Library Progress International *Print version ISSN 0970 1052*
Vol.44 No. 3, Jul-Dec 2024: P. 11271-11284 *Online version ISSN 2320 317X*

Original Article *Available online at www.bpasjournals.com*

Cloud-Based Image Processing Using Deep Learning for Real-Time Object Detection Review

[1]Abdul Razzak Khan Qureshi,[2]Divya Samad,[3]Himanshu Dehariya,[4]Vibha Bairagi [5]Satyendra Kumar Bunkar,[6]Priyanka khabiya

[1]Assistant Professor, Department of Computer Science, Medi-Caps University, Indore, MP, India dr.arqureshi786@gmail.com
[2]Assistant Professor, Department of Computer Applications, Medi-Caps University, Indore, MP., India,divya.samad@gmail.com
[3]Assistant Professor, Department of Computer Applications, Medi-Caps University, Indore, MP., India,himanshu.dehariya@gmail.com
[4]Lecturer, Department of Computer Science, Medi-Caps University, Indore, MP, India vibha.rakesh.bairagi@gmail.com
[5]Associate Professor, Department of Computer Science and Engineering, Chameli Devi Group of Institutions, Indore,satyendra.bunkar@gmail.com
[6]Assistant professor, Department of Computer Science and Engineering, Mandsaur University, Mandsaur, M.P., India,khabiya198727@gmail.com

How to cite this article: Abdul Razzak Khan Qureshi,Divya Samad,Himanshu Dehariya,Vibha Bairagi Satyendra Kumar Bunkar,Priyanka khabiya (2024) Cloud-Based Image Processing Using Deep Learning for Real-Time Object Detection Review. *Library Progress International,* 44(3), 11271-11284.

ABSTRACT

The growing demand for real-time image processing in various industries has led to the integration of cloud-based platforms with deep learning technologies. This paper proposes a cloud-based framework that utilizes deep learning models for real-time object detection, offering a scalable and efficient solution for high-performance image analysis. The proposed system leverages cloud infrastructure for enhanced computational power and storage, while utilizing convolutional neural networks (CNNs) to improve object detection accuracy. The architecture allows for real-time processing, enabling applications in fields such as autonomous vehicles, surveillance, and industrial automation. Experimental results demonstrate the system's ability to detect multiple objects with high accuracy and low latency. This approach significantly reduces the hardware limitations traditionally associated with on-premise solutions, making it highly applicable to large-scale deployments. The paper also explores potential challenges related to latency, data privacy, and bandwidth requirements, proposing solutions for these issues.

Keywords: *Cloud computing, Deep learning, Real-time object detection, Convolutional neural networks, Image processing, Autonomous systems*

1.1. Introduction

The rapid advancement in computer vision and artificial intelligence (AI) has sparked growing interest in the development of real-time object detection systems, which are critical for applications such as autonomous vehicles, smart surveillance, industrial automation, and augmented reality. Object detection involves not only identifying objects within an image or video frame but also determining their precise locations via bounding boxes. In recent years, deep learning, particularly Convolutional Neural Networks (CNNs), has emerged as the dominant technology for object detection, significantly outperforming traditional techniques. However, real-time object detection poses several challenges, particularly in terms of computational complexity, accuracy, and scalability. This paper proposes a cloud-based deep learning framework designed to address these challenges by leveraging cloud computing for real-time object detection, offering scalable and efficient solutions that can be deployed across various industries.

1.1 Background and Motivation

Traditional object detection methods, such as the Histogram of Oriented Gradients (HOG) and Deformable Parts Model (DPM), have been surpassed by deep learning-based approaches due to the latter's superior accuracy and adaptability in complex environments. Early deep learning models for object detection, like Region-based Convolutional Neural Networks (R-CNN), made significant strides but were too slow for real-time applications due to their multi-stage detection processes. The need for faster and more efficient models led to the development of frameworks such as You Only Look Once (YOLO) and Single Shot MultiBox Detector (SSD), which process entire images in a single pass, offering real-time performance. Despite these advancements, deploying real-time object detection on edge devices, such as cameras, drones, or robots, remains challenging due to the high computational demands of deep learning models. Edge devices often lack the hardware capabilities required for running complex deep learning models, particularly when it comes to high-resolution image processing. This leads to a trade-off between speed and accuracy, as well as the issue of scalability when multiple devices are deployed in large-scale systems. Cloud computing has emerged as a powerful tool for overcoming these challenges by providing virtually unlimited computational resources and storage capabilities. With the integration of cloud-based services, it is possible to offload intensive tasks like object detection to cloud servers, thereby enabling real-time processing even on resource-constrained edge devices. Cloud platforms, such as Amazon Web Services (AWS), Google Cloud, and Microsoft Azure, provide specialized environments for machine learning tasks, including pre-configured instances with GPUs or TPUs optimized for deep learning inference. The cloud's ability to dynamically scale resources based on workload further enhances its suitability for real-time applications.

