Comparative Analysis of YOLO Variants Based on Performance Evaluation for Object Detection

Aoxiang Chen

Guangdong Country Garden School, 523808 Guangdong, China

Abstract. This study focuses on analysing and exploring the You Only Look Once (YOLO) algorithm. Specifically, this article analyses the evolution and performance of three versions (YOLOv1, YOLOv5, and YOLOv8) in object detection. The research begins by detailing the fundamental concepts of object detection and the datasets commonly used in this field. It then delves into the specific architectures and experimental outcomes associated with each YOLO version. The analysis reveals that while YOLOv8 introduces advanced features and improvements, earlier versions like YOLOv5 may offer superior stability and performance under certain conditions, particularly in specific tasks such as car detection. The discussion emphasizes the significant impact of factors such as batch size on model performance, suggesting that fine-tuning these parameters can optimize the algorithm for particular applications. The study concludes that the future of YOLO development lies in exploring and refining different variants, particularly those of YOLOv8, to better meet diverse requirements. By focusing on five distinct YOLOv8 variants, the research aims to enhance the adaptability and effectiveness of the YOLO framework across a wide range of object detection challenges, thereby contributing valuable insights into the ongoing advancement of this technology.

1 Introduction

Object detection is a technology in computer vision and machine learning aimed at identify specific objects in images. The primary challenges in object detection involve classification, localization, size, and shape of objects. One-stage and two-stage methods are two main types of deep learning-based algorithms. The two-stage method first extracts image features and then generates a set of candidate zones that may include targets according to these features. This first step is called Region Proposal (RP), followed by region classification and bounding box regression. In contrast, the one-stage method proceeds directly from feature extraction to classification and bounding box regression, omitting the RP step. This difference marks the key distinction between these approaches.

Object detection technology provides assistance in various fields. For instance, in facial detection, it aids in identifying fugitives and locating missing persons, thus contributing to public safety. In autonomous driving, it enhances lane-changing capabilities by segmenting the scene and detecting lane markings. Moreover, object detection is widely applied in fields

Corresponding author: miaokaihong@ldy.edu.rs

such as defect detection in industry, traffic self-driving, augmented reality, and more [1]. The most widely used algorithms is You Only Look Once (YOLO). Since YOLOv1 was published by Redmon et al. in 2015, the algorithm has seen continuous development, with YOLOv8 being released on 10/01/2023. Due to its high accuracy and speed, YOLO is particularly suitable for real-time applications like surveillance systems, facial recognition, and autonomous driving [2,3].

Convolutional Neural Networks (CNNs) have achieved great success in the field of computer vision due to their powerful feature learning ability, significantly improving object detection performance. Region-based Convolutional Neural Networks (RCNN) is one such algorithm, building on the strengths of CNNs to outperform earlier models like Support Vector Machines (SVMs). R-CNN, Fast-RCNN, Faster-RCNN and other advanced models like these have further propelled progress in object detection and the broader computer vision field. Similarly, YOLO also leverages CNNs to perform end-to-end object detection, resizing the input image before feeding it into the network to predict objects. Compared to RCNN, YOLO operates on a unified platform with faster performance by converting object detection into a regression problem. However, YOLO's accuracy still faces challenges, especially when detecting objects that are clustered or closely positioned, which is why the algorithm continues to evolve [4].

Comparing various versions of the YOLO object detection models and offering a comprehensive overview of the evolution and key concepts within the field of object detection are the primary objective of this essay. The analysis specifically centres on YOLOv1, YOLOv3, and YOLOv8, with a focus on elucidating the distinct characteristics and advancements of each model. This study not only examines the experimental outcomes associated with the YOLO series but also delves into the future trajectory of object detection research, identifying the current challenges and limitations that the YOLO models face. By highlighting these aspects, providing valuable insights for the sustainable development of object detection technology is the aim of this essay.

