Deep Learning Techniques for Object Detection

Devashree Vaishnav[1], B. Rama Rao[2], Dattatray Bade[3]

[1]Student, [2]Professor, [3]Asst. Professor
Dept. of Electronics and Telecommunication Engineering, Vidyalankar Institute of Technology,
University of Mumbai, Maharashtra, India.

Abstract - *A Paradigm shift has been observed over the past few years from traditional algorithms to data-intensive algorithms with machines enabling predictions without human intervention. This is due to field of Deep Learning for Computer Vision which has observed a spike in trend from the research community. The field is vast with new terminologies showing up in every new research which might tend to overwhelm newcomers into this field. This paper aims to detail the Deep Learning techniques along with Convolutional Neural Network models and their respective versions. The techniques such as ResNet, R-CNN, SSD and YOLO for the task of Image classification for Object Detection have been discussed. This survey is a mapping of Deep Learning Techniques for Computer Vision emphasizing on Object Detection as the base application.*

Key Words: *Artificial Intelligence, Convolutional Neural Networks, Deep Learning, Object Detection, ResNet, R-CNN, SSD, YOLO.*

1. INTRODUCTION

With the boost in Big Data availability since the past few years, the field of Artificial Intelligence [1] has been flourishing with more and more models being developed and implemented on large platforms. The era of Big Data along with the increase in computational capabilities of modern hardware paired with the easier access to Data Science Tools for architecture design and visualization. The sudden technological trend shift towards the field of AI has rendered curiosity and spiked interest amongst researchers from all domains. This aided in the processing of large volumes of information within a shorter time span by combining Artificial Intelligence techniques with Cognitive technologies. Artificial Intelligence enabled systems capable of interpreting the real-world in a manner similar to human perception by allowing systems to interpret human behavioural aspects such as vision and speech.

From movie and music recommendations on streaming sites, product recommendations on e-commerce sites; calculating distance and traffic estimation in Map applications, Chatbots providing assistance, targeting advertising to real-world applications like self-driving cars, airplane simulators, face detection, path detection; Artificial Intelligence has entered all walks of life and has seen monumental growth.

The task of visual perception seems trivial to the human eye and within split seconds it is possible recognize multiple objects within visual scope of human eye. This task of visual perception is extremely complex for computers since they interpret in binaries and determining objects from an image requires representing each pixel in the image by Hex number to define colour, image segmentation, finding corners, edges, shapes, textures, patterns and guessing the object. This seemingly trivial task for humans converts into a complex task or machines to understand an image and extracting information out of it.

The field of Computer Vision relies on Deep Learning techniques along with Pattern Recognition so as to determine the presence of objects in an image or video. Computer Vision methods are utilized especially to recognize and interpret images by means of image analysis and pre/post-processing methods derived from Image Processing. The goal of computer vision is to understand the content of images which typically involves developing methods that attempt to reproduce the capability of human vision. For machines to understand the content of images may involve extracting a description from the image, which may be an object, a text description, a three-dimensional model etc.

The paper is organized as follows: Section II provides the prerequisites required for the field of Deep Learning, Section III details the Convolutional Neural Network base architecture with sub-sections discussing the layers of the architecture, Section IV highlights the Models widely used for Object Detection. Section V comprises a discussion of the various ways in which Deep Learning techniques are currently being implemented by researchers and Section VI concludes the survey with an emphasis on the purpose of said survey and further utilization of the same.

2. PREREQUISITES OF DEEP LEARNING

Artificial Intelligence, Machine Learning [2] and Deep Learning [3] are three separate concepts; though not mutually exclusive. Artificial Intelligence deals with computer programs learning to make decisions and imitate complex human behavioral aspects such as speech or vision.

Machine Learning is a branch of Artificial Intelligence which utilizes rules to gain new knowledge skills and continually improve its own performance by repetitive self-evaluation; based on statistical theories for training and testing the algorithms and evaluating the performance.

