Computer Science Review 38 (2020) 100301

Contents lists available at ScienceDirect

Computer Science Review

journal homepage: www.elsevier.com/locate/cosrev

Review article

A comprehensive and systematic look up into deep learning based object detection techniques: A review

Vipul Sharma *, Roohie Naaz Mir

Department of Computer Science & Engineering, National Institute of Technology, Srinagar, India

ARTICLE INFO

Article history:
Received 12 May 2020
Received in revised form 20 July 2020
Accepted 26 August 2020
Available online 11 September 2020

Keywords:
Convolutional neural networks
Object detection
Localization
Segmentation
Classification
Deep learning

ABSTRACT

Object detection can be regarded as one of the most fundamental and challenging visual recognition task in computer vision and it has received great attention over the past few decades. Object detection techniques find their application in almost all the spheres of life, most prominent ones being surveillance, autonomous driving, pedestrian detection and so on. The primary focus of visual object detection is to detect objects belonging to certain class targets with absolute localization in a realistic scene or an input image and also to assign each detected instance of an object a predefined class label. Owing to rapid development of deep neural networks, the performance of object detectors has rapidly improved and as a result of this deep learning based detection techniques have been actively studied over the past several years. In this paper we provide a comprehensive survey of latest advances in deep learning based visual object detection. Firstly we have reviewed a large body of recent works in literature and using that we have analyzed traditional and current object detectors. Afterwards and primarily we provide a rigorous overview of backbone architectures for object detection followed by a systematic cover up of current learning strategies. Some popular datasets and metrics used for object detection are analyzed as well. Finally we discuss applications of object detection and provide several future directions to facilitate future research for visual object detection with deep learning.

© 2020 Elsevier Inc. All rights reserved.

Contents

1. Introduction 2
   1.1. Summary of previous survey papers on object detection 4
   1.2. Contributions of this survey 6
   1.3. Article organization 6
2. Traditional detectors vs. state-of-the-art detectors. 6
   2.1. Traditional object detectors 6
      2.1.1. Viola jones detectors: 6
      2.1.2. Histogram of oriented gradients (HOG) detector: 6
      2.1.3. Deformable part based model (DPM): 6
   2.2. Deep learning based object detectors 7
      2.2.1. One stage object detectors: 7
      2.2.2. Two stage detectors: 8
      2.2.3. Latest object detectors: 9
3. Proposal generation and feature representation learning for object detection: 10
   3.1. Proposal generation: 10
      3.1.1. Traditional proposal generation techniques: 10
      3.1.2. Anchor based proposal generation methods: 10
      3.1.3. Keypoint based proposal generation methods: 11
   3.2. Feature representation learning for object detection: 11
      3.2.1. Multi-scale feature representation learning: 11
      3.2.2. Contextual reasoning: 12
      3.2.3. Deformable feature learning: 12

* Corresponding author.
E-mail addresses: vipul_1phd17@nitsri.net (V. Sharma), naaz310@nitsri.net (R.N. Mir).

https://doi.org/10.1016/j.cosrev.2020.100301
1574-0137/© 2020 Elsevier Inc. All rights reserved.

4. Backbone architecture for object detection: ... 12
4.1. Basic architecture of a CNN: ... 12
4.2. CNN backbone for object detection: ... 13
5. Learning strategies for object detection ... 14
5.1. Learning strategies for training: ... 14
5.2. Testing stage: ... 16
5.3. Other latest learning strategies: ... 16
6. Popular datasets and metrics used for object detection ... 16
6.0.1. PASCAL VOC dataset: ... 17
6.0.2. MS-COCO dataset: ... 17
6.0.3. ImageNet Dataset: ... 17
6.0.4. OpenImages dataset: ... 18
6.0.5. VIS Drone 2018 dataset: ... 19
6.0.6. LVIS [1] : ... 19
6.1. Pedestrian detection dataset: ... 19
6.2. Face detection datasets: ... 19
7. Applications of object detection ... 19
7.1. Face detection: ... 19
7.2. Pedestrian detection: ... 20
7.3. Anomaly detection: ... 20
7.4. License plate recognition: ... 20
7.5. Autonomous driving: ... 20
7.6. Traffic sign recognition: ... 20
7.7. Computer aided diagnosis (CAD) system: ... 20
7.8. Event detection: ... 21
7.9. Pattern detection: ... 21
7.10. Image caption generation: ... 21
7.11. Salient object detection: ... 21
7.12. Text detection: ... 21
7.13. Point clouds 3D object detection: ... 21
7.14. 2D 3D pose detection: ... 21
7.15. Fine grained visual recognition: ... 21
7.16. Edge detection: ... 22
8. Concluding remarks and future research areas ... 22
8.1. Future research trends: ... 22
Declaration of competing interest ... 22
References ... 22
Glossary ... 28

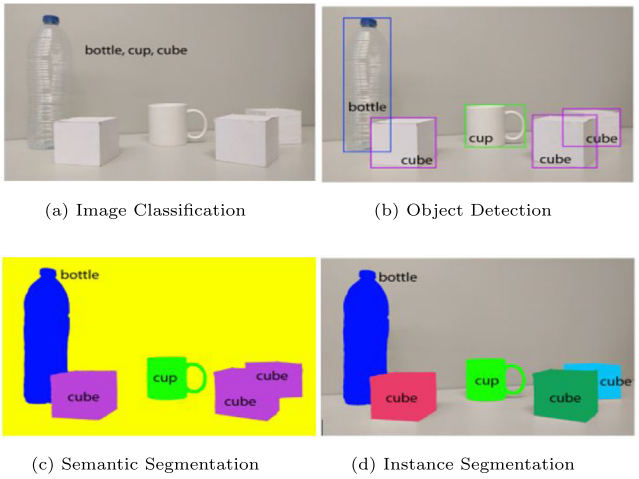
1. Introduction

Object detection is one of the most important computer vision tasks that deal with the detection of visual instances of objects of a certain class in a cluttered realistic scene or an input image. The primary objective of object detection techniques is to develop computational models which provide much needed information by applications based on computer vision. Due to its wide range of applications and latest advancements, object detection has attracted colossal amounts of attention in recent years. The task of object detection in both real world applications and academia is currently under rigorous investigation. Some prominent/ popular applications of object detection are autonomous driving, remote sensing, video surveillance and robotic vision. There are many factors that can be attributed for the speedy evolution of detection models. Among them deep convolutional neural networks and high computing powers of GPU can be considered as most important contributors for these advancements [2–4]. Presently deep learning based computational models are largely been employed for both generic and domain-specific object detection. These computational models act as backbone architecture in most of the object detectors and they are being used to perform a variety of tasks like extracting features from an input image, segmentation, classification and object localization etc. The field of computer vision encompasses various recognition problems, like image classification [5], semantic segmentation [6], object detection and instance segmentation [7,8]. An example covering these recognition problems is given in Fig. 1.

Image classification particularly aims to identify semantic categories/classes of objects in a given image, on the other hand the task of object detection is to not only identify the specific object categories but also to find out the precise locations of category specific objects with the help of bounding boxes. Talking of image segmentation, the aim is to assign a specific class label to each pixel thereby providing an even richer understanding of the image. However, if we are given an image comprising of multiple objects belonging to the same class category, then in that case semantic segmentation fails to draw a distinction between different object instances. So, in order to overcome this drawback a relatively new technique named instance segmentation is proposed which not only identify different object instances belonging to the same category but is also able to assign each of them a separate class pixel level mask. Therefore in instance segmentation pixel-level localization is required instead of bounding-box based object localization. A good object detection algorithm should have a robust understanding of semantic cues as well as spatial information about the image and in fact, as an important part of image understanding, object detection has been broadly used in many computer vision applications like face recognition [9–11], pedestrian detection [12–14], logo detection [15,16] and video analysis [17,18]. However in the meantime due to immense variations in lighting conditions and occlusions, different viewpoints and poses, the task of object detection along with localization has become a tedious one and it is very hard to accomplish with absolute perfection. So, due to these challenges much more attention has been given to this field in the recent years [7,19–21].

Table 1
List of Acronyms used in the paper.

Acronym	Full form	Acronym	Full form
mAP	Mean Average Precision	**NMS**	Non Maximum Suppression
FPS	Frames Per Second	**DCN**	Deformable Convolution Network
SIFT	Scale-Invariant Feature Transform	**MSCOCO**	Microsoft Common Objects in Context
SURF	Speeded Up Robust Features	**NAS-FPN**	Neural Architecture Search Feature Pyramid Network
HOG	Histogram of Oriented Gradients	**ResNet**	Residual Network
SVM	Support Vector Machine	**ReLU**	Rectified Linear Unit
DPM	Deformable Parts Model	**VGG**	Visual Geometry Group
ROI	Region of interest	**SGD**	Stochastic Gradient Descent
CNN	Convolution Neural Network	**DPN**	Dual Path Network
R-CNN	Region Proposal Convolution Neural Network	**RPN**	Region Proposal Network
R-FCN	Region-based Fully Convolution Network	**CRAFT**	Cascade Region Proposal Network and Fast-RCNN
FPN	Feature Pyramid Networks	**GAN**	Generative Adversarial Networks
SPP-net	Spatial Pyramid Pooling Network	**DSOD**	Deeply Supervised Object Detectors
YOLO	You Only Look Once	**ILSVRC**	ImageNet Large Scale Visual Recognition Challenge
SSD	Single Shot Multibox Detector	**LVIS**	Large Vocabulary Instance Segmentation
VOC	Visual Object Class	**M-DOD**	Multi-Domain Object Detection
PAMI	Pattern Analysis and Machine Intelligence	**CVIU**	Computational Vision and Image Understanding
ACM CS	Association of Computing Machinery CS	**PR**	Pattern Recognition
LIDAR	Light Detection and Ranging		



(a) Image Classification (b) Object Detection

(c) Semantic Segmentation (d) Instance Segmentation

Fig. 1. Progression of object recognition techniques: (a) image classification, (b) object detection, (c) semantic segmentation and (d) instance segmentation.

Table 1 include acronyms used in this paper.

The problem definition for generic object detection is to find out the precise positions, where the objects are located in a given image and what is the category to which each of these object instances belong to. Mathematical formulation of this problem of object detection is as given below:

Let us assume that, we are given a collection of ("M") images with annotations x1, x2, x3,…. xM, then for the *i*th image xi, let us assume that there are Ni objects present which belong to K categories/classes.

$$\text{yi} = \left\{\left(\text{k}_1^{\text{i}}, \text{p}_1^{\text{i}}\right), \left(\text{k}_2^{\text{i}}, \text{p}_2^{\text{i}}\right), \left(\text{k}_3^{\text{i}}, \text{p}_3^{\text{i}}\right) \ldots\ldots\ldots\ldots\ldots \left(\text{k}_{\text{Ni}}^{\text{i}}, \text{p}_{\text{Ni}}^{\text{i}}\right)\right\}$$

Here $k_j^i \left(k_j^i \in C\right)$ denotes category or class and p_j^{i} denotes the label mask of *j*th object in xi image respectively. The "d" detector is parameterized by "Q". For image "xi", the prediction $\text{y}_{\text{pred}}^{\text{i}}$ shares the same format as yi and it is as shown below:-

$$\text{y}_{\text{pred}}^{\text{i}} = \left\{\left(\text{k}_{\text{pred1}}^{\text{i}}, \text{p}_{\text{pred1}}^{\text{i}}\right), \left(\text{k}_{\text{pred2}}^{\text{i}}, \text{p}_{\text{pred2}}^{\text{i}}\right), \ldots\ldots\ldots\ldots\ldots \left(\text{k}_{\text{predNi}}^{\text{i}}, \text{p}_{\text{predNi}}^{\text{i}}\right)\right\}$$

In order to optimize the detector "d" a loss function loss is defined as : -

$$\ell(x, \theta) = \frac{1}{M}\sum_{i=1}^{M} \ell\left(y_{\text{pred}}^{i}, x_i, y_i; \theta\right) + \frac{\lambda}{2}\|\theta\|_2^2 \tag{1}$$

Here second term is a regularizer and λ is the trade-off parameter. In order to evaluate the quality of localization, we use a metric known as "intersection over union" between objects and the predictions as shown below:

$$\text{IoU}\left(p_{\text{pred}}, p_{\text{groundtruth}}\right) = \frac{\text{Area}\left(p_{\text{pred}} \cap p_{\text{groundtruth}}\right)}{\text{Area}\left(p_{\text{pred}} \bigcup p_{\text{groundtruth}}\right)} \tag{2}$$

Here $p_{\text{groundtruth}}$ refers to the ground truth bounding box. In order to find out whether the prediction accurately covers an object or not an IoU threshold value denoted by $^4\Omega''$ is set. Most commonly used value for this threshold is 0.5. for an object detection system, a prediction with correct class-label as well as successful localization prediction is referred to as positive prediction and negative-prediction otherwise.

For generic object detection mean average precision (mAP) over K categorical classes is used as an evaluation metric. Apart from detection accuracy, inference speed can also be regarded as an important metric to evaluate the performance of object detection system. In real world scenarios the efficiency of object detectors can also be evaluated using a metric called FPS (frames per second) i.e. how many number of images can be processed per second. A real time object detector can achieve an inference speed of 20 frames per second. Definition of "generic-object-detection" is to find out whether and where objects are located in a given image, this is known as object localization and which category/class each of the objects belong to. So, traditional object detection pipeline was divided into three steps (informative region selection, feature vector extraction and classification).

In informative region selection stage, as we know that different objects can be present at any position in an image and as the objective of this step is to search the specific locations in the image which may contain the objects. It is therefore very natural to scan the entire input image with the help of a mutiscale sliding window. The advantage of using a multiscale sliding window is that it can capture information about objects of different scales and aspect ratios and the obvious disadvantage is that it is computationally very expensive and produces many redundant windows. In order to identify different objects in an image, we need to extract features that can be used to represent meaningful information of the underlying detected objects. Therefore in the second stage of this detection pipeline, a fixed-size vector is extracted from the regions which are covered by the multiscale-sliding-window. The features extracted from these regions can be coded using descriptors like SIFT [22], SURF [23], Haar [24,25], HOG [26,27]. The primary reason behind using these descriptors to encode the extracted features is that, they are robust to scale,

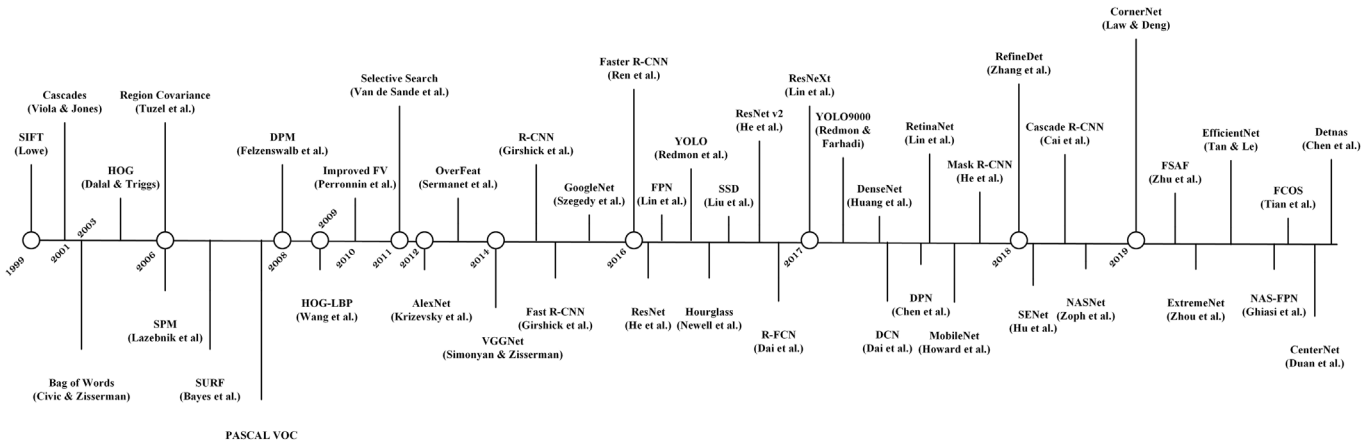


Fig. 2. Important milestones in object detection research since 1999.

illumination and rotation level variance. However, due to multi appearances, different backgrounds and poor illumination conditions, it is very difficult to manually design a descriptor for all kind of objects. Now having feature vectors in hand, a classifier is needed to separate a target object category from all the other object classes so as to make the feature representation more informative for visual recognition. For a classifier, support vector machines can be considered as a better option, this is due to their better performance on small scale training data. Apart from SVM, Adaboost [28,29], bagging [30] and DPM [31] can even be considered as viable options leading to further improvement in the detection accuracy.

Traditional methods developed for object detection aims at properly designing feature descriptors to obtain embedding for roi (“region of interest”). Owing to these advancements in feature vector representations and classification models, highly impressive results were obtained on PASCAL VOC datasets [32,33]. However during 2007 to 2012, minimum gains are obtained by building only ensemble models and using light variants of these traditional successful models. This is due to the following factors.

- During region proposal stage, a large number of redundant bounding boxes are generated using multi-window sliding strategy which leads to a large number of false positives for classification purpose.
- All the three stages of object detection pipeline are designed and optimized in a separate way and hence a global optimum solution for the entire setup is hard to obtain.
- It is very hard to bridge the semantic gap by using manually crafted low level features descriptors [34–36].

With the emergence of deep convolutional neural networks [37] and their widespread applications in image classification [5,38], object detection techniques based on deep learning [6,21] has achieved a significant amount of progress in recent years and they have outperformed the traditional detectors. Compared to manually crafted feature descriptors employed by traditional object detectors, deep CNN based object detectors generate hierarchical feature representations which is learned in a self automated way and try to show more discriminative expression power.

The advantages of CNN based methods against traditional object detectors is as summarized below:

- In contrast to shallow traditional models, a deep CNN based model provides an increased expressive power exponentially.
- CNN based object detectors provide an opportunity to optimize several detection related tasks together.
- In CNN based object detectors, hierarchical feature vectors/representations can be extracted automatically from the underlying data and can be disentangled through multi-level non linear mappings.

All these advantages made it obvious to design and develop deep learning based object detection techniques with expressive feature representation capability that can be optimized in an end-to-end manner. These properties further made it possible for CNN’s to find out their applicability into many research domains like image classification [39], face recognition [40], video analysis [41,42] and pedestrian detection [27,43,44].

Recent deep learning based object detection architectures can be broadly divided into two categories.

- Single stage detectors like YOLO [20,45] and its different variants [46,47].
- Two stage detectors like R-CNN [7] and its variants [19,21,48].

Single stage detectors directly maps class prediction of different objects present at each location of the extracted feature maps without utilizing the region classification step, whereas in case of two stage detectors, proposal generator is used to generate a coarse set of region proposals from which feature vectors are extracted which are then used to predict the category of objects using region classifiers. Compared to single stage detectors, two stage detectors generally achieve good detection performance and generate state-of-the-art results on publically available datasets, However, single stage detectors are comparatively more time efficient and hence are greatly applicable for real time object detection [2,4]. In Fig. 2 major developments and milestones of deep learning based object detection techniques are illustrated.

1.1. Summary of previous survey papers on object detection

Many prominent survey papers on object detection have been published over the years as summarized in Table 2. These cover many admirable reviews on some specific object detection problems like face detection [49,50], text detection [51], pedestrian detection [52–54], vehicle detection [55] etc. There are relatively very few survey papers which directly focuses on the problem of deep learning based generic object detection techniques except for Zhang et al. [56] who conducted object class detection survey in the year 2013, Jiao Licheng et al. [57] whose survey focuses on describing and analyzing deep learning based object detection task in the year 2019, followed by Zhao et al. [58] who provided a systematic review to summarize representative models and

Table 2
Summary of previous survey papers on object detection.

S.No	Title of Survey	References	Venue	Year	Content
1	Detecting faces in images: a survey	Yang et al. [49]	PAMI	2002	First survey of face detection from a single image
2	On road vehicle detection: a review	Sun et al. [59]	PAMI	2006	A review of computer vision based on-road vehicle detection systems
3	Toward category level object recognition	Ponce et al. [60]	Book	2007	Survey papers on object classification, detection, and segmentation
4	The evolution of object categorization and the challenge of image abstraction	Dickinson et al. [61]	Book	2009	An outline of the evolution of object categorization over 4 decades
5	Monocular pedestrian detection: survey and experiments	Enzweiler and Gavrila [53]	PAMI	2010	An assessment of three pedestrian detection techniques
6	Survey of pedestrian detection for advanced driver assistance systems	Geronimo et al. [54]	PAMI	2009	A thorough review of pedestrian detection for advanced driver assistance systems
7	Context based object categorization: a critical survey	Galleguillos and Belongie [62]	CVIU	2010	A review of perspective information for object categorization
8	Visual object recognition	Grauman and Leibe [63]	Tutorial	2011	Object recognition techniques based on instance and category
9	Pedestrian detection: an evaluation of the state of the art	Dollar et al. [52]	PAMI	2012	A detailed assessment of detectors in monocular images
10	50 years of object recognition: directions forward	Andreopoulos and Tsotsos [64]	CVIU	2013	A review of the evolution of object recognition systems over 5 decades
11	Object class detection: a survey	Zhang et al. [56]	ACMCS	2013	Survey of generic object detection methods before 2011
12	Representation learning: a review and new perspectives	Bengio et al. [65]	PAMI	2013	Unsupervised feature learning and deep learning, probabilistic models, autoencoders, manifold learning, and deep networks
13	Salient object detection: a survey	Borji et al. [66]	arXiv	2014	A survey of salient object detection techniques
14	A survey on face detection in the wild: past, present and future	Zafeiriou et al. [50]	CVIU	2015	A survey of face detection techniques in the wild since 2000
15	Text detection and recognition in imagery: a survey	Ye and Doermann [51]	PAMI	2015	A thorough review of text detection and recognition in colored images
16	Feature representation for statistical learning based object detection: a review	Li et al. [67]	PR	2015	Feature representation methods in statistical learning based object detection, including handcrafted and deep learning based features
17	Deep learning	LeCun et al. [68]	Nature	2015	An introduction to deep learning and applications
18	A survey on deep learning in medical image analysis	Litjens et al. [69]	MIA	2017	A survey on deep learning based image classification, object detection, segmentation and registration in medical image analysis
19	Recent advances in convolutional neural networks	Gu et al. [70]	Nature	2015	An introduction to deep learning and applications
20	Deep learning	LeCun et al. [68]	PR	2017	A broad survey of the recent advances in CNN and its applications in computer vision, speech and natural language processing
21	Object detection in 20 years: A survey	Zou et al. [71]	IEEE	2019	A comprehensive review in the light of technical evolution of different object detection techniques in last two decades
22	A survey of deep learning based object detection	Jiao et al. [57]	IEEE	2019	A broad survey focuses on describing and analyzing deep learning based object detection task
23	Deep learning	LeCun et al. [58]	arXiv	2019	A systematic review to summarize representative models and their different characteristics in several generic object detection application domains
24	Recent advances in deep learning for object detection	Wu et al. [4]	Neurocomputing	2020	Presents a comprehensive understanding of deep learning based object detection algorithms

their different characteristics in several generic object detection application domains in the year 2019. Apart from these two more surveys came in the same year (2019) conducted by Zou et al. [71] where they provided a comprehensive review in the light of technical evolution of different object detection techniques followed by Wu et al. [4] presenting a comprehensive understanding of deep learning based object detection algorithms. However the research reviewed in these surveys lack to provide a systematic view of deep learning based object detection techniques, the learning strategies that need to be employed, up to date state of art detection solutions and are therefore prior to the recent striking success and dominance of deep learning and related methods.

