International Journal of Communication Networks and Information Security
2024, 16(4)
ISSN: 2073-607X,2076-0930
https://ijcnis.org/

Research Article

Hybrid Convolutional Neural Network Model to ascertain the Objects in Dynamic Cluttered Environment

Kritika Vaid[1], Dr. Deepak Chandra Uprety[2]

[1]Research Scholar, Department of Computer Science& Engineering, IEC University Baddi, Solan (HP), India

[2]Research Guide and Associate Professor, Department of Computer Science& Engineering, IEC University Baddi, Solan (HP), India

*Corresponding Author:[1]kritikavaid172@gmail.com, [2]deepak.glb@gmail.com

ARTICLE INFO

Received: 10 Aug 2024
Accepted: 14 Sep 2024

ABSTRACT

The field of computer vision has made significant strides in object detection in recent years, primarily because of the introduction of deep learning techniques, specifically Convolutional Neural Networks (CNNs). We have introduced a novel method for the multi-object detection in multi-scene cluttered environment in the proposed work. In order to build multi-scale andmulti-scene object detection, in our work we have provides a multi-scale neural network basedonthehigherresponse of FastR-CNNarchitecture. For the experimental work,we have considered different categories of different objects. The dataset is designed to facilitate the development of object detection techniques. It comprises 12,165 object chips, each consisting of 256 pixels in both azimuth and range dimensions. This dataset encompasses diverse primary backgrounds and object sizes. Furthermore, well-known cutting-edge object detectors that have been trained on real-world images are modified to serve as baselines, guaranteeing the availability of reliable and practical reference points. Experimental results indicate that these object detectors not only enhance various quantitative metrics but also achieve unprecedented levels of accuracy, surpassing the capabilities observed in prior studies.

Keywords:Convolutional Neural Networks, Computer Vision, Object detection, Genetic Algorithm, Particle Swarm Optimization.

INTRODUCTION

Object detection represents a critical area of study within computer vision, finding applications across various domains such as surveillance, object recognition, event analysis, and human-computer interfaces [1, 2, 3]. While extensive research exists in the literature on object tracking, many current algorithms excel primarily in simple scenarios where the target object exhibits slow motion and minimal occlusion [1, 2]. However, in more complex situations characterized by factors like illumination changes, pose variations, rapid motion, partial occlusion, and background clutter, existing approaches may encounter tracking drifts. Hence, there is a demand for robust visual dynamic object detection methods to address these challenges effectively.

Based on their mode of representation, existing object detection techniques can be divided into two main groups: parts-based and holistic techniques. The holistic approach relies on global visual cues to model the target's appearance, which is effective for detecting large objects [4-6]. However, in scenarios with occlusion, deformation, or other local visual changes, the holistic approach may struggle to adapt, leading to mismatches or drifting during tracking. Conversely, the parts-based approach captures spatial information by modeling the target's appearance using local patches. [7-9]. While these patches may be loosely interconnected or unconnected, allowing for some spatial deformation, they provide flexibility in updating the visual model, making them suitable for short-term tracking. These approaches excel in handling motion variations and partial occlusion but may drift in situations with background clutter or motion blur due to their focus solely on local cues.Both the holistic and parts-based approaches concentrate on either global or local visual cues, potentially resulting in drifts during tracking. Moreover, these approaches often overlook important contextual cues, leading to distorted localization results.

Consequently, existing tracking methods may struggle to effectively track target objects in more complex environments.

LITERATURE

Object detection encompasses a wide range of applications and techniques for identifying and categorizing objects within images [1]. This section of the study aims to thoroughly explore and compare classical and modern object detection methods, along with their respective applications, using publicly available real-world datasets [2-7]. Object detection involves identifying specific items within images and assigning them to predefined categories based on their unique attributes [8-11]. While numerous object detection methods exist, each with its own strengths and weaknesses depending on factors such as application domain, implementation details, and choice of algorithm, this study categorizes classical techniques into two main types: region-based and classification/regression-based [12-16].

