EasyChair Preprint

№ 594

An Overview of Deep Learning-Based Object Detection Methods

Yassine Bouafia and Larbi Guezouli

October 29, 2018

An Overview of Deep Learning-Based Object Detection Methods

Abstract— **In recent years, there has been rapid development in the research area of deep learning. Deep learning was used to solve different problems, such as visual recognition, speech recognition and handwriting recognition and was achieved a very good performance. In deep learning, Convolutional Neural Networks (ConvNets or CNNs) are found to give the most accurate results, in solving object detection problems.**

In this paper we'll go into summarizing some of the most important deep learning models used for object detection tasks over this last recent year, since the creation of AlexNet in 2012. Then, we'll make a comparison in terms of speed and accuracy between the most used state-of-the-art methods in object detection.

Keywords— Object Detection, Deep Learning Methods, Convolutional Neural Networks

I. Introduction

Object detection is one of the most active field of research in computer vision, where it involves both object classification, classifying every object in the image and object localization, localizing each object by drawing a bounding box around it. Today with the continuous increase in the use of object detection in several interesting applications such as video surveillance, robotic, self-drive car, etc. it became necessary to develop more accurate and faster systems. Deformable Part Model [1] was the dominant detection framework before the widespread use of Convolutional Neural Networks. Recently Convolutional Neural Networks contributed to a significant increase in the accuracy of object detection and greatly surpassed other classic models such as Viola & Jones framework [2], and Histograms of Oriented Gradient (HoG)[3].

The rest of the paper is organized as follows. Firstly, Section II presents challenges and problems to build an ideal detector. Then, Section III provides a brief history of Convolutional Neural Networks. Next, Sections III presents set of datasets for object recognition. After that, Section IV offer an overview of a set of most important object detection methods during the past few years. Then in section V, we make a comparison between set of methods in terms of accuracy and speed. Finally, section VI concludes the overview.

II. Challenges and Problems

An ideal detector should have:

A. High accuracy in localization and recognition:

The detection must be able to locate and accurately recognize objects in images.

B. High efficiency in time and memory:

The detection task should run at a sufficiently high frame rate with acceptable memory and storage usage.

For accuracy, we have two main challenges:

- Firstly, intra-class variations, where each object category can have many different object instances. These instances varying in several features like color, texture, size, shape and different poses in case of non-rigid classes. The variations are caused by changes in a set of factors such as locations, weather conditions, cameras, backgrounds, illuminations, viewpoints, and distance. Further challenges can be added such as illumination, pose, scale, occlusion, background clutter, shading, blur , motion , noise corruption and poor resolution.
- In addition to intra-class variations, we have huge number of object categories in real world, where the number of object categories in existing benchmark datasets is much smaller than that can be recognized by humans.

For efficiency, the challenge is the need to detect objects in real time. This often requires big performance or sacrificing accuracy versus speed. On the other hand, we need to build an efficient detector that work in devices that have limited computational capabilities and storage space such as mobiles.

III. History of Convolutional Neural Networks

Convolutional Neural Networks is a deep learning architecture that have proven very effective in computer vision tasks. CNN was inspired from the cat's visual cortex. In 1962, Hubel and Wiesel's [4], found that cells in animal visual cortex are responsible for detecting light in receptive fields. Inspired by this discovery, Kunihiko Fukushima proposed a hierarchical model called Neocognitron[5]. Then, the first CNN was proposed by Hecht-Nielsenand LeCun et al., after many previous successful iterations since the year 1988, they developed a multi-layer artificial neural network trained with the backpropagation algorithm [6] called LeNet-5 [7] and it was used to classify handwritten digits. After this period the search in Deep Learning has entered a dark time. The next step for deep learning took place in 1999 owing to GPUs that make computers faster. Another big step was in 2009 when professor Fei-Fei Li launched ImageNet[10], a free data base of more than 14 million labeled images. With a large amount of data and the advent of GPUs, the field of CNN has gone through a renaissance phase. Several publications have established more efficient ways to train convolutional neural networks using GPU computing. In 2012 Krizhevsky, Ilya Sutskever, and Geoffrey Hinton won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) with deep CNN model called AlexNet, which was the beginning of a modern history of object detection.

