Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 132 (2018) 1706–1717

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Computational Intelligence and Data Science (ICCIDS 2018)

Application of Deep Learning for Object Detection

Ajeet Ram Pathak[a,]*, Manjusha Pandey[a], Siddharth Rautaray[a]

[a]*School of Computer Engineering, Kalinga Institute of Industrial Technology (KIIT) University, Bhubaneswar, India*

Abstract

The ubiquitous and wide applications like scene understanding, video surveillance, robotics, and self-driving systems triggered vast research in the domain of computer vision in the most recent decade. Being the core of all these applications, visual recognition systems which encompasses image classification, localization and detection have achieved great research momentum. Due to significant development in neural networks especially deep learning, these visual recognition systems have attained remarkable performance. Object detection is one of these domains witnessing great success in computer vision. This paper demystifies the role of deep learning techniques based on convolutional neural network for object detection. Deep learning frameworks and services available for object detection are also enunciated. Deep learning techniques for state-of-the-art object detection systems are assessed in this paper.

Keywords: Object detection; Computer vision; Deep learning; Convolutional neural network

* Corresponding author.
E-mail address: ajeet.pathak44@gmail.com

10.1016/j.procs.2018.05.144

1. Introduction

Gartner's 2018 technology trend states that Artificial Intelligence would be widely used trend among the industries and so the Computer vision! [1]. Industries based on automation, consumer markets, medical domains, defense and surveillance sectors are most likely domains extensively using computer vision. It is forecasted that CV market would reach $33.3 billion in 2019 fostering the remarkable growth in the domains of consumer, robotics, and machine vision.

Deep learning technology has become a buzzword nowadays due to the state-of-the-art results obtained in the domain of image classification, object detection, natural language processing. The reasons behind popularity of deep learning are two folded, viz. large availability of datasets and powerful Graphics Processing Units. As deep learning requires large datasets and powerful resources to perform training, both requirements have already been satisfied in this current era. Fig. 1 shows upsurge of Deep Learning with respect to Computer Vision in the recent lustrum.

Image classification, being the widely researched area in the domain of computer vision has achieved remarkable results in world-wide competitions such as ILSVRC, PASCAL VOC, and Microsoft COCO with the help of deep learning [2]. Motivated by the results of image classification, deep learning models have been developed for object detection and deep learning based object detection has also achieved state-of-the-results [3].

We aim to assess deep learning techniques based on convolutional neural network (CNN) for object detection. The beauty of convolutional neural networks is that they do not rely on manually created feature extractors or filters. Rather, they train *per se* from raw pixel level up to final object categories.

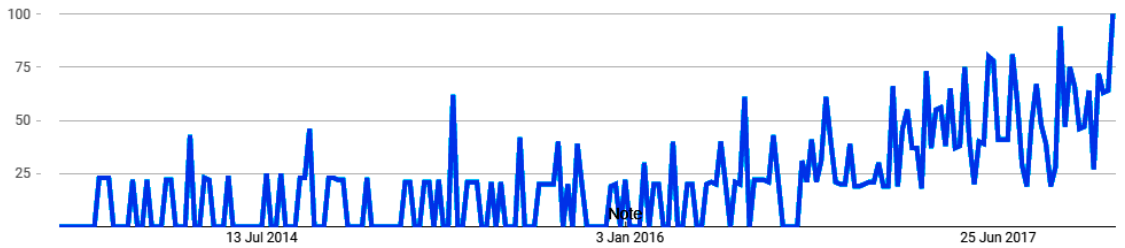

Fig. 1. Upsurge of deep learning for computer vision over the recent lustrum from March 2013 to January 2018 (Created by Google Trends)

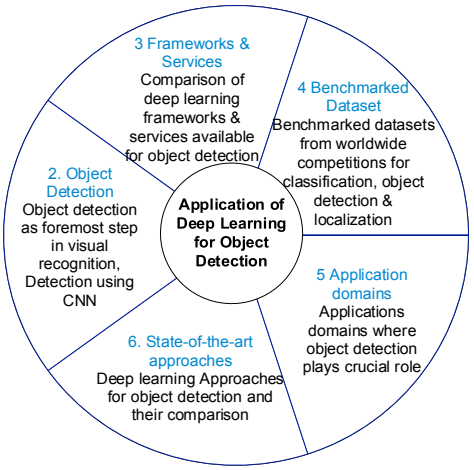

Fig. 2. Organization of the paper

Deep neural architectures handles complex models efficiently than shallow networks. CNNs are less accurate for smaller data but show significant/ record breaking accuracy on the large image datasets. But, CNNs require large amount of labeled datasets to perform computer vision related tasks (recognition, classification and detection).

The contents of the paper are portrayed as follows. Fig. 2 depicts the roadmap of the paper. Section 2 deals with object detection. Section 3 and 4 discusses frameworks and datasets of object detection. Application domains and state-of-the-art approaches are enunciated in section 5 and 6 respectively. Paper is concluded in section 7.