Machine learning is broadly classified into Supervised learning, Unsupervised Learning and Reinforcement learning. Supervised Machine Learning involves the utilization of prior data to learn about features and applying this learnt knowledge on new data to make predictions or decisions; the input as well as expected output are known to the machine learning model/algorithm prior to training and testing. Unsupervised Learning involves feature learning without any knowledge of prior data and learning the uncertainty by grouping similar data it encounters and pinpointing unknown features to make predictions or decisions based on the accumulated knowledge during feature learning process. The training process involves learning the values/parameters through large volumes of examples from labelled data for supervised learning whereas the data is unlabeled in case of unsupervised learning and then searching for those parameters in the test data.

Reinforcement learning focuses on maximizing the rewards gained from a certain task using both supervised and unsupervised techniques. It uses the available data to train the models to take cumulative action based on the decisions and predictions. Deep Learning is a subset of Machine Learning which utilizes the concepts of Machine Learning and implements cognitive decision making algorithms in areas like vision, speech etc. Deep Learning mimics the human brain neural tendencies in order to achieve Machine Learning. It deals with utilization of the Neural Networks in order to replicate human behavioral aspects through imitation of synaptic responses of the brain for the evaluation of complex functionalities via algorithms.

2.1 Neural Network

A neural network comprises interconnected neurons each having inputs and outputs, with the output of one layer becoming the input to the next layer; except the final layer whose output is the ultimate outcome of the entire neural network. The neurons are interconnected with each connected having a weight value associated to it. A neuron activates when the input sum of values surpasses a set threshold. The amount of forward propagation for the network is determined by the weights and propagation functions are used between neurons to compute the input to the next layer. By feeding data forward into the progressing network each subsequent hidden layer is capable of handling higher level features compared to prior layer. The weights can be modified during *Backpropagation* [4] so as to serve preferred output generation for a given input while the network is in process of learning. By modifying the weight and bias associated, the significance of a link between neurons can be emphasized or diminished as per requirement. A neural network learns the weights and generates predictions/detections by iteratively conducting forward and backward propagation on individual data in a training dataset. The accuracy of network relies upon the volume of data set, hence larger the amount of data, higher is the network accuracy at predicting outcomes.

The term *deep* in Deep Learning refers to the depth of the Neural Network i.e. the number of hidden layers in the network, which is usually more than three. Neural Networks face the *Vanishing Gradient Problem* which involves the difficulty in calculating the adjustments required to be made to weights during each step of training, depending upon the number of hidden layers in the network.

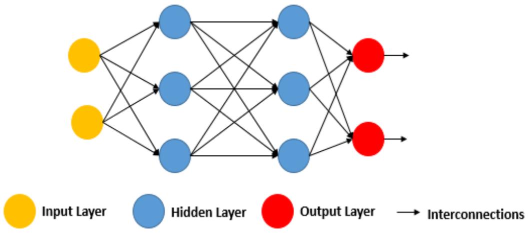


Fig.1. Neural Network

3. CONVOLUTIONAL NEURAL NETWORKS

Convolutional Neural Networks (ConvNet/CNN) are the most popular Deep learning technique in the field of Computer Vision; widely used to solve image and vision related problems, in particular, image classification, object detection and pattern recognition. CNNs can simply be thought of as simultaneous parallel filtering of multiple images at the same time. It is essentially a technique which can accept an input image and learn its various aspects (features) by assigning importance through learnable weights (specific pixel values of matrix/vector) and biases in order to distinguish those aspects in further stages.

Traditional image processing involves ample of explicit pre-processing to compute specific outcomes. With the presence of large volumes of data such as images with 4K (3840×2160) resolutions, it is computationally intensive to perform explicit pre-processing on images. Whereas CNNs require relatively low pre-processing due to their ability to learn the image filters/parameters (weights). CNNs achieve this learning by capturing Spatial and Temporal dependencies of an image without losing the critical features through relevant filter application and once match occurs that particular filter weight is learnt and reused in later stages as required. CNN tends to gradually shrink the size of feature maps with respect to the depth of network as it goes into deeper layers. The shallow layers cover small receptive fields while the deeper layers cover large receptive fields to provide abstract representation of features. This ability of weight learning along with reusability paired with scalability on massive datasets and reduction in general pre-processing makes the CNNs an essential in Deep Learning.

The components of a Convolutional Neural Network structure are as follows: Convolutional layers, Pooling Layers, Activation Function, Loss Function, Fully Connected Layer.

