

Received August 19, 2020, accepted August 29, 2020, date of publication September 3, 2020,  
date of current version September 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3021508

# Exploring Deep Learning-Based Architecture, Strategies, Applications and Current Trends in Generic Object Detection: A Comprehensive Review

LUBNA AZIZ<sup>ID</sup><sup>1,2</sup>, MD. SAH BIN HAJI SALAM<sup>3,4</sup>, (Member, IEEE),  
USMAN ULLAH SHEIKH<sup>5</sup>, AND SARA AYUB<sup>3</sup>

<sup>1</sup>Department of Computer Engineering, BUITEMES, Quetta 87100, Pakistan

<sup>2</sup>Department of Computer Engineering, UTM, Johor Bahru 81310, Malaysia

<sup>3</sup>Faculty of Computing, School of Electrical Engineering, UTM, Johor Bahru 81310, Malaysia

<sup>4</sup>Faculty of Engineering, School of Computing, UTM, Johor Bahru 81310, Malaysia

<sup>5</sup>Faculty of Electrical Engineering, UTM, Johor Bahru 81310, Malaysia

Corresponding author: Lubna Aziz (enrg.lubnaaziz@gmail.com)

**ABSTRACT** Object detection is a fundamental but challenging issue in the field of generic image analysis; it plays an important role in a wide range of applications and has been receiving special attention in recent years. Although there are numerous methods exist, an in-depth review of the literature concerning generic detection remains. This paper provides a comprehensive survey of recent advances in visual object detection with deep learning. Covering about 300 publications that we survey 1) region proposal-based object detection methods such as R-CNN, SPPNet, Fast R-CNN, Faster R-CNN, Mask RCN, RFCN, FPN, 2) classification/regression base object detection methods such as YOLO(v2 to v5), SSD, DSSD, RetinaNet, RefineDet, CornerNet, EfficientDet, M2Det 3) Some latest detectors such as, relation network for object detection, DCN v2, NAS FPN. Moreover, five publicly available benchmark datasets and their standard evaluation metrics are also discussed. We mainly focus on the application of deep learning architectures to five major applications, namely Object Detection in Surveillance, Military, Transportation, Medical, and Daily Life. In the survey, we cover a variety of factors affecting the detection performance in detail, such as i) a wide range of object categories and intra-class variations, ii) limited storage capacity and computational power. Finally, we finish the survey by identifying fifteen current trends and promising direction for future research.

**INDEX TERMS** Object detection and recognition, deep learning, convolutional neural networks (CNN), and neural network.

## I. INTRODUCTION

Object detection is a combination of image classification with precise object localization that provides a complete and proper understanding of the image. Previously, Manual feature extraction followed by shallow trainable architectures was used for object detection. However, with the advent of deep learning tools, we have overcome many limitations of traditional object detection techniques that have the ability to learn semantic and deep level features. Generic object detection further divided into different categories such as face detection [1], pedestrian detection [2] and skeleton detection [3], etc. It is a fundamental computer vision process that provides detailed semantic information of image and video.

The associate editor coordinating the review of this manuscript and approving it for publication was Naveed Akhtar .

It has many applications in various fields of life, such as human behavior analysis [4], face recognition [5], image classification [6], medical diagnosis, and autonomous driving [7], [8]. Recently this field gains the attention of many researchers [9], [10]. Object detection comprises two operations; object localization that determines the location of an object in the image, objects classification that determines to which category the object belongs. However, localization in object detection becomes difficult due to occlusions, significant variations in viewpoints, scales, poses, and lighting biasness.

Traditional object detection models have three main modules: informative region selection, extraction of features, and classification.

INFORMATIVE REGION SELECTION is a process of selecting the objects that appear in image at a different position with variable aspect ratios or sizes. It uses the multi-scale

sliding window to scan the whole image. Region selection is a highly computational process that produces redundant windows at all possible locations of an object in the image. Fixed sliding windows cause many unnecessary region productions.

**FEATURE EXTRACTION** is a fundamental task for object recognition that base on visual features extraction to represents the semantic and robust nature of the object. some feature representative are SIFT [11], HOG [12], and Haar-like [13]. Manual designing of a robust feature descriptor is intricate, which perfectly describes objects of all kinds due to the diversity of appearance, illumination condition, and background.

**CLASSIFICATION** is the process of categorizing the target object from all other categories. Besides that, it needs to make representation more informative, semantic, and hierarchical for visual recognition. Some effective classifier are AdaBoost [14] Support-Vector machine (SVM)[15], AdaBoost [14], and Deformable part-based model (DPM) [16].

The era of computer vision inventions begins with the development of the Deep Neural Network (DNN). It marked a major revolution with the invention of the CONV properties (R-CNN). DNN works differently from the traditional approach due to more in-depth architecture, the ability to learn sophisticated features, and a robust training algorithm that allows features to learn the informational representation of objects without manually designing them [17].

Since the invention of R-CNN, a significant number of different and improved models have been proposed in the field of object detection such as Fast R-CNN, which improves the object detection task by combining Bounding Box regression and classification task [18].

In contrast, Faster R-CNN generates a region proposal using the additional sub-network [9], while using fixed grid regression in YOLO for object detection tasks [19]. All of these object detection algorithms make real-time object detection more achievable by providing a better and more accurate way to identify objects on a basic R-CNN.

Object detection models are very effective across different application domains such as salient object detection [20], [21], face detection [22], [23], generic object detection [9], [10], [18] and pedestrian detection [24], [25] as shown in FIGURE 1.

Salient object detection uses segmentation on pixel-level and local contrast enhancement, while generic object detection uses bounding box regression (BB) for detection. Generic object detection is closely related to pedestrian and face detection by adapting multi-scaling and fusion of multi-features. Face and pedestrian images have regular geometric structures; however, complex variations in structures and layout are common limitations.

## II. CONTRIBUTIONS

Numerous surveys have been published in recent years on the generic object detection. In this survey, we have discussed many of the most advanced object detection models based on

deep learning. The main differences between this article and previous studies are mentioned below:

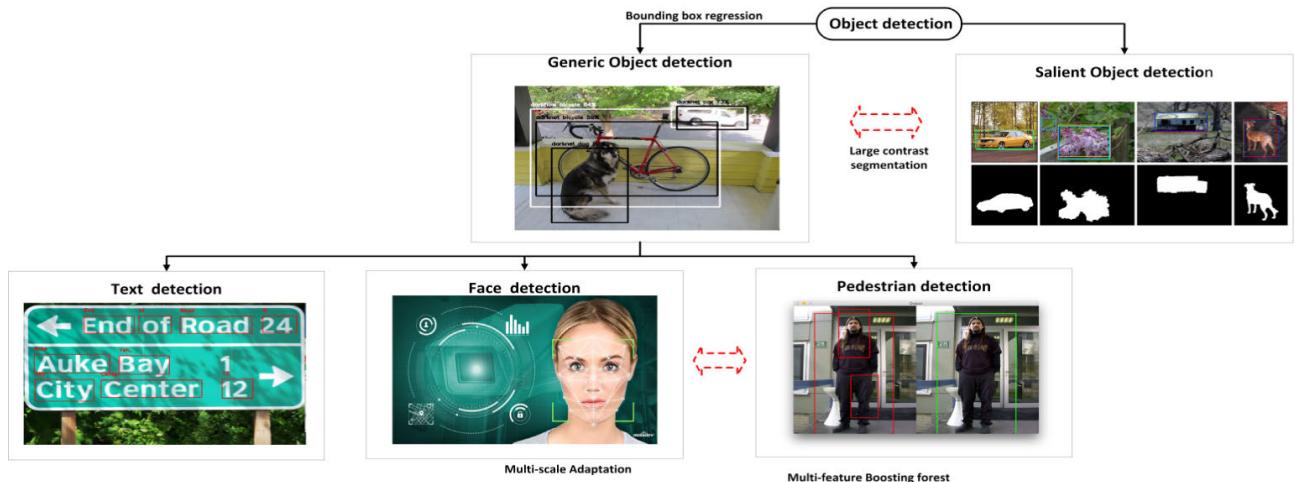
1. A comprehensive overview of state-of-the-art object detectors in the light of technical assessment is included in this article. The history of the development of object detection is spread over one quarterly period (1990 to 2020). Most of the previous surveys merely focus on the limited historical period or some specific detection tasks without considering the technical evaluations. The history highway presented in this survey not only helps to build the readers complete academic rankings also help in finding the future directions of this fast-growing field.
2. Moreover, unlike previous surveys based on object detection, this survey systematically and comprehensively reviews the in-depth exploration of the key-technologies of deep learning-based object detection methods. Following the development of the latest models, new trends have now emerged the models with new technologies such as bonding-box-regression, hard-negativity-manning, and multi-scale detection.
3. More in-depth analysis and discussion in various aspects of object detection provide for the first time in the field.

The rest of the paper is as follows. Section II provides a comparison with previously proposed surveys. However, Section III includes a brief history of deep learning and a brief introduction to CNN's underlying architecture. So far, Section 4 outlines the latest methods for detecting generic objects with a full range of backbone frameworks (uses for base feature extraction), benchmark datasets, and performance evaluation parameters. In contrast, Section V provides the role of object detection in five different fields. The last section presents some exciting trends and development trends in the future.

## III. COMPARISON WITH PREVIOUS SURVEYS

Many impressive surveys of generic object detection have been published, as shown in Table1. These surveys are performed on applications that detect particular objects such as text detection [26], face detection [27], [28], pedestrian detection [2], [29], [30] and vehicle detection [31]. Some of the recent surveys focus directly on generic object detection issues rather than working principles. However, most of the research reviewed in [32]–[34] dates back to before 2012, covering the period before the overwhelming and surprising success of deep learning methods. Deep learning leads to significant advances in areas such as object detection, natural language processing, genomics, speech recognition, drug discovery, medical imaging, and visual recognition by allowing systems to learn abstract, complex, and subtle representations.

Although there have been many published deep-learning-based surveys such as [17], [35], [36], nevertheless, recent improvements in the field of object detection need to be



**FIGURE 1.** Object detection application domain: object detection has two main sub-domains like salient object detection and generic object detection, which further divides into two branches (face detection and pedestrian detection); Saliency detection studies in the context of the visual system. Pedestrian detection is an essential task of any surveillance system, while face detection uses for security purposes.

put together, especially for new researchers who want to research computer vision. The scope of the paper is generic object detection, instead of specific object detection such as face recognition [37]–[39], pedestrian detection [40], [41], vehicle detection [42], and traffic sign detection [43] is not considered.

#### IV. DEEP LEARNING: A BRIEF HISTORY

Before we go into details of deep learning-based object detection, it is essential to explore the benefit of deep learning-based architecture (i.e., CNN). A neural network with deep architecture is known as a deep model. The era of the neural network begins in 1940 [53]; the basic idea behind it was to solve the common problem of learning by mimicking the human brain. The popularity of deep-learning increased in the late 1980s and 1990s with the development of a back-propagation algorithm proposed by Hinton *et al.* [54].

In early 2000, the popularity of deep learning began to decline due to a lack of big data, high computational power requirements, and performance insignificance as compared to other machine learning tools. The rise in popularity of Deep learning began in the year of 2006 with the fantastic and surprising results in speech recognition [55]. Some of the recovery factors of deep learning listed below:

1. The availability of large annotated training datasets such as ImageNet [56] is the main reason for its success.
2. The invention of high performance parallel computing systems such as the GPU cluster.
3. There are significant advances in deep learning model architecture and training strategies: Auto-Encoder (AE) [57] and Restricted Boltzmann Machine (RBM) [58] provide a good start through unsupervised and layer-wise pre-training strategies. The problem of overfitting during the training process can be solved using

data augmentation and dropout regularization [59], [60]. However, Batch normalization (BN) uses for time optimization in the training of deep neural networks [61]. The era of high performance begins with advances in network architecture, such as AlexNet [59], GoogleNet [62], VGG [63], Overfeat [64], and ResNet [65] etc.

The basis of deep learning models is a typical Convolutional Neural Network (CNN) model, such as VGG16 [17]. The featured map is an additional name for the layers in the CNN model, and its input layer is a 3D matrix of pixel intensity of three color channels (red, green, and blue).

A feature map makes an inner layer multi-channel image, and its pixel values are considered special features. Each neuron attaches to a neuron adjacent to the posterior layer. Filtering and pooling transformations on feature maps can create more robust feature specifications [59], [66], [67]. The filtering-transformation uses to filter the matrix convolution to obtain the corresponding field values of the neuron and the final response by applying non-linear activation functions such as ReLU or Sigmoid function [68]. Ultimately different flavors of pooling operations such as max pooling, average pooling, L2 pooling, global pooling, and local contrast normalization [69] are used to create more robust features.

Multiple Fully Connected Layers (FCs) are used with convolution and Pooling Layers to build the initial feature hierarchy in a supervised manner to perform various visual tasks. A specific conditional probability of each neuron in the output layer is obtained by using separate activation functions according to the visual task. Finally, network optimization is performed via SGD (Stochastic Gradient Descent) with objective functions such as mean square error (MSE) and cross-entropy loss. At the same time, cropping or rescaling operations are needed to handle different sizes.

*Some of the advantages of CNN over traditional methods are listed below:*

**TABLE 1.** A Surveys Comparison Table (Generic Object Detection).

S.no	Surveys	reference	years	Scope and content
<b>PEDESTRIAN DETECTION</b>				
1	Monocular pedestrian detection: survey and experiments	Enzweiler and Gavrila et al.[29]	2009	A survey paper focuses on three essential pedestrian detector evaluation
2	Survey of pedestrian detection for advanced driver assistance Systems	Geronimo et al.[30]	2010	This survey provides a comprehensive review of advanced driver assistance systems for pedestrian detection
3	Pedestrian detection: An evaluation of the state of the art	Dolar et al.[2]	2012	A survey paper focuses on pedestrian detector using monocular images
<b>FACE DETECTION</b>				
4	Detecting faces in images: a survey	Yang et al.[27]	2002	Face detection from single image covers in this survey.
5	A survey on face detection in the wild: past, present, and future	Zafeiriou et al.[28]	2015	This survey paper focuses on face detection since 2000 in the wild.
6	Face detection techniques: a review	Ashu Kumar [44]	2019	This paper presents a comprehensive survey of various techniques explored for face detection in digital images.
<b>VEHICLE DETECTION</b>				
7	On-road vehicle detection: a review	Sun et al.[31]	2006	This survey-based on-road vehicle detection systems
8	A survey on vehicle detection Techniques in aerial surveillance	Ramakrishnan et al.[45]	2012	This survey provides research related to vehicle detection techniques.
9	Object Detection in Optical Remote Sensing Images: A Survey and A New Benchmark	Ke Li et al.[46]	2020	This survey provided a comprehensive review based on object detection in computer vision and earth observation communities and proposed a new benchmark dataset DIOR.
10	A survey on 3d object detection methods for autonomous driving applications	E. Arnold et al.[47]	2019	The review contains 3D object detection methods and sensors and datasets in AVs.
<b>TEXT DETECTION</b>				
11	Text Detection, Tracking and Recognition in Video: A Comprehensive Survey	Xu-Cheng Yin et al. [48]	2016	This survey provides different techniques for text detection, recognition, and tracking in videos.
12	Scene Text Detection and Recognition: The Deep Learning Era	Shangbang Long et al. [49]	2018	Scene text detection and recognition in deep learning era describes in this survey
13	A Survey on Text Information Extraction from Born-Digital and Scene Text Images	S. P. Faustina Joan et al. [50]	2018	This survey provides a broad summary of text detection methods, benchmark datasets, and performance metrics.
<b>OBJECT DETECTION</b>				
14	50 years of object recognition: directions forward	Andreopoulos and Tsotsos et al.[33]	2013	It s provides a review of object recognition systems evaluation over five decades
15	Object Detection in 20 Years: A Survey	Zhengxia et al. [51]	2019	Approximately 400+ papers review for this survey paper based on object detection. It covers the technical aspect of object detection for 20 years (from the 1990s to 2019)
16	Deep learning for generic object detection: a survey	Liu, Li et al. [52]	2020	covering many aspects of generic object detection
17	Our survey	--	2020	A comprehensive review based on generic object detection covers the last decade + current trends + future directions

- The ability of the deep neural network to express is far higher than that of conventional methods.
- The deep neural network can learn the hierarchy of features automatically directly from data using a multi-phase structure that represents a multi-level representation ranging from pixel to high-level semantic features.
- CNN architecture can provide improvements in several tasks such as bounding box regression and classification in a multi-task learning manner such as the one used in Fast R-CNN [18].

Image super-resolution reconstruction [70], [71], face recognition [5], image classification [6], [72], medical diagnosis, image retrieval [73], [74], pedestrian detection [75], [76] and

video analysis are some of research areas where deep learning is performing incredibly well.

## V. GENERIC OBJECT DETECTION

Generic object detection is the process of localizing an object using a rectangular bounding box to indicate the confidence of the object in the image and to classify the object with a label. The Generic Object Detector divided into two sub-categories named region proposal base detector and the regression/classification based detector.

The **Region proposal detector** follows the traditional method of object detection, first driving the region's proposed generation then classifying the regions into different

categories. R-CNN [10], SPP-net [77], Fast R-CNN [18], Faster R-CNN [9], R-FCN [78], FPN [79], and Mask R-CNN [80] are some of the example of region proposal framework. The **regression/classification base detector** takes object detection as a regression problem for locally separated bounding boxes and possible class probabilities. A single neural network predicts the bounding boxes and class probabilities directly from the whole images in one assessment. Classification and regression-based framework mainly comprises of different methods such as Multibox [81], AttentionNet [82], G-CNN [83], YOLO [19], SSD [84], YOLOv2 [85], DSSD [86], and DSOD [87]. The correlation between these methods shows in FIGURE 2.

#### A. REGION PROPOSAL OBJECT DETECTOR

The region's proposed object detection framework mimics the human brain's attention span. First, it scans the entire scenario and then focuses on the region of interest. Among the other mention related work, OVERRRFEAT [64] has the most promising performance. It was the first time CNN has been introduced in sliding window mode, which predicts the bounding box directly from the top of the highlighted feature map after gaining the confidence of the underlying object category.

##### 1) R-CNN

It was a time when deep architecture was used to significantly improve accuracy and high-level feature of candidates' bounding boxes. In 2014, Ross Girshick proposed an object detection model called R-CNN to solve these problems and achieved a 30% improvement over the proposed methods (DPM HSC [88]) on PASCAL VOC 2012.

The R-CNN model includes three modules, such as region proposal, extraction of deep CNN-based features, and classification/localization. The architecture of R-CNN is shown in FIGURE 3.

##### a: GENERATION OF REGION PROPOSAL

The R-CNN model used a selection search [89] to extract a region proposal and generate 2000 regions' proposals from a single image. The saliency indication and bottom-up grouping have been used to provide a faster selection of more accurate arbitrary size candidate boxes and to reduce the search space for object detection [22], [56].

##### b: DEEP FEATURE EXTRACTION BASED ON CNN

At this stage, the CNN module [59] uses the fixed size resolution wrap or crop region proposal to extract approximately 4096-dimensional features. Due to its high learning potential, dominant expressive power, and CNN's highly advanced architecture, the high level of semantic and robust features draw from each region's proposal.

##### c: LOCALIZATION AND CLASSIFICATION

At this stage, several region proposals are score as a set of positive regions and background as the negative region with

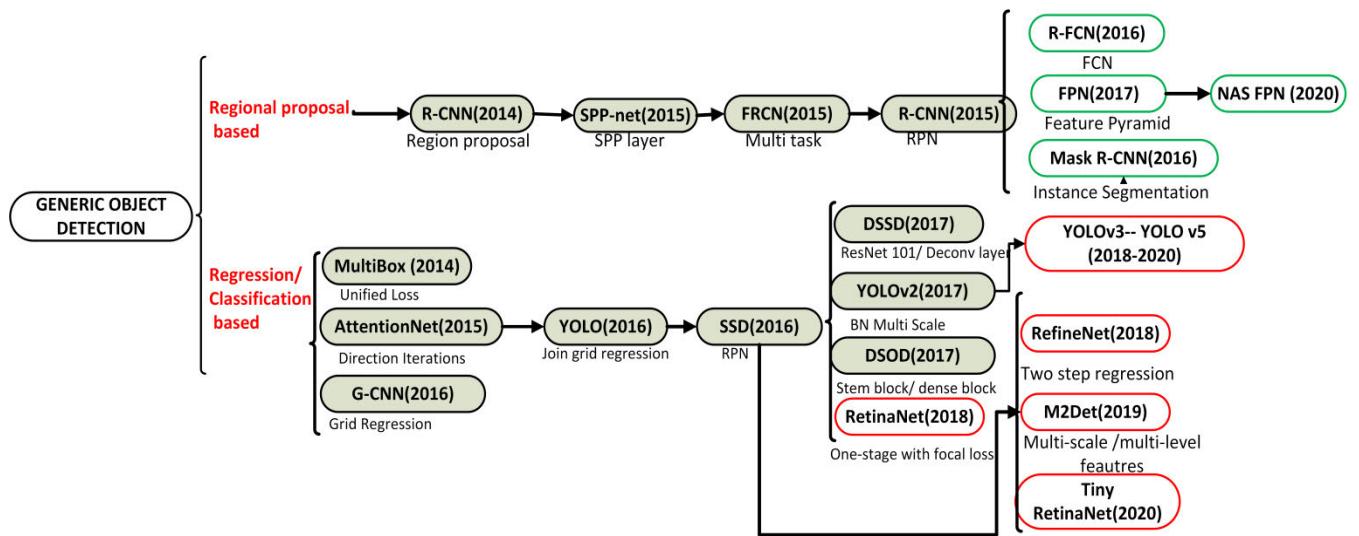
pre-trained category-specific linear SVM for various categories. The final object location is secured by adjusting and filtering the score regions using Bounding Box Regression (BB) and Non-Maximum Suppression (NMS), respectively. Typically, pre-trained models are used to solve the problem of insufficiently labeled data. Instead of using unsupervised training, R-CNN first performs the training process on a large auxiliary dataset such as ILSVRC and then implements a specific domain fine-tuning process for improvement.

*Asides from the significant use and improvement of CNN over traditional methods, there are still some gaps and disadvantages that need to be highlighted.*

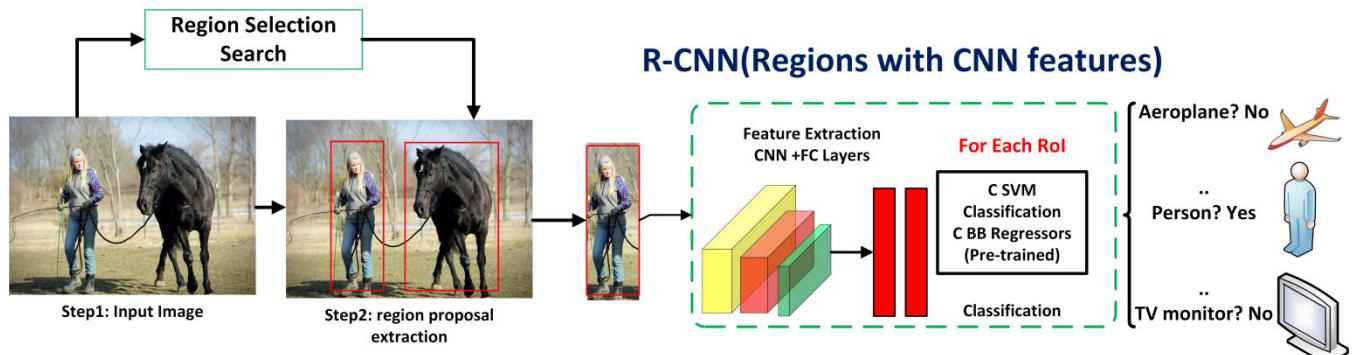
- The fully-connected layer (FC) requires a fixed-size input that directly leads to a re-computation of whole CNN for each region proposal and increases the test time.
- Multi-step training is R-CNN major drawback. Firstly, the Convolutional Network (ConvNet) requires fine-tuning for the region's proposal then apply fine-tuning to the Softmax classifier's learning, which replaces by SVM to fit in with ConvNet features, finally trained the bounding-box regressor.
- The R-CNN training phase is expensive in terms of time and space. It stores the extracted feature of each region proposal on disk. Even training small datasets take much time with deep networks like VGG16. The memory requirement for these datasets is also alarming.
- Region proposal generation using selection search is the time-expensive process that produces a large number of redundant regions.

