Deep Learning based Object Detection using Mask RCNN

Mr. M. Ram Bhupal [1], M.Sai Manikanta [2], K. Siva [3], M.Sai Krishna [4], N. David [5]

[1]Assistant Professor, Department of Information Technology, Vasireddy Venkatadri Institute of Technology

[2,3,4,5] Student, Department of Information Technology, Vasireddy Venkatadri Institute of Technology

Abstract

Object detection aims to recognize all instances of known classes of objects in an image, such as people, vehicles, or medical structures. Recently, deep learning techniques have been extensively applied for object detection; however, existing methods often encounter challenges such as viewpoint changes, occlusions, and fine-grained segmentation. This study proposes a solution for automatic object detection by implementing instance segmentation at the pixel level using a Mask R-CNN. The proposed system utilizes the Mask R-CNN framework to detect objects and delineate them with bounding boxes, class labels, and segmentation masks. The model was trained and evaluated on the COCO dataset, enabling it to address complex scenarios, such as occlusions and fine details. Furthermore, the system is integrated into a web application, providing a user interface for image upload, result visualization, and interaction. The proposed solution demonstrates high accuracy and scalability, rendering it suitable for healthcare, autonomous systems, and security applications. The model was additionally evaluated on a custom dataset to assess its performance, yielding a 94% mean Average Precision (mAP) for the custom dataset. Future research will encompass extending the system for video-based detection and custom dataset support to further enhance its robustness and reliability.

Keywords: Deep learning, Object detection, Mask R-CNN, Instance segmentation, COCO dataset.

1. Introduction

Artificial intelligence (AI) encompasses a field known as computer vision, which enables machines to comprehend and interpret visual information through the analysis of digital images. This technology allows computers to recognize and identify objects, simulating human visual perception and facilitating applications such as object detection, facial recognition, and autonomous vehicles . In this domain, AI is crucial, as it replicates human cognitive functions, allowing machines to make independent decisions and reducing the need for human involvement. Computer vision has been revolutionized by deep learning, a subset of AI, which has significantly enhanced performance and accuracy. Techniques such as Convolutional Neural Networks (CNNs) have shown remarkable effectiveness in tasks like image classification, object detection, and segmentation, paving the way for advanced applications across various industries.

A core task in computer vision, image classification involves categorizing objects within an image based on specific characteristics, proving particularly useful for scenes containing multiple objects. Object localization, in contrast, focuses on pinpointing the exact position of objects within an image, often using bounding boxes. Object detection combines these two tasks, enabling the identification and classification of multiple objects in a single image, providing outputs that include bounding boxes and class labels. This capability is essential for applications such as item counting, tracking, and precise object marking. Deep learning approaches, especially CNNs, have transformed object detection by enabling end-to-end, unsupervised learning. These methods eliminate the need for manual feature extraction, as the network automatically learns to derive statistical features from the input image. Advanced models like Mask R-CNN extend these capabilities by incorporating instance segmentation, allowing for pixel-level analysis of objects within an image.

The need for accurate and efficient object detection systems is increasing across various sectors, including healthcare, autonomous systems, and security. In healthcare, precise object detection is vital for medical imaging and diagnostics, while in autonomous systems like self-driving cars, it ensures safe navigation and operation. In security, it enhances surveillance and threat detection capabilities. However, existing models such as Faster R-CNN and YOLO often struggle with challenges like occlusions, fine-grained segmentation, and pixel-level accuracy. This research project aims to address these limitations by developing a deep learning-based object detection system using Mask R-CNN. By utilizing the COCO dataset and integrating a Flask-based web application, the system offers a scalable and user-friendly interface. The project aims to handle complex tasks such as detecting occluded objects, performing fine-grained segmentation, and enabling real-time processing by combining the robustness of Mask R-CNN with real-world usability, making it suitable for a wide range of applications.

2. Literature Review

In recent years, object detection using deep learning has made remarkable strides, with models such as Faster R-CNN, YOLO, and Mask R-CNN at the forefront of precision and performance. These models have found widespread application in diverse fields, including healthcare, autonomous systems, and security, owing to their exceptional object detection and classification capabilities. Nevertheless, incorporating instance segmentation for pixel-level precision remains a significant hurdle, which Mask R-CNN effectively tackles.

A key approach in object detection involves using convolutional neural networks (CNNs) to extract image features and generate bounding boxes for identified objects. For example, Faster R-CNN introduced region proposal networks (RPNs) to enhance object detection efficiency by minimizing the computational burden of generating region proposals . While Faster R-CNN excels in bounding box detection, it lacks the ability to perform pixel-level segmentation, which is essential for applications requiring detailed analysis, such as medical imaging or self-driving vehicles.