2. Object Detection

2.1. Object detection as foremost step in visual recognition activity

Object detection is the procedure of determining the instance of the class to which the object belongs and estimating the location of the object by outputting the bounding box around the object. Detecting single instance of class from image is called as single class object detection, whereas detecting the classes of all objects present in the image is known as multi class object detection. Different challenges such as partial/full occlusion, varying illumination conditions, poses, scale, etc are needed to be handled while performing the object detection. As shown in the figure 3, object detection is the foremost step in any visual recognition activity.

2.2. Object detection using CNN

Deep CNNs have been extensively used for object detection. CNN is a type of feed-forward neural network and works on principle of weight sharing. Convolution is an integration showing how one function overlaps with other function and is a blend of two functions being multiplied. Fig. 4 shows layered architecture of CNN for object detection. Image is convolved with activation function to get feature maps. To reduce spatial complexity of the network, feature maps are treated with pooling layers to get abstracted feature maps. This process is repeated for the desired number of filters and accordingly feature maps are created. Eventually, these feature maps are processed with fully connected layers to get output of image recognition showing confidence score for the predicted class labels. For ameliorating the complexity of the network and reduce the number of parameters, CNN employs different kinds of pooling layers as shown in the table 1. Pooling layers are translation-invariant. Activation maps are fed as input to the pooling layers. They operate on each patch in the selected map.

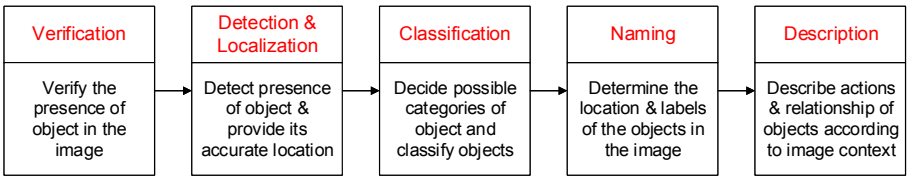

Fig. 3. Object detection as foremost step in visual recognition activity

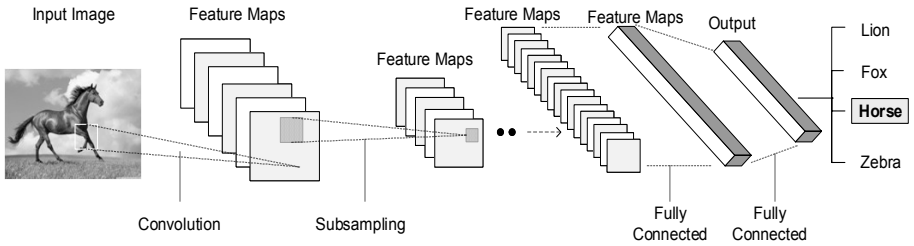

Fig. 4. Use of Convolutional neural network for object detection

Table 1. Pooling layers used for object detection.

Pooling layer	Description
Max pooling	It is widely used pooling in CNNs. It takes maximum value from the selected image patch and place in the matrix storing the maximum values from other image patches.
Average pooling	This pooling averages the neighborhood pixels.
Deformation pooling [40]	Deformable pooling has ability to extract deformable properties, geometric constraints of the objects.
Spatial pyramid pooling [53]	This pooling performs down-sampling of the image and produces feature vector with a fixed length. This feature vector can be used for object detection without making any deformations on the original image. This pooling is robust to object deformations.
Scale dependent pooling [54]	This pooling handles scale variation in object detection and helps to improve the accuracy of detection.

3. Frameworks and Services of Object Detection

The list of deep learning frameworks available till date is exhaustive. We have mentioned some significant deep learning frameworks in table 2. The frameworks are studied from the point of view of features exhibited, interface, support for deep learning model viz. convolutional neural network, recurrent neural network (RNN), Restricted Botltzmann Machine (RBM) and Deep Belief Network (DBN) and support for Multi-node parallel execution, developer of the framework and license. Table 3 show the list of services which can be used for object detection. These services can be availed through the APIs mentioned in the table.

4. Benchmarked Datasets of Object Detection

Datasets provide a way to train and verify the computer vision algorithms and therefore play crucial role for driving the research.

