Mask R-CNN: A Comparative Study on Improvements in Object Detection and Segmentation

Shashwat Shukla[1], Narayana Darapaneni[2], Anwesh Reddy Paduri[3], Sudha B G[4], Abhishek K[5], Arun Kumar BS[6], Chandrashekar KP[7], Deepak Kumar T P[8], Prajwal S Shetty[9], Rahul Kumar Verma[10]

IIIT Lucknow, India [1,10]

Northwestern University/Great Learning, US [2]

Great Learning, Bangalore, India [3,4]

PES University, Bangalore, India [5–9]

Corresponding author: Anwesh Reddy Paduri, Email: anwesh@greatlearning.in

Over the years there has been an increasing demand for image recognition as the world is moving towards a digital space. With the increasing demands, the application of Mask RCNN and expanded algorithms based on segmentation and YOLO have seen a major rise in the last 5 years. So the accuracy of the models has improved slightly since most projects take different approaches to the different datasets and have different metrics of Evaluation. By cross-referencing these approaches, a trend is observed that leads to higher success with the implementation of models using the hyperparameters and extra layers that have been added to the Mask RCNN in sequences. In this paper, the different approaches taken by different researchers are explored to understand how the implementations have progressed over the last half of the decade. In our research, we have studied analyzed 50 papers and found that the majority of papers were using the COCO dataset for training purposes with a specified set of hyper-parameters to measure the accuracy, performance, and memory consumption. The experiment findings were presented to suggest suitable RCNN architecture based on application or hardware attributes.

Keywords: Object Detection, Segmentation, Fast RCNN, Faster RCNN, Mask R-CNN

2023. In Satyasai Jagannath Nanda & Rajendra Prasad Yadav (eds.), *Data Science and Intelligent Computing Techniques*, 93–102. Computing & Intelligent Systems, SCRS, India. https://doi.org/10.56155/978-81-955020-2-8-9

Shashwat Shukla[1], *Narayana Darapaneni*[2], *Anwesh Reddy Paduri*[3], *Sudha B G*[4], *Abhishek K*[5], *Arun Kumar BS*[6], *Chandrashekar KP*[7], *Deepak Kumar T P*[8], *Prajwal S Shetty*[9], *Rahul Kumar Verma*[10]

1 Introduction

Image recognition using computer vision has been under development for a few years now. One of the prominent algorithms used during those times was CNN and at that time they could recognize handwritten digits. It was mostly used in the postal sector to read zip codes, pin codes, etc. The important thing to remember about any deep learning model is that it requires a large amount of data to train and requires a lot of computing resources. This was a major drawback for CNNs at that period and hence CNNs were only limited to the postal sectors.

As the demand and use case for image recognition increased, Researchers started exploring and developing new methodologies to tackle various scenarios. Different approaches were researched but the foundation was laid by LeChun, who proposed the Algorithm of Lenet-5 [1]. Lenet-5 achieved many milestones in this field and was widely accepted.
Following the years, the requirements in this field became more sophisticated and the Lenet-5 algorithm was not the ideal implementation method and researchers had to explore new strategies. An alternate approach was to use an exhaustive search. The downside was that it was computationally expensive and some objects were misclassified. To improve this, the selective search was proposed by Koen E. A. van de Sande in 2011 [2] where the number of locations to consider was reduced. The selective search uses a limited set of locations, using an expensive bag of words features but this was computationally expensive as well. In similar fashion different variations of methodologies were proposed which in turn led to improvement in performance and accuracy which lead to RCNN as proposed by Ross Girshick [3, 4]. RCNN is the state-of-the-art CNN-based deep learning object detection. It uses bounding boxes across the object regions, which then evaluates convolution networks independently on all the ROI [5, 6] to classify multiple image regions into the proposed class. Fast RCNN is an improved version of RCNN with RPN & it extracts features using ROI Pool from each candidate box and performs classification and bounding-box regression. Faster R-CNN is better by learning the attention mechanism with an RPN and Fast R-CNN architecture [7] but the breakthrough technology into Mask RCNN was proposed by Kaiming He et al [8]. This literature summary talks about the methodologies of these algorithms, implementation, and enhancements, which consequently leads to a greater insight into the field of image recognition.