The advantage of deep leaning is that, it permits computational models to learn implausible complex, restrained and intangible depictions, thereby making significant inroads into a large set of problems like medical image analysis, Natural Language Processing, speech recognition, generic object detection etc. Among various types of deep neural networks, deep convolutional neural networks have revolutionized the fields of image processing, video, speech and audio detection. There has been a wide range of many published surveys on deep learning including that of Bengio et al. [65], Lecun et al. [68], Gu et al. [70], Litjen et al. [69] and more recently in tutorials at ICCV and CVPR.

In disparity even though many deep learning based methods been set forth for object detection. we are not aware of any systematic and recent comprehensive survey covering latest strategies and techniques related to generic object detection. A detailed review and cover up of existing techniques is very much necessary for additional progress in object detection, notably for researchers who are willing to enter this field.

1.2. Contributions of this survey

The primary focus of this survey is to describe and analyze deep learning based object detection problem. Unlike existing surveys, this paper covers state of the art techniques for object detection and also provides various learning strategies for this task along with future research directions due to rapid development in computer vision research.

- This research survey is featured by an in-depth analysis and thorough discussion in various aspects, some of which are new to the best of our knowledge.
- In this paper, we have listed down some object detection learning strategies and have neglected to provide the basic information, apart from that latest object detection trends are also discussed so that readers can themselves witness the advancements in this field easily.
- Unlike previous surveys in this field, this paper provides a comprehensive and systematic look up into deep learning based object detection techniques, a write up related to most significant research trends and an up to date object detection algorithms.

So, the goal of this survey paper is to provide a thorough analysis of object detection techniques based on deep learning.

1.3. Article organization

Rest of the manuscript is organized in the following sections: In Section 2 we have thoroughly compared traditional detection techniques with current object detectors. Proposal generation and feature representation learning techniques for object detection are covered in Section 3. Backbone architectures for object detection are presented in Section 4 followed by learning strategies for object detection which are covered in Section 5. Popular datasets and metrics used for generic object detection are provided in Section 6. Detection algorithms for real-world-applications are covered in Section 7 and finally we conclude and discuss future research directions in Section 8.

2. Traditional detectors vs. state-of-the-art detectors.

In the last 20 years, it has been unanimously accepted that the advancements in the field of object detection has been made in two historical periods. One is traditional object detection era before the year 2014 and other is deep learning based object detection after 2014 [71].

2.1. Traditional object detectors

Most of the traditional object detection techniques were developed using manually crafted features. Due to the absence of advanced image representation techniques before 2012, the only choice researchers have is to design highly complex feature vector representations along with a colossal variety of speed up techniques to compensate the mismatch which arises due to the availability of limited computational resources.

2.1.1. Viola jones detectors:

P. viola and M. jones, 19 years ago, developed a human face detector and have achieved a real time detection accuracy [72,73]. This detector has outperformed any other algorithm in its time under comparable detection accuracy metric. It follows a sliding window approach and covers all the possible locations in an input image to figure out, if any of the windows capture a human face. Although sliding window approach is a time consuming process, but still VJ detector is able to improve its detection speed by incorporating three techniques: integral image, which is basically a way to speed up the convolution process. It helps in making the computational complexity of each window independent of its size. Second is feature selection, where adaboost algorithm [74] is incorporated to select a small set of feature vectors about 180k dimensional which are helpful for face detection. Moreover, in order to reduce the computational complexity, detection cascades are introduced which is a basically a multistage detection pipeline, wherein face targets are given more computational importance.

2.1.2. Histogram of oriented gradients (HOG) detector:

HOG feature descriptor was first of all proposed by N. dalal and B. Triggs in the year 2005. [26] HOG feature can be regarded as a break through advancement of SIFT features and contextual shapes. It was basically designed to perform computations on a dense grid of cells which are uniformly spaced. HOG can be used for detecting a wide variety of object categories, but the primary motivation behind developing HOG was pedestrian detection. The best advantage of HOG detector is that it can be used to detect objects of different sizes and it can be achieved by rescaling the input image a number of times, while keeping the size of detection window unaltered [75–77].

2.1.3. Deformable part based model (DPM):

In the year 2008, deformable part based model was proposed by P. Felzenszwalb [75]. It was originally developed as an extension to HOG detector and later a lot of improvements were made by R. Girshick [31,76,78,79]. In deformable part based model divide and conquer strategy is followed, where in a given input image can be considered as an amalgamation of detection operations carried out on different object within that image. For e.g. The problem of detecting a bird can be considered as a problem of detecting its beak, legs, tail and wings. In a basic DPM detector,

we have a root filter along with a number of part filters. The configurations related to these part filters can be learned using a weakly supervised learning strategy instead of specifying them manually. For improving the detection results R. Girshick [80] has formulated this whole process as a unique case of multi-instance learning by incorporating some other special techniques like bounding box regression and context priming. Moreover R. grishick has further developed a strategy for compiling detection models to be build as a cascade architecture which is 10 times faster than the traditional models without compromising its accuracy [76,79]. It should be noted that although majority of object detectors have outperformed DPM based models in terms of detection accuracy, but still most of them are profoundly influenced by its precious insights.

2.2. Deep learning based object detectors

Deep learning based object detectors can broadly be divided into two major categories: two stage detectors and one stage detectors. Speaking of two stage detectors, in the first stage a meager set of underlying proposals are generated and then in the next stage feature vectors are extracted from these proposals which are then encoded using deep convolutional neural networks followed by making predefined class predictions. A one stage detector on the other hand typically mulls over all the positions on the input image as prospective target objects and try to categorize each region of interest as either a target object or simply background. One stage detectors are much quicker and highly viable for real time object detection applications but exhibit comparatively poor performance when compared with two stage detectors which often report remarkable results on many publically available datasets.

2.2.1. One stage object detectors:

As mentioned earlier, one stage detectors do not possess a separate branch for generating proposals. These detectors basically mulls over all the locations of an input image as prospective objects and further try to categorize each ROI (region of interest) into either background or a predefined object class. Let us now quickly review some prominent state-of the-art one stage detectors.

(i) Overfeat: It is one of the earliest single stage deep learning based object detectors proposed by sermanet et al. [81]. We know that the task of object detection can be perceived as multi-section categorization problem and therefore overfeat comprehended the earlier classifiers into a detector by considering the deeper fully connected layers as 1 x 1 convolutions so as to permit arbitrary sized input images. The categorization arrangement of overfeat detector produces a grid of forecast on each input region to point out the presence of an object. Once objects are discovered, bounding box regressors are used to refine the predicted regions. For recognizing multi-scale objects the input image is reorganized and rescaled into multiple images which are then fed into the overfeat setup. At last the predictions across all these scales are grouped together. Overfeat has achieved a remarkable speed strength as compared to RCNN by distributing the overlapped computational regions using convolutional layers. Despite speed strength it is still difficult to optimize the training of classifier and regressor.

(ii) YOLO: In the year 2015, R. Joseph et al. [20] came up with a real time one stage detector based on deep learning called YOLO as shown in Fig. 3. YOLO basically perceives object detection as a regression problem and divides the entire image into a predefined number of matrix cells. Each of the underlying cell is then considered as a proposal to detect the occurrence of objects. It is exceptionally fast, an extremely quick version of YOLO runs at 155 fps (frames per second) and achieves an mAP of 52.7% on VOC2007 dataset whereas its enhanced versions work at 45 fps and achieves an mAP of 63.4% and 57.9% on VOC2007 and VOC2012 datasets respectively. YOLO basically follows a state-of-the-art strategy to use a single neural network for an entire image. This setup as we have discussed earlier partitions the input image into a number of regions and forecasts probabilities along with bounding boxes for each of these regions simultaneously. later on R. Joseph et al. came up with a chain of improvements on YOLO and released YOLOv2 and YOLO v3 versions [46,82] which not only improves the detection accuracy but also improves the detection speed. No matter YOLO has made a remarkable improvement in detection speed, but it still suffers from inaccurate localizations when compared with state-of-the-art two stage detectors. Moreover YOLO could recognize upto only two objects at a given location and hence it is very difficult for it to detect small and crowded objects.

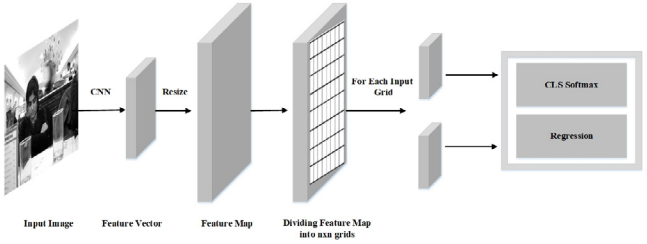


Fig. 3. YOLO framework for generic object detection.

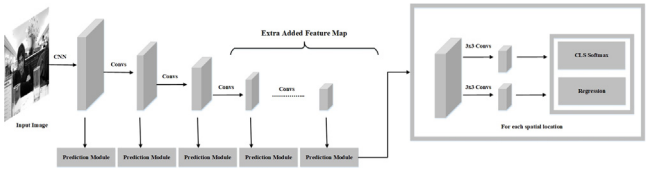


Fig. 4. SSD framework for generic object detection.

(iii) SSD: For overcoming the limitations of YOLO, Lin et al. in the year 2016 proposed single shot multibox detector [47,48] as shown in Fig. 4. The most important contribution of SSD is the use of multi-reference and multi-resolution techniques which exceptionally improves the detection accuracy of one stage detector for detecting small objects. Like YOLO, SSD also partitions the input image into matrix cells, but unlike recognizing objects from predefined grid cells as in YOLO, a set of anchors with varied scales, sizes and aspect ratios are produced to refine the bounding boxes. SSD achieves a remarkable mAP of 76.8% and 74.9% on VOC2007 and VOC 2012 datasets respectively. The primary difference between SSD and other former detectors is that SSD runs detection tests only on the deeper layers while former single stage detectors runs detection test on different layers of the network using different scales.

(iv) RetinaNet: In spite of its high detection speed and effortlessness, single stage detectors are outperformed by two stage detectors in terms of detection accuracy. The reason behind this is foreground-background category imbalance, so to overcome this problem Lin et al. [83] in the year 2017 came up with Retinanet [83] as shown in Fig. 5. In Retinanet an all new loss function called “focal

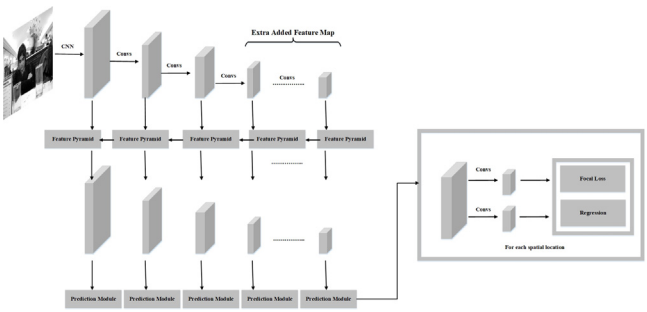


Fig. 5. RetinaNet framework for generic object detection.

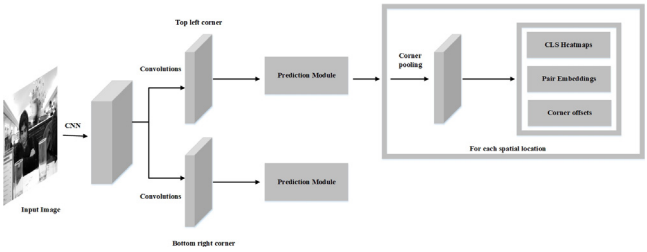


Fig. 6. Cornernet framework for generic object detection.

loss" has been introduced by suppressing the gradients of negative samples instead of straightaway discarding them. Focal loss function thus enables a single stage detector to achieve somewhat equivalent detection accuracy of two stage detectors without compromising the detection speed. In addition Retinanet achieves an mAP of 59.1% on MSCOCO dataset.

(v) Cornernet: Subsequent single stage object detectors made use of anchor boxes which are manually designed for training purpose. In the later years a chain of anchor free object detectors were introduced, where the idea was to find out the important locations in the bounding box instead of fitting an object to an anchor. Considering this idea, Law and Deng came up with an original anchor free setup called cornernet [84] as shown in Fig. 6, wherein objects are recognized as a pair of corners. On each possible locations of the generated feature map, pair embedding, corner offset and category specific thermal maps are predicted. Thermal maps computed the probabilities of being corners, corner offset are used for refining the corner locations and pair embedding is used for grouping together corner pairs belonging to the same object. Due to this cornernet achieved a significant performance on MSCOCO dataset [85,86].

2.2.2. Two stage detectors:

Two stage detectors divide the detection task into two phases (i) generation of proposals; and (ii) making class predictions from these underlying generated proposals; The idea basically is to find out regions with larger recall values in such a way that all the objects within the image categorize themselves in any case to one of the proposed regions, this is done during the first stage. Whereas in the second stage convolutional neural network based models are utilized to categorize these generated proposals into either an object from predefined class categories or simply into background. Now, let us review some prominent state-of-the-art two stage object detectors.

(i) **R-CNN:**It is basically a region based object detector proposed by Girshick et al. [7] in the year 2014. R. Girshick was

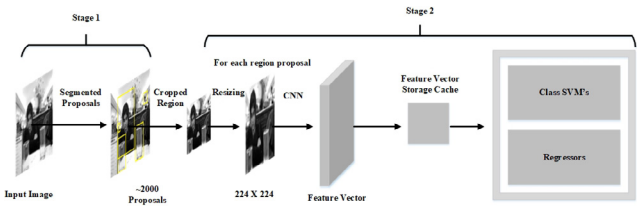


Fig. 7. R-CNN framework for generic object detection.

the first to showcase that a CNN based model can be used to achieve a remarkable object detection performance on publically available PASCAL visual object class datasets [33] than the traditional models which are based on simple HOG features. R-CNN drastically improved the detection performance and achieved 53.7% mAP (mean average precision) on PASCAL VOC dataset. A simple R-CNN detector shown in Fig. 7 consists of three components: (i) generation of proposals; (ii) extracting features from underlying proposals; (iii) categorization of regions into predefined object classes. Using selective search technique [87] for each of the input image, R-CNN produces a meager set of proposals about 2000 in number and discards the ones which can simply be discovered as a background region. Each of these proposals are cropped and adjusted into a fixed sized region which is then used to generate a feature vector (particularly 4096 dimensional) with the help of a deep convolutional neural network trained on ImageNet, this transfer of knowledge from ImageNet dataset offers considerable performance achievement. Finally linear SVM classifiers are bring into play to predict the presence of an object and to identify predefined object categories [71]. In addition to this the last fully connected layer of the underlying CNN is re-initialized for the detection work. Moreover R-CNN discards a large number of simple negatives before training begins and this helps in improving the pace of learning and mitigating the number of false positives. No matter R-CNN has made a remarkable progress, its shortcomings are also apparent: It is extremely time consuming for both training and testing because of superfluous feature computations from highly overlapped regions, which leads to an extremely slow detection speed which is 14 s per image with GPU.

(ii) **SPP-net:** In the year 2014, He et al. stirred by the idea of spatial pyramid matching [88] proposed SPP-net [89] as shown in Fig. 8 to overcome the shortcomings associated with R-CNN. The significant role of SPP-net is the introduction of spatial pyramid pooling layer which assists a CNN to produce a predetermined length feature vector representation despite of the size of image's area of significance devoid of rescaling. SPP divides the feature map into an M x M matrix for varying values of M and performs pooling on each and every cell of the created matrix to produce a feature vector. All the feature vectors thus obtained from each cell of the matrix are then joined together and provided as an input to region SVM categorizers and bounding box regressors. SPP-net is 20 times more efficient and faster than R-CNN without compromising its detection accuracy. On VOC 2007 SPP-net achieved a mean average precision (mAP) of 59.2%. No matter it has efficiently improved the detection speed and accuracy but still there are some shortcomings associated with it. Firstly training scenario is multiphase in nature and secondly it majorly depends on its fully connected layers for the detection task and simply ignores all the previous layers.

(iii) **Fast R-CNN:** To further improve the detection accuracy of R-CNN and SPP-net [7,79,89]. R. Girshick et al. in the year 2015 came up with fast R-CNN [19] as shown in Fig. 9 which is a multitask learning detector and can extract predetermined length feature vectors from the feature map like SPP-net did. But the difference between fast R-CNN and SPP-net is that in Fast R-CNN ROI pooling layer is used to extract the features associated with the regions. ROI pooling is a unique case of SPP which only considers a single scale to divide the underlying proposal in predetermined number of partitions and also back propagate the error signals to the convolution kernels. Extracted features are then fed as an input to a collection of fully connected layers followed by a classification and a regression layer which generated softmax probabilities for C + 1 classes and top left and bottom right coordinates for redefining the bounding boxes respectively. However Fast R-CNN is successful in combining the benefits of both R-CNN and SPP-net, but still its detection speed is limited by a number of factors.

(iv) **Faster R-CNN:** Although a lot of progress has been made designing learning detectors, but still generation of proposals heavily relied on conventional techniques like selective search [87] and edge boxes [90] which utilizes trivial visual signals and are hard to learn using a data driven approach. To address these issues S. Ren et al. came up with Faster R-CNN [21] as shown in Fig. 10 in the year 2015 shortly after the release of Fast R-CNN. Faster R-CNN makes use of a fully convolutional layer called RPN (region proposal network) which works on arbitrary images of varied sizes and creates a set of proposals probably on each feature map location. The feature vectors thus obtained from the derived feature map are then fed into a classification layer followed by a bounding box regression layer for object localization which finally leads to object detection. Faster R-CNN can be considered as a first end-to-end and near real time deep learning detection model that achieved an mAP of 73.2% on VOC2007 and an mAP of 70.4% on VOC2012 datasets respectively. No matter Faster R-CNN outperforms Fast R-CNN in terms of detection speed, still there is some computational idleness at successive detection stages.

(v) **R-FCN:** The problem with Faster R-CNN is that it computes feature map of an arbitrary input image and derives region based feature vectors from it which shares feature extraction computation through different regions of the image. Such extra computations could be exceptionally hefty because each of the input image may have hundreds of proposals. To overcome this additional computation cost Dai et al. [91] proposed region based fully convolutional networks (R-FCN) as shown in Fig. 11 which shares total computation cost in the region classification step. R-FCN achieved better performance as compared to Faster R-CNN without operating on region based fully connected layers. This is because R-FCN creates a location sensitive score map which encodes comparative location information related to different categories and also makes use of location sensitive ROI pooling layer to derive spatially aware region features by encoding each comparative location of the target region.

(vi) **Feature Pyramid Networks (FPN):** In deep convolutional neural networks features belonging to deep layers are semantically strong but spatially weak whereas features belonging to shallow layers are sematically weak but spatially strong. Lin et al. [48] considered this property and proposed FPN [48] wherein deep layer features are integrated with the shallow ones to enable detection of objects at different scales. The underlying idea is to reinforce spatially sound features with well to do semantic features from the deeper layers. Using this FPN in a traditional Faster R-CNN model achieves better detection results with mAP of 59.1% on COCO dataset. Further it should be noted that FPN has now become an essential building block of latest detectors.

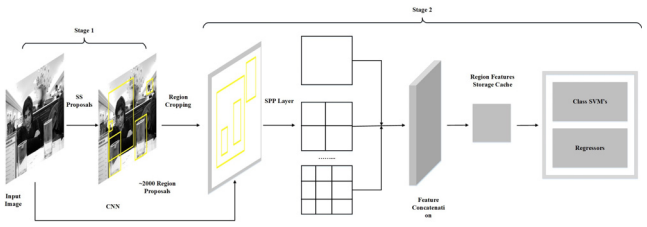


Fig. 8. SPP-net framework for generic object detection.