Many strategies have been proposed to address these issues, such as MCG [90] form a multiple hierarchical segmentation by exploring the different scales of image and aggregate different regions to produce proposals. The traditional graph-cut approach was replaced by geodesic based segmentation in the GOP [91]. Edge box method [92] extracts the object with fewer contours straggling their boundaries in bounding boxes instead of producing distinct segments. However, DeepBox [93] and SharpMask [94] uses pre-extracted re-ranking to avoid un-necessary region proposals.

Furthermore, some of the researchers have solved the problem of incorrect localization through better strategies such as Gupta *et al.* [95] propose object detection based on semantic segmentation on RGB-D images. It uses geocentric embedding for pixel encoding on depth images. Object detection, combined with a super-pixel classification framework, gives promising results on semantic segmentation tasks. Zhang *et al.* [96] perform sequential bounding box regression using the optimization of the Bayesian-based search algorithm and penalized localization inaccuracies using trained class-specific CNN classifiers with structure loss. Ouyang *et al.* [97] propose a novel technique base on deformable CNN that imposes a geometric penalty on the deformation of various object parts along with deformation-pooling constrain (def-pooling).



**FIGURE 2.** State-of-the-art proposed methods for generic object detection shows in the figure. Generic object detectors have two categories RPN base detector and regression/classification based detector. However, BN(Batch Normalization) [61], FRCN (Faster R-CNN) [9], FCN(Fully Convolutional Network), Deconv layers( Deconvolution layers), SPP (Spatial Pyramid Pooling), RPN( Region Proposal Network) are some of the basic terms used in object detection. The upper branch of the flow diagram lists the two-stage detection base on region proposal network. In contrast, the lower branch contains the list of the one-stage detector (regression /classification).



**FIGURE 3.** Architecture of R-CNN: R-CNN uses selective search to generate region proposal, and CNN use to produce the features map. A region contains an object like shown with the help of a red square is called region proposal. Finally, a Computed feature map passes to the classifier like SVM to classify the region. 1) Input image 2) region proposal extraction 3) Extraction of CNN feature against each-region 4) classification of each-region with SVM [100].

## 2) SPP-Net

R-CNN uses wrapping and cropping operations at the suggestion of each region proposal for the fully connected layer that takes only a fixed size input image. Cropping operation can cause partial content loss of the desired object, and wrapping operation can produce geometric distortion. These content losses and distortion can decrease object detection accuracy, especially in the varying image scales. A novel CNN architecture based on the theory of spatial pyramid matching [98], [99] named SPP-net was proposed in [77] that removes the limitation of a fixed size network. SPP-net uses multiple standard scale-filters to perform the image partition into the number of divisions and aggregates the quantified local feature to produce mid-level representation.

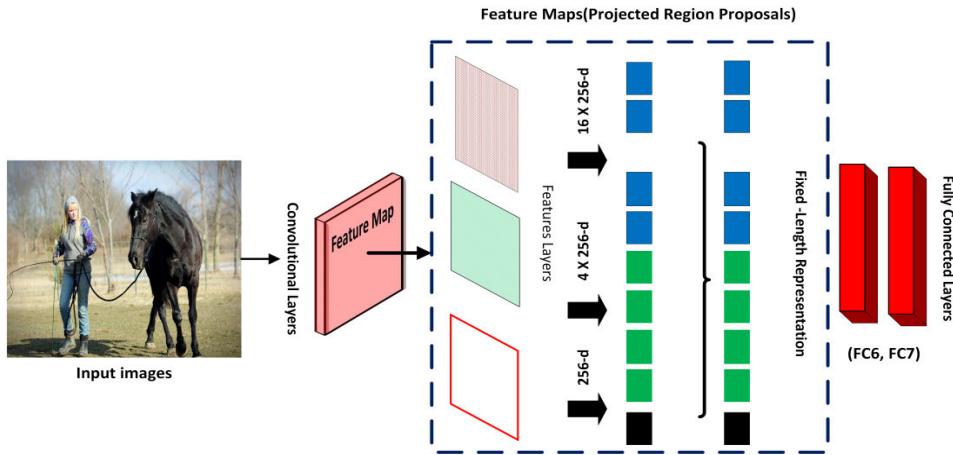
The architecture of SPP-net shown in FIGURE 4, which reuses the fifth Conv layer (conv5) feature maps to generate fixed-length feature vectors from the projection of arbitrary

size region proposal. The comparison between R-CNN and SPP-net is shown in FIGURE 5.

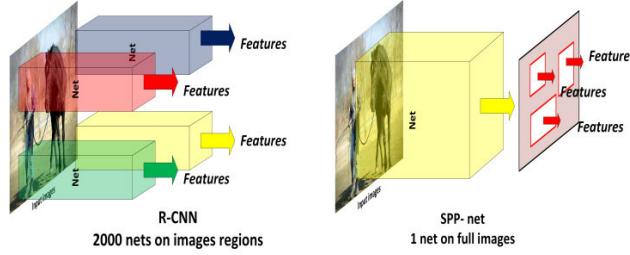
The local response strength and relationships with the spatial position of a feature map make it feasible for reusability [77]. The spatial pyramid layer (SPP layer) is stacked after the final-layer of the Conv layer in the architecture. If Conv5 has a three-level pyramid, then it has 256 features maps. The final feature vector of the region proposal has a dimension of 5376 after the SPP layer. The better result can be obtained from SPP-net with an accurate estimation of the corresponding scale of different region proposals. However, sharing computation costs can improve the efficiency over a testing period.

## 3) FAST R-CNN

However, SPP-net has shown impressive improvements in efficiency and accuracy in object detection over R-CNN.



**FIGURE 4.** SPP-net architecture [77]: an SPP-net (spatial pyramid pooling layer) insert between the FC layers and Conv layers. Conv<sub>5</sub> is the last layer contains 256 filters. SPP-net divides the input features into sub-images and extracts patch feature in each sub-image.



**FIGURE 5.** Comparison between R-CNN and SPP-net framework: R-CNN is very time-consuming because it computes the feature map of each region proposal separately for SVM. While in SPP-net, the Conv-layer computes fixed size features-map once for the entire image.

Still, it needs to be developed to meet storage space requirements during multi-stage pipelines such as extraction of features, network fine-tuning, SVM classifier and bounding box regressor training and fitting. Furthermore, unless the SPP layer causes an aggressive reduction in the accuracy of a deep network, the fine-tuning algorithm [77] does not update the conventional layer. Based on bounding box regression and multi-task loss classification, A novel CNN architecture proposes to overcome the problems mentioned earlier, called Fast R-CNN [18].

Fast R-CNN has the same architecture as the SPP-net, except for the use of the SSP layer of a single level pyramid, as shown in FIGURE 6. Fast-RCNN uses the Conv layers to generate the feature map by processing the whole image, and then use the pooling layer on ROI (Region of Interest) to extract the fixed size length feature vector of the region proposal.

These feature vectors fed two consecutive Fully-Connected layers before branching into two separate output layers. A layer is used for calculating softmax classification probabilities of C + 1 categories (C for object classes and an additional one for background), while other layers perform

refining of bounding box regression (four real-valued coordinates).

Multi-task loss is used to optimize the parameters in an end-to-end manner. A multi-task Loss for bounding box regression and classification is defined as follows:

$$L(p, u, t^u, v) = L_{cls}(p, u) + \lambda [u \geq 1] L_{loc}(t^u, v) \quad (1)$$

Log loss of ground truth calculates by  $L_{cls}(p, u) = -\log p_u$ . While  $u$  and  $p$  are driven from the discrete probability distribution  $p = (p_0, \dots, p_C)$  from the last FC layer over the  $C+1$  outputs. Predicted offset  $t^u = (t_x^u, t_y^u, t_w^u, t_h^u)$  use to evaluates  $L_{loc}(t^u, v)$ , where  $x, y, w, h$  denote the two coordinates of the bounding box center, width, and height, respectively. Each  $t^u$  adopts the parameter settings in [10] to specify an object proposal with height/width shift and scale-invariant translation in log-space. All background ROIs omitted by employed the inversion bracket indicator function  $[u \geq 1]$ .

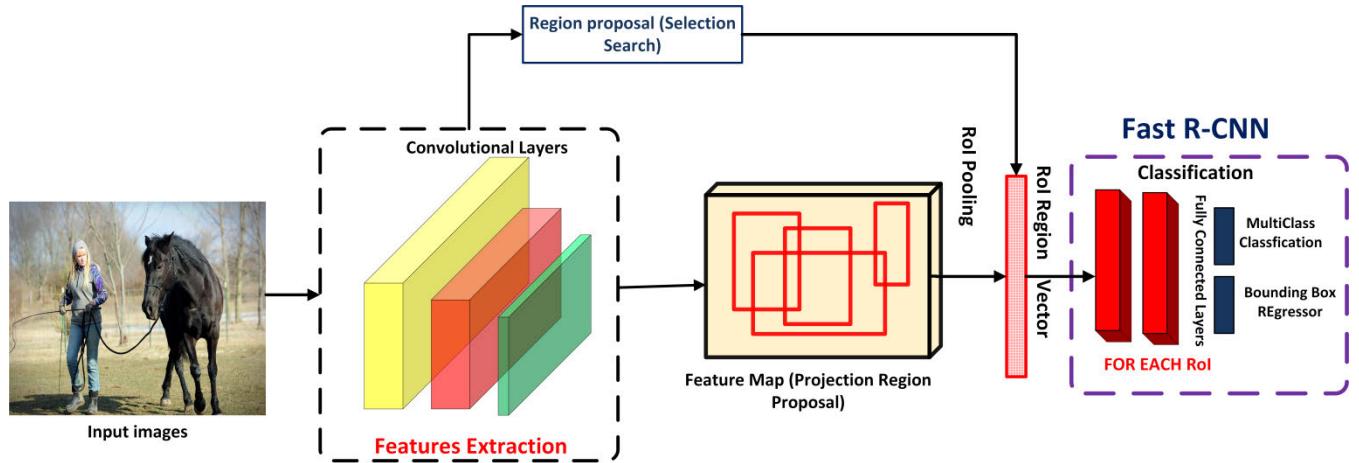
A smooth L<sub>1</sub> loss uses to fit the bounding box regressors properly and provides more robustness against outlier and eliminates the sensitivity in exploding gradients:

$$L_{loc}(t^u, v) = \sum_{i \in x, y, w, h} smooth_{L_1}(t_i^u - v_i) \quad (2)$$

where

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (3)$$

Back-propagation through the SPP layer on training instances (i.e., ROIs) from different images is inefficient. First, Fast-RCNN adopts the hierachal approach for mini-batches; it randomly sampled N different images, then each image sampled into  $R/N$  ROIs, where the number of ROIs is represented by  $R$ . Critically, the region of interest ROIs with the same-image shares the computations and memory in the backward and forward pass. In contrast, counting the FC layers requires an intensive amount of time during the forward-pass [18].



**FIGURE 6.** Fast R-CNN framework [18]: it consists of pre-trained CNN (train on ImageNet classification task), and an ROI pooling layer replaces the final pooling layer. While two branches replace final two FC layers: 1. softmax layer ( $K+1$  categories) 2. Bounding box regression branch.

The truncated Singular Value Decomposition (SVD) [101] can be used to accelerate the testing procedure and to compress the FC layers. Fast RCNN processes all layers of the network with multi-task loss and training in single-stage. It provides effective memory storage strategies and training schemes to improve accuracy.

#### 4) FASTER R-CNN

Most of the state of the art object-detection models are tied to region proposal generation methods such as Edge-Box and selective search, which hinders the improvement of accuracy. Ren *et al.* [9] proposed a model to address this issue by sharing the full image Conv features with detection networks called the Region Proposal Network (RPN). RPN can predict object bounding box and class confidence scores simultaneously using FC-network at each position. Analogous to [89], RPN generates proposals for rectangular object proposals set for randomly size images. RPN operates on the shared layers and specific Conv layers of an object detection network.

As FIGURE 7 shows the architecture of RPN, it is fully connected to the spatial window of size  $n \times n$  and slides over a Conv feature map. Each sliding window generates a low dimensional vector and is finally fed to two sibling FC layers, namely bounding box (BB) regression (reg) and classification layer (CLS). Complete architecture is the combination of  $n \times n$  Conv layer and two  $1 \times 1$  sibling Conv layers with the non-linear objective function (ReLU) in the output layer of  $n \times n$  the Conv layer. Comparing a proposal relative to bounding boxes (anchors) produces regression toward the true-bounding box. Faster R-CNN adopts three different scales and aspect ratios for detection. The loss function of Faster R-CNN is the same as (1).

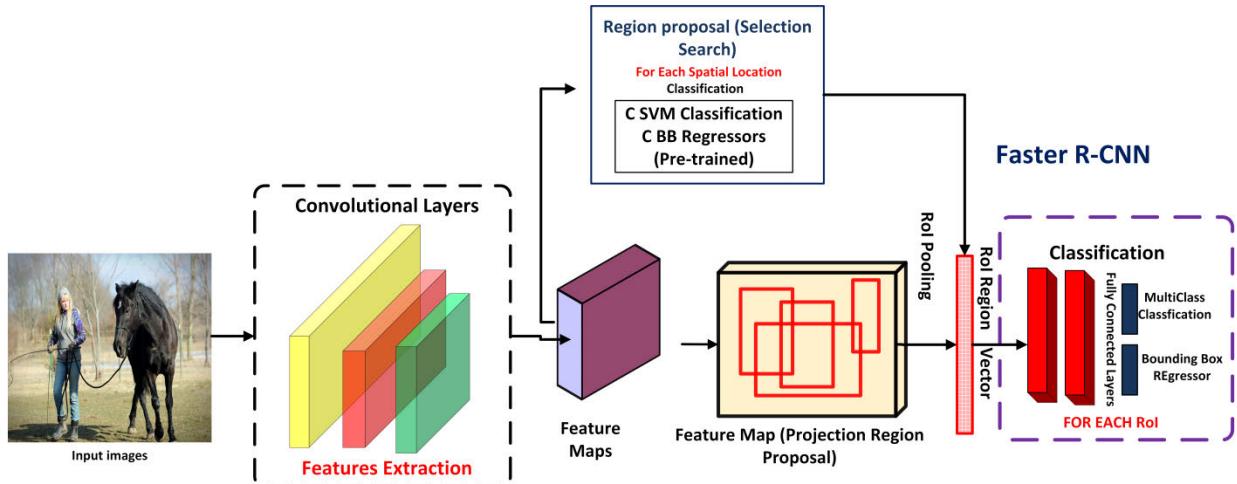
$$L(p_i, t_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \quad (4)$$

where,  $p_i$  is the predicted probability of  $i$ -th anchor being an object. The ground-truth label  $p_i^*$  equals to one (for positive anchor) otherwise zero.

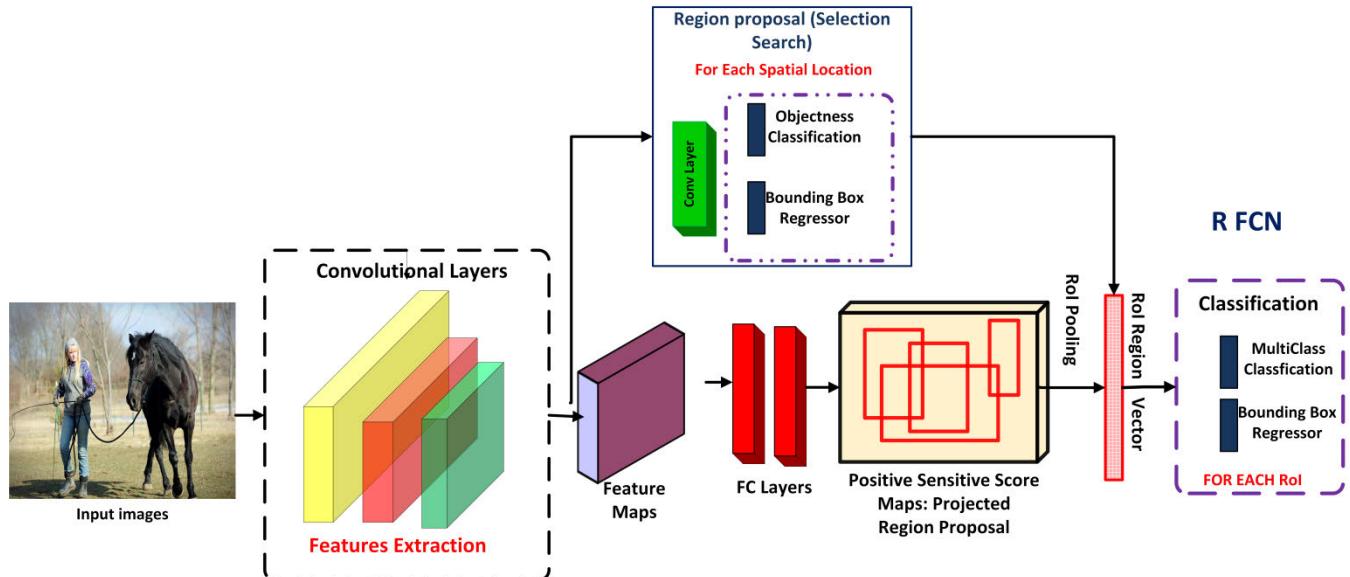
The predicted bounding box coordinates (Four parameters) are stored in  $t_i$  whereas  $t_i^*$  containing the information of positive anchor with overlapping to ground truth box. However,  $L_{cls}$ ,  $L_{reg}$  are binary log loss and smooth  $L_1$  loss similar to the (2). The losses normalize with the number of anchor locations ( $N_{reg}$ ) and mini-batch sizes ( $N_{cls}$ ) respectively. Use of back-propagation and SGD for end-to-end training of Faster R-CNN based on the fully- Convolutional network. With the invention of Faster R-CNN, all-region proposal base CNN networks are trained end-to-end manner. However, RPN produces regions that resemble objects (including backgrounds) rather than an object instance. It has difficulty dealing with extremely large or shaped objects.

#### 5) R-FCN (REGION BASE FULLY CONVOLUTIONAL NETWORK)

It is a deep network based on the ROI pooling layer, which is divided into two sub-networks, such as unshared ROI-wise subnetwork and shared fully-Convolutional subnetwork, which is independent of ROIs. This arrangement mimics the early proposed classification architectures (e.g., AlexNet [59] and VGG16 [18]), comprising of several Fully connected FC layers and Convolutional subnetwork that separated by specific spatial pooling layers. The new state-of-the-art classification networks is fully Convolutional, such as Residual Nets (ResNet) [65] and GoogLeNet [62], [102]. Therefore, a fully-Convolutional object detection network except ROIs- wise sub-network adapts these architectures and generates naive-solution [65]. Translation variance in object detection and translation invariance in image classification causes inconsistencies. Thus, the shifting of the object in the image does not affect the classification result, while any object translation in the bounding box has a robust and meaningful impact on the



**FIGURE 7.** The RPN in the framework of Faster R-CNN [9]. It aggregates the region proposal network with the CNN model. Faster –RCNN composes of RPN and fast-RCNN with share Conv-layers. Pre-defined anchor boxes represent as K, which are convoluted with each sliding window to produce vectors of fixed-length that is taken by classifier and regressor layer to obtain the corresponding output [9].



**FIGURE 8.** The Framework of R-FCN[78]: firstly, RPN generates candidates' ROI, which apply on a score map. Conv-layers use to create the feature map on the entire image. The computational cost of per-ROIs is negligible. FC layers introduce complexity in feature map (as presented by red columns).

object detection process. Translation invariance can be controlled by using manually inserting the RoI pooling layer into convolutions at the expense of additional unshared region-wise layers. So Li *et al.* [78] propose a fully convolutional region-based architecture, as shown in FIGURE 8.

The R-FCN network uses the Conv layer to produce a position-sensitive score map of size  $K^2$  with a fixed grid  $k \times k$  and to aggregate the score map response using the position-sensitive RoI pooling layer. Finally, the average of the position-sensitive score produces a  $C + 1 - d$  vector and computes classification across categories in each ROI. A class-agnostic bounding box is obtained by appending another  $4k^2 - d$  Conv layer. A more powerful classification network with fully-convolutional architecture can be used

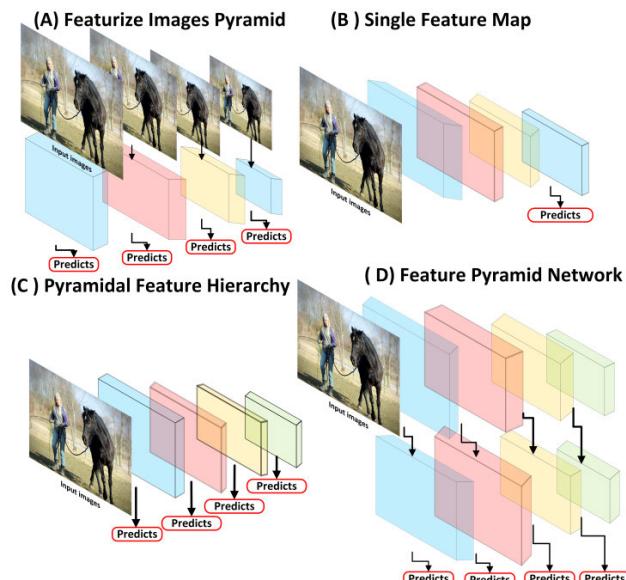
with R-FCN to accomplish object detection by sharing nearly all layers and obtained state-of-the-art results on both PASCAL VOC [103] and Microsoft COCO [104] datasets at a test speed of 170ms per image [105].

#### 6) FPN

As shown in FIGURE 9 (a), the scales invariance of object detection systems can be avoided by constructing feature pyramids on the image pyramids (featured image pyramids) [16], [77]. However, it rapidly increases memory consumption and training time. In some of the techniques, a single input scale is used to represent high-level semantics. In contrast, scale-variation can lead to an increase in robustness, as shown in FIGURE 9(b), and inconsistency between

train/test time increase due to the construction of the image pyramid during test time [9], [18]. As shown in FIGURE 9(c), the Deep ConvNet generates a feature map of various spatial resolutions using in-network feature hierarchy, and unusual depth introduces significant semantic gaps. Previously proposed methods have built feature pyramids from the middle layers and avoided using low-level features or sum transformed feature responses, and missing the higher resolution maps of the feature hierarchy.

However, FPN [79] architecture is based on the top-down pathway and bottom-up pathway. It combines low-resolution and semantically robust features with high resolution using several lateral connections, as shown in FIGURE 9(d). With the stride of 2, the down-sampling of feature maps produces feature hierarchy in the bottom-up pathway approach of forwarding backbone ConvNet. While in the top-down pathway approach, a reference set of feature maps is built by selecting the last layer of each network stage, which is a group of output maps of each fixed-size layer. Feature maps of higher network stages are un-sampled and enhanced using an authentic connection of the same spatial size from the bottom-up to build a top-down pathway. The channel dimensions have been reduced by appending a  $1 \times 1$  Conv layer to the un-sample map while element-wise addition using for emergence. The final feature map is generated by adding Conv  $3 \times 3$  to each merged-map and thereby reduces the aliasing effect. The most exceptional resolution map is obtained using multiple iterations.



**FIGURE 9.** The central concept of the Feature Pyramid Network (FPN). (a) Feature pyramid builds by using the image pyramid (Slow process) (b) For faster detection, only single scale features are used. (c) The ConvNet is used to compute the pyramidal-feature; it reused as an alternative to the image feature pyramid. (d) (b and c) both integrated into FPN. feature map is shown by blue outline whereas thicker outline used for semantically robust features [79].

Finally, the feature pyramid of all levels of rich semantics and scales is extracted that is trained end to end like

this state-of-the-art representation can be achieved without compromising memory and speed. Meanwhile, FPN does not use CNN architecture as the backbone and apply to different object detection stages (such as region proposal generation) and many other computer vision tasks (e.g., instance segmentation).