IV. Datasets for object detection

Datasets play a very important role in object detection research, they have been one of the most important factors for the progress in the field, unfortunately data is harder and more expensive to generate. Over the last decade, a number of datasets have been made public to evaluate object detection algorithms. These datasets are collected from different scenarios and can therefore be used as a reference for applications. Below in Table I, there are a set of the popular datasets for object recognition.

TABLE I. OBJECT DETECTION DATASETS

Dataset	Total Images	Categories	Image Size	Started Year
MNIST	60,000	10	28x28	1998
ImageNet	<14 Millions	21841	500x400	2009
Caltech101	9,145	101	300x200	2004
Caltech256	30,607	256	300x200	2007
MS COCO	<328,000	91	640x480	2014
PASCAL VOC(2012)	11,540	20	470x380	2005
CIFAR-10	60,000	10	32x32	2009
Scenes15	4,485	15	256x256	2006
Tiny images	<79 Millions	53,464	32x32	2006
SUN	131,072	908	500x300	2010
Open Images	<9 Millions	<6000	varied	2017

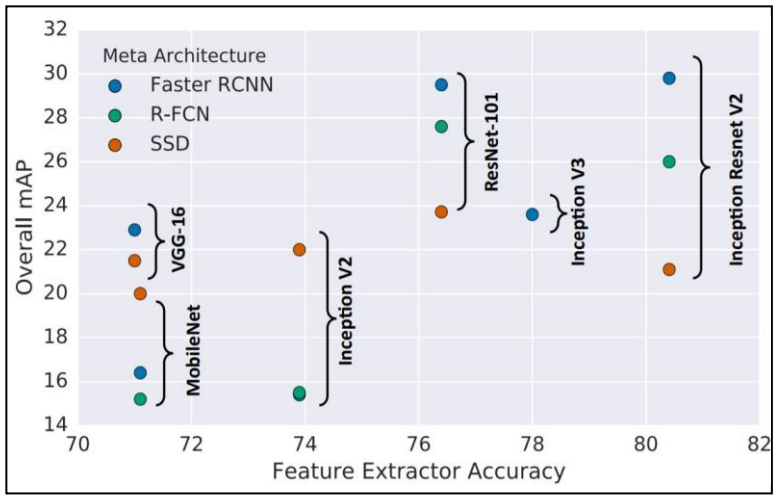

FIG. I. Accuracy of detector (mAP on COCO) vs accuracy of feature extractor (as measured by top-1 accuracy on ImageNet-CLS). [8]

V. Object detection methods based on deep learning

Currently we can organize object detectors in two main categories Fig. II:

Two-stage detectors: Such as Faster R-CNN that divides the detection process in two steps. The first step uses a Region Proposal Network to generate regions of interests that have a high probability of being an object. The second step then performs the final classification and bounding-box regression of objects by taking these regions as input. These two steps are named the Region Proposal Step and the Object Detection Step respectively. The dominant paradigm in modern object detection is based on a two-stage approach. Such models reach the highest accuracy rates, but are typically slow.

One-stage detectors: Such as YOLO and SSD, that treat object detection as a simple regression problem by taking an input image and learning the class probabilities and bounding box coordinates. The approach is simple and elegant because it completely eliminates region proposal generation, encapsulating all computation in a single network. Such models reach lower accuracy rates, but are much faster than two-stage object detectors and shown higher memory efficiency.