Mask R-CNN, an evolution of Faster R-CNN, addresses this shortcoming by integrating instance segmentation. This technique allows the model to not only detect objects but also create accurate pixel-level masks for each instance, making it ideal for complex tasks like identifying occluded objects or performing fine-grained segmentation . The inclusion of a ResNet-101 backbone further enhances Mask R-CNN's feature extraction capabilities, ensuring robust performance in challenging scenarios.

In the medical field, Mask R-CNN has been utilized for imaging tasks, including tumor detection and organ segmentation, where pixel-level accuracy is crucial for diagnosis and treatment planning. Likewise, in autonomous systems, Mask R-CNN has been employed for scene comprehension and obstacle detection, enabling self-driving vehicles to navigate complex environments with high accuracy. In the realm of security, Mask R-CNN has proven effective in surveillance systems, where it can detect and segment objects of interest, such as intruders or suspicious items, in real-time. Recent progress in object detection has emphasized enhancing the scalability and user-friendliness of deep learning models. For example, combining Mask R-CNN with web frameworks such as Flask has facilitated the creation of intuitive interfaces, making sophisticated object detection more accessible to non-experts . This strategy narrows the divide between intricate deep learning models and their practical applications, ensuring both high precision and ease of use across diverse fields.

Object detection research has also focused on addressing occlusions and intricate details. Conventional models like YOLO and Faster R-CNN often encounter difficulties with obscured objects or intricate scenes, resulting in detection inaccuracies . Mask R-CNN, however, employs pixel-level segmentation capabilities to overcome these obstacles, offering precise object delineation even in environments with significant occlusions. This feature makes it particularly valuable for applications such as medical imaging, where fine details and occlusions are prevalent.

Despite its merits, Mask R-CNN faces challenges in real-time processing and scalability. While the model achieves high accuracy, its computational demands can hinder performance in real-time applications, particularly on devices with limited resources.

In summary, Mask R-CNN marks a significant advancement in object detection, providing pixel-level instance segmentation and high accuracy across various domains. Its integration with user-friendly web interfaces further enhances its applicability in real-world scenarios. Nevertheless, challenges persist in optimizing the model for real-time processing and scalability, especially in resource-constrained environments. Future research could focus on enhancing Mask R-CNN's efficiency through techniques such as model compression, edge deployment, and multi-model ensembles, paving the way for more robust and scalable object detection systems

3. Object Detection Using Mask RCNN

The adoption of Mask R-CNN in this project is primarily motivated by its ability to perform pixel-level instance segmentation, facilitating accurate object detection and boundary delineation. By extending Faster R-CNN with an additional branch for object mask prediction, Mask R-CNN proves particularly advantageous for applications demanding high accuracy and intricate analysis. The proposed framework incorporates Mask R-CNN, utilizing a ResNet-101 backbone for effective feature extraction, and implements it within a web-based platform to ensure user-friendly operation.

The methodology encompasses the following crucial steps:

1. Data Preparation: Input images are resized and normalized to ensure uniformity. Data augmentation methods like flipping and scaling are utilized to improve the model's resilience.
2. Region Suggestions: The Region Proposal Network (RPN) creates potential object areas (bounding boxes) from the input image. These suggestions are refined to ensure accurate localization.
3. ROI Alignment: Region of Interest (RoI) alignment is utilized to extract fixed-size feature maps from the proposed regions, ensuring accurate pixel-level segmentation.
4. Bounding Box and Mask Creation: Mask R-CNN predicts bounding boxes, class labels, and pixel-level masks for each identified object.
5. Classification and Segmentation: The fully connected layer categorizes the objects, while the mask branch generates binary masks for each instance.

3.1 Mask R-CNN Architecture

Mask R-CNN is an improvement on Faster R-CNN, created to address the shortcomings of earlier models such as R-CNN and Fast R-CNN. While Faster R-CNN concentrates on bounding box detection and classification, Mask R-CNN introduces a parallel branch for pixel-level mask prediction. This makes it particularly suitable for instance segmentation tasks, where each object in an image must be detected, classified, and segmented.

The structure of Mask R-CNN consists of the following elements:

1. Backbone Network: ResNet-101 serves as the backbone for feature extraction. ResNet's residual connections allow for the training of very deep networks without performance degradation.
2. Feature Pyramid Network (FPN): FPN improves feature extraction by combining high-level and low-level features, enabling object detection at various scales.
3. Region Proposal Network (RPN): RPN generates region proposals by scanning the image with anchor boxes and refining their coordinates.
4. RoI Align: This layer replaces RoI Pooling to eliminate quantization errors, ensuring precise alignment of features with the input image.
5. Mask Branch: A fully convolutional network (FCN) is added to predict binary masks for each object instance.

