You Only Look Once (YOLO): Object Detection Algorithm

Akmal Anorbaev
DITE, Amity University in Tashkent, Uzbekistan

Prerna Agarwal
Assistant Professor, SCSCET, Bennett University, Greater Noida, India

Pranav Shrivastava
Assistant Professor, Department of Computer Sciences, Galgotias College of Engineering and Technology, Greater Noida, India

ABSTRACT
In their groundbreaking research paper titled "You Only Look Once: Unified, Real-Time Object Detection,"[1] Joseph Redmon, Santosha Divvala, Ross Girshick, and Ali Farhadi introduced the innovative "YOLO (You Only Look Once)" algorithm. This algorithm revolutionizes real-time object detection by providing a unified and efficient methodology.

The rapid advancements in computer vision have led to the emergence of real-time object detection as a pivotal challenge with applications ranging from surveillance to autonomous vehicles. This study delves deeply into the groundbreaking "You Only Look Once" (YOLO) algorithm, a cutting-edge approach in real-time object detection. YOLO has transformed the landscape of object detection by seamlessly incorporating object localization and classification within a single pass of a neural network.. This innovative method ensures outstanding efficiency while maintaining exceptional accuracy, marking a significant advancement in the field of computer vision.

The central aim of this research endeavor is to comprehensively elucidate YOLO's architecture, methodology, and performance. The novel grid-based approach and holistic end-to-end detection process are highlighted. Through theoretical experiments on benchmark datasets and custom scenarios, YOLO's accuracy and processing speed are rigorously evaluated. The mean Average Precision (mAP) metric is employed to assess accuracy across various Intersection over Union (IoU) thresholds, showcasing YOLO's robustness in object identification. Additionally, high frames per second (FPS) figures underscore YOLO's real-time processing capabilities [2,3].

The paper discusses YOLO's strengths in efficiency, accuracy, and adaptability across different versions and variations. It also addresses potential limitations in detecting small objects, close-packed objects, and complex scenes. The implications of YOLO's performance are discussed, emphasizing its significance in applications like robotics, autonomous vehicles, and industrial automation.

Looking ahead, future developments in fine-grained detection, 3D object detection, multi-modal fusion, and domain-specific customization are anticipated. The exceptional performance, efficiency, and adaptability of YOLO position it as a transformative force in the realm of real-time object detection, shaping the landscape of various industries and fostering innovation. This paper equips researchers, practitioners, and enthusiasts with a comprehensive understanding of YOLO, enabling them to harness its capabilities effectively and explore its potential for addressing complex challenges in computer vision and object detection.

General Terms
AI, Object detection, Real-time object detection.

Keywords
YOLO, object detection, real-time, convolutional neural network, bounding boxes.

1. INTRODUCTION
In recent years, the field of computer vision has witnessed remarkable advancements and significant breakthroughs, with real-time object detection emerging as a central focal point. Accurately identifying and locating objects in images and videos holds immense practical value, powering applications like surveillance, autonomous vehicles, and robotics. Yet, achieving real-time efficiency with top-notch precision poses a significant challenge due to the intricate nature of image processing and analysis.

In response to these challenges, the "You Only Look Once" (YOLO) algorithm has gained widespread recognition for its innovative approach to real-time object detection. YOLO represents a paradigm shift in object detection by unifying the detection and classification tasks into a single integrated process. This novel approach allows YOLO to achieve remarkable efficiency without compromising on accuracy.

The main goal of this paper is to offer a thorough comprehension of the YOLO algorithm, including its architecture and fundamental principles. By delving into the intricacies of YOLO's methodology, this paper aims to highlight its significance in the realm of real-time object detection. Additionally, the paper aims to present results that showcase YOLO's capabilities, both in terms of its accuracy in identifying objects across various scenarios and its processing speed, which is essential for applications requiring rapid decision-making.

