ssh: Single stage headless face detector." in *ICCV*, 2017, pp. 4885–4894.

[331] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li, "S^ 3fd: Single shot scale-invariant face detector," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 192–201.

[332] X. Liu, "A camera phone based currency reader for the visually impaired," in *Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*. ACM, 2008, pp. 305–306.

[333] N. Ezaki, K. Kiyota, B. T. Minh, M. Bulacu, and L. Schomaker, "Improved text-detection methods for a camera-based text reading system for blind persons," in *Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on*. IEEE, 2005, pp. 257–261.

[334] P. Sermanet, S. Chintala, and Y. LeCun, "Convolutional neural networks applied to house numbers digit classification," in *Pattern Recognition (ICPR), 2012 21st International Conference on*. IEEE, 2012, pp. 3288–3291.

[335] Z. Wojna, A. Gorban, D.-S. Lee, K. Murphy, Q. Yu, Y. Li, and J. Ibarz, "Attention-based extraction of structured information from street view imagery," *arXiv preprint arXiv:1704.03549*, 2017.

[336] Y. Liu and L. Jin, "Deep matching prior network: Toward tighter multi-oriented text detection," in *Proc. CVPR*, 2017, pp. 3454–3461.

[337] Y. Wu and P. Natarajan, "Self-organized text detection with minimal post-processing via border learning," in *Proc. ICCV*, 2017.

[338] Y. Zhu, C. Yao, and X. Bai, "Scene text detection and recognition: Recent advances and future trends," *Frontiers of Computer Science*, vol. 10, no. 1, pp. 19–36, 2016.

[339] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 7, pp. 1480–1500, 2015.

[340] L. Neumann and J. Matas, "Scene text localization and recognition with oriented stroke detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 97–104.

[341] X.-C. Yin, X. Yin, K. Huang, and H.-W. Hao, "Robust text detection in natural scene images," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 5, pp. 970–983, 2014.

[342] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1457–1464.

[343] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "End-to-end text recognition with convolutional neural networks," in *Pattern Recognition (ICPR), 2012 21st International Conference on*. IEEE, 2012, pp. 3304–3308.

[344] S. Tian, Y. Pan, C. Huang, S. Lu, K. Yu, and C. Lim Tan, "Text flow: A unified text detection system in natural scene images," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4651–4659.

[345] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Deep features for text spotting," in *European conference on computer vision*. Springer, 2014, pp. 512–528.

[346] X.-C. Yin, W.-Y. Pei, J. Zhang, and H.-W. Hao, "Multi-orientation scene text detection with adaptive clustering," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 9, pp. 1930–1937, 2015.

[347] Z. Zhang, W. Shen, C. Yao, and X. Bai, "Symmetry-based text line detection in natural scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2558–2567.

[348] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," *International Journal of Computer Vision*, vol. 116, no. 1, pp. 1–20, 2016.

[349] W. Huang, Y. Qiao, and X. Tang, "Robust scene text detection with convolution neural network induced mser trees," in *European Conference on Computer Vision*. Springer, 2014, pp. 497–511.

[350] T. He, W. Huang, Y. Qiao, and J. Yao, "Text-attentional convolutional neural network for scene text detection," *IEEE transactions on image processing*, vol. 25, no. 6, pp. 2529–2541, 2016.

[351] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, and X. Xue, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Transactions on Multimedia*, 2018.

[352] ——, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Transactions on Multimedia*, 2018.

[353] Y. Jiang, X. Zhu, X. Wang, S. Yang, W. Li, H. Wang, P. Fu, and Z. Luo, "R2cnn: rotational region cnn for orientation robust scene text detection," *arXiv preprint arXiv:1706.09579*, 2017.

[354] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network." in *AAAI*, 2017, pp. 4161–4167.

[355] W. He, X.-Y. Zhang, F. Yin, and C.-L. Liu, "Deep direct regression for multi-oriented scene text detection," *arXiv preprint arXiv:1703.08289*, 2017.

[356] Y. Liu and L. Jin, "Deep matching prior network: Toward tighter multi-oriented text detection," in *Proc. CVPR*, 2017, pp. 3454–3461.

[357] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "East: an efficient and accurate scene text detector," in *Proc. CVPR*, 2017, pp. 2642–2651.

[358] C. Yao, X. Bai, N. Sang, X. Zhou, S. Zhou, and Z. Cao, "Scene text detection via holistic, multi-channel predic-

tion," *arXiv preprint arXiv:1606.09002*, 2016.

[359] C. Xue, S. Lu, and F. Zhan, "Accurate scene text detection through border semantics awareness and bootstrapping," in *European Conference on Computer Vision*. Springer, 2018, pp. 370–387.

[360] P. Lyu, C. Yao, W. Wu, S. Yan, and X. Bai, "Multi-oriented scene text detection via corner localization and region segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7553–7563.

[361] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," in *European conference on computer vision*. Springer, 2016, pp. 56–72.

