IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

OBJECT DETECTION: A DETAILED REVIEW STUDY

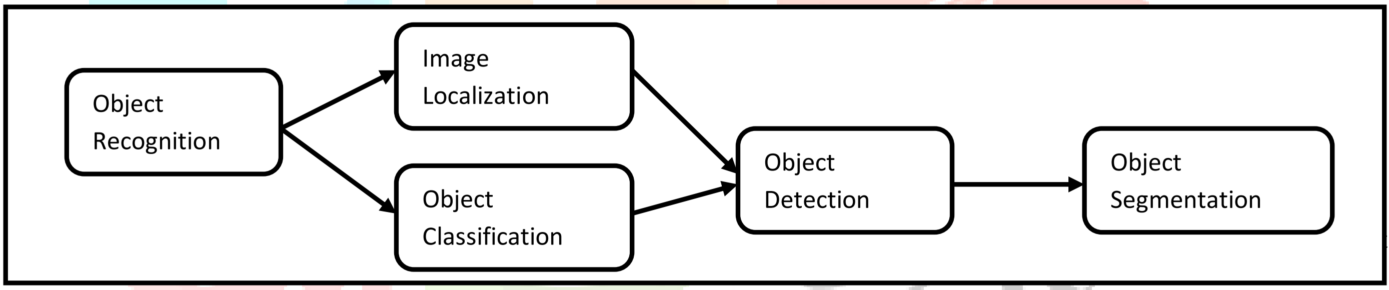
Anjali Jindia
Panjab University, Chandigarh, India

Abstract: There has been rapid development in the research area of deep learning in recent years. Deep learning was used to solve different problems, such as visual recognition, speech recognition and handwriting recognition and has achieved a very good performance. Convolutional Neural Networks (ConvNets or CNNs) are used in deep learning, which are found to give the most accurate results specially in solving object detection problems. In this paper we'll go into summarizing some of the deep learning models used for object detection tasks, performance evaluation metrics for evaluating these models, applications of Object detection, challenges in object detection and its future scope.

Keywords**:** AP, Convolutional Neural Networks, Deep Learning Methods, mAP, Object Detection.

I. INTRODUCTION

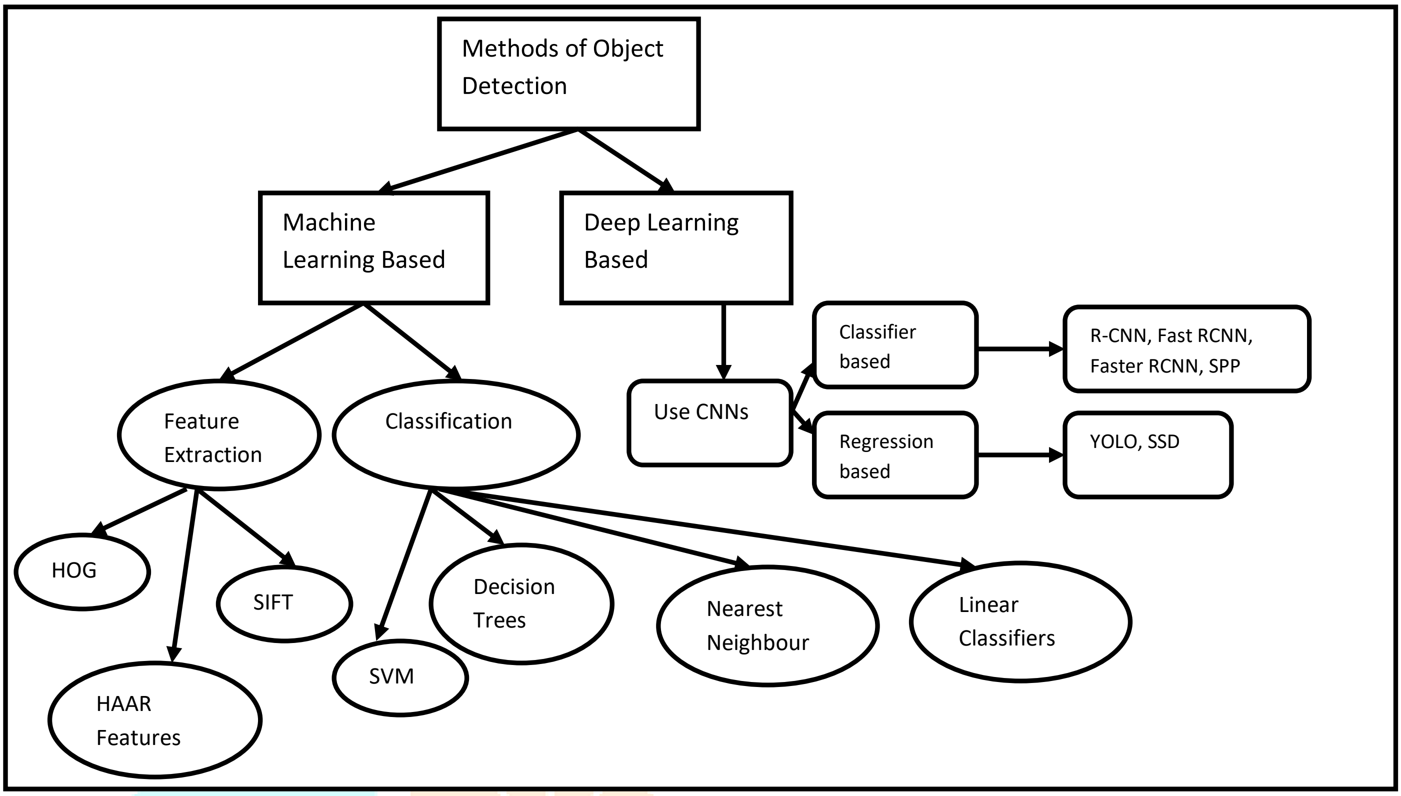
Object detection consists of two tasks: localizing and classifying.
Input: A photograph ie an image with one or more objects.
Output: One or more bounding boxes and a class label for each bounding box.
"*object recognition"* and "*object detection"* are same.