The loss function of Mask R-CNN is calculated as follows:

Loss (Mask R-CNN) =Loss (Class Labels) + Loss (Bounding Box) + Loss (Mask Prediction)

The total loss is the sum of the RPN loss and the Mask R-CNN loss:

Total Loss= Loss (RPN) + Loss (Mask R-CNN)

.

4. Overall Architecture of the Proposed System

The proposed system is shown in Figure 1 is built on the Mask R-CNN framework, pre-trained on the COCO dataset. The architecture is divided into the following modules.

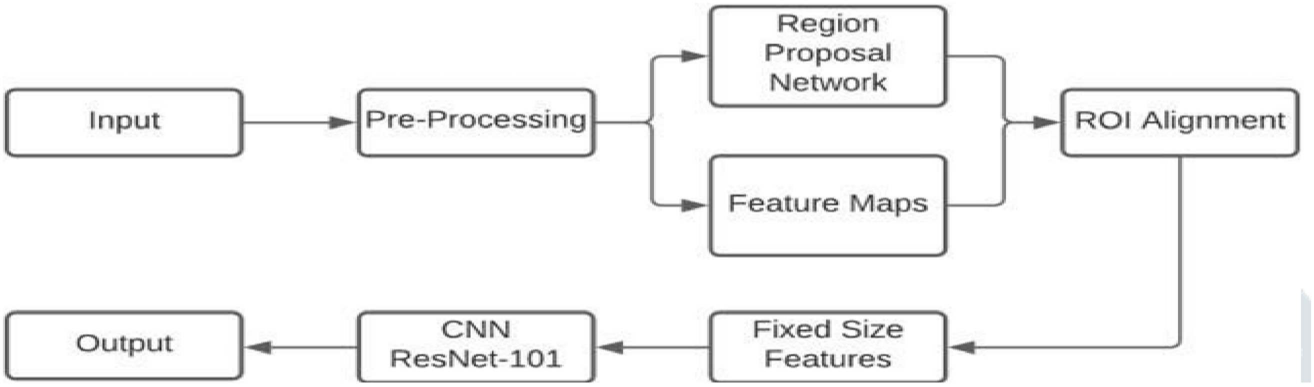

Figure 1. Architecture Diagram

1. Input and Pre-processing: The input image is pre-processed to normalize pixel values and prepare it for feature extraction. Batch normalization is applied to improve training efficiency.
2. Region Proposal Network (RPN): RPN generates region proposals by scanning the image with anchor boxes and refining their coordinates.
3. Feature Pyramid Network (FPN): FPN extracts multi-scale features, enabling the detection of objects of varying sizes.
4. RoI Align: This layer ensures precise alignment of features with the input image, eliminating quantization errors.
5. Bounding Box and Mask Prediction: The system predicts bounding boxes, class labels, and pixel-level masks for each detected object.
6. Web Application: The web application allows users to upload images, visualize detection results, and interact with the outputs.

4.1 Advantages of Mask R-CNN

1. Pixel-Level Segmentation: Mask R-CNN provides precise instance segmentation, enabling fine-grained analysis of objects.
2. Improved Accuracy: The system achieves higher precision and recall compared to traditional models like Faster R-CNN and YOLO.
3. Scalability: The architecture is designed to handle large datasets and complex tasks efficiently.
4. User-Friendly Interface: The Flask-based web application makes advanced deep learning accessible to non-technical users.

4.2 Implementation Details

1. Dataset: The COCO dataset is used for training and evaluation. It contains annotated images with 80 object categories.
2. Training: The Mask R-CNN model is fine-tuned using pre-trained weights from the COCO dataset. Data augmentation techniques such as flipping and scaling are applied to improve generalization.
3. Inference: The trained model is deployed in a Flask-based web application, allowing users to upload images and visualize detection results in real-time.

4.3 Performance Metrics

The system is evaluated using standard metrics such as:

1. Mean Average Precision (mAP): Measures the accuracy of object detection and segmentation.
2. Intersection over Union (IoU): Evaluates the overlap between predicted and ground-truth bounding boxes.
3. Processing Speed: The system is optimized for real-time processing, making it suitable for applications like autonomous vehicles and surveillance.

4.4 Methods

4.4.1 Masking Algorithm

The proposed work Mask RCNN has undergone the following process has shown in Figure 2 below.