Region-based approaches work in a step-by-step manner, segmenting input images into regions of interest before classifying them into pre-established groups. On the other hand, classification/regression-based techniques adopt a more holistic approach, performing image classification and object recognition simultaneously within a unified framework to mitigate noise and other issues [17-24]. Region-based Convolutional Neural Networks (R-CNN), Fast Region-based Convolutional Neural Networks (Fast R-CNN), Faster Region-based Convolutional Neural Networks (Faster R-CNN), and Mask Region-based Convolutional Neural Networks are a few of the techniques that fall under these categories [25-30]. Additionally, several advancements have been made to address issues such as inaccurate localization. In order to explicitly penalize localization inaccuracies, [31–33] trained class-specific CNN classifiers with a structured loss and utilized a Bayesian optimization-based search algorithm to guide bounding box regressions. In order to impose geometric constraints on object parts, [37] proposed a deformable deep CNN with a novel deformation constrained pooling layer. [34–36] also improved object detection for RGB-D images by incorporating semantically rich image and depth features. Proposing a way to enhance performance by biasing sampling to match ground truth bounding box statistics, [38–40] examined the function of proposal generation in CNN-based detectors.

PROPOSED METHODOLOGY

The initial convolutional layers play a crucial role in identifying small objects within input images. In addition to improving performance by offering semantic understanding, this integration of deep and initial convolutional layers also maintains the spatial structure of low-resolution images. After that, bounding boxes and objects from various classes are used to interpret the Single Shot Detector's (SSD) outputs. Multiple feature maps with varying scales are used for accuracy evaluations and confidence prediction. Illustrated in Figure 3.1 is the classical architecture of the Single Shot Detector.

In contrast to traditional sliding window methods, the majority of SSD approaches use a grid structure, where each grid cell is tasked with identifying objects in different areas of the input image. This region-based technique facilitates straightforward object detection within each grid, followed by classification into predefined classes or labels for detected objects. Grids without detected objects are designated as background images, while grids containing multiple objects may utilize techniques such as anchor box and receptive field. Each anchor or box is assigned to one grid, ensuring predefined allocation.

We propose an architecture with five max-pooling layers and twenty-two convolution layers. As seen in Figure 3.2, the first 16 convolution layers take features from the input image, and the last 6 convolution layers are used for object detection. Training is conducted on images sized 608 × 608, predominantly utilizing 3 × 3 and 1 × 1 filters, with filter numbers doubling after each pooling step.

The performance of end-to-end detection techniques is heavily influenced by anchor box quality. Anchor boxes based only on fixed sizes might not sufficiently cover the range of object sizes found in the dataset, even though they can be generated with fixed sizes and scales. Our approach introduces a multi-scale anchor box methodology to enhance detection accuracy, employing a grid size of 19 × 19 for detecting small objects. The final convolution layer generates 675 output tensors based on class presence probabilities and bounding box coordinates. Subsequently, anchor boxes are selected, and regression is applied to predict object classes along with bounding box coordinates. Next, among overlapping bounding boxes, the bounding box with the highest probability is chosen using a non-max suppression technique with IoU = 0.5.

During the prediction phase, feature maps serve to identify objects within predefined anchor boxes. However, not all anchor boxes may yield sufficient information for effective object detection, resulting in increased processing time. To mitigate this issue, our research introduces an efficient method for leveraging anchor boxes during prediction. This method involves sorting anchor boxes in descending order of size and applying prediction exclusively to boxes containing relevant information, thereby reducing computational and memory overhead. We validate this approach through an optimized multi-scale anchor box technique, which utilizes the canny edge detector algorithm to assess information presence within anchor boxes. Moreover, arranging anchor boxes in descending order further streamlines the detection process by prioritizing larger-scale boxes. These strategies are succinctly outlined in our work, elucidating both the training and detection phases of the proposed model (Figure 1 and 2).

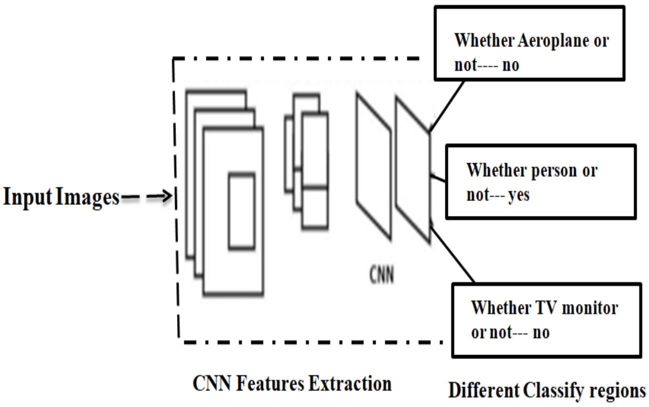

Input Images
CNN
Whether Aeroplane or not---- no
Whether person or not--- yes
Whether TV monitor or not--- no
CNN Features Extraction
Different Classify regions