o

II. METHODS OF OBJECT DETECTION

Methods for object detection may be either machine learning-based approaches or deep learning-based approaches. In Machine Learning approaches, firstly the features are defined using one of the methods and then a technique such as support vector machine (SVM) is used to do the classification. In deep learning techniques, end-to-end object detection is done without specifically defining features, and the techniques are typically based on convolutional neural networks (CNNs).



DEEP LEARNING BASED APPROACHES

Different algorithms for object detection exist and they can be split into two groups:

1. Based on classification – These algorithms work in two stages. In the first stage, interesting regions are selected from the image. Then those regions are classified using CNNs. Their speed is very slow because we have to run prediction for every selected region. Region-based convolutional neural network (RCNN) and their cousins Fast-RCNN and Faster-RCNN fall in this category.
2. Based on regression –Classes and bounding boxes for the whole image are predicated **in one run of the algorithm** instead of selecting interesting parts of an image. **YOLO (You only look once), SSD** fall in this category and are commonly used for real-time object detection.

R-CNN (Regions with CNN features)

'Region proposals/regions that *could* belong to a particular object are generated. Sub-segmentations of the image that could belong to one object — based on color, texture, size and shape are then generated using the selective search algorithm and iteratively similar regions are combined to form objects. Thereafter, each region is classified using class-specific linear Support Vector Machines (SVMs). So, RCNN is divided into three steps:

- Selective Search algorithm is used to scan the input image for possible objects. that generates category-independent regional proposals that define the set of candidate detectors available to the model's detector.
- A fixed-length feature vector is the extracted from each region by running a convolutional neural net (CNN) on top of each of these region proposals .
- To classify the region, output of each CNN is fed into an SVM (support vector machines) and to tighten the bounding box of the object (if such an object exists) output of each CNN is fed into a linear regressor.

Fast R-CNN

An image and a set of object proposals are fed as input.

A convolutional feature map is then produced by processing the image with the convolutional and max-pooling layers (to down-sample an input image, reducing its dimensionality by keeping the max value activated features) in the sub-regions binned).

To detect the Region of Interests (RoIs) selective search method is applied on the produced feature maps. To get a fixed-layer feature vector (with valid Region of Interests with fixed height and width as hyperparameters) for each region proposal, the feature maps size is reduced using a RoI pooling layer.