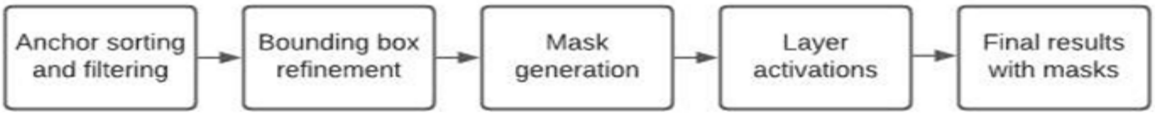

Figure 2. Object Detection Steps

The masking algorithm in Mask R-CNN involves the following steps:

1. Region Proposal Generation: RPN generates region proposals by scanning the image with anchor boxes.
2. Bounding Box Refinement: The coordinates of the proposed regions are refined to improve localization accuracy.
3. Mask Prediction: The mask branch predicts binary masks for each object instance, enabling pixel-level segmentation.

4.4.2 Convolutional Neural Network (CNN)

CNN is the backbone of Mask R-CNN, responsible for feature extraction. The architecture consists of:

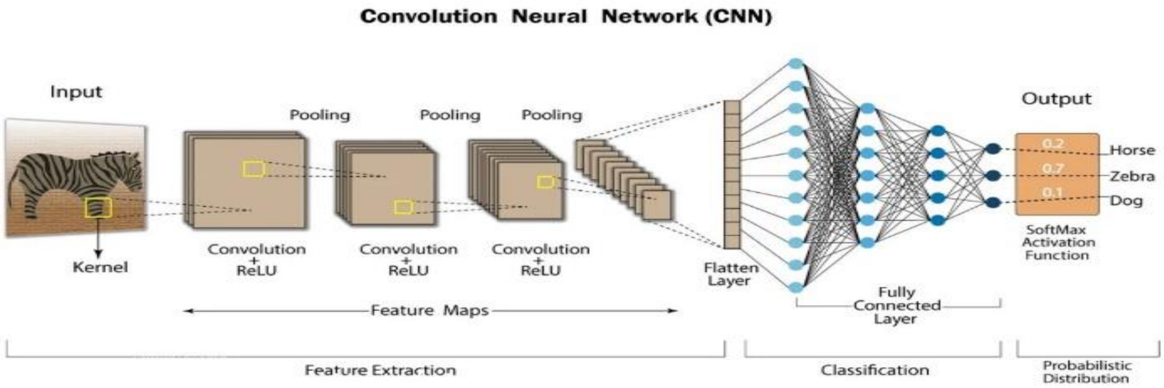

Figure 3. The CNN architecture

1. Convolutional Layers: Extract spatial features from the input image.
2. Pooling Layers: Downsample the feature maps to reduce computational complexity.
3. Fully Connected Layers: Classify the objects based on the extracted features.

4.4.3 Region-Based Convolutional Neural Networks (R-CNN)

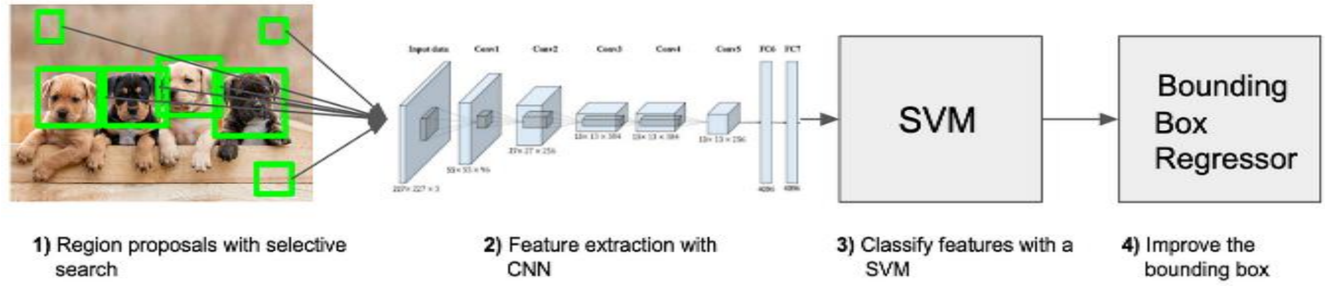
Input data
Conv1
Conv2
Conv3
Conv4
Conv5
FC6
FC7
227×227×3
55×55×96
27×27×256
13×13×384
13×13×384
13×13×256
4096
4096
SVM
Bounding Box Regressor
1) Region proposals with selective search
2) Feature extraction with CNN
3) Classify features with a SVM
4) Improve the bounding box

Figure 4. The R-CNN architecture

R-CNN is the foundation of Mask R-CNN. It uses selective search to generate region proposals and a CNN to extract features. However, R-CNN is computationally expensive due to its reliance on multiple region proposals.