2 Methodology

2.1 Dataset description and preprocessing

Dataset is essential for advancing machine learning process. Microsoft Common Objects in Context (MSCOCO), mainly used for experimenting image/object classification, detection and segmentation tasks with approaches based on Machine Learning/ Deep Learning (ML/DL), is one of the standards and most popular datasets. Although it has less categories, it includes more examples in each category. Commonly encountered objects like person, dog, and so on are included in 91 different categories. In addition to a huge number of instances in each category, it also observed multiple instances of each image with different features [5].

Another benchmarking dataset, mostly used for visual object classification, segmentation, and detection, is called Pascal Visual Object Classes (Pascal VOC). PASCAL VOC provides a standard image labelling and evaluation system [6]. Pascal VOC image dataset is divided into four main classes that are vehicle, household, animal and person, and these classes can be separated into 20 categories. The dataset has a high-quality and labeled completely image what is very suitable for examining the algorithm performance [6]. Pascal VOC has two subsets, VOC2007 and VOC2012. These two datasets are often used in object detection, but because they contain different images, they can be used as independent testing and training sets, even the fact that they are mutually exclusive allows them to be used as separate benchmarks, and this is why the two sets can be used together.

2.2 Proposed approach

The essay begins by introducing the fundamental concepts and principles of object detection, including key techniques such as YOLO and RCNN. It then provides a comprehensive analysis of the principles, architecture, and processes underlying YOLOv1, YOLOv5, and YOLOv8. Following this, the essay presents and examines the experimental details and results associated with these YOLO models, offering a thorough evaluation of their technical implementations. In the final section, the discussion shifts to the current challenges in the development of object detection technologies, exploring potential future directions and scalable solutions. This section also investigates the bottlenecks and prospects specific to the YOLO series. The platform is illustrated in Fig. 1.

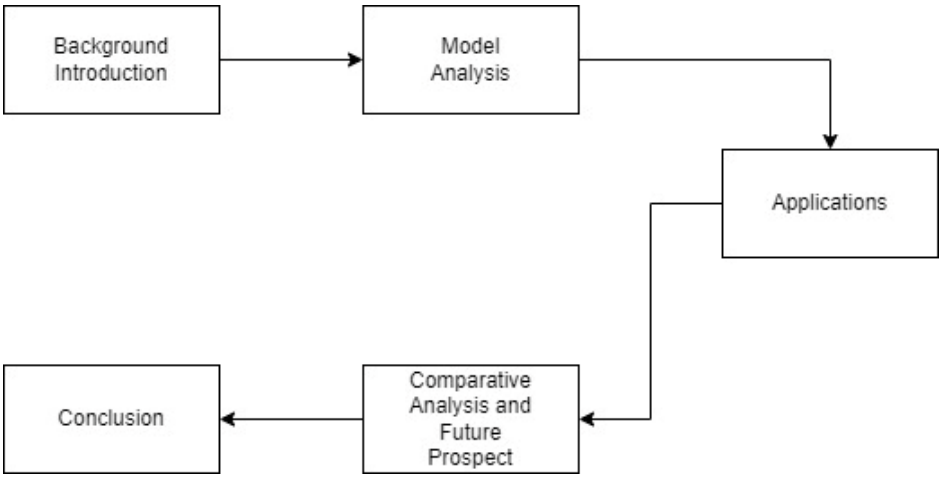

Fig. 1. The pipeline of the research (Picture credit: Original).

2.2.1 YOLOv1

At the 2016 Conference on Computer Vision and Pattern Recognition (CVPR), YOLOv1 was introduced by Joseph Redmon and colleagues. It marks an nonnegligible advancement in object detection. The YOLOv1 architecture is composed of 24 convolutional layers and 2 fully connected layers. Inspired by the GoogLeNet architecture, YOLOv1 innovatively replaced GoogLeNet's initial modules with a combination of 1*1 convolutions and 3*3 convolutional filters. The first 20 layers were pre-trained with a resolution of 224x224 pixels on the ImageNet dataset initially. Following this, four additional layers were appended, initialized with random weights, and subsequently trained on the VOC2007 and VOC2012 datasets. This approach enabled YOLOv1 to maintain a reliable balance between accuracy and speed, laying the foundation for subsequent developments in the YOLO series.