### 7) MASK R-CNN

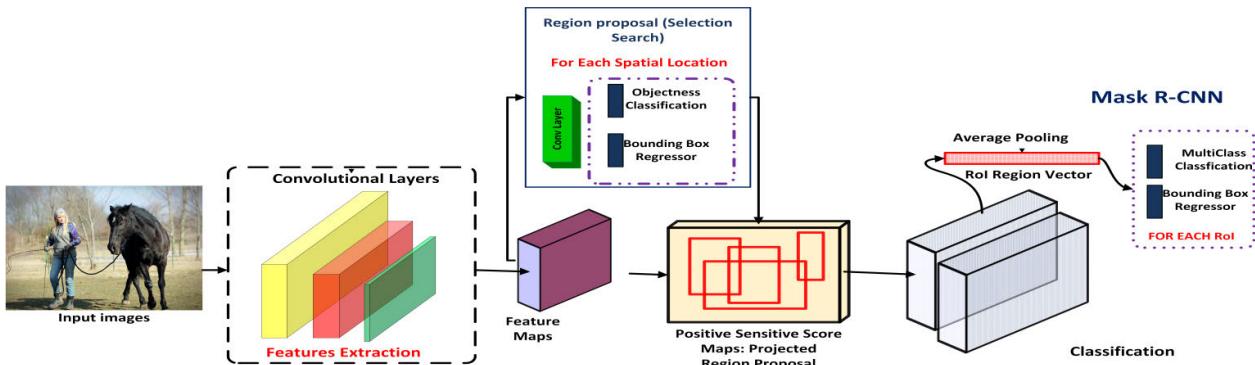
Instance-segmentation is a challenging task that consists of two independent functions, such as object detection and instance segmentation (Semantic segmentation [106]) in the image. While the Mask R-CNN uses an additional branch specifically for pixel-to-pixel segmentation mask prediction, parallel to the existing two branches (classification and bounding box regression prediction), similar to Faster R-CNN as shown in FIGURE 10 [80].

The segmentation mask branch maintains the explicit-object spatial layout encodes into the  $m \times m$  mask. With fewer parameters, this fully-convolutional architecture is more accurate than the model used in [106]. In Mask R-CNN, the multi-task loss is a combination of segmentation mask branch loss, classification, and bounding box regression loss. The loss of classification is related to the class ground-truth, while the prediction of the category depends on the branch of classification. RoI pooling is the core operation of Faster R-CNN that produces standard local quantization to extract features and introduces misalignment between features and RoI. It affects the classification results due to small translation robustness and has a significant negative impact on the pixel to pixel mask prediction. The Mask R-CNN uses the ROIAlign layer, which is free and straightforward from quantization to preserve the explicit per-pixel spatial correspondence. ROIAlign is obtained by replacing the Harsh quantization of RoI pooling with bilinear interpolation [107], and the input features values are extracted at quart regularly sampled locations computed in each RoI bin.

Regardless of its simplicity, the mask accuracy can be improved with minor changes under strict localization metrics. An additional mask branch with the Faster R-CNN model can assist in other object detection tasks with a small computational burden. Mask RCNN is an efficient and flexible framework that generates precise instance segmentation and object detection. It can easily be generalized to perform other tasks with minimal modification, such as human pose estimation [4]. It was the first time that Mask R-CNN used for scene instance segmentation and provide intelligent driving [108], while ensemble approaches can be applied for medical segmentation applications [109].

### 8) OTHER PRACTICAL WAYS TO DETECT OBJECTS

The previously proposed networks yield promising results, but it is struggling to localize small objects due to limited candidate box information and rough feature map. These phenomena become dramatically worse when dealing with the Microsoft COCO dataset, which consists of less prototypical images and objects with various scales that require more



**FIGURE 10.** An efficient framework (instance segmentation): Mask R-CNN [80]. It has two stages; the first stage generates region proposal of the object and second predicts the class, refine the BB, and create the pixel level mask. Both phases connect to the backbone structure.

precise localization. This issue can be tackled by gathering complementary information from multiple sources through multi-task learning [110], multi-scale representation [111], and context modeling [112].

**Learning of Multitask** is the process of determining the adequate representation of multiple correlated tasks in the same [113], [114]. StuffNet made a reasonable effort to accurately identify small objects using trained Conv features for 'stuff' such as amorphous categories (ground and water) and object segmentation [110]. Dai *et al.* [106] propose a three-phase multi-task network to address this issue, called regional instance classification, instance segmentation at the pixel level, and class-agnostic region proposal generation. Li *et al.* [115] suggest a multi-stage architecture based on region-based object detection and learn the segmentation features using weakly-supervised object segmentation cues.

**Multi-scale representation** combines multi-layers activation with skipping-connection to use the semantic information of different spatial resolutions [79]. Yang *et al.* [25] were used various scale-dependent features to investigating layer-wise cascaded rejection classifier (CRC) and scale-dependent pooling. Cai *et al.* [116] proposed MS-CNN that uses multiple scale-independent output layers to avoid instability between object size and respective fields.

**Contextual modelin** uses to improve detection efficiency. It uses features of or around the Region of Interest (RoI) of various support regions and resolutions to overcome the concerns of occlusions and local similarities. Zhu *et al.* [117] proposed a model called SegDeepM, which used the Markov Random Field as well as object segmentation to minimize reliance on initial candidate boxes. Zeng *et al.* [118] introduced a gated function to control message transmission in various support areas and propose a novel GBD-Net based on message transmission.

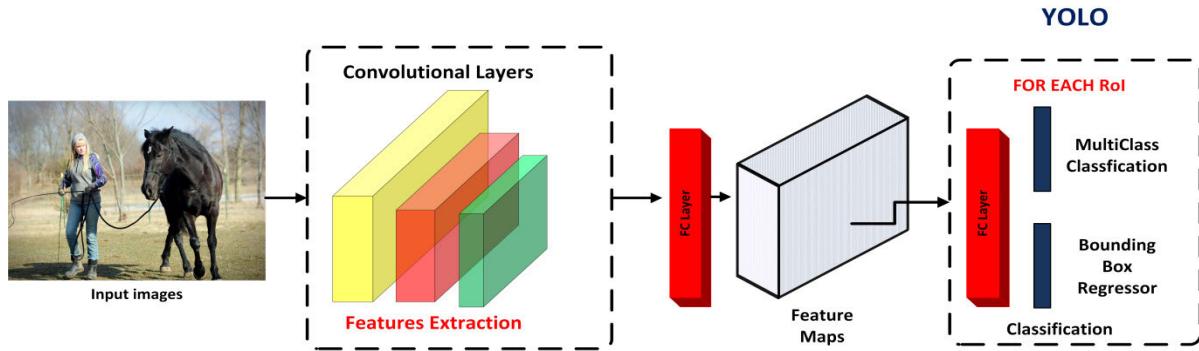
#### B. REGRESSION/CLASSIFICATION OBJECT DETECTOR

The region proposal base framework includes various correlated phases such as region proposal generation, feature extraction using CNN, Bounding Box (BB) regression, and classification, which trains separately. An alternative train-

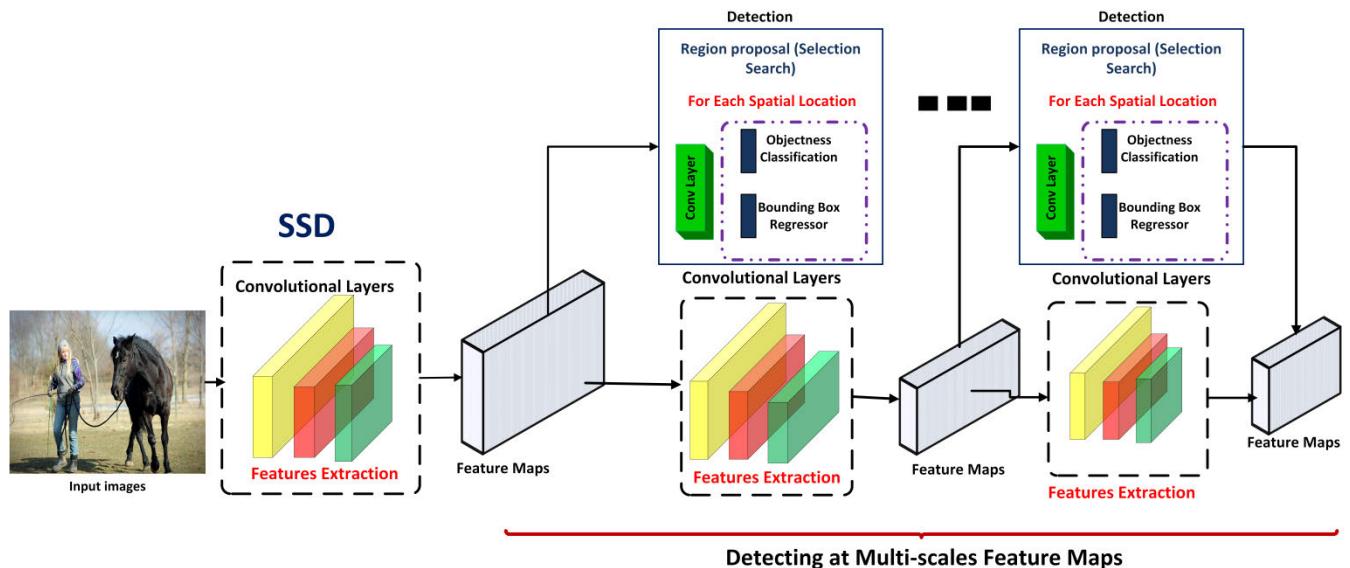
ing requires the development of share convolution parameters between the detection network and the RPN, which is used in the end-to-end module of Faster R-CNN. In real-time applications, the time spent handling different components becomes a hindrance. Fortunately, the time required for the object detection task is reduced with the invention of single-stage frameworks based on class probabilities, mapping directly from image pixel to BB coordinates, and global regression/classification. In this section, some pioneering one-stage object detectors (Convolutional architecture) are discussed, such as YOLO (You Only Look Once) [19], Single Shot MultiBox Detector (SSD) [84], RetinaNet, RefineNet, M2Det, and DSSD, etcetera.

Significant efforts have been made to improve the object detection models as regression/classification tasks. D. Erhan *et al.* [119] has used CNN-based regression to detect objects by developing test image binary masks and bounding box inference for extracted objects. Even so, locating the overlapping objects and using up-sampling to produce a bounding box is a difficult task. The author proposes a CNN model for object detection based on two parallel branches, as the first branch generates a class agnostic segmentation mask. In contrast, the object center is based on the likelihood of predicting the patch given in the second branch. The performance of the model is efficient because the class score and segment are obtained in the same model, which has mostly joint CNN operations. Yoo *et al.* [82] proposed an iterative end-to-end CNN model for object detection, called AttentionNet. AttentionNet generates a quantized weak direction for a target object and coverage to an accurate object bounding box with an ensemble of iterative prediction starting from top-left and bottom-right corner of an image. The efficiency of the model is quite disappointing when handling multiple categories of the object with the following two steps procedure.

Najibi *et al.* [83] proposed iterative proposal-free grid-based object detector (G-CNN) from the fixed grid to boxes tightly surrounding the objects based on extreme-scale. G-CNN trained the regressor to move through a repetitive process, and scale grid elements towards the target-object begin from a fixed multi-scale bounding box grid. However,



**FIGURE 11.** The main idea behind the YOLO (You Look Only Once) [19]: the architecture of YOLO has 24 Conv-layers, followed by two FC layers. Alternatively,  $1 \times 1$  Conv-layers reduce feature space from preceding layers. The Conv-layers are pre-trained on ImageNet classification task at half resolution and double the resolution for detection. First block use for Conv-layers, while FC-layers present as a red column in the diagram.



**FIGURE 12.** The architecture of SSD300. Prediction of offset to default anchor boxes and their confidence scores uses multiple layers with backbone VGG16. But it discards the FC layers. Instead of using standard FC layers of VGG16, it uses auxiliary convolutional layers. NMS is conducting on multi-scale refined bounding box for the final detection [84].

small or very overlapping objects are challenging to detect using G-CNN.

### 1) YOLO: YOU ONLY LOOK ONCE

Redmon et al [19] proposed a novel one-stage object detector, predicting the bounding box that uses the topmost-feature map and a direct evaluation of class probabilities. The idea behind YOLO is to divide the image into  $S \times S$  grid cells, and each grid cell is responsible for predicting the center of the object in the grid cell, as shown in FIGURE 10.

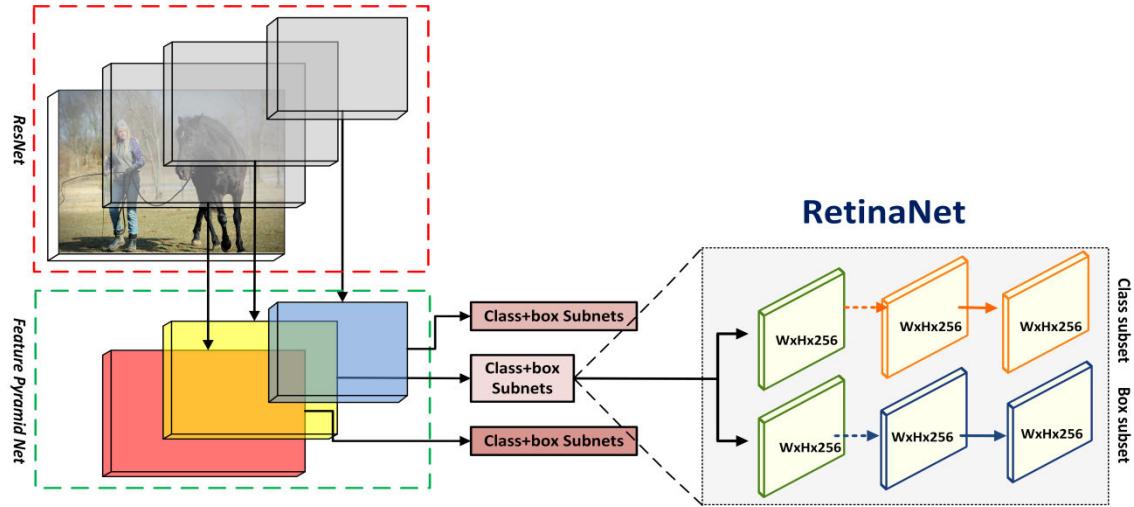
However, it predicts the Bounding box  $B$  and its corresponding confidence scores. The confidence score indicates the probability that an object is present in the grid which defines as,  $\Pr(\text{Object}) * \text{IOU}_{\text{pred}}^{\text{truth}}$  such that  $\Pr(\text{Object}) \geq 0$  and  $\text{IOU}_{\text{pred}}^{\text{truth}}$  indicates the confidence of its prediction. Regardless of the number of binding boxes, the probabilities

of a conditional class ( $\Pr(\text{Class}_i | \text{Object})$ ) are predicted in each grid cell. It should notice that it only considers the contribution of grid cells that contain objects. The confidence score of a particular class is a product of individual box confidence predictions and probability of conditional class at the testing time, which explains the following:

$$\begin{aligned} \Pr(\text{Object}) * \text{IOU}_{\text{pred}}^{\text{truth}} * \Pr(\text{Class}_i | \text{Object}) \\ = \Pr(\text{Class}_i) * \text{IOU}_{\text{pred}}^{\text{truth}} \end{aligned} \quad (5)$$

However, predicted box and existing probabilities of class-specific objects in BB are in focus for fitness between objects. Loss function optimization during training is defined as follows:

$$\lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{abj} [(x_i - \hat{x}_i)^2]$$



**FIGURE 13.** RetinaNet utilized ResNet-FPN as a backbone network to predict different sized objects[122]. The author uses the Conv-net feature hierarchy in a pyramidal shape. To make feature pyramid with strong semantic at all scale, the author combines the low- resolution features with high resolution through a top-down pathway and lateral connection.

$$\begin{aligned}
 & + (y_i - \hat{y}_i)^2] + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{abj} [\left( \sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 \\
 & + \left( \sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2] + \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{abj} (C_i - \hat{C}_i)^2 \\
 & + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{noobj} (C_i - \hat{C}_i)^2 \\
 & + \sum_{i=0}^{S^2} 1_i^{abj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2
 \end{aligned} \quad (6)$$

The subscript  $i$  representing the  $i$ th grid cell that point to a center of relative bounding box denotes as  $(x_i, y_i)$ , while  $(w_i, h_i)$  is a normalized height and width relative to image size,  $C_i$  is a confidence scores, where  $1_i^{abj}$  indicates the existence of objects and  $j^{th}$  bounding box predictor use for prediction represent as  $1_{ij}^{abj}$ . If an object is included in the grid cell, then the Loss function is penalized for classification error. However, the predictor penalizes bounding box coordinate and ground truth box errors (i.e., the highest IoU of any predictor in that grid cell achieved). The architecture of YOLO compose of twenty-four Conv layers and two FC layers; some of the Conv layers construct ensembles of inception modules with  $1 \times 1$  reduction layers followed by  $3 \times 3$  Conv layers.

In real-time, the model can process 45 FPS images, while other versions of YOLO can process 155 FPS with much better results than other real-time object detectors. Furthermore, YOLO can collaborate with Fast R-CNN and produces less FP (false positive) on the background. Several powerful strategies, such as dimension clusters, Batch Normalization, anchor boxes, and multi-scale training, were adapted to develop an improved version of YOLO [85]. Detecting real-time objects are very challenging due to the limited memory and computation power. To address these challenges, QI-CHAO et al. [120] suggested a lightweight network based on Darknet-53, with a Multi-scale feature pyramid for multi-scale detection object called Mini-YOLOv3.

## 2) YOLOv2

This framework is the second release of YOLO [19], which provides impressive improvements in speed and precision by adopting a series of design decisions for previous work [85].

**BATCH NORMALIZATION:** It is unreasonable to normalize the entire training set as SGD uses mini-batches to estimate the mean or variance of each activation function during training. Finally, it sampled the element of each mini-batch in the same distribution called the BN layer [61]. In YOLOv2, the BN layer is added before each convolutional layer for convergence and regularity. The use of batch normalization has increased mean AP by 2%.

**HIGH-RESOLUTION CLASSIFIER:** Backbone classifier has increased the input resolution from  $224 \times 224$  to  $448 \times 448$  in the detection process. To solve the problem of input resolution variation, YOLOv2 has included a fine-tuning process in the classification network for ten epochs on the ImageNet dataset, which increases the mAP by up to 4%.

**CONVOLUTIONAL WITH ANCHOR BOXES:** YOLO uses Fully-connected layers to generate the coordinates of the predicted boxes. However, in Faster R-CNN, the anchor boxes are used as a reference to generate the offset of predicted boxes. YOLOv2 adopts a high-speed R-CNN prediction mechanism for class prediction and objectness for every anchor box and removes the FC layers, increasing the recall by 7% while mAP decreases by 0.3%. YOLOv2 uses K-mean clustering on the bounding box of the training set for better detection, while Faster R-CNN empirically identified the size and aspect ratio of anchor boxes.

**FINE-GRAINED FEATURES & MULTI-SCALE TRAINING:** High- resolution feature maps can provide useful information for localizing small objects. YOLOv2 combines the low-resolution feature with high-resolution features by stacking adjacent-features across different channels, such as identity mapping in ResNet. The network can predict

detection to varying resolutions by randomly selecting image dimension size (320, 352, ..., 608) after every ten batches. YOLOv2 achieved 78.6% mAP and 40FPS on high-resolution detection in PASCAL VOC 2007.

A Novel backbone framework, DarkNet-19, proposes for YOLOv2. The backbone architecture consists of 19 convolutional layers and five max-pooling layers, which provide high accuracy and require minimal operations to process the image. The YOLOv2 has 78.6% mAP and 40FPS, while Faster R-CNN with ResNet backbone has 76.4% mAP and 19FPS, and SSD500 has 76.8% mAP and 19FPS.

### 3) YOLOv3

YOLOv3 has some improvement over YOLOv2, such as YOLOv3 uses independent logistics classifiers for multi-label classification for more complex datasets containing many overlapping labels [121]. In YOLOv3, three different scale feature maps are used to predict of the bounding box. At the same time, predicting 3D tensor encoding class, objectness, and bounding box base on the last convolutional layer. YOLOv3 suggests another profound and robust feature extractor called Darknet-53, inspired by ResNet.

Experimental results show that YOLOv3 (AP: 33%) is three times faster than DSSD (AP: 33.2%) but slower than RetinaNet (AP: 40.8%) on MSCOCO dataset and metrics. However, the old detection metric of mAP at IOU=0.5, YOLOv3 has 57.9% mAP, while in DSSD500 and RetinaNet [122], it is 53.3% and 61.1%, respectively. YOLOv3 can perform better for detection of a small object due to multi-scale predictions compared to medium and more massive sized objects.

### 4) SINGLE SHOT MULTIBOX DETECTOR (SSD)

YOLO has difficulty dealing with a generalization of objects in unusual aspect ratio/ configuration, and multiple down-sampling operations produce standard features. Due to the strong influence of spatial constraints on the prediction of the bounding box, It also struggles to detection a small object.

To address these problems, Liu et al [84] proposed a model inspired by MultiBox adopted anchor [81], RPN [9], and multi-scale representation [111], called Single Shot MultiBox Detector (SSD) to address these problems. SSD uses specific feature maps for detection instead of the default grid that is used in YOLO; SSD achieves better performance due to the ratio of different aspect ratio, a set of default anchor boxes, and scales to discretize the output space of bounding boxes.

SSD can handle objects of different sizes by combining the predictions of multiple feature maps with different resolutions. The architecture of SSD consists of a VGG16 backbone network with numerous feature layers for predicting default boxes offset of various scale and aspect ratio with their corresponding confidence scores at the end of the system. A weighted sum of Softmax (e.g., confidence loss) and Smooth L1 (e.g., localization loss) use for network training. NM is applying on multi-scale refined bounding boxes to get a final detection result.

SSD significantly performs three times faster than Faster R-CNN on PASCAL, VOC, and COCO by intelligently integrating with data augmentation, a large number of default chosen anchor boxes, and hard-negative mining. The SSD300 uses image size  $300 \times 300$  use in SSD300, which runs at 59 frames per second, and is faster and more efficient than the YOLO.

However, SSD yields the worst results when dealing with small objects. While Improve feature extractors backbone frameworks such as ResNet101 and additional large-scale context using some deconvolution layers with skip connections [86] and improve network structures such as Dense Block [87], and Stem-Block can be used to address this issue. Although, much useful research has been conducted since the invention of SSD, such as Cheng *et al.* [86] proposed an encoder-decoder hourglass structure to detect the object to pass contextual information before prediction called DSSD (Deconvolutional Single Shot Detector). It introduces a large-scale context in object detection by combining ResNet101 (as the backbone) with some deconvolution layers (to solve the problem of shrinking resolution of feature maps on CNN). Cheng *et al.* [123] proposed the Inception Single Shot Multi-Box Detector(I-SSD) with a new inception block inspired by GoogLeNet Inception block and the deep residual network; improve accuracy without increasing the complexity of the model and affecting its speed.

### 5) DSSD

Deconvolutional Single Shot MultiBox Detector is a modified version of the SSD that has two additional modules, such as the deconvolutional module and the prediction module [86].

Each prediction layer contains the residual block in the prediction module then adds the output of the residual-block and prediction layer by factor. The Deconvolutional block strengthens features by increasing the resolution of the feature maps. After a prediction module, each deconvolutional layer is used to predict objects of various sizes.

Initially, the author uses a pre-train Renet101backbone network on the ILSVRC CLSLOC dataset in the training process then performs the actual SSD network training on the detection dataset of  $321 \times 321$  inputs or  $513 \times 513$  inputs sizes. Finally, freeze the weights of the SSD module with the train deconvolution module. Experimental results show the improvement of the DSSD513 model on both PASCAL VOC and MS COCO datasets. However, the deconvolution module and prediction module improved the PASCAL COV 2007 test dataset by 2.2%.

### 6) RETINANET

Lin et al [122] proposed a unified object detector with a novel classification loss function called Focal Loss.