2 Methodology

CNN: This deep learning algorithm is a transfer learning-based Convolutional Neural Network (CNN) and the face detection task is done using the algorithm's approach to extract features directly from images, and these extracted features are learned while training the network on a collected dataset of images and is not pre-trained. In Detecting Faces with Face Masks [9][10], here we deal with the evaluation of several methods for face detection when the face is covered by a mask. The methods evaluated are Haar cascade and Histogram of Oriented Gradients as feature-based approaches, Multitask Cascade Convolutional Neural Network, Max Margin Object Detection, and Tiny Face as convolutional neural network-based approaches, also in another paper [11] we introduce a deep learning computer vision model to recognize if a person visible through the camera is wearing a mask or not. Before the advent of deep learning-based algorithms[33], feature extraction was performed followed by classification.

Some of the various techniques used for feature extraction were SURF, SIFT, local binary patterns, and histogram of gradients. Once these features were extracted the classification was performed by any machine learning algorithm like K-means clustering, Principal Component Analysis, random forests, and Support Vector Machines [12]. The performance of the three algorithms: KNN, SVM, and Mobile Net to find the best algorithm which is suitable for checking who wearing a masked face in a real-time situation.

The results show that Mobile Net is the best accuracy both from input images and input video from a camera (real-time) [13]. The primary concern of this work is about facial masks, especially enhancing the recognition accuracy of different masked faces. A feasible approach has been proposed that consists of first detecting the facial regions. The occluded face detection problem has been approached using Multi-Task Cascaded Convolutional Neural Network (MTCNN). Then facial features extraction is performed using the Google FaceNet embedding model. And finally, the classification task has been performed by a Support Vector Machine (SVM) [14].

Deep learning models extract features from data through a lot of layers contained in their structures, and through the different functions, they perform in those layers. CNN will extract advanced features from the input image by performing convolution processing in the hidden layers for the face recognition process. For the network to extract good features from the data, the data set must have a large amount of data representing the problem. A deep learning model is proposed to automatically detect whether people wearing face masks or not. The pre-trained Faster R-CNN Inception V2 deep learning [15] model is fine-tuned with the transfer learning method and trained and tested on the Simulated Masked Face Dataset (SMFD). The model trained in the TensorFlow environment is accurate enough to detect the face mask [16][17].

A. Lenet-5 consists of 7 layers of which 3 are Convolution layers, 2 are sub-sampling layers and 1 is a fully connected layer. Lenet-5 laid the foundation for CNN architectures, it was a breakthrough technology during its time; it could not perform well in complex problems due to limited training and computation limitations.

B. Mask R-CNN: It is a two-staged architecture very much like Faster R-CNN[18]. The first stage is RPN and the Second Stage is a regional proposal. The feature map proposed at the primary stage is ROI pooled by the region [19][20], and goes through the leftover network, yields the class, bounding box, and the binary mask. An illustration of ROI Align in Mask R-CNN is displayed below. Rather than adjusting the black rectangles to have integer-length, black rectangles of equivalent size are utilized. In light of the area overlapping by the feature map values, bilinear interpolation is utilized to acquire an intermediate pooled feature map, which is displayed at the bottom right of the figure. Then, at that point, max pooling is performed on this intermediate pooled feature map. Furthermore, we have some surveys on improvements or add-ons added with the Mask RCNN algorithm which helps in improving its performance and yields better results [21][22][23][24]. A paper from Xia H and Zhu F (2019) used expanded Mask R-CNN for retinal edema detection. Currently, ophthalmologists use OCT (optical coherence tomography imaging) to detect retinal edema, however, diagnosis becomes subjective as this requires high skills to report a gray image that has a large pixel value and less resolution. Expanded Mask R-CNN provided a recognition accuracy of 92.27% while Faster R-CNN and usual Mask R-CNN gave recognition accuracy of 87.95% and 90.65%, respectively [25].