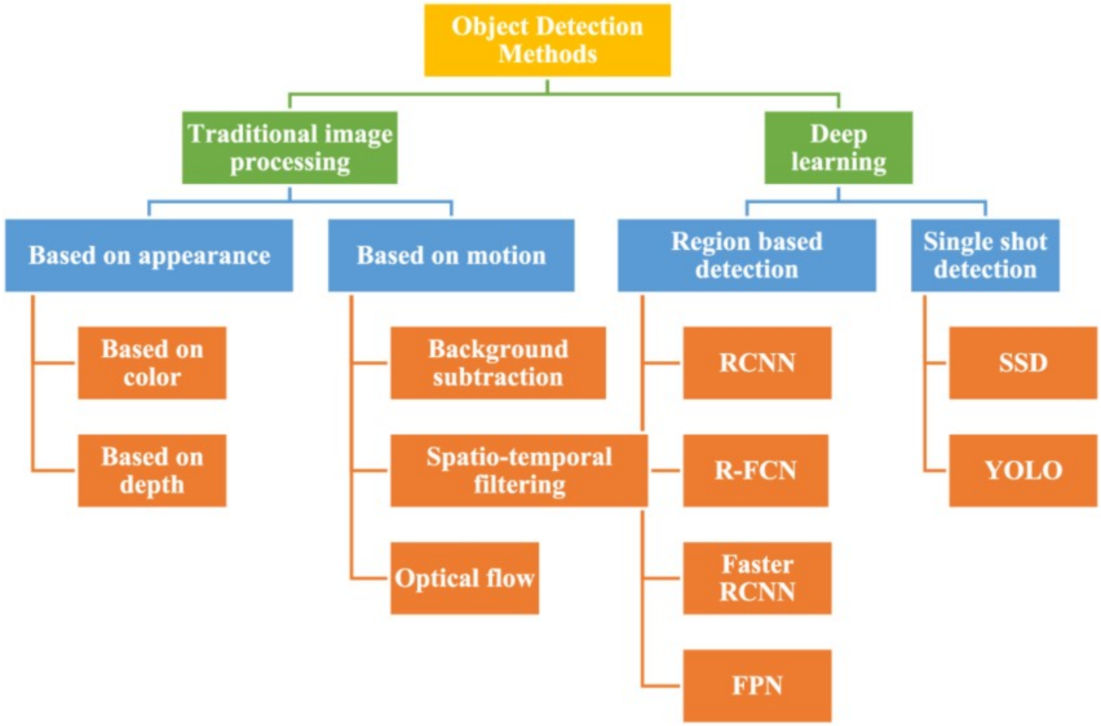

Figure.1: Classification of object detection methods.

1.2 Problem Statement

While cloud-based systems offer significant advantages for deep learning and real-time object detection, several challenges remain. Network latency is a major concern, particularly when transmitting large image or video data to and from the cloud. This latency can undermine the real-time performance required by applications such as autonomous driving or surveillance, where delays can lead to critical errors. Additionally, concerns around data privacy and security are amplified when using cloud services, especially when dealing with sensitive or personal data, as is the case with surveillance footage or healthcare imagery. The cost of cloud services is another factor that must be considered, particularly in applications that require continuous processing of high volumes of data. While the cloud can reduce hardware limitations on edge devices, the financial implications of constant data transmission and GPU utilization in the cloud can be substantial. Furthermore, real-time object detection demands a balance between speed and accuracy, which often means choosing models that sacrifice one for the other. This paper addresses these issues by proposing a cloud-based real-time object detection framework that leverages deep learning models. By utilizing cloud infrastructure, the framework is designed to scale efficiently, handle high-resolution image processing, and reduce the burden on edge devices. The proposed system incorporates techniques to minimize network latency and improve data security, offering a practical solution for real-time object detection across a variety of industries.