[362] A. d. l. Escalera, L. Moreno, M. A. Salichs, and J. M. Armingol, "Road traffic sign detection and classification," 1997.

[363] D. M. Gavrila, U. Franke, C. Wohler, and S. Gorzig, "Real time vision for intelligent vehicles," *IEEE Instrumentation & Measurement Magazine*, vol. 4, no. 2, pp. 22–27, 2001.

[364] C. F. Paulo and P. L. Correia, "Automatic detection and classification of traffic signs," in *Image Analysis for Multimedia Interactive Services, 2007. WIAMIS'07. Eighth International Workshop on*. IEEE, 2007, pp. 11–11.

[365] A. De la Escalera, J. M. Armingol, and M. Mata, "Traffic sign recognition and analysis for intelligent vehicles," *Image and vision computing*, vol. 21, no. 3, pp. 247–258, 2003.

[366] W. Shadeed, D. I. Abu-Al-Nadi, and M. J. Mismar, "Road traffic sign detection in color images," in *Electronics, Circuits and Systems, 2003. ICECS 2003. Proceedings of the 2003 10th IEEE International Conference on*, vol. 2. IEEE, 2003, pp. 890–893.

[367] S. Maldonado-Bascón, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gómez-Moreno, and F. López-Ferreras, "Road-sign detection and recognition based on support vector machines," *IEEE transactions on intelligent transportation systems*, vol. 8, no. 2, pp. 264–278, 2007.

[368] M. Omachi and S. Omachi, "Traffic light detection with color and edge information," 2009.

[369] Y. Xie, L.-f. Liu, C.-h. Li, and Y.-y. Qu, "Unifying visual saliency with hog feature learning for traffic sign detection," in *Intelligent Vehicles Symposium, 2009 IEEE*. IEEE, 2009, pp. 24–29.

[370] S. Houben, "A single target voting scheme for traffic sign detection," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, pp. 124–129.

[371] A. Soetedjo and K. Yamada, "Fast and robust traffic sign detection," in *Systems, Man and Cybernetics, 2005 IEEE International Conference on*, vol. 2. IEEE, 2005, pp. 1341–1346.

[372] N. Fairfield and C. Urmson, "Traffic light mapping and detection," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 5421–5426.

[373] J. Levinson, J. Askeland, J. Dolson, and S. Thrun, "Traffic light mapping, localization, and state detection for autonomous vehicles," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 5784–5791.

[374] C. Bahlmann, Y. Zhu, V. Ramesh, M. Pellkofer, and T. Koehler, "A system for traffic sign detection, tracking, and recognition using color, shape, and motion information," in *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*. IEEE, 2005, pp. 255–260.

[375] I. M. Creusen, R. G. Wijnhoven, E. Herbschleb, and P. de With, "Color exploitation in hog-based traffic sign detection," in *2010 IEEE International Conference on Image Processing*. IEEE, 2010, pp. 2669–2672.

[376] G. Wang, G. Ren, Z. Wu, Y. Zhao, and L. Jiang, "A robust, coarse-to-fine traffic sign detection method," in *Neural Networks (IJCNN), The 2013 International Joint Conference on*. IEEE, 2013, pp. 1–5.

[377] Z. Shi, Z. Zou, and C. Zhang, "Real-time traffic light detection with adaptive background suppression filter," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 3, pp. 690–700, 2016.

[378] Y. Lu, J. Lu, S. Zhang, and P. Hall, "Traffic signal detection and classification in street views using an attention model," *Computational Visual Media*, vol. 4, no. 3, pp. 253–266, 2018.

[379] M. Bach, D. Stumper, and K. Dietmayer, "Deep convolutional traffic light recognition for automated driving," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 851–858.

[380] S. Qiu, G. Wen, and Y. Fan, "Occluded object detection in high-resolution remote sensing images using partial configuration object model," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 5, pp. 1909–1925, 2017.

[381] Z. Zou and Z. Shi, "Ship detection in spaceborne optical image with svd networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 5832–5845, 2016.

[382] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40, 2016.

[383] N. Proia and V. Pagé, "Characterization of a bayesian ship detection method in optical satellite images," *IEEE Geoscience and Remote Sensing Letters*, vol. 7, no. 2, pp. 226–230, 2010.

[384] C. Zhu, H. Zhou, R. Wang, and J. Guo, "A novel hierarchical method of ship detection from spaceborne optical image based on shape and texture features," *IEEE Transactions on geoscience and remote sensing*, vol. 48, no. 9, pp. 3446–3456, 2010.

[385] S. Qi, J. Ma, J. Lin, Y. Li, and J. Tian, "Unsupervised ship detection based on saliency and s-hog descriptor from optical satellite images," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 7, pp. 1451–1455, 2015.

[386] F. Bi, B. Zhu, L. Gao, and M. Bian, "A visual search inspired computational model for ship detection in optical satellite images," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 4, pp. 749–753, 2012.