The application of Mask-RCNN is enormous in the medical field, especially in image processing. Radiologists and surgeons find it difficult to delineate the exact size and dimensions of tumors, particularly when they metastasize to surrounding areas. UlHaq MN et al (2021) proposed Mask-RCNN-based model for demarcating liver tumors that may help surgeons accurately identify and re-sect them. The authors collected computer tomography (CT) images, normalized them, applied ResNet 101 to map features, and finally used mask RCNN for liver tumor segmentation. The authors achieved a dice score of 0.95 with 0.12 and 0.15 Volumetric Overlap Error (VOE) and Relative Volume Difference (RVD), respectively which is considered to be highly significant. [3]

Multi-task loss on each sampled RoI is defined as

$$L = L_{class} + L_{box} + L_{mask}$$

Where

$$L_{class} + L_{box} = ((\frac{1}{N_{cls}}\sum_i L_{cls}(p_i, p_i^*) + \frac{1}{N_{cls}}\sum_i p_i^* L_1^{smooth}(t_i - t_i^*))$$

$$L_{mask} = -\frac{1}{m^2}\sum_{1\le i,j\le m}[y_{ij}\log y_{ij}^k + (1-y_{ij})\log(y_{yj}^k)]$$
$$L_{class} = (p_i, p_i^*) = -p_i^*\log p_i^* - (1-p_i^*)\log(1-p_i^*)$$
$$mAP = \frac{A\cap B}{A\cup B} = \frac{1}{N_T}\sum_i^{N_T}(\frac{N_i^{DR}}{N_i^D})$$ [24][26][27]

In detection and segmentation-based tasks, Region of Interest Align, or RoIAlign, is a method for extracting a tiny feature map from each RoI. As a result, the recovered features are correctly aligned with the input and the severe quantization of RoI Pool is removed as shown if Fig. 1.

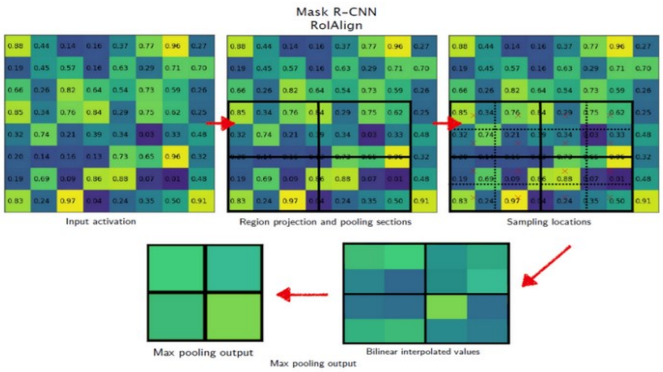

Fig. 1: Mask-RCNN RoI Align [15]

C. Semantic Segmentation: Different from classification, the Image is segmented at the pixel level. In this manner, an exact outcome needs to be passed as a judgment on the class of every pixel of the image [28]. However, CNN loses the image details during the time spent on convolution and pooling, which results in a smaller feature image due to which it can't well bring up the particular layout of the image. Thus accurate segmentation cannot be achieved. To tackle this issue, Jonathan Long et al [29] proposed FCN (Fully convolutional networks) for picture semantic division. At present, FCN has turned into the essential structure of semantic segmentation, and it is important to further develop feature resolution to create excellent segmentation results. The current segmentation strategies depend vigorously on the utilization of expanded convolution, which restricts the type of backbone networks. To keep up with this adaptability, different techniques were researched to supplant inflated convolution. As indicated by the work of V. Badrinarayanan, and A. Kendall [30][31], it is realized that Encoder-decoder or U-net engineering can supplant inflated convolution and further develop feature resolution. The explanation is that Encoder-decoder gradually samples high-level features from feed forward networks and combines them with features at lower levels to generate semantically meaningful high-resolution features. [32]