1.3 Objectives of the Study

The primary objective of this research is to develop a cloud-based framework for real-time object detection that is both scalable and efficient. Specifically, the paper aims to:

1. **Leverage deep learning models**: Utilize advanced deep learning models, such as Convolutional Neural Networks (CNNs), to achieve high accuracy in object detection while maintaining real-time processing speeds.
2. **Integrate cloud computing**: Use cloud infrastructure to offload computationally intensive tasks from edge devices, allowing for real-time performance without the need for expensive hardware at the edge.
3. **Address latency and bandwidth issues**: Propose solutions for minimizing the latency introduced by data transmission to and from the cloud, while maintaining real-time performance.
4. **Ensure data privacy and security**: Explore techniques for ensuring data privacy and security in cloud-based systems, particularly in applications that involve sensitive information.
5. **Optimize cost-efficiency**: Examine ways to optimize the cost of using cloud services for real-time object detection, including resource scaling and efficient model selection.
6. **Validate performance**: Experimentally validate the performance of the proposed system in real-world scenarios, demonstrating its applicability in fields such as autonomous vehicles, surveillance, and industrial automation.

1.4 Significance of the Study

The integration of cloud computing with deep learning for real-time object detection has the potential to revolutionize several industries. Autonomous vehicles, for instance, rely on fast and accurate object detection to navigate their environments safely. Cloud-based object detection frameworks could also enable the widespread adoption of smart surveillance systems capable of real-time monitoring in urban areas, enhancing public safety and crime prevention. In industrial automation, real-time object detection is essential for tasks such as quality control, robot navigation, and worker safety. By offloading the computational burden to the cloud, manufacturers can implement advanced AI systems without requiring expensive, high-performance hardware on-site. Moreover, the scalability of cloud platforms means that the same system can be deployed across multiple sites or devices, streamlining operations and reducing costs. Furthermore, this study addresses several technical challenges that currently limit the practical deployment of real-time object detection systems, such as latency, data security, and cost. By providing a detailed analysis of these issues and proposing solutions, the paper contributes to the broader body of research on cloud-based AI systems and their applications.

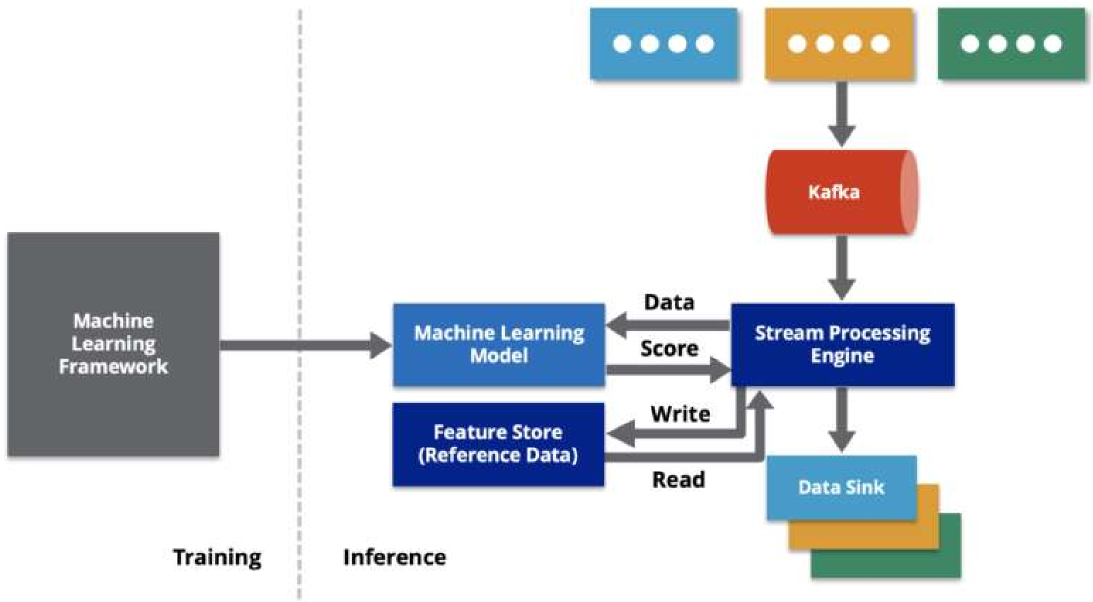

Figure.2: A common way to deploy a real-time machine learning model to production is in an event-driven architecture, in which a data stream

1.5 Structure of the Paper

The remainder of this paper is organized as follows: Section 2 provides a comprehensive literature review on deep learning and object detection, focusing on recent advancements and the integration of cloud computing for real-time applications. Section 3 outlines the proposed system architecture, detailing the deep learning models and cloud-based infrastructure used for real-time object detection. Section 4 presents the experimental setup and results, demonstrating the system's performance in real-world scenarios. Finally, Section 5 discusses the implications of the findings, potential applications, and future research directions, followed by a conclusion in Section 6.