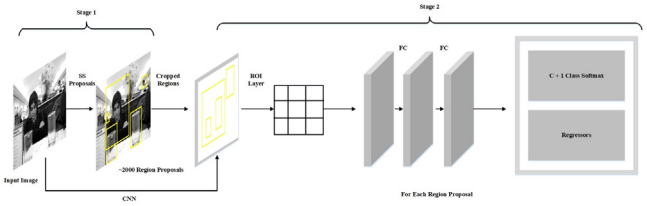


Fig. 9. Fast R-CNN framework for generic object detection.

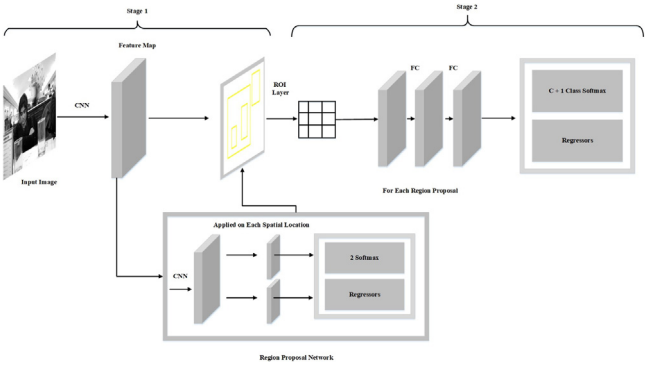


Fig. 10. Faster R-CNN framework for generic object detection.

(vii) **Mask R-CNN:** Mask R-CNN [8,92] was proposed by He et al. as an extension to Faster R-CNN purposely for instance segmentation which predicted bounding boxes and masks used for segmentation parallel from the underlying proposals and achieved state-of-the-art results.

2.2.3. Latest object detectors:

(i) Relation Networks for Detecting Objects: Different object targets present in an image having distinct façade features and geometrical information needs to be interacted together. Considering this requirement, He et al. [93] came up with an improved attention network known as object relation module which is inserted in the network before two fully connected layers to extract superior features for proper categorization and localization of target objects. This added relation unit supplies superior features to regressor and classifier, but substitute NMS post processing step thereby increasing the overall detection accuracy.

(ii) Deformable Convolutional Networks: Regular convolutional neural networks consider features of predefined square size kernels and due to this the amenable region field

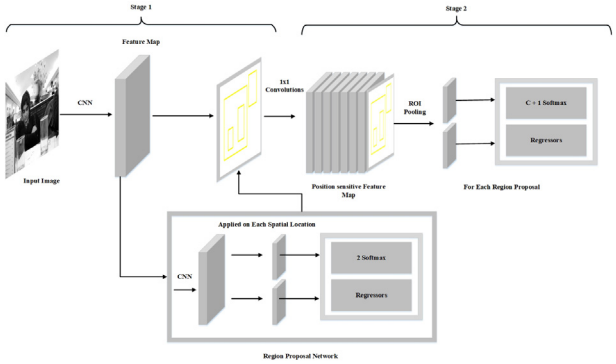


Fig. 11. R-FCN framework for generic object detection.

does not cover up the entire pixel area of the target objects properly. So, to overcome this problem and to adapt the geometric variations replicated in the useful spatial support regions of the target objects, DCN (Deformable Convolutional Networks) is proposed by Dai et al. [94] in which deformable kernels along with box offsets from the initial convolutional anchors can be learned and are used to adapt to the part locations of the target objects with different shapes and sizes. In addition to all this, DCN achieves a significant detection accuracy which is almost 4% higher than traditional convents and also achieves an mAP of 37.5% on MSCOCO.

(iii) NAS-FPN: In recent years Google Brain authors take up neural architectural search to discover some latest feature pyramid architecture called NAS-FPN [95] which consists of both bottom up and top down relations to fuse together features with a number of different scales. During the search, the underlying FPN architecture is repeated N no. of times and are concatenated together thereby helping deeper feature layers to randomly pick arbitrary features to emulate. Hoarding more pyramid networks, adding feature dimensions and adopting large capacity architectures helps in increasing the detection accuracy. Experiments have shown that the mAP of NAS-FPN superseded the original FPN by 2.9% when ResNet-50 with 256 dimensional features is used as a backbone network on COCO test-dev dataset. In conclusion we can say that the traditional baselines improve the detection accuracy by extracting rich features of target objects and adopting multi level and multi scale features for different sized object detection.

3. Proposal generation and feature representation learning for object detection:

3.1. Proposal generation:

Proposal generation, which is used for creating a set of rectangular bounding boxes surrounding potential target objects, plays a very significant role in object detection and recognition framework. The rectangular boxes thus generated are then used for the purpose of classification and localization refinement. Proposal generation methods can be categorized into three different categories, these are traditionally used computer vision techniques, key-point based methods and anchor based techniques. it should be noted that both single-stage and double-stage object detectors make use of proposal generation techniques, the important difference is, the single stage detection consider each region in the input image as a potential target proposal and thereby it makes an estimation regarding the object class and bounding box coordinates, whereas in double-stage detector, a meager set of proposals containing only foreground and background information are generated. Let us now briefly different proposal generation techniques.

3.1.1. Traditional proposal generation techniques:

These traditional methods make use of edge, color and corner information for generating proposals. These traditional methods can be divided into three principle categories. (i) calculating the objectness score of the candidate rectangular boxes. (ii) assimilating superpixels from the original image. (iii) engendering many foreground and background segments.

(i) Objectness Score based Methods: These methods are used to determine the score associated with each of the candidate proposals which is a parameter used to determine the presence of an object inside the bounding box. Arbelarez et al. [96] used visual measures like saliency and density of edges to assign the objectness scores. later Rahtu et al. [97] introduced a highly effective cascaded learning based approach to assign objectness scores to candidate proposals.

(ii) Assimilation of Superpixels: This method of proposal generation merges superpixels engendered from the image segmentation results. one such technique based on this concept is selective search [87] where multiple hierarchical segments were combined together using their visual factors followed by placing bounding boxes on the merged segments. Another similar idea was proposed by Manen et al. [98] where merging process was highly randomized.

(iii) Seed Segmentation based Proposal Generation: Here multiple foreground and background segments are generated and to avoid build up of hierarchical segments, CPMC [99] produced a number of overlapped regions initiated with different seed values. The ideas of selective search [87] and CPMC [99] were combined together by Enreds and Hoein [100] to merge generated superpixels with newly designed features which are then used as seeds for generating larger segments.

The most important advantage of traditional proposal generation methods is that they are straight forward and can produce proposals with high recall value, still they cannot be combinely optimized with the entire object detection pipeline, because they are based on low level visual features like color and edges.

3.1.2. Anchor based proposal generation methods:

It is defined as a large unit of supervised proposal generators which generate proposals based on already defined anchor points. For generating proposals in a supervised manner based on deep convolutional feature maps, Ren et al. [21] came up with region proposal network, where the network slid over the entire feature map using 3X3 convolution filter. For each location "k" anchors of different sizes and aspect ratios were taken into consideration. These different sizes and aspect ratios thereby helped in matching objects at varying scales in the entire image. Depending upon the ground truth available the object locations are matched with the most suitable anchors and a 256 dimensional feature vector is generated for each anchor which is fed into classification and regression layers Based on the ground truth, each of the anchor is predicted to be either an object or simply a background as shown in Fig. 12.

Despite showing remarkable performance, the anchors are designed manually using multiple scales and aspect ratios in a heuristic manner. These designs may not be optimal and hence it require different design strategies to build them. In the recent years many things have been proposed to improve the overall

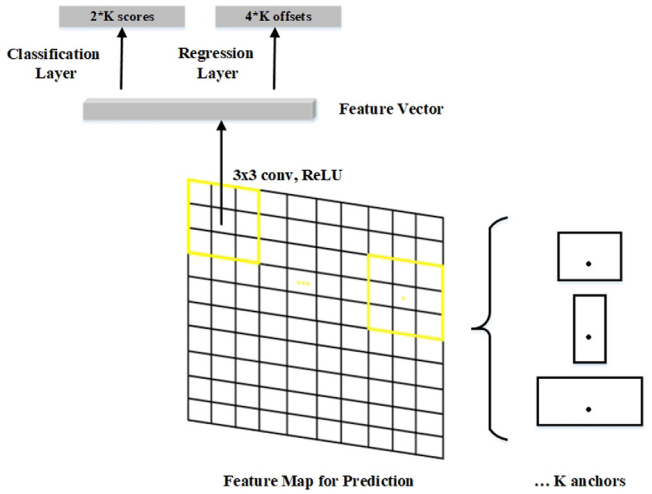


Fig. 12. Diagram of RPN, Here each feature of the feature map connects with a sliding window followed by two branches.

design choice of the anchors. Zhu et al. [101] proposed an anchor designing technique for matching small objects by enlarging the input image while reducing the stride rate. Xie et al. [102] came up with dimension decomposition region proposal network which is used for decomposing the dimensions of anchor boxes based on RPN, thereby helping to match objects with large scale variance and reduced searching space.

3.1.3. Keypoint based proposal generation methods:

These techniques are broadly divided into corner-based techniques and center-based methods. In corner-based techniques, bounding boxes are generated by merging together a pair of corners learned from the feature map. Denet [103] sculpted the distribution of corner points on the feature map an applied a bayesian classifier on each of the corners of the object to generate the confidence score of the bounding box. With these corner-based methods design of anchor points has been eliminated and thus it becomes one of the most effective method to generate high quality proposals. Using Denet [103] Law and Deng proposed a framework called CornerNet [84] which directly modeled categorical information on the corner points.

Coming to center-based proposal generation methods, the probability of a point being the center of an object is predicted on each possible location of the feature map and its weight and height are directly regressed without any prior knowledge of the anchors. Zhu et al. [104] proposed a feature selection anchor free model which can be inserted into single-stage object detector with the FPN structure. In this model an online feature selection module is used to train multi-level center-based branches at each level of the feature pyramid. Zhou et al. [85] developed a new center-based model based on single hourglass network [84] without an FPN module and they used it for 3 dimensional detection and human pose recognition. Duan et al. [105] proposed a model called centerNet which is based on both center-based and corner-based methods. CentreNet achieved a remarkable improvement when compared with other baseline models. All these anchor-free techniques for proposal generation presents a prominent future research direction.

3.2. Feature representation learning for object detection:

It is the most significant component in the entire object detection framework. Due to the presence of target objects in difficult environments and having different aspect ratios and scale invariance, there is an immediate need to train a vigorous and discriminative feature embedding of objects, so as to obtain significant detection results. So, here we propose some feature representation learning strategies for detection and recognition of objects. Exclusively we have identified three feature representation learning strategies: multi-scale feature representation learning, deformable feature representation learning and contextual reasoning.

3.2.1. Multi-scale feature representation learning:

We know that many object detection techniques based on deep convolutional neural networks like Fast R-CNN [106] and Faster R-CNN [21] make use of only a single layered feature map for detection. But to detect the objects across multiple scales and aspect ratios is somewhat challenging using a single feature map. Hierarchical features in different layers capturing varying scale information is learned using deep convolutional neural networks. As we have already seen that for detecting tiny objects, features with rich spatial information having narrow receptive fields and high resolutions can be extracted from shallow layers whereas features present in deep layers having large receptive fields and semantic rich information are more suitable for detecting large objects.

Some of the techniques like dilated/atrous convolutions [91, 94] were put forward to avoid downsampling which is one of the issues faced by researchers while detecting small objects using deep layer features. At the very same time, detection of larger objects from shallow layer features is also not optimal without having large scale receptive fields . Due to all these reasons, feature scaling issue has become a fundamental object detection problem. Four major paradigms dealing with multi-scale feature learning problems are image pyramid, prediction pyramid, integrated features and feature pyramid. These are as briefly discussed below:

(i) Image Pyramid: One of the innate ideas to deal with scale feature learning problem is image pyramid wherein input images are resized into multiple scales which are then used to train multiple object detectors responsible for detecting particular range of scales. Similarly during the testing phase images are again resized into different scales followed by many detectors and the detection results thus obtained are merged together. This whole process can become computationally very expensive. Singh et al. [107] carried out on all inclusive experiments on small scale object detection and have argued that learning a single scale detector to detect all scale objects is really a difficult task then learning scale dependent detectors with image pyramids. In their research they proposed a model called scale normalization for image pyramids (SNIP) [107] where many scale dependent object detector are used which are responsible for detecting certain scale objects.

(ii) Integrated Features: It is another multi-feature learning technique wherein a single feature map is constructed using features from multiple layers which is then used to make predictions. By combining semantic rich deep layer and spatially rich shallow layer features, the newly obtained feature map consists of highly rich information which can thus be used to detect objects at varying scales.

(iii) Prediction Pyramid: Coarse and fine-grained features from multiple layers are combined together in the framework called SSD proposed by Liu et al. [47] In SSD object predictions are made using multiple layers where each such layer is dedicated to detect objects of a certain scale. Later many models were proposed based on this principle [108–110].

Multi scale deep convolutional neural networks [108] used deconvolutional layers on different feature maps for improving their resolutions and later these improved feature maps are used for making final predictions. Receptive field blocknet was proposed by Liu et al. [110] for enriching the robustness and receptive fields using a receptive field block, which is used for capturing multi scale features using multiple branches with different convolution kernels, which are then fused together.

(iv) Feature Pyramid: Feature pyramid network was proposed by Lin et al. [48] which combines the advantages of both integrated features as well as the prediction pyramid. Here different scale features with lateral associations are combined in a top-down fashion for building a group of scale invariant feature maps which are then employed to train scale-dependent classifiers. particularly deep semantic rich features are used to reinforce the shallow spatially rich features.

These top-down features are then combined together using element wise summation or concatenation operation along with small convolutions for reducing the overall dimensionality. Significant amount of improvement has been witnessed by FPN for object detection along with other applications and has achieved state-of-the-art results in learning multi-scale features. Many different variants of FPN were later introduced with modified feature pyramid block [111–117]

3.2.2. Contextual reasoning:

In the field of object detection contextual information plays a significant role. Very often objects are likely to appear in specific environments and sometimes they do coexist with other objects. For detecting objects with insufficient features, contextual information can effectively help in improving the detection performance. For understanding the association between objects and their surroundings, context can improve the ability of the detector to understand the scene. As far as traditional object detection techniques are concerned, several efforts exploring context have been made [118].

3.2.3. Deformable feature learning:

For an object detector to be good. It should be absolutely robust to non-rigid twisting of objects. Just before deep learning era, DPM [119] had been successfully employed for object detection and recognition. For enabling object detectors on deep learning to model deformation of objects, many detection frameworks have been proposed which can directly model object parts [94,120–122]. Dai et al. [94] and Zhu et al. [121] have proposed deformable convolutional layers that can automatically learn the supporting location offsets to embed information sampled into standard sampling positions of the feature map.

Next we will discuss about the backbone architecture of object detectors.

4. Backbone architecture for object detection:

Generally to extract the features for performing object detection task where an image is provided as an input and feature map is produced as an output, backbone networks are employed. Majority of backbone architectures used for detection tasks are basically the networks which are used for carrying out the classification task by considering the last fully connected layers. Highly efficient version of this basic classification networks are also available. Lin et al. [48] replaced some of the intermediate layers with some specially designed to better meet specific requirements [20, 123]. For meeting different requirements emphasizing on detection accuracy vs. localization efficiency, researchers are free to make use of much deeper and densely connected backbone architectures like ResNext [124], AmoebaNet [95], ResNet [8]. Apart from all these there are various lightweight backbone networks which when applied to mobile devices can meet out the specific requirements. Some prominent light weight networks like MobileNet [125], ShuffleNet [126], Xception [127] are highly popular. Wang et al. [128] combined PeeleNet with SSD [129] and came out with a state-of-the-art real time object detection system having fast processing speed. For meeting out the requirement of high precision value and more accuracy applications, highly complex backbone networks are required. Whereas well designed architecture which can be adapted to meet the trade-off between speed and accuracy are required to meet the real time requirement for applications involving videos and webcameras. For exploring highly competitive detection accuracy, shallower and sparsely connected backbone networks are replaced by much deeper and densely connected counterparts. As an example He et al. [8] preferred ResNet [5] over VGG [130] to extract rich features for gaining high detection accuracy. As an efficient way to further enhance the network performance, high performance classification setup which can improve detection precision and reduce the complexity of object detection task are under consideration. With the immense advancements in deep learning techniques and unremitting improvement of computation power enormous progress has been made in the field of object detection. R-CNN [6] has successfully demonstrated that adopting convolutional weights from already trained models could come up with affluent semantic information for not only to train the detectors but also to enhance the detection performance. In the last few years, this strategy has been widely accepted and now it has become a norm for most of the object detectors. In this section we will review some basic architectures being extensively used for detection purpose, after providing a basic introduction of deep convolutional neural networks.

4.1. Basic architecture of a CNN:

For visual understanding and machine vision, deep CNN's has proven exceptionally useful [36,131]. A typical deep convolutional neural network as shown in Fig. 13, generally comprises of a series of convolutional layers, pooling layers, non linear activation layers and finally a bunch of fully connected layers. Sample image is fed as an input to the convolutional layer which generates a feature map from it by convolving operation performed by n X n kernels or anchors. The obtained feature map can be considered as a multi-channel sequence, where each channel holds different information related to the input image. Each pixel present in the feature map is represented with the help of a neuron which in turn interacts with a group of adjacent neurons from the previous feature map to produce something known as a receptive field. Just after generating the feature maps a non-linear activation function is used followed by a pooling layer (max or min pooling layer) to widen the receptive field as well as to mitigate the cost involved in computation. Using a sequence of convolutional layers, pooling layers and non linear activation layers, a deep convolutional neural network is formed. Predefined loss functions like stochastic gradient descent [132], Adam [133] etc are employed to optimize the entire network in an end-to-end manner. AlexNet [36] is one such example of deep CNN which comprises of five convolutional layers, 3 max pooling layers and 3 fully connected layers. Further it should be noted that each of the convolution layer is followed by a ReLU [134] which is a non-linear activation layer.

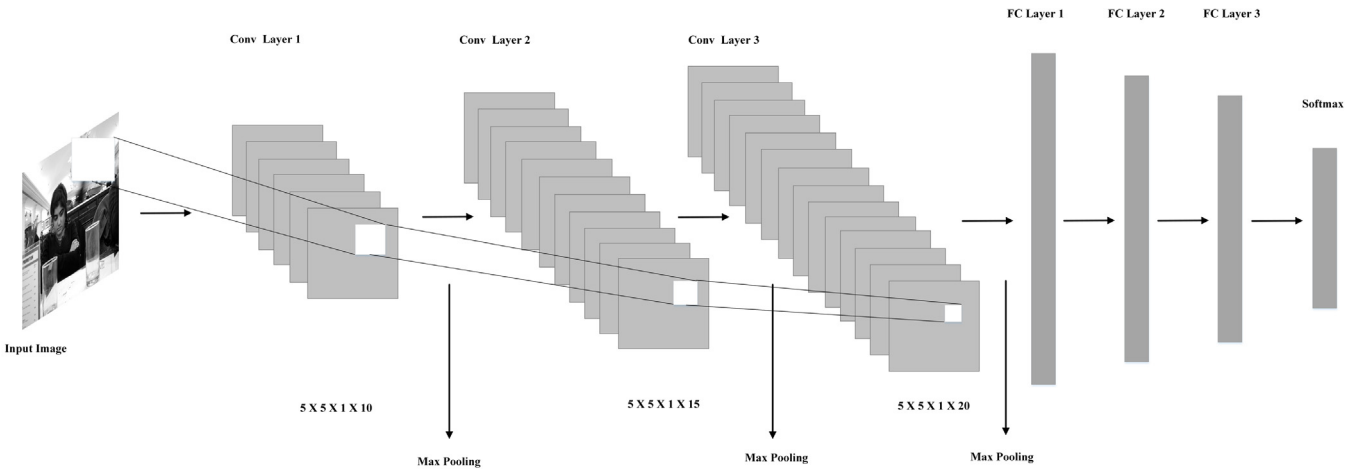


Fig. 13. Basic CNN Architecture.

Table 3
Details of prominent generic object detection architectures.