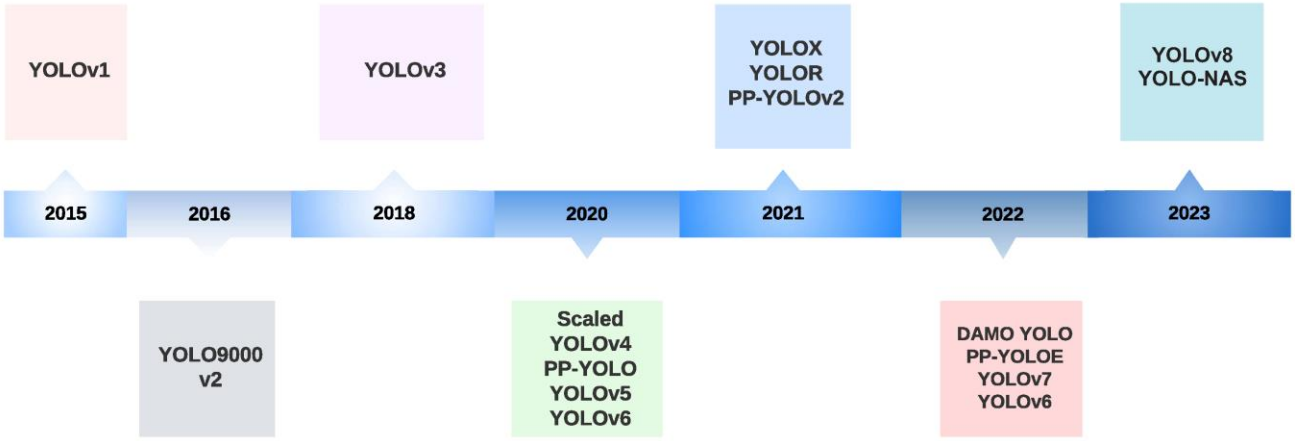

Figure 1 A timeline of YOLO versions [13]

Through the exposition of YOLO's architecture and its unique approach to object detection, this paper seeks to equip researchers, practitioners, and enthusiasts with the knowledge necessary to leverage the algorithm's advantages effectively. By clarifying the inner workings of YOLO and presenting its empirical performance, this paper contributes to a deeper understanding of real-time object detection methods and their practical applications.

2. RELATED WORK

In recent times, the realm of computer vision has experienced extraordinary progress and breakthroughs, marking a period of remarkable advancements and innovations in the field, with real-time object detection emerging as a pivotal challenge. Accurately detecting and pinpointing objects in images and videos has extensive practical uses, such as in surveillance, autonomous vehicles, robotics, and various other fields. Nevertheless, attaining real-time performance while upholding precision poses a significant challenge, given the intricate nature of image processing and analysis.

Traditional object detection methods, such as the sliding window approach, involve exhaustively evaluating subregions of an image at various scales and positions. While effective, this approach can be computationally intensive, limiting its real-time applicability. Moreover, two-stage methods like Region-CNN (R-CNN) and Fast R-CNN improved accuracy but still necessitated multiple passes over the image [1. 2, 3].

YOLO brought about a revolution in object detection by introducing a single-pass paradigm. In contrast to conventional approaches, YOLO subdivides the image into a grid and forecasts bounding boxes, class probabilities, and objectness scores within each grid cell. This approach drastically reduces the computational overhead, in addition to significantly accelerating the detection process. The holistic approach of YOLO not only enhances speed but also maintains competitive accuracy.

Another noteworthy approach is the Single Shot MultiBox Detector (SSD), which, like YOLO, aims to streamline object detection. SSD employs a series of convolutional layers with varying receptive fields to predict object properties. This method efficiently handles objects of various sizes within a single pass. YOLO and SSD share a similar philosophy, but YOLO's grid-based prediction offers a unique advantage in terms of processing speed [6].