- Microsoft COCO [21]: Microsoft Common Objects in Context dataset encompasses 91 categories of objects and 328000 images. Out of these images, 2.5 million images are labelled. Microsoft COCO dataset exhibits various features like object segmentation, recognition in context, multiple objects per image and 5 captions per image.
- ImageNet [22]: ImageNet dataset is based on WordNet hierarchy (lexicon database for English). Meaningful term in WordNet is known as “synset”. ImageNet provides 1000 images to define each synset. As claimed by creators of ImageNet, ImageNet would offer tens of millions of images for concepts in the WordNet hierarchy. It contains more than 14,197,122 images. To detect local features, images and synsets in ImageNet exhibits Scale-invariant feature transform (SIFT) features. The general resolution of images is around 480×410.
- 80 million tiny image dataset [23] consists of more than 79 million coloured images with 32×32 resolution. Each image is associated with text approximately showing the label and a link to the original image.
- CIFAR-10 and CIFAR-100 dataset [24]: The CIFAR 10 and CIFAR 100 datasets are the subsets of the 80 million tiny images dataset with labelled annotations. The CIFAR-10 dataset possesses 10 object categories with 6000 images per object category and has 60000 coloured images with 32×32 resolution. It consists of 50,000 training and 10,000 test images. The CIFAR-100 dataset has 100 object categories with 600 images per category with 500 training and 100 test images. 100 classes of objects are clustered into 20 super classes. Images are labelled according to the class it belongs i.e. images with “fine” label and “coarse” label (image belonging to super class).
- CUB-200-2011 [25]: Caltech-UCSD Birds-200-2011 is an updated version of CUB-200 dataset [26] having 200 categories of bird species. It consists of 11,788 images with single bounding box per image with 15 part locations and 312 binary attributes.
- Caltech-256 [27]: Caltech-256 dataset consists of 256 object categories having total of 30607 images with 80 images per category of object. This dataset is not recommended for object localization task.
- ILSVRC [28]: Similar to PASCAL VOC Challenges, ImageNet Large Scale Visual Recognition Challenge (ILSVRC) is conducted every year from 2010 till date for classification and detection of objects from millions of images with very large set of object categories. The object categories in ILSVRC are 10 times more than that in PASCAL VOC. PASCAL VOC has 20 object categories whereas ILSVRC has 200 object categories.

Table 2. Deep Learning Frameworks.

Name	Features	Interface	Deep learning model			Multi-node parallel execution	Developer	License
			CNN	RNN	DBN/ RBM			
Caffe [4]	Speed, modular structure, plaintext schema for modelling and optimization, Data storage and communication using blob	C, C++, command line interface, Python, and MATLAB	√	√		Yes	Berkeley Vision & Learning Center	BSD License
Microsoft Cognitive Toolkit CNTK [5]	Multi-dimensional dense data handling, automatic hyperparameter tuning, batch normalization	Python, C++, C#, and command line interface	√	√		Yes	Microsoft Research	MIT License
TensorFlow [6]	Math computations using data flow graph, inception, image classification, auto-differentiation, portability	C++, Python, Java, Go	√	√	√	Yes	Google Brain team	Apache 2.0
Theano [7]	Compilation of math expressions using multi-dimensional arrays	Python	√	√	√	Yes	Université de Montréal	BSD License
Torch [8]	N-dimensional array support, automatic gradient differentiation, support for neural models and energy models	C, C++, Lua	√	√	√	Yes	R. Collobert, K. Kavukcuoglu, C. Farabet	BSD License
Chainer [9]	Define-by-Run Scheme for defining network, multi-GPU parallelization, Forward/Backward Computation, Per-batch architecture	Python	√	√		Yes	Preferred Networks Inc.	MIT License
Keras [10]	Fast prototyping, modular, minimalistic modules, extensible, arbitrary connection schemes	Python	√	√	√	Yes	F. Chollet	MIT License
Deeplearning4j [11]	Distributed deep learning framework, micro-service architecture	Python, Java, Scala	√	√	√	Yes	Skymind engineering team	Apache 2.0
Apache Singa [12]	Scalable distributed training platform	Python, Java, C++	√	√	√	Yes	Apache Incubator	Apache 2.0
MXnet [13]	Blend of symbolic programming and imperative programming, portability, auto-differentiation	Python, R, C++, Julia	√	√		Yes	Distributed (Deep) Machine Learning Community	Apache 2.0
Neon [14]	Fastest performance on various hardware & deep networks (GoogLeNet, VGG, AlexNet, Generative Adversarial Networks)	Python	√	√	√	Yes	Intel	Apache 2.0

Table 3. Services available for object detection

Name	Service	Features	Access
Clarifai [15]	Image and Video Recognition Service	Image and video tagging, Model customization, visual similarity based image search, multi-language support, scalable processing of images and videos, Custom model (pre-trained model) for specific categories (like wedding, travel, food, face recognition, colour, Not Safe For Work model)	Client library to access API, HTTPS
Google Cloud Vision API [16]	Image Analysis Service and API	Label Detection, Explicit Content Detection, Facial Detection, Landmark Detection, Logo Detection, Image sentiment analysis, multi-language support	Integrated REST API
Microsoft Cognitive Service [17]	Computer Vision API	Recognition of face, speech and vision, detection of emotions and video, Motion tracking, speech and language understanding, image tagging and categorization, line drawing detection, Region dependent service availability, thumbnail generation from image and video	REST API
IBM Watson Vision Recognition Service [18]	Visual Recognition service for understanding image content	Image: Class description and class taxonomy, face detection, multi-language support, Image matching identification with confidence score	HTTP
Amazon Rekognition [19]	Deep learning based image recognition and analysis	Object and Scene Detection, Facial Analysis, Face Comparison, Facial Recognition, Integration with (AWS) Amazon Web Services, multi-language support	Command Line Interface, HTTP
CloudSight [20]	Image understanding API	Image description is sent as response when image is sent via REST API	REST API (REST library available in Ruby, Objective-C / Swift, Go, Python)