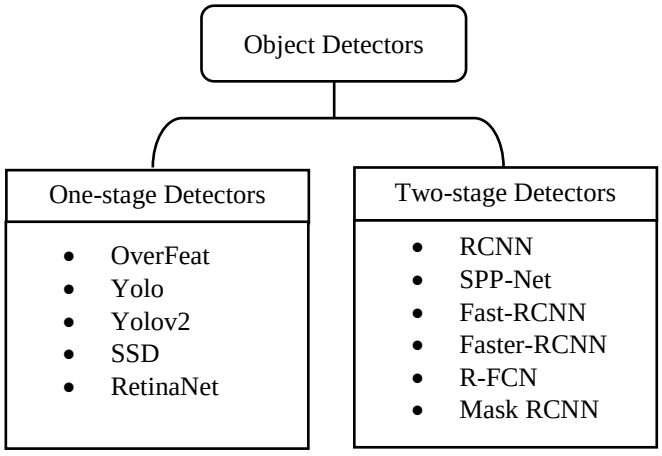

FIG. II. Main categories of object detectors

In this section we will show some of the most prominent detectors in recent years, as listed in FIG.III:

AlexNet [9]**:** is CNN for image classification created by A. Krizhevsky, I. Sutskever, and G. Hinton that was won the ILSVR 2012[10] competition and achieved winning top-5 test error rate of 15.3%, compared to 26.2% achieved by the second-best entry. Alexnet architecture consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax, also integrated various regularization techniques, such as data augmentation, dropout and used ReLU for the nonlinearity functions to decrease training time.

Overfeat [11]**:** Is a sliding window approach that can be uses for classification, localization and detection. Overfeat using Convolutional Networks that contains eight layers. The first five are convolutional layers and the remaining three are fully-connected. The output of the last fully-connected layer is fed to a 1000-way softmax. In the ILSVRC 2013 dataset, OverFeat ranked 4th in classification with 14.2% error, 1st in localization with 29.9% error (top 5 error rate) and 1st in detection established a new state of the art with 24.3% mean Average Precision (mAP).

R-CNN [12]**:** Quickly after OverFeat, Regions with CNN features or R-CNN from Ross Girshick, et al. achieves a mean average precision (mAP) of 53.7% on PASCAL VOC 2010, and 31.4% mAP on the ILSVRC2013 detection dataset, where it is considered a large improvement over OverFeat. R-CNN takes an input image, extracts around 2000 bottom-up region proposals using Selective Search [13] algorithm, computes features for each proposal using a large CNN and then classifies each region using class-specific linear SVMs.

ZFNet [14]**:** Was the winner of the ILSVRC 2013 competition with 11.2% error rate. This network built by Matthew Zeiler and Rob Fergus from NYU have very similar architecture to AlexNet with a minor modification (use 7x7 kernel instead of 11x11 to retain more information). ZFNet developed a new visualization technique named Deconvolutional Network (deconvnet), which helps to examine different feature activations and their relation to the input space.

SPPNet [15]**:** Spatial Pyramid Pooling Net is essentially an enhanced version of R-CNN by introducing two important concepts: adaptively-sized pooling. It uses spatial pooling after the last convolutional layer as opposed to traditionally used max-pooling, and computing feature volume only once. SPPNet ranked 3^{ed} among all 38 teams attending ILSVRC 2014 [16] with 8.06% error rate.

VGGNet [17]**:** Simonyan and Zisserman of the University of Oxford created a 19 layer CNN that strictly used 3x3 filters with stride and pad of 1, along with 2x2 maxpooling layers with stride 2. Although rank 2 in ILSVRC 2014 which achieved 7.32% it is currently the most preferred choice in the community for extracting features from images. The weight configuration of the VGGNet is publicly available and has been used in many other applications and challenges as a baseline feature extractor. VGGNet increased the depth of the network by adding more convolutional layers and taking advantage of very small convolutional filters in all layers. It was demonstrated that the representation depth is beneficial for the classification accuracy.

