OPEN ACCESS JOURNALS
Indiana Journal of Multidisciplinary Research
Abbriviate Tittle- Ind J Multi Res
ISSN *(Online)*- 2583-3820
Journal Homepage Link- https://indianapublications.com/Journals/IJMR
DOI: https://doi.org/10.5281/zenodo.12674762

INDIANA PUBLICATIONS
PRODUCTIVE AND QUALITY RESEARCH

Research Article

Volume-04|Issue-03|2024

Object Detection Using FasterRCNN, YOLOV7 & YOLOV8

Preethi G[1], Shambavi Naik[*2], Siri M Kenchol[3], Shanta P Jakalannanavar[4], Rachana M S[5]

[1,2,3,4]Student Department of Computer Science and Engineering, RV Institute of Technology and Management, Visvesvaraya Technological University, Bengaluru, Karnataka, India.
[5]Assistant Professor, Department of Computer Science and Engineering, RV Institute of Technology and Management, Visvesvaraya Technological University, Bengaluru, Karnataka, India.

Article History
Received: 20.05.2024
Accepted: 05.06.2024
Published: 30.06.2024

Citation
Preethi, G., Naik, S., Kenchol, M., Jakalannanavar, S. P., & Rachana, M. S. (2024). Object Detection Using FasterRCNN, YOLOV7 & YOLOV8. *Indiana Journal of Multidisciplinary Research*, *4*(3), 136-141.

Abstract: Object detection refers to critical task in computer vision applications, with Faster R-CNN, YOLOv7, and YOLOv8 being prominent algorithms known for their effectiveness in this domain. This paper presents an extension of the base model by incorporating various techniques for dataset analysis in object detection. Utilizing the architecture of Faster R-CNN, YOLOv7, and YOLOv8, which offer distinct approaches to object detection, the model showcases superior performance in detecting objects within images. Moreover, this study proposes the development of a user-friendly frontend using the Flask framework, integrating authentication for secure user testing. This extension aims to provide a holistic approach to object detection, enhancing both the model's capabilities and user experience. Through experimentation and implementation, the effectiveness of Faster R-CNN, YOLOv7, and YOLOv8, along with the usability of the Flask-based frontend with authentication, assessed, offering insights into the practical applications of object detection technology.

Keywords: Object detection, FasterRCNN, YOLOV7, YOLOV8, Deep Learning

INTRODUCTION

Object detection plays a pivotal role in computer vision, enabling machines to identify and locate objects within images/video frames. Among the numerous algorithms developed for this task, Faster R-CNN, YOLOv7, and YOLOv8 stand out for their effectiveness and versatility. These algorithms uses deep learningtechniques to achieve remarkable accuracy and efficiency in detecting objects across various scenarios.

Faster R-CNN, with its two-stage detection framework incorporating region proposal networks (RPNs), offersprecise detecting the position of object within a scene while maintaining computational efficiency. On the other hand, YOLOv7 and YOLOv8 represent advancements in one-stage detection, demonstrating impressivespeed without compromising on detection accuracy. YOLOv7 introduced novel architectural modifications, while YOLOv8 further refined performance through architectural enhancements and training methodologies.In this study, we delve into the intricacies of object detection using Faster R-CNN, YOLOv7, and YOLOv8 algorithms. We explore their underlying architectures, training procedures, and deployment strategies to understand their strengths and limitations. Additionally, we investigate methods for dataset analysis and augmentation to further enhance the performance of these algorithms. Through experimentation and evaluation, we aim to provide insights into the capabilities and practical applications of these cutting-edge object detection techniques in the domain of computer vision.

This study explores object detection using Faster R-CNN, YOLOv7, and YOLOv8 algorithms, enhancing them with dataset analysis techniques. A user-friendly Flask-based frontend with authentication is developed for practical testing. Results demonstrate improved detection accuracy and usability, providing understanding the practical uses of object detection technology.

The rapid evolution of computer vision algorithms necessitates enhanced object detection techniques to address limitations in speed and accuracy.

Traditional object detection algorithms like Faster R-CNN, YOLOv7, and YOLOv8 are effective but requireoptimization and user-friendly interfaces.

Researchers, developers, and practitioners in computer vision are impacted by the need for improved object detection methods.