Architecture	Proposal generation technique	Learning strategy	Loss function	Softmax layer	End-to-end training	Multi-scale input	Platform
RCNN [7]	Selective search	SGD, BP	Hinge Loss	Yes	No	No	Caffe
Fast-RCNN [19]	Selective search	SGD	Hinge Loss	Yes	No	No	Caffe
Faster-RCNN [19]	Region Proposal Network	SGD	Class Log Loss	Yes	Yes	Yes	Caffe
SPP-Net [89]	Edge Boxes	SGD	Hinge Loss	Yes	No	Yes	Caffe
R-FCN [91]	Region Proposal Network	SGD	Class Log Loss	No	Yes	Yes	Caffe
Mask R-CNN [8]	Region Proposal Network	SGD	Class Log Loss and Semantic Sigmoid Loss	Yes	Yes	Yes	TensorFlow/Keras
FPN [48]	Region Proposal Network	Synchronized SGD	Class Log Loss	Yes	Yes	Yes	TensorFlow
SSD [47]	Region Proposal Network	SGD	Class Softmax Loss	No	Yes	No	Caffe
YOLO [20]	Region Proposal Network	SGD	Class sum squared error loss	Yes	Yes	No	DarkNet

4.2. CNN backbone for object detection:

Here we will quickly review some common CNN backbone architectures such as ResNet [5,91], VGG16 [19,21], ResNeXt [83] and Hourglass [85] which are broadly being used for object detection purpose. AN overview of prominent generic object detection architectures as mentioned in [58] is provided in Table 3. VGG16 [130] which was build up based on AlexNet comprises of 5 clusters of convolutional layers and 3 fully connected layers. First two sets consist of two convolutional layers followed by next 3 sets comprising of three convolutional layers. In between each set a Max pooling layer is introduced to diminish the spatial dimensionality. VGG16 architecture has clearly highlighted that, if we increase the depth of the network by hoarding together many convolutional layers, we can increase the expressive potential of the model which further leads to better performance. However this strategy of increasing the number of convolutional layers suffers from a serious drawback of optimizing the entire network in an end-to-end manner. Now keeping this observation in mind, He et al. [5] came up with ResNet [91] which mitigated the drawbacks associated with optimization by establishing shortcut connections where a layer could omit the non-linear makeover and straightaway pass on the values to the next layer. It is as represented below in the form of an equation.

$$y_{k+1} = y_k + t_{k+1}(y_k, \theta) \tag{3}$$

Here y_k is the input feature present in the kth layer and t_{k+1} represents the set of operations (normalization, non-linear transformation, convolutions, activations etc.) carried out on the input feature y_k. $t_{k+1}(y_k, \theta)$ is a residual operation function for y_k input feature. Using this equation any feature map can be regarded as a sum of activations of previous layer and the residual function for considerably reducing the difficulties involved during the training process. Gradients from the deeper layers back to the shallow layers can easily be propagated through the shortcut connection using training setup involving "residual function blocks", depth of the model could be increased from 16 upto 152 thus allowing us to train high capacity network models. In the upcoming year, He et al. [135] came up with a pre-activation alternate of ResNet known as ResNetv2. The experiments carried out clearly showed that ResNetv2 could achieve better performance if batch normalization is done properly [136]. Due to this it becomes possible to successfully train models with more than 1000 layers. Huang et al. concluded that even though ResNet has overcome the shortcomings associated with training using shortcut connection. It did not make use of all the features produced by the preceding layers. Since the features from the shallow layers are missing hence they cannot be directly used for element wise operations. Hence they proposed DenseNet [137] which not only preserved the features from the shallow layers but also improved the flow of information by combining the inputs with the residual product instead of element wise addition.

$$y_{k+1} = y_k \circ t_{k+1}(y_k, \theta) \tag{4}$$

Here $\circ$ denotes the concatenation operation. DenseNet [137] too suffered from various shortcomings as Chen et al. [138] concluded that there is a lot of duplicity involved with a number of extracted features and computation cost involved is also very high. In the upcoming years, the advantages of both ResNet and DenseNet were combined together and a new model called Dual Path Network (DPN) was proposed wherein each individual channel y_k is divided into two segments [y_k^d and y_k^r] here y_k^d was used to perform intense connection computation and element wise summation was carried out using y_k^r, with non-sharable residual learning stems t_{k+1}^d and t_{k+1}^r. The final result thus produced is basically the concatenation of these two branches.

$$y_{k+1} = \left(y_k^r + t_{k+1}^r\left(y_k^r, \theta^r\right)\right) \circ \left(y_k^d \circ t_{k+1}^d\left(y_k^d, \theta^d\right)\right) \tag{5}$$

Xie et al. [124] came up with ResNeXt based on ResNet architecture which not only mitigated the computation and memory cost but also maintained the classification accuracy. ResNeXt espoused group convolution layers [36] which meagerly connects the feature map channels to incarcerate rich semantic features

thereby improving the backbone accuracy while maintaining the computation cost consistent with traditional ResNet. After some time Howard et al. [125] came out with strategy of setting the coordinates equal to the number of feature channels for each feature map and hence proposed a new architecture called MobileNet which not only lessen the cost of computation but also maintained the categorization accuracy without any considerable loss. Also it should be noted that this model was designed for being used on a mobile platform. We have already witnessed the benefits of increasing the model depth. In addition to that, some works discovered the benefits related to learning capabilities by increasing the width of the model. One such work was proposed by Sezegedy et al. called GoogleNet which consists of an inception unit [139] where many convolutional kernels of varying shapes and sizes were applied on a common feature map in a particular layer extracting multi-scale features which are then summed up as an output feature map. Later advanced version of this model was developed by introducing residual units [140]. The popular backbone architectures that we have just reviewed are all designed for the purpose of image classification and have been trained on Image Net which can be adapted to object detection applications. But it should be noted that these pre-trained models cannot be directly used for object detection because of highly probable divergence between classification and detection tasks. This divergence occurs because classification needs huge receptive fields for maintaining lower spatial invariance, while detection requires small receptive fields with high resolution spatial invariance. Moreover classification makes prediction from a single feature map whereas detection needs feature maps with multiple representations for detecting multi-scale objects. Keeping the above points in mind, Li et al. proposed DetNet [123] specifically developed for detection purposes. Using expanded convolutions to increase the receptive fields, DetNet [123] preserves high resolution feature maps for prediction. Moreover extracting multi-scale feature maps assists DetNet to detect varying shape objects easily. Another architecture which was not typically designed for classification purposes is Hourglass Network [141]. Hourglass which is a fully convolutional setup with a series of hourglass modules originally emerged in human pose recognition problem [141]. The beauty of hourglass module is, it first downsamples the given input image using a series of convolutional layers and pooling operations and then upsampling the generated feature maps using deconvolutional operations. Here also skip connections are used to avoid loss of information during downsampling. Since it can capture both local and global information, hence it is widely being used in state-of-the-art object detection systems [84–86]. In the next section we will review some learning strategies being used for object detection.

5. Learning strategies for object detection

Localization and classification jobs related to image classification and object detection entails optimization which makes it even harder to train vigorous detectors. Apart from that there are a number of concerns such as localization, imbalance sampling etc. which needs to be addressed. So there arises a strong need to design and develop ground breaking learning strategies for training capable and competent object detectors. In this section, a comprehensive review of state-of-the-art learning strategies for object detection is provided. Learning strategies for training the object detectors are different from their counterparts used during the testing phase. So we will review them separately.

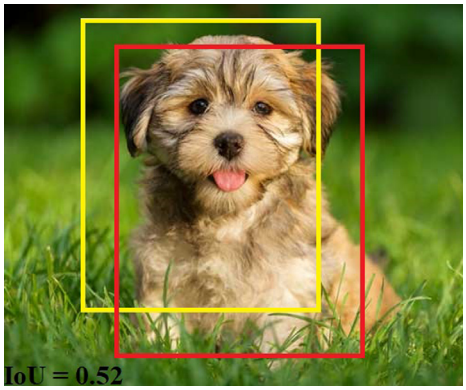


Fig. 14. Failure case of detection in high IoU threshold. Red box is groundtruth and yellow box is prediction.

5.1. Learning strategies for training:

Here we review some learning strategies extensively used for the training purpose.

(i) Data Intensification: It is one of the most important learning strategies for almost all deep learning systems, since they are considered as data-famished and therefore large amounts of training data will lead to better outputs. In object detection for generating training scraps with large number of visual features, increasing the training data, horizontal flips are used during training of Faster R-CNN object detector [19]. In one-stage detectors a highly intense data intensification strategy (including expansion, color jittering, rotation, cropping etc.) [47,108,142] is used for improving the detection accuracy.

(ii) Refining Localization: An expected outcome of an object detector is to provide a rigid bounding box or a stiff mask for each object in the image. For achieving this many measures have been proposed which refines the bounding box coordinates and helps in improving the localization accuracy. Exact localization is troublesome because proposal predictions are normally based on most discriminative object parts and not on the region encompassing the object. In some scenarios, the detection algorithms are required to make high quality predictions (with high IoU threshold value) see Fig. 14 for an explanation as how a detector may fail in a high IoU threshold setup. A normal approach for refining localization is to provide high quality region proposals. Here we review other methods for refining the localization. In fast R-CNN smooth L1 regressors are learned from end-to-end as shown below:

$$L_{\text{reg}}\left(p^c, g\right) = \sum_{i\in\{x,y,w,h\}} \text{L1Smooth}\left(p_i^c - g_i\right) \tag{6}$$

$$\text{L1Smooth}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \tag{7}$$

Here the prediction offset is denoted by $p^c = \left(p_x^c, p_y^c, p_w^c, p_h^c\right)$ for each of the predefined target class and "g" here denotes the ground truth of bounding boxes $\left(v = \left(g_x, g_y, g_w, g_h\right)\right)$, x,y,w, h represents center coordinates, width and height of bounding box respectively. In R-CNN architecture L2 auxiliary box regressors are used for localization refinement. Apart from these, some techniques even used supplementary setups to further process the localization. A looping bounding box regression technique was proposed by Gidaris et al. [143], where R-CNN backbone was

used for further processing the defined prediction offset multiple times. Some years later, another approach called LocNet was proposed by Gidaris et al. [144] which focused on bounding box coordinate distribution for refining the predictions. Zagoruyko at al. [145] introduced a multipath network where in a cascade of categorizers were developed and later optimized using predefined integral loss based on various quality parameters. Each ensemble classifier was individually optimized with a predefined IoU threshold value. The final results of predictions thus made is a combination of outputs from each of these classifiers. Tychen et al. derived a bounding box regression loss on a group of IoU values so as to maximize the IoU prediction offsets related to the objects. They also proposed Fitness-NMS [146] that learned fresh fitness score function of IoU between objects and their proposals. Taking inspiration from cornerNet [84] and DeNet [103] Lu et al. [147] came up with grid R-CNN which substitutes bounding box regressor with keypoint corner based system.

(iii) Learning by Cascades: It is defined as a coarse-to-fine learning strategy which gathers data from the result of given classifiers and develop a stronger one in a gushed manner. Coming to deep learning based detection mechanisms, CRAFT (Cascade Region Proposal Network and Fast-RCNN) was proposed by Yang et al. [148] which employs cascade learning techniques to learn RPN and region proposals. CRAFTS learn a regular RPN followed by a dual class Fast R-CNN which discards greater part of easy negatives. The remaining samples were then used to build the cascade region classifier. For classifying multiscale objects into one of the predefined target classes, a layer wise cascade classifier was introduced by Yang et al. Chang et al. [149] observed Faster-RCNN and find out that although localizing objects has improved a lot but still there are many classification errors that can be attributed to joint multitask optimization for regression and classification, they also noted that because of larger receptive fields of Faster R-CNN, noise is introduced in the detection process. So, to overcome these shortcomings a cascade model was build which was based on Faster R-CNN and R-CNN both. An initial set of prediction offsets are produced from a highly trained Faster R-CNN which are then used to refine the offsets using an R-CNN.

(iv) Adversarial Learning: This learning strategy has shown a remarkable advancement in generative models. The most famous and recent examples of adversarial learning is generative adversarial networks (GAN) [150] where in a generator and a discriminator are working together in a competitive manner. Generator here tries to model the data by producing faux images and then using them to puzzle the discriminator, whereas discriminator battles with the generator to identify the real ones from the faux images. GAN and its variants [151–153] have become largely popular and therefore can also be used for object detection. A new setup for detecting the smaller objects was proposed by Li et al. [154] called perceptual GAN where the generator module learned high-decree features from small objects using "adversarial learning technique" and discriminator tried to identify the real resolution features for small objects. A Faster R-CNN model which was trained using GAN examples was proposed by Wang et al. [155] where a learned mask was produced on region features followed by region classifiers due to which the detector becomes more robust after receiving more adversarial examples.

(v) Learning from scratch: Majority of object detectors profoundly relying on pretrained imageNet models suffer from poor performance, because of the prejudice of loss function and data distribution existing between classification and detection tasks. This problem can be mitigated to an extent by finetuning on the detection task but cannot be completely avoided. Apart from that using a pretrained classification model for detection purposes in a new domain can lead to further complications. Because of these reasons there is a strong need to train the detection models from scratch instead on only using pretrained ImageNet models. The primary difficulty one may face in training a model from scratch is the non availability of enough data for object detection which can further cause overfitting to occur. Moreover object detection models also need bounding box level annotations and therefore annotating a huge dataset for object detection is a time consuming and costly affair. There are various state-of-the-art works where object detectors trained from scratch was proposed Shen at al. [109] called DSOD (Deeply supervised object detectors) where the authors have argued that optimization difficulties can be mitigated significantly if densely connected network models with intense supervision are used. Shen et al. [156] came out with a gated recurrent feature pyramid which vigorously attuned supervision intensities of transitional layers for objects with various scales. This method proved to be more powerful as compared to traditional DSOD. Later He et al. [157] authenticated the difficulty associated with training detectors from scratch on MSCOCO dataset and concluded that the vanilla detectors can obtain better performance with atleast 10 K annotated images and proved that no specific model is required for training from scratch.

(vi) Imbalanced Sampling: In the field of object detection unevenness of positive and negative samples is a serious issue. This is because majority of proposals which are considered as region of interests are basically the background images and few among them are actually the objects we are looking for. Due to this imbalancing issue, two problems may arise, class imbalance and difficulty imbalance. Class imbalance means that majority of the generated proposals are just background images and very few among them actually contains the objects whereas difficulty imbalance means that it is much easier to classify the proposals as background while objects get hard to classify. Various techniques have been developed to get rid of this problem. Two stage detectors like R-CNN and Fast-RCNN discards a number of negative samples and upkeep 2000 proposals for further classification. In Fast-R-CNN [19] the ratio of positive to negative samples is fixed as 1:3 in each of the minibatch to overcome the difficulty of class imbalancing. For addressing class imbalance, random sampling can be used. To address the problem of difficulty imbalance, carefully fabricated loss functions can be used. For the purpose of object detection, a multi class classifier is trained over C+1 classes. If $p = (p_0, p_1, p_2, p_3 \ldots\ldots p_C)$ is the discrete probability distribution of output over C+1 classes and "v" is the ground truth class then Loss function is defined as:

$$Loss_{\text{cls}}(p, v) = -\log p_v \tag{8}$$

Instead of rejecting all the easy samples, a novel focal loss [83] was proposed by Liu et al. where an importance factor was assigned to each sample with respect to its loss value as shown.

$$Loss_{\text{FL}} = -\alpha\,(1 - p_v)^{\gamma}\log(p_v) \tag{9}$$

Here α and γ are the parameters to control the weight importance.

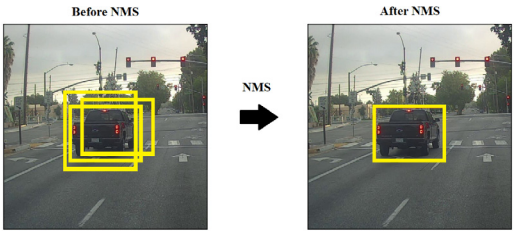


Fig. 15. Example to show how duplicate predictions are eliminated by NMS operation.

(vii) Distillation of Knowledge: Distillation of knowledge is defined as a training technique which extracts the knowledge from a cascade of models and imparts it into a single model using teacher–student training scheme. This learning technique was first of all used for classification of images [158]. Distillation of knowledge technique was also used in some object detection works [149,159] to improve the performance. A light weighted optimized object detector was proposed by Li et al. [159] and achieved comparable detection accuracy by using knowledge distillation strategy while maintaining higher inference speeds. Another optimized framework based on teacher–student training scheme was proposed by Cheng et al. [149] which showed an improvement in accuracy compared to single model optimization strategy.

5.2. Testing stage:

The algorithms and techniques used for object detection make intense predictions which cannot be used directly for evaluation due to large number of duplicates and moreover other learning strategies are needed to achieve better detection results. In this section some of the popular strategies being used during the testing phase are reviewed.

(i) Removal of Duplicates: Object detection techniques are known to make an intense set of predictions with many duplicates and therefore NMS appears to be a fundamental part of object detection to get rid of duplicate false positives. For single stage object detectors the proposals enclosing the same object may have similar confidence scores leading to false positives. For double-stage detectors the bounding box regressor can be regarded as a cause of generating the false positives. Therefore in both these cases NMS is needed to remove the duplicate predictions. Typically for each of the target class the generated prediction boxes are arranged in order of their confidence scores and the box with the highest score is henceforth selected. Let us denote this box with M. now IoU of all other boxes with M is compared and if the IoU value thus obtained is larger than some already defined threshold Ω, the boxes will be discarded. This whole process is repeated for all the predictions made. A typical example is given in Fig. 15. In addition to that the confidence of box B which overlaps with M larger than Ω is set to zero as shown below:

$$\text{Score}_{B} = \begin{cases} \text{Score}_{B} & \text{IoU}(B, M) < \Omega_{\text{test}} \\ 0 & \text{IoU}(B, M) \geq \Omega_{\text{test}} \end{cases} \tag{10}$$

However if an object simply lies within the threshold Ω of M, it will result in a missing prediction. To address this issue Navanneth et al. [160] proposed Soft-NMS technique where instead of directly discarding the prediction B, this new technique festered the confidence score of B as a linear function Func of its overlap with M as given below:

$$\text{Score}_{B} = \begin{cases} \text{Score}_{B} & \text{IoU}(B, M) < \Omega_{\text{test}} \\ F(\text{IoU}(B, M)) & \text{IoU}(B, M) \geq \Omega_{\text{test}} \end{cases} \tag{11}$$

Sift-NMS avoided discarding prediction of grouped objects and achieved a remarkable improvement in many common benchmarks.

(ii) Model Acceleration: For applying object detection in real world the algorithms and techniques are required to work efficiently. No matter current object detectors [5,161] can attain remarkable results on publically available datasets, but due to their inference alacrities, it is very difficult to apply them into real world applications. So here we will review some techniques on accelerating the object detectors. As we have already seen that two-stage detectors are usually slower compared to single-stage detectors, because of two stages involved, one which is used for generating the proposals and other for region classification. For sharing the cost of computations, R-FCN [91] extracts position sensitive features from spatially-sensitive feature maps. However, the number of channels involving spatially sensitive feature maps increased extensively with a large number of object categories. To overcome this problem of increased number of channels. Li et al. [162] developed Light-Headed R-CNN, which remarkably mitigated the number of channels in the final feature map thereby reducing the computational cost and increasing the inference speeds. A significant idea to accelerate the speed of the detection models is to use efficient backbone networks in place of detection architectures, e.g.: MobileNet [125,163] can be considered as an efficient model which was also adopted in many works [164,165]. Another approach to accelerate the model is to optimize them offline using model compression and quantization [166–173] on the learned setups.

5.3. Other latest learning strategies:

Image pyramids [5,174] which is a vividly used approach for improving the detection results can be used, this is because of its hierarchical nature which helps to make predictions at different object scales and finally merging the results to make a final call on prediction. Horizontal flipping [8,174] was also employed extensively during the testing phase and has achieved improved performance. All these learning strategies largely increased the detector capacities to handle multi-scale objects and hence are widely used in many object detection techniques despite of increasing the computation cost.

6. Popular datasets and metrics used for object detection

By object detection, we mean that whether an object belongs to a predefined class or not, if it belongs to a class, we need to identify and locate it in the given image. Localization of an object in an image is generally represented using a bounding box. For carrying out a standard comparison between different algorithmic approaches and defined goals for solutions, exigent and challenging datasets which are considered significant are being used in many different research areas. Earlier face detection techniques rely on various adhoc datasets, but in the later years, many state-of-the-art face detection datasets were created. Apart from that several datasets have been created for the problem of pedestrian detection, face detection etc. generic object detection datasets like PASCAL VOC [33], ImageNet [175], MSCOCO [92] are widely used object detection benchmarks. In this section we will review some most popular datasets along with their performance evaluation.

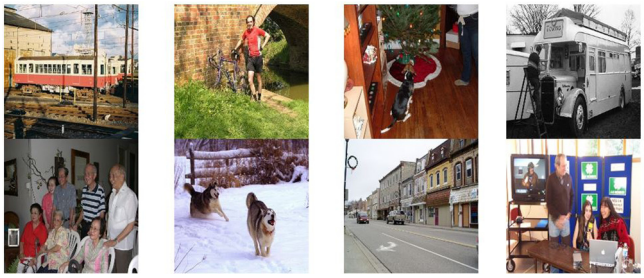


Fig. 16. Sample images from PASCAL VOC 2007 dataset.

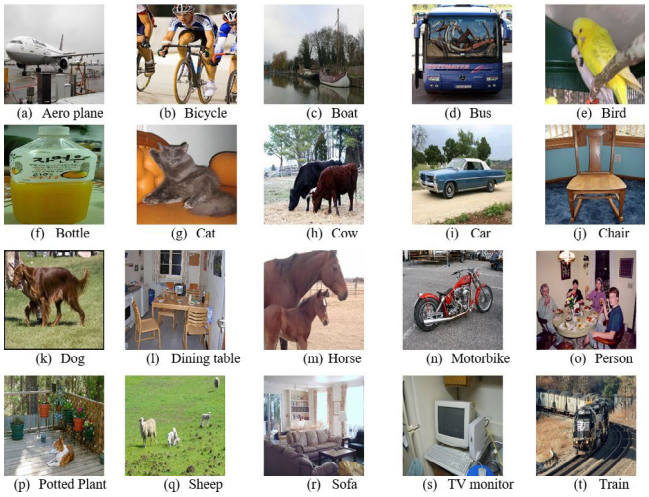


Fig. 17. Sample images from PASCAL VOC 2012 dataset.

Fig. 18. Sample annotated images from MS-COCO dataset.

