WJCMS

Wasit Journal of Computer and Mathematics Science

Journal Homepage: https://wjcm.uowasit.edu.iq/index.php/WJCM

e-ISSN: 2788-5879 p-ISSN: 2788-5879

Object Detection Techniques: A Review

Widad K. Mohammed[1,*], Haider D. A. Jabar[1], Mohammed A. Taha[2] and Saif Ali Abd Alradha Alsaidi[3]

[1]Ministry of Education, Baghdad, 10011, Iraq
[2]Ministry of Education, Babylon Education Directorates, Babylon, 51005, Iraq
[3]College of Education for Pure science,Wasit university,Wasit,52001,,Iraq

*Corresponding Author: Widad K. Mohammed

DOI: https://doi.org/10.31185/wjcms.165
Received: May 2023; Accepted: September 2023; Available online: September 2023

ABSTRACT: Humans can understand their surroundings clearly because they regularly notice objects in their environment. It is essential for the machine to perceive the surroundings similarly to how humans do in order to make it autonomous and capable of navigating in the human world. The machine can assess its surroundings and identify objects using object detection. This can simplify a number of tasks and enable the machine to recognize its surroundings. Making bounding boxes that surround the objects is essentially how object detection systems work to locate objects in an image. Object detection has applications such as autonomous robot navigation, surveillance, face detection, and vehicle navigation, etc. In this article surveyed and studied Object detection algorithms.

Keywords: object detection, R-CNN, Fast R-CNN, Faster R-CNN, Mesh R-CNN, Mask R-CNN.

1. INTRODUCTION

A computer vision approach called object detection makes it easier to identify the type and location of things in an image or video. This technology makes it feasible to recognize every object in an image or video and identify its exact location. [1]

Before the advent of deep learning in 2013, all object detection was carried out using traditional machine learning techniques. The histogram of directional gradients, the scale-invariant feature transform (SIFT), and the viola-jones object detection method are examples of common ones [2] [3]. These are considerably outperformed by the deep learning-based algorithms used today, which are helpful in a variety of applications such as anomaly detection, self-driving cars, surveillance systems, and facial recognition systems. RetinaNet, YOLO (You Only Look Once) [4], CenterNet, SSD (Single Shot Multibox Detector) [5], and Region Proposal are examples of neural networks (R-CNN, Fast-RCNN, Faster RCNN, Cascade R-CNN) Deep learning-based techniques employ an architecture for identifying object categories and detecting object features. [6]

The ability of object detection algorithms to recognize and classify objects in an image or video has made them increasingly important in computer vision applications [7]. Some of the most popular and frequently used object detection algorithms include the R-CNN, Fast R-CNN, Faster R-CNN, Mask R-CNN, and Mesh R-CNN [8].

In this study, we will review the importance of these algorithms, their potential limitations and challenges, the accuracy with which they operate and their effects on the field of computer vision are introduced. The rest of this paper is ordered as; Sect.2 will set the literature review, Sect. 3 will introduce the object detection algorithms used to carry out the survey, Sect. 4 which clarify the Comparison and Performance Analysis of these algorithms followed by Sect. 5 will be a discussion about the limitations and challenges and conclusion and future scope are included in the final section.

2. LITERTURE SURVEY

Shanlan Nie et al. [9] suggested using Mask R-CNN as a method to find inshore ships. The technique is tested using Google Earth data to show that it can recognize both battleships and merchant ships, and the framework is incorporated by Soft-NMS for better detection.

Zhen Yang et al. [10] presented an automatic inspection system that makes use of Mask R-CNN to increase tower crane drivers' operational safety. When using the MASK R-CNN method for image recognition, the tower crane camera captures both video and still images. Additionally, the detected mask layers' RGB color extraction was done to retrieve the pixels. Workers' coordinates, risk zones, pixel transformations, and the actual safety distance

Madhusri Maity et al. [11] introduced an evaluation of vehicle identification and tracking techniques using Faster Region-based Convolutional Neural Network(Faster R-CNN) with You Look Only Once (YOLO) to reduce fatal accidents mostly brought on by driver negligence and inadequate lighting or poor visibility in bad weather.

Jeremiah w. Johnson [12] demonstrated that a variety of microscope images of cell nuclei may be automatically segmented with excellent effectiveness using Mask-RCN.

