Dean&Francis
ISSN 2959-6157

Evolutionary Trajectory and Groundbreaking Innovations in the YOLO Object Detection Algorithm

Weihan Wu

City University of Macau, Macau Special Administrative Region, China
*Corresponding author:D22090100684@Cityu.edu.mo

Abstract:

This paper systematically traces the evolutionary trajectory of the YOLO series (versions 1 through 12) within the field of computer vision object detection. Pioneered by YOLOv1 in 2015, this framework introduced the groundbreaking single-stage detection and regression paradigm, enabling end-to-end detection through its S×S grid architecture. Its Fast YOLO variant demonstrated notable real-time performance advantages on the PASCAL VOC dataset. Subsequent iterations marked significant advancements: v2 incorporated batch normalization and anchor priors, enhancing efficiency with the Darknet-19 backbone while YOLO9000 expanded multi-category recognition capabilities; v3 optimized accuracy through Darknet-53 and multi-scale feature fusion; v4 formalized the modular "Backbone-Neck-Head" design. Enhancements continued from v8 to v12—v8's C2f module bolstered feature fusion, v9 addressed gradient misalignment via its PGI framework, v10 achieved NMS-free end-to-end detection, v11 improved efficiency with the C3k2 module, and v12 enhanced real-time capabilities via the R-ELAN structure. Through iterative development, the series exhibits substantial improvements in detection speed, accuracy, and adaptability to complex scenarios, securing its position as a mainstream solution. Future applications hold considerable promise for leveraging this technology in demanding contexts such as embodied intelligence, medical diagnostics, and tunnel inspection.

Keywords: Convolutional Neural Networks; single-stage object detection; YOLO; algorithms.

1. Introduction

Computer vision, as a pivotal domain of artificial intelligence, aims to construct semantic understanding systems for images and videos. Object detection algorithms identify object categories and spatially

ISSN 2959-6157

localize targets, quantitatively represented with bounding boxes, thus enabling extensive applications in intelligent surveillance, autonomous driving, and related fields.

Early object detection methods relied on handcrafted features. Limited by inadequate feature representation and computational inefficiency, these approaches struggled to satisfy demands for both detection accuracy and real-time performance in complex scenarios. Convolutional Neural Networks propelled revolutionary breakthroughs in object detection. The YOLO series algorithms, characterized by their single-stage detection framework and operational efficiency, have spurred extensive research and industrial adoption.

Since the introduction of YOLOv1 in 2015, the series has evolved through twelve iterations. Each generation refined network architectures and training methodologies, achieving substantial enhancements in detection speed, accuracy, and adaptability to intricate environments. These advancements solidified YOLO's mainstream dominance in object detection. Systematically organizing its evolutionary trajectory is essential for distilling algorithmic design principles and guiding future innovations—carrying significant theoretical and practical implications. This review comprehensively examines YOLOv1 to YOLOv12, with a focus on core innovations, performance optimization strategies, and scholarly contributions.

2. Evolutionary Trajectory of YOLO

YOLOv1 pioneered an end-to-end, single-stage detection framework by dividing images into S×S grids and directly regressing target class probabilities along with bounding box coordinates, eliminating the region proposal generation inherent to two-stage detectors. Its Fast YOLO variant achieved 52.7% mAP on PASCAL VOC with a 9-layer network, doubling the inference speed of contemporary approaches. The standard version elevated mAP to 63.4% [1], achieving the first effective balance between detection accuracy and real-time capability. Limitations included each grid predicting only two objects of the same class, insufficient anchor generalization, and constrained performance in dense scenes.

YOLOv2 (2017) enhanced convergence stability via batch normalization and addressed cross-resolution adaptation through iterative fine-tuning at 448×448 resolution, yielding mAP improvements of 2% and 4% respectively [2]. Its fully convolutional architecture accepted inputs at any resolution divisible by 32. K-means clustering generated five anchor priors, substantially increasing recall. Feature fusion through a passthrough layer improved small object detection. The Darknet-19 backbone attained 72.9% Top-1 accuracy on ImageNet [2], while its enhanced YOLO9000 version detected over 9,000 object categories, establishing foundational capabilities for multi-class recognition.