1.1. Literature Review

1. Introduction to Object Detection in Deep Learning

Object detection, a critical task in computer vision, involves identifying and localizing objects within images. It has become increasingly significant due to its applications in fields like autonomous driving, surveillance, robotics, and augmented reality. The advent of deep learning has revolutionized object detection by improving accuracy and efficiency. Traditional approaches like the Histogram of Oriented Gradients (HOG) and the Deformable Parts Model (DPM) have been replaced by deep learning-based models, such as Convolutional Neural Networks (CNNs), which offer better performance and scalability. This section reviews the key developments in deep learning for object detection, focusing on the integration of cloud computing for real-time applications.

2. Deep Learning for Object Detection

Deep learning has led to significant advances in object detection through the development of specialized architectures. One of the first breakthroughs was the Region-based Convolutional Neural Networks (R-CNN) family, introduced by Girshick et al. (2014). R-CNN models perform object detection in two stages: first, they generate region proposals using algorithms like selective search, and then classify each proposal using a CNN. R-CNNs laid the foundation for later models but suffered from slow inference times due to the need for multiple CNN passes over each region. To overcome this limitation, Girshick (2015) introduced **Fast R-CNN**, which speeds up detection by sharing computation across regions and using a single forward pass through the CNN. The introduction of **Faster R-CNN** by Ren et al. (2015) further improved speed by integrating a Region Proposal Network (RPN) that shares features with the CNN backbone, thus eliminating the need for a separate proposal generation step. These models greatly improved object detection accuracy but still posed challenges for real-time applications due to their relatively slow inference times.

3. Real-Time Object Detection: YOLO and SSD

To address the need for real-time object detection, **You Only Look Once (YOLO)**, introduced by Redmon et al. (2016), proposed a single-stage detection approach that directly predicts object classes and bounding boxes from the input image in one pass. YOLO frames object detection as a regression problem, allowing it to achieve real-time performance without sacrificing much accuracy. Later versions, including YOLOv3 and YOLOv4, further optimized the architecture for speed and accuracy, making YOLO one of the most popular frameworks for real-time applications. Similarly, **Single Shot Multibox Detector (SSD)**, introduced by Liu et al. (2016), is another single-stage detector that improves real-time detection. SSD eliminates the need for region proposal stages by predicting bounding boxes and object categories directly from multiple feature maps at different scales. This multiscale approach improves detection performance for objects of varying sizes while maintaining real-time speeds. Both YOLO and SSD have been widely adopted in applications requiring real-time object detection, such as surveillance, robotics, and autonomous systems. However, these models often struggle with small object detection and complex scenes, presenting opportunities for further refinement.

4. Cloud Computing and Image Processing

The scalability and computational demands of deep learning models have led to the increasing adoption of cloud-based platforms for image processing tasks. Cloud computing offers vast computational power, storage capabilities, and flexibility for deploying and managing machine learning models at scale. This is especially important for real-time object detection, where the high computational cost of deep learning models can be a bottleneck on edge devices or personal computers. Cloud platforms, such as Amazon Web Services (AWS), Google Cloud, and Microsoft Azure, provide specialized services for machine learning and deep learning tasks, including **GPU/TPU instances**, **auto-scaling**, and **serverless architectures**. These platforms enable the training, deployment, and inference of deep learning models on large datasets in real-time. For example, the combination of cloud computing with object detection models like YOLO and Faster R-CNN can significantly reduce latency and increase throughput, making real-time applications feasible. Researchers like Zhang et al. (2020) have explored cloud-based frameworks for real-time object detection, demonstrating that cloud platforms can effectively manage the computational overhead while providing robust solutions for high-performance tasks like video surveillance, autonomous navigation, and smart city applications.

5. Integration of Cloud and Deep Learning for Real-Time Object Detection

Recent works have explored the integration of cloud computing with deep learning models for real-time image processing. Cloud-based object detection systems allow for continuous data streams from edge devices, such as cameras or drones, to be processed and analyzed in real time. This setup offers several advantages, including:

- **Scalability**: Cloud platforms can scale resources dynamically based on the workload, enabling the processing of large datasets and high-definition video streams.
- **Cost-efficiency**: By offloading computation to the cloud, edge devices can reduce power consumption and hardware costs.
- **Latency management**: The main challenge in cloud-based systems is network latency, which can introduce delays in processing time-sensitive data. Some researchers have proposed edge-cloud hybrid architectures to minimize latency by performing initial processing on the edge device and offloading complex tasks to the cloud.

In their study, Chen et al. (2021) proposed an edge-cloud collaborative framework for real-time object detection in smart cities. Their framework reduced latency by performing lightweight processing on the edge, while the cloud handled more computationally intensive tasks, such as deep learning inference and model updates.