Model	Dataset	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	bike	person	plant	sheep	sofa	train	tv	mAP
Selective Search	VOC10	58.2	41.9	19.2	14	14.3	44.8	36.7	44.8	12.9	28.1	28.7	39.4	44.1	52.5	25.8	14.1	38.8	34.2	43.1	42.6	**33.9**
R-CNN T-Net	VOC10	64.2	69.7	50	41.9	32	62.6	71	60.7	32.7	58.5	46.5	56.1	60.6	66.8	54.2	31.5	52.8	48.9	57.9	64.7	**54.2**
R-CNN T-Net BB	VOC10	68.1	72.8	56.8	43	36.8	66.3	74.2	67.6	34.4	63.5	54.5	61.2	69.1	68.6	58.7	33.4	62.9	51.1	62.5	64.8	**58.5**
R-CNN O-Net	VOC10	71.6	73.5	58.1	42.2	39.4	70.7	76	74.5	38.7	71	56.9	74.5	67.9	69.6	59.3	35.7	62.1	64	66.5	71.2	**62.2**
R-CNN O-Net BB	VOC10	73.4	77	63.4	45.4	44.6	75.1	78.1	79.8	40.5	73.7	62.2	79.4	78.1	73.1	64.2	35.6	66.8	67.2	70.4	71.1	**66**
Fast R-CNN	VOC07++12	82	77.8	71.6	55.3	42.4	77.3	71.7	89.3	44.5	72.1	53.7	87.7	80	82.5	72.7	36.6	68.7	65.4	81.1	62.7	**68.8**
Faster R-CNN + SS	VOC07++12	82.3	78.4	70.8	52.3	38.7	77.8	71.6	89.3	44.2	73	55	87.5	80.5	80.8	72	35.1	68.3	65.7	80.4	64.2	**68.4**
Faster R-CNN + RPN	VOC07++12	76.5	79	70.9	65.5	52.1	83.1	84.7	86.4	52	81.9	65.7	84.8	84.6	77.5	76.7	38.8	73.6	73.9	83	72.6	**73.2**
Faster R-CNN + RPN	COCO+07++12	87.4	83.6	76.8	62.9	59.6	81.9	82	91.3	54.9	82.6	59	89	85.5	84.7	84.1	52.2	78.9	65.5	85.4	70.2	**75.9**

Table 1. Comparative analysis of all models implemented on VOC dataset[34]

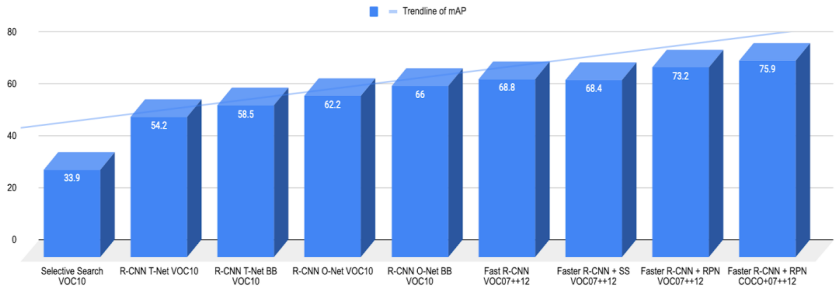

Fig. 2. Mean Average Precision w.r.t different models

Fig.2. is derived from table 1 and from the figure we can conclude that Faster R-CNN+RPN model have produced better results on COCO dataset.

3 Implementation

Our model was built on TensorFlow 1.13 and Keras 2.24 and used Matterport Mask-RCNN implementation. This project was first started in November 2017 and the last enhancement was done in April 2019. This model was trained on the COCO dataset and preloaded with its weights (mask_rcnn_coco.h5) file.

The metrics we have tracked are derived from RPN Bbox loss, RPN Class loss, MRCNN Bbox loss, MRCNN Class loss, and MRCNN Mask loss. The optimizer we have used in our project is SGD. We have also done some hyper-parameter tuning with backbone networks such as resnet50 and resnet101. We have trained the models with and without image augmentations.

We trained the entire model in the beginning, then we started fine-tuning the model from L3 and L5 layers.

System Requirements: 13GB RAM, 2CPU, 15GB GPU

8 images for 12 GB

Epochs: Min 16, Max 20 with steps per epoch = 200.

Image Size: 256 x 256 with 3 channels

Purpose of optimization Method-Optimization methods are used in many areas of study to find solutions that maximize or minimize some study parameters, such as minimizing costs in the production of a good or service, maximizing profits, minimizing raw material in the development of a good, or maximizing production.

Model 1 (Resnet50 + L5) - In This iteration we considered the pre-trained weights of coco till L4 and from L5 we fine-tuned the weights.