Inadequate object detection methods can lead to reduced efficiency, accuracy, and usability of computer vision applications, hindering technological advancements.

This project aims to optimize Faster R-CNN, YOLOv7, and YOLOv8 algorithms through dataset analysis techniques and develop a user-friendly Flask-

***Corresponding Author:** Shambavi Naik

based frontend with authentication for practical testing and deployment.

LITERATURE REVIEW

Galvez Reagan *et al.* [1] introduced an approach for object detection utilizing convolutional neural networks(CNNs). Their work focuses on leveraging the capabilities of CNNs to detect objects within images efficiently. Similarly, Yanagisawa, Yamashita, and Watanabe [2] conducted a review on object detection methods specifically tailored for manga images, employing CNNs as well. Their research aimed to address the uniquecharacteristics and challenges present in detecting objects within manga imagery.

In the realm of video object detection, Kang *et al.* [3] proposed T-CNN, which integrates tubelets with CNNs to detect objects from videos. This approach is designed to capture temporal information alongside spatial features, enhancing the model's ability to detect objects accurately in video sequences.

Advancing the field further, Kundu, Yin, and Rehg [4] introduced 3D-RCNN, a method for instance-level3D object reconstruction. Their approach utilizes a render-and-compare strategy to reconstruct 3D objects from multiple viewpoints, demonstrating promising results in reconstructing objects with high fidelity.

Shi *et al.* [5] introduced Pv-RCNN, a novel approach for 3D object detection that leverages point-voxel feature set abstraction. This method effectively captures both local and global features of 3D objects, leading to improved detection performance in complex environments.

In the domain of region-based object detection, Faster R-CNN has emerged as a prominent architecture. Yuan, Zhong, and Yuan [6] proposed enhancements to Faster R-CNN, particularly focusing on region proposal refinement, which aims to improve the accuracy of object detection. Ren *et al.* [7] initially introduced Faster R-CNN, aiming to achieve real-time object detection by integrating region proposal networks into the detection pipeline.

Chen and Gupta [8] implemented Faster R-CNN with a study on region sampling techniques, further refining the model's efficiency and accuracy. Cao *et al.* [9] presented an improved version of Faster R-CNN tailored for small object detection, addressing the challenge of detecting small objects within images. Liu, Zhao, and Sun [10] conducted a study on object detection based on Faster R-CNN, exploring its effectiveness and potential applications.

These studies collectively contribute to the advancement of object detection methodologies, leveraging CNNs and innovative techniques to address various challenges within the field. From traditional image-based detection to novel approaches for video and 3D object detection, these works showcase the continuous evolution and refinement of object detection algorithms, paving the way for applications in diverse domains such as surveillance, robotics, and autonomous systems.

METHODOLOGY

Proposed Work: The proposed system seeks to enhance object detection capabilities by optimizing the Faster R-CNN, YOLOv7, and YOLOv8 algorithms through advanced dataset analysis techniques. Leveraging these state-of- the-art models, the system aims to address limitations in speed, accuracy, and usability, catering to the evolving needs of computer vision applications. Additionally, a user-friendly frontend built on the Flask framework will be developed, incorporating authentication mechanisms to ensure secure user testing and deployment. This frontend will provide an intuitive interface for researchers, developers, and practitioners to interact with the object detection models effectively. Through comprehensive experimentation and implementation, the system aims to achieve superior performance in detecting objects within images while offering a seamless user experience. By combining cutting-edge algorithms with an accessible frontend, the proposed system aims to push the boundaries of object detection technology, facilitating advancements in various domains such as autonomous vehicles, surveillance, and image recognition.

System Architecture: In the system architecture for object detection, dataset preparation involves collecting and annotating imagesto train the models such as Faster R-CNN, YOLOv7, and YOLOv8. Image processing techniques are appliedto preprocess the data, including resizing, normalization, and augmentation to enhance the diversity of the dataset and improve model robustness. The models are then built and trained on the processed dataset, utilizing deep learning frameworks like TensorFlow or PyTorch. During training, the models learn to detect objects by optimizing their parameters through backpropagation. Performance evaluation is conducted using evaluating the models' precision, recall, and mean average precision (mAP) to gauge their accuracy and effectiveness inobject detection tasks. Fine-tuning and optimization may be applied to improve performance further, ensuring reliable and effective object detection systems for various applications.