GoogleNet(Inception) [18]***:*** Is the winner of ILSVRC 2014 with 6.7% top 5 error rate. Their architecture consisted of 22 layers deep when counting only layers with parameters (or 27 layers if we also count pooling). Instead of traditionally stacking up conv and maxpooling layer sequentially, it stacks up Inception modules, which consists of multiple parallel conv and maxpooling layers with different kernel sizes. It uses 1x1 conv layer to reduce the depth of feature volume output.

Fast R-CNN [19]***:*** Similar to R-CNN, it used Selective Search to generate object proposals, but instead of extracting all of them independently and using SVM classifiers, it applied several convolutional and max pooling layers on the complete image to produce a conv feature map. For each object proposal a region of interest (RoI) pooling layer extracts a fixed-length feature vector from the feature map. Each feature vector is fed into a sequence of fully connected (fc) layers that finally branch into two sibling output layers one that produces softmax probability and another layer that outputs four real-valued numbers for encodes refined bounding-box positions. Fast R-CNN achieved top accuracy on PASCAL VOC 2012 [20] with a mAP of 66%.

Faster R-CNN [21]**:** Slowest part in Fast R-CNN was Selective Search or Edge boxes [22]. Faster R-CNN replaces selective search by a very small convolutional network called Region Proposal Network (RPN) after the last convolutional layer to generate regions of Interests. From that stage, the same pipeline as R-CNN is used region of interest (RoI) pooling, fully connected (FC), and then classification and regression heads. Faster R-CNN introduces the idea of anchor boxes [21] to handle the variations in aspect ratio and scale of objects. Faster R-CNN achieves state-of-the-art object detection accuracy on PASCAL VOC 2007 (73.2% mAP) and 2012 (70.4% mAP) using 300 proposals per image.

YOLO [23]**:** YOLO (You Only Look Once) look at the complete image at once as opposed to looking at only a generated region proposals in the previous methods. It uses a single convolutional neural network (24 conv layers followed y 2 FC layers) for both classification and localization tasks. YOLO frame detection as a regression problem. It divides the image into an S×S grid and for each grid cell predicts B bounding boxes, confidence for those boxes, and C class probabilities. This model allowing real time object detection (45 frames per second) and achieves a mAP of 63.4% on the VOC 2007 test set.

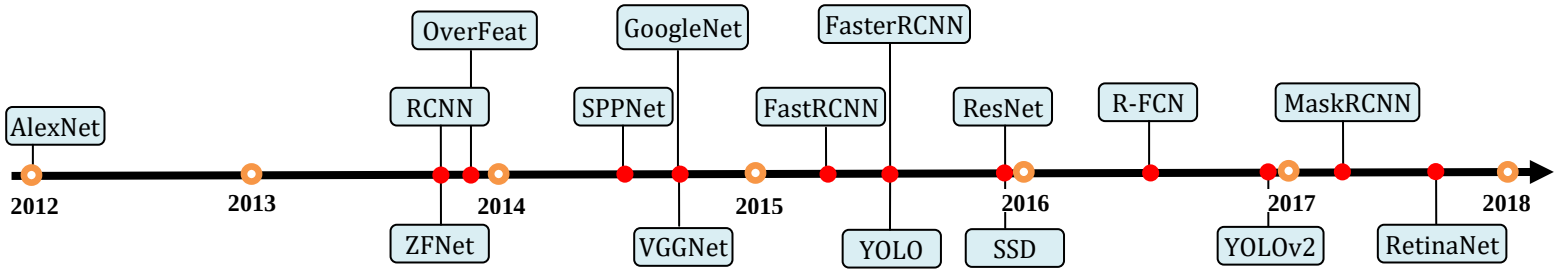

FIG. III. Chronology of object detectors based on the point in time of the first arXiv version

Fast YOLO (Tiny YOLO) [23]**:** is smaller version of YOLO. It is much faster (runs at more than 155 fps) but less accurate than the normal YOLO model (57.1%). Fast YOLO uses a neural network with fewer convolutional layers (9 instead of 24) and fewer filters in those layers. Other than the size of the network, all training and testing parameters are the same between YOLO and Fast YOLO.