Table 4 .Description of datasets of PASCAL VOC challenges [29].

Year	Statistics of image dataset				Basic Challenge: Classification and detection and additions to the challenges
	Object classes	Images	Annotated objects	Segmentation	
2005	4	1,578	2,209	-	Classification, detection with ROC-AUC metric
2006	10	2,618	4,754	-	Classification, detection with different dataset – Flickr images and Microsoft Research Cambridge (MSRC) dataset with ROC-AUC metric
2007	20	9,963	24,640	-	Segmentation, person layout taster with avg. precision metric
2008	20	4,340	10,363	-	Addition of Occlusion flag
2009	20	7,054	ROI objects 17,218	3,211	Augmenting previous year's dataset with new set of images
2010	20	10,103	ROI objects 23,374	4,203	Action classification
2011	20	11,530	ROI objects 27,450	5,034	Action classification with more object classes
2012	20	11,530	ROI objects 27,450	6,929	Segmentation with increased dataset size, use of annotated reference points

- PASCAL VOC Challenge dataset [29]: In order to keep pace with recent technology and evaluate solutions to challenging problems in computer vision community, various problems are made open to CV community. Pattern Analysis, Statistical Modelling and Computational Learning (PASCAL) Visual Object Classes (VOC) is renowned project which provided standardized image datasets for recognition of objects, tools for accessing data sets and ran challenges to evaluate the performance of various methods put forth by CV researchers on object class recognition from year 2005 to 2012. Most of the research papers in the domain of object detection follow PASCAL VOC challenges in order to compare and benchmark their proposed system with the standard datasets provided by PASCAL VOC challenge. Table 4 gives the overview of PASCAL VOC challenges from

2005 to 2012. Year-wise comparison of datasets is done on the basis of number of object categories, total number of images, count of annotated objects in addition with region of interest (ROI) based annotated objects, segmentation and basic challenge along with new challenges added every year.

Table 5.Description of image classification datasets of ILSVRC [28].

Year	Object classes	Training images	Validation images	Test images	Challenges
2010	1000	1.2 million	50,000	150,000	Image classification
2011	1000	1.2 million	50,000	150,000	Classification, Classification with localization
2012	1000	1.2 million	50,000	150,000	Classification, Classification with localization , fine-grained Classification
2013	1000	1.2 million	50,000	150,000	Classification, Classification with localization , fine-grained Classification
2014	1000	1.2 million	50,000	150,000	Classification, Classification with localization , fine-grained Classification

Table 6. Description localization datasets of ILSVRC [28].

Year	Object classes	Train images	Train Bbox annotated	Validation images	Validation Bbox annotated	Test image
2013	200	395909	345854	21121	55501	40152
2014-17	200	456567	478807	21121	55501	40152
2016 -17	30	456567	478807	21121	55501	40152

Table 7. Description of object detection datasets of ILSVRC [28].

Year	Object classes	Train images with Bbox annotation	Train Bbox annotated	Validation images with Bbox annotation	Validation Bbox annotated	Test images with Bbox annotation
2011	1000	315,525	344,233	50,000	55,388	100,000
2012-14	1000	523,966	593,173	50,000	64,058	100,000

Table 4 shows PASCAL VOC dataset statistics. Table 5 shows the statistics of image classification datasets used in ILSVRC. Year 2013 focused on simple object detection from images annotated with bounding box. Table 6 depicts the features of single-object localization datasets used in ILSVRC. Later years witnessed the complexity of object detection challenges in which objects are to be detected from images containing clutters, occlusions and objects at multiple scales. Recent challenge focuses on detecting object from video with fully labelled category of 30 objects. The statistics of object detection datasets in ILSVRC is shown in table 7.

There are also some datasets based on image parsing providing better image annotation than image labelling. These datasets are LabelMe[30], and SUN2012 [31]. Scene UNderstanding (SUN) [31] dataset consists of 899 categories of scenes with more than 130K images. LabelMe is web based annotation tool. Each photograph exhibits multiple objects per image and objects are annotated with bounding polygon. Though the annotators have freedom to choose object and label it, the naming convention of objects is not standardized.