Beibei Xu et al. [13] demonstrated the application of the cutting-edge instance segmentation framework with mask R-CNN under diverse settings to apply cattle counting in intensive housing, extensive production meadows, and feedlots.

Kang Zhao et al. [14] provided a technique for localizing each building polygon in the specified area that combines building boundary regularization satellite images with Mask R-CNN, and it was found that the proposed approach and Mask R-CNN produce performance that is nearly equivalent in terms of completeness and accuracy, which immediately relates to several cartographic and technical applications.

Dongbo Zhao and Hui Li [15] studied the use of R-CNN, Fast R-CNN, and Faster R-CNN based on region proposal in vehicle target detection and provided a summary of the general design of the vehicle detection algorithm. Additionally, it concentrates on the examination of the Faster R-CNN detection algorithm's non-maximum suppression technique and shared convolution layer analysis.

3. OBIECT DETECTION ALGORITHMS

Object detection is a fascinating field that is attracting a lot of attention for both academic and business reasons. Due to advancements in the most recent hardware and processing resources, advancements in this field were fast and ground-breaking [16] .

Object detection is a subset of object recognition, as object detection specifically involves locating and identifying objects within an image or video, while object recognition also includes the ability to classify the objects that have been detected. CNN-based deep learning object detection systems include R-CNN. Fast, Faster, and Mask R-CNN is a variation of the methodology based on various applications and requirements [17].

3.1 REGION-BASED CONVOLTIONAL NEURAL NETWORKS (R-CNN)

The R-CNN family of machine learning models, developed by Ross Garthic et al., is utilized in a variety of computer vision applications, particularly in object detection. R-CNN takes an input image and applies the Selective Search technique to extract regions of interest (ROI). Each ROI is a rectangle that may represent the edge of an item in the input image. There could be up to 2000 ROIs, depending on the scenario. Each ROI is subsequently sent through a neural network to generate output features. A group of support-vector machine classifiers is used to analyze each ROI's output features and determine what kind of object (if any) is present therein [18]. The R-CNN would typically include the following components, As can be seen in Figure (1):

1. Input image: The image is passed into the network for object detection
2. Region Proposal Network (RPN): A network that takes the input image and generates a set of potential object regions, or "region proposals."
3. Feature extractor: A convolutional neural network (CNN), such as VGG or ResNet, that is used to extract features from the region proposals.
4. Classifier: A classifier, such as a support vector machine (SVM), that is trained to classify the regions as containing an object or not.
5. Bounding box repressor: A regress that refines the coordinates of the bounding box around the object.
6. Output: The final output of the network is a set of bounding boxes, each with an associated class label and confidence score.