YOLOv3 (2018) employed a 53-layer convolutional backbone, Darknet-53, which improved computational efficiency via residual connections and strided convolutions, achieving 77.2% Top-1 accuracy at 256×256 resolution [3]. It incorporated feature pyramid structures [4] for anchor-based detection and feature fusion across three scales. Logistic regression replaced SoftMax for multi-label classification support. Processing 320×320 images required only 22ms, matching SSD in mAP while operating three times faster [3], constituting a major milestone in real-time detection.

YOLOv4 formalized the modular "Backbone-Neck-Head" architecture. The CSPDarknet-53 backbone reduced computation via cross-stage connections. Its neck integrated Spatial Pyramid Pooling—SPP and PANet for enhanced multi-scale feature fusion. The detection head maintained anchor-based mechanisms with optimized Non-Maximum Suppression (NMS) post-processing. Proposed strategies included a "Bag of Freebies" (Mosaic augmentation, CIoU loss) and structural improvements, achieving 43.5% AP with 50 FPS inference on the COCO dataset [5] and surpassing contemporary detectors.

YOLOv5 migrated implementation to PyTorch, introducing AutoAnchor for automated anchor calibration [6]. YOLOX (an evolution of v3) adopted an anchor-free mechanism, elevating AP by 5.9% [7]. YOLOv6 utilized scale-differentiated designs (RepVGG backbone, Rep-PAN neck) enabling adaptation across scenarios from 35.9% AP (1234 FPS) to 57.2% AP (29 FPS) [8]. YOLOv7 introduced an E-ELAN architecture and compound scaling, reducing parameters by 39% versus v4-tiny while attaining 56.8% AP at ≥30 FPS on GPUs [9], thus overcoming edge deployment constraints.

YOLOv8 launched multi-scale variants with optimized CSPLayers and C2f modules. Its anchor-free head combined with Distribution Focal Loss (DFL) improved small object detection, yielding 53.9% AP at 280 FPS for YOLOv8X [10]. YOLOv9 proposed a Programmable Gradient Information (PGI) framework to mitigate deep network information bottlenecks, reducing parameters by 16% and computations by 27% while increasing AP by 1.7% [11]. YOLOv10 achieved end-to-end detection without NMS dependency, delivering 1.8× faster inference than RT-DETR-R18 [12]. YOLOv11 utilized C3k2 modules and a C2PSA mechanism to maintain ~47% accuracy in the 2-6ms low-latency range [13]. The latest YOLOv12 employs an R-ELAN backbone optimized with 7×7 separable convolutions and FlashAttention, reaching 49% mAP50-95 at 1-5ms latency [14], setting new benchmarks for complex real-time detection.

The YOLO series demonstrates a sustained evolution—shifting from anchor dependency to anchor-free designs, single-scale to multi-scale fusion, and fundamental augmentation to sophisticated label assignment. Through successive iterations, the algorithm progressively enhanced both detection accuracy and operational speed, as substantiated in Table 1. This progression also establishes a comprehensive technological ecosystem spanning edge to cloud deployments.

3. Core Architecture Analysis

3.1 YOLOv1

YOLO (You Only Look Once) completes detection tasks through a single network evaluation pass, fundamentally differentiating itself from earlier approaches. Previous methods typically adapted classifiers using sliding windows, requiring hundreds to thousands of image evaluations, or employed two-stage detection involving initial region proposal generation followed by classification. YOLO instead adopts a regression-based framework to directly predict detection outcomes, as illustrated in Figure 1. This approach demonstrates superior generalization across domains compared to alternative detectors.

Within the YOLOv1 architecture, holistic contextual representations facilitate concurrent determination of all bounding box coordinates and their co-associated class probabilities. The system divides input images into an S×S grid, assigning grid cells responsibility for detecting objects centered within them. Interleaved 1×1 convolutional layers reduce feature dimensionality.

A Fast YOLO variant warrants discussion for pushing the boundaries of rapid target detection. This version employs a streamlined 9-layer convolutional neural network while retaining all original training and testing parameters.

Evaluated on the PASCAL dataset, Fast YOLO achieved unprecedented speed as the fastest object detector of its era. Experimental results demonstrate it doubled the speed of prior real-time detection models while maintaining 52.7% mean average precision[1]. The standard YOLO algorithm further advanced detection accuracy, achieving 63.4% mAP while preserving real-time efficiency[1].