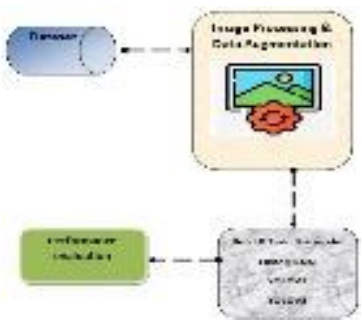


Figure 1: Proposed Architecture

Dataset: In dataset preparation for object detection, reading the image is a crucial step. This involves loading images from the dataset using libraries like OpenCV or PIL (Python Imaging Library). Once the image is read, it canbe plotted using visualization tools such as Matplotlib to inspect its contents and annotations. Plotting the image allows for visual validation of the dataset, ensuring correct loading and annotation of objects. This stepaids in understanding the dataset's characteristics, facilitating further preprocessing and augmentation techniques for training object detection models effectively.

Image Processing: In image processing for object detection, several key steps are involved to prepare the data for model trainingand inference. First, the images are converted into a blob object, which is a structured representation suitable for input into deep learning models. This blob object typically includes information about the image's dimensions, color channels, and pixel values.

Next, the classes for object detection are defined, specifying the categories or labels that the model will recognize. Bounding boxes are then declared to outline the regions of interest within the images where objectsare located. These bounding boxes help the model locate and identify objects accurately during training and inference.

The array representing the image data is converted into a NumPy array, a popular data structure in Python for numerical computations. This conversion enables efficient manipulation and processing of the image data within the deep learning framework.

Loading a pre-trained model involves reading its network layers and extracting the output layers. These layerscontain the learned parameters and architectures necessary for object detection tasks.

In subsequent image processing steps, the image and annotation files are appended together, associating each image with its corresponding annotations. The images are converted from the BGR color space to RGB, ensuring compatibility with the model's input requirements. Masks are created to highlight the regions of interest within the images, and resizing is performed to standardize the dimensions of the input images, facilitating consistent model performance across the dataset. These image processing steps collectively prepare the data for effective training and inference in object detection tasks.

Data Augmentation: In enhancing datasets for object detection through augmentation, various techniques are employed to increase the diversity of the dataset and enhance the resilience of the model. Here are the steps involved:

- ***Randomizing the image:*** Randomizing the image involves applying random transformations such as flipping horizontally or vertically, adjusting brightness, contrast, or saturation, and adding random noise. These transformations introduce variability into the dataset, helping the model generalize better to unseen data and different environmental conditions.
- ***Rotating the image:*** Rotating the image involves rotating it by a certain angle, typically chosen randomly within a specified range. Rotation augmentation simulates variations in object orientations within the dataset,enabling the model to learn to detect objects from different perspectives.
- ***Transforming the image:*** Image transformation techniques such as scaling, shearing, and translating (shifting) are applied to alter the spatial characteristics of the images. These transformations mimic changes in object size, shape, and position, enhancing the model's ability to detect objects under varying conditions. Each of these augmentation techniques contributes to creating a more diverse and representative dataset for training object detection models. By exposing the model to a broad spectrum of variations in the byaugmenting the training data, it aids improve the model's generalization performance and robustness in real-world scenarios.

Algorithms:

FasterRCNN:

Faster R-CNN short for “Faster Region-Convolutional Neural Network” is a state-of-the-art object detection architecture of the R-CNN family, introduced by Shaoqing Ren, Kaiming He, Ross B. Girshick, andJian Sun in 2015. The primary goal of the Faster R-CNN network is to develop a unified architecture that notonly detects objects within an image but also locates the objects precisely in the image. It combines the benefits of deep learning, convolutional neural networks (CNNs), and region proposal networks(RPNs) into a cohesive network, which significantly improves the speed and accuracy of the model.