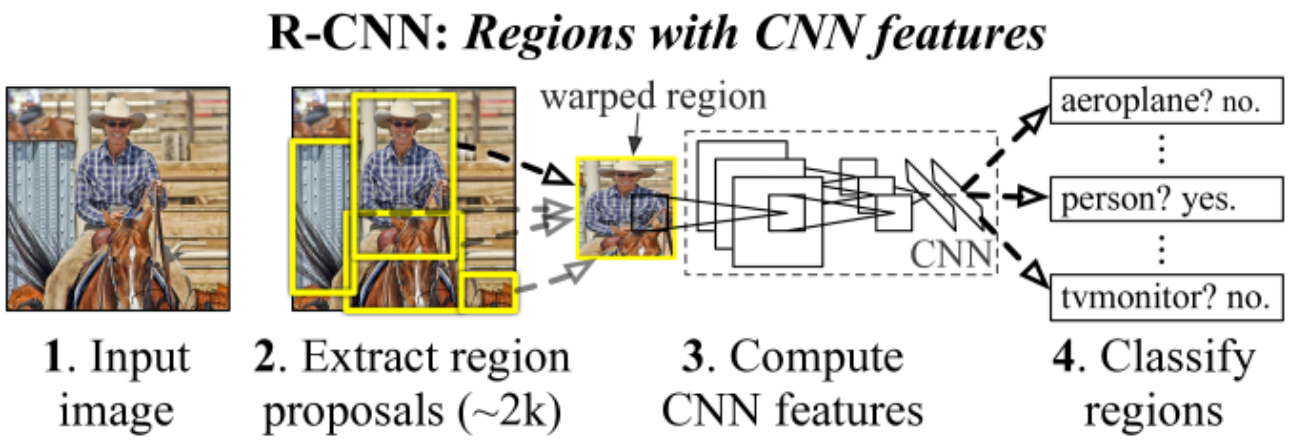

FIGURE 1. **R-CNN block diagram [18]**

3.2 FAST REGION-BASED CONVOLUTIONAL NEURAL NETWORK

Fast R-CNN is an object recognition approach that first extracts features from an entire image using a convolutional neural network (CNN), and then utilizes a region proposal network (RPN) to find areas of the image that could contain objects. These regions, also known as region of interests (RoIs), are then passed through the CNN to classify the objects within them. [19]

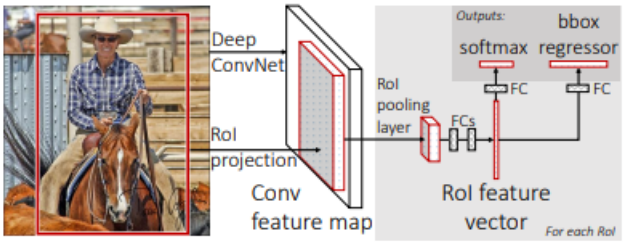

FIGURE 2. **Fast R-CNN architecture [19]**

Fast R-CNN, As can be illustrating in Figure (2) would include the following steps:

1. Input image: The image is passed into the network for object detection
2. Region proposals: A set of potential object regions, or "region proposals," are generated using a technique such as a sliding window or selective search.
3. Feature extraction: A convolutional neural network (CNN) is used to extract features from the entire input image.
4. Pooling: The features are passed through a spatial pyramid pooling layer, which partitions the feature maps into sub-regions and applies max pooling to each sub-region.
5. Classification and bounding box regression: The combined features are processed through fully connected layers to determine if a region contains an object or not and to fine-tune the bounding box's coordinates.

Fast R-CNN is faster than the original R-CNN because it shares computation of the CNN on the entire image among all the object proposals, instead of running the CNN independently on each proposal, this way it also reduces the number of parameters, and thus, it's more efficient.

3.3 FASTER REGION-BASED CONVOLUTIONAL NEURAL NETWORKS

An object detection framework called Faster R-CNN uses both a convolutional neural network (CNN) and a region proposal network (RPN) [20]. The CNN is applied to each region to classify and identify the exact location of each object

after the RPN creates probable object regions in an image. Faster R-CNN can operate faster than the original R-CNN while still retaining a high level of accuracy because of the usage of an RPN. A two-stage detection framework is used, with the first stage generating region proposals and the second stage classifying and locating the object using CNN. [21]

It uses a unified model consisting of the region proposal network (RPN) Faster R-CNN incorporates the ROI generation inside the neural network itself, unlike Fast R-CNN, which employed Selective Search to produce ROIs. FAST R-CNN with shared convolutional feature layers is the next step. [11] [22]

Instead of using a selective search algorithm on a feature map to identify suggested regions, a separate grid is used to predict region proposals. The predicted suggested region is then resampled using a pooling layer ROI, which is then used to classify the image within the suggested region and predict displacement values in the bounding boxes steps. This is why Faster R-CNN is faster than FAST R-CNN. This is why Faster R-CNN is faster than FAST R-CNN., as shown in Figure (3).