Figure 1. Flow process of the data processing

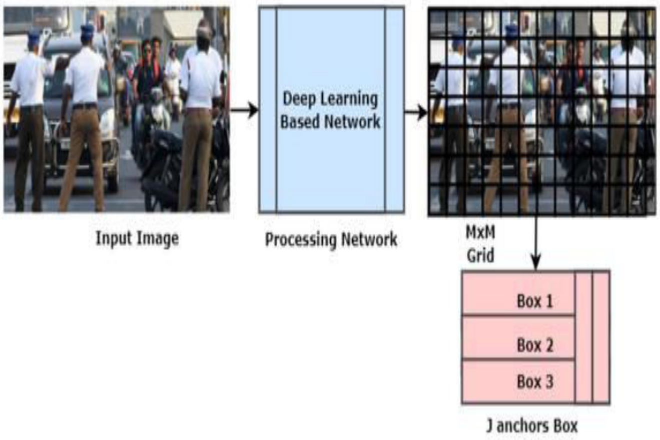


Figure 2. Processing of the Input image for Bounding Box

The feature maps utilized for predicting objects within predefined anchor boxes may not uniformly contain sufficient information for object detection. Despite this, prediction scores are computed for all anchor boxes, leading to increased execution time during both training and testing phases of object detection. Consequently, real-time object detection becomes time-intensive and diminishes the Frames Per Second (FPS) rate. To address this challenge, our research introduces an effective method for leveraging anchor boxes in prediction tasks. Specifically, rather than applying prediction to all anchor boxes indiscriminately; we employ a selective approach to account for the potential absence of information in certain anchor boxes. To validate this approach, anchor boxes are sorted in descending order based on their sizes. If a large-sized anchor box contains no information, exploration of smaller anchor boxes within the same grid is deemed unnecessary. This methodology is implemented within the final layer convolution block, resulting in significant savings in computational resources and memory usage.This approach is delineated in two key components:The following is how an efficient multi-scale anchor box determination procedure evaluates the presence of information:

The canny edge detector algorithm [23] is applied exclusively to the anchor box section. This algorithm necessitates minimum and maximum threshold values to distinguish between weak and strong edges. To streamline this procedure, we employ the Otsu binary threshold [24], which provides the min Value and max Value thresholds automatically.

Subsequently, the resulting output image undergoes black and white pixel frequency calculation. If the frequency of white pixels falls below 30, it indicates absence of information; otherwise, information is deemed present. Iterative experimentation is used to determine the threshold value of 30, which is a hyper-parameter that is customized for the Pascal VOC-2007 dataset.

To optimize small object detection, anchor boxes are arranged in descending order, starting with larger-scale boxes before progressing to smaller ones. This hierarchical arrangement obviates the need to process smaller-scale anchor boxes if information is absent from larger-scale counterparts. This streamlined approach reduces the computational overhead associated with predicting the presence of information within anchor boxes, thereby enhancing efficiency.

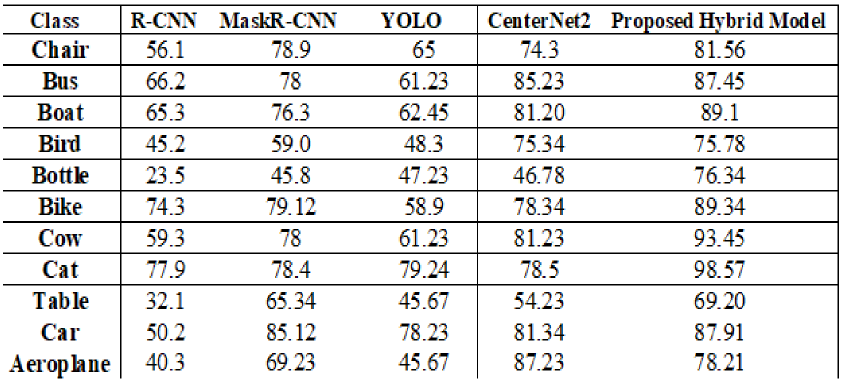
EXPERIMENTAL RESULTS

The result section highlights the outcomes of training various object detection models, focusing on comparing multiple algorithms such as R-CNN, Faster R-CNN, YOLO, and SSD. These models are designed to detect objects in images in a single pass and are suitable for deployment on mobile devices. Among these, Faster R-CNN stands out for its superior accuracy, particularly in multi-stage object detection scenarios.