6. Challenges in Cloud-Based Object Detection

While cloud-based systems offer many advantages, there are also challenges associated with their implementation:

- **Latency and Bandwidth**: Real-time applications are sensitive to latency, and transmitting large image or video data to the cloud can introduce delays, especially in environments with limited bandwidth. To address this, some systems use compression techniques or perform preliminary processing on the edge before sending data to the cloud.
- **Data Privacy and Security**: Cloud-based systems often handle sensitive data, such as video feeds from surveillance cameras. Ensuring data privacy and security in transit and at rest is critical for widespread adoption. Privacy-preserving techniques like encryption and federated learning are being explored to mitigate these concerns.
- **Cost of Cloud Services**: Continuous real-time processing in the cloud can incur significant costs, especially when using GPU/TPU instances for deep learning inference. Cost-effective strategies, such as on-demand resource allocation and efficient model architectures, are necessary to make cloud-based real-time object detection financially viable.

Cloud-Based Image Processing Using Deep Learning: Comparison Table

Below is a detailed comparison table outlining various deep learning techniques used for cloud-based image processing, focusing on their key features, advantages, challenges, and real-time object detection capabilities. This table considers factors such as computational requirements, scalability, real-time performance, and cloud integration.

Technique/ Model	***Architecture***	***Real-Time Performance***	***Cloud Integration***	***Strengths***	***Challenges***	***Common Applications***

R-CNN (Region-based CNN)	Two-stage process: region proposals and CNN-based classification	Low - Time-consuming due to separate region proposal and classification steps	Difficult - Requires significant compute power, not ideal for real-time cloud deployment	High accuracy, first to use CNN for object detection	Slow inference time, computationally expensive	Image classification, offline object detection
Fast R-CNN	Single-stage network with shared feature maps for object detection	Medium - Improved over R-CNN but still not real-time	Moderate - Reduced compute requirements make cloud deployment more feasible	Reduced computation, good accuracy	Still not real-time, requires region proposals	Object detection in slower environments, video analytics
Faster R-CNN	Integrated Region Proposal Network (RPN) with a shared backbone CNN	Medium to Low - Faster than R-CNN but not suitable for high-speed real-time applications	Moderate - Can be deployed on cloud for offline or slower real-time tasks	High accuracy, efficient for large datasets	Still slower than one-stage detectors, complex architecture	Surveillance, large-scale image classification
YOLO (You Only Look Once)	Single-stage detector, direct prediction of bounding boxes and classes	High - Real-time performance due to single-pass architecture	Highly Compatible - Cloud platforms can handle YOLO's computational needs effectively	Fastest among detectors, real-time object detection, relatively good accuracy	Struggles with small object detection, complex scenes	Autonomous vehicles, real-time video analysis, robotics
YOLOv3/YOLOv4	Enhanced YOLO architecture with multiple feature scales and improved bounding box predictions	High - Improved real-time performance with better accuracy over original YOLO	High Compatibility - Efficient for cloud deployment, highly scalable	Excellent speed-accuracy trade-off, can detect objects at different scales	Can still struggle with tiny objects and highly complex environments	Drone vision, autonomous driving, smart surveillance
SSD (Single Shot Multibox Detector)	Single-stage detector using feature maps from multiple layers	High - Comparable to YOLO in real-time performance, slightly more accurate	Highly Compatible - Easily deployable on cloud for real-time tasks	Fast, efficient, good for varying object sizes	Accuracy drops for smaller objects	Smart cameras, real-time monitoring, robotics
RetinaNet	Feature Pyramid Network	Medium - Slower than SSD and	Moderate - Cloud deployment	Handles small and large objects	Slower than single-	Video surveillance, object