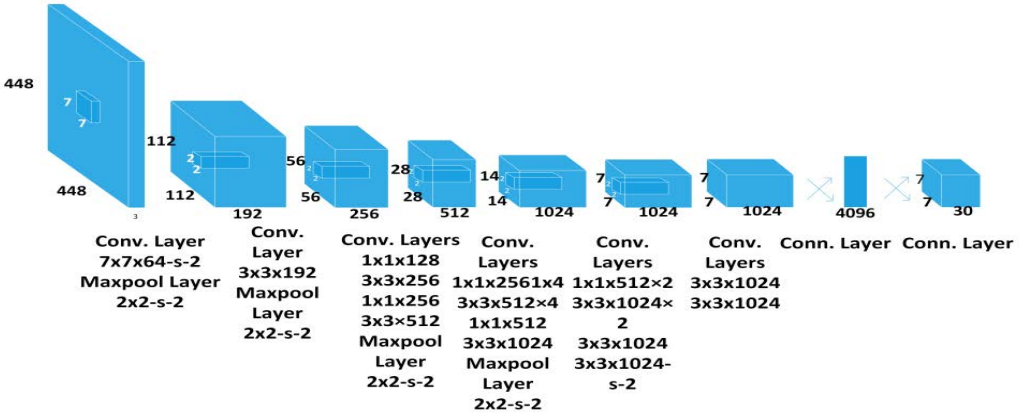

Fig.1 YOLOv1 Architecture Diagram Schematic

Notable limitations of YOLOv1 include its restriction of detecting only two objects of the same class per grid cell, constraining performance in complex scenes with proximal objects. The model exhibits reduced efficiency when detecting objects with aspect ratios absent from training data. Independent bounding box predictions per grid cell also impede accurate localization of adjacent or overlapping objects.

3.2 YOLOv4

The year 2020 marked a watershed for the industrial deployment of YOLO architecture. YOLOv4 emerged as a high-precision object detection model optimized for real-time execution on conventional GPUs. This iteration introduced enhancements across network architecture, training protocols, and data augmentation relative to its predecessor, systematically optimizing the speed-accuracy tradeoff through empirical evaluation of multiple refinements. During this phase, the detector architecture was formally redefined through a novel conceptual framework comprising Backbone, Neck, and Head modules[5], as illustrated in Figure 2.

The Backbone serves as the core feature extractor, utilizing CNN architecture pre-trained on Large-cardinality image classification tasks such as ImageNet. It hierarchically captures multi-scale features—shallow layers extract elementary textures and edges, while deeper layers discern high-level semantics and object components.

Functioning as the bridge between Backbone and Head, the Neck architecture aggregates and refines multi-level

ISSN 2959-6157

features via custom convolutional layers or feature pyramid networks. It specifically enhances spatial dependencies and semantic representations across scales.

The Head module executes prediction tasks using features propagated from Backbone and Neck. Its sub-networks specialize in distinct functionalities (classification, localization, instance segmentation) to generate detection outputs for candidate objects. Non-Maximum Suppression (NMS) filters redundant predictions during post-processing, retaining only the highest-confidence detections.

Building upon CSPDarknet-53's demonstrated efficacy in YOLOv3, YOLOv4 retained this architecture for its Backbone. The Neck incorporated a SPP module to expand receptive fields and employed PANet for feature aggregation, replacing YOLOv3's FPN—supplemented by a Spatial Attention Module (SAM) [5]. The detection Head maintained YOLOv3's anchor-based mechanism. The CSPNet configuration reduced computational load without compromising accuracy, while the SPP block enlarged receptive fields without impacting inference speed.

Notably, YOLOv4 pioneered the complementary "Bag-of-Freebies" and "Bag-of-Specials" strategies. The former boosted training efficacy via mosaic augmentation (four-image composition) and CIoU loss optimization; the latter leveraged the CSPDarknet53 Backbone and SPP+PANet Neck structure [5].

On the COCO 2017 dataset, YOLOv4 attained 43.5% AP and 65.7% AP50 while exceeding 50 FPS on NVIDIA V100 GPUs [5], surpassing other empirically leading detectors in both speed and precision metrics.