3.1 Convolutional Layer

This layer performs the task of passing a kernel/mask/template over the input image to generate a convolved output. The layer comprises multiple filters working in conjunction to perform simultaneous convolution operations on the input image. Each individual filter is basically a *k x k* matrix of weights (parameters) W_i which are to be learned and generate affine transform of the input image. A linear combination of all the simultaneous outputs generated by each of these filters is produced based on the filter size and neighbourhood pixels of image under consideration. The region being processed is known as the *local receptive field* [5]. A convolutional operation may have different padding, default being zero-padding and different strides (step between each local filtering) and default being 1.

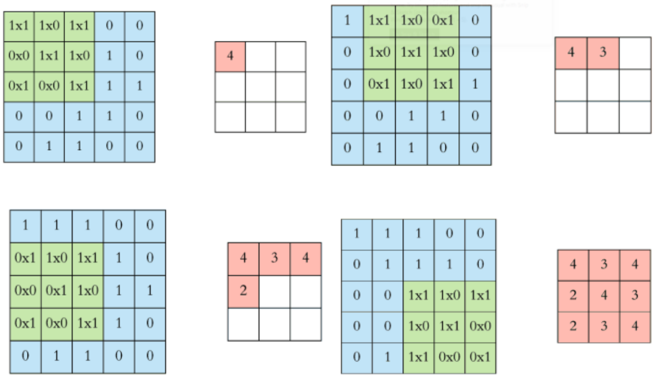


Fig.2. Examples of filter (green) sliding over image (blue) to compute feature map (red).

3.2 Pooling Layer

Pooling layers are placed after each convolutional layer in the base architecture for downsampling purposes in order to reduce the spatial dimensions of the convolved feature vector. This layer used for the purpose of dimensionality reduction in order to reduce the computational power required in data processing. Pooling is essential since the depth (third dimension) of the *tensor* often increases and thus it becomes crucial to reduce the first two dimensions (feature and target) in order to strike a balance. The significance of pooling layer is highlighted by the fact that it extracts dominant features (rotational and positional invariant features).

Pooling is performed by either of the two methods: *Max Pooling* and *Average Pooling*. Max Pooling involves noise suppression along with dimensionality reduction followed by de-noising and returns the maximum value from the portion of image covered by the kernel/mask/template. Average Pooling involves returning average of all values but does not perform noise suppression. Pooling Layers along with Convolutional Layers form the generic infrastructure of an i-layer CNN and these layers can be increased or decreased to capture low level features and better suit the usability and requirements of the CNN, though at the cost of higher number of computations.

3.3 Feature Maps and Activation Functions

Activation functions are used after convolutional layers and fully connected layers for the purpose of firing a neuron to indicate its priority over other neurons in the same layer. Rectified Linear Unit (ReLU) is a linearly incremental function for positive values with constant negative values; which is the most common activation function used in CNNs. Another variation is the Leaky ReLU with α =0.01 and the Parametric ReLU (PReLU) which lets minor negative features parametrize by 0 ≤ α ≤ 1 [6]

On each convolution a new vector is formed and passed through the activation function to generate a feature map. A *tensor* is formed by stacking the feature maps which is then further relayed to the next layer as an input.

3.4 Fully Connected Layer

Fully Connected (FC) layer essentially learns the non-linear combinations of high level feature representations of the outputs from all previous convolutions and converts it into a suitable form to be given as input to Neural Network for the task of classification or prediction. This layer performs the function of classification by learning the weights and generates a single column vector of scalar elements unlike the previous layers generating matrices. This flattened output generated by the fully connected layer is usually the reshaped data from the previous layer, is fed to the neural network and *Backpropogation* is applied via multiple epochs (similar to iterations) to compute classification. Usually this layer distinguishes between the dominant and low-level features to generate the class probabilities for each class C using underlying classification techniques such as *Softmax classifier* [7]. Loss/Cost functions determine the performance intensity of the model and for the softmax classifier, often used in neural networks, the cross-entropy loss is employed.

