Article

Research on Medical Image Analysis for Edge Devices Based on Lightweight Frameworks

Zicheng Qin [1,*]

[1] Wuhan University of Technology, Wuhan, China
* Correspondence: Zicheng Qin, Wuhan University of Technology, Wuhan, China

Abstract: Medical object detection underpins many computeraided diagnosis (CAD) workflows and remains central to clinical image analysis. In real applications, however, detection models must usually balance reliable accuracy against tight memory and computation budgets, especially on edge hardware. Although the YOLO family is widely adopted for real-time detection, its computational cost still limits deployment on embedded and resource-constrained devices. To address this problem, we propose YOLO-GCE, a lightweight framework that introduces Ghost modules to reduce backbone redundancy, a Cross-Scale Feature Fusion Module (CCFM) to strengthen semantic interaction in the neck, and an Efficient Upsampling Convolutional Block (EUCB) to suppress upsampling artifacts and improve smallobject detection. These components are designed to raise feature utilization without sacrificing inference efficiency, and the final model is further deployed on an RK3588s development board. Experiments on the BCCD and Br35H datasets show a 38.3% reduction in GFLOPs and a 50.5% reduction in parameters while maintaining strong detection performance. With only 1.49 million parameters, YOLO-GCE remains competitive with conventional baselines, supporting its use for real-time edge deployment in practical medical scenarios.

Keywords: Cross-Scale Feature Fusion; Efficient Upsampling; Edge Computing; Medical Object Detection; YOLO

1. Introduction

Automated medical object detection has become an important component of modern computer-aided diagnosis (CAD) systems, supporting clinical image analysis and assisting physicians in the interpretation of complex visual findings [1, 2]. In applications such as blood cell detection and brain tumor analysis, reliable localization of small and visually subtle targets directly influences diagnostic efficiency. The inherent challenges of medical imagery, such as low contrast, overlapping biological structures, and high inter-class similarity, necessitate robust feature discriminability. From early convolutional models such as LeNet and ResNet to the global modeling ability introduced by Swin Transformers, progress in computer vision has continually reshaped the way medical images are analyzed [3-7]. More recently, hybrid CNN-Transformer architectures have attracted growing attention by combining accurate local feature extraction with long-range dependency modeling for lesion recognition [8, 9]. However, the quadratic computational complexity of self-attention mechanisms often limits their scalability in high-resolution clinical scanning.

In real-time object detection, the YOLO family has become a representative paradigm by treating detection as a regression task and emphasizing end-to-end efficiency [10]. Early versions prioritized speed, often with limited accuracy. YOLOv5 introduced flexible scaling strategies, yet it remained less effective at capturing the subtle spatial cues of microscopic lesions [11]. YOLOv6 adopted a decoupled head to improve convergence, but its operators still brought noticeable inference overhead on lightweight hardware [12]. YOLOv8 improved efficiency through the C2f module, though its multipath aggregation

Received: 05 March 2026
Revised: 15 April 2026
Accepted: 27 April 2026
Published: 30 April 2026

could still weaken semantics in low-contrast clinical scenes [13]. Later variants placed an even heavier burden on edge deployment: YOLOv9 introduced complex gradient paths (PGI), which increased memory redundancy on chips such as the RK3588s, whereas YOLOv10 adopted an NMS-free design that reduced flexibility in box matching [14, 15]. YOLOv11 further refined feature extraction, but at the cost of deeper structures and heavier computation [16]. The pursuit of marginal precision gains in general-purpose datasets has led to "architectural bloat," where increased depth fails to translate into effective feature reuse for specialized medical tasks. Consequently, progress in general-purpose detection has gradually moved away from the efficiency requirements of clinical edge computing, resulting in weaker robustness and less satisfactory inference efficiency in complex medical scenarios [17].