Then the feature vectors are fed to fully connected layers which branch into two output layers, one producing softmax probability estimates over several object classes and the other producing four real-value numbers for each of the object classes (numbers represent the position of the bounding box for each of the objects).

Faster R-CNN

The Faster Region-based Convolutional Network is a combination of RPN and the Fast R-CNN model. The slow selective search algorithm is replaced with a fast neural net by Faster R-CNN. So, it introduced the region proposal network (RPN). It is composed of a feature extraction network followed by two subnetworks.

- A pretrained CNN is used as the feature extraction network.

- The first subnetwork after the feature extraction network is a region proposal network (RPN) which is trained to generate object proposals i.e. the areas in the image where objects are likely to exist.
- Then to predict the actual class of each object proposal, the second subnetwork is trained.

R-FCN

R-FCN stands for Region-based Fully Convolutional Net. Being fully convolutional, it shares 100% of the computations across every single output but suffers from one drawback.

On the one side, we want to learn location invariance in a model (while classifying an object): i.e. regardless of where the object appears in the image, we want to classify it and on the other hand, when detecting the object, we want to learn location variance i.e. if the object is at a particular location, we want to draw a box at that location. So, the problem of compromising between location invariance and location variance arises if we try to share convolutional computations across 100% of the net. R-FCN provides solution i.e. position-sensitive score maps.

One relative position of one object class is represented by each position-sensitive score map. For example, one score map might activate wherever it detects the object. Another score map might activate where it sees the presence of other object of a particular class. To recognize certain parts of each object, these score maps or convolutional feature maps are trained.

Hence, the working of R-FCN can be summed up as follows:

- A CNN is run over the input image
- To generate a score bank of the “position-sensitive score maps”, a fully convolutional layer is added. There should be $k^2(C+1)$ score maps, with k^2 representing the number of relative positions to divide an object (e.g. 2^2 for a 2 by 2 grid) and C+1 representing the number of classes plus the background.
- To generate regions of interest (RoI’s), a fully convolutional region proposal network (RPN) is run.
- Divide each RoI into same k^2 “bins” or subregions as the score maps
- Check the score bank for each bin, to see if that bin matches the corresponding position of some object. For example, if I’m on the “top-left” bin, I will grab the score maps that correspond to the “top-left” corner of an object and average those values in the RoI region. This process is repeated for each class.
- To get a single score per class, average the bins, once each of the k^2 bins has an “object match” value for each class
- RoI is classified with a softmax over the remaining C+1 dimensional vector.

SPP

In Spatial Pyramid Pooling, the entire network structure is called SPP-net. The main aim of Feature extraction process is to extract a large collection of features from each of the region proposals of the image. This came to be major reason for introducing spatial pyramid pooling into the convolutional neural network architecture. SPP has the ability to manage images of different scales, sizes and aspect ratios, which makes it an adaptive technique.

Initially, feature maps of the input images are generated using a number of convolutional layers and the feature maps are generated only once in the SPP-net. These feature maps are allowed to pass through the SPP layer which outputs n number of M-dimensional vectors. Here n is the number of filters available in the final convolutional layer. SPP is faster than RCNN but gives reduced accuracy for very deep neural networks.

YOLO

YOLO considers object detection as a simple regression problem and takes an input image to learn the class probabilities and bounding box coordinates. For both classification and localization tasks YOLO applies a single neural network to the full image, instead of applying the model to an image at multiple locations and scales. The image is divided into regions by the network and bounding boxes and probabilities for each region are predicted. Predicted probabilities are used to weigh these bounding boxes. Finally, high scoring detections are taken by thresholding the detections by some value. This model allows real time object detection.

SSD

SSD stands for Single-Shot Detector. We take only one single shot to detect multiple objects within the image which makes SSD much faster as compared to two-shot RPN-based approaches.