5. Application Domains of Object Detection

Object detection is applicable in many domains ranging from defense (surveillance), human computer interaction, robotics, transportation, retrieval, etc. Sensors used for persistent surveillance generate petabytes of image data in few hours. These data are reduced to geospatial data and integrated with other data to get clear notion of current scenario. This process involves object detection to track entities like people, vehicles and suspicious objects from the raw imagery data [32]. Spotting and detecting the wild animals in the territory of sterile zones like industrial area, detecting the vehicles parked in restricted areas are also some applications of object detection.

Detecting the unattended baggage is very crucial application of object detection. For autonomous driving, detecting objects on the road would play important role. Detection of faulty electric wires when the image is captured from drone cameras is also application of object detection. Detecting the drivers' drowsiness on the highway in order to avoid accident may be achieved by object detection.

The requirements of aforementioned applications vary according to the use case. Object detection analytics can be performed offline, online or near real time. Other factors like occlusions, rotation invariance, inter-class and intra-class variation, and multi-pose object detection need to be considered for object detection.

6. State-of-the-art deep learning based approaches of Object Detection

Table 8 compares deep learning methods for object detection which is useful for the research community to work further in the domain of deep learning based object detection. Szegedy et al. pioneered the use of deep CNN for object detection [33] by modeling object detection as a regression problem. They have replaced last layer in the AlexNet [2] with regression layer for object detection. Both the tasks of detection and localization have been performed using object mask regression. DeepMultiBox [34] extended the approach of [33] to detect multiple objects in an image.

How the CNN learns the feature is a major issue. The task of visualizing the CNN features is done by Zeiler et al. [35]. They applied both CNN and deconvolution process for visualization of features. This approach outperforms [2]. They have also justified that performance of deep model is affected by the depth of the network. Overfeat model [36] applies Sliding window approach based on multi-scaling for jointly performing classification, detection and localization. Girshick et al. [37] proposed deep model based on Region proposals. In this approach, image is divided into small regions and then deep CNN is used for getting feature vectors. Features vectors are used for classification by linear SVM. Object localization is done using bounding-box regression. On the similar lines, [38] used regionlets for generic object detection irrespective of context information. They designed Support Pixel Integral Image metric to extract features based histogram of gradients, covariance features and sparse CNN.

Earlier before the dawn of deep learning, object detection was preferably performed using deformable part model technique [39]. Deformable part model technique performs multi-scale based object detection and localization. Based on the principles of this model, Ouyang et al. [40] put forth pooling layer for handling the deformation properties of objects for the sake of detection.

Table 8. Comparison of deep learning based Object Detection methods

Method	Working	Features	Reference
Deep saliency network	CNNs are used for extracting the high-level and multi-scale features.	It is challenging to detect the boundaries of salient regions due to the fact that pixel residing in the boundary region have similar receptive fields. Due to this, network may come with inaccurate map and shape of the object to be detected.	[48-51]
Generating image (or pixels)	This method is used when the occurrence of occlusions and deformations is rare in the dataset.	This method generates new images with occlusions and deformations only when training data contains occurrences of occlusions and deformations.	[55]
Generating all possible occlusions and deformations	In this method, all sets of possible occlusions and deformations are generated to train the object detectors.	This method is not scalable since deformations and occlusions incur large space.	[56], [57]
Adversarial learning	Instead of generating all deformations and occlusions, this method use adversarial network which selectively generates features mimicking the occlusions and deformations which are hard to be recognized by the object detector.	As this method generates the examples on-the fly, it is good candidate to be applied in real time object detection. As it selectively generates the features, it is also scalable.	[58]
Part-based method	This method represents object as collection of local parts and spatial structure. This method exhaustively searches for multiple parts for object detection.	This method addresses the issue of intra-class variations in object categories Such variations occur due to variation in poses, cluttered background, partial occlusions	[39]
CNN with part-based method	In this method, deformable part model is used for modelling the spatial structure of the local parts whereas CNN is used for learning the discriminative features.	This method handles the issue of partial occlusions. But requires multiple CNN models for part based object detection. Finding out the optimal number of parts per object is also challenging.	[59-61]
Fine-grained object detection method	This methods works on annotated object parts during training phase. Part-localization is the fundamental component used in testing phase.	This method has capability to figure out the differences in inter-class objects at finer level. And they work more on discriminative parts compared to generic object detection methods.	[62-65]

For fine-grained level of object detection and localization, Huang et al. [41] proposed task-driven progressive part localization (TPPL) framework. Spatial Pyramid pooling layer and swarm optimization approach is used for detecting the object in the image region. Zhu et al. put forth hybrid method based on segmentation and context modeling for object detection [42] by employing Markov Random Field. The use of multi-scale models and context models is done in [43] for joint object detection and localization.

Approaches mentioned in [33-43] have focused on object detection with intention of maintaining the accuracy of detection. The approaches mentioned in [44-47] have focused on near real time detection of object by maintaining the trade-offs among the performance metrics.