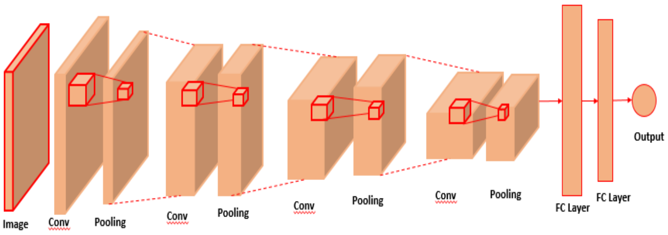


Fig.3. General Architecture of a Convolutional Neural Network.

4. DEEP LEARNING MODELS

Convolutional Neural Networks are widely used for the task of image classification, object detection and pattern recognition in the field of computer vision. The purpose of said survey being object detection techniques, some of the popular architectures are highlighted.

4.1 ResNet (Residual Networks)

K. He et al. [8] proposed residual blocks for networks with 34 to 152 layers after the success of VGG-Net [9] claiming that optimization issues in the latter leads to initial accuracy saturation on increasing the depth of network with further degradation and failure to train leading to underfitting. It provides an error rate of 3.6% and set new records in classification, detection, and localization through a single network architecture. ResNet focusses on preserving the original characteristics of initial feature vector *x* prior to transformation by some layer $f_l(x)$ by skipping weight layers and performing sum $f_l(x) + x$. It proposes a solution to the *vanishing gradient/diffusion problem* by introducing the gradient as an additive term. Moreover the architecture rids of fully connected hidden layers and performs Average pooling after convolution and generates the output layer. The core idea of ResNet is to add shortcut connections that by-pass two or more stacked convolutional layers by performing identity mapping, which are then added together with the output of stacked convolutions.

4.2 R-CNN (Region based CNN)

Ross Girshick et al. [10] proposed R-CNN which is a region-proposal based CNN used in object detection tasks. R-CNN models showed that a CNN could lead to dramatically higher performance on PASCAL VOC datasets [11] as compared to the systems based on simpler HOG-like features. An R-CNN model performs four main tasks of generation of region proposals that are category independent, extraction of fixed-length feature vector from each proposed region, classification of objects in an image based on a set of class specific linear Support Vector Machines [12] , performing regression to determine a bounding box pertaining to each prediction.

A selective search method is used to extract region-proposals followed by a CNN for extracting 4096-dimension feature vectors from each region-proposal. Each category trains category-independent SVMs which do not share parameters among different SVMs while all the CNN parameters are shared across all categories. Region-proposals are of same size since the fully connected layers require fixed length vector inputs. Since various images given as input to the network have different sizes and aspect ratios, the region proposals extracted initially also have different sizes and aspect ratios. These region-proposal pixels are warped by a tight bounding box around them to the required size of 227×227. Stochastic Gradient Descent [13] is used to fine-tune the CNN parameters so as to warp the region-proposals. The feature extraction network comprises 5 convolutional layers followed by 2 fully connected layers.

R-CNN is trained on the ImageNet Classification Dataset [14] and last fully connected layer is replaced by the CNNs ImageNet specific 1000-way classification layer. The last fully connected layer is randomly initialized and is the (N+1)-way classification layer for N number of object classes with 1 background. The IOU (intersection over union) threshold is set to 0.5 over which all the region-proposals that do not cross this threshold are negatives and discarded. While those region-proposals that cross the threshold are taken as positives and ones with maximum IOU overlap are assigned to ground-truth bounding box.

A drawback of R-CNN is the time taken on SVMs classification since it initially performs a ConvNet forward pass for each region-proposal without sharing computations. This is overcome by Fast R-CNN [15] by extracting features from an entire input image before passing to the region of interest (RoI) pooling layer and extracting fixes size feature map from region-proposals of different sizes and aspect ratios, thus eliminating the need for warping while still retaining spatial information of features in the region-proposals. Further the RoI pooling layer provides fixed size features as required followed by fully connected layers to perform regression and obtain bounding boxes. Since feature extraction happens once for the entire image, the classification and localization at the CNN occurs at a faster rate compared to the R-CNN. Comparatively, R-CNN comprises 4 stages of pre-training, fine-tuning parameters, SVM classification and bounding box regression; while Fast-RCNN is an en-to-end process which utilizes joint-training for classification and bounding box regression using a multi-task loss on each labeled RoI. Fast R-CNN utilizes truncated SVD to accelerate the forward pass of computing fully connected layers.