Generating the regions of interest and classifying those regions is done in a “single shot,” in SSD and it simultaneously predicts the bounding box and the class as it processes the image. To an input image and a set of ground truth labels, SSD does the following:

- The image is passed through a series of convolutional layers, yielding several sets of feature maps at different scales (e.g. 10x10, then 6x6, then 3x3, etc.)
- A 3x3 convolutional filter is used for each location in each of these feature maps, to evaluate a small set of default bounding boxes.
- The bounding box offset and the class probabilities are simultaneously predicted for each box.
- The ground truth box are matched with these predicted boxes based on IoU (Intersection over Union), during training. A “positive” label is assigned to the best predicted box along with all other boxes that have an IoU with the truth >0.5.

RETINANET

For small objects, detecting them in different scales is very challenging. It is time consuming and demands high memory to process multiple scale images. So, a pyramid of feature is created and used for object detection. However, feature maps closer to the image layer, are composed of low-level structures that are not effective for accurate object detection.

RetinaNet uses Feature Pyramid Network (FPN) and Focal loss for training. So, it is called a single stage detector.

Feature Pyramid Network (**FPN**) is a feature extractor designed for better accuracy and speed and replaces the feature extractor of detectors like Faster R-CNN and generates multiple feature map layers (**multi-scale feature maps**). It is a structure for multi scale object detection. Feature maps of different scale on multiple levels in the network are produced which helps both classifier and regressor networks. As it's less clumsy than feeding the network the same image at various resolutions, this makes the training faster.

To address the single-stage object detection problems with the imbalance where there is a very large number of possible background classes and just a few foreground classes, the concept of Focal Loss is designed.

III. PERFORMANCE EVALUATION

The main aim of an object detection models is firstly to identify if an object is present in the image and the class of the object (Classification) and secondly to predict the co-ordinates of the bounding box around the object when an object is present in the image (Localization). So, we need to evaluate the performance of both classification as well as localization of using bounding boxes in the image

Measuring the performance of object detection model

- **Intersection over Union (IoU)**

It is also referred to as the Jaccard Index. It quantifies the similarity between the ground truth bounding box and the predicted bounding box to evaluate how good the predicted box is. The IoU score ranges from 0 to 1. An IoU of 1 implies that predicted and the ground-truth bounding boxes perfectly overlap. We can set a threshold value for the IoU to determine if the object detection is valid or not not. All bounding box candidates with an IoU greater than some threshold are usually kept.

- **Predictions: TP - FP - FN**

The predictions are classified into True Positives (TP), False Negatives (FN), and False Positives (FP).

- TP – "true positive" – a number of detected objects that should be detected (i.e. "true" objects are included in the "ground truth".
- FP – "false positives" – a number of detected objects that shouldn't be detected (i.e. not included in the "ground truth").
- FN – "false negatives" – a number of objects that weren't detected despite being included in the "ground truth".

Let's set IoU to 0.6, in that case

- if IoU ≥0.6, classify the object detection as True Positive(TP)
- if Iou <0.6, then it is a wrong detection and classify it as False Positive(FP)
- False Negative(FN) is when a ground truth is present in the image and model failed to detect the object.

Precision and Recall are then computed for each class where predictions of TP, FP and FN are accumulated.

- **Precision**

It means how many of the detected objects are really existing in the analyzed image ("ground truth").

Precision is the the probability of the *predicted* bounding boxes matching actual ground truth boxes, also referred to as the positive predictive value.

Precision= (true object detection) / (all detected boxes)

Precision scores range from 0 to 1, a high precision implies that most detected objects match ground truth objects. E.g. Precision = 0.9, when an object is detected, 90% of the time the detector is correct.

- **Recall**

How many of objects that should be detected (i.e. "true" objects), were detected? Recall measures the probability of *ground truth objects* being correctly detected.

Recall= (true object detection) / (all ground truth boxes)

Recall ranges from 0 to 1 where a high recall score means that most ground truth objects were detected. E.g, recall =0.7, implies that the model detects 70% of the objects correctly.

Interpretations

- ➢ High recall but low precision implies that all ground truth objects have been detected, but most detections are incorrect (many false positives).
- ➢ Low recall but High precision implies that all predicted boxes are correct, but most ground truth objects have been missed (many false negatives).
- ➢ High precision and high recall, the ideal detector has most ground truth objects detected correctly.

- **Average Precision**

It measures the performance of object detectors which is a *single number* metric that encapsulates both precision and recall and summarizes the Precision-Recall curve by averaging precision across recall values from 0 to 1.