Model 2 (Resnet50 +L3) - In this iteration we considered the pre-trained weights of coco till L2 and from L3 we fine-tuned the weights.

Model 3 (ResNet 50 + augmentation) - In this iteration we considered the basic ResNet model and image augmentation with scaling image +/- 2

Model 4 (ResNet 101 with augmentation) - In this iteration we considered the basic ResNet model and image augmentation with scaling image +/- 2

Model 5 (ResNet 101) – In this iteration of the backbone network changed from ResNet 50 to ResNet 101

Table 2: Comparative Analysis of all models Implemented on Pneumonia Dataset

	Hyperparameter	Val_loss	Val_rpn_Classloss	Val_rpn_bbox_loss	Val_mrcnn_class_loss	Val_mrcnn_bbox_loss	Val_mrcnn_mask_loss
BaseModel	ResNet50	1.299373	0.017626	0.394497	0.122656	0.396552	0.368032
Model1	ResNet50 – L5	1.300705	0.015643	0.411534	0.119809	0.390725	0.362986
Model2	ResNet50 – L3	1.288251	0.016096	0.412266	0.115695	0.379142	0.365044
Model3	ResNet50 – Image Augmentation	1.268916	0.016680	0.380740	0.118789	0.389161	0.363537
Model4	ResNet101 – Image Augmentation	1.244234	0.014482	0.399649	0.099544	0.367485	0.363067
Model5	ResNet101	1.267256	0.015285	0.365733	0.117356	0.390759	0.378116

The parameters we have monitored during optimization are:-

- Loss: Sum of the below 5 loss parameters
- RPN Class Loss: RPN anchor classifier loss
- RPN BBox Loss: RPN bounding box loss
- MRCNN Class Loss: Loss for the classifier head of Mask R-CNN
- MRCNN BBox Loss = Loss for Mask R-CNN bounding box refinement
- MRCNN Mask Loss = Mask binary cross-entropy loss for the masks head
- mAP: Mean Average Precision

Each of these loss metrics is the sum of all the loss values calculated individually for each of the regions of interest. The calculation of the losses is described in [8]. The classification loss values are fundamentally subjected to the confidence a score of true class, thus the classification loss reflects how sure the model is while predicting the class name, or as such, as how close the model is to foreseeing the right class. On account of MRCNN Class Loss, all the object classes are covered.

However, on account of RPN Class Loss, the only classification that is done is marking the anchor boxes as foreground or background (which is the justification for why this loss will in general have lower values, as there are just 'two classes' than can be predicted).

The bounding box loss values reflect the distance between the genuine box boundaries - that is, the (x, y) coordinates of the box area. It is by its nature a regression loss, and it punishes larger absolute differences. Consequently, it, at last, shows how great the model is at finding objects inside the picture, on account of RPN BBox Loss; and how great the model is at exactly foreseeing the area(s) inside a picture compared to the various articles that are available, on account of MRCNN BBox Loss.

The mask loss, comparably to the classification loss punishes wrong per-pixel paired classifications (foreground/background, regarding the true class label). It is determined diversely for each region of interest: Mask R-CNN encodes a binary mask per class for each of the ROI and the mask loss for a particular ROI is determined exclusively on the mask compared to its actual class, which keeps the mask loss from being impacted by class predictions. The mAP compares the ground-truth

bounding box to the detected box and returns a score. The higher score is better the overall performance of the models is tabulated as follows. Overall the models showed steady growth with transfer learning but once the backbone network was changed to the Resnet101 and with the augmentations specified in the previous section we observed that it outperformed the other models. As we can see below Model 4 has the lowest loss of 1.244. This signifies that Model 4 is the best if we consider the performance of model w.r.t to loss.