The drawback of Fast R-CNN is that for RoI proposal requires selective search method which is time consuming since it takes the same amount of time as running time of detection network. Hence Faster R-CNN [16] was proposed with a separate Region Proposal Network (RPN) which is a fully convolutional neural network dedicated to the task of region-proposal generation. Due to sharing a set of common layers amongst the Detection network and RPN, the overall computational speed is accelerated. Faster R-CNN achieves 69.9% mAP (mean average precision) on the PASCAL VOC 2007 test dataset; as compared to 66.9% mAP for Fast R-CNN and 66.0% mAP of R-CNN. The results for three versions of R-CNN trained and tested on PASCAL VOC 2007 test dataset are shown in Table.1. The general architecture of R-CNN is shown in Fig.4 and result of detection is shown in Fig.5.

Table- 1: Table of Parameters for each version of R-CNN

Version	Parameters		
	Mean Average Precision (mAP)	Speed	Time per test image
R-CNN	66.0%	1x	50 sec
Fast R-CNN	66.9%	2x	2 sec
Faster R-CNN	69.9%	250x	250 sec

[a] Values from results based on experiments conducted by Girshick[10]

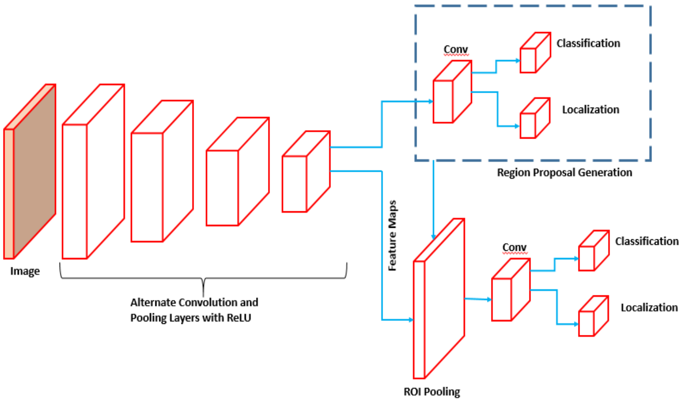


Fig.4. General Architecture of Region based Convolutional Neural Network (R-CNN)

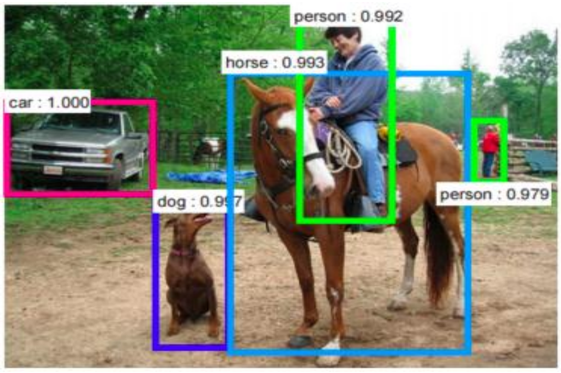


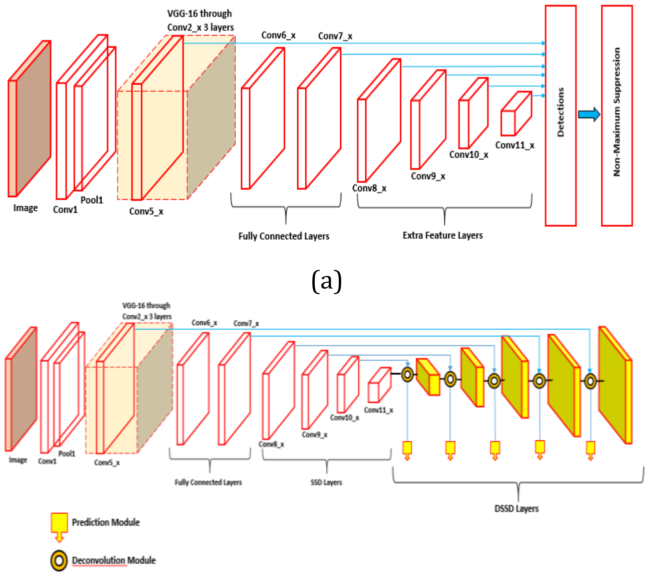
Fig.5. Example detections using RPN proposals of Faster R-CNN on PASCAL VOC 2007. Image from Girschick [16]