6.0.1. PASCAL VOC dataset:

For detecting normal object classes, a multi-year attempt from the year 2005 to 2012 was dedicated for the creation of a sequence of benchmark datasets which were vividly used. The PASCAL VOC datasets [33] consists of 20 visual object classes extended over 11000 images. These 20 classes can be grouped into four main branches — animals, vehicles, people and domestic items. In addition to that the classes of objects which are semantically similar like trucks vs. buses add up to increased difficulty levels for detection purposes. Some examples from VOC 2008 and VOC 2012 datasets are given in Fig. 16 & 17 respectively. For the evaluation of classification and detection performance interpolated average precision (Salton and McGill 1986) was used as a criterion for PASCAL datasets. It was used to reprimand the detection techniques for absent object instances, for false positive detections and also for duplicate detections of a single object instance.

$$\text{Recall}(t) = \frac{\sum_{ij} 1\left[s_{ij} \geq t\right] z_{ij}}{N} \tag{12}$$

$$\text{Precision}(t) = \frac{\sum_{ij} 1\left[s_{ij} \geq t\right] z_{ij}}{\sum_{ij} 1\left[s_{ij} \geq t\right]} \tag{13}$$

Here "t" is the threshold value to evaluate the IoU between ground truth and prediction made. In VOC datasets "t" is set to 0.5. "N" is the total number of predicted bounding boxes, "i" is the index of the *ith* image and "j" is the index of the *jth* object. Precision/recall curve is computed for a given detection task along with different object categories where recall is the fraction of all the positive samples above a given threshold value and precision defined the fraction of samples above the defined threshold which are from the positive class. To obtain the ultimate results, mean Average Precision (mAP) is computed across all the object categories. Comparative results of some state-of-the-art detection models on VOC 2007 and VOC 2012 Test Sets are summarized in Tables 4 and 5 respectively.

6.0.2. MS-COCO dataset:

MSCOCO is expanded as Microsoft Common Objects in context dataset [92]. It consists of 91 object categories found in everyday life in their natural environment and 82 of these object categories have more than 5000 instances with labels. As a total this dataset has 2,500,000 labeled instances spread over 328,000 images. With respect to another popular ImageNet dataset [186]. MS-COCO has less number of class objects but has more number of instances per object category and compared to PASCAL-VOC dataset, MS-COCO has a significantly much larger number of instances per object category typically exceeding 27 K on an average. Furthermore MSCOCO dataset consists of 3.5 object classes per image as compared to ImageNet which has 1.7 classes per image on an average and for PASCAL VOC this figure stands at 1.4. As we already know that objects which are tiny in size often requires more of contextual reasoning for their recognition and images from MSCOCO dataset are super rich in this contextual information. The largest class in MSCOCO dataset is persons having almost 800,000 instances where as the smallest one is the hair dryer with roughly 600 instances from the entire dataset. Some typical annotated images from MSCOCO are given in Fig. 18. The metric used to measure the detection performance is similar to what we have seen for PASCAL VOC. The threshold value here can be chosen from a range of values [0.5,0.95] with an interval of 0.05 to measure the mean Average Precision (mAP). Furthermore to measure the performance of the object detector to detect different sized objects, special average precision for small, medium and large objects are computed separately. Comparison of detection performance of some state-of-the-art techniques on MSCOCO test-dev dataset are summarized in Table 6.

6.0.3. ImageNet Dataset:

It is one of the most important dataset with near about 200 object classes [175,196]. The ISLVRC task of object detection assesses the capability of an algorithm to localize and recognize all the possible instances of target object categories present in an image. The metric used to measure the performance of an algorithm using this dataset is called loosen threshold and mathematically it is as given below:

$$t = \min\left(0.5, \frac{wh}{(w+10)(h+10)}\right) \tag{14}$$

Here "w" and "h" are the width and height of the ground truth bounding box. The beauty of this threshold is that it allows the predicted bounding box to extend atmost 5 pixels on an average in each possible direction around the object.

Table 4
Performance analysis of some state-of-the-art in terms of mean Average Precision (mAP) using PASCAL VOC 2007 test dataset.

Models	Aeroplane	Bicycle	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Mbike	Person	Plant	Sheep	Sofa	Train	TV	mAP
R-CNN (Alex) [7]	68.1	72.8	56.8	43.0	36.8	66.3	74.2	67.6	34.4	63.5	54.5	61.2	69.1	68.6	58.7	33.4	62.9	51.1	62.5	68.6	**58.5**
R-CNN (VGG) [19]	73.4	77.0	63.4	45.4	44.6	75.1	78.1	79.8	40.5	73.7	62.2	79.4	78.1	73.1	64.2	35.6	66.8	67.2	70.4	71.1	**66.0**
SPP-Net [89]	68.5	71.7	58.7	41.9	42.5	67.7	72.1	73.8	34.7	67.0	63.4	66.0	72.5	71.3	58.9	32.8	60.9	56.1	67.9	68.8	**60.9**
OHEM +Fast-RCNN [176]	80.6	85.7	79.8	69.9	60.8	88.3	87.9	89.6	59.7	85.1	76.5	87.1	87.3	82.4	78.8	53.7	80.5	78.7	84.5	80.7	**78.9**
HyperNet VGG [32]	84.2	78.5	73.6	55.6	53.7	78.7	79.8	87.7	49.6	74.9	52.1	86	81.7	83.3	81.8	48.6	73.5	59.4	79.9	65.7	**71.415**
Faster R-CNN [21]	70.0	80.6	70.1	57.3	49.9	78.2	80.4	82.0	52.2	75.3	67.2	80.3	79.8	75.0	76.3	39.1	68.3	67.3	81.1	67.6	**69.9**
GCNN [177]	68.3	77.3	68.5	52.4	38.6	78.5	79.5	81.0	47.1	73.6	64.5	77.2	80.5	75.8	66.6	34.3	65.2	64.4	75.6	66.4	**66.8**
Bayes [178]	74.1	83.2	67.0	50.8	51.6	76.2	81.4	77.2	48.1	78.9	65.6	77.3	78.4	75.1	70.1	41.4	69.6	60.8	70.2	73.7	**68.5**
SDP +CRC [179]	76.1	79.4	68.2	52.6	46.0	78.4	78.4	81.0	46.7	73.5	65.3	78.6	81.0	76.7	77.3	39.0	65.1	67.2	77.5	70.3	**68.9**
SubCNN [27]	70.2	80.5	69.5	60.3	47.9	79.0	78.7	84.2	48.5	73.9	63.0	82.7	80.6	76.0	70.2	38.2	62.4	67.7	77.7	60.5	**68.5**
StuffNet30 [180]	72.6	81.7	70.6	60.5	53.0	81.5	83.7	83.9	52.2	78.9	70.7	85.0	85.7	77.0	78.7	42.2	73.6	69.2	79.2	73.8	**72.7**
NOC [181]	76.3	81.4	74.4	61.7	60.8	84.7	78.2	82.9	53.0	79.2	69.2	83.2	83.2	78.5	68.0	45.0	71.6	76.7	82.2	75.7	**73.3**
MR-CNN + S-CNN [182]	80.3	84.1	78.5	70.8	68.5	88.0	85.9	87.8	60.3	85.2	73.7	87.2	86.5	85.0	76.4	48.5	76.3	75.5	85.0	81.0	**78.2**
HyperNet [183]	77.4	83.3	75.0	69.1	62.4	83.1	87.4	87.4	57.1	79.8	71.4	85.1	85.1	80.0	79.1	51.2	79.1	75.7	80.9	76.5	**76.3**
SSD300 [47]	80.9	86.3	79.0	76.2	57.6	87.3	88.2	88.6	60.5	85.4	76.7	87.5	89.2	84.5	81.4	55.0	81.9	81.5	85.9	78.9	**79.6**
SSD512 [47]	86.6	88.3	82.4	76.0	66.3	88.6	88.9	89.1	65.1	88.4	73.6	86.5	88.9	85.3	84.6	59.1	85.0	80.4	87.4	81.2	**81.6**

Table 5
Performance analysis of some state-of-the-art in terms of mean Average Precision (mAP) using PASCAL VOC 2012 test dataset.

Models	Aeroplane	Bicycle	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Mbike	Person	Plant	Sheep	Sofa	Train	TV	mAP
R-CNN (Alex) [7]	71.8	65.8	52.0	34.1	32.6	59.6	60.0	69.8	27.6	52.0	41.7	69.6	61.3	68.3	57.8	29.6	57.8	40.9	59.3	54.1	**53.3**
R-CNN (VGG) [7]	79.6	72.7	61.9	41.2	41.9	65.9	66.4	84.6	38.5	67.2	46.7	82.0	74.8	76.0	65.2	35.6	65.4	54.2	67.4	60.3	**62.4**
Fast-RCNN [19]	82.3	78.4	70.8	52.3	38.7	77.8	71.6	89.3	44.2	73.0	55.0	87.5	80.5	80.8	72.0	35.1	68.3	65.7	80.4	64.2	**68.4**
OHEM +Fast-RCNN [184]	90.1	87.4	79.9	65.8	66.3	86.1	85.0	92.9	62.4	83.4	69.5	90.6	88.9	88.9	83.6	59.0	82.0	74.7	88.2	77.3	**80.1**
Faster R-CNN [21]	84.9	79.8	74.3	53.9	49.8	77.5	75.9	88.5	45.6	77.1	55.3	86.9	81.7	80.9	79.6	40.1	72.6	60.9	81.2	61.5	**70.4**
StuffNet30 [180]	83.0	76.9	71.2	51.6	50.1	76.4	75.7	87.8	48.3	74.8	55.7	85.7	81.2	80.3	79.5	44.2	71.8	61.0	78.5	65.4	**70.0**
NOC [181]	82.8	79.0	71.6	52.3	53.7	74.1	69.0	84.9	46.9	74.3	53.1	85.0	81.3	79.5	72.2	38.9	72.4	59.5	76.7	68.1	**68.8**
MR-CNN + S-CNN [182]	85.5	82.9	76.6	57.8	62.7	79.4	77.2	86.6	55.0	79.1	62.2	87.0	83.4	84.7	78.9	45.3	73.4	65.8	80.3	74.0	**73.9**
HyperNet [183]	84.2	78.5	73.6	55.6	53.7	78.7	79.8	87.7	49.6	74.9	52.1	86.0	81.7	83.3	81.8	48.6	73.5	59.4	79.9	65.7	**71.4**
ION [185]	87.5	84.7	76.8	63.8	58.3	82.6	79.0	90.9	57.8	82.0	64.7	88.9	86.5	84.7	82.3	51.4	78.2	69.2	85.2	73.5	**76.4**
SSD512 [47]	91.4	88.6	82.6	71.4	63.1	87.4	88.1	93.9	66.9	86.6	66.3	92.0	91.7	90.8	88.5	60.9	87.0	75.4	90.2	80.4	**82.2**
SSD300 [47]	91.0	86.0	78.1	65.0	55.4	84.9	84.0	93.4	62.1	83.6	67.3	91.3	88.9	88.6	85.6	54.7	83.8	77.3	88.3	76.5	**79.3**
YOLO [20]	77.0	67.2	57.7	38.3	22.7	68.3	55.9	81.4	36.2	60.8	48.5	77.2	72.3	71.3	63.5	28.9	52.2	54.8	73.9	50.8	**57.9**
YOLO + Fast R-CNN [20]	83.4	78.5	73.5	55.8	43.4	79.1	73.1	89.4	49.4	75.5	57.0	87.5	80.9	81.0	74.7	41.8	71.5	68.5	82.1	67.2	**70.7**
YOLOv2 [20]	88.8	87.0	77.8	64.9	51.8	85.2	79.3	93.1	64.4	81.4	70.2	91.3	88.1	87.2	81.0	57.7	78.1	71.0	88.5	76.8	**78.2**
R-FCN [19]	92.3	89.9	86.7	74.7	75.2	86.7	89.0	95.8	70.2	90.4	66.5	95.0	93.2	92.1	91.1	71.0	89.7	76.0	92.0	83.4	**85.0**

Table 6
Comparison of detection performance of some state-of-the-art techniques on MSCOCO test-dev dataset.

Model	Backbone architecture	Year of introduction	AP	AP_50	AP_75	AP_S	AP_M	AP_L
Single-Stage-Object-Detectors								
SSD512 [16]	VGG-16	2016	28.8	48.5	30.3	10.9	31.8	43.5
SSD513 [113]	ResNet-101	2017	31.2	50.4	33.3	10.2	34.5	49.8
DSSD513 [113]	ResNet-101	2017	33.2	53.3	35.2	13.0	35.4	51.1
STDN513 [187]	DenseNet-169	2018	31.8	51.0	33.6	14.4	36.1	43.4
CornerNet511 [84]	Hourglass-169	2018	40.5	56.5	43.1	19.4	42.7	53.9
CornerNet511 [86]	Hourglass-169	2019	44.9	62.4	48.1	25.6	47.4	57.4
GHM SSD [188]	ResNeXt-101	2018	41.6	62.8	44.2	22.3	45.1	55.3
FPN-Reconfig [116]	ResNeXt-101	2018	34.6	54.3	37.3	NA	NA	NA
FCOS [189]	ResNeXt-101	2019	42.1	62.1	45.2	25.6	44.9	52.0
FSAF [104]	ResNeXt-101	2019	42.9	63.8	46.3	26.6	46.2	52.7
ExtremeNet [190]	Hourglass-104	2019	40.2	55.5	43.2	20.4	43.2	53.1
M2Det800 [117]	VGG-16	2019	41.0	59.7	45.0	22.1	46.5	53.8
RefineDet512 [174]	ResNet-101	2018	36.4	57.5	39.5	16.6	39.9	51.4
YOLOv2 [46]	DarkNet-19	2017	21.6	44.0	19.2	5.0	22.4	35.5
Two-Stage-Object-Detectors								
Fast R-CNN [19]	VGG-16	2015	19.7	35.9	NA	NA	NA	NA
Faster R-CNN [21]	VGG-16	2015	21.9	42.7	NA	NA	NA	NA
Faster R-CNN w FPN [48]	ResNet-101	2016	36.2	59.1	39.0	18.2	39.0	48.2
Faster R-CNN by G-RMI [161]	Inception-ResNet-v2	2017	34.7	55.5	36.7	13.5	38.1	52.0
OHEM [184]	VGG-16	2016	22.6	42.5	22.2	5.0	23.7	37.9
ION [185]	VGG-16	2016	23.6	43.2	23.6	6.4	24.1	38.3
R-FCN [91]	ResNet-101	2016	29.9	51.9	NA	10.8	32.8	45.0
CoupleNet [191]	ResNet-101	2017	34.4	54.8	37.2	13.4	38.1	50.8
Deformable R-FCN [91]	Aligned-Inception-ResNet	2017	37.5	58.0	40.8	19.4	40.1	52.5
DeNet-101 [103]	ResNet-101	2017	33.8	53.4	36.1	12.3	36.1	50.8
Mask-RCNN [8]	ResNeXt-101	2017	39.8	62.3	43.4	22.1	43.2	51.2
Fitness-NMS [146]	ResNet-101	2017	41.8	60.9	44.9	21.5	45.0	57.5
Relation Net [93]	ResNet-101	2018	39.0	58.6	42.9	NA	NA	NA
DeepRegionlets [192]	ResNet-101	2018	39.3	59.8	NA	21.7	43.7	50.9
C-Mask RCNN [193]	ResNet-101	2018	42.0	62.9	46.4	23.4	44.7	53.8
DCN + R-CNN [149]	ResNet-101 + ResNet-152	2018	42.6	65.3	46.5	26.4	46.1	56.4
Cascade R-CNN [194]	ResNet-101	2018	42.8	62.1	46.3	23.7	45.5	55.2
Grid R-CNN [147]	ResNeXt-101	2019	43.2	63.0	46.6	25.1	46.5	55.2
DCN-v2 [121]	ResNet-101	2019	44.8	66.3	48.8	24.4	48.1	59.6
TridentNet [195]	ResNet-101	2019	42.7	63.6	46.5	23.9	46.6	56.6

6.0.4. OpenImages dataset:

This dataset consists of 1.9 million images containing approximately 15 million objects spread over 600 categories. The images in this dataset are already annotated with image level labels, object segmentation masks, visual relationships and object bounding boxes [197]. Some important aspects of this dataset are, firstly the bounding boxes are largely drawn manually. Secondly the images are really assorted and consists of complex scenes

Table 7
Statistics of some popular generic object detection datasets.

Dataset	Training subset		Validation subset		Testing subset		Trainval subset	
	Images	Objects	Images	Objects	Images	Objects	Images	Objects
Pascal VOC-2007	2501	6301	2510	6307	4952	14976	5011	12608
Pascal VOC-2007	5717	13609	5823	13841	10991	NA	11540	27450
ILSVRC-2014	456567	478807	20121	55502	40152	NA	476688	534309
ILSVRC-2014	456567	478807	20121	55502	65500	NA	476688	534309
MS-COCO-2015	82783	604907	40504	291875	81434	NA	123287	896782
MS-COCO-2018	118287	860001	5000	36781	40670	NA	123287	896782

with many objects and thirdly, the dataset often offers visual relationship annotations. Since it is a very diverse dataset, hence different metrics are been used to measure the performance of an algorithm using it. Firstly, the unannotated class categories are discarded to avoid the wrong number of false negatives. Secondly, if an object belongs to a predefined target class, then the detection model should come up with detection results for each of the relevant class. Thirdly, if a detection occurs inside a group of boxes and if the intersection of the detection and the box partitioned by the area of detected region exceeds threshold of 0.5 then it is counted as a true positive.

6.0.5. VIS Drone 2018 dataset:

In the year 2018, a new dataset which comprises of videos and images captured using a drone was developed and is called VIS Drone 2018 [198]. It is a large scale visual object detection and tracking dataset aims at proceeding with visual understanding on a drone based platform. The images and videos in this dataset are captured over several urban areas of 14 different Chinese cities. Typically speaking it comprises of 263 video sequences and 10209 images with affluent annotations like ground truth bounding boxes, object category, truncation ratios etc. it comprises of more than 2.5 million annotated instances. This dataset adopts MS-COCO metric for evaluating the performance of different object detection algorithms.

6.0.6. LVIS [1] :

It is a newly collected benchmark comprising of 164000 images spread over 1000 plus object categories. It is a pretty new dataset without any pre-existing results as of now.

The statistics of some of these well known generic object detection datasets are provided in Table 7.

6.1. Pedestrian detection dataset:

In this subsection, we will review most famous and commonly used datasets for pedestrian object detection along with the metrics beings used for evaluation purposes.

(i) Caltech [52]: It is one of the most popular and tricky datasets used for pedestrian detection task. It comprises of approximately 10 h of VGA video sequence recorded by a vehicle bound camera driving through the streets of Los Angeles metropolitan city. The training and testing set comprises of 42782 and 4024 video frames respectively.
(ii) ETH [199]: This dataset is basically used as a testing set to assess the performance of detection models trained on Citypersons dataset. It comprises of 1804 frames spread over three video sequences.
(iii) INRIA [26]: It comprises of HD images of pedestrians mostly collected from vacation resorts comprising of 2120 images which are further divided into two sets consisting of 1832 images for training purpose and remaining 288 images for carrying out the testing task.
(iv) KITTI [200]: It consists of 7481 labeled HD images for training and 7518 images for testing purpose. The person category in this dataset is subdivided into two sub categories of pedestrian and cyclist. The models trained on it are evaluated using three metrics with difference in the min. bounding box height and max. occlusion level.

For INRIA [26] and ETH [199] datasets, log average miss rate metric is used to evaluate the performance of detectors and for KITTI, mean Average Precision with IoU threshold of 0.5 is used for evaluation purposes. Comparative analysis of pedestrian detection datasets is given in Table 8 followed by comparison of person detection benchmarks which are covered in Table 9. Table Information from PIOTR ET AL, IEEE TPAMI2012 [52]. Table Information from PIOTR ET AL, IEEE TPAMI2012 [52].

6.2. Face detection datasets:

Here many popular face detection datasets are reviewed along with their evaluation metrics.

(i) PASCAL FACE [33]: This dataset comprises of 1335 labeled faces from 851 different images. It is collected from PASCAL person layout set and is commonly used as a test set.
(ii) FDDB [205]: It stands for Face detection datasets and benchmark. It is a widely used dataset comprising of 5171 faces spread over 2845 images. It is also used as a test set for evaluating the performance of face detection models.
(iii) Wider-Face [206]: It is a very large dataset comprising of 3203 images with almost 400 K faces of varied range and scales. It is divided into three subsets, 40% for training, 10% for validating and remaining 50% is kept for testing purposes. The annotated training and validation datasets are available online at http://shuoyang1213.me/WIDERFACE/.

For Wider-Face and PASCAL-Face mAP with IoU threshold of 0.5 is used as a metric to evaluate the performance of face detectors and for evaluating FDDB dataset two annotation types (bound box level and eclipse level) are mostly used. The details of evaluation metrics (discussed above) are summarized in Table 10.

7. Applications of object detection

Object detection finds its applications in many fields like medical, military, security etc to assist people. Here in this section, some prominent applications of object detectors are reviewed.

7.1. Face detection:

Aim of face detection is to detect all the faces present in an image. Face detection is still considered as a difficult task because of different occlusion and illumination variations. Many state-of-the-art face detectors have been build which precisely detect the faces from a given image. A novel Wasserstein CNN approach was proposed by He et al. [207] to learn invariant features for face detection. To enhance and speed up the discriminative abilities of a DCNN based face recognizer, appropriate loss functions need

Table 8
Comparative analysis of pedestrian detection datasets.

Dataset	Setup	Training set			Testing set		
		Pedestrians	Positive images	Negative images	Pedestrians	Positive images	Negative images
Caltech [52]	Mobile	192K	67K	61K	155K	65K	56K
INRIA [26]	Photo	1208	614	1218	566	288	453
ETH [199]	Mobile	2388	499	NA	12K	1804	NA
TUD-Brussels [201]	Mobile	1776	1092	218	1498	508	NA
Daimler-DB [53]	Mobile	192K	67K	61K	155K	65K	56K

Table 9
Comparison of person detection benchmarks.

Dataset	Seasons	Countries	Cities	Images	Pedestrians	Image resolution	Weather	Train-Val-Test-Split (%)
Caltech [52]	1	1	1	249884	289395	640 X 480	Dry	50-0-50
KITTI [202]	1	1	1	14999	9400	1240 X 376	Dry	50-0-50
City Persons [203]	3	3	27	5000	31514	2048 X 1024	Dry	60-10-30
TDC [204]	1	1	1	14674	8919	2048 X 1024	Dry	71-8-21

Table 10
Details of evaluation metrics for generic object detection.