The R-CNN has two separate phases; a set of region proposals is generated in the first phase, while each candidate location is classified in the second phase. A two-stage object detector can perform better than a one-stage object detector

because it produces a dense set of candidate locations and filters out the majority of negative-locations. The extreme foreground-background class imbalance is the main reason when network training converges in the one-stage detector. Therefore, the proposed loss function called focal loss can minimize the weight-loss assigned to easy or well-classified examples.

In the training process, focal loss avoids a large number of simple negative cases and concentrates on the hard training examples. By training unbalanced positive and negative instances and inheriting the speed of a previously proposed one-stage detector, the RetinaNet substantially eliminated the disadvantages of one-stage detectors.

Experimental results show that RetinaNet has a 6% improvement in AP with Resnet-101 FPN as compared to DSSD513 on the MS-COCO test dataset. With ResNeXt-101-FPN, RetinaNet has improved the AP by 9%. RetinaNet shows notable improvements in detection precision on small and medium objects by large margins.

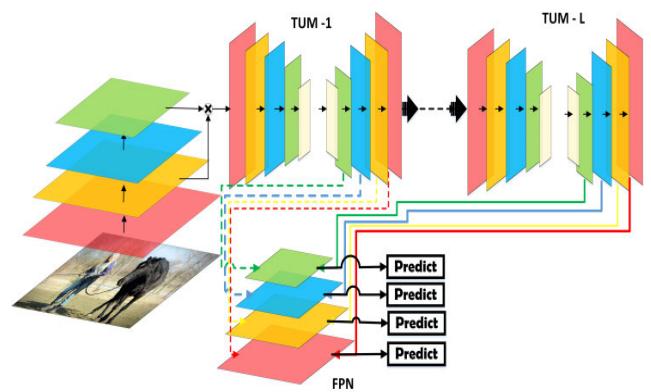
## 7) TINY RETINANET (REAL-TIME DETECTION)

Chang et al [124] proposes a novel one-stage detector with MobileNetV2-FPN as a backbone (feature extractor). Its architecture consists of Stem block backbone network and SEnet, followed by two subnets with a specific task. It improves accuracy and reduces the information loss. It uses the RetinaNet focal loss as a classification loss. A model is tested on PASCAL VOC 07/12 with 71.4%mAP and 73.8%mAP, respectively.

## 8) M2Det

Zhao et al. [125] suggested a multi-level feature pyramid network (ML-FPN) that develops a more compelling feature pyramid to overcome the issue of scale variation across object instances. The working principle of the model is based on three main steps to achieve the final incremental feature pyramid. In the first step, Multi-level features extracted from multiple layers in the backbone are fused as a base feature. The base feature is fed into a block consisting of two modules, namely Thinned U-shape Modules, and the Feature Fusion Modules jointly, and obtains the decoder layers of TUM as the features for the next step. Finally, decoder layers of equivalent scale are integrated to construct the feature-pyramid consisting of multilevel-features. So far, multi-scale and multi-level features have been developed. The rest of the network follows the SSD architecture to achieve the results of classification and bounding box localization in an end-to-end manner.

The M2Det, one-stage detector with VGG backbone, achieves 41.0% AP at 1.8FPS speed with a single-scale inference strategy and 44.2% AP with multi-scale inference strategy on MS COCO test-dev dataset. It performed 0.9% better on the RetinaNet800 but twice as slow as the RetinaNet800. The multi-level feature pyramid network used in M2Det is shown in FIGURE 14.



**FIGURE 14.** M2Det style Feature Pyramid network. Feature pyramid is constructed by using FFMv2, FFMv1, thinned U-shape encoder and decoder network followed by scale-wise aggregation module [125].

## 9) REFINE-DET

RefineDet [126] consists of two interconnected modules, the refinement module, and the object detection module. The transfer connection block is used between modules to transfer and enhance features from former to latter modules to better object prediction. The end-to-end training process involves three stages, such as pre-processing, detection (two inter-connected modules), and NMS. The one-step regression method is used in classic one-stage detectors such as SSD, YOLO, and RetinaNet to achieve final results. The two-step cascade regression method can better predict hard objects, especially small objects and more precise locations of objects.

## 10) OBJECT AS POINTS

Although the image classification area has recently become less active, object detection research is not yet mature. In 2018, a paper entitled “CornerNet: Detecting Objects as Paired keypoints” introduced a new perspective on detector training [127]. Since preparing anchor box target is a daunting task, is it really necessary to use them as before? This new trend of digging anchor boxes is called “anchor free” object detection. Corner box supports boundary box regression using heat maps produced by box corners. The scheme is inspired by the Hourglass network, which uses heat maps to estimate human suffixes. The object center is described using a heat-map, and the network regresses the box height and width of the box directly from these centers.

The CornerNet is using each pixel as a grid cell. With the help of Gaussian distributed heat maps, it is easier to exchange training than previous attempts to register the bounding box size directly. The elimination of anchor boxes is also effective as previously detector relay on IOU between ground truth and anchor box to assign training targets. Some of the neighboring anchors may get a positive target for the same object and network to learn the multiple anchors for the same object. Non-maximum suppression is (the greedy algorithm) used to fix this issue. Now we have one peak per object in the heat-map by eliminating anchors. Since NMS is

sometimes difficult to implement and slow to run, getting rid of NMS is a waste of resources. One big advantage is that it operates in a variety of environments with limited resources.

### 11) EFFICIENT-DET

Efficient-Det is an exciting development in the object detection area [128]. This research proved that the FPN structure is a powerful technique to improve the detection of network performance at various scales. Different flavours of FPN seen in YOLOv3 and RetinaNet before applying regression and classification. Plain-layer FPN structure may benefit from more design optimization in NAS-FPN and PANet. A new structure of an FPN called BiFPN is proposed in Efficient-Det. BiFPN allows the feature aggregation back and forth by adding cross-layer connections. It removes some useless parts from the architecture from the original PANet to justify the efficiency of the network. Weight feature fusion and additional learnable weight to feature aggregation are also innovated to improve the efficiency of a network over FPN. It also introduces a principle way to scale an object detection network. It has the same accuracy as YOLO v3 while having much fewer FLOPs.

### LATEST DETECTORS:

**Relation Network for Object Detection:** Hu et al. [129] propose that Relation object detection network includes an adapted attention module that considers the interaction between different targets in an image, including geometry information and physical feature. The relation module is used in the head of the detector before fed to classifier and regressor to produce more enhanced features for accurate classification and localization. It replaces the NMS post-processing step to gain higher accuracy than NMS. The performance of backbone networks such as Faster R-CNN, FPN, and DCN on the COCO test-dev dataset may increase efficiency by 0.2, 0.6, and 0.2%, respectively.

**DCNv2:** Dai et al. [130] propose a deformable convolutional network(DCN) that adapts geometric variation that reflects in the productive spatial support region of target for learning. ConvNets can only focus on the features of the fixed square size (according to the kernel); thus, the corresponding field does not adequately cover the entire pixel of a target object to represent it. To overcome this issue, the deformable ConvNets can produce deformable kernel, and the offset from the fixed size initial convolution kernel is learned from the networks. However, deformable RoI pooling is also useful for the localizing objects of different shapes. A deformable ConvNet can produce 4% higher accuracy than three plain ConvNets. It has a 37.5% mAP (mean Average Precision) under strict COCO evaluation criteria. DCNv2 uses more layers than DCNv1. The learnable scalar is used to modulate all deformable layers, which enhance the accuracy and deformable effects. The feature mimicking is used to improve detection accuracy by incorporating a mimic feature loss to the per-RoI feature of DCN, which is similar to useful features extracted from crop images. Experimental results show that DCNv2 [131] with strong backbones achieved a 5%

improvement in mAP over DCNv1 on the COCO2017 test-dev dataset under the strict evaluation criteria of MSCOCO.

**NAS-FPN:** A new feature pyramid architecture is found when the authors from Google Brains adapt neural architecture search, named NAS-FPN, which provides top-down and bottom-up connections for feature fusion of different scale [132]. It repeats the FPN architecture N times and concatenates them in the form of monumental architecture during the search phase, it imitates by picking arbitrary level features using high-level feature layers. Most of the significant efficient architectures use the connection between high-resolution input feature map and output layer to generate high-resolution features to identify small objects. Adopting high-capacity architecture, stacking more pyramid networks, and adding feature dimensions significantly increases detection accuracy. Experimental results show that the mean average precision of NAS-FPN increases up to 2.9% on the COCO test-dev dataset over the original FPN by adopting ResNet-50 as the backbone of 256 feature dimensions. NAS-FPN can achieve 48.0% mAP on the COCO test-dev dataset by utilizing an excellent backbone like AmoebaNet and stacked seven FPN of 384 feature dimensions.

## C. BACKBONE CNN ARCHITECTURE

Some CNN models are used as the backbone in the detection frameworks, such as AlexNet, ZFNet, VGGNet, GoogLeNet, Inceptionseries, ResNet, DenseNet, and SENet explains in Table 3. A survey of recent advances in CNN architecture can be found in Gu et.al [133]. The current trend suggests that increasing layer depth could improve the strength of CNN architecture representation, such as AlexNet has eight layers, and VGGNet16 [63] has 16 layers. In contrast, some dense network architecture has 100 layers, such as ResNet and DenseNet. Some architectures such as AlexNet [59], the ZFNet [134], and VGGNet have a large number of parameters despite being few layers deep since the large fraction of the parameters come from the fully connected layers. Recent developments at CNN show that new architectures such as Inception, ResNet, and DenseNet have great depth with a fewer number of parameters, avoiding FC layers. The number of parameters in GoogLeNet has been dramatically reduced with the use of carefully designed topologies of Inception modules [62] as compared to AlexNet, ZFNet, or VGGNet. Similarly, ResNet won the ILSVRC2015 classification task using skip connection for learning profound networks with hundreds of layers. InceptionResNets [135]combines the Inception networks with shortcut connections, which can significantly speed up network training. Huang et al. [136] proposed an architecture that extends ResNet under the name DenseNet, which consists of dense block integrated into feed-forward fashion, providing some compelling benefits such as feature reuse, parameter efficiency, and implicit deep-supervision. Recently, He et al. [65] proposed a block called Squeeze and Excitation( SE) blocks, which enhance the performance of existing deep architecture at minimal additional computational cost, adaptively recalibrating channel-wise

**TABLE 2.** An overview of properties and performance milestone of Generic object detection.

Detector	Proposal	Backbone	Input Image	Speed (FPS)	Publish in	Optimization	Loss Function	Softmax Layer	End-to-End Train	Language	Deep Learning Platform	Merits and Limitations
R-CNN[10]	Selective search	AlexNet	fixed	<0.1	CVPR-14	SGD-BP	classification(Hinge loss)+ Bounding box regression	yes	No	Matlab	Caffe	<b>Merits:</b> Improved performance from previously proposed methods; CNN models combine with RP based methods <b>Limitations:</b> sequentially-trained multistage pipeline ( fine-tuning of CNN, External RPN computation, each warped RP passing through CNN, SVM and BBR training); slow training; Training is expensive in term of space and time;
SPP-Net[77]	EdgeBoxe s	ZFNet	Arbitrar y	<1	ECCV14	SGD	classification(Hinge loss)+ Bounding box regression	yes	No	Matlab	Caffe	<b>Merits:</b> First proposed CNN architecture with SPP; use of Convolutional feature map; Faster than OverFeat; Accelerate RCNN without sacrificing performance evaluation by orders of magnitude; <b>Limitation:</b> Inherit disadvantages of RCNN; not a significant improvement in results; Fine-tuning unable to update the CONV layers before SPP layer
Fast - RCNN[18]	Selective Search	AlexNet, VGGM, VGG16		<1	ICCV15	SGD	ClassLog Loss+Bounding Box Regression	yes	No	Python	Caffe	<b>Merits:</b> Ignoring RP generation (end-to-end detector training); designed RoI pooling layer; improved performance as compared to SPPNet; for feature storage no disk storage required <b>Limitations:</b> computation of external RP become a bottleneck; Not feasible for real-time applications
Faster-RCNN[9]	Region proposal Network	ZFNet, VGG	Arbitrar y	<5	NIPS15	SGD	Class Log Loss+Bounding Box Regression	yes	Yes	Python/Matlab	Caffe	<b>Merits:</b> Instead of using SS. Proposed RPN used for generating High-quality; Introduce translation invariant and multi-scale anchor boxes; Fast aggregate RCNN and RPN into a single network by sharing CONV layers; it faster than Fast RCNN without compromising on performance; Can run testing at 5 FPS with VGG16 <b>Limitation:</b> still real-time fails short; complex Training; not a streamlined process
R- FCN[78]	Region proposal Network	ResNet101	Arbitrar y	<10	NIPS16	SGD	Class Log Loss+Bounding box Regression	no	Yes	Matlab	Caffe	<b>Merits:</b> Network is Fully convolutional detector; block of specialized CONV layers used for designing a set of position-sensitive score maps; without compromising on accuracy it is faster than Faster RCNN <b>Limitation:</b> Not suitable for real-time application, whereas Training is not a streamlined process.
Mask R-CNN[80]	Region Proposal Network	ResNet101, ResNeXt101	Arbitrar y	<5	ICCV17	SGD	ClassLog Loss +Bounding Box Regression + Semantic Sigmoid Loss	yes	Yes	Matlab/Python	Tensorflow/ Keras	<b>Merits:</b> an extended version of Faster R-CNN framework uses to generate instance segmentation with an additional mask detection branch in parallel with BB prediction branch; achieve outstanding performance with Feature Pyramid Network (FPN) <b>Limitation:</b> Falls short of real-time applications
FPN[79]	Region Proposal Network	--	Arbitrar y			Synchronized SGD	Class Log Loss+Bounding Box Regression	yes	Yes	Python	Tensorflow	<b>Merits:</b> FPN is substantially faster running at 6 to 7 FPS; a significant improvement over several stable baseline and competition winner shows in FPN <b>Limitation:</b> densely sampled image pyramids used for existing mask proposal methods; computationally expensive.
YOLO[19]	--	GoogLeNet like	Fixed	<25 VGG	CVPR16	SGD	Class Sum SquareerrorLoss+bounding box regression+object confidence+ background confidence	yes	Yes	C	Darknet	<b>Merits:</b> First unified detector framework (elegant and efficient); exclude RP method completely; Faster than previously proposed detector; YOLO and, Fast YOLO run at 45 and 155 FPS respectively; <b>Limitation:</b> have difficulty to localized tiny objects, dramatic accuracy falls as compared to the state of the art detectors;
SSD[84]	--	VGG16	Fixed	<60	ECCV16	SGD	ClassSoftmaxLoss+bounding Box Regression	no	Yes	C++/Python	Caffe	<b>Merits:</b> utilizes multi-scale Conv layers to perform detection hybrid approach of RPN and YOLO; First accurate and efficient unified detector; Faster and significantly more accurate than YOLO; frames per second is 59 FPS; <b>Limitations:</b> struggling in detecting small objects
YOLOv2[85]	--	Darknet	Fixed	<50	CVPR17	SGD	Class Sum SquareerrorLoss+bounding box regression+object confidence+ background confidence	yes	Yes	C	Darknet	<b>Merits:</b> Achieve high accuracy and high speed; Propose a faster DarkNet19; improved the speed and accuracy by using several existing strategies; YOLO9000 can detect over 9000 object categories in real-time <b>Limitations:</b> struggling in detecting small objects
DSSD[86]		ResNet101	Fixed	<66	---	SGD	Joint localization Loss+ confidence loss(Softmax)	No	Yes	Python	Caffe	<b>Merits:</b> DSSD shows great improvement in Small object detection <b>Limitation:</b> only encoder-decoder hourglass model used with SSD frame; other models could test
RetinaNet		FPN/ResNet	Fixed	AP <25		SGD	Focal Loss( cross-entropy loss)			Python	Pytorch/ Caffe	<b>Merits:</b> Focal loss use to address the issue of class imbalance.

This table summarizes the merits and limitation of state-of-the-art algorithms with references. Some other parameters used in the detector are also specified, such as backbone framework, input image size, loss function, and the working principle.

feature responses by explicitly modeling the interdependences between Convolutional feature channels, and therefore

win the ILSVRC2017 classification task. Research on CNN architectures is remained active, with emerging networks

**TABLE 3.** Backbone Framework of DCNN commonly used in Generic object detection.

S/no	Backbone Framework	No of Parameter $\times 10^6$	CONV+FC (no of layers)	Testing Error	First used in	Merits
01	AlexNet[59]	57	5+2	15.3%	Girshick et al. [10]	ImageNet classification problem used the first-time DCNN; new ways open with the used of CNN feature map; winning the ILSVRC2012 competition
02	ZFNet(fast)[134]	58	5+2	14.8%	He et al. [77]	Similar to Alexnet Except size of filters, stride for convolution, and specific layers: no of filters
03	OverFeat[64]	140	6+2	13.6%	Sermanet et al. [64]	Similar to Alexnet Except size of filters, stride for convolution, and specific layers: no of filters
04	VGGNet[63]	134	13+2	6.8%	Girshick et al.[9]	Depth of network increase step by step by stacking $3 \times 3$ Conv filters
05	GoogLeNet[62]	6	22	6.7%	Szegedy et al. [61]	Multi-branches of convolution layer with different numbers and sizes of filters and concatenate the produce features maps first time in Inception module; global averaging and inclusion of bottleneck.
06	Inception v2[61]	12	31	4.8%	Howard et al.[235]	Batch normalization give a boost to fast training
07	Inception v3[102]	22	47	3.6%	--	Reduction of spatial resolution and inclusion of separable convolution
08	YOLONet[85]	64	24+1	--	Redmon et al. [19]	Inspired from GoogLeNet
09	ResNet50[65]	23.4	49	3.6%	(ResNets) He et al. [65]	Deeper networks can learn with identity mapping
10	ResNet101[65]	42	100	--	He et al.[65]	GoogLeNet with Bottleneck and Global average pooling and requires fewer parameters as compared to VGG.
11	InceptionResNet V1[135]	21	87	3.1%(Ensemble)	--	Same computational cost as Inception v3 with a combination of Inception module and identity mapping(Training process is faster)
12	InceptionResNet v2[135]	30	95	--	Huang et al. [229]	A costlier residual version of Inception, significant improvement in performance
13	Inception v4[135]	41	75	--	--	A slow InceptionresNet v2: no residual connections, An Inception variant with roughly the having same performance of recognition
14	ResNeXt[156]	23	49	3.0%	Xie et al[156]	Repeated block with the same topology that aggregates asset of transformations
15	DenseNet201[136]	18	200	--	Zhou et al[42]	Every layer connected in a feed-forward manner, reduction in parameters, improve the vanishing gradient problem, encourage features to reuse
16	DarkNet[85]	20	19	--	Redmon and Farhadi et al. [85]	Lesser no of parameters inspired by VGGNet
17	MobileNet[235]	3.2	27+1	--	Howard et al. [235]	Depth-wise separable convolutions based on Lightweight deep CNN
18	SeResNet[269]	26	50	2.3%	Hu et al.[269]	Squeeze and Excitation block: compulsory for existing CNNs backbone, channel-wise attention

The state-of-the-art deep architecture of backbone frameworks use in object detection summarizes in tableIII. Architecture layers, merits, and properties such as no of parameters, testing time discuss here.

such as Hourglass [127], Dilated Residual Networks [137], Xception [138], DetNet [139], Dual Path Network (DPN) [140], fish-Net [141], CBNet [142], DetNAS [143] and GLoRe [144], etc.

#### D. DATASETS AND PERFORMANCE EVALUATION

##### 1) DATASET

With recent advances in deep learning computer vision, object detection applications can evolve rapidly. In addition to significant improvements in performance, the current approach has primarily controlled the need for large-scale image datasets. Modern evolving techniques use end-to-end pipelines to improve the performance of real-time transactions. Besides that, data is of significant importance, whether used to compare and measure the performance of competitive algorithms or to solve the challenging or complex existing problems. A large amount of big annotated data is the main reason behind the tremendous success of the use of deep learning techniques in object detection. The Internet plays a vital role in building a comprehensive dataset to provide access to a wide range of images covering the vastness and diversity of objects. Five datasets are very popular in the field of generic object detection, namely as PASCAL VOC 2007 [145], PASCAL VOC2012 [103], ImageNet [56], Microsoft

COCO [104] and OpenImages [146]. Some selected images of the benchmark dataset shown in FIGURE 15 and Table 4 summarize the specification and attributes of these datasets. Creating massively interpreted datasets requires crowd funding strategies. First, define the target object set categories, secondly collect a collection of images from a diversity of dimensions to represent the specified category selected on the Internet, and finally annotate the collected images.

Each dataset has its particular object detection challenges, including interpretation of commonly available datasets, an annual competition, standardized evaluation software, and similar workshops. Details of the statistics, such as the total number of images, training samples, validation, and test sets of these datasets discuss in Table 5.

##### a: PASCAL VOC 2007/2012

Everingham et al. [147], [148] proposed a series of benchmark datasets for object detection and classification over several years and illustrate the paradigm of standardized evaluation of recognition algorithms in the form of annual competitions. Initially, the dataset consisted of five categories in 2005 and expanded to twenty categories of everyday life in 2009. Since 2009, the number of images in the dataset has

**TABLE 4.** Benchmark Generic Object detection Databases.

S/no	Datasets	Reference	No of images	Images Size	Categories	per category images	Objects per images	Initial Date	Description
1	PASCAL VOC (2007/ 2012)	[103, 145]	11,540	470x380	20	303-4087	2.4	2005	The dataset consists of 20 categories from daily life; Contain large no of real-world training images; multiple objects in one image; images of scenes; complex sample; Intra-class variations
2	ImageNet (2015)	[56]	14,197,122	500x400	21,841	—	1.5	2010	dataset based on WordNet hierarchy; per image more instance and objects; Benchmark for ILSVRC challenged ; Object- centric images
3	MS COCO (2014)	[104]	328,000+	640 x 480	91	—	7.3	2014	The dataset contains real-world scenes; multiple objects in each image and fully annotated; images with objects segmentation
4	Place(2017)	[270]	10 million+	256 x 256	434	—	—	2014	The Largest scenes recognition annotated dataset; Places 365 (Standard); Places 365(Challenge); Places205 and Place 88 (Benchmarks) ; four subsets
5	Open Images( 2019)	[150]	9 million+	Varying	6000+	—	8.3	2017	A large scale object detection dataset: Open Images v5; visual relationship detection and instance segmentation

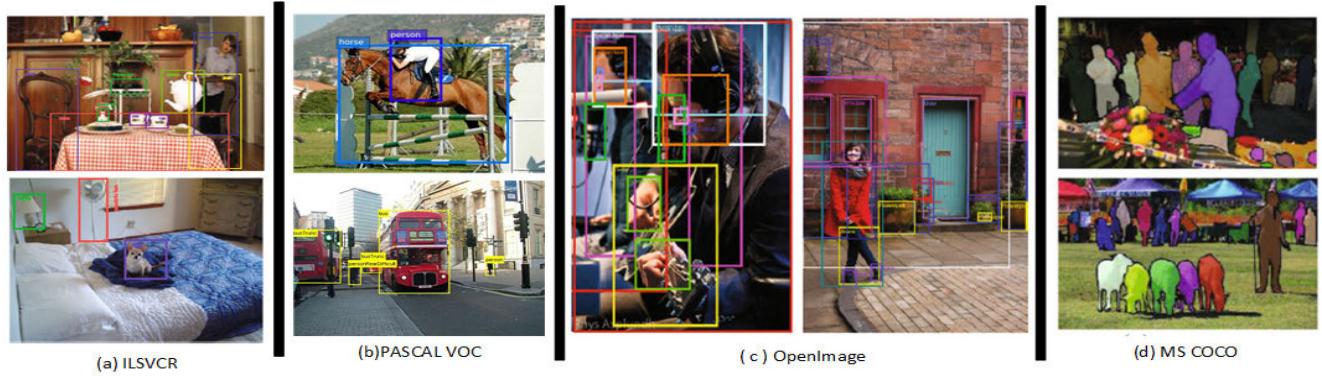
Summary of primary benchmark dataset for generic object detection (computer vision task) given in Table III

**TABLE 5.** Object detection datasets statistics.

Object challenge	categories	Total images			Annotated objects			Training + Val set	
		Train set	Val set	Test set	Train- set	Val-set	Instance	Boxes	Per image boxes
<b>Object detection challenge PASCAL VOC:</b>									
VOC07		2501	2501	4952	6301	6307	5011	12608	2.5
VOC08		2111	2221	4133	5082	5281	4332	10364	2.4
VOC09	20	3473	3581	6650	8505	8713	7054	17218	2.3
VOC10		4998	5105	9637	11577	11797	10103	23374	2.4
VOC11		5717	5823	10,994	13609	13841	11540	27450	2.4
VOC12		5717	5823	10,991	13609	13841	11540	27450	2.4
<b>Object detection challenge ILSVRC:</b>									
ILSVRC13		395909		40152	345854		416030	401356	1
ILSVRC14		456567		40152	478807		476668	534309	1.1
ILSVRC15	200	456567	20121	51294	478807	55502	476668	534309	1.1
ILSVRC16		456567		60000	478807		476668	534309	1.1
ILSVRC17		456567		65500	478807		476668	534309	1.1
<b>Object detection challenge MSCOCO:</b>									
MSCOCO15		82783	40504	81343	604907	291875			
MSCOCO16	80	82783	40504	81434	604907	291875	123287	896782	7.3
MSCOCO17		118287	5000	40670	860001	36781			
MSCOCO18		118287	5000	40670	860001	36781			
<b>Open images challenge object detection(OICOD):</b>									
OICOD18		500	1643042	100000	99999	11498734	696410	1743042	12195144
									7

increased every year. previous images have also been retained for test results, which are compared by year.