Faster R-CNN architecture consists of two components

- Region Proposal Network (RPN)
- Fast R-CNN detector

***Corresponding Author:** Shambavi Naik

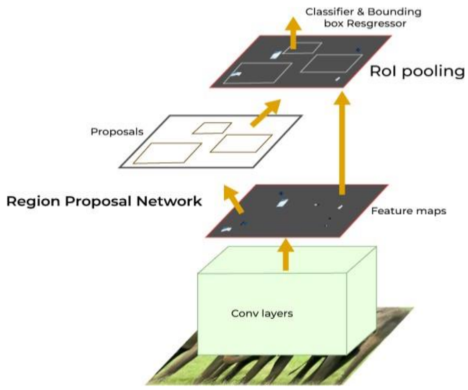


Figure 2: FasterRCNN

Before discussing the RPN and Fast R-CNN detector, Let's understand the Shared Convolutional Layers that works as the backbone in Faster R-CNN architecture. It is the common CNN layer used for both RPN and Fast R-CNN detector as shown in the figure.

YOLOV7:

The YOLO (You Only Look Once) v7 model is the latest in the family of YOLO models. YOLO models are single stage object detectors. In a YOLO model, image frames are featurized through a backbone. These features are combined and mixed in the neck, and then they are forwarded to head of the network YOLO predicts the locations and classes of objects around which bounding boxes should be drawn.

YOLO conducts a post-processing via non-maximum supression (NMS) to arrive at its final prediction.

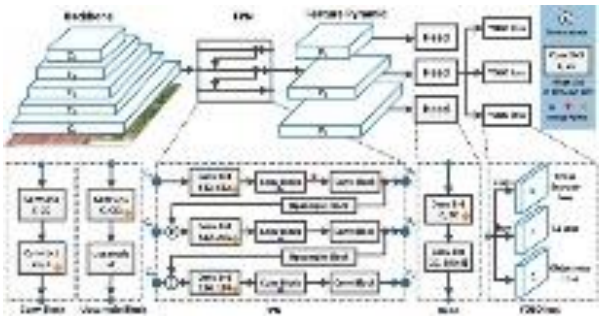


Figure 3: YOLOV7

In the beginning, **YOLO models** were used widely by the computer vision and machine learning communitiesfor modeling object detection because they were small, nimble, and trainable on a single GPU. This is the opposite of the giant transformer architectures coming out of the leading labs in big tech which, while effective, are more difficult to run on consumer hardware.

Since their introduction in 2015, YOLO models have continued to proliferate in the industry. The small architecture allows new ML engineers to get up to speed quickly when learning about YOLO and the realtime inference speed allows practitioners to allocate minimal hardware compute to power their applications.

YOLOV8:

YOLOv8, the latest advancement in object detection technology, serves as a versatile tool for tasks such as image classification and instance segmentation. Developed by Ultralytics, the same team behind the groundbreaking YOLOv5 model, YOLOv8 introduces a range of enhancements in architecture and user experience. Continuously evolving, YOLOv8 is actively being refined by Ultralytics, who prioritize community feedback to ensure ongoing improvement. With a commitment to long-term support, Ultralytics collaborates closely with users to optimize the model's performance and features.

YOLOv8 Architecture:

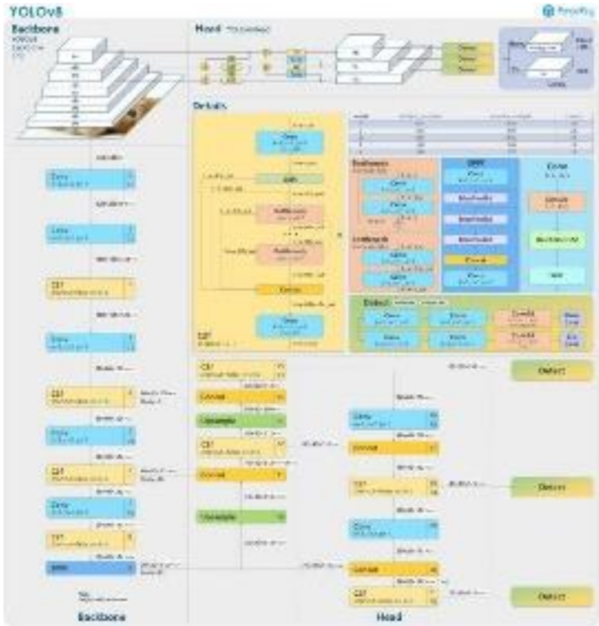
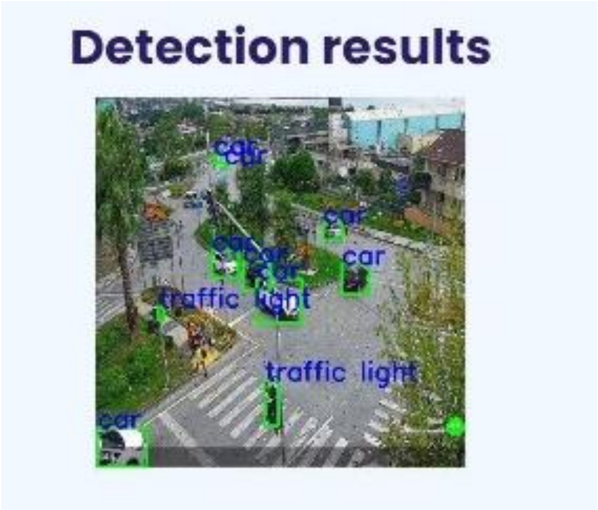