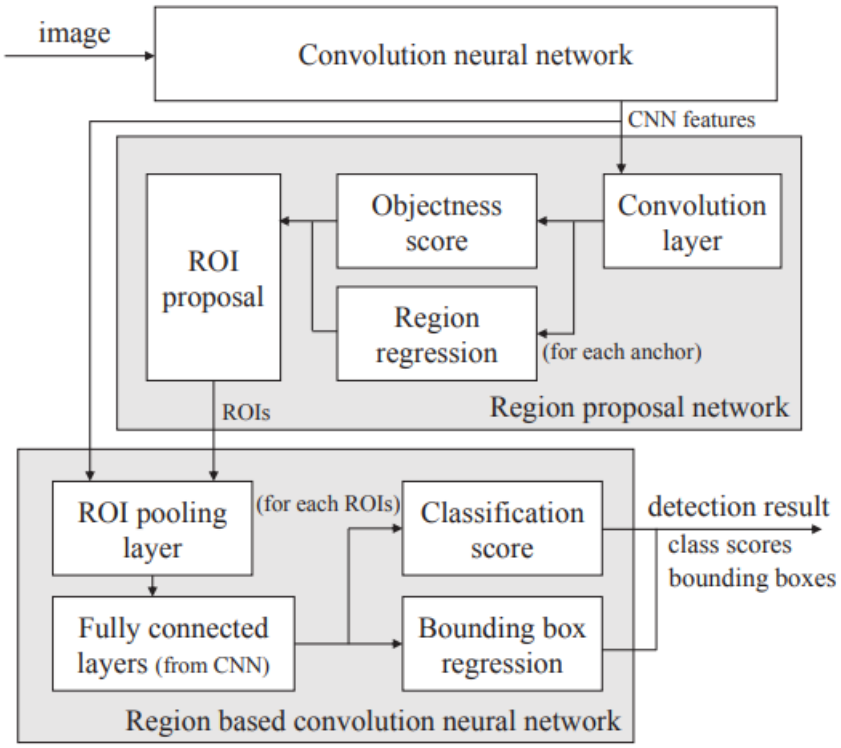

FIGURE 3. The flowchart of faster R-CNN [23]

Squares anchor boxes are just references, marked with different proportions and scales in order to accommodate different types of objects, objects that are elongated like edges. Remove similar bounding boxes results that match the 'Object' class predictions. (Faster R-CNN) would include the following steps:

1. Image input: For object detection, the image is transmitted to the network.
2. Feature extraction: To extract features from the full input image, a convolutional neural network (CNN), such as VGG or ResNet, is utilized.
3. Region Proposal Network (RPN): a small network that creates a list of probable object areas, or "region proposals," using CNN's feature maps as a starting point.
4. RoI pooling: A RoI (Region of Interest) pooling layer divides the feature maps into sub-regions that correspond to the region proposals and applies maximum pooling to each sub-region before the feature maps are passed through.
5. Classification and Bounding box regression: The pooled features are passed through fully connected layers to classify the regions as containing an object or not and refine the coordinates of the bounding box around the object.

6. Output: The final output of the network is a set of bounding boxes, each with an associated class label and confidence score.

3.4 MASK REGION-BASED CONVOULATIONAL NEURAL NETWORK

For each object instance it successfully locates in an image, a Framework for object instance segmentation method simultaneously builds a top-notch segmentation mask. Combining the existing branch for bounding box recognition with the fresh branch for object mask prediction. 5 frames per second is used to run faster R-CNN. It is also straightforward to adapt Mask R-CNN to different tasks. Additionally, ROI Align, a new technique that can represent fractions of a pixel, took the place of ROI Pooling. [24].

The method of region proposals in Mask R-CNN is a two-stage process. First, a set of potential object regions, or "region proposals," are generated using a technique such as a sliding window or selective search. These region proposals are then passed through a convolutional neural network (CNN) to classify and refine the regions. The final output of this stage is a set of high-confidence region proposals that are likely to contain objects. In the second stage, a fully convolutional network (FCN) is applied to each region proposal to generate a segmentation mask for the object within that region [25] ,As shown in figure (4).