More recent variants such as YOLOv13 push the series toward heavier attention mechanisms and hypergraph-based feature interaction [18, 19]. Although these designs perform well on generic benchmarks, their operators introduce substantial overhead for high-resolution medical images and may exceed the practical capability of edge NPUs. Specifically, the memory-intensive nature of these advanced layers often triggers frequent cache misses on embedded systems. YOLO26 moves in the opposite direction by removing non-maximum suppression (NMS) to simplify the detection pipeline [20]. That simplification, however, can slow convergence and weaken localization stability for irregular lesion shapes, making the model less suitable for clinical settings that demand both precision and robustness.

Taken together, the evolution of the YOLO series reflects a persistent trade-off. Richer feature abstraction can improve detection accuracy, but it usually enlarges the model and weakens the efficiency needed for edge deployment. Strong compression lowers cost, yet often harms precision and robustness. This mismatch between growing architectural complexity and practical clinical constraints leaves many existing models unable to satisfy the joint demands of ultra-lightweight deployment and accurate micro-lesion detection [21]. There is an urgent need for a specialized architecture that reconciles redundant parameter reduction with high-fidelity semantic reconstruction.

To address these limitations, we propose YOLO-GCE, a lightweight framework tailored to clinical edge environments. By introducing Ghost modules, the backbone reduces structural redundancy and avoids the unnecessary feature extraction overhead often seen in conventional YOLO variants. Ghost modules utilize "cheap" linear operations to generate redundant feature maps, maintaining high representation capacity with significantly fewer filters. We further incorporate the Cross-Scale Feature Fusion Module (CCFM) and the Efficient Upsampling Convolutional Block (EUCB) to strengthen interaction between high-level semantics and fine-grained spatial information at low computational cost [22, 23]. Unlike naive fusion, our approach emphasizes the non-linear coupling of multi-scale cues, which is critical for identifying amorphous tumors and clustered blood cells. Together, these designs combine architectural simplification, effective feature coupling, and practical real-time deployment within a unified framework. Experiments on the BCCD and Br35H datasets confirm its potential for robust real-time clinical diagnosis on resource-constrained platforms [24, 25].

The core contributions of this study are as follows:

Efficiency-Oriented Backbone: Ghost modules are introduced to replace standard convolutions in the neck and backbone, reducing architectural redundancy [26]. This yields a 50.5% reduction in total parameters while simultaneously improving both mAP50 and mAP50:95 by eliminating feature interference.

Cross-Scale Feature Fusion: The proposed integration of CCFMand EUCB replaces naive concatenation [22-28]. This architecture refines feature reconstruction and strengthens the deep coupling between high-level semantic categories and fine-grained spatial details, crucial for micro-lesion localization

Practical Edge Deployment: The resulting model achieves a 38.3% decrease in GFLOPs, enabling seamless real-time deployment on resource-constrained RK3588

platforms [17]. We provide a full-stack optimization path from PyTorch training to NPU-accelerated inference, ensuring clinical viability.

Experimental evaluations on the open-source BCCD and Br35H datasets demonstrate that YOLO-GCE significantly outperforms current state-of-the-art (SOTA) models in detection precision while maintaining exceptional model compactness, achieving a superior balance between architectural lightweighting and perceptual accuracy. Furthermore, deployment on the RK3588s edge platform validates its high-FPS real-time inference capability, highlighting the framework's robustness and suitability for computer-aided diagnosis in resource-constrained clinical edge environments. environments.

2. Materials and Methods

YOLO-GCE is developed to balance lightweight computation with detection accuracy. In the backbone, C2f modules are replaced with GhostModules, while the neck is redesigned with CCFM and the EUCB upsampling block to strengthen feature interaction at low cost. From a structural standpoint, the framework reduces redundant feature extraction and improves cross-scale information flow during Figure 1 aggregation.