The comparative analysis involved training Faster R-CNN, YOLO, and SSD algorithms using the Custom Dataset. The training was conducted on Google Collaborators Notebooks using the Tensor Flow Object Detection API and Robo flow. To visualize the training and evaluation metrics, Tensor board, an interactive tool by Tensor Flow, was employed. The graphical representation of the results was crafted using MATLAB 2020a and Python.

For the execution of the experiments, a Windows 11-based system with 16 GB RAM and a GTX GPU was utilized, ensuring sufficient computational resources for training and evaluation tasks. These specifications contribute to the reliability and efficiency of the experimentation process, facilitating meaningful comparisons between the different object detection algorithms (Table 1).

Table1: Comparative Analysis of the Mathematical results



Class	R-CNN	MaskR-CNN	YOLO	CenterNet2	Proposed Hybrid Model
Chair	56.1	78.9	65	74.3	81.56
Bus	66.2	78	61.23	85.23	87.45
Boat	65.3	76.3	62.45	81.20	89.1
Bird	45.2	59.0	48.3	75.34	75.78
Bottle	23.5	45.8	47.23	46.78	76.34
Bike	74.3	79.12	58.9	78.34	89.34
Cow	59.3	78	61.23	81.23	93.45
Cat	77.9	78.4	79.24	78.5	98.57
Table	32.1	65.34	45.67	54.23	69.20
Car	50.2	85.12	78.23	81.34	87.91
Aeroplane	40.3	69.23	45.67	87.23	78.21

Here, the table 2 represents the mathematical parameters processing of the Convolutional Neural Network architecture's different layers when the image process through the model. Here the First column represents the Input stages and the processing stages. In our work, we have proposed the 5 Convolutional layers and 2 pooling layers for the down samples of the features. RoI Align represents the Region of interest to draw the bounding box on the target area. We have used 3 fully connected layers represented by FC6, FC7 and FC8. Kernel Size simply is the filter which is a small matrix that slides over the input image during the convolution operation. Its size determines the receptive field: how much of the input it "sees" at once. Common kernel sizes include 1x1, 3x3, 5x5, or 7x7. But here for the optimization we have proposed 3x3 filter size in our work since larger kernels capture more complex features, but they also increase computational cost and smaller kernels are computationally efficient but may miss fine details. Padding is the process of adding extra pixels around the input image before applying convolution. It helps preserve spatial dimensions and prevents the output feature map from shrinking. Stride determines how much the kernel moves (slides) across the input image. A stride of 1 means the kernel moves one pixel at a time. In our work, we have used a mixture of 1 and 2 stride for better information retain. The output size represents the Image pixel at each stage. As a raw image, when we input the image which is having a large number of pixels needs to be down sampled after every stage processing provided the information should be retained and finally after the fully connected layers which combine all the neurons from the different processing stages to get the final output (Figure 3 and 4).

Table2: Processing of the Convolutional layers

Name	Kernel Number	Kernel Size	Stride	Padding	Parameters	Output Size
Input 1	-	-	-	-	-	1x3x1024x768
Input 2	-	-	-	-	-	64x4
Conv 1-1	64	3x3	1	1	1792	1x64x1024x768
Conv 1-2	64	3x3	1	1	36928	1x64x1024x768
Conv 1-3	64	3x3	1	1	36928	1x64x512x384
Pool 1	-	3x3	2	1	-	1x64x257x193
Conv2-1	128	3x3	1	1	73856	1x128x257x193
Conv2-2	256	3x3	2	1	295168	1x256x129x97
Pool2	-	3x3	2	1	-	1x256x129x97
Conv3	384	3x3	1	1	885120	1x384x65x49
Conv4	384	3x3	1	1	1327488	1x384x65x49
Conv5	256	3x3	1	1	884992	1x256x65x49
RoI Align	-	-	-	-	-	64x256x6x6
FC6	-	-	-	-	9438208	64x1024
FC7	-	-	-	-	4100	64x4
FC8	-	-	-	-	2050	64x2