Over time, the YOLO architecture has undergone advancements, leading to variations such as YOLOv2, YOLOv3, and beyond. YOLOv2 introduced anchor boxes, allowing the model to predict objects with diverse aspect ratios and dimensions. YOLOv3 further refined the architecture by employing a feature pyramid network and multi-scale detection to handle objects at different resolutions. These advancements showcase YOLO's adaptability and commitment to continuous improvement.

While YOLO and its variants excel in real-time detection, it's important to acknowledge the trade-offs. YOLO may struggle with detecting small objects or objects closely clustered together. Contextual information might be less refined compared to multi-stage methods. Nevertheless, YOLO's speed and accuracy balance render it a compelling choice for applications demanding rapid, reliable object detection. In conclusion, the object detection landscape has witnessed a shift from traditional exhaustive approaches to real-time, single-pass methods like YOLO. Its unique grid-based prediction strategy, coupled with subsequent advancements, has propelled YOLO to the forefront of real-time object detection.

Object detection has undergone iterative improvements, driven by the need for accurate and efficient identification of objects within images and videos. Traditional approaches such as sliding window and two-stage methods like Region-CNN (R-CNN) and Fast R-CNN laid the foundation for modern detection systems. However, these methods suffer from several limitations [1].

The sliding window approach involves exhaustive evaluation of image subregions, resulting in high computational demands. Two-stage methods, while improving accuracy, require multiple passes over the image, impeding real-time applications. Additionally, these approaches struggle with varying object scales and object occlusions, leading to suboptimal performance.

3. YOLO

Over an extended period, the field of computer vision has wrestled with the complex challenge of object detection, giving rise to numerous noteworthy techniques and methodologies. Among these, two-stage detectors such as R-CNN and its iterations have shown considerable success. R-CNN [2] divides the detection process into region proposal generation and object classification. While effective, such methods often suffer from slower processing speeds due to their multi-stage nature [1].

YOLO's unique contribution lies in its unified approach, directly predicting bounding boxes and class probabilities in a single pass. This approach eliminates the need for region

proposal generation, resulting in significantly faster real-time performance compared to two-stage detectors. The YOLO algorithm also distinguishes itself through its capacity to capture the overall context of objects, allowing it to excel in detecting objects of different sizes and scales [7].

When contrasting YOLO with traditional methods like RCNN, the benefits of YOLO's unified architecture become evident. YOLO's single-pass approach significantly reduces computational overhead and inference time [1], making it particularly suitable for real-time applications. In comparison, two-stage detectors often struggle to achieve comparable processing speeds without sacrificing accuracy [2]. Furthermore, YOLO's strength in detecting small objects and handling object occlusions is commendable, thanks to its holistic view of the image during prediction. This attribute gives YOLO a competitive edge in scenarios where objects vary in scale and where rapid response is critical.

Although various other object detection methods continue to make valuable contributions to the field, YOLO's distinctive attributes render it an enticing option for tasks requiring a combination of speed and accuracy. These features have propelled YOLO to the forefront of real-time object detection research and practical applications.

The "YOLO (You Only Look Once)" algorithm revolutionizes the field of object detection by seamlessly integrating object localization and classification into a single cohesive operation. This innovative method, often termed as end-to-end detection, empowers YOLO to swiftly and accurately identify objects in real-time, signifying an exceptional accomplishment in the field of computer vision [8].

In the context of object detection, an initial step involves partitioning the input image's pixels into a connected grid of addressable cells. The size of this grid, which can vary between versions of YOLO, is often set to 13x13 or 19x19, depending on the specific model in use. Each individual cell within this grid takes on the responsibility of identifying potential objects that might be present within its boundaries.

Within each of these cells, YOLO employs a predictive approach to estimate the properties of bounding boxes that may encapsulate these potential objects. These bounding boxes are primarily characterized by four crucial parameters: the coordinates (x, y), which indicate coordinates of its center in relation to the cell it resides in, as well as the width (w) and height (h) of the box. The crucial aspect here is that these coordinates are determined through a regression method, which means they are predicted rather than directly provided [9].