ResNet [24]**:** Residual Neural Network won the ILSVRC 2015 competition with an unbelievable 3.6% error rate (human performance is 5-10%). ResNet is a new 152 layers network architecture with "skip connections" and features heavy batch normalization. In this technique they were able to train a very deep neural network with 152 layers. Instead of transforming the input representation to output representation, ResNet sequentially stacks residual blocks, each computes the change it wants to make to its input, and add that to its input to produce its output representation. This is slightly related to boosting.

SSD [25]**:** Like YOLO, SSD (Single Shot Detector) is a method for detecting objects in images using a single deep neural network for both tasks of object localization and classification. It was released by C. Szegedy et al. at the end of November 2016 and reached new records in terms of performance and precision for object detection tasks, scoring over 74% mAP at 59 frames per second on standard datasets such as PascalVOC and COCO.

R-FCN [26]**:**Is a region-based, fully convolutional network for accurate and efficient object detection. In Faster RCNN after the RPN stage, each region proposal had to be cropped out and resized from the feature map and then fed into the Fast RCNN network. This step is the most time consuming .The R-FCN is an attempt to make the the network faster by making it fully convolutional and delaying this cropping step, the idea is increase speed by maximizing shared computation. As result R-FCN show competitive results on the PASCAL VOC 2007 datasets with 83.6% mAP. Meanwhile, is achieved at a test-time speed of 170ms per image, which is faster than Faster R-CNN.

YOLOv2 [27]**:** After various improvements to the YOLO detection method, we have YOLOv2 state-of-the-art on standard detection tasks like PASCAL VOC and COCO. YOLOv2 offered an easy trade-off between speed and accuracy. At 67 FPS, YOLOv2 gets 76.8 mAP on VOC 2007. At 40 FPS, YOLOv2 gets 78.6 mAP. YOLOv2 is real-time object detection system that can detect over 9000 object categories.

Mask R-CNN [28]**:** Running at 5 fps, it was built by the Facebook AI research team (FAIR) in April 2017 this approach efficiently detects objects in an image while simultaneously generating a high-quality segmentation mask for each instance. Mask RCNN extends Faster RCNN by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition. Mask RCNN consists of two stages. The first stage, proposes candidate object bounding boxes where there might be an object. Second, it predicts the class of the object, refines the bounding box and generates a mask in pixel level of the object based on the first stage proposal.

RetinaNet [29]**:** The Facebook AI research team (FAIR) design and train a simple dense detector called RetinaNet is a one-stage object detector, which has the performance of two-stage detectors. The main contribution of this detector is a new loss function called Focal loss, which significantly increased the accuracy. RetinaNet is essentially a Feature Pyramid Network with the cross-entropy loss replaced by Focal loss. The results show that when trained RetinaNet with the focal loss, we have for the first time one stage object detector that is able to match the speed of previous one-stage detectors while matches the state-of-the-art COCO AP of more complex two-stage detectors, such as the Feature Pyramid Network (FPN) or Mask R-CNN .

VI. Comparison

In this part we make a comparison between the different detectors results in terms of both accuracy and speed represented by mean average precision (mAP) and Frame Per Second (FPS) respectively, for this purpose we plot them together to get a full picture of variation in performance between the different detectors.

Table II show results on Pascal VOC 2007 The comparison of these methods as shown in Fig. IV.We note through the Fig. IV. an affinity at the accuracy level between deferent methods with a slight superiority of R-FCN by 80.5% mAP come after him YOLOv2 544 (544 for 544×544 input size) by 78.6% mAP. On the other hand, we notice the large difference in speed between the various methods. Tiny YOLO outperformed all other methods in terms of speed by 155 FPS.
We also notice that YOLOv2 and SSD300 make a good compromise between speed and accuracy.