The popularity of Pascal VOC is slowly waning due to the availability of other improved datasets in the market, such



**FIGURE 15.** Some instances from the PASCAL VOC, ILSVRC, MS COCO, and Open Images. PASCAL VOC has an XML file, unlike the MSCOCO that has a JSON file. PASCAL VOC creates a file for each image separately. In contrast, MS COCO creates one file for the entire dataset for training, testing, and validation. The BB data formats are different in COCO and PASCAL VOC. However, Open Images has 9M images (Largest dataset). It has been annotated at the image-level.

as ImageNet, MSCOO, and OpenImage. Average Precision (AP) measures the performance of object detection for each category, and Mean Average Precision (mAP) is in all twenty classes.

#### b: ILSVRC(IMAGENET LARGE SCALE VISUAL RECOGNITION CHALLENGE)

Russakovsky et al. proposed a dataset driven from ImageNet [56], increasing the number of classes and images and scaling up the training and evaluation standards of object detection tasks based on PASCAL VOC. The number of images in the ImageNet dataset increased to over 1.2 million with more than 1000 different object categories, namely ImageNet1000. It provides a standardized benchmark for the ILSVRC image classification challenge.

#### c: MICROSOFT COCO

Lin et al. [104] proposed a database, namely MSCOCO database, based on familiar objects in natural everyday complex scenes to provide richer image understanding. The Objects are labeled with fully segmented instances to test the accurate detector evaluation. The Microsoft COCO dataset has a total of three hundred thousand thoroughly segmented images, with an average of seven object instances per image in a total of 80 categories. Some key points made MSCOCO more challenging than PASCAL 2012, such as the existence of fewer iconic objects and amid clutter or heavy occlusion with a wide range of scales, with a high percentage of small objects [149] and the evaluation metric requirement for accurate objects-localization. The performance of the object detection task evaluates the use of *AP* under different degrees of *IoU* and different sizes of an object. The MS COCO object detection challenge is based on two main object detection tasks (for example, using either instance segmentation or bounding box output). Currently, MS COCO has become the standard for object detection, as ImageNet was in its time.

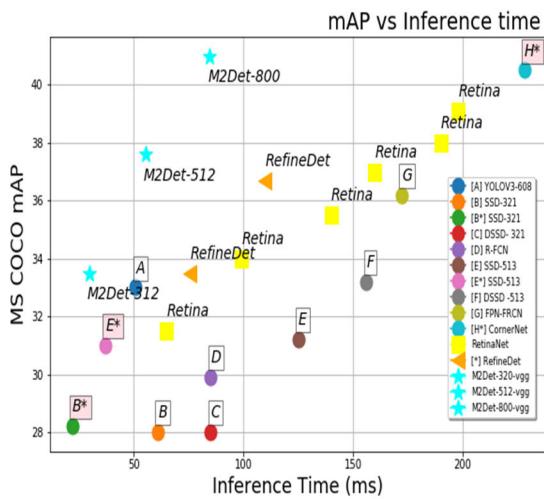
#### d: THE OPEN IMAGE CHALLENGE OBJECT DETECTION(OICOD)

Kuznetsova et al. [150] propose the largest publicly available dataset driven from OpenImageV4 (Currently, it was version5 2019). OICOD provides a significant increase in the number of classes, images, bounding boxes, and instance segmentation masks, and also proposed a substantial annotation process, which makes it different from other previous object detection datasets such as ILSVRC and MS COCO. OpenImage V4 uses classifiers to annotate images and only uses labels that have significantly high scores for human verification, while ILSVRC and MS COCO have an exhaustively annotated dataset. Human confirmed positive-labels for object instances interpret in OICOD.

## 2) EXPERIMENT EVALUATION

In this paper, the performance of various object detection methods using three benchmark datasets is compared such as PASCAL VOC 2007/2012 [103], [145], Microsoft COCO [104] and Open image, while evaluated algorithm are SPP-net [77], Fast R-CNN [18], NOC [16], Bayes [96], Mr-CNN& S-CNN [116], Faster R-CNN [9], HyperNet [112], ION [111], MS-GR [115], StuffNet [110], SSD300 [84], SSD512 [84], OHEM [151], SDP+CRC [25], GCNN [83], subCNN [152], GBD-Net [118], PVANET [153], YOLO [19], YOLOv2 [85], R-FCN [78], FPN [79], Mask R-CNN [80], DSSD [86], R-CNN [10] and DSOD [87]. The performance of object detection algorithms is evaluated with three parameters, such as recall, precision, and Frame per Second (FPS).

The **Average Precision (AP)** is the performance evaluation terms computed for each category, derived from precision and recall. However, the **mean Average Precision (mAP)** is an average measure of performance that is calculated for all object categories. Details of performance metric can be found in [148], [154], [155]. Prediction detection  $\{(b_j, c_j, p_j)\}_j$  of the test image,  $I$  is the standard outputs of a detector while  $j$  is indexed of  $b_j$ -object as BB predicted category represents  $c_j$ , while the confidence score is represented by  $p_j$ .



**FIGURE 16.** Speed (ms) vs. accuracy (mAP) on COCO test-dev.

A predicted detection is considered as True Positive (TP) if the ground truth label  $c_g$  is equal to  $c$ , and overlap ratio that is Intersection over Union( $IoU$ ) between ground truth Bounding Box  $b^g$  and predicted BB  $b$  is not smaller than a predefined threshold  $\varepsilon$ , which explains the following:

$$IoU(b, b^g) = \frac{\text{area}(b \cap b^g)}{\text{area}(b \cup b^g)} \quad (7)$$

where  $\cup$  and  $\cap$  represent the intersection and union, respectively. Predicted detection shows False Positive (FP) for other values of  $\varepsilon$  except 0.5. For the acceptance of the predicted class label,  $c$  compares the confidence level  $p$  to some threshold  $\beta$ . The Comparative results from various detection algorithms using PASCAL VOC show that the robust backbone network can produce better prediction results (comparison among R-CNN with VGG16 or with AlexNet, and SPP-net with ZF-Net [134]). Object detection performance improves with the invention of end-to-end multi-task architecture (FRCN) [78], SPP layer (SPP-Net) [79], and RPN (Faster R-CNN). The importance of data augmentation is increasing with the demand for robust multi-level features in deep learning-based models.

Some other factors have a substantial impact on the performance of the object detector such as hard-negative samples mining (e.g., OHEM), multi-scale representation (e.g., ION), contextual information (e.g., StuffNet, HyperNet), modified classification network (e.g., NOC), multi-region and multi-scale feature extraction (e.g., MR-CNN). YOLO produces an abysmal result for object localizations of high IoU on PASCAL VOC2012. Some strategies, such as batch normalization, anchor box, and fine-grained features, are used for correct R-CNN (YOLO+FRCN) localization errors (YOLOv2). Since the introduction of MSCOCO, special consideration has been given to the bounding box location accuracy rather than using the IOU threshold.

This dataset is more challenging than PASCAL VOC 2012 due to the existence of less iconic, diversified scales

objects and stricter requirements on object localization. In MS-COCO, Average precision with different degrees of IOUs for the evaluation of object detection performance for this dataset. The object detection performance and localization can improve by using multi-scale training and test with the support of complementary information from other related tasks and additional information in different resolution (R-FCN). Some algorithms, such as DSSD and FPN, can create improved feature pyramids to achieve multi-scale representations. Object detector based on regression/classification (such as SSD and YOLO) is not performing well due to significant localization errors than region proposal based methods, i.e., Faster R-CNN and R-FCN. Contextual information is very beneficial for identifying small objects as it provides contextual information for consulting nearby surrounding objects (multi-path and GBN-Net).

MS COCO contains many non-standard objects that reduce the performance of the object detector. However, the performance can improve with the invention of the robust backbone models (e.g., ResNeXt [156]) and other useful strategies like multi-task learning [80], [130]. Some performance evaluation metric for the PASCAL VOC, ILSVRC, and MS COCO object detector summarizes in Table 6 with some metric modification for the OpenImages Challenges proposed in Kuznetsova *et al.* [150]. Table 9 shows the time analysis of various object detection algorithms on the NVIDIA Titan x except for the selection-search, which processed on CPU Intel i7-6700k.

### 3) CHALLENGES OF GENERIC OBJECT DETECTION

High accuracy and high efficiency are the two main competing objectives for the ideal generic object detection task. In the *high-efficiency detection task*, memory, and storage requirements to run the entire detection task must be acceptable in real-time. However, high-quality detection requires accurate recognition and localization of objects in images or video frames.

**A wide range of object categories and intra-class variations** are the two main challenges in detection accuracy. Intrinsic factors and imaging conditions are the two types of intra-class. The intrinsic-factor is a possible variation in object instances of a particular category in terms of one or more materials, texture, color, shape, size, and object, which appears in different poses and non-rigid deformations. Variations in the imaging condition are due to unconstrained environmental impacts such as weather conditions, lighting, camera models, physical locations, illuminations, backgrounds, occlusion, and viewing distance. Significant variations in object appearance are caused by intra-class such as scale, cluster, pose, illumination, blur, occlusion, cluster, shading, and motion. Poor resolution, noise corruption, digitization patterns, and filtering distortions can increase the challenges of object detection, as shown in FIGURE 17. In practice, the current object detector focuses primarily on structured object categories, such as twenty categories in

**TABLE 6.** Performance evaluation metric (generic object detection).

S/no	Matric term	Abbreviations	Description
01	TP	True Positive	True-positive detection / image
02	FP	False Positive	Per image false-positive detection
03	$\beta$	Confidence threshold	Confidence threshold uses for Precision $P(\beta)$ and Recall $R(\beta)$
04	$\epsilon$	IOU threshold	<b>PASSCAL VOC</b> ~ 0.5 <b>ILSVRC</b> $\min(0.5, \frac{wh}{(w+10)(h+10)})$ ; $w, x, h$ is the size of a GT box <b>MS COCO</b> Ten IOU threshold $\epsilon \in (0.5: 0.05: 0.95)$
05	$P(\beta)$	Precision	Precision is a correct detection divided by total object-detection returned by the detector with the confidence of at least $\beta$
06	$R(\beta)$	Recall	The recall is a fraction of all $N_c$ objects detected by the detection having the confidence of at least $\beta$
07	AP	Average Precision	AP is calculated over the different levels of $R(\beta)$ and a variation in confidence level $\beta$ against a single category.
08	mAP	Mean Average Precision	<b>PASSCAL VOC</b> Average AP at a single IOU of all categories <b>ILSVRC</b> Average AP at a modified IOU of all categories <b>MS COCO</b> $Ap_{coco}$ : mAP average over Ten IOUs: (0.5: 0.05: 0.95) $Ap_{coco}^{IOU=0.75}$ : mAP at IOU = 0.75 (strict metric) $Ap_{coco}^{medium}$ : mAP for objects of the area between 322 and 962 $Ap_{coco}^{large}$ : mAP for large objects of the area larger than 962
09	AR	Average Recall	The maximum recall gave a fixed number of detection per image, mean overall classes and IOU threshold
10	AR	Average Recall	<b>MS COCO</b> $AR_{coco}^{max=1}$ : AR gives (Maximum detections /image =1) $AR_{coco}^{max=10}$ : AR gives (Maximum detections /image =10) $AR_{coco}^{max=100}$ : AR gives (Maximum detections /image =100)  $AR_{coco}^{small}$ : AR for (Object area < 322 ) small objects $AR_{coco}^{medium}$ : AR for (322< Object area < 962) $AR_{coco}^{large}$ : AR for ( Object Area > 962) large objects

This table summarizes the performance evaluation parameters of benchmark datasets such as PASCAL VOC (07+12), MS COCO, and ILSVRC.

**TABLE 7.** PASCAL VOC 2007: Test set results comparison (%).

Detector	Trained on	aero	Bike	bird	Ship	container	bus	TV	cat	dog	cow	table	m-bike	horse	car	person	plant	sheep	sofa	train	chair	mAP	
R-CNN(AlexNet)	VOC 07	68.1	72.8	56.8	43.0	36.8	66.3	68.6	67.6	61.2	63.5	54.5	68.6	69.1	74.2	58.7	33.4	62.9	51.1	62.5	34.4	58.5	
R-CNN(VGG16)	VOC 07	73.4	77.0	63.4	45.4	44.6	75.1	71.1	79.8	79.4	73.7	62.2	73.1	78.1	78.1	64.2	35.6	66.8	67.2	40.5	66.0		
SPP-Net(ZF)	VOC 07	68.5	71.7	58.7	41.9	42.5	67.7	68.8	66.0	67.0	63.4	71.3	72.5	72.1	58.9	32.8	60.9	56.4	67.9	34.7	60.9		
G-CNN	VOC 07	68.3	77.3	68.5	52.4	38.6	78.5	66.4	81.0	77.2	73.6	64.5	75.8	80.5	66.6	34.3	65.2	64.4	75.6	47.1	66.8		
Bayes	VOC 07	74.1	83.2	67.0	50.8	51.6	76.2	73.7	77.2	77.3	78.9	65.6	75.1	78.4	81.4	70.1	41.4	69.6	60.8	70.2	48.1	68.5	
Fast-R-CNN	VOC 07	77.0	78.1	69.3	59.4	38.3	81.6	70.4	86.2	84.7	78.8	68.9	76.6	82.0	78.6	69.9	31.8	70.1	74.8	80.4	42.8	70.0	
	07+12																						
SDP+CRC	VOC 07	76.1	79.4	68.2	52.6	46.0	78.4	70.3	81.0	78.6	73.5	65.3	76.7	81.0	78.4	77.3	39.0	65.1	67.2	77.5	46.7	68.9	
SubCNN	VOC 07	70.2	80.5	69.5	60.3	47.9	70.0	60.5	81.2	82.7	73.9	63.0	76.0	80.6	78.7	70.2	38.2	69.4	67.7	77.7	48.5	68.5	
StuffNet	VOC 07	72.6	81.7	70.6	60.5	53.0	81.5	73.8	83.9	85.0	78.9	70.7	77.0	85.7	83.7	78.7	42.2	73.6	69.2	79.2	52.2	72.7	
NOC	VOC 07	76.3	81.4	74.4	61.7	60.8	84.7	75.7	82.9	83.2	79.2	69.2	78.5	83.2	78.2	68.0	45.0	71.6	76.7	82.2	53.0	73.3	
	+12																						
MR-CNN&S-CNN	VOC 07	80.3	84.1	78.5	70.8	68.5	88.0	81.0	87.8	87.2	85.2	73.7	85.0	86.5	85.9	76.4	48.5	76.3	75.5	85.0	60.3	78.2	
	07+12																						
HyerNet	VOC 07	77.4	83.3	75.0	69.1	62.4	83.1	76.5	87.4	85.1	79.8	71.4	80.0	85.1	87.4	79.1	51.2	79.1	75.7	80.9	57.1	76.3	
MS-GR	VOC 07	80.0	81.0	77.4	72.1	64.3	88.2	75.5	88.4	87.3	85.4	73.1	85.1	87.4	88.1	79.6	50.1	78.4	79.5	86.9	64.4	78.6	
OHEM+Fast-R-CNN	VOC 07	80.6	85.7	79.8	69.9	60.8	88.3	80.7	89.6	87.1	85.1	76.5	82.4	87.3	87.9	78.8	53.7	80.5	78.7	84.5	59.7	78.9	
	07+12																						
ION	07+12+S	80.2	85.2	78.8	70.9	62.6	86.6	83.5	89.8	88.4	86.9	76.5	83.4	87.5	86.9	80.5	52.4	78.1	77.2	86.9	61.7	79.2	
Faster R-CNN	VOC 07	70.0	80.6	70.1	57.3	49.9	78.2	67.6	82.0	80.3	75.3	67.2	75.0	79.8	80.4	76.3	39.1	68.3	67.3	81.1	52.2	69.9	
Faster R-CNN	07+12	76.5	79.0	70.9	65.5	52.1	83.1	72.6	86.1	84.8	81.9	65.7	77.5	84.6	84.7	76.7	28.8	73.6	73.9	83.0	52.0	73.2	
Faster R-CNN	07+12+CO	84.3	82.0	77.7	68.9	65.7	88.1	78.9	88.9	85.9	86.3	70.8	80.1	87.6	88.4	82.3	53.6	80.4	75.8	86.6	63.6	78.8	
CO	SSD300	07+12+CO	80.9	86.3	79.0	76.2	57.6	87.3	78.9	88.6	87.5	85.4	76.7	84.5	89.2	88.2	81.4	55.0	81.9	81.5	85.9	60.5	79.6
CO	SSD512	07+12+CO	86.6	88.3	82.4	76.0	66.3	88.6	81.2	89.1	86.5	88.4	73.6	85.3	88.9	88.9	84.6	59.1	85.0	80.4	87.4	65.1	81.6

The table summarizes the detection results of the fundamentals object detector on PASCAL VOC2007. The S in ION '07+12+S' denotes SBD segmentation labels.; VOC 07 is used for PASCAL VOC 2007, 'VOC 07+12': the combination of PASCAL VOC2007 and VOC2012, '07+12+COCO': these algorithms are trained on MS-COCO trainval35k at first and then fine-tuned onPASCAL VOC 07+12

**TABLE 8.** PASCAL VOC 2012: Test set results comparison (%).

Detector	Trained on	aero	Bike	bird	Boat	Bottle	bus	TV	cat	dog	cow	table	m-bike	horse	car	person	plant	sheep	sofa	train	Chair	mAP
R-CNN[AlexNet][10]	VOC 12	71.8	65.8	52.0	34.1	32.6	59.6	54.1	69.8	68.6	52.0	41.7	68.3	61.3	60.0	57.8	29.6	57.8	40.9	59.3	27.6	53.3
R-CNN(VGG16)[10]	VOC 12	79.6	72.7	61.9	41.2	41.9	65.9	60.3	84.6	82.0	67.2	46.7	76.0	74.8	66.4	65.2	35.6	65.4	54.2	67.4	38.5	62.4
Bayes[66]	VOC 12	82.9	76.4	64.1	44.6	49.4	70.3	66.2	84.6	82.7	68.6	55.8	79.9	77.1	71.2	68.7	41.4	69.0	60.0	72.0	42.7	66.4
Fast R-CNN[18]	VOC	82.3	78.4	70.8	52.3	38.7	77.8	64.2	89.3	87.5	73.0	55.0	80.8	83.4	71.6	72.0	35.1	68.3	65.7	80.4	44.2	68.4
	07+12																					
StuffNet[110]	VOC 012	83.0	76.9	71.2	51.6	50.1	76.4	65.4	87.8	85.0	74.8	55.7	80.5	81.2	75.7	79.5	44.2	71.8	61.0	78.5	48.3	70.0
NOC[16]	VOC 07	82.8	79.0	71.6	52.3	53.7	74.1	68.1	84.9	85.0	74.3	53.1	79.5	81.3	69.0	72.2	38.9	72.4	59.5	76.7	46.9	68.8
	+12																					
MR-CNN&S-CNN[271]	VOC	85.5	82.9	76.6	57.8	62.7	79.4	74.0	86.6	87.0	79.3	62.2	74.7	83.4	77.2	78.9	45.3	73.4	65.8	80.3	55.0	73.9
	07+12																					
HyerNet[112]	VOC	84.2	78.5	73.6	55.6	58.3	78.7	65.7	87.7	86.0	74.6	52.1	83.3	81.7	79.8	81.8	48.6	73.5	59.4	79.9	49.6	71.4
	07+12																					
OHEM+Fast R-CNN[151]	07+12+COCO	90.1	87.1	79.9	65.8	53.7	86.2	77.3	92.9	90.6	83.4	69.5	88.9	88.9	85.0	83.6	59.0	82.0	74.7	88.2	62.4	80.1
	o																					
ION[111]	07+12+S	87.5	84.7	76.8	63.8	66.3	82.6	73.5	90.9	88.9	82.0	64.7	84.7	86.5	79.0	82.3	51.4	78.2	69.2	85.2	57.8	76.4
Faster R-CNN[9]	07+12	84.9	79.8	74.3	53.9	58.3	77.5	61.5	88.5	86.9	77.1	55.3	80.9	81.7	75.2	79.6	40.1	72.6	60.9	81.2	45.6	70.4
Faster R-CNN[9]	07+12+CO	87.4	83.6	76.8	62.9	59.6	81.9	70.2	91.3	89.0	82.6	59.0	84.7	85.5	82.0	84.1	52.2	78.9	65.5	85.4	54.9	75.9
	CO																					
YOLO[19]	07++12	77.0	67.2	57.7	38.3	22.7	68.3	50.8	81.4	77.2	60.8	48.5	71.3	72.3	55.9	63.5	28.9	52.2	54.8	73.9	36.2	57.9
YOLO+ Fast R-CNN[19]	07++12	83.4	78.5	73.5	55.8	43.4	79.1	67.2	89.4	87.5	75.5	67.0	81.0	80.9	73.1	74.7	41.8	71.5	68.5	82.1	49.4	70.7
YOLO v2[85]	07+12+CO	88.8	87.0	77.8	64.9	51.8	85.2	76.8	93.1	91.3	81.4	70.2	87.2	88.1	79.3	81.0	57.7	78.1	71.0	88.5	64.4	78.2
	CO																					
SSD300[84]	07+12+CO	91.0	86.9	78.1	65.0	55.4	84.9	76.5	93.4	91.3	83.6	67.3	88.5	88.9	84.0	85.6	54.7	83.8	77.3	88.3	62.1	79.3
SSD512[84]	07+12+CO	81.4	88.6	82.6	71.4	63.1	87.4	80.4	93.9	92.0	86.6	66.3	90.8	91.7	88.1	88.5	60.9	87.0	75.4	90.2	66.9	82.2
R-FCN (ResNet101)[78]	07+12+coc	92.3	89.9	86.7	74.7	75.2	86.7	83.4	95.8	95.0	90.4	66.5	92.1	93.2	89.0	91.1	71.0	89.7	76.0	92.0	70.2	85.0
	o																					

\*table summarizes the detection results comparison of some fundamental models of object detection on PASCAL VOC 2012. PASCAL VOC has twenty categories, as indicates by the name of table columns. 07++12': the combination of PASCAL VOC2007 train+ validation and test and VOC2012 train + validation. '07++12+COCO': trained on MS- COCO trainval35k at first, then fine-tuned on 07++12.