4 Conclusion

X-ray image analysis is considered as a tedious and crucial task for radiology experts. In this paper, we have used the Mask-RCNN approach to solve this problem. The performance of Mask RCNN can be enhanced using various factors like input feature map pixels, batch size, trained weights, backbone networks, activation functions, and hyper-parameter tuning which improves the accuracy and classification. We can conclude that the various parameters as explained above have an impact and we can see the change in performance of the Mask RCNN algorithm. The proposed model has shown better results compared to previous iterations. The Future goal would be to fine tune the model with different activation functions and implement it in a packaged application which can be directly deployed in radiology departments. X-ray image analysis is considered as a tedious and crucial task for radiology experts. In this paper, we have used the Mask-RCNN approach to solve this problem. The performance of Mask RCNN can be enhanced using various factors like input feature map pixels, batch size, trained weights, backbone networks, activation functions, and hyper-parameter tuning which improves the accuracy and classification. We can conclude that the various parameters as explained above have an impact and we can see the change in performance of the Mask RCNN algorithm. The proposed model has shown better results compared to previous iterations. The Future goal would be to fine tune the model with different activation functions and implement it in a packaged application which can be directly deployed in radiology departments.

REFERENCES

1. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. (1998). "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2323.
2. Van de Sande, K. E., Uijlings, J. R., Gevers, T., & Smeulders, A. W. (2011, November). Segmentation as selective search for object recognition. In 2011 international conference on computer vision (pp. 1879-1886). IEEE.
3. Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2015). Region-based convolutional networks for accurate object detection and segmentation. IEEE transactions on pattern analysis and machine intelligence, 38(1), 142-158.
4. Htet, K. S., & Sein, M. M. (2020, October). Event Analysis for Vehicle Classification using Fast RCNN. In 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE) (pp. 403-404). IEEE.
5. Wu, X., Wen, S., & Xie, Y. A. (2019, August). Improvement of Mask-RCNN object segmentation algorithm. In International Conference on Intelligent Robotics and Applications (pp. 582-591). Springer, Cham.
6. Wang, T., Hsieh, Y. Y., Wong, F. W., & Chen, Y. F. (2019, November). Mask-RCNN based people detection using a top-view fisheye camera. In 2019 International Conference on Technologies and Applications of Artificial Intelligence (TAAI) (pp. 1-4). IEEE.
7. Girshick, R. (2015). Fast r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 1440-1448).
8. He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 2961-2969).
9. Prinosil, J., & Maly, O. (2021, July). Detecting Faces With Face Masks. In 2021 44th International Conference on Telecommunications and Signal Processing (TSP) (pp. 259-262). IEEE.
10. Nayak, R., & Manohar, N. (2021, July). Computer-Vision based Face Mask Detection using CNN. In 2021 6th International Conference on Communication and Electronics Systems (ICCES) (pp. 1780-1786). IEEE.
11. Yu, J., & Zhang, W. (2021). Face mask wearing detection algorithm based on improved YOLO-v4. Sensors, 21(9), 3263.
12. Sakshi, S., Gupta, A. K., Yadav, S. S., & Kumar, U. (2021, March). Face mask detection system using CNN. In 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) (pp. 212-216). IEEE.
13. Vijitkunsawat, W., & Chantngarm, P. (2020, October). Study of the performance of machine learning algorithms for face mask detection. In 2020-5th international conference on information technology (InCIT) (pp. 39-43). IEEE.
14. N. Darapaneni et al., "Inception C-Net(IC-Net): Altered Inception Module for Detection of Covid-19 and Pneumonia using Chest X-rays," 2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS), 2020, pp. 393-398, doi: 10.1109/ICIIS51140.2020.9342741.
15. Negi, A., Kumar, K., Chauhan, P., & Rajput, R. S. (2021, February). Deep neural architecture for face mask detection on simulated masked face dataset against covid-19 pandemic. In 2021 international conference on computing, communication, and intelligent systems (ICCCIS) (pp. 595-600). IEEE.
16. Öztürk, G., Eldoğan, O., Karayel, D., & ATALI, G. (2021). Face Mask Detection on LabVIEW. Artificial Intelligence Theory and Applications, 1(2), 9-18.
17. Wang, L., Lin, Y., Sun, W., & Wu, Y. (2021, June). Improved faster-RCNN algorithm for mask wearing detection. In 2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC) (Vol. 4, pp. 1119-1124). IEEE.