For the last couple years, many results are exclusively measured with the COCO object detection dataset. COCO dataset is harder for object detection and usually detectors achieve much lower mAP. Table III show results on COCO dataset The comparison of these methods as shown in Fig. V.
We note through the Fig. V. RetinaNet-100-800 achieved the best result in accuracy by 37.8 mAP followed by Faster RCNN-ResNet (use ResNet as backbone) wich achieved 34.9 mAP.YOLOv2 achieve the best performance in speed by 21.6 FPS.

Larger input size leads to better results in accuracy but it is the opposite of speed. The possibility of run a detector at different resolutions allowed an easy trade-off between speed and accuracy. We would also like to emphasize here that the choice of the feature extractors uses to build your detector impacts detection accuracy as shown in Fig. I.

TABLE II. PASCAL VOC 2007 DATASET RESULTS

Method	*mAP*	*FPS*
Tiny YOLO	52,7	155
YOLO	63,4	45
YOLO v2 288	69	91
YOLO v2 544	78,6	40
Fast R-CNN	70	0.5
Faster R-CNN	73,2	7
SDD 300	74,3	58
SDD 512	76,8	23
R-FCN	80.5	6

TABLE III. COCO DATASET RESULTS

Method	*mAP*	*FPS*
YOLOv2	21,6	40
SDD321	28	16
R-FCN	29,9	12
SDD513	31,2	8
RetinaNet-50-500	32,5	14
RetinaNet-100-800	37,8	5
Faster RCNN	21,9	/
Faster RCNN(ResNet)	34,9	/