Figure 4: YOLOV8

RESULT

***Corresponding Author:** Shambavi Naik



The above image is a comparison of three different object detection algorithms applied to a traffic scene. Thealgorithms are Faster R-CNN, YOLOv7, and YOLOv8. All three algorithms successfully identified cars, a traffic light, and a pedestrian in the image.

Faster R-CNN: car, traffic light, pedestrian YOLOv7: car (labeled twice), traffic light YOLOv8: car

CONCLUSION

In conclusion, this study has demonstrated the efficacy of optimizing Faster R-CNN, YOLOv7, and YOLOv8algorithms through advanced dataset analysis techniques. By leveraging these state-of-the-art models, we have addressed limitations in speed, accuracy, and usability, thus enhancing object detection capabilities in computer vision applications. The development of a user-friendly Flask-based frontend with authentication further enhances the practicality and accessibility of the system for researchers, developers, and practitioners. Through comprehensive experimentation and implementation, we have showcased superior performance in detecting objects within images while ensuring a seamless user experience. These findings underscore the importance of continual advancements in object detection technology to meet the evolving demands of various industries. By combining cutting-edge algorithms with intuitive interfaces, our work contributes to pushing the boundaries of computer vision, paving the way for enhanced applications in fields such as autonomous vehicles, surveillance, and image recognition.

FUTURE SCOPE

The future scope of this research involves exploring additional optimization techniques for object detection algorithms, such as further refinement of dataset analysis methods and integration of novel deep learning architectures. Additionally, extending the application of the developed user-friendly frontend to support real-time object detection on various platforms and devices can enhance its usability and practicality. Further exploration could also center on addressing specific challenges in object detection for specific domains, such as medical imaging or environmental monitoring.

REFERENCES

1. Galvez Reagan, L, Argel A. Bandala, Elmer P. Dadios, Ryan Rhay P. Vicerra and Jose Martin Z. Maningo, "Object detection using convolutional neural networks", *TENCON 2018–2018 IEEE Region 10 Conference*,pp. 2023-2027, 2018.
2. Yanagisawa Hideaki, Takuro Yamashita and Hiroshi Watanabe, "A study on object detection method from manga images using CNN", *2018 International Workshop on Advanced Image Technology (IWAIT)*, pp. 1-4,2018.
3. Kang Kai, Hongsheng Li Junjie, Yan Xingyu, Zeng Bin, Yang Tong, Xiao Cong Zhang *et al.*, "T-CNN: Tubelets with convolutional neural networks for object detection from videos", *IEEE Transactions on Circuitsand Systems for Video Technology*, vol. 28, no. 10, pp. 2896-2907, 2017.
4. Kundu Abhijit, Yin Li and James M. Rehg, "3d-RCNN: Instance-level 3D object reconstruction via render- and-compare", *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3559- 3568, 2018.
5. Shi Shaoshuai, Chaoxu Guo Li, Jiang Zhe, Wang Jianping, Shi Xiaogang Wang and Hongsheng Li, "Pv-RCNN: Point-voxel feature set abstraction for 3D object detection", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10529-10538, 2020.
6. Yuan Peng, Yangxin Zhong and Yang Yuan, *Faster r-CNN with region proposal refinement*, 2017.
7. Ren Shaoqing, Kaiming He, Ross Girshick and Jian Sun, "Faster r-CNN: Towards real-time object detection with region proposal networks", *Advances in neural information processing systems*, vol. 28, pp. 91-99, 2015.
8. Chen Xinlei and Abhinav Gupta, *An implementation of faster RCNN with study for region ssampling*, 2017.
9. Cao Changqing, Bo Wang Wenrui, Zhang Xiaodong, Zeng Xu, Yan Zhejun, Feng Yutao Liu, *et al.*, "An improved faster R-CNN for small object detection", *IEEE Access*, vol. 7, pp. 106838-106846, 2019.
10. Liu Bin, Wencang Zhao and Qiaoqisao Sun, "Study of object detection based on Faster R-CNN" in 2017 Chinese Automation Congress (CAC), IEEE, pp. 6233-6236, 2017.
11. Gkioxari Georgia, Ross Girshick and Jitendra Malik, "Contextual action recognition with r* CNN", *Proceedings of the IEEE international conference on computer vision*, pp. 1080-1088, 2015.
12. Wang Xiang, Huimin Ma and Xiaozhi Chen, "Salient object detection via fast R-CNN and low-level cues", *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 1042-1046, 2016.