- **Mean average precision (mAP)**

If the dataset contains *N* class categories, the mAP averages AP over the *N* classes.

Overall, the mAP calculation is divided into 2 main steps:

1. Calculate **AP** (average precision) per each class of detected objects and configured **IoU threshold**.
2. Calculate **mAP** as a mean from previously calculated AP values.

IV. APPLICATIONS OF OBJECT DETECTION

1. OPTICAL CHARACTER RECOGNITION

Optical Character Recognition, or OCR, is a technology that enables you to convert different types of documents, such as scanned paper documents, PDF files or images captured by a digital camera into editable and searchable data.

2. SELF DRIVING CARS

It is a vehicle that is capable of sensing its environment and moving safely with little or no human input. It is also known as also known as an autonomous vehicle (AV), connected and autonomous vehicle (CAV), driverless car, robo-car, or robotic car.

3. TRACKING OBJECTS

Object tracking is the process of taking an initial set of object detections, creating a unique ID for each of the initial detections and then tracking each of the objects. Object tracking has a variety of uses, some of which are surveillance and security, traffic monitoring, video communication, robot vision and animation.

4. FACE DETECTION AND FACE RECOGNITION

Face detection is a technology that identifies human faces in digital images. Face recognition describes a biometric technology that goes way beyond recognizing when a human face is present.

5. IDENTITY VERIFICATION THROUGH IRIS CODE

Iris recognition is an automated method of biometric identification that uses mathematical pattern-recognition techniques on video images of one or both of the irises of an individual's eyes, whose complex patterns are unique, stable, and can be seen from some distance.

7. SMILE DETECTION

Smile detection is a special task in facial expression analysis with various potential applications such as photo selection, user experience analysis, smiling payment and patient monitoring. Conventional approaches often extract low-level face descriptors and detect smile based on a strong binary classifier.

8. PEDESTRIAN DETECTION

It provides the fundamental information for semantic understanding of the video footages due to which, pedestrian detection is an essential and significant task in any intelligent video surveillance system. Due to the potential for improving safety systems, it has an obvious extension to automotive applications.

9. DIGITAL WATERMARKING

A digital watermark is a pattern of bits inserted into a digital file – image, audio or video. Such messages usually carry copyright information of the file. It may be used for a wide range of applications such as Copyright protection, Source tracking, Broadcast monitoring, Video authentication, Software crippling on screen casting and video editing software programs, ID card security, Fraud and Tamper detection.

10. MEDICAL IMAGING

Medical imaging refers to techniques and processes used to create images of various parts of the human body for diagnostic and treatment purposes within digital health. Medical image processing tools are playing an increasingly important role in assisting the clinicians in diagnosis, therapy planning and image-guided interventions. Accurate, robust and fast tracking of deformable anatomical objects such as the heart, is a crucial task in medical image analysis.

V. CHALLENGES IN OBJECT DETECTION

1. **Dual priorities**: **object classification and localization**

The first main challenge in object detection is that we not only want to classify the image objects but also to determine the objects' positions, generally referred to as the object localization task.

2. **Speed for real-time detection**

To meet the real-time demands of video processing, Object detection algorithms need to be incredibly fast at prediction time along with the need to accurately classify and localize important objects.

3. **Multiple spatial scales and aspect ratios**

Practitioners must leverage several techniques to ensure that the detection algorithms are able to capture objects at multiple scales and views as for many applications of object detection, items of interest may appear in a wide range of sizes and aspect ratios.

4. **Limited data**

Another major hurdle is the limited amount of annotated data currently available for object detection. Gathering ground truth labels along with accurate bounding boxes for object detection, still remains incredibly tedious work.

5. **Class imbalance**

It is one of the main issue for most classification problems including the object detection. For example, consider a photograph containing a few main objects and the remainder of the image is filled with background.