**TABLE 9.** Testing Consumption Comparison(PASCAL VOC 2007 Test set).

Methods	Trained on	Rate/FPS	mAP%	Test Time (Sec/ img)
SS+ R-CNN[10]	07	0.03	66.0	32.84
SS+ SPP-net[77]	07	0.44	63.1	2.3
SS + FRCN[18]	07+12	0.6	66.9	1.72
SDP+CRC[25]	07	2.1	68.9	0.47
SS+ HyperNet*[112]	07+12	5	76.3	0.20
MR-CNN &S-CNN[271]	07+12	0.03	78.2	30
ION[111]	07+12+S	0.5	79.2	1.92
Faster R-CNN(VGG16)[9]	07+12	9.1	73.2	0.11
Faster R-CNN (ResNet101)[9]	07+12	0.4	<b>83.8</b>	2.24
YOLO[19]	07+12	45	63.4	0.02
SSD300[84]	07+12	46	74.3	0.02
SSD512[84]	07+12	19	76.8	0.05
R-FCN(ResNet101)[78]	07+12+COCO	59	83.6	0.17
YOLO v2[85]	07+12	40	78.6	0.03
DSSD321(ResNet101)[86]	07+12	13.6	78.6	0.07
DSOD300[87]	07+12+COCO	17.4	81.7	0.06
PVANET+ [272]	07+12+COCO	21.7	<b>83.8</b>	0.05
PVANET+(compress)[272]	07+12+COCO	31.3	82.9	0.03
YOLO v3- 320	COCO	22	57.3	0.05
YOLO v3- 416	COCO	29	57.9	0.05
YOLO v3- 608	COCO	51	57.9	0.05

The table summarizes the time analysis of state-of-the-art models with mean average precision across all categories. Selective Search SS[10], 'Fast mode Selective Search SS\* [18], the Speed up version of HyperNet: HyperNet\* and PVANET+ (compress): PVANET with additional bounding box voting and compressed fully Convolutional layers. Faster R-CNN and PVANET have the highest mAP% as compare to other state-of-the-art models.

PASCAL VOC [148], ILSVRC [154] with two hundred types and ninety-one classes in MS COCO [104].

The demand for visual data analysis increases with the prevalence of mobile/wearable devices and social media networks. Due to **limited storage capacity and computational power**, the efficient object detection task becomes critical with mobile/wearable devices. The efficiency challenges increase with the possibility of a wide range of objects categories, location, and scales diversion within a single image. An object detector should be able to handle high data rates, past invisible objects, and unknown situations.

Manual annotation becomes impossible with the increase in images and categories, which can lead to weakly supervised strategies.

## VI. GENERIC OBJECT DETECTION IN DIFFERENT FIELDS

Human has been taking the assistance of AI (computer vision in particular) to perform many of his daily tasks in different areas, such as security military, transportation, medical, and daily life fields. Detail descriptions of the methods and techniques used in these fields listed below.

**TABLE 10.** MS-COCO Test set: Testing consumption comparison.

State-of-art algorithm	Data	Backbone	AP	AP <sub>50</sub>	AP <sub>75</sub>	AP <sub>S</sub>	AP <sub>M</sub>	AP <sub>L</sub>
Fast R-CNN[18]	Train	VGG-16	19.7	35.9	--	--	--	--
Faster R-CNN[9]	Train-val	VGG-16	21.9	42.7	--	--	--	--
OHEM[151]	Train-val	VGG-16	22.6	42.5	22.2	5.0	23.7	37.9
ION[63]	Train	VGG-16	23.6	43.2	23.6	6.4	24.1	38.3
OHEM++[151]	Train-val	VGG-16	25.5	45.9	26.1	7.4	27.7	40.3
R-FCN[78]	Train-val	ResNet-101	29.9	51.9	--	10.8	32.8	45.0
CoupleNet[273]	Train-val	ResNet-101	34.4	54.8	37.2	13.4	38.1	52.0
Faster R-CNN G-RMI[229]	--	Inception ResNetv2	34.7	55.5	36.7	13.5	38.1	52.0
Faster R-CNN +++[65]	Train-val	ResNet-101-C4	34.9	55.7	37.4	15.6	38.7	50.9
Faster R-CNN (FPN)[79]	Train-val35K	ResNet-101-FPN	36.2	59.1	39.0	18.2	39.0	48.2
Faster R-CNN (TDM)[274]	Train-val	Inception-ResNetv2-TDM	36.8	57.7	39.2	16.2	39.8	52.1
Deformable R-FCN[130]	Train-val	Aligned-Inception-ResNet	37.5	58.0	40.8	19.4	40.1	52.5
umd-det[275]	Train-val	ResNet101	40.8	62.4	44.9	23.0	43.4	53.2
Cascade R-CNN[230]	Train-val35K	ResNet101-FPN	42.8	62.1	46.3	23.7	45.5	55.2
SNIP[149]	Trainval35K	DPN-98	45.7	<b>67.3</b>	51.1	29.3	<b>48.8</b>	57.1
Fitness-NMS[276]	Trainval35K	ResNet-101	41.8	60.9	44.9	21.5	45	57.5
Mask R-CNN[80]	Trainval35K	ResNeXt-101	39.8	62.3	43.4	22.1	43.2	51.2
DCNv2+faster R-cCNN[131]	train118k*	ResNet-101	44.8	66.3	48.8	24.4	48.1	<b>59.6</b>
G-RMI[229]	Trainval32k	Ensemble of five models	41.6	61.9	45.4	23.9	43.5	54.9
YOLOv2[85]	Trainval35K	DarkNet-53	33	57.9	34.4	18.3	35.4	41.9
YOLOv3[121]	Trainval35K	DarkNet-19	21.6	44	19.2	5	22.4	35.5
SSD300*	Trainval35K	VGG-16	25.1	43.1	25.8	6.6	22.4	35.5
RON384+++[277]	Trainval	VGG-16	2.4	49.5	27.1	--	--	--
SSD321[84]	Trainval35K	ResNet-101	28.0	45.4	29.1	6.2	28.3	49.3
DSSD321[86]	Trainval35k	ResNet-101	28.0	46.1	29.2	7.4	28.1	47.6
SSD512*[84]	Trainval35K	VGG-16	28.8	48.5	30.3	10.9	31.8	43.6
SSD513	Trainval35K	ResNet-101	31.2	50.4	33.3	10.2	34.5	49.8
DSSD513[86]	Trainval35K	ResNet-101	31.2	50.4	33.3	10.2	34.5	49.8
RetinaNet500[122]	Trainval35K	ResNet-101	33.2	53.3	35.2	13.0	35.4	51.1
RetinaNet800[122]	Trainval35K	ResNet-101	34.4	53.1	36.8	14.7	38.5	49.1
M2Det512[125]	Trainval35K	ResNet-101-FPN	39.1	59.1	42.3	21.8	42.7	50.2
M2Det512[125]	Trainval35K	VGG-16	37.6	56.6	40.5	18.4	43.4	51.2
M2Det512[125]	Trainval35K	ResNet-101	38.8	59.4	41.7	20.5	43.9	53.4
M2Det800[125]	Trainval35K	VGG-16	41.0	59.7	45.0	22.1	46.5	53.8
RefineDet320[126]	Trainval35K	VGG-16	29.4	49.2	31.3	10.0	32.0	44.4
RefineDet512[126]	Trainval35K	VGG-16	33.0	54.5	35.5	16.3	36.3	44.3
RefineDet320[126]	Trainval35K	ResNet-101	32.0	51.4	34.2	10.5	34.7	50.4
RefineDet512[126]	Trainval35K	ResNet-101	36.4	57.5	39.5	16.6	39.9	51.4
RefineDet320+[126]	Trainval35K	VGG-16	35.2	56.1	37.7	19.5	37.2	47.0
RefineDet512+[126]	Trainval35K	VGG-16	37.6	58.7	40.8	22.7	40.3	48.3
RefineDet320+[126]	Trainval35K	ResNet-101	38.6	59.9	41.7	21.1	41.7	52.3
REFineDet512+	Trainval35K	ResNet-101	41.8	<b>62.9</b>	<b>45.7</b>	<b>25.6</b>	45.1	54.1
CornerNet512[127]	Trainval35K	Hourglass	40.5	57.8	45.3	20.8	44.8	56.7
NAS-FPN[132]	Trainval35K	RetinaNet	45.4	--	--	--	--	--
NAS-FPN[132]	Trainval35K	AmoebaNet	<b>48.0</b>	--	--	--	--	--

This table summarized the detection results on the MS COC test-dev of a state-of-the-art model with different combinations of baselines. AP, AP<sub>50</sub>, AP<sub>75</sub> represent score % based on IoU, AP<sub>S</sub>: Average precision of small objects, AP<sub>M</sub> : Average precision of medium object, AP<sub>L</sub>: Average precision of large objects. RetinaNet has the highest precision in small object detection, while NAS-FPN has the highest average precision. DCNv2+Faster R-CNN models trained on 118K images of the COC 2017.

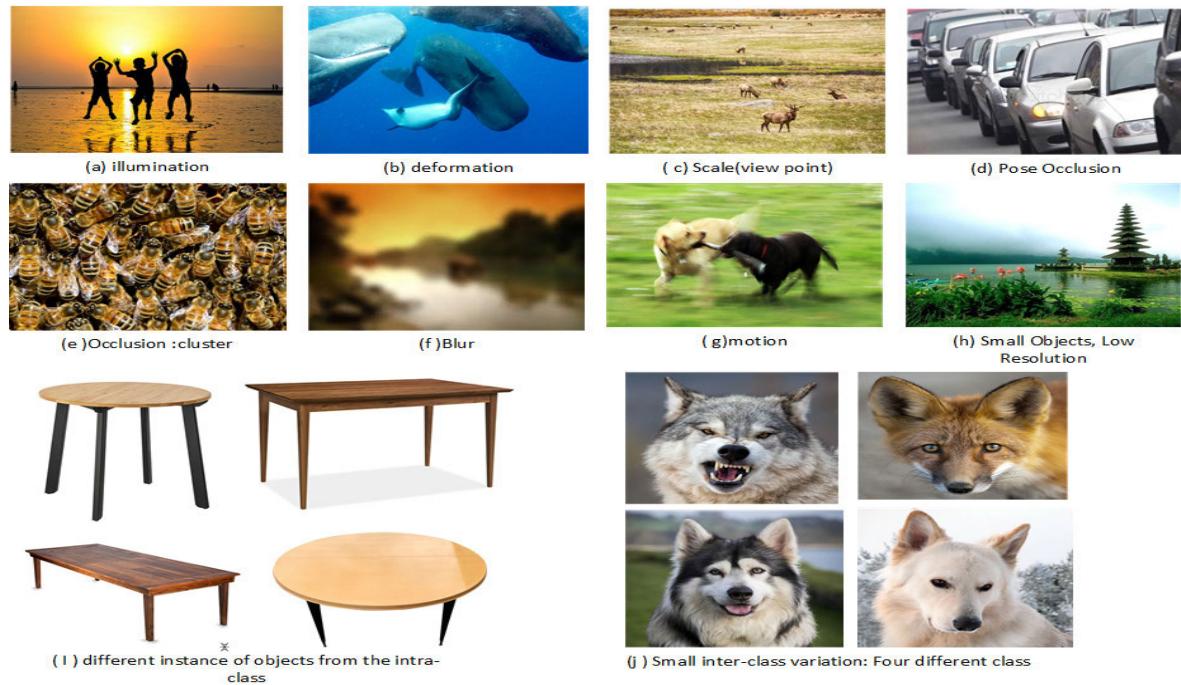
#### A. OBJECT DETECTION IN SURVEILLANCE

The pedestrian detection, face detection, fraud detection, anomaly detection, and fingerprint detection are some of the well-known applications used in surveillance matters.

- FACE DETECTION uses to detect the human faces in the image or video, but illumination and variation in poses and resolution make it difficult. Many notable innovations found in the past few years, such as Author [157], perform multiple tasks( facial landmarks localization with detection and head pose estimation) simultaneously without affecting the performance of an individual assignment. A novel approach, named Wasserstein's convolutional neural network (WCNN), uses to learn invariant features between visual and

near-infrared face images [158]. The architecture comprises low-level layers (trained on the broad visible spectrum of face images) and High-level layers (comprises of three parts, i.e., NIR, VIS, and hybrid NIR-VIS layer). It also designs the appropriate loss function that can enhance the discriminative power of DCNNs based, large-scale face recognition. However, cosine-based softmax losses [159]–[161] provide better results in deep learning-based face recognition.

High discriminative features were achieved using an Additive Angular Margin Loss(AcrFace) for face recognition [162]. Gue *et al.* [163] proposed an innovative technique for a single image per person for face recognition called fuzzy sparse auto-encoder.



**FIGURE 17.** Significant variation appears in imaging condition due to changes in the appearance of the same class (a, h) such as lighting effect, camera models, weather conditions, occlusion, physical locations, and viewing distance. Variation in pose, blur, motion, shading, clutter, occlusion, and scales adds challenges. The intra-class variation instances shown in (i). In contrast, cases in (j) have some examples of interclass—the majority of pictures from ImageNet [154] and MS COCO [104].

• PEDESTRIAN DETECTION aims to detect pedestrians in natural landscapes. The benchmark dataset for pedestrian detection is the EuroCity person dataset, which includes pedestrians, cyclists, and other riders in urban traffic scenes [164]. The cascaded approach uses for real-time pedestrian detection named Complexity-aware cascaded pedestrian detectors [165], [166]. For more details, please refer to the survey (deep learning-based pedestrian detection) [167].

• ANOMALY DETECTION is an instrumental tool in fraud detection, climate analysis, and any type of detection in healthcare monitoring. A point-wise approach uses to analyze the data in many anomaly detection techniques [168]–[170]. Some unsupervised methods have been used to search the contiguous interval of time and regions in space named “Maximally Divergent Intervals” (MDI) [171].

## B. OBJECT DETECTION IN MILITARY

Remote sensing object detection, topographic survey, flyer detection are some of the applications of the military field.

• REMOTE SENSING OBJECT DETECTION is a challenging task that used to detect objects on remote sensing images or videos. Existing object detection techniques for remote sensing is prolonged due to enormous input size with small targets, which makes it infeasible for practical use and hard to detect.

• Another hurdle is the extensive and complex background that leads to severe false detections. The researchers adopt the data fusion approach to address these issues. Due to lack of information and minor deviations, the main focus of the strat-

egy is small goals that lead to significant inaccuracy. Remote sensing images have different characteristics than natural images; thus, transfer learning to a new domain using robust architectures such as Faster R-CNN, FCN, SSD, and YOLO is not working well for remote sensing detection. Designing remote sensing dataset-specific for detector remains a hot research spot in this domain. Zhang et al [172] propose an approach to address the issue of lacking rotation and scaling invariance in RSI object detection using rotation and scaling robust structure. Cheng et al [173] propose a CNN-based RSI object detection models using the rotation-invariant layer to deal with rotation problems. The author suggests another effective method to learn a rotation-invariant and Fisher discriminative CNN model to solve the issues of object rotation, within-class variability, and between-class similarity [174].

Furthermore, the author uses the rotation-invariant and fisher discrimination regularity to optimize the new objective function and improve the performance of the existing framework [175]. Shahzad et al. [176] proposed a novel detection model based on automatic labeling and recurrent neural networks. Real-time remote sensing methods proposed in [177]. Long et al [178] proposed a framework that would concentrate on automatically and accurately locating objects. Li et al [179] proposed a novel framework based on RPN (to deal multi-scale and multi-angle features of geospatial objects) and local–contextual feature fusion network (to address the appearance ambiguity problem).

The methods proposed in [179]–[182] used deep neural networks to perform detection tasks on remote sensing datasets. Some of the remote-sensing object detection

benchmark datasets are VHR-10 [183], HRRSD [172], DOTA [184], DLR 3K Munich [185] and VEDAI [186]. For a more detailed study on remote sensing object detection, we recommend readers refer to [46], [187].

### C. OBJECT DETECTION IN TRANSPORTATION

Deep learning greatly facilitates humans in many applications of transportation fields such as autonomous driving, traffic sign recognition, and license plate recognition.

- LICENSE PLATE RECOGNITION is gaining fame with the popularity of automobiles industry related to crime tracking, residential access traffic violations tracking. License plate recognition models become more robust and stable with the use of edge information, sliding concentric windows, connected component analysis, texture features, and mathematical morphology. At the same time, many deep learning methods for license plate recognition provide beneficial assistance in daily life [188], [189].

- AUTONOMOUS DRIVING vehicle needs accurate estimates of their surroundings to operate reliably. Additionally, it is beneficial to transform the deep learning methods and sensory data into semantic information. 3D object detection methods provide information about size and location (monocular, point-cloud, and fusion). Monocular image-based detection predicts 2D bounding boxes than extrapolated them to 3D, which limits the accuracy of localization. Point-cloud based methods are time-consuming as it projects point clouds into a 2D image to generate a 3D representation directly in a structure.

At the same time, fusion-based techniques fuse both front view images and point-clouds to produce a robust detection. Lu *et al.* [190] proposed novel architecture based on 3D convolutions and RNNs, to generate a centimeter-level localization accuracy in different real-world driving scenarios. 3D car instance understanding and sensor fusion techniques are notable in autonomous driving [191], [192]. For further studies, please refer to the recently published survey [193].

- TRAFFIC SIGN RECOGNITION is an essential part of autonomous driving. Real-time accurate traffic sign recognition helps drive by acquiring temporal and spatial information of the potential sign. The literature contains very beneficial deep learning methods, such as [194]–[197].

### D. OBJECT DETECTION IN MEDICAL

Medical image detection (x-rays, CT images, MRI, fundus images), tumor detection, dental disease detection, skin disease detection, and healthcare monitoring are some of the active areas medicine where deep learning is contributing. The novel viruses are a significant issue for global public health. Technology can assist the medical practitioner to identify possible causes. It is beneficial for viral diseases like COVID-19 that can easily be transmitted and have asymptomatic infectivity periods. Hemdan *et al.* [198] use seven different architecture of CNN in COVIDx-Net, such as VGG19 and Google MobileNet v2. Each model can analyze the x-ray to classify the patient status (either infected or not).

- COMPUTER-AIDED DIAGNOSIS SYSTEM can assist the doctors in classifying and diagnosing the different types of cancer. The CAD framework has three main steps, such as image segmentation, feature extraction, classification, and object detection. Due to data privacy and scarcity, there usually exists a distribution difference of data between target and source domain. Therefore, medical image detection needs a domain adaptation framework [199].

- Deep learning has shown its perfection and miracles in the medical field, which have significant data in the form of images and numbers. Li *et al* [200] propose an attention mechanism in the CNN frameworks for *Glaucoma detection* and design large-scale attention-based glaucoma dataset. A DNA modifications detection framework (Deep-Mod) establish with the help of bidirectional RNN and long short-term memory(LSTM) [201]. Schubert *et al* [202] propose cellular morphology neural networks (CMNs) for automated neuron reconstruction and detection of synapses. For further detail, please refers to these surveys [203], [204].

### E. OBJECT DETECTION IN DAILY LIFE

The event detection, pattern detection, intelligent home, commodity detection, image caption generation, rain/ shadow detection, and species identification are some of the application of life fields. Goldman *et al* [205] proposed a novel object detector for densely packed scenes such as retail shelf displays and set up dataset SKU-110K to meet this challenge.

- EVENT DETECTION uses to discover real-world events on the Internet such as festivals, talks, protests, natural disasters, elections. Multi-domain event detection (MED) provides full details of the events. Yang *et al* [206] proposed an event detection framework for detecting real-world events from multi-domain data. Wang *et al* [207] design a novel event detector using online social interaction features and construct affinity graphs. Schinas *et al.* [208] incorporate 100 million photos/ videos to develop the multi-model graph-based system. For detailed information, please refer to surveys on event detection [209], [210].

- There are some challenges in PATTERN DETECTION, such as pose variation, varying illumination, scene occlusion, and sensor noise. The research literature about the repeated pattern or periodic structure detection provides a stable baseline in both 2D images [211], [212] and 3d cloud-points [213]–[216].

- IMAGE CAPTION GENERATION is a process in which a computer understands the semantic of an image and automatically generates a caption for the photograph in natural language. The process of image caption involves computer vision and natural language processing. These technologies are difficult to integrate. Multi-model embedding [217], encoder-decoder framework [218], [219], attention mechanism [220], [221], and reinforcement learning [222], [223] are widely used to address this issue. Yao *et al.* [224] proposed a novel framework using Graph Convolutional Network and LSTM (GCN-LSTM) to explore the connection between objects in spatial and semantic domains. For detailed

information, please refer to the image caption generation [225] survey.

- Rain detection, shadow detection, and species identification are some of the applications where deep learning performs significantly. Yang et al [226] proposed a novel joint rain detector to detect raindrops in a single image. Zheng et al. [227] proposed a Distraction-aware Shadow Detection Network (DSDNet) using explicit learning and integration of visual distraction regions semantics. Accurate identification of species is the basis of taxonomic research. Handegard et al. [228] used a deep learning model to classify the species present in the image automatically.

## VII. DISCUSSION

The following are some of the vital factors in detecting generic object:

### A. REGION BASE VERSUS CLASSIFICATION /REGRESSION BASE FRAMEWORKS

- A significant drawback of the region-based detector is the requirement for high computational power. Still, its structure is more flexible and efficient than the unified framework, which is suitable for region-based classification.

- One-stage detectors (YOLO and SSD) requires less time as compared to the two-stage framework due to light-weight backbone networks, avoiding pre-processing algorithms, fewer candidate region requirements for prediction, and the use of the FC subnetwork. The feature extractor (Backbone network) is the most time-consuming step in object detection [9], [127].

- In general, a unified detection framework has unsatisfactory performance and difficulty in detecting smaller objects [19], [84], [229].

- Fully-convolutional pipeline architecture, sliding windows from different layers of the backbone, its combined information, and exploring complementary data from other correlated tasks are some of the crucial design choices to design a better detection framework.

- The two-stage framework is the future of object detection in terms of a speed-accuracy trade-off because of the success of cascade for object detection [230]–[233] and instance segmentation on COCO [234].

### B. BACKBONE NETWORKS

The backbone network plays a vital role in the performance of object detection tasks. Generally more in-depth backbone framework such as ResNet [65], ResNeXt [156], Inception-ResNet [135], and Darknet53 require high computational power and big data for training to perform well. Some backbone networks are specially designed to focus on speed rather than accuracy, such as MobileNet [235].

### C. ROBUSTNESS IN OBJECT RECOGNITION DATASET

Real-world images have many variations in terms of brightness, angle of the image capturing, blur, deformations, background clutter, occlusion, resolution, noise, and camera

distortions, which makes it more challenging to detect the object. Object size/scale is significant in the object detection task. In contrast, different techniques are used to handle the pose variation and small object detection challenges such as the use of image pyramids by enlarging the small image and shrink the large one. Furthermore, various techniques such as the use of independent Conv feature maps (SSD [84]), incorporate dilated convolutions [139], [236], use of anchor with different scale, and aspect ratios with higher parameters, and up-scaling can be used for the small object detection [237], [238]. Super-resolution techniques still do not play an essential role in improving the detection accuracy of small objects compared to large ones. Besides that, some applications such as autonomous driving required only general identification of the existence of small objects rather than localization over a vast region.

- A spatial transformer network is used to handle occlusion, deformation, and other factors. Regression is used to obtain the deformation field and wraps the feature map in the deformation field [130]. A deformable part-based model [239] considers the spatial constraint to find the maximum response to a part filter [97], [100], [240]. The little research is dedicated to addressing the issue of rotation invariance and occlusion in generic object detection because of less relation variance found in famous benchmark object detection datasets, namely PASCAL VOC, COCO, and ImageNet. In contrast, face detection vigorously is based on occlusion handling study.

### D. DETECTION PROPOSAL

Detection proposals have significantly reduced search spaces. However, this undoubtedly requires improvement in the accuracy of localization, recall, speed, and repeatability for future detection proposals [241]. RPN is a dominant region proposal framework based on the CNN detection proposal generation method. It recommends that the proposed detection method in the future should be evaluated based on object detection rather than merely assessing the detection proposal.