4.3 SSD (Single Shot Detectors)

Liu et al. [17] proposed Single Shot Detector model in pursuit of higher accuracy. The output space for bounding boxes is discretized by SSD into a set of default boxes over different aspect ratios and scales on each feature map. SSD models generate a number of priors (multiple boxes for every feature point), used to match ground truth boxes to determine the labels and bounding boxes. It utilizes the method of scaling the representations of ground-truth (actual values) boxes to the same scale to boost the positive targets (boxes having an object assigned) by matching ground-truth boxes to multiple priors. The criteria for matching a ground-truth to prior box is the Jaccard index or IOU (intersection over union) which means the more the overlap the better the result.

The large number of priors labelled as background tend unbalance the dataset. To balance it, *Hard Negative Mining* (only count the background priors with highest confidence into the computation of total loss function while ignoring others) is utilized, thus lowering the ratio between background priors and matched priors. Moreover, since multiple priors are detected, Non-Maximum Suppression (NMS) needs to be applied in order to keep the bounding boxes with higher probabilities while removing those with lower probabilities. SSD512 version achieved 76.9% mAP (mean average precision) on the VOC2007 dataset since it uses multi-scale bounding boxes on the top feature map. When small objects are to be detected, models employing large receptive fields may tend to confuse. Hence SSD uses shallow layers to predict small objects and deeper layers for predicting large objects since small objects will not require large receptive fields. Despite using shallow layers specifically for smaller object detections, SSD suffers with estimating small objects and tend to be less accurate than two-stage detectors but are significantly faster.

Fu et al. [18] proposed DSSD (Deconvolutional SSD) to overcome the barrier of low detection accuracy for small objects. The DSSD model is a combination of residual-101 network and large scale context in object detection by simply adding a series of deconvolutional layers after the original SSD model architecture. These deconvolutional layers in turn prove beneficial to increase the feature map resolution. DSSD achieved 81.5% mAP on VOC2007, 80.0% mAP on VOC2012, and 33.2% mAP on COCO.



(a)

(b)

Fig.6. Network Architectures (a) Single Shot Detector (SSD) and (b) Deconvolutional Single Shot Detector (DSSD)

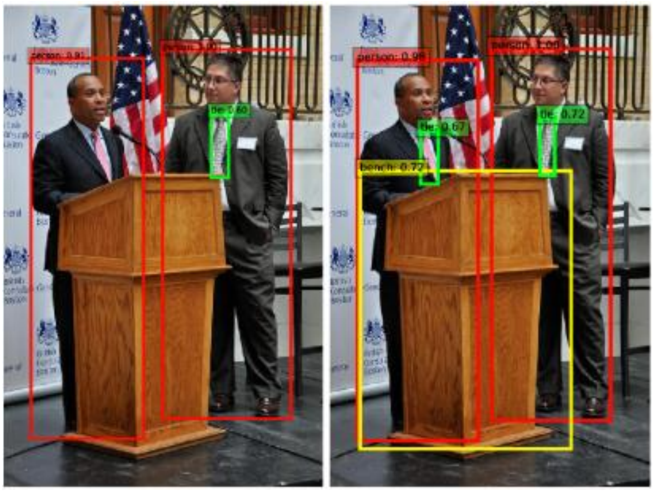


Fig.7. Example of Detections by (a) SSD and (b) DSSD showing smaller objects such as Bench and Tie detected by DSSD and neglected by SSD. Image taken from Fu et.al [18]

4.4 You Only Look Once (YOLO)

Redmon et al. [19] proposed the YOLO model that approaches object detection as a regression problem to determine bounding box coordinates and class probabilities. The model only requires to *look once* at the entire image to determine which objects are present, by dividing the entire image into S×S grids and determining the bounding boxes based center of object lying in particular grid cell. Each grid cell predicts B bounding boxes with center co-ordinates (x,y) along with width (w) and height (h) and the confidence score for each predicted bounding box with C-dimensional conditional class probabilities for C categories. The confidence score is the estimation of prediction accuracy for each bounding box and is determined by the product of $P_{r(obj)}$ denoting probability of box containing an object and IOU denoting intersection over union. YOLO is a fully convolutional network and its eventual output is generated by applying a 1 x 1 kernel on a feature map. The base network runs at 45 fps without batch processing on a Titan X GPU.