2.2.2 YOLOv5

Glen Jocher, founder and CEO of Ultralytics [7], released YOLOv5 [8,9] a few months after the release of YOLOv4 in 2020. It used an Ultralytics algorithm called AutoAnchor which can check and adjust anchor boxes when they are not suitable for the training settings and datasets, and it also developed in Pytorch instead of Darknet. Optimized from YOLOv1 algorithm, YOLOv5 moves out the full connection layers in YOLOv1 algorithm [10]. Due to the use of darknet-19 network model to extract depth feature of the image, target region is predicted by anchor points [11]. Five scaled versions are published from YOLOv5:

YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l and YOLOv5x. These versions are classified upon to the convolution modules' width and depth when they are applied on different applications and hardware.

2.2.3 YOLOv8

Ultralytics, the company behind YOLOv5 [9], introduced YOLOv8 [8] in January 2023. This new version expands support in for a variety of computer vision tasks. While its backbone architecture remains largely similar to YOLOv5, key modifications have resulted in enhanced detection accuracy. To further boost performance, particularly in detecting smaller objects, YOLOv8 incorporates advanced loss functions such as Complete Intersection over Union (CIoU) [12] for bounding-box regression and Distribution Focal Loss (DFL) [13] for improved localization accuracy. Classification tasks utilize binary cross-entropy as the loss function [9]. Like its predecessor, YOLOv8 comes in five scalable versions: YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x, each offering varying levels of speed and accuracy, similar to the scaling options provided in YOLOv5. These versions allow users to have choices depending on computational efficiency and model performance for specific application needs.

3 Result and Discussion

3.1 Results analysis

While YOLOv1 has become outdated compared to its successors, it serves as a crucial reference point for understanding the advancements in the YOLO series. YOLOv5, for instance, represents a significant evolution from YOLOv1, incorporating numerous architectural optimizations that have enhanced its performance. However, it is important to note that during tests on the VHR-10 dataset, YOLOv5 exhibited greater variability in recall rate compared to YOLOv1, indicating some trade-offs in the pursuit of improved accuracy and precision.

Object detection models are commonly evaluated using metrics such as Recall, Precision, Accuracy, and the F1 score, which are derived from four key values: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). These metrics offer a detailed view of how effectively a model identifies objects and minimizes errors. In Table 1, a comparison between YOLOv5 and YOLOv8 is shown for car detection. Although YOLOv8 generally surpasses YOLOv5 in most performance aspects, it performed worse in this specific task. This indicates that despite overall improvements, it still encounters challenges in certain contexts, underscoring the complexities and ongoing refinements in object detection models.

Table 1. The result of the YOLOs.

YOLO	FN	TN	Precision	Recall	F1	TP	Accuracy	FP
YOLOv5	61	35	0.947	0.969	0.958	1105	0.928	139
YOLOv8	80	64	0.931	0.944	0.937	1086	0.894	129

3.2 Discussion

It is intriguing to observe that, despite the continuous advancements in the YOLO series, the latest versions are not always unequivocally superior to their predecessors. As evidenced by Table 1, earlier versions like YOLOv5 can sometimes demonstrate greater stability or even outperform newer versions like YOLOv8 in specific scenarios, such as car detection. This suggests that the progression of YOLO models is not a linear path of improvement, but rather a series of trade-offs and optimizations tailored to various tasks.

Research has indicated that factors such as batch size can significantly impact the performance of these models, further complicating the comparison between different YOLO versions. For instance, varying batch sizes can lead to fluctuations in accuracy, recall, and other key metrics, underscoring the importance of fine-tuning the models according to the specific conditions and requirements of each application.