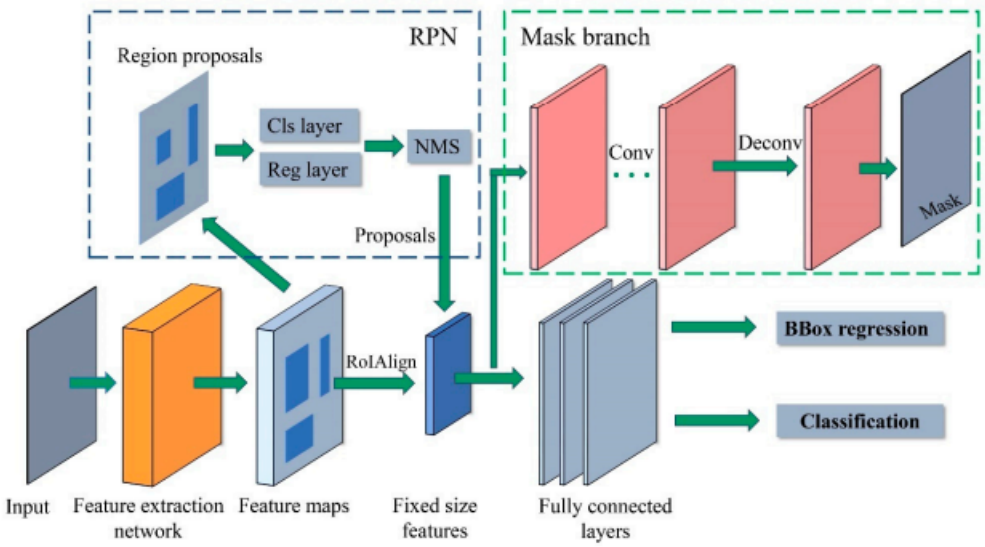

FIGURE 4. Architecture of Mask R-CNN [26]

A feature extractor, a region proposal network (RPN), and a classification and regression network are some of the typical parts of a mask R-CNN. Mask R-CNN typically looks like this:

1. Input image: The input to the network is an image.
2. Feature extraction: The image is passed through a feature extractor (such as a ResNet or a VGG network) to extract features from the image.
3. Region Proposal Network (RPN): To create region proposals, the RPN is fed the feature map from the feature extractor. The RPN creates a set of region suggestions using anchors and a sliding window method
4. Proposal classification: Each region proposal is classified as "object" or "background".
5. Proposal regression: Bounding boxes for each region proposal are refined using bounding box regression.
6. RoI Align: The feature map is aligned with the region of interest (RoI) to extract features for the RoI.
7. Class prediction: The features for the RoI are passed through a fully connected layer to predict the class of the object in the RoI.
8. Bounding box regression: To improve the boundary area for the item within the RoI, the characteristics for the RoI are additionally sent via a fully connected layer.

9. Output: For each object in the image, the Mask R-CNN produces a set of foretasted categories.

3.5 MESH REGION-BASED CONVOULATIONAL NEURAL NETWORK

A mesh representation of the item is used by the three-dimensional (3D) object recognition technique known as Mesh R-CNN to increase its resilience and accuracy. The primary concept underlying Mesh R-CNN is to represent an object using a 3D mesh, that is made up of a group of connected triangles. Compared to conventional 2D object detection techniques, that depend on points, this mesh representation accurately and in-depth depicts the form and structure of the item [27].

Several parts make up the Mesh R-CNN model, comprising a 2D item detector, a mesh generating system, and a mesh improvement network. The item in the image is found using the 2D object detector, which also creates a rough 3D bounding box. Then, using this bounding box, the mesh generation network creates a crude mesh depiction of the object. To provide a more precise and thorough representation of the object, a mesh improvement network polishes the mesh [28].

As it can handle objects with complicated shapes and structures and be used for a variety of tasks, including self-driving cars, robotics, and virtual reality, Mesh R-CNN is a promising method for 3D object detection [29].

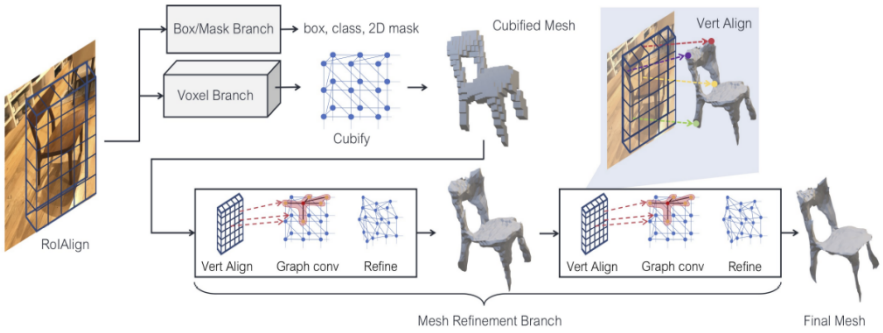

FIGURE 5. Mesh R-CNN [29]

The Mesh R-CNN typically includes six steps, as shown in figure (5)

1. Input: An image or a point cloud.
2. Backbone: A convolutional neural network (CNN) that extracts features from the input.
3. Region Proposal Network (RPN): A network that proposes regions of interest (ROIs) in the image or point cloud.
4. Region-of-Interest (ROI) alignment: The features of the proposed ROIs are extracted and aligned.
5. Detection and semantic segmentation: The aligned features are fed into separate branches for object detection and semantic segmentation.
6. Output: The final output includes bounding boxes and class labels for the detected objects, as well as a semantic segmentation map for the entire image or point cloud.