Saliency-inspired approaches are inspired from human vision which has the capability to choose the important information from the complex image. These approaches follow the basis of contrasts in the image. Deep learning achieved remarkable performance in salient object detection [48-51].

Girshick et al. extended the previous work of region of interest pooling from [43] by introducing multi-task training and multi-scale training for faster object detection in [44]. As region of interest pooling works exhaustively in each image region, it is computationally expensive. To alleviate this problem, a method based on region proposal network is put forth in [45] which uses fully convolutional network for simultaneous detection and localization.

For faster object detection, instead of using dense CNN, Kim et al. [46] used shallow CNN in the approach – PVANET. This approach works in pipeline architecture in the consecutive steps of regional proposal generation, feature extraction, and classification based on region of interest. "You Only Look Once (YOLO)" [47] is a popular and widely used framework for object detection at real time due to its characteristic of scanning the image only once while training and testing for inferring the information at context and appearance level.

As image classification datasets possess large amount of training datasets compared to object detection dataset, for harnessing the power of large data available with image classification and use it for object detection, Redmon and Farhadi applied the hierarchical classification method [66]. Their approach namely YOLO9000 is the improved model of YOLO framework [47] and performs detection of around 9000 object categories at real time. YOLO9000 makes use of method for combining the distinct dataset (which are inherently not meant for object detection) and joint training approach in which the model is trained on both ImageNet dataset and Microsoft COCO dataset.

It is expected for object detection systems to robustly perform object detection invariant to illumination, occlusions, deformations and intra-class variations. As occlusions and deformations follow long-tail statistical distribution, there are chances that datasets miss the rare occlusions and deformations of objects. This hinders the performance of object detection systems. Therefore, Wang et al. [58] put forth the approach based on adversary network in which network selectively generates the features of occlusions and deformations which are hard to be recognized by object detector. They used Spatial Dropout Network and Spatial Transformer Network based on adversarial network to generate occlusion and deformation features respectively.

Fine-grained object detection requires finding the subtle differences among inter-class object categories. Fang et al. [67] put forth co-occurrence layer for integrating CNN with part-based method. The co-occurrence layer encodes the co-occurrence between various parts detected by the neurons. This layer does not need part-level annotation as required in part-based models and generates the co-occurrence features using single-stream network.

As assessed from the literature, deep learning methods especially based on convolutional neural networks are applicable to both generic object detection and fine-grained object detection and localization. CNNs being the backbone of object detection techniques are very useful for automatically learning the features used for object detection.

7. Conclusion and Future Directions

Object detection is considered as foremost step in deployment of self driving cars and robotics. In this paper, we demystified the role of deep learning techniques based on CNN for object detection. Deep learning frameworks and services available for object detection are also discussed in the paper. Benchmarked datasets for object localization and detection released in worldwide competitions are also covered. The pointers to the domains in which object detection is applicable has been discussed. State-of-the-art deep learning based object detection techniques have been assessed and compared.

Future directions can be stated as follows. Due to infeasibility of humans to process large surveillance data, there is a need to bring data closer to the sensor where data are generated. This would result into real time detection of objects. Currently, object detection systems are small in size having 1-20 nodes of clusters having GPUs. These systems should be extended to cope with real time full motion video generating frames at 30 to 60 per second. Such object detection analytics should be integrated with other tools using data fusion. The main issue is how to integrate processing into a centralized, powerful GPU for processing data obtained from various servers simultaneously and performs near real time detection analysis. To exploit the representational power of deep learning, large datasets over the size of 100 terabytes are essential. More than 100 million images are required to train the self-driving cars [52]. Deep learning libraries should be augmented with prototyped environments in order to provide paramount throughput and productivity dealing with massive linear algebra based operations. The datasets of image classification are widely available compared to that of object detection, the methods can be devised by which datasets meant for other tasks other than object detection would be applicable to be used for object detection. Existing methods are developed considering object detection as fundamental problem to be solved. There is scope of developing new design mechanisms capable of providing “Object Detection as a Service” in complex applications such as drone cameras, automated driving cars, robots navigating the areas such as planets, deep sea bases, and industrial plants where high level of precision in certain tasks is expected.

