Comparative Analysis of CNN-Based Object Detection Models: Faster R-CNN, SSD, and YOLO

Zhenyi Su *

School of Electronic and Computer Engineering, Xiamen University Malaysia Xiamen, Sepang, Selangor Darul Ehsan, Malaysia

* Corresponding Author Email: CST2209163@xmu.edu.my

Abstract. Target detection is widely used in the current environment. With the rapid development of deep learning, innovative models like Convolutional Neural Networks (CNNs) were born. CNNs have been widely used in many practical applications for object detection, since CNNS outperform traditional models in terms of speed and accuracy. This paper first introduces three well-known CNN-based target detection models: Region-based Convolutional Neural Network (Faster R-CNN), Single Shot Multibox Detector (SSD) and You Only Look Once (YOLO). Then this paper gives data on speed, accuracy and resource consumption. Based on these data, the advantages and disadvantages of these three models in different scenarios are systematically and comprehensively evaluated and analyzed. The purpose of the paper is to give researchers a deep understanding of the different characteristics of the three models and which model should be used best in what situations by examining their strengths and limitations. Finally, this paper analyzes the different characteristics of the three models to facilitate the researcher's possible subsequent improvements.

Keywords: Object Detection, Convolutional Neural Networks, Region-based Convolutional Neural Network, Single Shot Multibox Detector, You Only Look Once.

1. Introduction

In the field of computer vision, object detection stands as a crucial core problem. It aimed at identifying specific objects within images or videos and determining where they are. Object detection is extensively applied across various industries, including autonomous driving, industrial automation, surveillance systems, and medical imaging. For instance, object detection is instrumental in identifying abnormal events or behaviors in surveillance systems. In industrial automation, it is used to pinpoint the location or size of different parts. And it enables machines to perform high-quality tasks with precision. In autonomous driving, vehicles need to detect surrounding obstacles and pedestrians in real time to navigate safely. In medical imaging, object detection also plays a role in assisting doctors in quickly locating abnormalities.

After AlexNet took the top position in the ImageNet competition in 2012. Deep learning technology represented CNNs developed rapidly. It leads to a boom in object detection technology as well [1]. Traditional object detection methods often relied on handcrafted feature engineering and machine learning algorithms. While these methods performed reasonably well in simple scenarios, they struggled with large-scale datasets. In contrast, CNNs is able to continuously improve detection accuracy through training on massive datasets. Moreover, detection speed of CNNs is significantly enhanced by the parallel computing capabilities. It enabling the possibility of real-time applications in object detection.

It is significant to choose the appropriate detection algorithm. Because different application scenarios have different requirements for different detection algorithms For example, detection accuracy and robustness are critical in industrial automation. Any mistake in recognizing product shapes or identifying part locations could negatively impact product quality. Therefore, detection models must offer extremely high precision and stability to adapt to the complex industrial environment.

It is very meaningful to compare different CNN models in terms of accuracy, speed, and resource consumption. Such comparative studies allow researchers and developers to determine which model

is best suited for a given scenario. Researchers can understand the strengths and weaknesses of various architectures by analyzing the differences between models, such as their accuracy, speed, and resource usage. In some cases, certain models excel at detecting small objects, while others may be better at handling larger ones. This analysis helps guide future algorithm optimization and development.

To improve detection speed, Liu et al. introduced a new model called SSD [2]. SSD employs multi-scale feature maps and default boxes. It significantly boosts detection speed without compromising accuracy. SSD completes both object localization and classification in a single network forward pass. It simplifies the candidate region generation process and makes it highly suitable for large and real-time tasks.

Additionally, Redmon proposed a model to further enhance real-time detection capabilities. It's called YOLO model. YOLO turns the detection task into a simple regression problem [3]. It also achieves efficient simultaneous predictions of object classes and locations through just one forward pass of the network. YOLO's high speed makes it suitable for real-time applications. However, YOLO performs poorly when detecting small objects in complex scenes and sometimes even miss detections, especially in dense environments.

Overall, each model has its unique strengths and limitations. It drives researchers to conduct more systematic comparisons and analyses in order to find optimal solutions for different application scenarios.

The focus of this paper is to conduct a systematic performance evaluation of several prominent CNN models by comparing their speed, accuracy, and resource consumption. Specifically, The paper will compare classic object detection models: Faster R-CNN, SSD, and YOLO. The paper comprehensively assesses their performance through related technical concepts and data analysis in various scenarios. The goal is to assist researchers and engineers. There are suitable models to choose from in different scenarios.