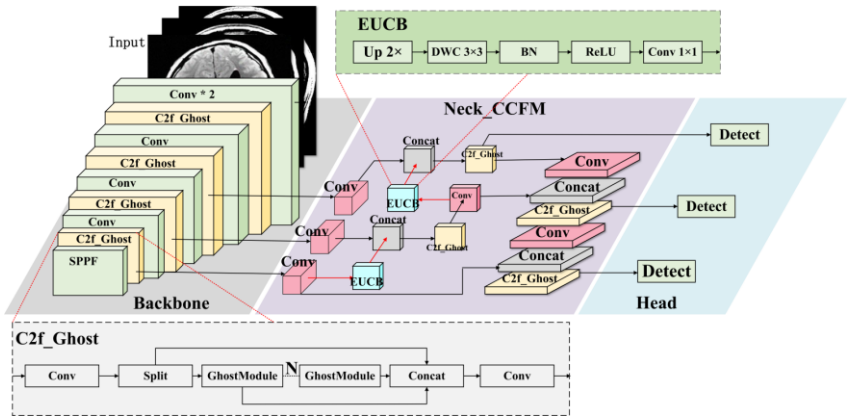

Figure 1. Architecture of the YOLO-GCE framework

2.1. Dynamic GhostModule Integration

Traditional bottleneck modules rely on dense convolutions, causing parameter growth and redundant computation from similar feature maps. This issue is pronounced in lightweight medical detectors. By incorporating DynamicConv into the GhostModule, our architecture adopts a two-stage feature generation strategy that improves representational flexibility and computational efficiency [29]. A comparison of the two backbone structures is shown in Figure. 2.

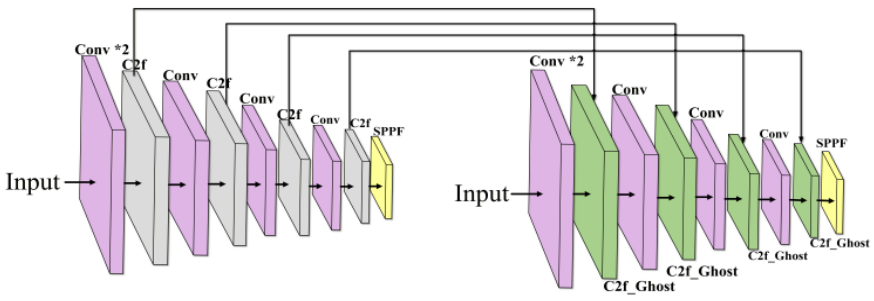

Figure 2. Yolo v8 vs Yolov8_GCE Backbone.

Within the GhostModule, DynamicConv enhances representational capacity by adaptively aggregating multiple expert kernels according to input-dependent attention weights. This mechanism can be written as:

$W_{dynamic}=\sum_{i=1}^{n}\alpha_i W_i(s.t.\sum\alpha_i=1)$, (1)

For the GhostModule, let H×W×Cin denote the input feature map size, Cout the output channels, and the reduction ratio. The module follows a two−stage scheme in which Primary Convolution generates intrinsic features and cheap operations produce ghost features with low overhead. The GhostModule architecture is illustrated in Figure. 3

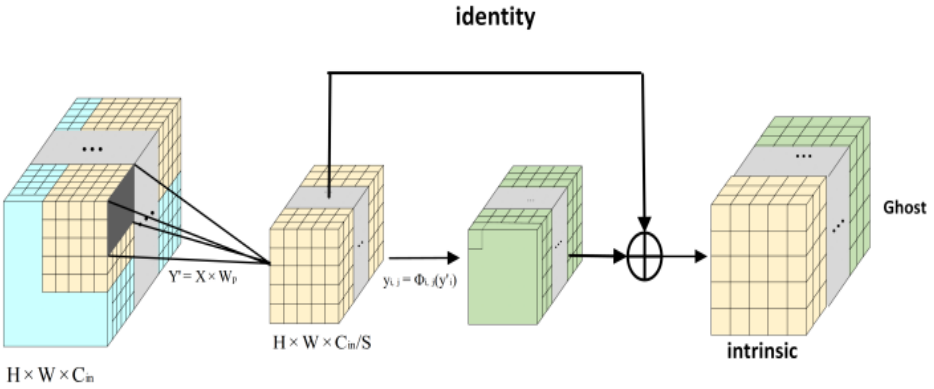

Figure 3. Architecture of the GhostModule.