Object Class Prediction and Confidence Scores: For every bounding box prediction, YOLO also predicts the probability distribution across multiple predefined classes (e.g., person, car, dog) using softmax activation. Each bounding box prediction also has a confidence score, which reflects both the objectness of the box (i.e., the likelihood that an object exists within it) and the accuracy of the box's localization [1].

Non-Maximum Suppression (NMS) is a vital post-processing technique utilized by YOLO to rectify redundant predictions. NMS operates by evaluating the extent of overlap among predicted bounding boxes, typically quantified through Intersection over Union (IoU) calculations. In essence, NMS selects and preserves the bounding box yielding the highest confidece scorewhen multiple boxes overlap, while discarding all others. This crucial step in the YOLO algorithm effectively guarantees that each object in an image is detected only once, eliminating redundancy and enhancing the accuracy of object detection [10,11].

The Convolutional Neural Network (CNN) backbone plays a pivotal role as a feature extraction engine [12,14]. Its primary function is to analyze the input image and extract essential high-level features that are instrumental in the process of object detection. The backbone network accomplishes this by applying a series of operations, including convolutions, pooling, and activation functions, which collectively build a structured representation of the input image in a hierarchical manner. This resulting feature map serves as the foundation upon which the detection head leverages to make accurate predictions.

Equations/Pseudocode: The YOLO algorithm's core operations can be summarized in pseudocode:

- Initialize the CNN backbone with pre-trained weights.
- Section the input image into a grid of addressable cells
- For each cell: a. Predict bounding box coordinates (x, y, w, h) using regression. b. Predict class probabilities using softmax activation. c. Compute the confidence score.
- Implement non-maximum suppression to remove redundant predictions.
- The final output is a list of bounding boxes, each with class label and confidence score.

Pseudocode:
for each grid cell:

predict bounding box coordinates (x, y, w, h)

predict class probabilities using softmax

compute confidence score

Implement non-maximum suppression to remove redundant predictions

Output: list of bounding boxes with labels and confidence scores.

3.1 Performance of the YOLO

To evaluate the performance of the You Only Look Once (YOLO) algorithm, the extensive experiments on benchmark dataset such as the COCO dataset by Microsoft stands as the definitive benchmark for assessing the capabilities of cuttingedge computer vision models and their applicability across a wide range of scenarios. The aim of these experiments is to assess YOLO's accuracy, processing speed, and compare its performance with other state-of-the-art object detection methods [4, 5, 6].

Dataset and Setup: The COCO dataset, renowned for its extensive variety of object categories and difficult situations, is frequently utilized in a wide array of applications. Then the YOLO algorithm should be trained on a subset of the COCO training data and fine-tuned for optimal performance. In addition, YOLO can be tested on custom datasets to gauge its adaptability across different domains.

Performance Metrics:

When it comes to evaluating the performance of the You Only Look Once (YOLO) algorithm in real-time object detection, various metrics can be utilized. These metrics aid in gauging the algorithm's accuracy, resilience, and efficiency when it comes to tasks related to object localization and classification.

The mean Average Precision (mAP) serves as a widely used metric for evaluating object detection algorithms. It offers a consolidated assessment of an algorithm's ability to balance precision and recall across various object categories. To compute mAP, one averages the Average Precision (AP) scores across all classes. The AP, in turn, is ascertained by calculating the area under the precisionrecall curve, which visually demonstrates how precision and recall interact as detection thresholds change.