Abbreviation	Meaning	Description
Ω	IoU Threshold	Used for evaluating localization accuracy.
D	All Predictions	Peak predictions with highest confidence score generated by the object detector.
TP	True Positive	Exact predictions made by object detector.
FP	False Positive	Sham predictions made by object detector.
P	Precision	Fraction of true positives out of all predictions.
AP	Average Precision	Computed over different values of recall.
TPR	True Positive Rate	Portion of positive rate over false positives.
FPPI	False Positive Per Image	Portion of false positive for each image.
FPS	Frames Per Second	Number of images processes per second.
MR	Missing Rate	Average miss rate over diverse FPPI rates equally placed in log-space.

to be designed. One such loss function is cosine based softmax loss [208–211] which is widely been used. Guo et al. [212] proposed a fuzzy logic based sparse auto encoder framework for face recognition.

7.2. Pedestrian detection:

This application aims at detecting pedestrians from a natural environment. Braun et al. [213] developed a EuroCity person dataset which typically comprises of pedestrians and cyclists. For real time pedestrian detection, some cascaded pedestrian detection models were developed [214–216].

7.3. Anomaly detection:

A noteworthy part in fraud detection, healthcare monitoring and weather scrutiny is played by anomaly detection technique. Current anomaly detection techniques study the data using a point wise criteria [217–220]. For analyzing contiguous time and space intervals Barz et al. [221] proposed an unsupervised technique called Maximally Divergent Intervals for anomaly detection.

7.4. License plate recognition:

With the growing craze for cars and vehicles, license plate recognition is considered as a hot topic today as it is required in vehicular and traffic violation tracking. For making license plate recognition more robust and trustworthy, several techniques like edge detection, morphology, sliding concentric windows etc are all clubbed together to achieve that. In recent years deep learning based methods [222–227] also provide solutions for this application.

7.5. Autonomous driving:

For automatically driving a car a perfect perception is needed to operate in a reliable manner. Deep learning based perception systems are normally employed for this purpose which transfers multi sensory data and converts it into semantic knowledge thereby enabling automatic driving. Object detection is a fundamental aspect of this driving system. In recent years Lu et al. [228] make use of fresh setups containing 3D convolutions and RNN's to achieve localization accuracy in several real world driving scenarios. Apart from that song et al. [228] developed a 3D car instance understanding benchmark for autonomous driving and Banerjee et al. [229] utilized sensor fusion to extract efficient features.

7.6. Traffic sign recognition:

For the sake of security and rule following, real time accurate traffic sign recognition is required which assists in driving by obtaining temporal and spatial information of various traffic signs. Deep learning based object detection methods [230–243] tries to solve this problem with high accuracy.

7.7. Computer aided diagnosis (CAD) system:

These systems can assist doctors and physicians to categorize different kinds of cancer. The fundamental tasks carried out by a CAD setup can be recognized as image acquisition followed by segmentation, feature extraction, classification and finally object detection. Because of significant entity differences, lack of data and privacy concerns, there generally exists a difference of data distribution between source and target domains. Hence domain adaptation setups [244] are required for medical image detection.

7.8. Event detection:

This application focuses on recognizing real world events from the web, like festivals, talk shows, disasters, election campaigns, protests etc. with the advancement of social media, multi-domain event detection systems can provide a detailed description about them. For disposing off multi-domain data an event detection system was developed by Yang et al. [245] Wang et al. [246] added some online social interaction features by developing affine based graphs for event detection purposes. Apart from these advancements, a multi dimensional graph based model for detecting events from millions of video sequences and images was developed by Schinas et al. [247].

7.9. Pattern detection:

Pattern detection always suffer from the difficulties of scene occlusion, varying illuminations, pose variations and sensor based noises. For addressing duplicate patterns and periodic structure detections, researchers have designed and developed some state-of-the-art benchmarks for both 2D [248,249] and 3D images [129,250–260].

7.10. Image caption generation:

This application aims at automatically generating captions for a given image by capturing underlying semantic information from the image and expressing it using natural languages. Image captioning requires both computer vision and NLP technologies which itself are very challenging. So for addressing these issues, reinforcement learning [261,262], attention networks [263,264], encoder–decoder networks are widely been used for this purpose. Apart from all these deep CNN's based techniques are proven to be highly effective and efficient [2,265].

7.11. Salient object detection:

For detecting salient objects, deep neural networks are utilized to foresee saliency scores of image regions and to produce correct saliency maps. Deep neural networks for detecting salient objects normally need to put together multi-level features of the backbone architecture. To obtain fast speed without compromising accuracy. Wu et al. [266] suggested that shallower layer features are sufficient to precisely obtain the saliency map. for detecting salient objects, Wang et al. [267] have used fixation prediction techniques and to accurately detect salient objects, Wang et al. [268] incorporated prior knowledge of saliency into recurrent fully convolutional networks. To properly explore the structure of objects, an attentive feedback module has been designed by Feng et al. [269].

7.12. Text detection:

The aim of text detection is to discover the text region of a given image or a video and it is one of the most important precondition for many of the computer vision tasks like categorization and video analysis. Although there has already been many thriving optical character recognition systems, but still the detection of text in cluttered natural scenes is a challenging task because of blurring, different orientations, lighting conditions and various other distortions. Very recently researchers have found out that randomly-oriented text directions [270–272] is a research direction that requires attention. Generally deep learning based scene text detection can be broadly categorized into two classes. The first category considers the scene text as a particular object and using text box regression techniques try to locate the text within that scene, whereas in the second category, the textual image is directly segmented and requires complicated post processing step. further, in order to get the required orientation of text boxes, some of the text detection techniques require multifaceted post processing, hence they are not as efficient as those methods which are directly based on detection networks. Lyu et al. [273] combined together the ideas proposed in above two text detection categories along with dividing the text regions into relative positions to recognize the underlying text. Ma et al. [274] developed a novice rotation based technique and an end to end text detection system to generate inclined region proposals with text orientation information.

7.13. Point clouds 3D object detection:

A compared to image based object detection. LIDAR based point cloud tries to provide depth information which can further be used for accurately locating the objects and characterizing their shapes. LIDAR point cloud based 3D object detection plays an important role in autonomous driving, robotics and virtual reality applications. Apart from numerous applications of point clouds in object detection, this technique also meet some challenges like sparsity of LIDAR point clouds, inconsistent sampling of the 3D-space, occlusions and relative pose variation. Qi et al. [275] developed an end to end deep neural network called PointNet which can learn features directly from LIDAR clouds. For efficient mapping and processing of enormous 3D data Engelcke et al. [276] proposed sparse convolution layers and L1 regularization. Zhou et al. [277] developed a general end to end 3D object detection framework called VoxelNet that can predict correct 3D bounding boxes by learning a discriminative feature representation from point clouds.

7.14. 2D 3D pose detection:

The aim of human pose detection is to locate the 2D or 3D pose of the body joints along with the base classes and eventually returning the average pose of the maximum scoring class. Emblematic 2D human pose detection technique [237,238,278–280] make use of deep CNN architecture. Rogez et al. [239] came up with an end to end architecture for joint 2D and 3D pose estimation in natural scenes thereby predicting 2D and 3D poses of many people together. The human pose estimation techniques can broadly be divided into two categories, single stage and multi stage methods. The best performing methods are typically based on single stage backbone architectures [240–242]. However the most representative multi stage techniques are convolutional pose machines [243], Hourglass network [280] and MSPN [281].

7.15. Fine grained visual recognition:

The aim is to recognize an exact class of objects in each of the basic level category like identifying the model of a car or recognizing the species of a mammal. This task is a bit challenging due to visual differences which are minimal between the class categories and which can easily be overwhelmed by factors like pose, location of an object and its viewpoint in the given image. Krause et al. [282] make use of 3D object representation to generalize across multiple viewpoints at the level of both locations and local features. Bilinear model consisting of two CNN streams was introduced by Lin et al. [283] wherein the outputs from these two CNN streams are multiplied using outer product at each of the image location which are then pooled together to get an image descriptor. A finely grained discriminative localization method based on saliency guided faster R-CNN was introduced by He at al. [284] and later they introduced a weakly supervised version for fast fine-grained image categorization [285].

7.16. Edge detection:

It aims at extracting the boundary of an object and salient edges from an image which is considered very important for many higher level computer vision tasks like object recognition, segmentation etc. Edge detection also suffers from major challenges like the edges of different scales which are present in an image need both object level boundaries and useful local region details. furthermore to predict different parts in the final detection each of the CNN layers should be trained by proper layer specific supervision. For addressing these issues, He et al. [286] came up with a bi-directional cascade network which allows one layer supervised by labeled edges while adopting dilated convolutions for generating multi-scale features.

8. Concluding remarks and future research areas

Due to its potent learning capabilities and usefulness in dealings with occlusions, scale alterations and background exchanges, deep learning based object detection techniques has become a highly researched area. In this paper we have provided a comprehensive survey of latest advances in deep learning based visual object detection. The review starts with surveying a large body of recent works in literature followed by analyzing traditional and current detectors. Then a rigorous overview of backbone architectures along with systematic coverup of prominent learning strategies is performed. Finally some popular datasets and benchmarks for visual object detection are discussed along with some application areas to gain a thorough understanding of the object detection landscape. With the increasingly powerful visual object detectors in the fields of security, transportation, military etc. the applications of object detection is therefore witnessing a sharp increase. Despite all these advancements, there is still much room for further development. Below we have provided some latest trends in this domain for facilitating future research in visual object detection with deep learning.

8.1. Future research trends:

(i) Video Object Detection: As we know that video object detection suffers from motion target ambiguities, really tiny target objects, truncations and occlusions etc, which make it extremely difficult to achieve high accuracy and efficiency. Henceforth, investigating motion based goals and multifaceted data sources like video sequences are one of the most promising future research areas.

(ii) Weakly Supervised Object Detection: Weakly supervised object detection models focus on utilizing a small set of completely annotated images for detecting a much larger number of non-annotated counterparts. Hence it is a significant problem for future studies where large proportion of annotated and labeled images with target objects and bounding boxes are used to efficiently train a network for achieving high effectiveness.

(iii) Multi-Domain Object Detection: As we know that area specific detectors always tend to perform better, achieving high detection accuracies on a predefined dataset. So the future basically lies in developing a universal object detector which is capable of detecting multi domain objects without having any prior knowledge.

(iv) Salient Object Detection: This area aims at stressing on salient object regions in an image. Salient object detection is applied to a broad spectrum of object detection applications in different areas. Salient object regions of importance in each frame can help to accurately detect the objects in a continuous scene or video sequence. Hence for important recognition and detection tasks, saliency guided object detection can be considered as a preliminary process.

(v) Multi-task Learning: For improving the detection performance, amassing multi level features of backbone architectures can be considered as a significant step. Moreover performing many computer vision tasks simultaneously like, object detection, semantic and instance segmentation etc can improve the performance by a large extent using richer information. Adopting this technique is an efficient way to combine multiple tasks in a model presents a series of challenges for researchers to improve the detection accuracy without compromising the processing speed.

(vi) Unsupervised Object Detection: Developing automatic annotation techniques to get rid of manual annotating is an exciting and promising trend for unsupervised object detection. For sharp detection tasks, unsupervised object detection is a future research direction.

(vii) Remote Sensing Real Time Detection: Remote sensing images find their applications in both military and agricultural domains. Automatic detection models and incorporated hardware units will enhance rapid development of these fields.

(viii) GAN based Object Detectors: As we know that deep learning based object detector often requires huge amounts of data for training purposes, whereas GAN based object detectors is an influential structure which generates fake images. Combining real world scenarios and simulated data produced by GAN helps in making the detectors to grow more robust and to obtain greater generalization capabilities.

The research in object detection with deep learning needs further study. We hope that deep learning based object detectors will make life changing contributions in years to come.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Agrim Gupta, Piotr Dollar, Ross Girshick, LVIS: A dataset for large vocabulary instance segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 5356–5364.

[2] Licheng Jiao, Fan Zhang, Fang Liu, Shuyuan Yang, Lingling Li, Zhixi Feng, Rong Qu, A survey of deep learning-based object detection, IEEE Access 7 (2019) 128837–128868.

[3] Jensen Huang Nvidia, Accelerating AI with GPUs: A new computing model, 2020, Retrieved on June 20, 2020 at 11:45 am, from URL https://blogs.nvidia.com/blog/2016/01/12/accelerating-ai-artificial-intelligence-gpus/.

[4] Xiongwei Wu, Doyen Sahoo, Steven C.H. Hoi, Recent advances in deep learning for object detection, Neurocomputing (2020).

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, Alan L. Yuille, Semantic image segmentation with deep convolutional nets and fully connected crfs, 2014, arXiv preprint arXiv:1412.7062.

[7] Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra 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.

[8] Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick, Mask r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2961–2969.

[9] Yi Sun, Ding Liang, Xiaogang Wang, Xiaoou Tang, Deepid3: Face recognition with very deep neural networks, 2015, arXiv preprint arXiv:1502.00873.

[10] Yi Sun, Yuheng Chen, Xiaogang Wang, Xiaoou Tang, Deep learning face representation by joint identification-verification, in: Advances in Neural Information Processing Systems, 2014, pp. 1988–1996.

[11] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, Le Song, Sphereface: Deep hypersphere embedding for face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 212–220.

[12] Jianan Li, Xiaodan Liang, ShengMei Shen, Tingfa Xu, Jiashi Feng, Shuicheng Yan, Scale-aware fast R-CNN for pedestrian detection, IEEE Trans. Multimedia 20 (4) (2017) 985–996.

[13] Jan Hosang, Mohamed Omran, Rodrigo Benenson, Bernt Schiele, Taking a deeper look at pedestrians, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 4073–4082.

[14] Anelia Angelova, Alex Krizhevsky, Vincent Vanhoucke, Abhijit Ogale, Dave Ferguson, Real-time pedestrian detection with deep network cascades, 2015.

[15] Steven CH Hoi, Xiongwei Wu, Hantang Liu, Yue Wu, Huiqiong Wang, Hui Xue, Qiang Wu, Logo-net: Large-scale deep logo detection and brand recognition with deep region-based convolutional networks, 2015, arXiv preprint arXiv:1511.02462.

[16] Hang Su, Xiatian Zhu, Shaogang Gong, Deep learning logo detection with data expansion by synthesising context, in: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2017, pp. 530–539.

[17] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, Li Fei-Fei, Large-scale video classification with convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1725–1732.

[18] Hossein Mobahi, Ronan Collobert, Jason Weston, Deep learning from temporal coherence in video, in: Proceedings of the 26th Annual International Conference on Machine Learning, 2009, pp. 737–744.

[19] Ross B. Girshick, Fast R-CNN, 2015, CoRR arXiv:1504.08083.

[20] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788.

[21] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, in: Advances in Neural Information Processing Systems, 2015, pp. 91–99.

[22] David G. Lowe, Object recognition from local scale-invariant features, in: Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 2, Ieee, 1999, pp. 1150–1157.

[23] Herbert Bay, Tinne Tuytelaars, Luc Van Gool, Surf: Speeded up robust features, in: European Conference on Computer Vision, Springer, 2006, pp. 404–417.

[24] Rainer Lienhart, Jochen Maydt, An extended set of haar-like features for rapid object detection, in: Proceedings. International Conference on Image Processing, vol. 1, IEEE, 2002, pp. I–I.

[25] Eleonora Vig, Michael Dorr, David Cox, Large-scale optimization of hierarchical features for saliency prediction in natural images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2798–2805.

[26] Navneet Dalal, Bill Triggs, Histograms of oriented gradients for human detection, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, IEEE, 2005, pp. 886–893.

[27] Yu Xiang, Wongun Choi, Yuanqing Lin, Silvio Savarese, Subcategory-aware convolutional neural networks for object proposals and detection, in: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2017, pp. 924–933.

[28] Yoav Freund, Robert E. Schapire, A desicion-theoretic generalization of on-line learning and an application to boosting, in: European Conference on Computational Learning Theory, Springer, 1995, pp. 23–37.

[29] Yoav Freund, Robert E. Schapire, et al., Experiments with a new boosting algorithm, in: Icml, vol. 96, Citeseer, 1996, pp. 148–156.

[30] David Opitz, Richard Maclin, Popular ensemble methods: An empirical study, J. Artif. Intell. Res. 11 (1999) 169–198.

[31] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Trans. Pattern Anal. Mach. Intell. 32 (2010) 1627–1645.

[32] Mark Everingham, Luc Van Gool, Christopher K.I. Williams, John Winn, Andrew Zisserman, The PASCAL visual object classes challenge 2007 (VOC2007) results, 2007.

[33] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, Andrew Zisserman, The pascal visual object classes (voc) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.

[34] David G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.

[35] Timo Ojala, Matti Pietikainen, Topi Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7) (2002) 971–987.

[36] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.

[37] Guimei Cao, Xuemei Xie, Wenzhe Yang, Quan Liao, Guangming Shi, Jinjian 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.

[38] Subarna Tripathi, Gokce Dane, Byeongkeun Kang, Vasudev Bhaskaran, Truong Nguyen, LCDet: Low-complexity fully-convolutional neural networks for object detection in embedded systems, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017, pp. 94–103.

[39] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, Trevor Darrell, Caffe: Convolutional architecture for fast feature embedding, in: Proceedings of the 22nd ACM International Conference on Multimedia, 2014, pp. 675–678.

[40] Zhenheng Yang, Ramakant Nevatia, A multi-scale cascade fully convolutional network face detector, in: 2016 23rd International Conference on Pattern Recognition (ICPR), IEEE, 2016, pp. 633–638.

[41] Ngiam Jiquan, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, AY Ng, Multimodal deep learning, in: Proceedings of the 28th International Conference on Machine Learning (ICML-11), vol. 689696, 2011.

[42] Yu-Gang Jiang, Zuxuan Wu, Jinhui Tang, Zechao Li, Xiangyang Xue, Shih-Fu Chang, Modeling multimodal clues in a hybrid deep learning framework for video classification, IEEE Trans. Multimed. 20 (11) (2018) 3137–3147.

[43] Denis Tomè, Federico Monti, Luca Baroffio, Luca Bondi, Marco Tagliasacchi, Stefano Tubaro, Deep convolutional neural networks for pedestrian detection, Signal Process., Image Commun. 47 (2016) 482–489.

[44] Zhong-Qiu Zhao, Haiman Bian, Donghui Hu, Wenjuan Cheng, Hervé Glotin, Pedestrian detection based on fast R-CNN and batch normalization, in: ICIC 2017: Intelligent Computing Theories and Application, Springer, 2017, pp. 735–746.

[45] Chen Zhang, Joohee Kim, Object detection with location-aware deformable convolution and backward attention filtering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 9452–9461.

[46] Joseph Redmon, Ali Farhadi, YOLO9000: better, faster, stronger, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7263–7271.

[47] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C Berg, Ssd: Single shot multibox detector, in: European Conference on Computer Vision, Springer, 2016, pp. 21–37.

[48] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2117–2125.

[49] Ming-Hsuan Yang, David J. Kriegman, Narendra Ahuja, Detecting faces in images: A survey, IEEE Trans. Pattern Anal. Mach. Intell. 24 (1) (2002) 34–58.

[50] Stefanos Zafeiriou, Cha Zhang, Zhengyou Zhang, A survey on face detection in the wild: past, present and future, Comput. Vis. Image Underst. 138 (2015) 1–24.

[51] Qixiang Ye, David Doermann, Text detection and recognition in imagery: A survey, IEEE Trans. Pattern Anal. Mach. Intell. 37 (7) (2014) 1480–1500.

[52] Piotr Dollar, Christian Wojek, Bernt Schiele, Pietro Perona, Pedestrian detection: An evaluation of the state of the art, IEEE Trans. Pattern Anal. Mach. Intell. 34 (4) (2011) 743–761.

[53] Markus Enzweiler, Dariu M. Gavrila, Monocular pedestrian detection: Survey and experiments, IEEE Trans. Pattern Anal. Mach. Intell. 31 (12) (2008) 2179–2195.

[54] David Geronimo, Antonio M. Lopez, Angel D. Sappa, Thorsten Graf, Survey of pedestrian detection for advanced driver assistance systems, IEEE Trans. Pattern Anal. Mach. Intell. 32 (7) (2009) 1239–1258.

[55] Zehang Sun, George Bebis, Ronald Miller, On-road vehicle detection: A review, IEEE Trans. Pattern Anal. Mach. Intell. 28 (5) (2006) 694–711.

[56] Xin Zhang, Yee-Hong Yang, Zhiguang Han, Hui Wang, Chao Gao, Object class detection: A survey, ACM Comput. Surv. 46 (1) (2013) 1–53.

[57] Licheng Jiao, Fan Zhang, Fang Liu, Shuyuan Yang, Lingling Li, Zhixi Feng, Rong Qu, A survey of deep learning-based object detection, IEEE Access 7 (2019) 128837–128868.

[58] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, Xindong Wu, Object detection with deep learning: A review, IEEE Trans. Neural Netw. Learn. Syst. 30 (11) (2019) 3212–3232.

[59] Zehang Sun, George Bebis, Ronald Miller, On-road vehicle detection: A review, IEEE Trans. Pattern Anal. Mach. Intell. 28 (5) (2006) 694–711.

[60] Jean Ponce, Martial Hebert, Cordelia Schmid, Andrew Zisserman, Toward Category-Level Object Recognition, vol. 4170, Springer, 2007.

[61] Sven J. Dickinson, Aleš Leonardis, Bernt Schiele, Michael J. Tarr, Object Categorization: Computer and Human Vision Perspectives, Cambridge University Press, 2009.