References

[1] Technology Trends, https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2018
[2] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. (2012). “Imagenet Classification with Deep Convolutional Neural Networks.” In Advances in Neural Information Processing Systems, 1097–1105.
[3] Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. (2017). “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.” IEEE transactions on pattern analysis and machine intelligence, **39(6):** 1137–49.
[4] Caffe, http://caffe.berkeleyvision.org/
[5] Microsoft Cognitive Toolkit CNTK, https://www.microsoft.com/en-us/research/product/cognitive-toolkit/
[6] TensorFlow, https://www.tensorflow.org/
[7] Theano, http://deeplearning.net/software/theano/
[8] Torch, http://torch.ch/
[9] Chainer, http://chainer.org/
[10] Keras, https://keras.io/
[11] Deeplearning4j, https://deeplearning4j.org
[12] Apache Singa, http://singa.incubator.apache.org/
[13] MXnet, http://mxnet.io/
[14] Neon, http://neon.nerva-nasys.com/docs/latest
[15] Clarifai, https://clarifai.com/
[16] Google Cloud Vision API, https://cloud.google.com/vision/
[17] Microsoft Cognitive Service, https://www.microsoft.com/cognitive-services/en-us/computer-vision-api
[18] IBM Watson Vision Recognition Service, http://www.ibm.com/watson/developercloud/visual-recognition.html
[19] Amazon Rekognition, https://aws.amazon.com/rekognition/
[20] CloudSight, https://cloudsight.readme.io/v1.0/docs
[21] Lin, Tsung-Yi et al. (2014). “Microsoft Coco: Common Objects in Context.” In European Conference on Computer Vision, 740–55.
[22] Deng, Jia et al. (2009). “Imagenet: A Large-Scale Hierarchical Image Database.” In Computer Vision and Pattern Recognition, 248–55.
[23] Torralba, Antonio, Rob Fergus, and William T Freeman. (2008). “80 Million Tiny Images: A Large Data Set for Nonparametric Object and Scene Recognition.” IEEE transactions on pattern analysis and machine intelligence, **30(11):** 1958–70.
[24] Krizhevsky, Alex, and Geoffrey Hinton. (2009). “Learning Multiple Layers of Features from Tiny Images.” Thesis ch.3
[25] Wah, Catherine et al. (2011). “The Caltech-Ucsd Birds-200-2011 Dataset.”
[26] Welinder P., Branson S., Mita T., Wah C., Schroff F., Belongie S., Perona, P. (2010). “Caltech-UCSD Birds 200”. California Institute of Technology. CNS-TR-2010-001. 2010