4. COMPARIOSON AND PERFOMANCE ANALYSIS

R-CNN, Mesh R-CNN, Fast R-CNN, Faster R-CNN, and Mask R-CNN are all popular object detection algorithms used in computer vision. These algorithms differ in their approach to detecting objects within an image, with some focusing on speed and others on accuracy. A comparison between algorithms in computation time, Method of Region proposals and Prediction speed is illustrated, see Table 1, while the techniques that used in each algorithm, see Table 2.

In general, Mesh R-CNN is considered to be the most recent and advanced algorithm, while R-CNN, Fast R-CNN, and Faster R-CNN are considered to be older and less sophisticated algorithms. Mask R-CNN is an extension of Faster R-CNN that adds instance segmentation capabilities. Performance analysis of these algorithms would involve comparing their accuracy, speed, and memory usage in various object detection tasks.

Object detection algorithms are widely used in various applications However, these algorithms are not without limitations and challenges. Some of the common limitations and challenges associated with object detection algorithms include

Table 1. A comparison between algorithms in computation time, Method of Region proposals and Prediction speed.

Properties	R-CNN	Fast R-CNN	Faster R-CNN	Mask R-CNN	Mesh R-CNN
Computation Time	High computation time	High computation time	Low computation time	The computation time depends on several factors, including the complexity of the input image, the number of object instances in the image, and the hardware being used.	The complexity of the input mesh, the number of faces, and the number of vertices all affect how long Mesh R-CNN takes to compute, the number of regions being segmented, and the computational resources available.
Method of Region proposals	Selective Search	Selective Search	RPN	RPN	identify and extract regions of interest (ROIs) in 3D point clouds
Prediction speed	(40-50)seconds	2 seconds	0.2 seconds	5fps [24]	The prediction speed of Mesh R-CNN can vary depending on the specific implementation and hardware it is running on.

Table 2. The techniques of each algorithm.

	ROI	RPN	Selective Search	ROI Align	ROI Pooling	voxel prediction	mesh refinement
R-CNN	✓		✓				
Fast R-CNN			✓		✓		
Faster R-CNN		✓					
Mask R-CNN		✓		✓			
Mesh R-CNN						✓	✓

high computational cost, difficulty in detecting small or occluded objects, and limited generalization ability. Additionally, object detection algorithms may struggle with variations in lighting, viewpoint, and object appearance, making it difficult to achieve high accuracy in real-world scenarios. The level of quality and variety of training data can also have an impact on how well object detection algorithm's function. These constraints and difficulties are illustrated in Table 3.

Table 3. Limitations and Challenges

Algorithm	Limitations and Challenges
R-CNN	1. Computation: The R-CNN model is expensive in terms of computation because it necessitates the lengthy process of executing a CNN on each region suggestion. 2. Restricted scalability: The R-CNN approach has a high computational cost and is not particularly suitable for large-scale object recognition jobs because the number of area recommendations grows as the size of the image. 3. Restricted object diversity: Because the selective search method may not produce region recommendations for these kinds of items, the R-CNN system cannot be well suited for recognizing small or heavily occluded objects. 4. R-CNN technique is not real-time; it requires some time to recognize objects in images. 5. Limited robustness: Because CNN characteristics are not invariant to changes in illumination, posture, or perspective of the items in the image, the R-CNN system is not robust to these kinds of changes. 6. Limited accuracy: Because the R-CNN model is based on area recommendations, which may or might not always include the items of interest, it is not as precise as other types of object detection methods.
Fast R-CNN	1. It necessitates an enormous quantity of storage for handling the region suggestions and feature maps, making it challenging to implement to massive data sets or systems that operate in real time. 2. To achieve outstanding results using the Fast R-CNN model, an effective RPN must be designed as the level of accuracy of the region suggestions provided by the RPN heavily influences the efficacy of the Fast R-CNN model. 3. Finding the best settings for a particular dataset might be challenging because the algorithm can be susceptible to the hyperparameters. Due to this, getting the model to perform well might be difficult, especially when using new datasets. 4. The model is dependent on the level of detail of the data that has been annotated, and effective execution with the model may be challenging if the data is badly annotated.
Faster R-CNN	1. Speed: Because the model needs a region proposal step, then an additional categorization step for each suggested region, it can be slower during the testing phase. 2. Memory: The model needs a lot of memory for training because it needs to keep the map of features for each proposed region. 3. Scale: Because the model was created to perform well with items of a specific size, it may have trouble detecting things at other scales. 4. Overfitting: If the model has not been taught with enough data, it may be vulnerable to overfitting. 5. Complexity: Because of the model architecture's complexity and difficulty in implementation, it may be difficult for researchers to experiment with different model alterations. 6. Restrictions on 2D photos: The model is only able to handle 2D images and is unable to process 3D pictures or films. 7. Limited to a single class: The model is incapable of detecting more than one class of objects in a single image and is only capable of detecting a single class of objects per image.