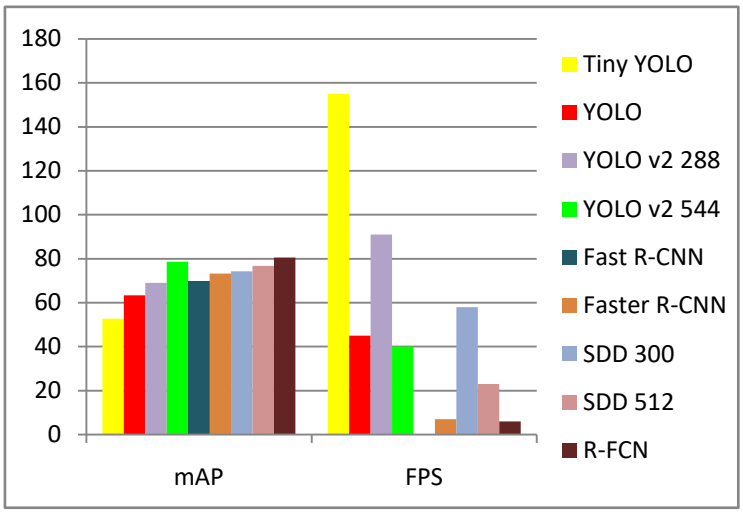

FIG. IV. COMPARISON OF RESULTS ACHIEVED IN PASCAL VOC 2007

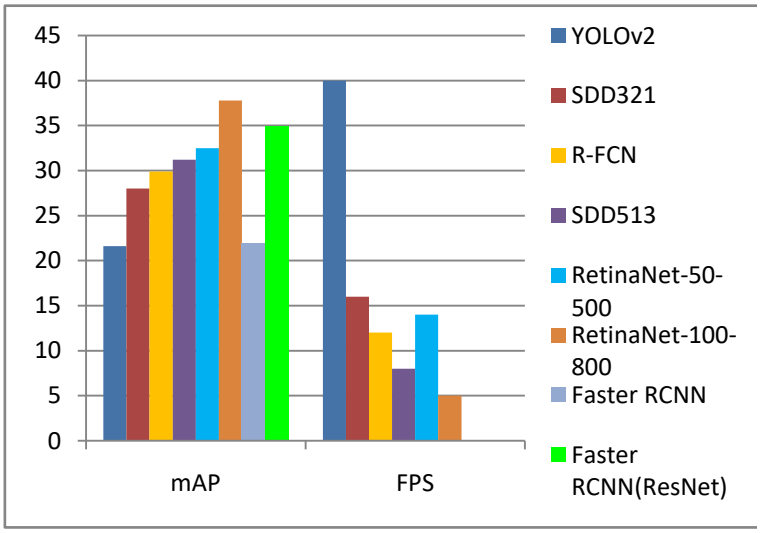

FIG. V. COMPARISON OF RESULTS ACHIEVED IN COCO

VII. Conclusion

In this paper we presented an overview of object detection methods based on deep learning. We started by a brief history of Convolutional Neural Networks and reviewed most important object detection method that used CNN architecture. We selected most used state of the art methods to compare them on their performances.

Choice of a right object detection method is crucial and depends on the problem you are trying to solve and the set-up. Object Detection is the backbone of many practical applications of computer vision such as autonomous cars, security and surveillance, and many industrial applications. Hopefully, this post gave you an intuition and understanding behind each of the popular algorithms for object detection.

References

[1] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester. "Discriminatively trained deformable part models", release 5, 2012.

[2] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 2001, vol. 1, pp. I-511-I–518.

[3] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection", IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005, vol. 1, pp. 886–893.

[4] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex", The Journal of Physiology, 160(1):106, 1962.

[5] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position, In Biological cybernetics, 36(4):193–202, 1980.

[6] Hecht-Nielsen, Robert. "Theory of the backpropagation neural network," IJCNN, in International Joint Conference on Neural Networks. IEEE, 1989.

[7] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. "Backpropagation applied to handwritten zip code recognition", Neural computation, 1989.

[8] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara,Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, Kevin Murphy," Speed/accuracy trade-offs for modern convolutional object detectors".in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA ,pp. 3296-3305,2017

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks", In Communications of the ACM, vol. 60, no. 6, pp. 84–90, May 2017.

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, Li Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge", In International Journal of Computer Vision, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[11] P. S. D. Eigen, X. Z. M. Mathieu, and R. F. Y. LeCun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks", In International Conference on Learning Representations (ICLR), 2013,pp. 16.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation",

In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.

[13] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A.W. Smeulders. "Selective search for object recognition", In International Journal of Computer Vision (IJCV), 2013, pp.157-171.

[14] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks", In Computer Vision – ECCV 2014, pages 818–833. Springer, 2014.

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 9, pp. 1904-1916, Sept. 1 2015.

[16] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh,S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein et al., "Imagenet large scale visual recognition challenge", in International Journal of Computer Vision, 2015, 115(3), pp 211–252

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", in arXiv:1409.1556, p. 14, 2014.

[18] C. Szegedy et al., "Going deeper with convolutions", , in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA,2015, pp. 1–9.

[19] R. B. Girshick. "Fast R-CNN", in CoRR, abs/1504.08083, 2015.

[20] M. Everingham, L. Van Gool, C. K. I.Williams, J.Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) Challenge", In International Journal of Computer Vision (IJCV), 2010, 88(2), pp 303–338.

[21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[22] C. L. Zitnick and P. Dollár." Edge boxes: Locating object proposals from edges", In European Conference on Computer Vision (ECCV), 2014, pp. 391-405.

[23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 779-788.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition", in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 770-778.

[25] W. Liu et al., "SSD: Single Shot MultiBox Detector", in European Conference on Computer Vision, 2016: Computer Vision – ECCV 2016 pp 21-37.

[26] Jifeng Dai, Yi Li, Kaiming He, Jian Sun, "RFCN: object detection via regionbased fully convolutional networks". In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016

[27] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger", in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 6517-6525.

[28] Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick, "Mask RCNN". In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018

[29] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, Piotr Dollár, "Focal loss for dense object detection". In: : IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.