18. N. Darapaneni, R. Choubey, P. Salvi, A. Pathak, S. Suryavanshi and A. R. Paduri, "Facial Expression Recognition and Recommendations Using Deep Neural Network with Transfer Learning," 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2020, pp. 0668-0673, doi: 10.1109/UEMCON51285.2020.9298082.

19. Huang, Z., Zhong, Z., Sun, L., & Huo, Q. (2019, January). Mask R-CNN with pyramid attention network for scene text detection. In 2019 IEEE Winter Conference on Applications of Computer Vision (WACV) (pp. 764-772). IEEE.

20. Ullo, S. L., Mohan, A., Sebastianelli, A., Ahamed, S. E., Kumar, B., Dwivedi, R., & Sinha, G. R. (2021). A new mask R-CNN-based method for improved landslide detection. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, 3799-3810.

21. Haq, M. N. U., Irtaza, A., Nida, N., Shah, M. A., & Zubair, L. (2021, January). Liver Tumor Segmentation using Resnet based Mask-R-CNN. In 2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST) (pp. 276-281). IEEE.

22. Agrawal, T., & Urolagin, S. (2020, January). Multi-angle parking detection system using mask r-cnn. In Proceedings of the 2020 2nd International Conference on Big Data Engineering and Technology (pp. 76-80).

23. Liu, T., Guo, X., & Pei, X. (2021, August). Research on Recognition of Working Area and Road Garbage for Road Sweeper Based on Mask R-CNN Neural Network. In 2021 4th International Conference on Control and Computer Vision (pp. 76-82).

24. Xia, H., & Zhu, F. (2019, August). Expanded Mask R-CNN's Retinal Edema Detection Network. In Proceedings of the Third International Symposium on Image Computing and Digital Medicine (pp. 166-170).

25. Lin, Z., Guo, Z., & Yang, J. (2019, February). Research on texture defect detection based on faster-RCNN and feature fusion. In Proceedings of the 2019 11th International Conference on Machine Learning and Computing (pp. 429-433).

26. Li, X., Long, R., & Jin, K. (2019, May). DeepFuse neural networks. In Proceedings of the ACM Turing Celebration Conference-China (pp. 1-5).

27. Bhatti, H. M. A., Li, J., Siddeeq, S., Rehman, A., & Manzoor, A. (2020, December). Multi-detection and segmentation of breast lesions based on mask rcnn-fpn. In 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) (pp. 2698-2704). IEEE.

28. Siheng, X. I. O. N. G., Yadong, L. I. U., Rui, X. U., Ying, D. U., Zihan, C. O. N. G., Yingjie, Y. A. N., & Xiuchen, J. I. A. N. G. (2020, August). Power equipment recognition method based on mask R-CNN and bayesian context network. In 2020 IEEE Power & Energy Society General Meeting (PESGM) (pp. 1-5). IEEE.

29. N. Darapaneni et al., "Food Image Recognition and Calorie Prediction," 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2021, pp. 1-6, doi: 10.1109/IEMTRONICS52119.2021.9422510.

30. Badrinarayanan, V., Kendall, A., & SegNet, R. C. (2015). A deep convolutional encoder-decoder architecture for image segmentation. arXiv preprint arXiv:1511.00561, 5.
31. Songhui, M., Mingming, S., & Chufeng, H. (2019, November). Objects detection and location based on mask RCNN and stereo vision. In 2019 14th IEEE International Conference on Electronic Measurement & Instruments (ICEMI) (pp. 369-373). IEEE.

32. Park, J., Shin, C., & Kim, C. (2019, February). PESSN: Precision Enhancement Method for Semantic Segmentation Network. In 2019 IEEE International Conference on Big Data and Smart Computing (BigComp) (pp. 1-4). IEEE.

33. N. Darapaneni et al., "Object Detection of Furniture and Home Goods Using Advanced Computer Vision," 2022 Interdisciplinary Research in Technology and Management (IRTM), 2022, pp. 1-5, doi: 10.1109/IRTM54583.2022.9791508..

34. Tahir, H., Khan, M. S., & Tariq, M. O. (2021, February). Performance analysis and comparison of faster R-CNN, mask R-CNN and ResNet50 for the detection and counting of vehicles. In 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS) (pp. 587-594). IEEE.