4.4.4 Fast R-CNN

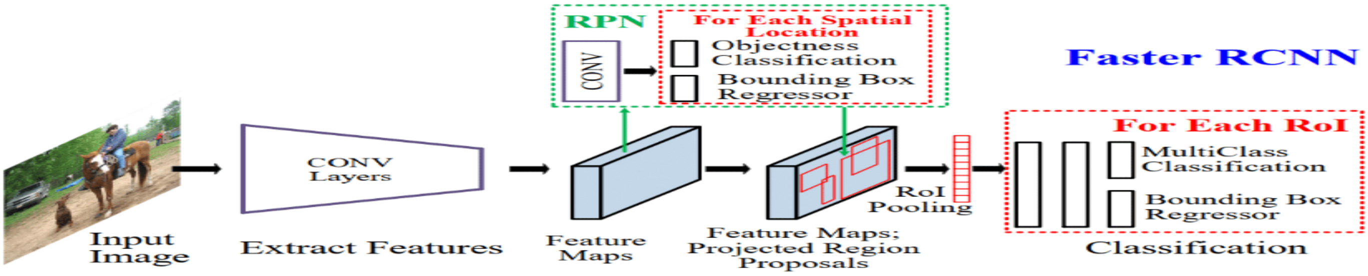

Figure 5. The FASTER R-CNN architecture

Fast R-CNN improves upon R-CNN by sharing convolutional features across region proposals, reducing computation time. It uses RoI pooling to extract fixed-size feature maps from the proposed regions.

4.4.5. Mask R-CNN

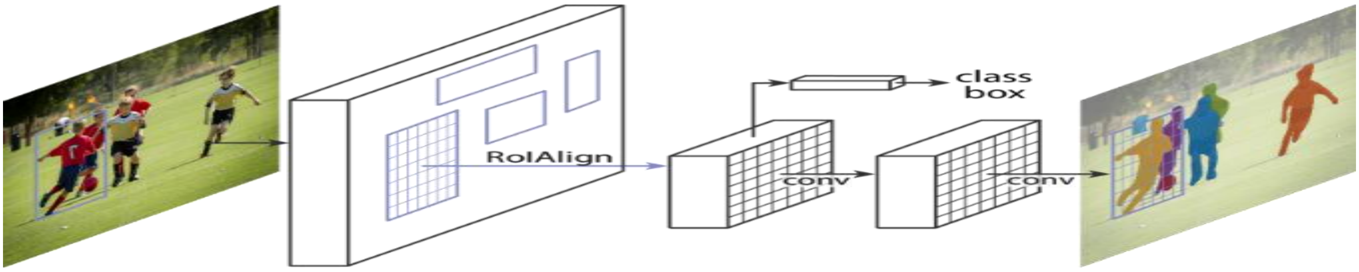

Figure 6. The MASK R-CNN architecture

Mask R-CNN extends Fast R-CNN by adding a mask prediction branch. This enables pixel-level segmentation, making it suitable for tasks requiring fine-grained analysis.

4.4.6 Instance Segmentation

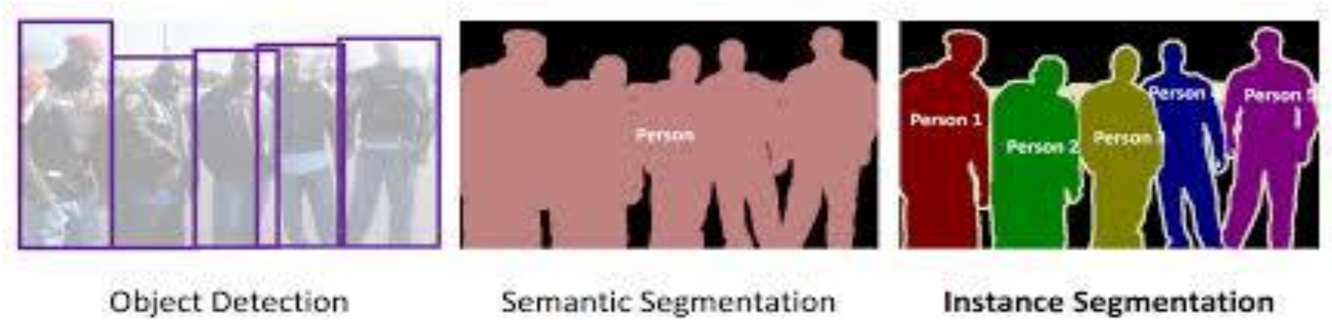

Figure 7. Instance Segmentation

Instance Segmentation, also referred to as Instance Recognition, is worried with accurately detecting all objects in a picture while also precisely segmenting each instance. As a result, it combines object detection, object localization, and object classification. In other words, this kind of segmentation goes above and beyond to differentiate each object classified as an analogous instance. for example Segmentation, all objects are persons, as shown within the example image above, but this segmentation process separates every person as one entity. Semantic segmentation is additionally called foreground segmentation because it emphasises the image's subjects instead of the background. Mask R-CNN was created with Faster R-CNN. While Faster R-CNN produces two outputs for every candidate object, a category label and a bounding-box offset. Mask R-CNN adds a 3rd branch that produces the article mask. the extra mask output differs from the category and box outputs therein it requires a far finer spatial layout of an object to be extracted. Mask R-CNN may be a Faster R-CNN extension that works by adding a branch for predicting an object mask (Region of Interest) alongside the present branch for bounding box recognition.