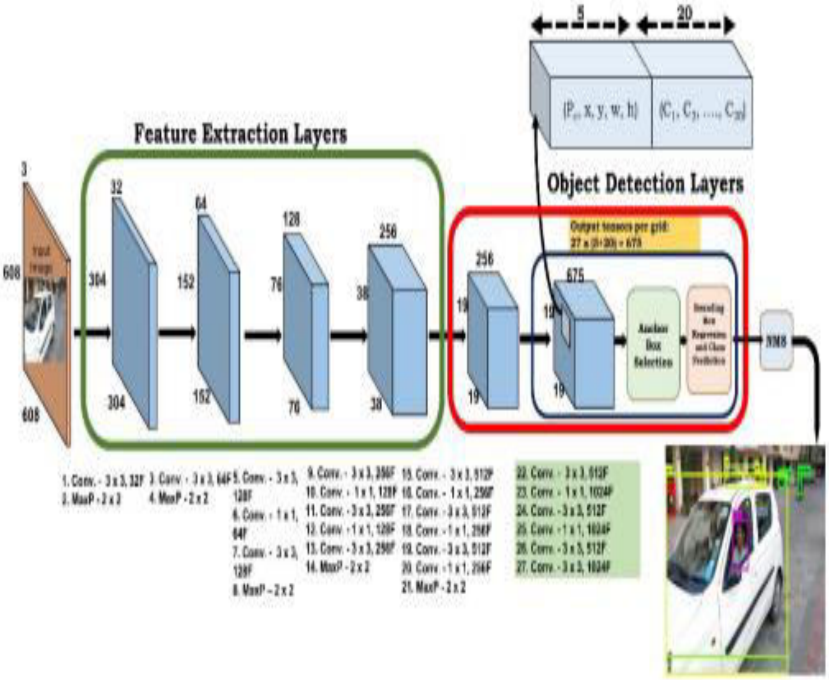
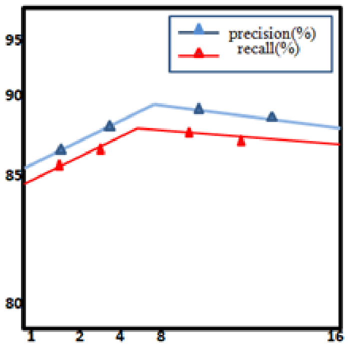
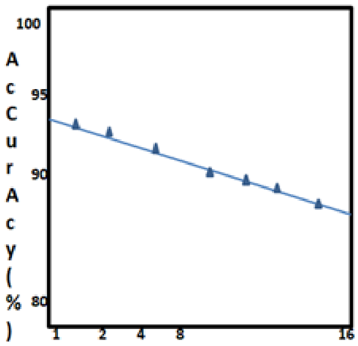


Figure 3. Output of the Detected Objects





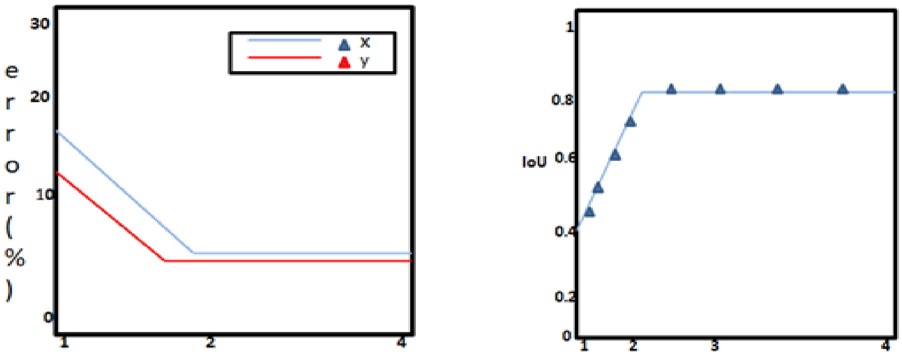


Figure 4. Graphical Representation of the Results

A predicted bounding box's accuracy is evaluated according to how closely its Intersection over Union (IoU) overlaps with the ground truth bounding box. This is done in accordance with the ILSVRC, which is the accepted method for measuring detection results. Other methods characterized by hand-designed features, exhibits decreased precision and recall compared to the other two methods due to its sensitivity to the contrast between ships and the background. Conversely, YOLOv2, leveraging Convolutional Neural Networks (CNNs) for object detection in complex environments, surpasses technique [14] in detecting adjacent targets. In contrast, our method excels in both recall and precision metrics. Furthermore, we compute the other mathematic parameters also for each approach, considering only accurately predicted bounding boxes in the calculation.