2. Related Technical Concepts

2.1. Faster R-CNN

Region-based Convolutional Neural Network (Faster R-CNN) represents a significant advancement comparing R-CNN and Fast R-CNN in the past. It introduced the Region Proposal Network (RPN). RPN allows the system to generate region proposals directly. The need for an external algorithm such as selective search is reduced [4]. This integration not only reduces computational overhead but also enhances the accuracy and speed of region proposals. It completely speeds up the overall detection process. Faster R-CNN employs a two-stage approach. Firstly, the RPN generates candidate regions, and then the classifier categorizes these regions. Although it achieves high accuracy, Faster R-CNN has significant computational demands. As a result, they usually require powerful hardware to operate. It prevents them from becoming popular in real-time applications

2.2. SSD

Single Shot MultiBox Detector (SSD) is unlike the two-stage nature of Faster R-CNN. It adopts a single-stage methodology. It has the ability to increasing detection speed greatly by executing both localization and classification in one forward pass through the network[2]. SSD utilizes a series of feature maps to predict bounding boxes, which enables it to handle objects of varying scales effectively. SSD strikes a good balance between speed and accuracy by incorporating predictions from multiple feature levels. SSD is effective in detecting objects at different scales. But SSD may struggle with small objects in complex scenes. Because of its use of relatively large default anchor boxes. And it can lead to lower localization precision. This focus on efficiency makes SSD well-suited for applications that require a trade-off between speed and accuracy. Such as mobile applications or real-time video processing.

2.3. YOLO

You Only Look Once (YOLO) redefines the object detection problem by framing it as a regression task rather than a classification task. Unlike Faster R-CNN and SSD. YOLO takes the entire image into account during detection. It divides the image into a fixed grid and simultaneously predicting bounding boxes and class probabilities for each cell [3]. This approach allows YOLO to have both impressive real-time performance and competitive accuracy. YOLO's unique architecture significantly reduces inference time. Thus, it is suitable for time-sensitive applications, such as video surveillance and real-time target tracking. However, early YOLO gave up some accuracy in order to ensure extremely high speed [5]. In particular, YOLO does not perform well when faced with overlapping or densely packed objects. Because its grid-based approach struggled to accurately predict small or crowded items [6]. Later versions such as YOLOv3 solve many of the above problems [7].

3. Detailed Comparative Analysis of Faster R-CNN, SSD, and YOLO in Terms of Accuracy, Speed, and Resource Consumption

This section presents a thorough comparison of Faster R-CNN, SSD, and YOLO across three crucial aspects: accuracy, speed, and resource consumption.This section lists and analyzes the data. And providing an understanding of the strengths and limitations of these models. The experimental results are presented in tables following each subsection.

3.1. Accuracy

Based on the gathered data, SSD has demonstrated itself as the most precise model. It boasts an mAP of 76.8%, outperforming both Faster R-CNN and YOLO in terms of detection accuracy. The experiments revealed that SSD excels particularly in detecting medium and large objects. This advantage contributes to its higher accuracy. Also, SSD achieves an excellent balance between speed and accuracy. It has the ability to maintain relatively high precision without compromising speed. This makes SSD a popular choice for applications where target sizes vary but high precision remains essential. Such as in traffic monitoring and public safety.

Faster R-CNN achieves a relatively high accuracy of 73.2%. Compared to SSD and YOLO, Faster R-CNN employs a two-stage detection method. The two-stage detection method gives it an inherent advantage in detection accuracy. This is particularly true when dealing with overlapping or partially occluded objects and small objects. At this point Faster R-CNN outperforms the other two detection models significantly. Therefore, Faster R-CNN is more suitable for applications requiring high accuracy. For instance, in clinical diagnostics in the medical field. The cost of speed and resource consumption is less critical but precision is paramount. In such scenarios, Faster R-CNN's advantages over the other two models become particularly pronounced.

Finally, YOLO has shown significant improvements in accuracy when it is compared to its earlier versions. It has an mAP of 63.4%. From the Table 1[2-4], it is evident that YOLO has the lowest accuracy among the three models. This may be attributed to YOLO's non-maximum suppression (NMS) possibly removing some closely located bounding boxes. And it results in suboptimal performance when handling densely packed objects. Although its precision lags behind SSD and Faster R-CNN, YOLO performs well in situations that demand rapid responses. YOLO's real-time detection capabilities come at the cost of a slight decrease in accuracy. Additionally, the lower resolution of its feature maps reduces its ability to detect small objects. In practical applications, there is even a possibility of missed detections.

It is also important to note that while SSD and Faster R-CNN achieve higher accuracy, YOLO provides a different trade-off by balancing accuracy with other factors. The specific strengths of each model largely determine their areas of applicability.