[387] J. Han, P. Zhou, D. Zhang, G. Cheng, L. Guo, Z. Liu, S. Bu, and J. Wu, "Efficient, simultaneous detection of multi-class geospatial targets based on visual saliency modeling and discriminative learning of sparse coding," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 89, pp. 37–48, 2014.

[388] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, "Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 6, pp. 3325–3337, 2015.

[389] J. Tang, C. Deng, G.-B. Huang, and B. Zhao, "Compressed-domain ship detection on spaceborne optical image using deep neural network and extreme learning machine," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 3, pp. 1174–1185, 2015.

[390] Z. Shi, X. Yu, Z. Jiang, and B. Li, "Ship detection in high-resolution optical imagery based on anomaly detector and local shape feature," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 8, pp. 4511–4523, 2014.

[391] A. Kembhavi, D. Harwood, and L. S. Davis, "Vehicle detection using partial least squares," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 6, pp. 1250–1265, 2011.

[392] L. Wan, L. Zheng, H. Huo, and T. Fang, "Affine invariant description and large-margin dimensionality reduction

for target detection in optical remote sensing images," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 7, pp. 1116–1120, 2017.

[393] H. Zhou, L. Wei, C. P. Lim, D. Creighton, and S. Nahavandi, "Robust vehicle detection in aerial images using bag-of-words and orientation aware scanning," *IEEE Transactions on Geoscience and Remote Sensing*, no. 99, pp. 1–12, 2018.

[394] M. ElMikaty and T. Stathaki, "Detection of cars in high-resolution aerial images of complex urban environments," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 10, pp. 5913–5924, 2017.

[395] L. Zhang, Z. Shi, and J. Wu, "A hierarchical oil tank detector with deep surrounding features for high-resolution optical satellite imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 10, pp. 4895–4909, 2015.

[396] C. Zhu, B. Liu, Y. Zhou, Q. Yu, X. Liu, and W. Yu, "Framework design and implementation for oil tank detection in optical satellite imagery," in *Geoscience and Remote Sensing Symposium (IGARSS), 2012 IEEE International*. IEEE, 2012, pp. 6016–6019.

[397] G. Liu, Y. Zhang, X. Zheng, X. Sun, K. Fu, and H. Wang, "A new method on inshore ship detection in high-resolution satellite images using shape and context information," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 3, pp. 617–621, 2014.

[398] J. Xu, X. Sun, D. Zhang, and K. Fu, "Automatic detection of inshore ships in high-resolution remote sensing images using robust invariant generalized hough transform," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 12, pp. 2070–2074, 2014.

[399] J. Zhang, C. Tao, and Z. Zou, "An on-road vehicle detection method for high-resolution aerial images based on local and global structure learning," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 8, pp. 1198–1202, 2017.

[400] W. Diao, X. Sun, X. Zheng, F. Dou, H. Wang, and K. Fu, "Efficient saliency-based object detection in remote sensing images using deep belief networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 2, pp. 137–141, 2016.

[401] P. Zhang, X. Niu, Y. Dou, and F. Xia, "Airport detection on optical satellite images using deep convolutional neural networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 8, pp. 1183–1187, 2017.

[402] Z. Shi and Z. Zou, "Can a machine generate humanlike language descriptions for a remote sensing image?" *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 6, pp. 3623–3634, 2017.

[403] X. Han, Y. Zhong, and L. Zhang, "An efficient and robust integrated geospatial object detection framework for high spatial resolution remote sensing imagery," *Remote Sensing*, vol. 9, no. 7, p. 666, 2017.

[404] Z. Xu, X. Xu, L. Wang, R. Yang, and F. Pu, "Deformable convnet with aspect ratio constrained nms for object detection in remote sensing imagery," *Remote Sensing*, vol. 9, no. 12, p. 1312, 2017.

[405] W. Li, K. Fu, H. Sun, X. Sun, Z. Guo, M. Yan, and X. Zheng, "Integrated localization and recognition for inshore ships in large scene remote sensing images," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 6, pp. 936–940, 2017.

[406] O. A. Penatti, K. Nogueira, and J. A. dos Santos, "Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?" in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 44–51.

[407] L. W. Sommer, T. Schuchert, and J. Beyerer, "Fast deep vehicle detection in aerial images," in *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*. IEEE, 2017, pp. 311–319.

[408] L. Sommer, T. Schuchert, and J. Beyerer, "Comprehensive analysis of deep learning based vehicle detection in aerial images," *IEEE Transactions on Circuits and Systems for Video Technology*, 2018.

[409] Z. Liu, J. Hu, L. Weng, and Y. Yang, "Rotated region based cnn for ship detection," in *Image Processing (ICIP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 900–904.

[410] H. Lin, Z. Shi, and Z. Zou, "Fully convolutional network with task partitioning for inshore ship detection in optical remote sensing images," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 10, pp. 1665–1669, 2017.

[411] ——, "Maritime semantic labeling of optical remote sensing images with multi-scale fully convolutional network," *Remote Sensing*, vol. 9, no. 5, p. 480, 2017.