Continued on next page

Table 3 continued

Mask R-CNN	1. An expensive computation and can be slow to run on large images or videos. This can make it impractical for real-time applications or for use on resource-constrained devices. 2. The model relies on region proposals, which can be difficult to generate accurately, especially for small or occluded objects. Additionally, the model may struggle to detect objects with unusual or irregular shapes. 3. The model requires a large amount of labeled training data to achieve high accuracy, which can be time-consuming and expensive to collect. 4. The model is not robust to changes in lighting, viewpoint, and other variations in the image. This can make it difficult to apply the model to real-world images, which may contain significant amounts of noise or other variations. 5. The class imbalance problem can be challenging for the model, especially when there are fewer positive instances than negative examples.
Mesh R-CNN	1. Computational Complexity: As mentioned earlier, Mesh R-CNN is computationally intensive, which can make it difficult to run on resource-constrained devices or in real-time applications. 2. Limited Datasets: Mesh R-CNN effectiveness is strongly influenced by the quality of the training dataset. Currently, there are limited datasets available for training this model, which can limit its overall performance. 3. Handling Occlusions: Mesh R-CNN model may struggle with handling occlusions, where one object may be blocking the view of another object. 4. Handling Scale Variations: The model may also have difficulty handling variations in object scale, which can lead to inaccuracies in the generated meshes. 5. Handling Non-rigid objects: Generating accurate meshes for non-rigid objects such as cloth, hair, and fur, is a challenging task, and the model can struggle with it. 6. Handling Complex Scenes: The model may also have difficulty handling complex scenes with multiple objects and cluttered backgrounds.

5. CONCLUSION AND FUTURE SCOPE

We have reviewed the methods for object detection in this work, focusing on the various R-CNN algorithms. The field of object detection has witnessed various algorithms in just three years, each of which has overcome the earlier one. Due to the overlap between the suggested regions in each image and the need to repeatedly conduct the CNN computation, we found that the R-CNN basic approach was excessively slower. A second version, the Fast R-CNN, overcame this. Some of them include decreased detection accuracy and difficulty segmenting objects with wide ranges in size. Faster R-CNN then improved upon this method by employing the classifier findings instead of the selective search approach to obtain region suggestions. The addition of Pixel Level Segmentation using Mask R-CNN. Numerous applications, including face detection, inshore ship detection, on-ground airplane detection, etc., have successfully used these techniques. According to their intended use. Each of the algorithms presented offers some benefits as well as some limitations. The goal of the upcoming study is to close the methodologies' discovered gaps. Computational Complexity in Mesh R-CNN especially in real-time applications, etc.

FUNDING

None

ACKNOWLEDGEMENT

None

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

[1] M. Wu, "Object detection based on RGC mask R-CNN," *IET Image Process*, vol. 14, no. 8, pp. 1502–1508, 2020.
[2] G. A. Montazer and D. Giveki, "Content based image retrieval system using clustered scale invariant feature transforms," *Optik (Stuttg)*, vol. 126, no. 18, pp. 1695–1699, 2015.