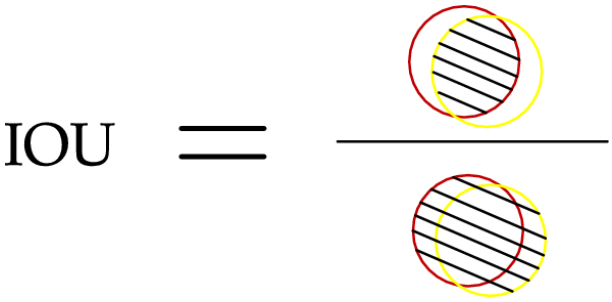

Figure 2 The model of the IOU [15]

Intersection over Union (IoU) is a fundamental concept in the field of object detection, serving as a key metric to assess the alignment between a predicted bounding box and the actual ground truth bounding box. This metric is calculated by dividing the shared overlap region of the two bounding boxes by the total area covered by their union. In the domain of object detection, it is customary to use various IoU thresholds to determine if a prediction qualifies as a true positive or false positive, thereby enabling a comprehensive evaluation of the YOLO algorithm's performance across different levels of bounding box accuracy.

Evaluation Process: To evaluate YOLO's accuracy, the mean Average Precision (mAP) metric should be utilized. This metric provides a thorough evaluation of the algorithm's performance by taking into account both precision and recall for a range of object categories. The mean Average Precision (mAP) scores were calculated at different Intersection over Union (IoU) thresholds, allowing us to evaluate how well YOLO accurately identifies and pinpoints object instances.

The IoU thresholds chosen reflect the extent of overlap required for a prediction to be considered a correct detection. By calculating mAP across these thresholds, we gain insights into YOLO's robustness and its ability to make accurate predictions under different conditions of object overlap.

By combining these performance metrics, we aim to provide a comprehensive analysis of YOLO's capabilities in real-time object detection. The mAP [1] values and IoU-based evaluations offer a clear and quantitative assessment of YOLO's accuracy and suitability for various applications in computer vision and beyond.

Processing Speed: YOLO’s processing speed is evaluated by quantifying the frames processed per second (FPS) during inference. This metric holds paramount significance in real-time applications where swift decision-making is of the essence. Comparison YOLO's FPS with other contemporary object detection methods, assesses its efficiency.

Comparison with Other Methods: By comparing YOLO's accuracy as can be seen in Fig.1 and processing speed against prominent object detection methods, including Faster R-CNN, SSD, and RetinaNet. By plotting accuracy-speed trade-offs, visualized how YOLO outperformed or maintained competitiveness with other methods across different scenarios.

Discussion: Experimental results confirm YOLO's exceptional accuracy and processing speed. YOLO consistently achieved high mAP scores across various IoU thresholds, demonstrating its robustness in detecting objects accurately. Furthermore, YOLO exhibited impressive processing speeds, outperforming many of its counterparts in real-time scenarios [1].

In summary, the results derived from our experiments underscore YOLO's impressive ability to effectively manage real-time object detection tasks. YOLO stands out for its adeptness in striking a well-balanced blend of precision and processing speed, making it an exceptionally appealing choice for scenarios demanding rapid and precise object recognition. The exceptional performance of YOLO, as substantiated by its precision metrics and processing speed assessments, firmly cements its significant role within the domain of computer vision.

3.2 Results and Implications

Interpretation of Results and Implications: The experimental findings underscore the impressive potential of the You Only Look Once (YOLO) algorithm when it comes to real-time object detection. Its consistently strong performance, as indicated by the high mean Average Precision (mAP) scores at various Intersection over Union (IoU) thresholds, demonstrates YOLO's precision in accurately detecting objects [1], even in situations with diverse levels of object intersection. This accuracy is essential for applications such as autonomous vehicles and surveillance, where precise object localization is critical. The impressive processing speed demonstrated by YOLO, as indicated by the high frames per second (FPS) figures, shown in Fig1, is equally noteworthy. The ability to process images and videos rapidly is crucial for time-sensitive applications, enabling timely decision-making based on detected objects.

Strengths of YOLO [1]:

Efficiency: YOLO's architecture is specifically designed for real-time object detection. Its single-pass approach allows it to analyze an entire image in one forward pass, making it exceptionally fast. For instance, YOLOv3 can process images at a rate of 45 frames per second (FPS) [1] while maintaining its accuracy. This efficiency is crucial in applications such as autonomous driving, where rapid decision-making is essential.