Given these insights, future development in the YOLO series may increasingly focus on creating diversified versions of the model, each optimized for particular tasks or datasets. This approach would allow for greater flexibility and adaptability, enabling the YOLO framework to meet a broader range of needs across different use cases and industries.

4 Conclusion

This study provides a comprehensive overview of object detection, highlighting the commonly used datasets and offering a comparative analysis of three various versions of the YOLO algorithm: YOLOv1, YOLOv5, and YOLOv8. The analysis delves into the specifics of each algorithm, including their architectures and the datasets on which they were tested. Through this comparison, the study demonstrates that while newer versions of YOLO introduce significant improvements, they are not universally superior to their predecessors. In fact, older versions may outperform newer ones under certain conditions, as evidenced by variations in results such as those influenced by batch size adjustments. The discussion section further evaluates these findings, underscoring the importance of fine-tuning architectural elements to optimize performance for specific tasks. This insight reveals that the evolution of YOLO models is not merely about creating more powerful algorithms but also about adapting them to meet specialized needs. Looking ahead, future research will focus on exploring the performance of various YOLOv8 variants. By investigating five different variants, the aim is to identify configurations that offer the most appropriate balance of speed, accuracy, and stability for diverse applications. This continued exploration will help refine the YOLO framework, making it even more versatile and effective across a broader range of object detection challenges.

References

1. A. Vijayakumar & S. Vairavasundaram, Yolo-based object detection models: A review and its applications. Multimedia Tools and Applications, 1-40 (2024)
2. A. Benjumea, I. Teeti, F. Cuzzolin & A. Bradley, YOLO-Z: Improving small object detection in YOLOv5 for autonomous vehicles. arxiv preprint:2112.11798 (2021)
3. H. Ma, T. Celik, H. Li, Fer-yolo: Detection and classification based on facial expressions. In Proceedings of the Image and Graphics: International Conference, 6(8), 28–39 (2021)
4. P. Jiang, D. Ergu, F. Liu, Y. Cai & B. Ma, A Review of Yolo algorithm developments. Procedia computer science, 199, 1066-1073 (2022)

5. T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan & C.L. Zitnick, Microsoft coco: Common objects in context. In Computer Vision–ECCV Proceedings, 740-755 (2014)
6. W. Zhiqiang & L. Jun, A review of object detection based on convolutional neural network. In Chinese control conference, 11104-11109 (2017)
7. J. Terven, D.M. Córdova-Esparza & J.A. Romero-González, A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. Machine Learning and Knowledge Extraction, 5(4), 1680-1716 (2023)
8. G. Jocher, "YOLOv5 by Ultralytics", 2020, Retrieved on 2024, Retrieved from: https://github.com/ultralytics/yolov5
9. X. Yu, T. W. Kuan, et. al. Yolo v5 for sdsb distant tiny object detection. In 2022 10th International Conference on Orange Technology (ICOT), 1-4 (2022)
10. W. Wu, H. Liu, L. Li, Y. Long, X. Wang, Z. Wang & Y. Chang, Application of local fully Convolutional Neural Network combined with YOLO v5 algorithm in small target detection of remote sensing image. PloS one, 16(10), e0259283 (2021)
11. X. Lu, B. Wang, X. Zheng, Sound Active Attention Framework for Remote Sensing Image Captioning. IEEE Transactions on Geoscience and Remote Sensing, 58(3): 1985–2000 (2020)
12. Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, D. Ren, Distance-IoU loss: Faster and better learning for bounding box regression. In Proceedings of the AAAI Conference on Artificial Intelligence, 34, 12993–13000 (2020)
13. X. Li, W. Wang, L. Wu, S. Chen, X. Hu, J. Li & J. Yang, Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. Advances in Neural Information Processing Systems, 33, 21002-21012 (2020)