### E. OTHER FACTORS

Other factors, such as novel training strategies, data augmentation, different combinations of backbone networks, and multiple detection frameworks, can affect the quality of object detection tasks. Some real-world challenges, such as object detection in mobility such as 3D point clouds, video, remotely sensed imagery, and RGBD images remain unresolved issues. Even with the advances in technology, object detection still yields unsatisfactory results from some constrained. Such as poorly labeled data or annotations with fewer bounding boxes, categories of unseen objects, wearable devices, and the ability to adapt and evolve several environmental changes to detect objects in the open world. The future research direction on these challenges is as follows:

1. In general, object detection algorithms do not have the ability to detect objects outside the training dataset. The

ultimate goal is to develop an object identification framework capable of localizing and recognizing the thousands of novel objects categories in the open-world scenes with accuracy and efficiency [242], [243]. Larger-scale datasets need to be developed with significantly more classes as existing benchmark datasets cover few hundreds of object categories that are far below the human-recognized categories.

2. The success of generic object detection mainly depends on detection frameworks. A unified framework is more straightforward and faster, while a Region-based detector is more accurate and efficient.

3. **Network acceleration** [244]–[248] and the design of a **compact, lightweight network** in the field of object detection is one of the new and growing research areas [235], [249]–[253]. Deeper CNN networks require more computational power, numbers of parameters, bulk data, and GPU for training.

4. Segmenting the object instance at the pixel-level requires a more vibrant and detailed understanding of image contents [80], [104], [254].

5. Currently, most state-of-the-art object detectors are fully supervised models that lack scalability due to the absence of fully annotated datasets. The data annotation process is laborious, become hardening with the volume of the dataset [104], [147], [154].

6. The success of the object detector majorly based on intensively large annotated training datasets. In contrast, the human can learn visual concepts very quickly from a few instances of events and can often generalize well [242], [255], [256]. Therefore, detecting the Few/Zero-Shot object is a very appealing task that should be done [242], [257]–[261].

7. New practices such as autonomous vehicles, robotics, and un-crewed aerial vehicles [262]–[264], video [265], [266], and point clouds [267], [268] are some of the challenges where object detection can play a distinct role.

The field of generic object detection still needs to complete substantial research efforts. However, the last five years have been a significant and golden time for object detection. We are optimistic about future developments and opportunities in the field of object detection.

## VIII. FUTURE DIRECTION AND CURRENT TRENDS

### A. HYBRID APPROACH

The two-stage detector is time-consuming and inefficient because it uses a dense tailing process to obtain the most reference boxes. This problem can be solved by maintaining high accuracy and avoiding affordable redundancy. On the other hand, due to the fast processing speed, the one-stage detector is very suitable for real-time applications. Its low accuracy is still a barrier to the use of high precision requirement applications. These methods need to combine to take advantage of both one-stage and two-stage detectors. But how to bring them together is a big challenge.

### B. OBJECT DETECTION IN VIDEO (DYNAMIC TARGETS)

It is challenging to achieve an excellent video object detection performance in a real-life scene and remote scene due to video defocus, motion target ambiguity, motion blur, small objects, occlusion, truncation, and intense target movements. Researchers can focus on more complex source data and dynamic targets for future research.

### C. EFFICIENT POST-PROCESSING METHODS

Post-processing is the initial step for the final results in the three (for one-stage detector) or four (for a two-stage-detector) stage detection procedure. The accuracy score of the detector is evaluated by sending the highest prediction results of an object in a metric program. The post-processing methods such as NMS and its improvements can eliminate well-located but high classification confidence objects. Experimenting with more efficient and accurate post-processing methods is another direction for the researchers.

### D. WEAKLY SUPERVISED OBJECT DETECTION METHODS

Due to availability and to achieve high efficiency, it is more fruitful for network training to replace a significant portion of fully-annotated images with high proportion labeled images that only have class labels but does not have object bounding boxes. Besides that, the weakly supervised object detection uses a limited amount of fully annotated images to detect non-fully annotated ones. Therefore, the availability of non-annotated big data diverts our attention to a significant problem, such as the development of WSOD methods.

### E. OBJECT DETECTION IN MULTI-DOMAIN

The detection performance of a specific domain-related detector in a particular domain (dataset) is always high. Therefore, there is a need for a universal-detector known as a multi-domain detector that is capable of working on various domain images without prior knowledge of the new domain. Therefore, domain transfer is difficult without affecting performance.

### F. 3D OBJECT DETECTION

3D object detection becomes a hot and active research direction with the invention of 3D sensors and diverse applications of 3D comprehension. The LiDAR point cloud can be used to locate the objects accurately and describe their shapes and provide reliable depth information. It can be feasible to use object detection techniques of LiDAR data for 2D data as well.

### G. SALIENCY DETECTION

Salient object detection emphasizes highlighting significant object regions in the images. At the same time, the object of interest in video object detection is classified and located in a continuous scene. SOD can be applied to a broad spectrum of object-level applications in various areas. It can also assist in accurately detecting the object by providing a salient region

of interest in each frame of video. Therefore, it can be helpful in a high-level recognition task, challenging detection task, and highlighting target detection.

#### **H. UNSUPERVISED OBJECT DETECTION**

Supervise methods for object detection requires a well-annotated dataset for the training process, which is time expensive and inefficient. Bounding box annotation of each object in large datasets requires a significant amount of time, effort, and impractical. It is needed to develop automatic annotation strategies to eliminate human annotation requirements in the supervised object detection task.

#### **I. FEATURE FUSION & MULTI-TASK LEARNING**

Feature fusion is a process that is used to improve the detection performance by aggregating the feature from multiple levels. Furthermore, performing various tasks simultaneously, such as semantic and instance segmentation along with object detection, can improve the efficiency of each task due to in-depth information. Maintaining processing speed and improve accuracy during multi-task learning is a challenging task for the researcher.

#### **J. MULTI-SOURCE INFORMATION ASSISTANCE**

Access to multi-source information is convenient due to the development of big data technology and the popularity of social media. Many social media sources also provide textual descriptions along with pictures, which can assist in object detection tasks. The fusion of multidisciplinary information could lead to future research direction for the researcher.

#### **K. TERMINAL OBJECT DETECTION SYSTEM**

AI Terminalization can help to deal with a massive amount of information and solve the problem in a better and faster. Lightweight networks emerge from developing a more efficient and reliable terminal detector used in a variety of applications. The FPGA based detection network is very feasible for real-time applications.

#### **L. MEDICAL IMAGING AND DIAGNOSIS**

AI-based Medical Devices are getting fame due to its promising accuracy. The FDA (U.S.Food and Drug Administration) approves the use of AI-based software called IDX-DR, for detecting diabetic retinopathy with an accuracy of more than 87.4% in April 2018. A combination of image recognition and smart devices makes the cell phone a powerful family diagnostic tool. The current state of epidemics in the world, such as COVID-19, increases the need for technology. This direction is full of challenges and expectations.

#### **M. ADVANCE MEDICAL BIOMETRICS**

Medical risk factors can be studied and monitored more effectively by using a deep neural network that had been difficult to quantify previously. Medical images such as retinal (fundus) images and speech patterns may help identify the risk of heart disease. Similarly, X-ray, Ct images, and immune pattern

monitoring may help to diagnose other significant disorders. Soon, passive monitoring can be possible with medical biometrics.

#### **N. REAL-TIME DETECTION AND REMOTE SENSING AIRBORNE**

Precise analysis of remote sensing images is very beneficial for agriculture fields and military defense. Automatic detection software and integrated hardware can open new opportunities for countries in these fields.

#### **O. GAN BASED DETECTOR**

Data augmentation always helps in deep learning. The deep learning-based systems require a massive amount of images for the training process and a powerful technique of data augmentation, such as Generative Adversarial Network that used to generate fake images closer to reality. Object detector becomes more robust and obtains strong generalization ability using a combination of the real-world scene, and GAN made simulated data.

#### **REFERENCES**

- [1] K.-K. Sung and T. Poggio, "Example-based learning for view-based human face detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 1, pp. 39–51, 1998.
- [2] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2012.
- [3] H. Kobatake and Y. Yoshinaga, "Detection of spicules on mammogram based on skeleton analysis," *IEEE Trans. Med. Imag.*, vol. 15, no. 3, pp. 235–245, Jun. 1996.
- [4] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 7291–7299.
- [5] Z. Yang and R. Nevatia, "A multi-scale cascade fully convolutional network face detector," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 633–638.
- [6] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, vol. 2014, pp. 675–678.
- [7] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "DeepDriving: Learning affordance for direct perception in autonomous driving," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2722–2730.
- [8] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1907–1915.
- [9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
- [10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.
- [11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.
- [12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. CVPR*, 2005, pp. 886–893.
- [13] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in *Proc. Int. Conf. Image Process.*, vol. 1, 2002, p. 1.
- [14] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997.
- [15] C. Cortes and V. Vapnik, "Support vector machine," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
- [16] S. Ren, K. He, R. Girshick, X. Zhang, and J. Sun, "Object detection networks on convolutional feature maps," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 7, pp. 1476–1481, Jul. 2017.

- [17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [18] R. Girshick, "Fast R-CNN ICCV," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
- [19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.
- [20] N. Liu, J. Han, D. Zhang, S. Wen, and T. Liu, "Predicting eye fixations using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 362–370.
- [21] J. Chen, Q. Li, W. Wu, H. Ling, L. Wu, B. Zhang, and P. Li, "Saliency detection via topological feature modulated deep learning," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 1630–1634.
- [22] H. Jiang and E. Learned-Miller, "Face detection with the faster R-CNN," in *Proc. 12th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2017, pp. 650–657.
- [23] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun, "Joint cascade face detection and alignment," in *Proc. Eur. Conf. Comput. Vis. Springer*, 2014, pp. 109–122.
- [24] A. Mateus, D. Ribeiro, P. Miraldo, and J. C. Nascimento, "Efficient and robust pedestrian detection using deep learning for human-aware navigation," 2016, *arXiv:1607.04441*. [Online]. Available: <http://arxiv.org/abs/1607.04441>
- [25] F. Yang, W. Choi, and Y. Lin, "Exploit all the layers: Fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2129–2137.
- [26] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 7, pp. 1480–1500, Jul. 2015.
- [27] M.-H. Yang, D. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 1, pp. 34–58, Jan. 2002.
- [28] S. Zafeiriou, C. Zhang, and Z. Zhang, "A survey on face detection in the wild: Past, present and future," *Comput. Vis. Image Understand.*, vol. 138, pp. 1–24, Sep. 2015.
- [29] M. Enzweiler and D. M. Gavrila, "Monocular pedestrian detection: Survey and experiments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2179–2195, Dec. 2009.
- [30] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 7, pp. 1239–1258, Jul. 2010.
- [31] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 694–711, May 2006.
- [32] K. Grauman and B. Leibe, "Visual object recognition," *Synth. Lectures Artif. Intell. Mach. Learn.*, vol. 5, no. 2, pp. 1–181, 2011.
- [33] A. Andreopoulos and J. K. Tsotsos, "50 years of object recognition: Directions forward," *Comput. Vis. Image Understand.*, vol. 117, no. 8, pp. 827–891, 2013.
- [34] X. Zhang, Y.-H. Yang, Z. Han, H. Wang, and C. Gao, "Object class detection: A survey," *ACM Comput. Surv.*, vol. 46, no. 1, pp. 1–53, 2013.
- [35] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
- [36] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, Dec. 2017.
- [37] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5325–5334.
- [38] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.
- [39] Y. Zhou, D. Liu, and T. Huang, "Survey of face detection on low-quality images," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2018, pp. 769–773.
- [40] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster R-CNN doing well for pedestrian detection?" in *Proc. Eur. Conf. Comput. Vis. Springer*, 2016, pp. 443–457.
- [41] J. Hosang, M. Omran, R. Benenson, and B. Schiele, "Taking a deeper look at pedestrians," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4073–4082.
- [42] Y. Zhou, L. Liu, L. Shao, and M. Mellor, "DAVE: A unified framework for fast vehicle detection and annotation," in *Proc. Eur. Conf. Comput. Vis. Springer*, 2016, pp. 278–293.
- [43] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2110–2118.
- [44] A. Kumar, A. Kaur, and M. Kumar, "Face detection techniques: A review," *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 927–948, Aug. 2019.
- [45] V. Ramakrishnan, A. K. Prabhavathy, and J. Devishree, "A survey on vehicle detection techniques in aerial surveillance," *Int. J. Comput. Appl.*, vol. 55, no. 18, pp. 43–47, 2012.
- [46] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, "Object detection in optical remote sensing images: A survey and a new benchmark," *ISPRS J. Photogramm. Remote Sens.*, vol. 159, pp. 296–307, Jan. 2020.
- [47] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, "A survey on 3D object detection methods for autonomous driving applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3782–3795, Oct. 2019.
- [48] X.-C. Yin, Z.-Y. Zuo, S. Tian, and C.-L. Liu, "Text detection, tracking and recognition in video: A comprehensive survey," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2752–2773, Jun. 2016.
- [49] S. Long, X. He, and C. Yao, "Scene text detection and recognition: The deep learning era," 2018, *arXiv:1811.04256*. [Online]. Available: <http://arxiv.org/abs/1811.04256>
- [50] S. F. Joan and S. Valli, "A survey on text information extraction from born-digital and scene text images," *Proc. Nat. Acad. Sci., India Sect. A, Phys. Sci.*, vol. 89, no. 1, pp. 77–101, 2019.
- [51] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," 2019, *arXiv:1905.05055*. [Online]. Available: <http://arxiv.org/abs/1905.05055>
- [52] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 261–318, Feb. 2020.
- [53] W. Pitts and W. S. McCulloch, "How we know universals the perception of auditory and visual forms," *Bull. Math. Biophys.*, vol. 9, no. 3, pp. 127–147, Sep. 1947.
- [54] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Cognit. Model.*, vol. 5, no. 3, p. 1, 1988.
- [55] G. H. L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.
- [56] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
- [57] L. Deng, M. L. Seltzer, D. Yu, A. Acero, A.-R. Mohamed, and G. Hinton, "Binary coding of speech spectrograms using a deep auto-encoder," in *Proc. 11th Annu. Conf. Int. Speech Commun. Assoc.*, 2010, pp. 1692–1695.
- [58] G. Dahl, M. A. Ranzato, A.-R. Mohamed, and G. E. Hinton, "Phone recognition with the mean-covariance restricted Boltzmann machine," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 469–477.
- [59] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [60] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012, *arXiv:1207.0580*. [Online]. Available: <http://arxiv.org/abs/1207.0580>
- [61] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: <http://arxiv.org/abs/1502.03167>
- [62] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
- [63] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: <http://arxiv.org/abs/1409.1556>
- [64] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," 2013, *arXiv:1312.6229*. [Online]. Available: <http://arxiv.org/abs/1312.6229>

- [65] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
- [66] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Weakly supervised object recognition with convolutional neural networks," in *Proc. NIPS*, 2014, pp. 1545–5963.
- [67] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1717–1724.
- [68] F. Wadley, "Probit analysis: A statistical treatment of the sigmoid response curve. 2nd ed. D. J. Finney. New York-London: Cambridge Univ. Press, 1952. 318 pp. \$7.00," *Amer. Assoc. Advancement Sci.*, vol. 41, no. 11, pp. 627–627, Nov. 1952.
- [69] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. LeCun, "Learning invariant features through topographic filter maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1605–1612.
- [70] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2528–2535.
- [71] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1520–1528.
- [72] Z.-Q. Zhao, B.-J. Xie, Y.-M. Cheung, and X. Wu, "Plant leaf identification via a growing convolution neural network with progressive sample learning," in *Proc. 12th Asian Conf. Comput. Vis.* Singapore: Springer, Nov. 2014, pp. 348–361.
- [73] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky, "Neural codes for image retrieval," in *Proc. Eur. Conf. Comput. Vis.* Zürich, Switzerland: Springer, 2014, pp. 584–599.
- [74] J. Wan, D. Wang, S. C. H. Hoi, P. Wu, J. Zhu, Y. Zhang, and J. Li, "Deep learning for content-based image retrieval: A comprehensive study," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 157–166.
- [75] D. Tomè, F. Monti, L. Baroffio, L. Bondi, M. Tagliasacchi, and S. Tubaro, "Deep convolutional neural networks for pedestrian detection," *Signal Process., Image Commun.*, vol. 47, pp. 482–489, Sep. 2016.
- [76] Z.-Q. Zhao, H. Bian, D. Hu, W. Cheng, and H. Glotin, "Pedestrian detection based on fast R-CNN and batch normalization," in *Proc. Adv. Neural Inf. Process. Syst.* Spain: Springer, 2016, pp. 735–746.
- [77] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.
- [78] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379–387.
- [79] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2117–2125.
- [80] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2961–2969.
- [81] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2147–2154.
- [82] D. Yoo, S. Park, J.-Y. Lee, A. S. Paek, and I. S. Kweon, "AttentionNet: Aggregating weak directions for accurate object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2659–2667.
- [83] M. Najibi, M. Rastegari, and L. S. Davis, "G-CNN: An iterative grid based object detector," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2369–2377.
- [84] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. 14th Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 21–37.
- [85] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 7263–7271.
- [86] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," 2017, *arXiv:1701.06659*. [Online]. Available: <http://arxiv.org/abs/1701.06659>
- [87] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "DSOD: Learning deeply supervised object detectors from scratch," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 1919–1927.
- [88] X. Ren and D. Ramanan, "Histograms of sparse codes for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3246–3253.
- [89] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, Sep. 2013.
- [90] P. Arbelaez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *Proc. CVPR*, 2014, pp. 328–335.
- [91] P. Krähenbühl and V. Koltun, "Geodesic object proposals," in *Proc. 13th Eur. Conf. Comput. Vis.* Zürich, Switzerland: Springer, Sep. 2014, pp. 725–739.
- [92] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2014, pp. 391–405.
- [93] W. Kuo, B. Hariharan, and J. Malik, "DeepBox: Learning objectness with convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2479–2487.
- [94] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár, "Learning to refine object segments," in *Proc. 14th Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 75–91.
- [95] S. Gupta, R. Girshick, P. Arbelaez, and J. Malik, "Learning rich features from RGB-D images for object detection and segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 345–360.
- [96] Y. Zhang, K. Sohn, R. Villegas, G. Pan, and H. Lee, "Improving object detection with deep convolutional networks via Bayesian optimization and structured prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 249–258.
- [97] W. Ouyang, X. Wang, X. Zeng, S. Qiu, P. Luo, Y. Tian, H. Li, S. Yang, Z. Wang, C.-C. Loy, and X. Tang, "DeepID-net: Deformable deep convolutional neural networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 2403–2412.
- [98] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2006, pp. 2169–2178.
- [99] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher kernel for large-scale image classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Heraklion, Greece: Springer, Sep. 2010, pp. 143–156.
- [100] R. Girshick, F. Iandola, T. Darrell, and J. Malik, "Deformable part models are convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 437–446.
- [101] J. Xue, J. Li, and Y. Gong, "Restructuring of deep neural network acoustic models with singular value decomposition," in *Proc. Interspeech*, 2013, pp. 2365–2369.
- [102] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2818–2826.
- [103] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. (2011). *The Pascal Visual Object Classes Challenge 2012 (voc2012) Results* (2012). [Online]. Available: <http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html>
- [104] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proc. 13th Eur. Conf. Comput. Vis. (ECCV)*. Zürich, Switzerland: Springer, Sep. 2014, pp. 740–755.
- [105] Y. Wang, "Multi-candidate association online multi-target tracking based on R-FCN framework," *Opto-Electron. Eng.*, vol. 47, no. 1, 2020, Art. no. 190136.
- [106] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3150–3158.
- [107] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.
- [108] J. Nan and L. Bo, "Infrared object image instance segmentation based on improved mask-RCNN," *Proc. SPIE*, vol. 11187, Nov. 2019, Art. no. 111871E.
- [109] A. O. Vuola, S. U. Akram, and J. Kannala, "Mask-RCNN and U-Net ensembled for nuclei segmentation," in *Proc. IEEE 16th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2019, pp. 208–212.
- [110] S. Brahmbhatt, H. I. Christensen, and J. Hays, "StuffNet: Using 'Stuff' to improve object detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 934–943.

- [111] S. Bell, C. L. Zitnick, K. Bala, and R. Girshick, "Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2874–2883.
- [112] T. Kong, A. Yao, Y. Chen, and F. Sun, "HyperNet: Towards accurate region proposal generation and joint object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 845–853.
- [113] A. Pentina, V. Sharmanska, and C. H. Lampert, "Curriculum learning of multiple tasks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5492–5500.
- [114] J. Yim, H. Jung, B. Yoo, C. Choi, D. Park, and J. Kim, "Rotating your face using multi-task deep neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 676–684.
- [115] J. Li, X. Liang, J. Li, Y. Wei, T. Xu, J. Feng, and S. Yan, "Multistage object detection with group recursive learning," *IEEE Trans. Multimedia*, vol. 20, no. 7, pp. 1645–1655, Jul. 2018.
- [116] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Proc. 4th Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 354–370.
- [117] Y. Zhu, R. Urtasun, R. Salakhutdinov, and S. Fidler, "SegDeepM: Exploiting segmentation and context in deep neural networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4703–4711.
- [118] X. Zeng, W. Ouyang, B. Yang, J. Yan, and X. Wang, "Gated bi-directional cnn for object detection," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*. Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 354–369.
- [119] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2553–2561.
- [120] Q.-C. Mao, H.-M. Sun, Y.-B. Liu, and R.-S. Jia, "Mini-YOLOv3: Real-time object detector for embedded applications," *IEEE Access*, vol. 7, pp. 133529–133538, 2019.
- [121] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: <http://arxiv.org/abs/1804.02767>
- [122] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2980–2988.
- [123] C. Ning, H. Zhou, Y. Song, and J. Tang, "Inception single shot MultiBox detector for object detection," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2017, pp. 549–554.
- [124] M. Cheng, J. Bai, L. Li, Q. Chen, X. Zhou, H. Zhang, and P. Zhang, "Tiny-RetinaNet: A one-stage detector for real-time object detection," in *Proc. 11th Int. Conf. Graph. Image Process. (ICGIP)*, vol. 11373, Jan. 2020, Art. no. 113730R.
- [125] Q. Zhao, T. Sheng, Y. Wang, Z. Tang, Y. Chen, L. Cai, and H. Ling, "M2Det: A single-shot object detector based on multi-level feature pyramid network," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 9259–9266.
- [126] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Single-shot refinement neural network for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4203–4212.
- [127] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 734–750.
- [128] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 10781–10790.
- [129] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei, "Relation networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3588–3597.
- [130] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 764–773.
- [131] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable ConvNets v2: More deformable, better results," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 9308–9316.
- [132] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "NAS-FPN: Learning scalable feature pyramid architecture for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 7036–7045.
- [133] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, and T. Chen, "Recent advances in convolutional neural networks," *Pattern Recognit.*, vol. 77, pp. 354–377, May 2018.
- [134] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 818–833.
- [135] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell. (AAAI)*, Feb. 2016, pp. 1–12. [Online]. Available: <https://arxiv.org/abs/1602.07261>
- [136] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4700–4708.
- [137] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 472–480.
- [138] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1251–1258.
- [139] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, and J. Sun, "DetNet: A backbone network for object detection," 2018, *arXiv:1804.06215*. [Online]. Available: <http://arxiv.org/abs/1804.06215>
- [140] Y. Chen, J. Li, H. Xiao, X. Jin, S. Yan, and J. Feng, "Dual path networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4467–4475.
- [141] S. Sun, J. Pang, J. Shi, S. Yi, and W. Ouyang, "FishNet: A versatile backbone for image, region, and pixel level prediction," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 754–764.
- [142] Y. Liu, Y. Wang, S. Wang, T. Liang, Q. Zhao, Z. Tang, and H. Ling, "CBNet: A novel composite backbone network architecture for object detection," 2019, *arXiv:1909.03625*. [Online]. Available: <http://arxiv.org/abs/1909.03625>
- [143] Y. Chen, T. Yang, X. Zhang, G. Meng, X. Xiao, and J. Sun, "DetNAS: Backbone search for object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 6638–6648.
- [144] Y. Chen, M. Rohrbach, Z. Yan, Y. Shuicheng, J. Feng, and Y. Kalantidis, "Graph-based global reasoning networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 433–442.
- [145] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes challenge 2007 (VOC2007) results," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.
- [146] I. Krasin *et al.*, "Openimages: A public dataset for large-scale multi-label and multi-class image classification," vol. 2, no. 3, pp. 2–3, 2017. [Online]. Available: <https://github.com/openimages>
- [147] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes challenge: A retrospective," *Int. J. Comput. Vis.*, vol. 111, no. 1, pp. 98–136, Jan. 2015.
- [148] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.
- [149] B. Singh and L. S. Davis, "An analysis of scale invariance in object Detection-SNIP" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3578–3587.
- [150] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov, T. Duerig, and V. Ferrari, "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," 2018, *arXiv:1811.00982*. [Online]. Available: <http://arxiv.org/abs/1811.00982>
- [151] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 761–769.
- [152] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Subcategory-aware convolutional neural networks for object proposals and detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 924–933.
- [153] A. Dundar, J. Jin, B. Martini, and E. Culurciello, "Embedded streaming deep neural networks accelerator with applications," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 7, pp. 1572–1583, Jul. 2017.
- [154] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
- [155] D. Hoiem, Y. Chodpathumwan, and Q. Dai, "Diagnosing error in object detectors," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2012, pp. 340–353.
- [156] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1492–1500.