[62] Carolina Galleguillos, Serge Belongie, Context based object categorization: A critical survey, Comput. Vis. Image Underst. 114 (6) (2010) 712–722.

[63] Kristen Grauman, Bastian Leibe, Visual object recognition, Synth. Lect. Artif. Intell. Mach. Learn. 5 (2) (2011) 1–181.

[64] Alexander Andreopoulos, John K. Tsotsos, 50 years of object recognition: Directions forward, Comput. Vis. Image Underst. 117 (8) (2013) 827–891.

[65] Yoshua Bengio, Aaron Courville, Pascal Vincent, Representation learning: A review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.

[66] Ali Borji, Ming-Ming Cheng, Qibin Hou, Huaizu Jiang, Jia Li, Salient object detection: A survey, Comput. Vis. Media (2019) 1–34.

[67] Yali Li, Shengjin Wang, Qi Tian, Xiaoqing Ding, Feature representation for statistical-learning-based object detection: A review, Pattern Recognit. 48 (11) (2015) 3542–3559.

[68] Yann LeCun, Yoshua Bengio, Geoffrey Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[69] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, Clara I Sánchez, A survey on deep learning in medical image analysis, Med. Image Anal. 42 (2017) 60–88.

[70] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al., Recent advances in convolutional neural networks, Pattern Recognit. 77 (2018) 354–377.

[71] Zhengxia Zou, Zhenwei Shi, Yuhong Guo, Jieping Ye, Object detection in 20 years: A survey, 2019, arXiv preprint arXiv:1905.05055.

[72] Paul Viola, Michael Jones, Rapid object detection using a boosted cascade of simple features, in: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, vol. 1, IEEE, 2001, pp. I–I.

[73] Paul Viola, Michael J. Jones, Robust real-time face detection, Int. J. Comput. Vis. 57 (2) (2004) 137–154.

[74] Yoav Freund, Robert Schapire, Naoki Abe, A short introduction to boosting, J. Japanese Soc. Artif. Intell. 14 (771–780) (1999) 1612.

[75] Pedro Felzenszwalb, David McAllester, Deva Ramanan, A discriminatively trained, multiscale, deformable part model, in: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008, pp. 1–8.

[76] Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, Cascade object detection with deformable part models, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2010, pp. 2241–2248.

[77] Tomasz Malisiewicz, Abhinav Gupta, Alexei A. Efros, Ensemble of exemplar-svms for object detection and beyond, in: 2011 International Conference on Computer Vision, IEEE, 2011, pp. 89–96.

[78] Ross B. Girshick, Pedro F. Felzenszwalb, David A. Mcallester, Object detection with grammar models, in: Advances in Neural Information Processing Systems, 2011, pp. 442–450.

[79] Ross Brook Girshick, From Rigid Templates to Grammars: Object Detection with Structured Models, Citeseer, 2012.

[80] Stuart Andrews, Ioannis Tsochantaridis, Thomas Hofmann, Support vector machines for multiple-instance learning, in: S. Becker, S. Thrun, K. Obermayer (Eds.), Advances in Neural Information Processing Systems 15, MIT Press, 2003, pp. 577–584, http://papers.nips.cc/paper/2232-support-vector-machines-for-multiple-instance-learning.pdf.

[81] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, Yann LeCun, Overfeat: Integrated recognition, localization and detection using convolutional networks, 2013, arXiv preprint arXiv:1312.6229.

[82] Joseph Redmon, Ali Farhadi, Yolov3: An incremental improvement, 2018, arXiv preprint arXiv:1804.02767.

[83] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, Piotr Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2980–2988.

[84] Hei Law, Jia Deng, Cornernet: Detecting objects as paired keypoints, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 734–750.

[85] Xingyi Zhou, Dequan Wang, Philipp Krähenbühl, Objects as points, 2019, arXiv preprint arXiv:1904.07850.

[86] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, Qi Tian, Centernet: Object detection with keypoint triplets, 2019, arXiv preprint arXiv:1904.08189 1(2), 4.

[87] Jasper R.R. Uijlings, Koen E.A. Van De Sande, Theo Gevers, Arnold W.M. Smeulders, Selective search for object recognition, Int. J. Comput. Vis. 104 (2) (2013) 154–171.

[88] Jim Kleban, Xing Xie, Wei-Ying Ma, Spatial pyramid mining for logo detection in natural scenes, in: 2008 IEEE International Conference on Multimedia and Expo, IEEE, 2008, pp. 1077–1080.

[89] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (9) (2015) 1904–1916.

[90] C.L.awrence Zitnick, Piotr Dollar, Edge boxes: Locating object proposals from edges, in: European Conference on Computer Vision, Springer, 2014, pp. 391–405.

[91] Jifeng Dai, Yi Li, Kaiming He, Jian Sun, R-fcn: Object detection via region-based fully convolutional networks, in: Advances in Neural Information Processing Systems, 2016, pp. 379–387.

[92] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, C Lawrence Zitnick, Microsoft coco: Common objects in context, in: European Conference on Computer Vision, Springer, 2014, pp. 740–755.

[93] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, Yichen Wei, Relation networks for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3588–3597.

[94] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, Yichen Wei, Deformable convolutional networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 764–773.

[95] Golnaz Ghiasi, Tsung-Yi Lin, Quoc V. Le, Nas-fpn: Learning scalable feature pyramid architecture for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7036–7045.

[96] Bogdan Alexe, Thomas Deselaers, Vittorio Ferrari, Measuring the objectness of image windows, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2189–2202.

[97] Esa Rahtu, Juho Kannala, Matthew Blaschko, Learning a category independent object detection cascade, in: 2011 International Conference on Computer Vision, IEEE, 2011, pp. 1052–1059.

[98] Santiago Manen, Matthieu Guillaumin, Luc Van Gool, Prime object proposals with randomized prim's algorithm, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 2536–2543.

[99] Joao Carreira, Cristian Sminchisescu, CPMC: Automatic object segmentation using constrained parametric min-cuts, IEEE Trans. Pattern Anal. Mach. Intell. 34 (7) (2011) 1312–1328.

[100] Ian Endres, Derek Hoiem, Category-independent object proposals with diverse ranking, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2) (2013) 222–234.

[101] Chenchen Zhu, Ran Tao, Khoa Luu, Marios Savvides, Seeing small faces from robust anchor's perspective, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5127–5136.

[102] Lele Xie, Yuliang Liu, Lianwen Jin, Zecheng Xie, DeRPN: Taking a further step toward more general object detection, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 9046–9053.

[103] Lachlan Tychsen-Smith, Lars Petersson, Denet: Scalable real-time object detection with directed sparse sampling, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 428–436.

[104] Chenchen Zhu, Yihui He, Marios Savvides, Feature selective anchor-free module for single-shot object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 840–849.

[105] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, Qi Tian, Centernet: Keypoint triplets for object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 6569–6578.

[106] Ross Girshick, Fast r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1440–1448.

[107] Bharat Singh, Larry 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.

[108] Zhaowei Cai, Quanfu Fan, Rogerio S Feris, Nuno Vasconcelos, A unified multi-scale deep convolutional neural network for fast object detection, in: European Conference on Computer Vision, Springer, 2016, pp. 354–370.

[109] Zhiqiang Shen, Zhuang Liu, Jianguo Li, Yu-Gang Jiang, Yurong Chen, Xiangyang Xue, Dsod: Learning deeply supervised object detectors from scratch, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1919–1927.

[110] Songtao Liu, Di Huang, et al., Receptive field block net for accurate and fast object detection, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 385–400.

[111] Jimmy Ren, Xiaohao Chen, Jianbo Liu, Wenxiu Sun, Jiahao Pang, Qiong Yan, Yu-Wing Tai, Li Xu, Accurate single stage detector using recurrent rolling convolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5420–5428.

[112] Jisoo Jeong, Hyojin Park, Nojun Kwak, Enhancement of SSD by concatenating feature maps for object detection, 2017, arXiv preprint arXiv:1705.09587.

[113] Cheng-Yang Fu, Wei Liu, Ananth Ranga, Ambrish Tyagi, Alexander C Berg, Dssd: Deconvolutional single shot detector, 2017, arXiv preprint arXiv:1701.06659.

[114] Sanghyun Woo, Soonmin Hwang, In So 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.

[115] Hongyang Li, Yu Liu, Wanli Ouyang, Xiaogang Wang, Zoom out-and-in network with recursive training for object proposal, 2017, arXiv preprint arXiv:1702.05711.

[116] Tao Kong, Fuchun Sun, Chuanqi Tan, Huaping Liu, Wenbing Huang, Deep feature pyramid reconfiguration for object detection, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 169–185.

[117] Qijie Zhao, Tao Sheng, Yongtao Wang, Zhi Tang, Ying Chen, Ling Cai, Haibin Ling, M2det: A single-shot object detector based on multi-level feature pyramid network, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 9259–9266.

[118] Carolina Galleguillos, Serge Belongie, Context based object categorization: A critical survey, Comput. Vis. Image Underst. 114 (6) (2010) 712–722.

[119] Pedro Felzenszwalb, Ross Girshick, David McAllester, Deva Ramanan, Discriminatively trained mixtures of deformable part models, PASCAL VOC Challenge (2008).

[120] Wanli Ouyang, Xiaogang Wang, Xingyu Zeng, Shi Qiu, Ping Luo, Yonglong Tian, Hongsheng Li, Shuo Yang, Zhe Wang, Chen-Change Loy, et al., Deepid-net: Deformable deep convolutional neural networks for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2403–2412.

[121] Xizhou Zhu, Han Hu, Stephen Lin, Jifeng Dai, Deformable convnets v2: More deformable, better results, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 9308–9316.

[122] Ross Girshick, Forrest Iandola, Trevor Darrell, Jitendra Malik, Deformable part models are convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 437–446.

[123] Zeming Li, Chao Peng, Gang Yu, Xiangyu Zhang, Yangdong Deng, Jian Sun, Detnet: A backbone network for object detection, 2018, arXiv preprint arXiv:1804.06215.

[124] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, Kaiming He, Aggregated residual transformations for deep neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1492–1500.

[125] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam, Mobilenets: Efficient convolutional neural networks for mobile vision applications, 2017, arXiv preprint arXiv:1704.04861.

[126] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, Jian Sun, Shufflenet: An extremely efficient convolutional neural network for mobile devices, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6848–6856.

[127] François Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1251–1258.

[128] C.X. Ling R.J. Wang, Pelee: A real-time object detection system on mobile devices, Adv. Neural Inf. Process. Syst. (2018) 1963–1972.

[129] Emmanuel J. Candès, Xiaodong Li, Yi Ma, John Wright, Robust principal component analysis?, J. ACM 58 (3) (2011) 1–37.

[130] Karen Simonyan, Andrew Zisserman, Very deep convolutional networks for large-scale image recognition, 2014, arXiv preprint arXiv:1409.1556.

[131] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.

[132] Herbert Robbins, Sutton Monro, A stochastic approximation method, Ann. Math. Stat. (1951) 400–407.

[133] Diederik P. Kingma, Jimmy Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.

[134] Vinod Nair, Geoffrey E. Hinton, Rectified linear units improve restricted boltzmann machines, in: Proceedings of the 27th International Conference on Machine Learning (ICML-10), 2010, pp. 807–814.

[135] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Identity mappings in deep residual networks, in: European Conference on Computer Vision, Springer, 2016, pp. 630–645.

[136] Sergey Ioffe, Christian Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015, arXiv preprint arXiv:1502.03167.

[137] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, Kilian Q Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.

[138] Hanliang Jiang, Fei Gao, Xingxin Xu, Fei Huang, Suguo Zhu, Attentive and ensemble 3D dual path networks for pulmonary nodules classification, Neurocomputing (2019).

[139] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.

[140] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, Alexander A Alemi, Inception-v4, inception-resnet and the impact of residual connections on learning, in: Thirty-First AAAI Conference on Artificial Intelligence, 2017.

[141] Alejandro Newell, Kaiyu Yang, Jia Deng, Stacked hourglass networks for human pose estimation, in: European Conference on Computer Vision, Springer, 2016, pp. 483–499.

[142] Bharat Singh, Mahyar Najibi, Larry S. Davis, SNIPER: Efficient multi-scale training, in: Advances in Neural Information Processing Systems, 2018, pp. 9310–9320.

[143] Jiayuan Gu, Han Hu, Liwei Wang, Yichen Wei, Jifeng Dai, Learning region features for object detection, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 381–395.

[144] Spyros Gidaris, Nikos Komodakis, Locnet: Improving localization accuracy for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 789–798.

[145] Sergey Zagoruyko, Adam Lerer, Tsung-Yi Lin, Pedro O. Pinheiro, Sam Gross, Soumith Chintala, Piotr Dollár, A multipath network for object detection, 2016, arXiv preprint arXiv:1604.02135.

[146] Lachlan Tychsen-Smith, Lars Petersson, Improving object localization with fitness nms and bounded iou loss, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6877–6885.

[147] Xin Lu, Buyu Li, Yuxin Yue, Quanquan Li, Junjie Yan, Grid r-cnn, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 7363–7372.

[148] Bin Yang, Junjie Yan, Zhen Lei, Stan Z. Li, Craft objects from images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 6043–6051.

[149] Bowen Cheng, Yunchao Wei, Honghui Shi, Rogerio Feris, Jinjun Xiong, Thomas Huang, Revisiting rcnn: On awakening the classification power of faster rcnn, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 453–468.

[150] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, Generative adversarial networks, in: Annual Conference on Neural Information Processing Systems (NeurIPS), 2014, pp. 2672–2680.

[151] Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2223–2232.

[152] Alec Radford, Luke Metz, Soumith Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, 2015, arXiv preprint arXiv:1511.06434.

[153] Andrew Brock, Jeff Donahue, Karen Simonyan, Large scale gan training for high fidelity natural image synthesis, 2018, arXiv preprint arXiv:1809.11096.

[154] Jianan Li, Xiaodan Liang, Yunchao Wei, Tingfa Xu, Jiashi Feng, Shuicheng Yan, Perceptual generative adversarial networks for small object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1222–1230.

[155] Xiaolong Wang, Abhinav Shrivastava, Abhinav Gupta, A-fast-rcnn: Hard positive generation via adversary for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2606–2615.

[156] Zhiqiang Shen, Honghui Shi, Jiahui Yu, Hai Phan, Rogerio Feris, Liangliang Cao, Ding Liu, Xinchao Wang, Thomas Huang, Marios Savvides, Improving object detection from scratch via gated feature reuse, 2017, arXiv preprint arXiv:1712.00886.

[157] Kaiming He, Ross Girshick, Piotr Dollár, Rethinking imagenet pre-training, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 4918–4927.

[158] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, Distilling the knowledge in a neural network, 2015, arXiv preprint arXiv:1503.02531.

[159] Quanquan Li, Shengying Jin, Junjie Yan, Mimicking very efficient network for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6356–6364.

[160] Navaneeth Bodla, Bharat Singh, Rama Chellappa, Larry S. Davis, Soft-NMS–improving object detection with one line of code, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 5561–5569.

[161] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, et al., Speed/accuracy trade-offs for modern convolutional object detectors, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7310–7311.

[162] Zeming Li, Chao Peng, Gang Yu, Xiangyu Zhang, Yangdong Deng, Jian Sun, Light-head r-cnn: In defense of two-stage object detector, 2017, arXiv preprint arXiv:1711.07264.

[163] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510–4520.

[164] Alexander Womg, Mohammad Javad Shafiee, Francis Li, Brendan Chwyl, Tiny SSD: A tiny single-shot detection deep convolutional neural network for real-time embedded object detection, in: 2018 15th Conference on Computer and Robot Vision, CRV, IEEE, 2018, pp. 95–101.

[165] Yuxi Li, Jiuwei Li, Weiyao Lin, Jianguo Li, Tiny-dsod: Lightweight object detection for resource-restricted usages, 2018, arXiv preprint arXiv:1807.11013.

[166] Wenling Shang, Kihyuk Sohn, Diogo Almeida, Honglak Lee, Understanding and improving convolutional neural networks via concatenated rectified linear units, in: International Conference on Machine Learning, 2016, pp. 2217–2225.

[167] Yong-Deok Kim, Eunhyeok Park, Sungjoo Yoo, Taelim Choi, Lu Yang, Dongjun Shin, Compression of deep convolutional neural networks for fast and low power mobile applications, 2015, arXiv preprint arXiv:1511.06530.

[168] Yihui He, Xiangyu Zhang, Jian Sun, Channel pruning for accelerating very deep neural networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1389–1397.

[169] Yunchao Gong, Liu Liu, Ming Yang, Lubomir Bourdev, Compressing deep convolutional networks using vector quantization, 2014, arXiv preprint arXiv:1412.6115.

[170] Yujun Lin, Song Han, Huizi Mao, Yu Wang, William J Dally, Deep gradient compression: Reducing the communication bandwidth for distributed training, 2017, arXiv preprint arXiv:1712.01887.

[171] Jiaxiang Wu, Cong Leng, Yuhang Wang, Qinghao Hu, Jian Cheng, Quantized convolutional neural networks for mobile devices, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4820–4828.

[172] Song Han, Huizi Mao, William J Dally, Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding, 2015, arXiv preprint arXiv:1510.00149.

[173] Song Han, Jeff Pool, John Tran, William Dally, Learning both weights and connections for efficient neural network, in: Advances in Neural Information Processing Systems, 2015, pp. 1135–1143.

[174] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, Stan Z. Li, Single-shot refinement neural network for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4203–4212.

[175] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252.

[176] Saining Xie, Zhuowen Tu, Holistically-nested edge detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1395–1403.

[177] Mahyar Najibi, Mohammad Rastegari, Larry 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.

[178] Yuting Zhang, Kihyuk Sohn, Ruben Villegas, Gang Pan, Honglak Lee, Improving object detection with deep convolutional networks via bayesian optimization and structured prediction, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 249–258.

[179] Fan Yang, Wongun Choi, Yuanqing 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.

[180] Samarth Brahmbhatt, Henrik I. Christensen, James Hays, Stuffnet: Using 'stuff'to improve object detection, in: 2017 IEEE Winter Conference on Applications of Computer Vision, WACV, IEEE, 2017, pp. 934–943.

[181] Shaoqing Ren, Kaiming He, Ross Girshick, Xiangyu Zhang, Jian Sun, Object detection networks on convolutional feature maps, IEEE Trans. Pattern Anal. Mach. Intell. 39 (7) (2016) 1476–1481.

[182] Spyros Gidaris, Nikos 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.

[183] Tao Kong, Anbang Yao, Yurong Chen, Fuchun 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.

[184] Abhinav Shrivastava, Abhinav Gupta, Ross 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.

[185] Sean Bell, C. Lawrence Zitnick, Kavita Bala, Ross 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.

[186] Zhengxia Zou, Zhenwei Shi, Yuhong Guo, Jieping Ye, Object detection in 20 years: A survey, 2019, arXiv preprint arXiv:1905.05055.

[187] Peng Zhou, Bingbing Ni, Cong Geng, Jianguo Hu, Yi Xu, Scale-transferrable object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 528–537.

[188] Buyu Li, Yu Liu, Xiaogang Wang, Gradient harmonized single-stage detector, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 8577–8584.

[189] Zhi Tian, Chunhua Shen, Hao Chen, Tong He, Fcos: Fully convolutional one-stage object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 9627–9636.

[190] Xingyi Zhou, Jiacheng Zhuo, Philipp Krahenbuhl, Bottom-up object detection by grouping extreme and center points, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 850–859.

[191] Yousong Zhu, Chaoyang Zhao, Jinqiao Wang, Xu Zhao, Yi Wu, Hanqing Lu, Couplenet: Coupling global structure with local parts for object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4126–4134.

[192] Hongyu Xu, Xutao Lv, Xiaoyu Wang, Zhou Ren, Navaneeth Bodla, Rama Chellappa, Deep regionlets for object detection, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 798–814.

[193] Zhe Chen, Shaoli Huang, Dacheng Tao, Context refinement for object detection, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 71–86.

[194] Zhaowei Cai, Nuno Vasconcelos, Cascade r-cnn: Delving into high quality object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6154–6162.

[195] Yanghao Li, Yuntao Chen, Naiyan Wang, Zhaoxiang Zhang, Scale-aware trident networks for object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 6054–6063.

[196] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, Li Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 248–255.

[197] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, et al., The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale, 2018, arXiv preprint arXiv:1811.00982.

[198] Pengfei Zhu, Longyin Wen, Xiao Bian, Haibin Ling, Qinghua Hu, Vision meets drones: A challenge, 2018, arXiv preprint arXiv:1804.07437.

[199] Andreas Ess, Bastian Leibe, Luc Van Gool, Depth and appearance for mobile scene analysis, in: 2007 IEEE 11th International Conference on Computer Vision, IEEE, 2007, pp. 1–8.

[200] Andreas Geiger, Philip Lenz, Christoph Stiller, Raquel Urtasun, Vision meets robotics: The kitti dataset, Int. J. Robot. Res. 32 (11) (2013) 1231–1237.

[201] Christian Wojek, Stefan Walk, Bernt Schiele, Multi-cue onboard pedestrian detection, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 794–801.

[202] Andreas Geiger, Philip Lenz, Raquel Urtasun, Are we ready for autonomous driving? The kitti vision benchmark suite, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 3354–3361.

[203] Shanshan Zhang, Rodrigo Benenson, Bernt Schiele, Citypersons: A diverse dataset for pedestrian detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3213–3221.

[204] Xiaofei Li, Fabian Flohr, Yue Yang, Hui Xiong, Markus Braun, Shuyue Pan, Keqiang Li, Dariu M Gavrila, A new benchmark for vision-based cyclist detection, in: 2016 IEEE Intelligent Vehicles Symposium, IV, IEEE, 2016, pp. 1028–1033.