During pre-training phase the first 20 convolutional layers are used along with an average pooling layer followed by a fully connected layer. The architecture is structured as a feature extraction network pipeline comprising 24 alternate convolution and pooling layers followed by 2 fully connected layers for classification. The precision of visual information is improved by using double the input resolution of 224×224 of the pre-training stage. Thus the overall network performance improves as compared to the pre-training phase.

YOLO9000 (YOLO v2) [20] is based on a custom deep architecture darknet-19 and is achieved by performing joint training for both object detection and classification for 9,000 separate object classes. It is trained simultaneously on both the ImageNet classification dataset and COCO detection dataset and can predict detections for object classes that do not contain labelled detection data. The limitations of YOLO v2 are that it struggles while detecting small objects due to input downsampling and to solve this issue it utilizes an identity mapping to concatenate feature maps from a previous layer to capture low level features of small objects.

YOLO v3 [21] introduces residual blocks similar to ResNet along with Upsampling and detections at three different scales. YOLO v3 uses a variant of Darknet-53, which originally has 53 layer network trained on Imagenet. For the task of detection, 53 more layers are stacked onto it, resulting in a 106 layer fully convolutional underlying architecture. The output layer generates detections by applying 1 x 1 kernels on feature maps of three different sizes at three different places. By allowing joint training and detection for entire image at once, YOLO provides as advantage over other systems since it is essentially a single CNN performing both the tasks of bounding box prediction and determining class probabilities for those bounding boxes.

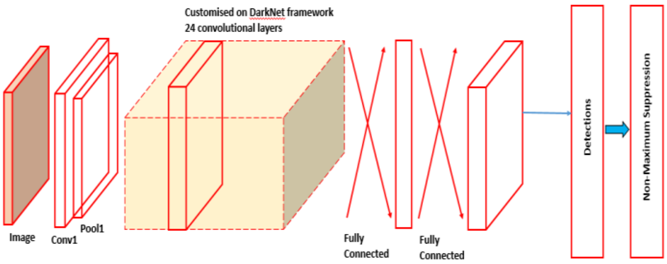


Fig.8. YOLO Network Architecture containing a custom base network of 24 convolutional layers based on the DarkNet framework followed by 2 fully connected layers.

Fig.9. Example detection using YOLO version1 running on sample artwork taken from the internet. Image from Redmon [19]

5. DISCUSSION

The field of Deep Learning is rapidly evolving with new models being developed due to the emergence of powerful computational resources. Detection models with high accuracy and efficiency are the need of the hour with the increasing demand for highly accurate real-time systems. From constructing new architectures and improving model speed considerations to enhancing classification confidence scores and generating models with sophisticated localization techniques; the research of Deep Learning has been flourishing over the past decade due to the generation of large volumes of data every millisecond.

A topic of interest amongst researchers is the combination of single stage and two-stage detectors by retaining high processing speed of one-stage networks in order to maintain accuracy while eliminating redundancy. Real-time System research in Computer Vision focuses on Video Object Detection pertaining to motion-blur and de-focus, target ambiguity in video frames, occlusions and truncations etc, which is difficult to achieve in local as well as remote sensing areas making it a key requirement in video related research. Another area of interest in Deep learning research is Unsupervised Learning algorithms aiming to make models more intelligent and less dependent on human training for learning and models gaining experience on own terms by employing self-evaluation tactics. 3D object detection is yet another field of research involving LiDAR point cloud for obtaining depth information to generate localization and characterization of objects. Computer Vision applications also extend to the medical field with images of patients routinely analysed for detection of anomalies. Neural Networks are being employed to detect cancerous cells from images, detection of heart risks from patients' speech patterns and retinal image analysis with further research focusing on passive patient monitoring.