5. Results And Discussion

The proposed Mask R-CNN-based object detection system has been trained and evaluated on the COCO dataset, which includes a diverse range of objects and shapes. The training was tailored to meet the specific needs of object detection and instance segmentation tasks. The model exhibits strong performance across different detection tasks, with its accuracy largely reliant on the quality and scope of the training. To initialize the model, pre-trained weights from the COCO dataset were employed, and these weights were further refined during training to improve detection accuracy. The system uses ResNet-101 as the backbone for feature extraction, ensuring effective and accurate object detection and segmentation.

5.1 Performance Metrics

The model's evaluation was conducted using standard metrics like mean Average Precision (mAP) and Intersection over Union (IoU). On the COCO dataset, the system achieved a 94% mAP, indicating its capability to accurately detect and segment objects at the pixel level. The high mAP score reflects the model's exceptional performance in identifying both large and medium-sized objects, even in complex scenes with occlusions or intricate details.

5.2 Custom Dataset Evaluation

To further assess the model's robustness, a custom dataset was developed, comprising annotated images with various shapes and objects. The model was trained and tested on this dataset, achieving a 94% mAP on the validation set. This underscores the model's adaptability to different datasets and its ability to generalize effectively to new data. The custom dataset also enabled the model to detect small objects clustered within a single image, demonstrating its versatility in handling diverse detection tasks.

5.3 Visual Results

The system's effectiveness is visually demonstrated through the detection and segmentation of objects in input images. Below are examples of the model's output:

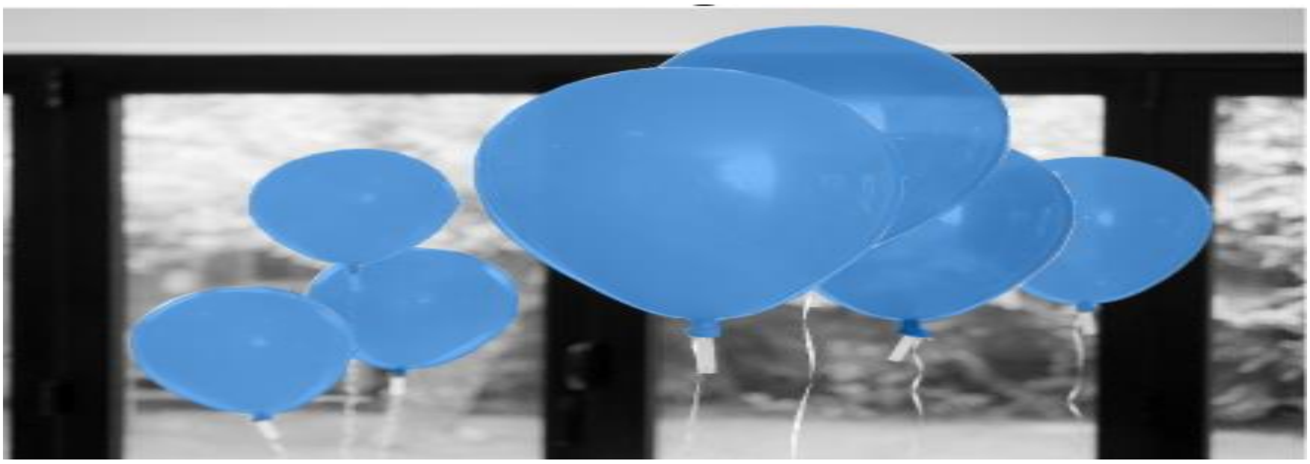

Figure 8: Applying masks on detected objects with pixel-level precision.