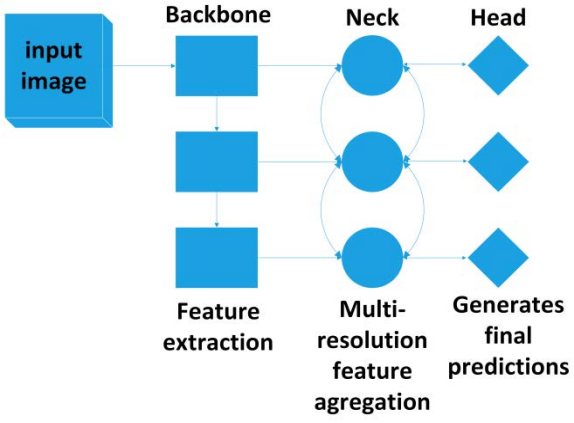

Fig. 2 "Backbone-Neck-Head" Architecture Diagram Schematic

3.3 YOLOv12

As the latest iteration of the YOLO algorithms launched in 2025, YOLOv12 incorporates key architectural enhancements such as FlashAttention, adaptive MLP ratios, and refined convolution strategies. These innovations overcome the challenge faced by its predecessor v11 in maintaining high throughput under stringent hardware constraints. Its schematic architecture is depicted in Figure 3.

The core architectural innovation lies in the redesigned backbone network, which implements the novel R-ELAN structure. This architectural reformulation integrates deeper convolutional hierarchies with engineered residual connections, effectively mitigating gradient dissipation bottlenecks while enhancing feature reuse efficacy, consequently elevating discriminative capacity for intricate object details across multi-scale and geometrically varied contexts[14].

Further computational efficiency is achieved through 7×7 separable convolutions, which preserve spatial context while utilizing fewer parameters compared to conventional large-kernel operations or positional encodings [14]. Additionally, partitioning feature maps into distinct regions and applying FlashAttention routines significantly reduces both memory transfers and computational overhead. This optimization enables real-time inference even at substantially elevated input resolutions.

Aligning with YOLOv11's approach, YOLOv12 offers multiple scaled variants tailored to different computational capabilities and performance requirements. Smaller variants (e.g., 12n and 12s) excel in latency-sensitive applications, while the larger variant (12x) maintains high precision or complex scenarios [14].

Comparative evaluations confirm that YOLOv12 consistently surpasses YOLOv10 and YOLOv11 in both mAP and detection speed. Notably, within the critical low-latency regime (1-5 ms), the YOLOv12s variant sustains an mAP50-95 accuracy of approximately 49% [14].

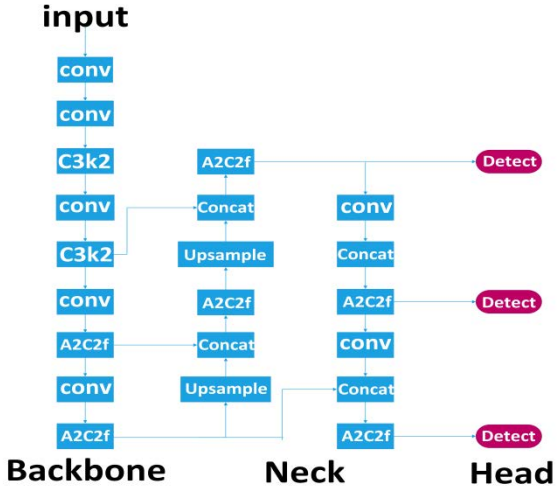

Fig. 3 YOLOv12 Architecture Diagram Schematic

4. Conclusion and future prospects

This review traces the technical evolution of the YOLO series in computer vision. Commencing with YOLOv1's foundational single-stage detection and regression output, through to the latest architectural enhancements in YOLOv12 illustrated in Table 1 successive versions have achieved concurrent gains in accuracy and speed. These improvements have consistently focused on refining network architectures, training strategies, feature fusion mechanisms, and loss functions. YOLOv2 introduced performance-boosting elements like batch normalization and anchor priors; YOLOv3 expanded convolutional layers and refined anchor assignment strategies; YOLOv4 adopted a novel architectural description and incorporated diverse optimization techniques. Subsequent versions, including v8, v9, v10, v11, and v12, respectively made breakthroughs in areas such as architectural design, computational efficiency, and real-time capability. Through multi-generational advancement, the YOLO series has achieved marked improvements in critical metrics—detection speed, precision, and adaptability to complex scenes—solidifying its mainstream status in object detection.

Looking forward, YOLO holds significant potential for broader deployment in complex scenarios and diverse applications. Key future directions include:

Embodied Intelligence: Projects like BEHAVIOR demonstrate frameworks integrating YOLO with robotic systems, pushing detection capabilities into the action dimension via visual perception-decision-execution loops.