VI. SCOPE OF OBJECT DETECTION

Object detection is a key ability for most computer and robot vision system. Although great progress has been observed in the last years, we are still far from achieving human-level performance. Object detection has not been used much in many areas where it could be of great help. For ex. as mobile robots and autonomous machines, are starting to be more widely deployed (e.g., quad-copters, drones and soon service robots), the need of object detection systems is gaining more importance. We also need to consider the need of object detection systems for nano-robots (robots that will explore areas that have not been seen by humans, such as depth parts of the sea or other planets). In recent years we have observed a tremendous refinement in the domain of object detection especially over remote sensing images. Remote sensing images are high quality images taken from the top of the earth at varying heights, at varying light conditions for variety of applications. The accuracy of the object detection results can be improved by increasing the count of the images. The increased number of objects over these images has increased the complexity of the image background which results in difficulty to analyze these images properly. Thus object detection is however a vital task for remote sensing images. Remote sensing images are of high importance and useful as it is used for environment monitoring and tracking, change detection and so on. Tracking the growth of a city, reduction in agricultural lands, decreasing the area of forest lands, disappearing water resources all these comes under the category of change detection and environment monitoring. These are some exclusive applications of remote sensing images and are research oriented. Monitoring the oil reserves, volcano eruption, military surveillance, monitoring and controlling the forest fire, plotting areas of oasis in desert, weather forecasting all those demand remote sensing images of high resolution in nature.

VII. CONCLUSION

Deep learning based object detection has made a great advent over the past few years in the development as well as the use of advanced object detection techniques. This review provides a short and crisp idea about different object detection techniques and lists the highlights and challenges of each. The work also concentrates on the various applications of object detection. Apart from this, the paper also mention the promising future of object detection in the field of remote sensing images. This survey provides a valuable observation in object detection and promote new research. Object detection is customarily considered to be much harder than image classification, particularly because of these five challenges: dual priorities, speed, multiple scales, limited data, and class imbalance. Researchers have dedicated much effort to overcome these difficulties, yielding oftentimes amazing results; however, significant challenges still persist.

REFERENCES

[1] https://cv-tricks.com/object-detection/faster-r-cnn-yolo-ssd/

[2] https://lilianweng.github.io/lil-log/2018/12/27/object-detection-part-4.html

[3] https://heartbeat.fritz.ai/a-2019-guide-to-object-detection-9509987954c3

[4] https://towardsdatascience.com/deep-learning-for-object-detection-a-comprehensive-review-73930816d8d9

[5] https://en.wikipedia.org/wiki/Object_detection

[6]https://towardsdatascience.com/faster-r-cnn-object-detection-implemented-by-keras-for-custom-data-from- googles-open-images-125f62b9141a

[7] https://www.wandb.com/articles/object-detection-with-retinanet

[8]Jonathan Hui, https://medium.com/@jonathan_hui/understanding-feature-pyramid-networks-for-object-detection-fpn-45b227b9106c

[9] https://medium.com/@14prakash/the-intuition-behind-retinanet-eb636755607d

[10] Nikhil Yadav, Utkarsh Binay, “Comparative Study of Object Detection Algorithms”, International Research Journal of Engineering and Technology (IRJET) Volume: 04 Issue: 11,| Nov -2017, pp. 586-591

[11] https://blog.objectivity.co.uk/comparing-object-detection-models/

[12] https://towardsdatascience.com/evaluating-performance-of-an-object-detection-model-137a349c517b

[13]https://manalelaidouni.github.io/manalelaidouni.github.io/Evaluating-Object-Detection-Models-Guide-to-Performance-Metrics.html

[14] https://medium.com/zylapp/review-of-deep-learning-algorithms-for-object-detection-c1f3d437b852

[15]https://towardsdatascience.com/faster-r-cnn-object-detection-implemented-by-keras-for-custom-data-from-googles-open-images-125f62b9141a

[16] https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8741359

[17] https://www.quora.com/What-are-some-interesting-applications-of-object-detection

[18] Er. Navjot Kaur, Er. Yadwinder Kaur “Object classification Techniques using Machine Learning Model”, International Journal of Computer Trends and Technology (IJCTT) – Volume 18,Number 4 – Dec 2014, ISSN: 2231-5381, pp. 170-174

[19] https://towardsdatascience.com/5-significant-object-detection-challenges-and-solutions-924cb09de9dd

[20] Ajeet Ram Pathak, Manjusha Pandey, Siddharth Rautaray, “Application of Deep Learning for Object Detection”, ScienceDirect, Procedia Computer Science 132 (2018), pp. 1706-1717