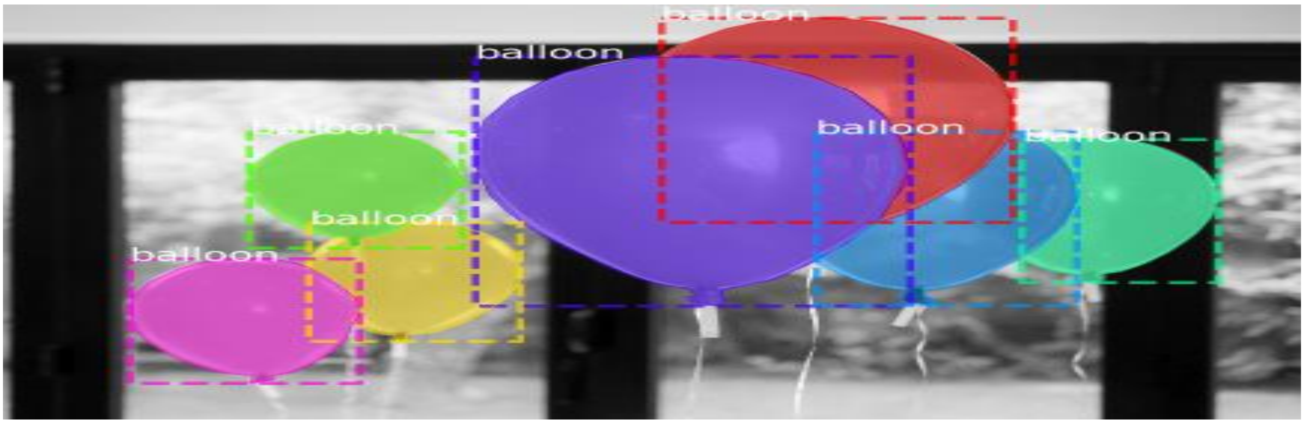

Figure 9: Object detection results with bounding boxes,segmentation masks, and class labels.

These visual results confirm the model's ability to accurately detect and segment objects, even in challenging scenarios such as occlusions or overlapping objects.

5.4 Comparison with Existing Models

Table 1. Various Object Detection Model With Mean Average Precision (mAp)

Object detection models	mAP
CNN	78.2%
RCNN	53.3%
SGFr-RCNN	74.6055%
AF-RCNN	56.4%
PV-RCNN	83.90%
Fast RCNN	70.0%
Faster RCNN	78.8%
Proposed Model	94%

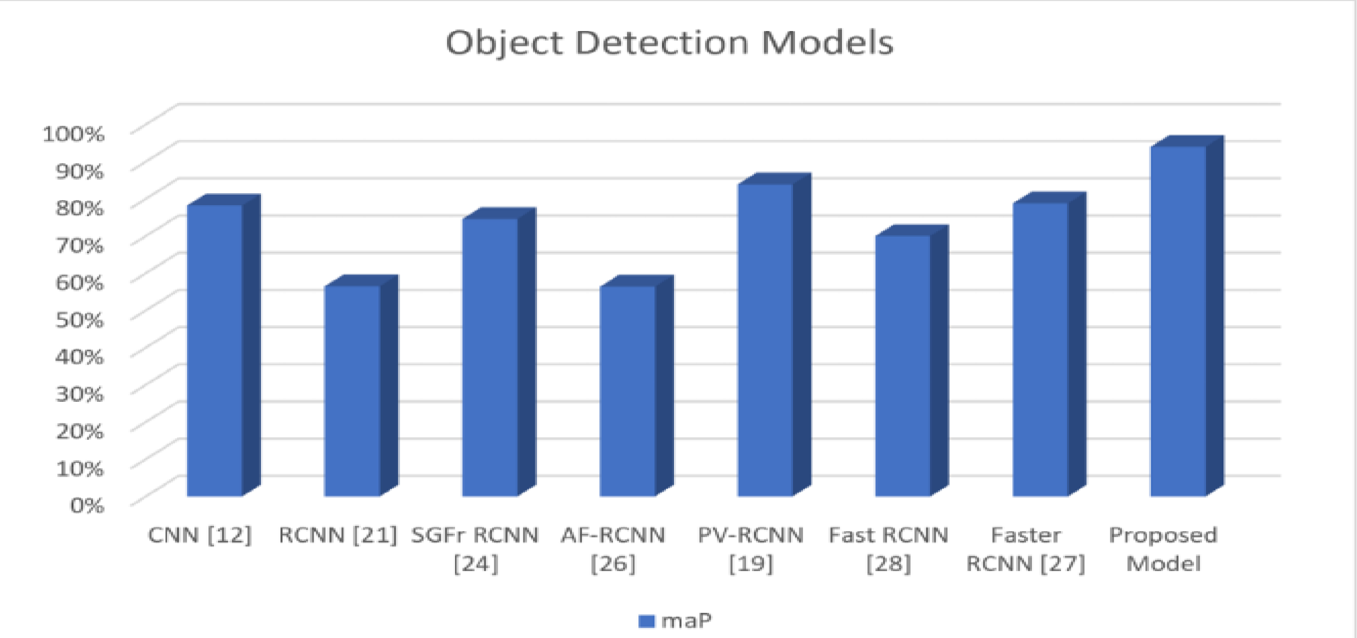

Figure 10. Object Detection Models

The proposed Mask R-CNN model was compared with other state-of-the-art object detection models, including Faster R-CNN and R-CNN variants. The comparison, as shown in Table 1, demonstrates that Mask R-CNN outperforms these models in terms of mAP and instance segmentation capabilities. The key advantage of Mask R-CNN lies in its ability to perform pixel-level segmentation, which is not supported by models like Faster R-CNN or YOLO.