[27] Griffin, Gregory, Alex Holub, and Pietro Perona. (2007). "Caltech-256 Object Category Dataset."
[28] Russakovsky, Olga et al. (2015). "ImageNet Large Scale Visual Recognition Challenge." Int. Journal of CV, **115(3):** 211–52.
[29] Everingham, Mark et al. (2015). "The Pascal Visual Object Classes Challenge: A Retrospective." Int. journal of CV, **111(1):** 98–136.
[30] Russell, Bryan C, Antonio Torralba, Kevin P Murphy, and William T Freeman. (2008). "LabelMe: A Database and Web-Based Tool for Image Annotation." International journal of computer vision, **77(1–3):** 157–73.
[31] Xiao, Jianxiong et al. (2010). "Sun Database: Large-Scale Scene Recognition from Abbey to Zoo." In CVPR, 3485–92.
[32] Chang, Wo L. (2015). NIST Big Data Interoperability Framework: Volume 3, Use Cases and General Requirements.
[33] Szegedy, Christian, Alexander Toshev, and Dumitru Erhan. (2013). "Deep Neural Networks for Object Detection." In Advances in Neural Information Processing Systems, 2553–61.
[34] Erhan, Dumitru, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov. (2014). "Scalable Object Detection Using Deep Neural Networks." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2147–54.
[35] Zeiler, Matthew D, and Rob Fergus. (2014). "Visualizing and Understanding Convolutional Networks." In European Conference on Computer Vision, 818–33.
[36] Sermanet, Pierre et al. (2013). "Overfeat: Integrated Recognition, Localization and Detection Using Convolutional Networks." arXiv preprint arXiv:1312.6229.
[37] Girshick, Ross, Jeff Donahue, Trevor Darrell, and Jitendra Malik. (2014). "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 580–87.
[38] Wang, Xiaoyu, Ming Yang, Shenghuo Zhu, and Yuanqing Lin. (2015). "Regionlets for Generic Object Detection." IEEE transactions on pattern analysis and machine intelligence, **37(10)**: 2071–84.
[39] Felzenszwalb, Pedro F, Ross B Girshick, David McAllester, and Deva Ramanan. (2010). "Object Detection with Discriminatively Trained Part-Based Models." IEEE transactions on pattern analysis and machine intelligence,**32(9)**: 1627–45.
[40] Ouyang, W et al. (2015). "DeepID-Net: Deformable Deep Convolutional Neural Networks for Object Detection." In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2403–2412.
[41] Huang, Chen, Zhihai He, Guitao Cao, and Wenming Cao. (2016). "Task-Driven Progressive Part Localization for Fine-Grained Object Recognition." IEEE Transactions on Multimedia, **18(12)**: 2372–83.
[42] Huang, Chen, Zhihai He, Guitao Cao, and Wenming Cao. (2016). "Task-Driven Progressive Part Localization for Fine-Grained Object Recognition." IEEE Transactions on Multimedia, **18(12)**: 2372–83.
[43] Ohn-Bar, Eshed, and Mohan Manubhai Trivedi. (2017). "Multi-Scale Volumes for Deep Object Detection and Localization." Pattern Recognition, **61:** 557–72.
[44] Girshick, Ross. (2015). "Fast R-Cnn." arXiv preprint arXiv:1504.08083.
[45] Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. (2017). "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." IEEE transactions on pattern analysis and machine intelligence **39(6):** 1137–49.
[46] Kim, Kye-Hyeon et al. (2016). "PVANET: Deep but Lightweight Neural Networks for Real-Time Object Detection." arXiv preprint arXiv:1608.08021.
[47] Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. "You Only Look Once: Unified, Real-Time Object Detection." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 779–88.
[48] Liu, Nian, and Junwei Han. (2016). "Dhsnet: Deep Hierarchical Saliency Network for Salient Object Detection." In Computer Vision and Pattern Recognition
[49] Li, Xi et al. (2016). "Deepsaliency: Multi-Task Deep Neural Network Model for Salient Object Detection." IEEE Transactions on Image Processing, **25(8)**: 3919–30.
[50] Wang, Lijun, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. (2015). "Deep Networks for Saliency Detection via Local Estimation and Global Search." In Computer Vision and Pattern Recognition (CVPR), 183–92.
[51] Li, Guanbin, and Yizhou Yu. (2016). "Deep Contrast Learning for Salient Object Detection." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 478–87.
[52] Bojarski, Mariusz et al. (2016). "End to End Learning for Self-Driving Cars." arXiv preprint arXiv:1604.07316.
[53] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition." IEEE transactions on pattern analysis and machine intelligence **37(9):** 1904–16.
[54] Yang, Fan, Wongun Choi, and Yuanqing Lin. (2016). "Exploit All the Layers: Fast and Accurate Cnn Object Detector with Scale Dependent Pooling and Cascaded Rejection Classifiers." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2129–37.
[55] Denton, Emily L, Soumith Chintala, Rob Fergus, and others. (2015). "Deep Generative Image Models Using A Laplacian Pyramid of Adversarial Networks." In Advances in Neural Information Processing Systems, 1486–94.
[56] Shrivastava, Abhinav, Abhinav Gupta, and Ross Girshick. (2016). "Training Region-Based Object Detectors with Online Hard Example Mining." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 761–69.
[57] Takác, Martin, Avleen Singh Bijral, Peter Richtárik, and Nati Srebro. (2013). "Mini-Batch Primal and Dual Methods for SVMs." In ICML (3), 1022–30.
[58] Wang, Xiaolong, Abhinav Shrivastava, and Abhinav Gupta. (2017). "A-Fast-Rcnn: Hard Positive Generation via Adversary for Object Detection." arXiv preprint arXiv:1704.03414 2.
[59] Girshick, Ross, Forrest Iandola, Trevor Darrell, and Jitendra Malik. (2015). "Deformable Part Models Are Convolutional Neural Networks." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 437–46.
[60] Wan, Li, David Eigen, and Rob Fergus. (2015). "End-to-End Integration of a Convolution Network, Deformable Parts Model and Non-Maximum Suppression." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 851–59.
[61] Girshick, Ross, Forrest Iandola, Trevor Darrell, and Tian, Yonglong, Ping Luo, Xiaogang Wang, and Xiaoou Tang. (2015). "Deep Learning Strong Parts for Pedestrian Detection." In Proceedings of the IEEE International Conference on Computer Vision, 1904–12.
[62] Chai, Yuning, Victor Lempitsky, and Andrew Zisserman. (2013). "Symbiotic Segmentation and Part Localization for Fine-Grained Categorization." In Computer Vision (ICCV), 2013 IEEE International Conference on, 321–28.

[63] Göring, Christoph, Erik Rodner, Alexander Freytag, and Joachim Denzler. 2014. "Nonparametric Part Transfer for Fine-Grained Recognition." In CVPR, pages-7.
[64] Lin, Di, Xiaoyong Shen, Cewu Lu, and Jiaya Jia. (2015). "Deep Lac: Deep Localization, Alignment and Classification for Fine-Grained Recognition." In Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on, 1666–74.
[65] Zhang, Ning, Jeff Donahue, Ross Girshick, and Trevor Darrell. (2014). "Part-Based R-CNNs for Fine-Grained Category Detection." In European Conference on Computer Vision, 834–49.
[66] Redmon, J, and A Farhadi. (2017). "YOLO9000: Better, Faster, Stronger." In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 6517–25.
[67] Shih, Ya-Fang et al. (2017). "Deep Co-Occurrence Feature Learning for Visual Object Recognition." In Proc. Conf. Computer Vision and Pattern Recognition.