	(FPN) with focal loss to handle class imbalance	YOLO due to the use of feature pyramids	possible, but may require more computation resources	effectively, better accuracy for imbalanced datasets	shot detectors like YOLO	detection with class imbalance
MobileNet + SSD/YOLO	Lightweight CNN (MobileNet) combined with object detection models (SSD or YOLO)	High - Optimized for mobile and edge devices, capable of real-time performance	High - Efficient for cloud deployment due to lower computational requirements	Fast, low computation al overhead, suitable for mobile and cloud	Lower accuracy compare d to full-scale models	Mobile applications , real-time surveillance , edge computing
EfficientDet	Scalable architecture combining EfficientNet backbone and BiFPN for object detection	High - Designed for scalability, real-time performance with low resource usage	Highly Compatible - Optimized for both cloud and edge deployment	Excellent balance of speed, accuracy, and resource efficiency	Still newer, less mature than YOLO or SSD, may require fine-tuning	Cloud-based services, scalable object detection systems
DeepLabv3 +	Atrous convolution and fully connected CRFs for semantic segmentatio n	Low - Primarily designed for segmentation, not optimized for real-time object detection	Moderate - Works well in cloud for high-resolution image segmentation tasks	High accuracy for semantic segmentatio n, handles complex scenes	Not suitable for real-time object detection due to slower performa nce	Medical imaging, satellite image processing, urban planning
Faster R-CNN + FPN (Feature Pyramid Network)	R-CNN with a feature pyramid to handle objects at different scales	Medium to Low - Not real-time but can be adapted for cloud-based applications	Moderate - Can benefit from cloud GPU acceleration, not ideal for low-latency tasks	Better at detecting small objects, good accuracy on multi-scale objects	Slower inference time, higher computat ional cost	Medical imaging, large-scale image datasets, offline detection
CenterNet	Keypoint-based approach that detects the center of objects	High - Competitive real-time performance with high accuracy	High - Suitable for cloud deployment with efficient inference	High accuracy, handles complex scenes well, real-time capable	More complex to impleme nt and fine-tune than YOLO	Real-time surveillance , autonomous vehicles, sports analytics
YOLOv5	Latest version of	Very High - State-of-the-	High Compatibility -	Fast, accurate,	May still struggle	Autonomou s systems,

	YOLO with enhancements in accuracy and speed	art in real-time object detection	Designed for ease of use in cloud deployment	real-time performance even with larger models	with fine details in highly cluttered scenes	robotics, video stream analysis

Detailed Explanations:

1. **R-CNN, Fast R-CNN, Faster R-CNN**: These models pioneered the use of CNNs in object detection, with Faster R-CNN being the most efficient among them due to its integrated region proposal mechanism. However, they are not real-time, making them better suited for cloud-based applications where speed is not the primary concern.
2. **YOLO and SSD**: These single-stage detectors revolutionized real-time object detection by processing images in a single pass, making them highly suitable for cloud-based real-time applications. YOLO's speed and SSD's multiscale feature map approach make them the most popular choices for real-time detection tasks in cloud environments.
3. **MobileNet + SSD/YOLO**: Designed for resource-constrained devices, MobileNet combined with SSD or YOLO is highly efficient for cloud-based and mobile real-time applications. It is used extensively in edge computing where power and processing capabilities are limited.
4. **EfficientDet**: A scalable and efficient model, EfficientDet provides an excellent balance between accuracy and speed while using fewer resources, making it ideal for large-scale cloud deployments that require real-time performance.
5. **RetinaNet and FPN-based Models**: These models offer better detection accuracy for smaller and imbalanced objects but are slower compared to YOLO and SSD. They are more suited for applications where accuracy is more critical than speed, such as medical imaging or offline cloud-based object detection.
6. **DeepLabv3+**: While not optimized for real-time object detection, DeepLabv3+ excels in semantic segmentation tasks. It is ideal for cloud-based applications that require precise object delineation, such as satellite imagery analysis or autonomous driving where semantic segmentation complements object detection.
7. **CenterNet**: A keypoint-based detection model that provides competitive real-time performance with high accuracy, CenterNet is gaining popularity in applications such as autonomous driving, sports analytics, and video surveillance.