[205] Vidit Jain, Erik Learned-Miller, Fddb: A Benchmark for Face Detection in Unconstrained Settings, Technical Report, UMass Amherst Technical Report, 2010.

[206] Shuo Yang, Ping Luo, Chen-Change Loy, Xiaoou Tang, Wider face: A face detection benchmark, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 5525–5533.

[207] Ran He, Xiang Wu, Zhenan Sun, Tieniu Tan, Wasserstein cnn: Learning invariant features for nir-vis face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 41 (7) (2018) 1761–1773.

[208] Xiao Zhang, Rui Zhao, Yu Qiao, Xiaogang Wang, Hongsheng Li, Adacos: Adaptively scaling cosine logits for effectively learning deep face representations, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 10823–10832.

[209] Yu Liu, Hongyang Li, Xiaogang Wang, Rethinking feature discrimination and polymerization for large-scale recognition, 2017, arXiv preprint arXiv:1710.00870.

[210] Rajeev Ranjan, Carlos D Castillo, Rama Chellappa, L2-constrained softmax loss for discriminative face verification, 2017, arXiv preprint arXiv:1703.09507.

[211] Feng Wang, Xiang Xiang, Jian Cheng, Alan Loddon Yuille, Normface: L2 hypersphere embedding for face verification, in: Proceedings of the 25th ACM International Conference on Multimedia, 2017, pp. 1041–1049.

[212] Yuwei Guo, Licheng Jiao, Shuang Wang, Shuo Wang, Fang Liu, Fuzzy sparse autoencoder framework for single image per person face recognition, IEEE Trans. Cybern. 48 (8) (2017) 2402–2415.

[213] Markus Braun, Sebastian Krebs, Fabian Flohr, Dariu M. Gavrila, Eurocity persons: A novel benchmark for person detection in traffic scenes, IEEE Trans. Pattern Anal. Mach. Intell. 41 (8) (2019) 1844–1861.

[214] Zhaowei Cai, Mohammad Javad Saberian, Nuno Vasconcelos, Learning complexity-aware cascades for pedestrian detection, IEEE Trans. Pattern Anal. Mach. Intell. (2019).

[215] Mohammad Javad Saberian, Nuno Vasconcelos, Learning optimal embedded cascades, IEEE Trans. Pattern Anal. Mach. Intell. 34 (10) (2012) 2005–2018.

[216] Piotr Dollár, Ron Appel, Serge Belongie, Pietro Perona, Fast feature pyramids for object detection, IEEE Trans. Pattern Anal. Mach. Intell. 36 (8) (2014) 1532–1545.

[217] Song Liu, Makoto Yamada, Nigel Collier, Masashi Sugiyama, Change-point detection in time-series data by relative density-ratio estimation, Neural Netw. 43 (2013) 72–83.

[218] Pavel Senin, Jessica Lin, Xing Wang, Tim Oates, Sunil Gandhi, Arnold P. Boedihardjo, Crystal Chen, Susan Frankenstein, Grammarviz 3.0: Interactive discovery of variable-length time series patterns, ACM Trans. Knowl. Dis. Data (TKDD) 12 (1) (2018) 1–28.

[219] Meng Jiang, Alex Beutel, Peng Cui, Bryan Hooi, Shiqiang Yang, Christos Faloutsos, A general suspiciousness metric for dense blocks in multimodal data, in: 2015 IEEE International Conference on Data Mining, IEEE, 2015, pp. 781–786.

[220] Elizabeth Wu, Wei Liu, Sanjay Chawla, Spatio-temporal outlier detection in precipitation data, in: International Workshop on Knowledge Discovery from Sensor Data, Springer, 2008, pp. 115–133.

[221] Björn Barz, Erik Rodner, Yanira Guanche Garcia, Joachim Denzler, Detecting regions of maximal divergence for spatio-temporal anomaly detection, IEEE Trans. Pattern Anal. Mach. Intell. 41 (5) (2018) 1088–1101.

[222] Gong Cheng, Junwei Han, A survey on object detection in optical remote sensing images, ISPRS J. Photogramm. Remote Sens. 117 (2016) 11–28.

[223] Palaiahnakote Shivakumara, Dongqi Tang, Maryam Asadzadehkaljahi, Tong Lu, Umapada Pal, Mohammad Hossein Anisi, CNN-RNN based method for license plate recognition, CAAI Trans. Intell. Technol. 3 (3) (2018) 169–175.

[224] Muhammad Sarfraz, Mohammed Jameel Ahmed, An approach to license plate recognition system using neural network, in: Exploring Critical Approaches of Evolutionary Computation, IGI Global, 2019, pp. 20–36.

[225] Hui Li, Peng Wang, Chunhua Shen, Toward end-to-end car license plate detection and recognition with deep neural networks, IEEE Trans. Intell. Transp. Syst. 20 (3) (2018) 1126–1136.

[226] Jinxing Qian, Bo Qu, Fast license plate recognition method based on competitive neural network, in: 2018 3rd International Conference on Communications, Information Management and Network Security, CIMNS 2018, Atlantis Press, 2018.

[227] Rayson Laroca, Evair Severo, Luiz A Zanlorensi, Luiz S Oliveira, Gabriel Resende Gonçalves, William Robson Schwartz, David Menotti, A robust real-time automatic license plate recognition based on the YOLO detector, in: 2018 International Joint Conference on Neural Networks, IJCNN, IEEE, 2018, pp. 1–10.

[228] Xibin Song, Peng Wang, Dingfu Zhou, Rui Zhu, Chenye Guan, Yuchao Dai, Hao Su, Hongdong Li, Ruigang Yang, Apollocar3d: A large 3d car instance understanding benchmark for autonomous driving, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 5452–5462.

[229] Koyel Banerjee, Dominik Notz, Johannes Windelen, Sumanth Gavarraju, Mingkang He, Online camera lidar fusion and object detection on hybrid data for autonomous driving, in: 2018 IEEE Intelligent Vehicles Symposium, IV, IEEE, 2018, pp. 1632–1638.

[230] Jia Li, Zengfu Wang, Real-time traffic sign recognition based on efficient CNNs in the wild, IEEE Trans. Intell. Transp. Syst. 20 (3) (2018) 975–984.

[231] T. Arinaga T. Moritani, Traffic sign recognition system, US Patent 9, 865, 165, 2018.

[232] Sara Khalid, Nazeer Muhammad, Muhammad Sharif, Automatic measurement of the traffic sign with digital segmentation and recognition, IET Intell. Transp. Syst. 13 (2) (2018) 269–279.

[233] A. lvarez Garcıa, J.A.A. Arcos-Garcıa, L.M. Soria-Morillo, Deep neural network for traffic sign recognition systems: An analysis of spatial transformers and stochastic optimisation methods, Neural Netw. 99 (2018) 158–165.

[234] Dong Li, Dongbin Zhao, Yaran Chen, Qichao Zhang, Deepsign: Deep learning based traffic sign recognition, in: 2018 International Joint Conference on Neural Networks, IJCNN, IEEE, 2018, pp. 1–6.

[235] Bo-Xun Wu, Pin-Yu Wang, Yi-Ta Yang, Jiun-In Guo, Traffic sign recognition with light convolutional networks, in: 2018 IEEE International Conference on Consumer Electronics-Taiwan, ICCE-TW, IEEE, 2018, pp. 1–2.

[236] Shuren Zhou, Wenlong Liang, Junguo Li, Jeong-Uk Kim, Improved VGG model for road traffic sign recognition, Comput. Mater. Continua 57 (1) (2018) 11–24.

[237] Xianjie Chen, Alan L Yuille, Articulated pose estimation by a graphical model with image dependent pairwise relations, in: Advances in Neural Information Processing Systems, 2014, pp. 1736–1744.

[238] Xiaochuan Fan, Kang Zheng, Yuewei Lin, Song Wang, Combining local appearance and holistic view: Dual-source deep neural networks for human pose estimation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1347–1355.

[239] Gregory Rogez, Philippe Weinzaepfel, Cordelia Schmid, Lcr-net++: Multi-person 2d and 3d pose detection in natural images, IEEE Trans. Pattern Anal. Mach. Intell. 42 (5) (2019) 1146–1161.

[240] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, Jian Sun, Cascaded pyramid network for multi-person pose estimation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7103–7112.

[241] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, Kevin Murphy, Towards accurate multi-person pose estimation in the wild, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4903–4911.

[242] Bin Xiao, Haiping Wu, Yichen Wei, Simple baselines for human pose estimation and tracking, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 466–481.

[243] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, Yaser Sheikh, Convolutional pose machines, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4724–4732.

[244] Zhuoling Li, Minghui Dong, Shiping Wen, Xiang Hu, Pan Zhou, Zhigang Zeng, CLU-CNNs: Object detection for medical images, Neurocomputing 350 (2019) 53–59.

[245] Zhenguo Yang, Qing Li, Liu Wenyin, Jianming Lv, Shared multi-view data representation for multi-domain event detection, IEEE Trans. Pattern Anal. Mach. Intell. (2019).

[246] Yanxiang Wang, Hari Sundaram, Lexing Xie, Social event detection with interaction graph modeling, in: Proceedings of the 20th ACM International Conference on Multimedia, 2012, pp. 865–868.

[247] Manos Schinas, Symeon Papadopoulos, Georgios Petkos, Yiannis Kompatsiaris, Pericles A. Mitkas, Multimodal graph-based event detection and summarization in social media streams, in: Proceedings of the 23rd ACM International Conference on Multimedia, 2015, pp. 189–192.

[248] Olivier Teboul, Iasonas Kokkinos, Loic Simon, Panagiotis Koutsourakis, Nikos Paragios, Shape grammar parsing via reinforcement learning, in: CVPR 2011, IEEE, 2011, pp. 2273–2280.

[249] Peng Zhao, Tian Fang, Jianxiong Xiao, Honghui Zhang, Qinping Zhao, Long Quan, Rectilinear parsing of architecture in urban environment, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2010, pp. 342–349.

[250] Sam Friedman, Ioannis Stamos, Online detection of repeated structures in point clouds of urban scenes for compression and registration, Int. J. Comput. Vis. 102 (1–3) (2013) 112–128.

[251] Chao-Hui Shen, Shi-Sheng Huang, Hongbo Fu, Shi-Min Hu, Adaptive partitioning of urban facades, ACM Trans. Graph. 30 (6) (2011) 1–10.

[252] Grant Schindler, Panchapagesan Krishnamurthy, Roberto Lublinerman, Yanxi Liu, Frank Dellaert, Detecting and matching repeated patterns for automatic geo-tagging in urban environments, in: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008, pp. 1–7.

[253] Changchang Wu, Jan-Michael Frahm, Marc Pollefeys, Detecting large repetitive structures with salient boundaries, in: European Conference on Computer Vision, Springer, 2010, pp. 142–155.

[254] Chao-Hui Shen, Shi-Sheng Huang, Hongbo Fu, Shi-Min Hu, Image-based procedural modeling of facades, ACM Trans. Graph. 26 (2007) 85–95.

[255] Olga Barinova, Victor Lempitsky, Elena Tretiak, Pushmeet Kohli, Geometric image parsing in man-made environments, in: European Conference on Computer Vision, Springer, 2010, pp. 57–70.

[256] Mateusz Kozinski, Raghudeep Gadde, Sergey Zagoruyko, Guillaume Obozinski, Renaud Marlet, A MRF shape prior for facade parsing with occlusions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2820–2828.

[257] Andrea Cohen, Alexander G. Schwing, Marc Pollefeys, Efficient structured parsing of facades using dynamic programming, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3206–3213.

[258] Silvia Gandy, Benjamin Recht, Isao Yamada, Tensor completion and low-n-rank tensor recovery via convex optimization, Inverse Problems 27 (2) (2011) 025010.

[259] Ji Liu, Przemyslaw Musialski, Peter Wonka, Jieping Ye, Tensor completion for estimating missing values in visual data, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2012) 208–220.

[260] Juan Liu, Emmanouil Z. Psarakis, Yang Feng, Ioannis Stamos, A kronecker product model for repeated pattern detection on 2d urban images, IEEE Trans. Pattern Anal. Mach. Intell. 41 (9) (2018) 2266–2272.

[261] Oriol Vinyals, Alexander Toshev, Samy Bengio, Dumitru Erhan, Show and tell: A neural image caption generator, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3156–3164.
[262] Jiuxiang Gu, Jianfei Cai, Gang Wang, Tsuhan Chen, Stack-captioning: Coarse-to-fine learning for image captioning, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.
[263] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, Yoshua Bengio, Show, attend and tell: Neural image caption generation with visual attention, in: International Conference on Machine Learning, 2015, pp. 2048–2057.
[264] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, Lei Zhang, Bottom-up and top-down attention for image captioning and visual question answering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6077–6086.
[265] A. Deshpande, J. Aneja, A.G. Schwing, Convolutional image captioning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5561—5570.
[266] Zhe Wu, Li Su, Qingming Huang, Cascaded partial decoder for fast and accurate salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3907–3916.
[267] Wenguan Wang, Jianbing Shen, Xingping Dong, Ali Borji, Ruigang Yang, Inferring salient objects from human fixations, IEEE Trans. Pattern Anal. Mach. Intell. (2019).
[268] Linzhao Wang, Lijun Wang, Huchuan Lu, Pingping Zhang, Xiang Ruan, Salient object detection with recurrent fully convolutional networks, IEEE Trans. Pattern Anal. Mach. Intell. 41 (7) (2018) 1734–1746.
[269] Mengyang Feng, Huchuan Lu, Errui Ding, Attentive feedback network for boundary-aware salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 1623–1632.
[270] Zheng Zhang, Chengquan Zhang, Wei Shen, Cong Yao, Wenyu Liu, Xiang Bai, Multi-oriented text detection with fully convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4159–4167.
[271] Cong Yao, Xiang Bai, Nong Sang, Xinyu Zhou, Shuchang Zhou, Zhimin Cao, Scene text detection via holistic, multi-channel prediction, 2016, arXiv preprint arXiv:1606.09002.
[272] Tong He, Weilin Huang, Yu Qiao, Jian Yao, Accurate text localization in natural image with cascaded convolutional text network, 2016, arXiv preprint arXiv:1603.09423.
[273] Pengyuan Lyu, Cong Yao, Wenhao Wu, Shuicheng Yan, Xiang 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.
[274] Jianqi Ma, Weiyuan Shao, Hao Ye, Li Wang, Hong Wang, Yingbin Zheng, Xiangyang Xue, Arbitrary-oriented scene text detection via rotation proposals, IEEE Trans. Multimed. 20 (11) (2018) 3111–3122.
[275] Martin Engelcke, Dushyant Rao, Dominic Zeng Wang, Chi Hay Tong, Ingmar Posner, Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks, in: 2017 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2017, pp. 1355–1361.
[276] Charles R. Qi, Hao Su, Kaichun Mo, Leonidas J. Guibas, Pointnet: Deep learning on point sets for 3d classification and segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 652–660.
[277] Yin Zhou, Oncel Tuzel, Voxelnet: End-to-end learning for point cloud based 3d object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4490–4499.
[278] Zhe Cao, Tomas Simon, Shih-En Wei, Yaser Sheikh, Realtime multi-person 2d pose estimation using part affinity fields, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7291–7299.
[279] Adrian Bulat, Georgios Tzimiropoulos, Human pose estimation via convolutional part heatmap regression, in: European Conference on Computer Vision, Springer, 2016, pp. 717–732.
[280] Alejandro Newell, Kaiyu Yang, Jia Deng, Stacked hourglass networks for human pose estimation, in: European Conference on Computer Vision, Springer, 2016, pp. 483–499.
[281] Wenbo Li, Zhicheng Wang, Binyi Yin, Qixiang Peng, Yuming Du, Tianzi Xiao, Gang Yu, Hongtao Lu, Yichen Wei, Jian Sun, Rethinking on multi-stage networks for human pose estimation, 2019, arXiv preprint arXiv:1901.00148.
[282] Jonathan Krause, Michael Stark, Jia Deng, Li Fei-Fei, 3d object representations for fine-grained categorization, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2013, pp. 554–561.
[283] Tsung-Yu Lin, Aruni RoyChowdhury, Subhransu Maji, Bilinear cnn models for fine-grained visual recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1449–1457.
[284] Xiangteng He, Yuxin Peng, Junjie Zhao, Fine-grained discriminative localization via saliency-guided faster R-CNN, in: Proceedings of the 25th ACM International Conference on Multimedia, 2017, pp. 627–635.
[285] Xiangteng He, Yuxin Peng, Junjie Zhao, Fast fine-grained image classification via weakly supervised discriminative localization, IEEE Trans. Circuits Syst. Video Technol. 29 (5) (2018) 1394–1407.
[286] Jianzhong He, Shiliang Zhang, Ming Yang, Yanhu Shan, Tiejun Huang, Bi-directional cascade network for perceptual edge detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3828–3837.

Glossary

Average Precision: It is a popular metric for measuring the accuracy of object detectors like faster R-CNN etc. It calculates he average precision value for recall value over 0 to 1.

Precision: It is used for measuring as how accurate is our prediction.

Recall: It measures as how good we find all the positives. For example we can find 60% of all the positive cases in our total M predictions.

IoU: It stands for intersection over union. It is used for measuring the overlap between two boundaries. It is particularly used to measure as how much our predicted boundary overlaps with the ground truth.

NMS: It stands for non maximum suppression. It is defined as a technique which is used for filtering out region proposals based on some criteria.

FPS: It stands for frames per second. It is defined as a frequency rate at which consecutive images appears on the display.

DCN: It stands for deformable convolution network. It consists of two parts, a regular convolution layer and an another layer to learn 2d offset for each input.

SIFT: It stands for scale invariant feature transform. It is defined as a feature detection technique in computer vision to detect and explain local image features.

MSCOCO: It stands for microsoft common objects in context. It is a dataset containing images of 91 object types with a total of 2.5 million labeled instances in 328K images.

SURF: It stands for speeded up robust features. It is defined as a local feature detector and descriptor which can be used for tasks like object recognition, 3d reconstruction, classification etc. Its feature descriptor is based on the summation of Haar wavelet response around points of interest.

NAS-FPN: It stands for neural architecture search feature pyramid network. It consists of a combination of top-down and bottom-up connections to fuse features across different scales.

HOG: It stands for histogram of oriented gradients. It is defined as a feature descriptor used in image processing and computer vision for the task of object detection. It count occurrences of gradient orientations in localized portions of an image.

ResNet: It stands for residual network. It makes it possible to train upto hundreds or even thousands of layers and still achieve remarkable performance.

ResNet: It stands for support vector machine. It is defined as a supervised machine learning model which makes use of classification algorithms for two group classification problems.

ReLU: It stands for rectified linear unit. It is one of the most commonly used activation function deep learning models. A value of 0 is returned, if the function receives a negative input, but for a positive value "x" the same value is returned back.

DPM: It stands for deformable part model detector. It is a learning based object detection FPGA IP core developed for embedded computer vision applications.

VGG: It stands for visual geometry group. It is an innovative object recognition model which supports upto 19 layers. Instead of using a large receptive field like AlexNet, VGG uses very small receptive field (3 x 3 with a stride of 1).

ROI Pooling: It stands for region of interest pooling. It is used for utilizing single feature map for all the proposals generated by RPN in a single pass. It solves the problem of fixed image size requirement for object detection.

SGD: It stands for stochastic gradient descent. It is defined as an iterative technique for optimizing an objective function with suitable smoothness properties.

CNN: It stands for convolutional neural network. It is defined as a deep learning algorithm which can take an input image, assign importance to various aspects in the given image and be able to differentiate one from another.

DPN: It stands for dual path network. It is an image classification model which picks the advantages of both ResNet and DenseNet and also outperforms them in the image classification task.

R-CNN: It stands for region proposal convolution neural network, which allows us to work with 2000 image regions instead of trying to classify a huge number of regions, thereby minimizing the total number of computations required.

RPN: It stands for region proposal network. It is a neural network used for generating proposals for object detection in faster R-CNN.

R-FCN: It stands for region based fully convolutional network. It is used for accurate and efficient object detection. It is a region based detector which is fully convolutional with almost all the computations shared on the entire image.

FPN: It stands for feature pyramid network. It is defined as a top-down architecture with lateral connections which is developed for building high-level semantic feature maps at all scales. It has achieved a remarkable improvement as a generic feature extractor in many applications.

GAN: It stands for generative adversarial network. It is defined as a network wherein given a training dataset, it learns to generate new data with same statistics as the training set.

SPP-Net: It stands for spatial pyramid pooling network. It is defined as a network which adds a new layer between the fully connected layer and the convolutional layer to map any sized input down to a fixed sized output.

DSOD: It stands for deeply supervised object detector. It is defined as a framework which can learn object detector from scratch.

YOLO: It stands for you only look once. It is defined as a state-of-the-art real time single stage object detector which processes images at 30 fps and has an mAP of 57.9% on COCO test-dev dataset.

ILSVRC: It stands for imagenet large scale visual recognition challenge. It is an annual computer vision competition developed on a subset of a publicly available dataset called imagenet. It evaluates algorithms for object detection and image classification at large scales.

SSD: It stands for single shot multibox detector. It is a method used for detecting objects in images using a single deep neural network. It discretizes the output space of bounding boxes into a set of default boxes over different aspect ratios and scales per feature map location.

LVIS: It stands for large vocabulary instance segmentation. It is a large scale dataset comprises of 2 million high quality instance segmentation masks for over 1000 entry level object categories in 164K images.

VOC: It stands for visual object class. It is a project which provides standardized image datasets for object class recognition and provides a common set of tools for accessing the datasets and annotations. It ran challenges evaluating performance on object class recognition from 2005 to 2012.

M-DOD: It stands for multi domain object detection. It is defined as a technique to universally detect multi domain objects without its prior knowledge.