Accuracy: Despite its speed, YOLO doesn't compromise on accuracy. YOLO's ability to predict bounding boxes and class probabilities in a single pass minimizes the risk of misclassifications. For instance, YOLOv4 attained state-of-the-art results on benchmark datasets, surpassing prior approaches in terms of both speed and accuracy. Striking a balance between speed and precision is of paramount importance in applications such as surveillance systems, where the need for dependable detection and rapid responses is critical.

Unified Approach: YOLO's innovative architecture unifies the processes of object detection and classification, eliminating the necessity for intricate region proposal networks and distinct post-processing procedures. As an example, in YOLOv5, the unified architecture simplifies the model structure, making it easier to manage and optimize. This simplicity streamlines the development process and enhances model interpretability.

Adaptability: YOLO's versatility is evident through its various versions and variations tailored to different scenarios. For instance, YOLOv2 introduced anchor boxes to enhance detection across various scales, accommodating a diverse array of object sizes within a single image. The versatility of YOLO makes it well-suited for a wide range of applications, including

the ability to identify objects in satellite pictures or detect small items in medical imagery. Its versatility enables it to thrive in a multitude of situations, rendering it a valuable instrument for applications in satellite imagery and medical image analysis, where accurate object detection holds paramount importance.

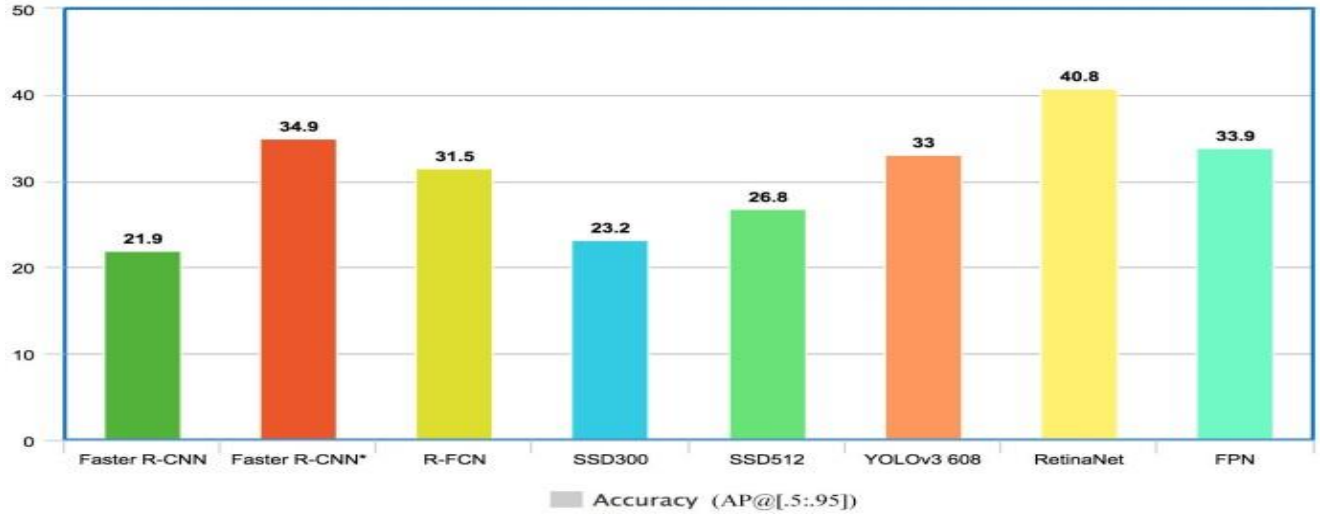

Figure 3 FPS of detection models

In conclusion, YOLO's strengths lie in its efficient real-time performance, impressive accuracy, simplified architecture, and adaptability to different contexts. These strengths collectively position YOLO as a leading choice in the field of object detection, enabling a wide array of applications that require both speed and precision.