CONCLUSION

Intoday'sworld,advancementintechnologylikemachinelearning,artificiallearning, and deep learning is becoming a solution to the real problems. Convolutional neural networks (CNNs) with deep learning are one of these technologies that has greatly advanced the field of object detection and classification problems.This paper is all about how are objects detection problems is getting resolved withbetter accuracy in no time using CNN model that too with increased accuracy.We have studied various literatures and then we have projected our own algorithm which is inspired by Convolutional neuralnetwork. The hybrid proposed model promises the better results in terms of various mathematical as well as qualitative manner. In the proposed work, we have bounding box the cluttered data taken from various different sources based on real time and performed various algorithm, the mathematical tables shows the various results obtained from the different algorithm. The proposed model shows the better accuracy in terms of enhancement of other parameters also. To improve the work in the future the optimization algorithm may be implemented to address the problem of early convergence.

REFERENCES

[1] Bloisi, Domenico, and Luca Iocchi. "Argos—A video surveillance system for boattrafficmonitoringinVenice."InternationalJournalofPatternRecognitionandArtificialIntelligence23.07 (2009): 1477-1502.

[2] Fefilatyev, Sergiy, et al. "Detection and tracking of ships in open sea with rapidlymovingbuoy-mountedcamerasystem." Ocean Engineering54(2012): 1-12.

[3] Kazemi, Samira, et al. "Open data for anomaly detection in maritime surveillance."ExpertSystemswithApplications40.14(2013): 5719-5729.

[4] Frost,Duncan,andJules-RaymondTapamo."Detectionandtrackingofmovingobjectsinamaritimeenvironmentusinglevelsetwithshapepriors. "EURASIPJournalon Imageand Video Processing2013.1 (2013):42.

[5] Bloisi,Domenico,etal."Automaticmaritimesurveillancewithvisualtargetdetection." Proc.oftheInternational DefenseandHomeland SecuritySimulationWorkshop(DHSS).2011.

[6] Schweitzer, Haim, Rui Deng, and Robert Finis Anderson."A dual-bound algorithmfor very fast and exact template matching." IEEE transactions on pattern analysis andmachineintelligence33.3 (2011): 459-470.

[7] Blaschke, Thomas. "Object based image analysis for remote sensing." ISPRS journalofphotogrammetryandremotesensing65.1 (2010): 2-16.

[8] Porathe, Thomas, Johannes Prison, and Yemao Man. "Situation awareness in remotecontrol centres for unmanned ships." Proceedings of Human Factors in Ship Design &Operation,26-27 February 2014,London,UK.2014.

[9] Fiorini, Michele, and Jia-Chin Lin. Clean Mobility and Intelligent Transport Systems.Vol.1.IET, 2015.

[10] Robert-Inacio, F., A. Raybaud, and E. Clement. "Multispectral target detection andtracking for seaport video surveillance." Proc Image Vision Comput New Zealand.Hamilton,NewZealand.December(2007):5-7.

[11] Hampapur, Arun, et al. "Smart video surveillance: exploring the concept of multiscalespatiotemporaltracking."IEEE SignalProcessingMagazine22.2(2005):38-51.

[12] Prasad, Dilip K., et al. "Challenges in video based object detection in maritime scenario using computer vision." arXiv preprint arXiv: 1608.01079 (2016).

[13] Chen, Chuan Cheng. Attenuation of electromagnetic radiation by haze, fog, clouds,andrain. Vol.1694. No. PR. RANDCORPSANTAMONICACA,1975.

[14] Szpak, Zygmunt L., and Jules R. Tapamo. "Maritime surveillance: Tracking shipsinside a dynamic background using a fast level-set." Expert systems with applications38.6(2011): 6669-6680.

[15] Elfes, Alberto. "Sonar-based real-world mapping and navigation." IEEE Journal onRoboticsandAutomation3.3 (1987):249-265.

[16] Hansen,RoyEdgar."Syntheticaperturesonartechnologyreview."MarineTechnologySocietyJournal47.5 (2013): 117-127.

[17] Horne, John K. "Acoustic approaches to remote species identification: a review."Fisheriesoceanography9.4 (2000): 356-371.

[18] Hayes, Michael P., and Peter T. Gough. "Synthetic aperture sonar: a review of currentstatus."IEEEJournal ofOceanicEngineering34.3 (2009): 207-224.