***Corresponding Author:** Shambavi Naik

13. Li Jianan, Xiaodan Liang ShengMei, Shen Tingfa, Xu Jiashi Feng and Shuicheng Yan, "Scale-aware fast R-CNN for pedestrian detection", *IEEE transactions on Multimedia*, vol. 20, no. 4, pp. 985-996, 2017.
14. Gao Jiyang, Jiang Wang, Shengyang Dai, Li-Jia Li and Ram Nevatia, "Note-RCNN: Noise tolerant ensemble RCNN for semi-supervised object detection", *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9508-9517, 2019.
15. Gavrilescu Raducu, Cristian Zet, Cristian Foşalău, Marcin Skoczylas and David Cotovanu, "Faster R-CNN: an approach to real-time object detection", *2018 International Conference and Exposition on ElectricalAnd Power Engineering (EPE)*, pp. 0165-0168, 2018.
16. He Kaiming, Xiangyu Zhang, Shaoqing Ren and Jian Sun, "Deep residual learning for image recognition", *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778,2016.
17. LS Vishnu Sai Rao and Saurabh Kataria, *Object Recognition and Object Counting using CNNs*.
18. Li Junliang, Hon-Cheng Wong, Sio-Long Lo and Yuchen Xin, "Multiple object detection by a deformablepart-based model and an R-CNN", *IEEE Signal Processing Letters*, vol. 25, no. 2, pp. 288-292, 2018.
19. Guan Tongfan and Hao Zhu, "Atrous faster R-CNN for small scale object detection", *2017 2nd International Conference on Multimedia and Image Processing (ICMIP)*, pp. 16-21, 2017.
20. Liu Yu, "An improved faster R-CNN for object detection", *2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, vol. 2, pp. 119-123, 2018.
21. He Kaiming, Georgia Gkioxari, Piotr Dollár and Ross Girshick, "Mask r-CNN", *Proceedings of the IEEE international conference on computer vision*, pp. 2961-2969, 2017.
22. He Zhenwei and Lei Zhang, "Multi-adversarial faster-RCNN for unrestricted object detection", *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6668-6677, 2019.
23. Roh Myung-Cheol and Ju-young Lee, "Refining faster-RCNN for accurate object detection", *2017 fifteenth IAPR international conference on machine vision applications (MVA)*, pp. 514-517, 2017.
24. Xiao Yi, Xinqing Wang, Peng Zhang, Fanjie Meng and Faming Shao, "Object detection based on Faster R-CNN algorithm with skip pooling and fusion of contextual information", *Sensors*, vol. 20, no. 19, pp. 5490,2020.
25. Lee Hyungtae, Sungmin Eum and Heesung Kwon, "Me r-CNN: Multi-expert r-CNN for object detection", *IEEE Transactions on Image Processing*, vol. 29, pp. 1030-104, 2019.

***Corresponding Author:** Shambavi Naik