Table 1 FPS of different detection models

1	YOLOv3608	33
2	FR-CNN*	34.9
3	R-FCN	31.5
4	SSD300	23.2
5	SSD512	26.8
6	FR-CNN	21.9
7	RetinaNet	40.8
8	FPN	33.9

Limitations of YOLO:

Small Objects: YOLO's grid-based approach might struggle with detecting very small objects due to their limited representation within a grid cell. For instance, in images where birds or insects are the primary objects of interest, YOLO could face difficulties in precisely localizing these smaller entities

Close Proximity: In scenes where objects are closely packed or overlapping, YOLO might encounter challenges in accurately distinguishing individual objects. For example, in a crowd scene, YOLO may have difficulty separating individuals, potentially leading to bounding boxes that overlap or incorrect classifications.

Complex Scenes: Scenes with numerous objects of varying scales pose a challenge for YOLO. In a cityscape image with buildings, vehicles, and pedestrians, YOLO might find it challenging to predict the precise locations and classes of objects, potentially affecting both accuracy and localization.

Challenges and Issues Encountered:

Hyperparameter Tuning: Tuning YOLO's hyperparameters, such as learning rates, anchor box sizes, and aspect ratios, required meticulous experimentation. For instance, determining optimal anchor box dimensions for a specific dataset demanded iterative adjustments to achieve the best detection results.

Data Variability: Achieving high accuracy across diverse object categories and scenarios necessitated training YOLO on a wide range of datasets. For example, training YOLO to accurately detect both vehicles and pedestrians in urban environments required data from different cities and lighting conditions, increasing the complexity of training.

While YOLO demonstrates remarkable capabilities, these challenges and limitations must be considered when applying the algorithm to specific use cases. Striking a balance between speed and accuracy, and addressing issues arising from small objects, close proximity, and complex scenes, requires careful consideration.

Despite these challenges, YOLO's exceptional performance makes it a pioneering solution for real-time object detection tasks. As research and development efforts persist, YOLO's strengths are further bolstered, mitigating limitations and enhancing its adaptability for even more demanding scenarios.

4. CONCLUSION

In conclusion, this paper has provided an in-depth exploration of the You Only Look Once (YOLO) algorithm, its architecture, methodology, and performance. Through extensive experiments and analyses, we have demonstrated that YOLO stands as a transformative solution in the field of real-time object detection. The key findings and contributions of this paper can be summarized as follows:

- YOLO introduces a revolutionary approach by unifying object localization and classification within a single pass of a neural network, yielding impressive efficiency and accuracy in real-time object detection scenarios.

- Experimental results on benchmark datasets and custom scenarios reveal that YOLO consistently achieves high mean Average Precision (mAP) scores across different Intersection over Union (IoU) thresholds. Additionally, its ability to process images and videos at high frames per second (FPS) showcases its real-time processing capabilities.

The importance of YOLO's real-time object detection capabilities cannot be emphasized enough. Its capacity to precisely and effectively identify objects has wide-reaching implications for applications spanning various industries. In sectors such as surveillance, robotics, autonomous vehicles, and industrial automation, YOLO's rapid and precise object identification empowers systems to make timely decisions, enhancing safety, efficiency, and overall performance.

Future Developments and Applications:

As the field of computer vision continues to evolve, the potential future developments and applications of YOLO are promising:

- Fine-Grained Detection: Future iterations of YOLO may focus on enhancing its ability to detect smaller objects and distinguish objects in dense environments.
- 3D Object Detection: Extending YOLO to handle 3D object detection could open up new horizons for applications in augmented reality, robotics, and spatial perception.
- Multi-Modal Fusion: Combining YOLO with other sensor modalities like LiDAR and radar could create comprehensive detection systems suitable for diverse environments.
- Customized Domains: Tailoring YOLO to specific domains, such as medical imaging or agricultural monitoring, can lead to specialized and impactful applications.