Table 1. Precision Comparison of Faster R-CNN, YOLO and SSD. Data: ”07+12”: union of VOC2007 and VOC2012 trainval.

Method	data	Mean Average Precision (mAP)
Faster R-CNN	07+12	73.2
SSD	07+12	63.4
YOLO	07+12	76.8

3.2. Speed

Detection speed is a critical factor for many real-time applications. In this section, the speed of each model is evaluated by measuring frames processed per second (FPS) using a standardized GPU setup.

Among the three models, YOLO is the fastest, with an average speed of 45 FPS from Table 2[2-4]. Such data indicates that YOLO is suitable for applications like drone operations. the priority of model response speed is higher than detection accuracy in these applications. In drone applications, drones equipped with YOLO can perform real-time object detection during flight. This enables drones to effectively recognize their surroundings with low latency during flight and enhances their ability to respond to changes in the external environment.

Compared to YOLO, SSD's detection speed is not as outstanding. SSD delivers an average speed of 30 FPS. Although not as fast as YOLO, it still maintains good speed while maintaining moderate accuracy. This balance makes SSD an excellent choice for real-time detection tasks with relatively high accuracy requirements. Such as handheld devices or augmented reality applications. It’s crucial to strike a balance between performance and user experience in these applications. In augmented reality (AR), SSD's detection speed allows objects in the real world to be identified and overlaid with digital information quickly enough to create an immersive experience for users without disrupting interactivity.

On the other hand, Faster R-CNN operates at an average speed of only 7 FPS. It makes it the slowest of the three models. Although it provides high accuracy, its inference speed is noticeably slower, limiting its applicability in real-time scenarios. As a result, Faster R-CNN is better suited for tasks that demand high accuracy but don’t require real-time detection, such as medical image analysis. In medical diagnostics detailed images need to be analyzed with the utmost precision. Faster R-CNN's slower speed is less of a concern compared to its ability to accurately identify abnormalities in complex medical scans.

Overall, YOLO's high FPS provides an advantage in applications where real-time analysis is essential, while SSD serves well in mixed-use environments requiring both speed and accuracy. Faster R-CNN's slower speed may seem like a disadvantage, but its accuracy makes it indispensable in fields where precise detection is non-negotiable.

Table 2. Speed Comparison of Faster R-CNN, YOLO and SSD.

Method	data	Frames Per Second (FPS)
Faster R-CNN	07+12	7
SSD	07+12	45
YOLO	07+12	19

3.3. Resource Consumption

Resource consumption is another key factor, especially when deploying object detection models on devices with limited computational capabilities [8,9]. In this paper, resource usage is evaluated based on GPU VRAM requirements and floating-point operations per second (FLOPS). It reflects each model's memory and computational demands.

Based on the collected data, Faster R-CNN has the highest resource consumption. It requires about 11 GB of GPU VRAM during inference from Table 3[2-4]. Such high resource demands limit its

deployment to environments with constrained hardware capabilities. Thus, it is more suitable for use in cloud or high-performance workstations. Its FLOPS requirements further highlight the need for powerful computing resources. Faster R-CNN's resource-intensive nature is one of the main reasons it is predominantly used in applications where computational power is not a limitation. In research environments or large-scale industrial applications, the availability of advanced hardware ensures that its high resource consumption does not hinder its deployment. Therefore, Faster R-CNN is more suitable for deployment in such environments.

As a single-stage detector, SSD is inherently designed to reduce resource consumption. This is clearly reflected in the data, where SSD requires only 6GB of GPU VRAM, significantly less than Faster R-CNN. This makes SSD more deployable in environments with moderate hardware capabilities, such as industrial automation. In industrial automation computational efficiency and accuracy are both necessary. SSD achieves an effective balance among them and enables deployment on existing hardware without the need for costly upgrades. The relatively lower memory requirements also make SSD more accessible for projects on a tighter budget.

By contrast, YOLO has the lowest resource consumption. It needs around 4.5 GB of GPU VRAM for inference. This emphasizes YOLO's efficiency and makes it ideal for deployment in resource-constrained environments. Such as embedded systems, mobile devices, or edge computing platforms. Its lower FLOPS and VRAM demands allow it to be widely utilized in applications where hardware is limited. YOLO's ability to be applied in the aforementioned drone applications is precisely due to its compact architecture. It significantly reduces computational resource requirements. This allows YOLO to perform real-time object detection without compromising the functionality of the device.