- [157] R. Ranjan, V. M. Patel, and R. Chellappa, "HyperFace: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 1, pp. 121–135, Jan. 2019.
- [158] R. He, X. Wu, Z. Sun, and T. Tan, "Wasserstein CNN: Learning invariant features for NIR-VIS face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1761–1773, Jul. 2019.
- [159] X. Zhang, R. Zhao, Y. Qiao, X. Wang, and H. Li, "AdaCos: Adaptively scaling cosine logits for effectively learning deep face representations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 10823–10832.
- [160] Y. Liu, H. Li, and X. Wang, "Rethinking feature discrimination and polymerization for large-scale recognition," 2017, *arXiv:1710.00870*. [Online]. Available: <http://arxiv.org/abs/1710.00870>
- [161] R. Ranjan, C. D. Castillo, and R. Chellappa, "L2-constrained softmax max loss for discriminative face verification," 2017, *arXiv:1703.09507*. [Online]. Available: <http://arxiv.org/abs/1703.09507>
- [162] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 4690–4699.
- [163] Y. Guo, L. Jiao, S. Wang, S. Wang, and F. Liu, "Fuzzy sparse autoencoder framework for single image per person face recognition," *IEEE Trans. Cybern.*, vol. 48, no. 8, pp. 2402–2415, Aug. 2018.
- [164] M. Braun, S. Krebs, F. Flohr, and D. M. Gavrila, "EuroCity persons: A novel benchmark for person detection in traffic scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1844–1861, Aug. 2019.
- [165] Z. Cai, M. Saberian, and N. Vasconcelos, "Learning complexity-aware cascades for deep pedestrian detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3361–3369.
- [166] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, Aug. 2014.
- [167] A. Brunetti, D. Buongiorno, G. F. Trotta, and V. Bevilacqua, "Computer vision and deep learning techniques for pedestrian detection and tracking: A survey," *Neurocomputing*, vol. 300, pp. 17–33, Jul. 2018.
- [168] S. Liu, M. Yamada, N. Collier, and M. Sugiyama, "Change-point detection in time-series data by relative density-ratio estimation," *Neural Netw.*, vol. 43, pp. 72–83, Jul. 2013.
- [169] P. Senin, J. Lin, X. Wang, T. Oates, S. Gandhi, A. P. Boedihardjo, C. Chen, and S. Frankenstein, "GrammarViz 3.0: Interactive discovery of variable-length time series patterns," *ACM Trans. Knowl. Discovery from Data*, vol. 12, no. 1, pp. 1–28, 2018.
- [170] M. Jiang, A. Beutel, P. Cui, B. Hooi, S. Yang, and C. Faloutsos, "A general suspiciousness metric for dense blocks in multimodal data," in *Proc. IEEE Int. Conf. Data Mining*, Nov. 2015, pp. 781–786.
- [171] E. Rodner, B. Barz, Y. Guanche, M. Flach, M. Mahecha, P. Bodesheim, M. Reichstein, and J. Denzler, "Maximally divergent intervals for anomaly detection," 2016, *arXiv:1610.06761*. [Online]. Available: <http://arxiv.org/abs/1610.06761>
- [172] Y. Zhang, Y. Yuan, Y. Feng, and X. Lu, "Hierarchical and robust convolutional neural network for very high-resolution remote sensing object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5535–5548, Aug. 2019.
- [173] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.
- [174] G. Cheng, P. Zhou, and J. Han, "RIFD-CNN: Rotation-invariant and Fisher discriminative convolutional neural networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2884–2893.
- [175] G. Cheng, J. Han, P. Zhou, and D. Xu, "Learning rotation-invariant and Fisher discriminative convolutional neural networks for object detection," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 265–278, Jan. 2019.
- [176] M. Shahzad, M. Maurer, F. Fraundorfer, Y. Wang, and X. X. Zhu, "Buildings detection in VHR SAR images using fully convolution neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1100–1116, Feb. 2019.
- [177] J. Pei, Y. Huang, W. Huo, Y. Zhang, J. Yang, and T.-S. Yeo, "SAR automatic target recognition based on multiview deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2196–2210, Apr. 2018.
- [178] Y. Long, Y. Gong, Z. Xiao, and Q. Liu, "Accurate object localization in remote sensing images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2486–2498, May 2017.
- [179] K. Li, G. Cheng, S. Bu, and X. You, "Rotation-insensitive and context-augmented object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2337–2348, Apr. 2018.
- [180] F. Zhang, B. Du, L. Zhang, and M. Xu, "Weakly supervised learning based on coupled convolutional neural networks for aircraft detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 9, pp. 5553–5563, Sep. 2016.
- [181] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, "Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3325–3337, Jun. 2015.
- [182] Q. Li, Y. Wang, Q. Liu, and W. Wang, "Hough transform guided deep feature extraction for dense building detection in remote sensing images," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 1872–1876.
- [183] G. Cheng, J. Han, P. Zhou, and L. Guo, "Multi-class geospatial object detection and geographic image classification based on collection of part detectors," *ISPRS J. Photogramm. Remote Sens.*, vol. 98, pp. 119–132, Dec. 2014.
- [184] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3974–3983.
- [185] K. Liu and G. Mattyus, "Fast multiclass vehicle detection on aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 9, pp. 1938–1942, Sep. 2015.
- [186] S. Razakarivony and F. Jurie, "Vehicle detection in aerial imagery: A small target detection benchmark," *J. Vis. Commun. Image Represent.*, vol. 34, pp. 187–203, Jan. 2016.
- [187] G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, vol. 117, pp. 11–28, Jul. 2016.
- [188] P. Shivakumara, D. Tang, M. Asadzadehkaljahi, T. Lu, U. Pal, and M. H. Anisi, "CNN-RNN based method for license plate recognition," *CAAI Trans. Intell. Technol.*, vol. 3, no. 3, pp. 169–175, Sep. 2018.
- [189] I. Palij, V. Turchenko, V. Koval, A. Sachenko, and G. Markowsky, "Approach to recognition of license plate numbers using neural networks," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, vol. 4, Jul. 2004, pp. 2965–2970.
- [190] W. Lu, Y. Zhou, G. Wan, S. Hou, and S. Song, "L3-net: Towards learning based LiDAR localization for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 6389–6398.
- [191] X. Song, P. Wang, D. Zhou, R. Zhu, C. Guan, Y. Dai, H. Su, H. Li, and R. Yang, "ApolloCar3D: A large 3D car instance understanding benchmark for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 5452–5462.
- [192] K. Banerjee, D. Notz, J. Windelen, S. Gavarraju, and M. He, "Online camera LiDAR fusion and object detection on hybrid data for autonomous driving," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1632–1638.
- [193] M. Liang, B. Yang, S. Wang, and R. Urtasun, "Deep continuous fusion for multi-sensor 3d object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 641–656.
- [194] J. Li and Z. Wang, "Real-time traffic sign recognition based on efficient CNNs in the wild," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 975–984, Mar. 2019.
- [195] H.-Y. Lin, C.-C. Chang, V. L. Tran, and J.-H. Shi, "Improved traffic sign recognition for in-car cameras," *J. Chin. Inst. Engineers*, vol. 43, no. 3, pp. 300–307, 2020.
- [196] Y. Wu, Z. Li, Y. Chen, K. Nai, and J. Yuan, "Real-time traffic sign detection and classification towards real traffic scene," *Multimedia Tools Appl.*, vol. 79, pp. 18201–18219, Mar. 2020.
- [197] C. Zhang, X. Yue, R. Wang, N. Li, and Y. Ding, "Study on traffic sign recognition by optimized Lenet-5 algorithm," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 34, no. 1, Jan. 2020, Art. no. 2055003.
- [198] E. El-D. Hemdan, M. A. Shouman, and M. E. Karar, "COVIDX-Net: A framework of deep learning classifiers to diagnose COVID-19 in X-ray images," 2020, *arXiv:2003.11055*. [Online]. Available: <http://arxiv.org/abs/2003.11055>

- [199] Z. Li, M. Dong, S. Wen, X. Hu, P. Zhou, and Z. Zeng, “CLU-CNNs: Object detection for medical images,” *Neurocomputing*, vol. 350, pp. 53–59, Jul. 2019.
- [200] L. Li, M. Xu, X. Wang, L. Jiang, and H. Liu, “Attention based glaucoma detection: A large-scale database and CNN model,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 10571–10580.
- [201] Q. Liu, L. Fang, G. Yu, D. Wang, C.-L. Xiao, and K. Wang, “Detection of DNA base modifications by deep recurrent neural network on Oxford nanopore sequencing data,” *Nature Commun.*, vol. 10, no. 1, pp. 1–11, Dec. 2019.
- [202] P. J. Schubert, S. Dorkenwald, M. Januszewski, V. Jain, and J. Kornfeld, “Learning cellular morphology with neural networks,” *Nature Commun.*, vol. 10, no. 1, pp. 1–12, Dec. 2019.
- [203] S. Naji, H. A. Jalab, and S. A. Kareem, “A survey on skin detection in colored images,” *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 1041–1087, Aug. 2019.
- [204] F. Altaf, S. M. S. Islam, N. Akhtar, and N. K. Janjua, “Going deep in medical image analysis: Concepts, methods, challenges, and future directions,” *IEEE Access*, vol. 7, pp. 99540–99572, 2019.
- [205] E. Goldman, R. Herzig, A. Eisenschatz, J. Goldberger, and T. Hassner, “Precise detection in densely packed scenes,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 5227–5236.
- [206] Z. Yang, Q. Li, L. Wenyin, and J. Lv, “Shared multi-view data representation for multi-domain event detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1243–1256, May 2019.
- [207] Y. Wang, H. Sundaram, and L. Xie, “Social event detection with interaction graph modeling,” in *Proc. 20th ACM Int. Conf. Multimedia*, 2012, pp. 865–868.
- [208] S. Oh, S. McCloskey, I. Kim, A. Vahdat, K. J. Cannons, H. Hajimirsadeghi, G. Mori, A. G. A. Perera, M. Pandey, and J. J. Corso, “Multimedia event detection with multimodal feature fusion and temporal concept localization,” *Mach. Vis. Appl.*, vol. 25, no. 1, pp. 49–69, Jan. 2014.
- [209] A. Goswami and A. Kumar, “A survey of event detection techniques in online social networks,” *Social Netw. Anal. Mining*, vol. 6, no. 1, p. 107, Dec. 2016.
- [210] Z. Saeed, R. A. Abbasi, O. Maqbool, A. Sadaf, I. Razzak, A. Daud, N. R. Aljohani, and G. Xu, “What’s happening around the world? A survey and framework on event detection techniques on twitter,” *J. Grid Comput.*, vol. 17, no. 2, pp. 279–312, 2019.
- [211] O. Teboul, I. Kokkinos, L. Simon, P. Koutsourakis, and N. Paragios, “Shape grammar parsing via reinforcement learning,” in *Proc. CVPR*, Jun. 2011, pp. 2273–2280.
- [212] P. Zhao, T. Fang, J. Xiao, H. Zhang, Q. Zhao, and L. Quan, “Rectilinear parsing of architecture in urban environment,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 342–349.
- [213] G. Ferrigno, N. A. Borghese, and A. Pedotti, “Pattern recognition in 3D automatic human motion analysis,” *ISPRS J. Photogramm. Remote Sens.*, vol. 45, no. 4, pp. 227–246, Aug. 1990.
- [214] G. D. Pais, S. Ramalingam, V. M. Govindu, J. C. Nascimento, R. Chellappa, and P. Miraldo, “3DRegNet: A deep neural network for 3D point registration,” 2019, *arXiv:1904.01701*. [Online]. Available: <http://arxiv.org/abs/1904.01701>
- [215] D. Ryumin, I. Kagirov, D. Ivanko, A. Axyonov, and A. A. Karpov, “Automatic detection and recognition of 3D manual gestures for human-machine interaction,” *ISPRS Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. XLII-2/W12, pp. 179–183, May 2019.
- [216] Z. Zhang, Z. Li, N. Bi, J. Zheng, J. Wang, K. Huang, W. Luo, Y. Xu, and S. Gao, “PPGNet: Learning point-pair graph for line segment detection,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 7105–7114.
- [217] J. Hitschler, S. Schamoni, and S. Riezler, “Multimodal pivots for image caption translation,” 2016, *arXiv:1601.03916*. [Online]. Available: <http://arxiv.org/abs/1601.03916>
- [218] W. Jiang, L. Ma, X. Chen, H. Zhang, and W. Liu, “Learning to guide decoding for image captioning,” in *Proc. 32nd Conf. Artif. Intell. (AAAI)*, 2018, pp. 1–8.
- [219] K. Cho, A. Courville, and Y. Bengio, “Describing multimedia content using attention-based encoder-decoder networks,” *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 1875–1886, Nov. 2015.
- [220] L. Li, S. Tang, L. Deng, Y. Zhang, and Q. Tian, “Image caption with global-local attention,” in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4133–4139.
- [221] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention,” in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2048–2057.
- [222] Z. Ren, X. Wang, N. Zhang, X. Lv, and L.-J. Li, “Deep reinforcement learning-based image captioning with embedding reward,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 290–298.
- [223] X. Zhang, X. Zhao, Z. Li, J. Xia, R. Jain, and W. Chao, “Social image tagging using graph-based reinforcement on multi-type interrelated objects,” *Signal Process.*, vol. 93, no. 8, pp. 2178–2189, Aug. 2013.
- [224] T. Yao, Y. Pan, Y. Li, and T. Mei, “Exploring visual relationship for image captioning,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 684–699.
- [225] S. Bai and S. An, “A survey on automatic image caption generation,” *Neurocomputing*, vol. 311, pp. 291–304, Oct. 2018.
- [226] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan, “Deep joint rain detection and removal from a single image,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1357–1366.
- [227] Q. Zheng, X. Qiao, Y. Cao, and R. W. H. Lau, “Distraction-aware shadow detection,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 5167–5176.
- [228] V. Allken, N. O. Handegard, S. Rosen, T. Schreyeck, T. Mahiout, and K. Malde, “Fish species identification using a convolutional neural network trained on synthetic data,” *ICES J. Mar. Sci.*, vol. 76, no. 1, pp. 342–349, Jan. 2019.
- [229] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, “Speed/Accuracy trade-offs for modern convolutional object detectors,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 7310–7311.
- [230] Z. Cai and N. Vasconcelos, “Cascade R-CNN: Delving into high quality object detection,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6154–6162.
- [231] B. Cheng, Y. Wei, H. Shi, R. Feris, J. Xiong, and T. Huang, “Decoupled classification refinement: Hard false positive suppression for object detection,” 2018, *arXiv:1810.04002*. [Online]. Available: <http://arxiv.org/abs/1810.04002>
- [232] B. Cheng, Y. Wei, H. Shi, R. Feris, J. Xiong, and T. Huang, “Revisiting RCNN: On awakening the classification power of faster RCNN,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 453–468.
- [233] Z. Cai and N. Vasconcelos, “Cascade R-CNN: High quality object detection and instance segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Nov. 28, 2019, doi: [10.1109/TPAMI.2019.2956516](https://doi.org/10.1109/TPAMI.2019.2956516).
- [234] K. Chen, W. Ouyang, C. C. Loy, D. Lin, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, and J. Shi, “Hybrid task cascade for instance segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 4974–4983.
- [235] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “MobileNets: Efficient convolutional neural networks for mobile vision applications,” 2017, *arXiv:1704.04861*. [Online]. Available: <http://arxiv.org/abs/1704.04861>
- [236] Y. Li, Y. Chen, N. Wang, and Z. Zhang, “Scale-aware trident networks for object detection,” 2019, *arXiv:1901.01892*. [Online]. Available: <http://arxiv.org/abs/1901.01892>
- [237] K. Sun, B. Xiao, D. Liu, and J. Wang, “Deep high-resolution representation learning for human pose estimation,” 2019, *arXiv:1902.09212*. [Online]. Available: <http://arxiv.org/abs/1902.09212>
- [238] K. Sun, Y. Zhao, B. Jiang, T. Cheng, B. Xiao, D. Liu, Y. Mu, X. Wang, W. Liu, and J. Wang, “High-resolution representations for labeling pixels and regions,” 2019, *arXiv:1904.04514*. [Online]. Available: <http://arxiv.org/abs/1904.04514>
- [239] P. F. Felzenswalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
- [240] L. Wan, D. Eigen, and R. Fergus, “End-to-end integration of a convolutional network, deformable parts model and non-maximum suppression,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 851–859.
- [241] J. Hosang, R. Benenson, P. Dollár, and B. Schiele, “What makes for effective detection proposals?” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 4, pp. 814–830, Apr. 2016.
- [242] B. M. Lake, R. Salakhudinov, and J. B. Tenenbaum, “Human-level concept learning through probabilistic program induction,” *Science*, vol. 350, no. 6266, pp. 1332–1338, Dec. 2015.

- [243] B. Hariharan and R. Girshick, "Low-shot visual recognition by shrinking and hallucinating features," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 3018–3027.
- [244] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "Model compression and acceleration for deep neural networks: The principles, progress, and challenges," *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 126–136, Jan. 2018.
- [245] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4107–4115.
- [246] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," 2015, *arXiv:1510.00149*. [Online]. Available: <http://arxiv.org/abs/1510.00149>
- [247] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf, "Pruning filters for efficient ConvNets," 2016, *arXiv:1608.08710*. [Online]. Available: <http://arxiv.org/abs/1608.08710>
- [248] Y. Wei, X. Pan, H. Qin, W. Ouyang, and J. Yan, "Quantization mimic: Towards very tiny CNN for object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 267–283.
- [249] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker, "Learning efficient object detection models with knowledge distillation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 742–751.
- [250] J. M. Alvarez and M. Salzmann, "Learning the number of neurons in deep networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2270–2278.
- [251] G. Huang, S. Liu, L. Van der Maaten, and K. Q. Weinberger, "CondenseNet: An efficient DenseNet using learned group convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2752–2761.
- [252] X. Lin, C. Zhao, and W. Pan, "Towards accurate binary convolutional neural network," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 345–353.
- [253] R. Yu, A. Li, C.-F. Chen, J.-H. Lai, V. I. Morariu, X. Han, M. Gao, C.-Y. Lin, and L. S. Davis, "NISP: Pruning networks using neuron importance score propagation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9194–9203.
- [254] R. Hu, P. Dollár, K. He, T. Darrell, and R. Girshick, "Learning to segment every thing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4233–4241.
- [255] I. Biederman, "Recognition-by-components: A theory of human image understanding," *Psychol. Rev.*, vol. 94, no. 2, p. 115, 1987.
- [256] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, Apr. 2006.
- [257] H. Chen, Y. Wang, G. Wang, and Y. Qiao, "LSTD: A low-shot transfer detector for object detection," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 2836–2843.
- [258] X. Dong, L. Zheng, F. Ma, Y. Yang, and D. Meng, "Few-example object detection with model communication," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1641–1654, Jul. 2019.
- [259] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 1126–1135.
- [260] B. Kang, Z. Liu, X. Wang, F. Yu, J. Feng, and T. Darrell, "Few-shot object detection via feature reweighting," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 8420–8429.
- [261] M. Ren, E. Triantafillou, S. Ravi, J. Snell, K. Swersky, J. B. Tenenbaum, H. Larochelle, and R. S. Zemel, "Meta-learning for semi-supervised few-shot classification," 2018, *arXiv:1803.00676*. [Online]. Available: <http://arxiv.org/abs/1803.00676>
- [262] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, "3D object proposals for accurate object class detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 424–432.
- [263] B. Pepik, R. Benenson, T. Ritschel, and B. Schiele, "What is holding back convnets for detection?" in *Proc. German Conf. Pattern Recognit.* Cham, Switzerland: Springer, Oct. 2015, pp. 517–528.
- [264] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond Pascal: A benchmark for 3D object detection in the wild," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Mar. 2014, pp. 75–82.
- [265] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Detect to track and track to detect," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 3038–3046.
- [266] K. Kang, W. Ouyang, H. Li, and X. Wang, "Object detection from video tubelets with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 817–825.
- [267] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 652–660.
- [268] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum PointNets for 3D object detection from RGB-D data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 918–927.
- [269] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
- [270] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1452–1464, Jul. 2017.
- [271] S. Gidaris and N. Komodakis, "Object detection via a multi-region and semantic segmentation-aware CNN model," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1134–1142.
- [272] S. Hong, B. Roh, K.-H. Kim, Y. Cheon, and M. Park, "PVANet: Lightweight deep neural networks for real-time object detection," 2016, *arXiv:1611.08588*. [Online]. Available: <https://arxiv.org/abs/1611.08588>
- [273] Y. Zhu, C. Zhao, J. Wang, X. Zhao, Y. Wu, and H. Lu, "CoupleNet: Coupling global structure with local parts for object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 4126–4134.
- [274] A. Shrivastava, R. Sukthankar, J. Malik, and A. Gupta, "Beyond skip connections: Top-down modulation for object detection," 2016, *arXiv:1612.06851*. [Online]. Available: <http://arxiv.org/abs/1612.06851>
- [275] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS—Improving object detection with one line of code," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 5561–5569.
- [276] L. Tychsen-Smith and L. Petersson, "Improving object localization with fitness NMS and bounded IoU loss," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6877–6885.
- [277] T. Kong, F. Sun, A. Yao, H. Liu, M. Lu, and Y. Chen, "RON: Reverse connection with objectness prior networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5936–5944.



**LUBNA AZIZ** received the B.S. and M.S. degrees in computer engineering from the Balochistan University of Information Technology, Engineering and Management Sciences (BUITEMS), Quetta, Pakistan, in 2008 and 2017, respectively.

She is currently a Lecturer with BUITEMS. She is also a Ph.D. Scholar with the University of Technology Malaysia. Her research interests include image processing (bio-imaging), machine learning, deep learning, and computer vision (specifically object detection). She received two-time Gold Medal for her academic career.



**MD. SAH BIN HAJI SALAM** (Member, IEEE) received the bachelor of science degree from the University of Pittsburgh, Pittsburgh, PA, USA, and the master's and Ph.D. degrees in computer science with a specialization in speech processing and AI from the University of Technology Malaysia. He is currently serving as the Head of Vicubela and a Senior Lecturer with the Faculty of Computing, UTM.



**USMAN ULLAH SHEIKH** received a Ph.D. degree in image processing and computer vision from the Universiti Teknologi Malaysia, in 2009.

He is a Senior Lecturer at the University Technology Malaysia (Faculty of Electrical Engineering) with a specialization in Computer Vision and machine learning. His research work is mainly on computer vision, machine learning, and embedded system design.



**SARA AYUB** received the M.S. degree from NUST, Pakistan, with a specialization in signal and image processing. She is currently a Ph.D. Scholar with the Faculty of Engineering, UTM. She is also working as an Assistant Professor with BUITEMS.