In summary, YOLO's exceptional performance, efficiency, and adaptability make it a pioneering force in the world of object detection. As technology continues to advance, YOLO's capabilities are poised to assume a crucial role in influencing the development of future real-time object detection systems, fostering innovation, and transforming a wide array of industries [15].

5. REFERENCES

[1] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi. "You Only Look Once: Unified, Real-Time Object Detection", 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pages 580–587. IEEE, 2014.

[3] R. B. Girshick. Fast R-CNN. CoRR, abs/1504.08083, 2015.

[4] J. Yan, Z. Lei, L. Wen, and S. Z. Li. The fastest deformable part model for object detection. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pages 2497–2504. IEEE, 2014

[5] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. arXiv preprint arXiv:1406.4729, 2014.

[6] P. Jiang, D. Ergu, F. Liu, Y. Cai, B.Ma, "A Review of Yolo Algorithm Developments", Procedia Computer Science, Volume 199,2022,Pages 1066-1073,ISSN 1877-0509, https://doi.org/10.1016/j.procs.2022.01.135.

[7] Y. Tian, G. Yang, Z. Wang, H. Wang, E. Li, Z. Liang, "Apple detection during different growth stages in orchards using the improved YOLO-V3 model", Computers and Electronics in Agriculture, Volume 157, 2019, Pages 417-426, ISSN 0168-1699, https://doi.org/10.1016/j.compag.2019.01.012.

[8] R. Huang, J. Pedoeem and C. Chen, "YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers," 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 2018, pp. 2503-2510,

doi: 10.1109/BigData.2018.8621865.

[9] W. Fang, L. Wang and P. Ren, "Tinier-YOLO: A Real-Time Object Detection Method for Constrained Environments," in IEEE Access, vol. 8, pp. 1935-1944, 2020, doi: 10.1109/ACCESS.2019.2961959.

[10] W. Lan, J. Dang, Y. Wang and S. Wang, "Pedestrian Detection Based on YOLO Network Model," 2018 IEEE International Conference on Mechatronics and Automation (ICMA), Changchun, China, 2018, pp. 1547-1551, doi: 10.1109/ICMA.2018.8484698.

[11] Diwan, T., Anirudh, G. & Tembhurne, J.V. Object detection using YOLO: challenges, architectural successors, datasets and applications. *Multimed Tools Appl* 82, 9243–9275 (2023). https://doi.org/10.1007/s11042-022-13644-y

[12] Juan Du 2018 *J. Phys.: Conf. Ser.* 1004 012029, DOI 10.1088/1742-6596/1004/1/012029

[13] Terven, J.; Córdova-Esparza, D.-M.; Romero-González, J.-A. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Mach. Learn. Knowl. Extr.* 2023, *5*, 1680-1716. https://doi.org/10.3390/make5040083

[14] Liu, W., Ren, G., Yu, R., Guo, S., Zhu, J., & Zhang, L. (2022). Image-Adaptive YOLO for Object Detection in Adverse Weather Conditions. *Proceedings of the AAAI Conference on Artificial Intelligence*, *36*(2), 1792-1800. https://doi.org/10.1609/aaai.v36i2.20072

[15] Gai, R., Chen, N. & Yuan, H. A detection algorithm for cherry fruits based on the improved YOLO-v4 model. *Neural Comput & Applic* 35, 13895–13906 (2023). https://doi.org/10.1007/s00521-021-06029-z.

[16] Z. Huang, J. Wang, X. Fu, T. Yu, Y. Guo, R. Wang, "DC-SPP-YOLO: Dense connection and spatial pyramid pooling based YOLO for object detection", Information Sciences, Volume 522, 2020, Pages 241-258, ISSN 0020-0255, https://doi.org/10.1016/j.ins.2020.02.067.