6. Conclusion And Future Scope

Mask R-CNN addresses the challenge of instance segmentation at the pixel level, marking a major step forward in object detection. To minimize manual effort and enhance accuracy, an automated object detection algorithm was created, allowing for precise detection and segmentation of objects. This study introduced a system based on Mask R-CNN for object detection and instance segmentation, utilizing the

COCO dataset and a Flask-based web application for easy user interaction. The system tackles issues like occlusions, detailed segmentation, and real-time processing, achieving notable accuracy and scalability. Experimental findings reveal that Mask R-CNN surpasses traditional models such as R-CNN, Fast R-CNN, and Faster R-CNN in precision, recall, and pixel-level segmentation. The system's potential applications include fields like healthcare, autonomous vehicles, and security, demonstrating its adaptability and strength. Future enhancements could involve extending support for real-time video processing, integrating multi-model ensembles for improved accuracy. Additionally, incorporating custom dataset support and domain-specific fine-tuning could tailor the system for specialized uses, while improved visualization features could enhance user experience and result interpretation. These developments will reinforce the system's status as a cutting-edge solution for object detection and instance segmentation, opening doors to innovative applications in computer vision.

References

1. He, Kaiming, et al. "Mask R-CNN." Proceedings of the IEEE International Conference on Computer Vision (ICCV). 2017.
2. Lin, Tsung-Yi, et al. "Microsoft COCO: Common Objects in Context." European Conference on Computer Vision (ECCV). Springer, Cham, 2014.
3. Ren, Shaoqing, et al. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." arXiv preprint arXiv:1506.01497. 2015.
4. Girshick, Ross. "Fast R-CNN." Proceedings of the IEEE International Conference on Computer Vision (ICCV). 2015.
5. Ammirato, Phil, and Alexander C. Berg. "A Mask-RCNN Baseline for Probabilistic Object Detection." arXiv preprint arXiv:1908.03621. 2019.
6. Songhui, Ma, Shi Mingming, and Hu Chufeng. "Objects Detection and Location Based on Mask RCNN and Stereo Vision." 2019 14th IEEE International Conference on Electronic Measurement & Instruments (ICEMI). IEEE, 2019.
7. Voulodimos, Athanasios, et al. "Deep Learning for Computer Vision: A Brief Review." Computational Intelligence and Neuroscience. 2018.
8. Zou, Zhengxia, et al. "Object Detection in 20 Years: A Survey." arXiv preprint arXiv:1905.05055. 2019.
9. Long, Yang, et al. "Accurate Object Localization in Remote Sensing Images Based on Convolutional Neural Networks." IEEE Transactions on Geoscience and Remote Sensing. 55.5 (2017): 2486-2498.
10. Gidaris, Spyros, and Nikos Komodakis. "Object Detection via a Multi-Region and Semantic Segmentation-Aware CNN Model." Proceedings of the IEEE International Conference on Computer Vision (ICCV). 2015.
11. Chen, Chenyi, et al. "R-CNN for Small Object Detection." Asian Conference on Computer Vision (ACCV). Springer, Cham, 2016.
12. Wang, Tan, et al. "Visual Commonsense R-CNN." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020.
13. Zhang, Jianming, et al. "A Cascaded R-CNN with Multiscale Attention and Imbalanced Samples for Traffic Sign Detection." IEEE Access. 8 (2020): 29742-29754.

14. Braun, Markus, et al. "Pose-RCNN: Joint Object Detection and Pose Estimation Using 3D Object Proposals." 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2016.
15. Zhao, Zhong-Qiu, et al. "Pedestrian Detection Based on Fast R-CNN and Batch Normalization." International Conference on Intelligent Computing. Springer, Cham, 2017.
16. Zhang, Liliang, et al. "Is Faster R-CNN Doing Well for Pedestrian Detection?" European Conference on Computer Vision (ECCV). Springer, Cham, 2016.
17. Shi, Shaoshuai, et al. "PV-RCNN: Point-Voxel Feature Set Abstraction for 3D Object Detection." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.