[19] Ward, K. D., C. J. Baker, and S. Watts. "Maritime surveillance radar. Part 1: Radarscattering from the ocean surface." IEE Proceedings F (Radar and Signal Processing).Vol.137. No. 2.IETDigitalLibrary,1990.

[20] Watts,S.,C.J.Baker,andK.D.Ward."Maritimesurveillanceradar.Part2:Detectionperformancepredictioninseaclutte r."IEEProceedingsF(RadarandSignalProcessing). Vol.137. No.2.IET DigitalLibrary,1990.

[21] Vicen-Bueno, R., etal. "Ship detection by differentdata selection templates andmultilayerperceptronsfromincoherentmaritime radardata."IET radar, sonar&navigation5.2 (2011): 144-154.

[22] Pasquariello, Guido, etal. "Automatic target recognition for navaltraffic controlusingneural networks."Imageand visioncomputing16.2 (1998):67-73.

[23] Sevgi,Levent,AnthonyPonsford,andHingC.Chan."Anintegratedmaritimesurveillancesystembasedonhigh-frequencysurface-waveradars.1.Theoreticalbackground and numerical simulations." IEEE antennas and propagation magazine43.4(2001): 28-43.

[24] Ponsford, Anthony M., LeventSevgi, and Hing C. Chan. "An integrated maritimesurveillancesystembasedonhigh-frequencysurface-waveradars.2.Operationalstatusandsystemperformance."IEEEAntennasandPropagationMagazine43.5(2001):52-63

[25] X. Li and S. Wang, “Object detection using convolutional neural networks in a coarse-to-fine manner,” IEEE Geosci. Remote Sens. Lett., vol. 14, no. 11, pp. 2037–2041, Nov. 2017.

[26] Q. An, Z. Pan, and H. You, “Ship detection in Gaofen-3 SAR images based on sea clutter distribution analysis and deep convolutional neural network,” Sensors, vol. 18, no. 2, p. 334, 2018.

[27] D. Cheng, G. Meng, S. Xiang, and C. Pan, “FusionNet: Edge aware deep convolutional networks for semantic segmentation of remote sensing harbor images,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 10, no. 12, pp. 5769–5783, 2017.

[28] V. Nair and G. E. Hinton, “Rectified linear units improve restricted Boltzmann machines,” in Proc. 27th Int. Conf. Mach. Learn., 2010, pp. 807–814.

[29] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in Proc. 13th Int. Conf. Artif. Intell. Stats., 2010, pp. 249–256.

[30] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Region-based convolutional networks for accurate object detection and segmentation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 1, pp. 142–158, Jan. 2016.

[31] M. Najibi, M. Rastegari, and L. S. Davis, “G-CNN: An iterative grid based object detector,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 2369–2377.

[32] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” in ´ Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 2980–2988.

[33] Y. Jia et al., “Caffe: Convolutional architecture for fast feature embedding,” inProc. 22nd ACM Int. Conf. Multimedia, ACM, 2014, pp. 675–678.

[34] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 6517–6525.

[35] L. Zhang, L. Lin, X. Liang, and K. He, “Is faster r-cnn doing well for pedestrian detection?” in ECCV, 2016.

[36] Y. Tian, P. Luo, X. Wang, and X. Tang, “Deep learning strong parts for pedestrian detection,” in ICCV, 2015.

[37] J. Liu, S. Zhang, S. Wang, and D. N. Metaxas, “Multispectral deep neural networks for pedestrian detection,” arXiv:1611.02644, 2016.

[38] Y. Tian, P. Luo, X. Wang, and X. Tang, “Pedestrian detection aided by deep learning semantic tasks,” in CVPR, 2015.

[39] X. Du, M. El-Khamy, J. Lee, and L. Davis, “Fused dnn: A deep neural network fusion approach to fast and robust pedestrian detection,” in WACV, 2017.

[40] Q. Hu, P. Wang, C. Shen, A. van den Hengel, and F. Porikli, “Pushing the limits of deep cnns for pedestrian detection,” IEEE Trans. Circuits Syst. Video Technol., 2017.

[41] D. Tome, L. Bondi, L. Baroffio, S. Tubaro, E. Plebani, and D. Pau, ´ "Reduced memory region based deep convolutional neural network detection," in ICCE-Berlin, 2016.

[42] J. Hosang, M. Omran, R. Benenson, and B. Schiele, "Taking a deeper look at pedestrians," in CVPR, 2015.