With modern research focusing on development of models extracting rich features, resolving issues pertaining to occlusions in images, combining single and two-stage detectors to provide better results, applying NMS for enriched post-processing, resolving positive-negative imbalance, pose estimations etc, the field of Computer Vision has been largely impacted by Deep learning. Though these models are currently being applied to all walks of life such as military, bio-medical, transportation, commerce, domestic resources; there still lies a higher scope for improvement.

6. CONCLUSION

Convolutional Neural Networks improve upon the classic techniques that require hand-engineered feature extraction by learning the weights. Though a rapid advancement is observed in the field, there still remain grey areas which can be explored further to gain a better understanding of this field so as to allow Object Detection and Recognition to potentially benefit by the application of Deep Learning Techniques. Though CNNs prove beneficial to resolve complex problems which are not generalized such as non-linear issues and uncertain datasets; it still remains saturated to classification problems at large and other areas are unexplored. Apart from focusing on the technical challenges, Deep Learning opens opportunities for new applications such as truncated object detection and recognition, occlusion detection, building 3D image structures from 2D images, in which Object Detection can make a difference. CNNs serve as a tool allowing scaling of image processing structures beyond traditional utilization to tackle real-world problems with implicit models that generalize well.

Pertaining to the survey carried out, we aim to utilize Deep learning models for Object Detection and Recognition to design an algorithm and a device to capture static images in real-time by a visually impaired user and provide an auditory feedback explaining the scenario in the acquired image. With the paradigm shift in the way algorithms function without human intervention, Deep Learning is the key to design implicit general models to tackle unprecedented large-scale influential challenges.

REFERENCES

1. D. Ostrowski, "Artificial Intelligence with Big Data," *2018 First International Conference on Artificial Intelligence for Industries (AI4I)*, Laguna Hills, CA, USA, 2018, pp. 125-126.

2. Y. Bengio. Learning Deep Architectures for AI. Foundations and Trends in Machine Learning. Now Publissher Inc, 2009.

3. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016, http://www.deeplearningbook.org.

4. G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," Science, vol. 313, no. 5786, pp. 504–507, 2006.

5. A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," arXiv preprint arXiv:1901.06032, 2019.

6. Bing Xu, Naiyan Wang, Tianqi Chen, Mu Li, "Empirical Evaluation of Rectified Activations in Convolution Network", arxiv: 1505.00853, 2015.

7. R. Ranjan, C. D. Castillo, and R. Chellappa, "L2-constrained softmax loss for discriminative face verification," arXiv preprint arXiv:1703.09507, 2017.

8. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," arXiv: 1512.03385, 2015.

9. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

10. R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, 2014: 580-587.

11. M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," International Journal of Computer Vision, vol. 88, pp. 303–338, Jun 2010

12. Huan-Jun Liu, Yao-Nan Wang and Xiao-Fen Lu, "A method to choose kernel function and its parameters for support vector machines," *2005 International Conference on Machine Learning and Cybernetics*, Guangzhou, China, 2005, pp. 4277-4280 Vol. 7.

13. A. Arcos-Garcıa, J. A. Alvarez-Garcıa, and L. M. Soria-Morillo, "Deep neural network for traffic sign recognition systems: An analysis of spatial transformers and stochastic optimisation methods," Neural Networks, vol. 99, pp. 158–165, 2018.

14. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," International Journal of Computer Vision, vol. 115, pp. 211–252, Dec 2015.

15. Girshick R. Fast R-CNN. IEEE International Conference on Computer Vision, 2015: 1440-1448.

16. S. Q. Ren, K. M. He, R. Girshick, J. Sun. Faster R-CNN: towards realtime object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.

17. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg. SSD: Single shot multibox detector. European Conference on Computer Vision, 2016: 21-37.

18. C. Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg. DSSD: Deconvolutional single shot detector. arXiv:1701.06659, 2016.

19. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.

20. J. Redmon, and A. Farhadi. Yolo9000: Better, faster, stronger. arXiv:1612.08242, 2016

21. J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," CoRR, vol. abs/1804.02767, 2018.