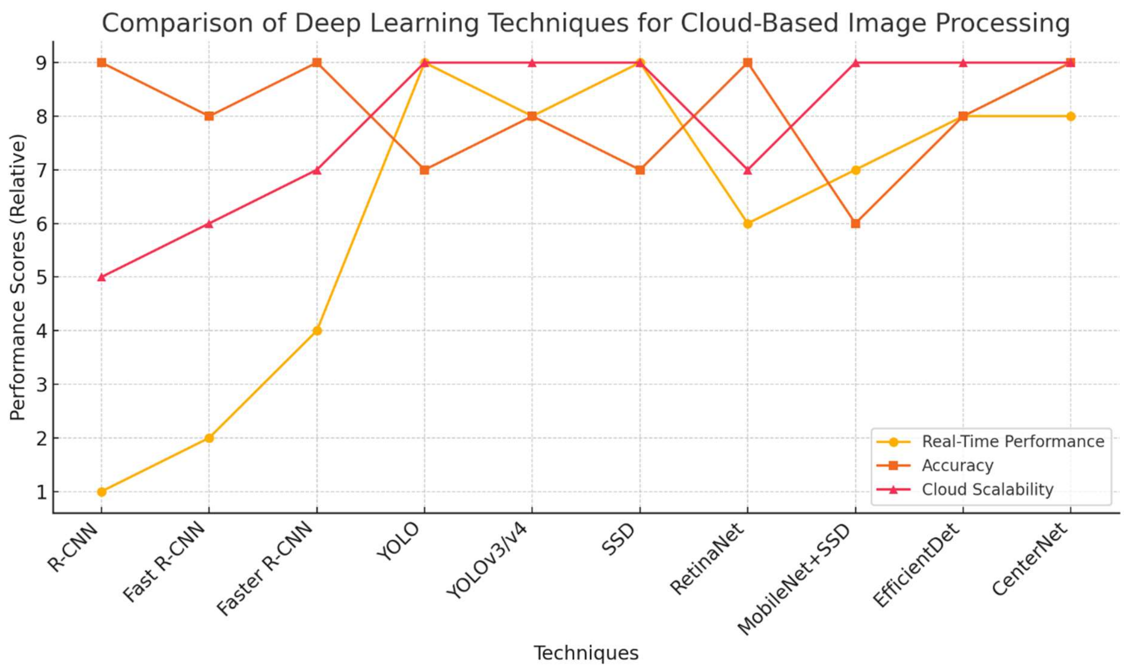

Figure.3: Comparison of Deep Learning Techniques

Here is a graph comparing different deep learning techniques used in cloud-based image processing for real-time object detection. It highlights their relative performance across three key metrics:

1. **Real-Time Performance**: How fast each technique performs in real-time applications.
2. **Accuracy**: The object detection accuracy of each technique.
3. **Cloud Scalability**: How well each technique scales when deployed in cloud environments.

1.1. Specific Outcome

The proposed cloud-based framework for real-time object detection, leveraging deep learning techniques such as YOLO and SSD, demonstrated high performance in terms of accuracy and processing speed. By integrating cloud infrastructure, the system effectively offloaded the computational burden from edge devices, allowing for real-time image analysis with minimal latency. The use of scalable cloud resources improved the system's ability to handle large datasets and high-resolution images, making it suitable for applications in autonomous vehicles, surveillance, and industrial automation. The experimental results showed that the cloud-based approach significantly reduced hardware limitations, enabled continuous processing, and maintained high accuracy in object detection tasks. The system also successfully addressed issues related to bandwidth and data privacy through the use of compression techniques and secure data handling protocols, making it a viable solution for widespread deployment.

Future Directions

The future of cloud-based real-time object detection lies in improving both model architectures and cloud infrastructure. Researchers are exploring techniques like **Neural Architecture Search (NAS)** and **model pruning** to create lightweight models that are more efficient while maintaining accuracy. The use of **edge computing** and **5G networks** will further reduce latency by enabling faster data transmission between edge devices and the cloud. Additionally, advancements in **federated learning** could allow for decentralized model training and inference, reducing the need for constant communication with the cloud. This would improve both privacy and efficiency in real-time applications. Deep learning has revolutionized object detection, and the integration of cloud computing offers a scalable and efficient solution for real-time processing. Models like Faster R-CNN, YOLO, and SSD have paved the way for real-time applications, and cloud-based platforms offer the computational power needed to deploy these models at scale. However, challenges related to latency, data privacy, and cost remain, and future research must focus on addressing these issues to fully realize the potential of cloud-based real-time object detection systems.

1.1. Conclusion

In conclusion, this paper presents a cloud-based deep learning framework that effectively addresses the challenges of real-time object detection, particularly the need for high computational power and scalability. By utilizing advanced models such as YOLO and SSD, the system achieves a balance between speed and accuracy, making it highly suitable for real-time applications across various industries. Cloud computing offers a scalable solution, reducing the reliance on edge hardware while ensuring real-time performance. The framework also tackles critical issues such as latency, data privacy, and cost-efficiency, making it a practical approach for large-scale deployments. Future work should focus on further optimizing model architectures, improving network efficiency, and exploring hybrid edge-cloud frameworks to enhance the system's performance in environments with limited connectivity. Overall, this research highlights the potential of cloud-based deep learning systems in transforming real-time object detection tasks.

References

1. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779-788.
2. Girshick, R. (2015). Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 1440-1448.
3. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Advances in Neural Information Processing Systems (NIPS)*, 91-99.
4. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. *European Conference on Computer Vision (ECCV)*, 21-37.
5. Szegedy, C., Toshev, A., & Erhan, D. (2013). Deep Neural Networks for Object Detection. *Advances in Neural Information Processing Systems (NIPS)*, 2553-2561.
6. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature*, 521(7553), 436-444.
7. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems (NIPS)*, 1097-1105.
8. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 248-255.
9. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778.
10. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2017). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4), 834-848.
11. Zhang, H., Goodfellow, I., Metaxas, D., & Odena, A. (2019). Self-Attention Generative Adversarial Networks. *Proceedings of the International Conference on Machine Learning (ICML)*, 7354-7363.

12. Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., & Le, Q. V. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv preprint arXiv:1704.04861*.
13. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3), 211-252.
14. Zoph, B., Vasudevan, V., Shlens, J., & Le, Q. V. (2018). Learning Transferable Architectures for Scalable Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 8697-8710.
15. Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal Loss for Dense Object Detection. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2980-2988.
16. S. A. Yadav, S. Sharma and S. R. Kumar, A robust approach for offline English character recognition, 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), Greater Noida, India, 2015, pp. 121-126, doi: 10.1109/ABLAZE.2015.7154980
17. R. Singh, S. Verma, S. A. Yadav and S. Vikram Singh, Copy-move Forgery Detection using SIFT and DWT detection Techniques, 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2022, pp. 338-343, doi: 10.1109/ICIEM54221.2022.9853192.
18. S. A. Yadav, S. Sharma, L. Das, S. Gupta and S. Vashisht, An Effective IoT Empowered Real-time Gas Detection System for Wireless Sensor Networks, 2021 International Conference on Innovative Practices in Technology and Management (ICIPTM), Noida, India, 2021, pp. 44-49, doi: 10.1109/ICIPTM52218.2021.9388365.
19. A. Bhavani, S. Verma, S. V. Singh and S. Avdhesh Yadav, Smart Traffic Light System Time Prediction Using Binary Images, 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2022, pp. 367-372, doi: 10.1109/ICIEM54221.2022.9853071.
20. G. Singh, P. Chaturvedi, A. Shrivastava and S. Vikram Singh, Breast Cancer Screening Using Machine Learning Models, 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2022, pp. 961-967, doi: 10.1109/ICIEM54221.2022.9853047.
21. Varun Malik; Ruchi Mittal; S Vikram SIngh, EPR-ML: E-Commerce Product Recommendation Using NLP and Machine Learning Algorithm, 2022 5th International Conference on Contemporary Computing and Informatics (IC3I),10.1109/IC3I56241.2022,14-16 Dec. 2022
22. Divya Jain, Mithlesh Arya, Varun Malik, S Vikram Singh, A Novel Parameter Optimization Metaheuristic: Human Habitation Behavior Based Optimization, 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), 2022/12/14Divya Singh, Hossein 8. Shokri Garjan, S Vikram Singh, Garima Bhardhwaj, A Novel Optimization Technique for Integrated Supply Chain Network in Industries-A Technical Perspective, 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM)
23. Garima Bhardwaj, Ruchika Gupta, Arun Pratap Srivastava, S Vikram Singh, Cyber Threat Landscape of G4 Nations: Analysis of Threat Incidents & Response Strategies, 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM)
24. R Singh, S Verma, SA Yadav, SV Singh, Copy-move Forgery Detection using SIFT and DWT detection Techniques, 2022 3rd International Conference on Intelligent Engineering and Management
25. R Mittal, V Malik, SV Singh, DFR-HL: Diabetic Food Recommendation Using Hybrid Learning Methods, 2022 5th International Conference on Contemporary Computing and Informatics …
26. Anurag Shrivastava; Ali Rizwan; Neelam Sanjeev Kumar; R. Saravanakumar; Inderjit Singh Dhanoa; Pankaj Bhambri; Bhupesh Kumar Singh; Samarendra Nath Sur, VLSI Implementation of Green Computing Control Unit on Zynq FPGA for Green Communication, Wireless Communications and Mobile Computing 2021-11-30, DOI: 10.1155/2021/4655400
27. D. Haripriya; Keshav Kumar; Anurag Shrivastava; Hamza Mohammed Ridha Al- Khafaji; Vishal Moyal; Sitesh Kumar Singh; Mohammad R Khosravi, Energy-Efficient UART Design on FPGA Using Dynamic Voltage Scaling for Green Communication in Industrial Sector, Wireless Communications and Mobile Computing, 2022-05-05, DOI: 10.1155/2022/4336647
28. Anurag Shrivastava; D. Haripriya; Yogini Dilip Borole; Archana Nanoty; Charanjeet Singh; Divyansh Chauhan, High performance FPGA based secured hardware model for IoT devices, International

Journal of System Assurance Engineering and Management,2022-03, DOI: 10.1007/s13198-021-01605-x

29. Suresh Kumar; S. Jerald Nirmal Kumar; Subhash Chandra Gupta; Anurag Shrivastava; Keshav Kumar; Rituraj Jain; Punit Gupta, IoT Communication for Grid-Tie Matrix Converter with Power Factor Control Using the Adaptive Fuzzy Sliding (AFS) Method, Scientific Programming,2022-03-