[3] L. Shi and J. H. Lv, "Face detection system based on AdaBoost algorithm," *Appl. Mech. Mater*, vol. 380, no. 4, pp. 3917–3920, 2013.
[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit*, pp. 779–788, 2016.
[5] W. Liu, "SSD: Single shot multibox detector," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, pp. 21–37, 2016.
[6] B. Mahaur, N. Singh, and K. K. Mishra, "Road object detection: a comparative study of deep learning-based algorithms," *Multimed. Tools Appl*, vol. 81, no. 10, pp. 14247–14282, 2022.
[7] N. Yadav and U. Binay *Comparative Study of Object Detection Algorithms*, pp. 586–591, 2017.
[8] L. Du, R. Zhang, and X. Wang, "Overview of two-stage object detection algorithms," *J. Phys. Conf. Ser*, vol. 1544, no. 1, 2020.
[9] S. Nie, Z. Jiang, H. Zhang, B. Cai, and Y. Yao, "Inshore ship detection based on mask r-cnn," *Int. Geosci. Remote Sens. Symp*, pp. 693–696, 2018.
[10] Z. Yang, Y. Yuan, M. Zhang, X. Zhao, Y. Zhang, and B. Tian, "Safety distance identification for crane drivers based on mask r-cnn," *Sensors (Switzerland)*, vol. 19, no. 12, 2019.
[11] M. Maity, S. Banerjee, and S. S. Chaudhuri, "Faster R-CNN and YOLO based Vehicle detection: A Survey," *Proc. - 5th Int*, vol. 2021, pp. 1442–1447, 2021.
[12] J. W. Johnson *Adapting Mask-RCNN for Automatic Nucleus Segmentation*, pp. 1–7, 2018.
[13] B. Xu, "Automated cattle counting using Mask R-CNN in quadcopter vision system," *Comput. Electron. Agric*, vol. 171, pp. 105300–105300, 2020.
[14] K. Zhao, "Deep Learning-based Building Labeling 3.1. Mask R-CNN for Initial Polygon Generation," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Work*, pp. 247–251, 2018.
[15] D. Zhao and H. Li, "Forward vehicle detection based on deep convolution neural network," *AIP Conf. Proc*, vol. 2073, 2019.
[16] R. Padilla, S. L. Netto, E. A. B. Da, and Silva, "A Survey on Performance Metrics for Object-Detection Algorithms," *Int. Conf. Syst. Signals, Image Process*, pp. 237–242, 2020.
[17] Z. Zou, Z. Shi, Y. Guo, and J. Ye *Object Detection in 20 Years: A Survey*, 2019.
[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit*, pp. 580–587, 2014.
[19] R. Girshick, "Fast R-CNN," *Proc. IEEE Int. Conf. Comput. Vis*, pp. 1440–1448, 2015.
[20] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High Performance Visual Tracking with Siamese Region Proposal Network," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit*, pp. 8971–8980, 2018.
[21] A. Salvador, X. Gir, and F. Marqu, "Faster R-CNN Features for Instance Search," *IEEE Xplore*, pp. 9–16, 2013.
[22] Y. W. Chao, S. Vijayanarasimhan, B. Seybold, D. A. Ross, J. Deng, and R. Sukthankar, "Faster R-CNN for Temporal Action Localization," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit*, pp. 1130–1139, 2018.
[23] C. Lee, H. J. Kim, and K. W. Oh, "Comparison of faster R-CNN models for object detection," *Int. Conf. Control. Autom. Syst*, vol. 0, no. Iccas, pp. 107–110, 2016.
[24] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell*, vol. 42, no. 2, pp. 386–397, 2020.
[25] T. Vu, T. Bao, Q. V. Hoang, C. Drebenstetd, P. V. Hoa, and H. H. Thang, "Measuring blast fragmentation at Nui Phao open-pit mine, Vietnam using the Mask R-CNN deep learning model," *Min. Technol. Trans. Inst. Min. Metall*, vol. 130, no. 4, pp. 232–243, 2021.
[26] Z. Yang, R. Dong, H. Xu, and J. Gu *Instance segmentation method based on improved mask R-cnn for the stacked electronic components*, vol. 9, 2020.
[27] Z. Zhou, Q. Lai, S. Ding, and S. Liu, "Joint 2D object detection and 3D reconstruction via adversarial fusion mesh r-cnn," *Proc. - IEEE Int. Symp. Circuits Syst*, pp. 0–4, 2021.
[28] Y. Wu, "Monocular Instance Level 3D Object Reconstruction based on Mesh R-CNN," *Proc. - 2020 5th Int. Conf*, vol. 2020, pp. 1–6, 2020.
[29] G. Gkioxari, F. Ai, . . M. R-Cnn, and I. Xplore pp. 9785–9795, 2020.