Medical Detection: YOLO exhibits robust detection capabilities across multiple medical domains, including skin lesion classification and cardiac anomaly detection. Its utility extends to surgical instrument detection and tracking, enhancing procedural safety and efficiency [15]. Future efforts may focus on developing dedicated YOLO variants tailored for medical detection applications.

Crater Detection: The YOLO v8 platform has facilitated the creation, training, and deployment of models for lunar crater detection [16], with promising prospects for continued application in this domain.

Table.1 Performance Comparison Schematic of YOLOv1 to v12

Version	Year	Accuracy Metric	Speed Metric	Efficiency Improvements
V_1	2015	63.4% mAP(standard), 52.7% mAP (Fast)	2x faster than contemporaries	-
V_2	2017	72.9% Top-1, +4% mAP	-	Reduced computation
V_3	2018	77.2% Top-1	22ms (320×320)	Superior to ResNet
V_4	2020	43.5% AP, 65.7% AP50	>50 FPS	Cross-stage connections
V_5	2020	-	-	-
X	2021	+5.9% AP	-	-
V_6	2022	35.9%-57.2% AP	29-1234 FPS	Quantization support
V_7	2022	56.8% AP	≥30 FPS	39% fewer params
V_8	2023	53.9% AP (v8X)	280 FPS (v8X)	-
V_9	2024	+1.7% AP	-	16% fewer params, 27% less computation
V_10	2024	-	1.8x RT-DETR-R18	Lightweight design
V_11	2024	~47% accuracy	2-6ms latency	-
V_12	2025	49% mAP50-95	1-5ms latency	7x7 separable convolutions

References

[1] REDMON J, DIVVALA S, GIRSHICK R, etc. You Only Look Once: Unified, Real-Time Object Detection[J/OL]. arXiv preprint arXiv:1506.02640, 2015.

[2] REDMON J, FARHADI A. YOLO9000: Better, Faster, Stronger[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 7263-7271.

[3] REDMON J, FARHADI A. YOLOv3: An Incremental Improvement[J/OL]. arXiv preprint arXiv:1804.02767, 2018.

[4] LIN T Y, DOLLÁR P, GIRSHICK R, etc. Feature Pyramid Networks for Object Detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 2117-2125.

[5] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. YOLOv4: Optimal Speed and Accuracy of Object Detection[J/OL]. arXiv

ISSN 2959-6157

preprint arXiv:2004.10934, 2020.
[6] DO T. Evolution of YOLO algorithm and YOLOv5: The state-of-the-art object detection algorithm[D]. Oulu: Oulu University of Applied Sciences, 2021.
[7] GE Z, LIU S, WANG F, et al. YOLOX: Exceeding YOLO series in 2021[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 8377-8386.
[8] LI C, LI L, JIANG H, et al. YOLOv6: A single-stage object detection framework for industrial applications[J]. arXiv preprint arXiv:2209.02976, 2022.
[9] WANG C Y, BOCHKOVSKIY A, LIAO H Y M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 7464-7475.
[10] HUSSAIN M. YOLOv1 to v8: Unveiling each variant—A comprehensive review of YOLO[J]. IEEE Access, 2024, 12: 40573-40627. DOI: 10.1109/ACCESS.2024.3378568.
[11] WANG C Y, YEH I H, LIAO H Y M. YOLOv9: Learning what you want to learn using programmable gradient information[J]. arXiv preprint arXiv:2402.13616, 2024.
[12] WANG A, CHEN H, LIU L, et al. YOLOv10: Real-time end-to-end object detection[J/OL]. 2024.
[13] KHANAM R, HUSSAIN M. YOLOv11: An overview of the key architectural enhancements[J/OL]. 2024.
[14] ALIF M A R, HUSSAIN M. YOLOv12: A breakdown of the key architectural features[J/OL]. 2025.
[15] RAGAB M G, ABDULKADIR S J, MUNEER A, et al. A comprehensive systematic review of YOLO for medical object detection (2018 to 2023)[J]. IEEE Access, 2023.
[16] ZHANG W, GOODWILL J, CHASE T, et al. Evaluation and integration of YOLO models for autonomous crater detection[C]//IEEE Aerospace Conference. 2024.