Moreover, YOLO's efficiency also makes it attractive for use in edge computing, where latency and resource constraints are critical considerations. It ensures that data processing happens close to the source of data collection. Thus, reducing the need for constant connectivity to a central server and improving response times.

Table 3. Resource Consumption Comparison of Faster R-CNN, YOLO and SSD.

Method	GPU VRAM Usage (GB)	FLOPS (GigaFLOPS)
Faster R-CNN	11.0	256
SSD	4.5	127
YOLO	6.0	148

The comparative analysis of Faster R-CNN, SSD, and YOLO from Table 4 provides a comprehensive understanding of their strengths and weaknesses in terms of accuracy, speed, and resource consumption[2-4]. Firstly, Faster R-CNN is best suited for tasks demanding high precision, such as detailed image analysis or medical diagnostics. But its need for considerable computational resources makes it unsuitable for real-time applications. Secondly, SSD strikes a balance between speed and precision with its top-notch accuracy. Therefore, SSD is applicable to a wide range of use cases that require moderate accuracy and responsiveness. Finally, YOLO, focused on speed and minimal resource requirements, excels in real-time detection tasks, making it perfect for applications requiring quick responses, though it sacrifices some accuracy in the process.

Table 4. Comprehensive Performance Comparison of Faster R-CNN, YOLO and SSD.

Method	data	GB	FLOPS	mAP	FPS	Batch size
Faster R-CNN	07+12	11.0	256	73.2	7	1
SSD	07+12	4.5	127	63.4	45	1
YOLO	07+12	6.0	148	76.8	19	1

4. Conclusion

This paper begins with an introduction to the centerpiece of this paper: target detection techniques. It then describes the phenomenon of target detection becoming more and more important in today's general trend. On this basis, this paper starts to introduce CNNs models which are very pioneering in

the field of target detection. From this, three improved models based on CNNs, Faster R-CNN, SSD, and YOLO, are introduced. After this, this paper begins a detailed description of the three models. Describing their model characteristics and the corresponding domains of application. After concluding the introductory part, this paper introduces three types of data on speed, accuracy, and resource consumption. It aimed to evaluate and analyze the different strong points of these three models in a multidimensional way. Finally, the paper imports the advantages of Faster R-CNN, SSD and YOLO in different aspects.

Firstly, Faster R-CNN has high accuracy and excellent adaptability in different scenarios, but high resource consumption and slow detection speed. The high accuracy plays a significant role in scenarios where precision is of utmost importance. Like medical diagnostics and detailed image analysis. Everything has its pros and cons. The high accuracy comes with a slower inference speed. In addition to that, bumper resource requirements limit its application in real-time environments.

Secondly, SSD is relatively balanced. It has a high level of accuracy while maintaining a high level of detection speed. Its single-stage architecture significantly reduces computation time and its multi-scale feature maps enable efficient detection of objects at varying scales. What's rare is that ssd maintains high speed and accuracy without consuming too much computing resources. Thus, it is suitable for industrial automation and augmented reality applications. In these applications, both speed and accuracy are essential but not at the cost of extensive computational resources.

Finally, YOLO has the fastest response time and lowest resource consumption of the three models. In practice, it can be used in the environments with limited computing power like Drone applications. Although YOLO may not achieve the same accuracy as Faster R-CNN or SSD, its high speed makes it more suitable for time-sensitive tasks.

In conclusion, the choice of an object detection model depends on the specific requirements of the application—whether accuracy, speed, or resource efficiency is the main concern. Understanding these trade-offs is crucial for selecting the most appropriate model to achieve optimal performance in each scenario.

References

[1] Alzubaidi L, Zhang J, Humaidi A J, Al-Dujaili A, Duan Y, Al-Shamma O, ... & Farhan L. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data, 2021, 8, 1 - 74.

[2] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C Y, & Berg A C. SSD: Single shot multibox detector. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11 – 14, 2016, Proceedings, Part I 14, pp. 21 - 37.

[3] Ren S, He K, Girshick R, & Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 39 (6), 1137 - 1149.

[4] Redmon J. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[5] Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, & Zagoruyko S. End-to-end object detection with transformers. In European Conference on Computer Vision, Springer International Publishing, 2020, pp. 213 - 229.

[6] Bochkovskiy A, Wang C Y, & Liao H Y M. YOLOv4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934, 2020.

[7] Redmon J. YOLOv3: An incremental improvement. arXiv preprint arXiv: 1804.02767, 2018.

[8] Lin T Y, Dollár P, Girshick R, He K, Hariharan B, & Belongie S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2117 - 2125.

[9] Tan M, Pang R, & Le Q V. EfficientDet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10781 - 10790.