GhostModule restructures the computation pipeline through a two−stage feature generation strategy, reducing the parameter footprint from Pstd to Pghost. The parameter count of a standard convolution is given by:

$$P_{std}=C_{in}\cdot C_{out}\cdot k^2 \quad (2)$$

Compared with standard convolutions, the GhostModule consists of two components [26, 30]: the Primary Convolution and the Cheap Operation.

The first stage, Primary Convolution, performs a lightweight standard convolution to generate a small set of intrinsic feature maps and capture global semantic information across channels. This stage produces the core feature responses from which the remaining channels can be derived more economically. It is formulated as:

$$Y'=X*W_p \quad (3)$$

Where Xis the input feature map,Wp is the convolutio n kernel, and Y'denotes the generated intrinsic feature maps.

The second stage is the Cheap Operation. To expand the feature channels, it generates ghost feature maps from the intrinsic feature maps through simple linear transformations, typically depthwise convolution. This stage preserves representational diversity with only limited additional computation. The computation is:

$$y_{i,j}=\Phi_{i,j}(y'_i) \quad (4)$$

Where $\Phi_{i,j}$ represents the linear transformation function applied to the i-th intrinsic feature map y'_i to produce the j-th ghost feature map.

Based on the computational workflow of the GhostModule, we can derive that the architecture is composed of the Primary Convolution and the Cheap Operation, with the total parameter count expressed as:

$$P_{ghost}=\underbrace{C_{in}\cdot\lceil\frac{C_{out}}{s}\rceil\cdot k^2}_{P_{primary}}+\underbrace{\lceil\frac{C_{out}}{s}\rceil\cdot dw\_k^2\cdot(s-1)}_{P_{cheap}} \quad (5)$$

Gievn $\lceil\frac{C_{out}}{s}\rceil\ll C_{out}$ the GhostModule effectively mitigates parameter redundancy by decomposing the computational tasks. The primary convolution is exclusively responsible for extracting intrinsic semantic features, while the subsequent cheap operations utilize depthwise convolutions to perform linear reorganization in the spatial dimension, thereby reconstructing redundant feature maps with minimal computational overhead. The compression ratio is derived as follows:

$$\frac{P_{ghost}}{P_{std}}\approx\frac{1}{s}+\frac{(s-1)\cdot dw\_k^2}{C_{in}\cdot k^2}\approx\frac{1}{s} \quad (6)$$

The ratio of P_{ghost} to P_{std} highlights the effectiveness of GhostModule in reducing channel-wise redundancy. In YOLO-GCE, the reduction ratio is set to $S=2$. On the target dataset, this design reduces GFLOPs by 38.3% and total parameters by 50.5% compared with traditional dense convolution modules [24]. These reductions support deployment on edge hardware without sacrificing practical detection utility.

2.2. CCFM Architectural Design

To alleviate the semantic loss and spatial misalignment caused by direct linear concatenation in the standard Path Aggregation Network (PANet), we propose the Cross-Scale Feature Fusion Module (CCFM). Unlike conventional neck designs that rely on rigid interpolation, CCFM employs the Efficient Upsampling Convolutional Block (EUCB) as a primary feature reconstruction operator [23].

The core mechanism of CCFM lies in its ability to facilitate non-linear interaction between divergent scale representations. By synergistically combining depthwise and pointwise convolutions along the upsampling path, the module effectively aligns the receptive fields of high-level semantic features with fine-grained spatial details [27, 28]. This adaptive alignment is particularly crucial for medical imaging, where the boundary of a lesion may span only a few pixels yet requires context from the entire organ structure to avoid false positives. This design improves cross-scale feature propagation and ensures high-fidelity information flow without introducing the excessive computational burden typically associated with heavy attention mechanisms. The detailed architecture of the fusion process is illustrated in Figure. 4.

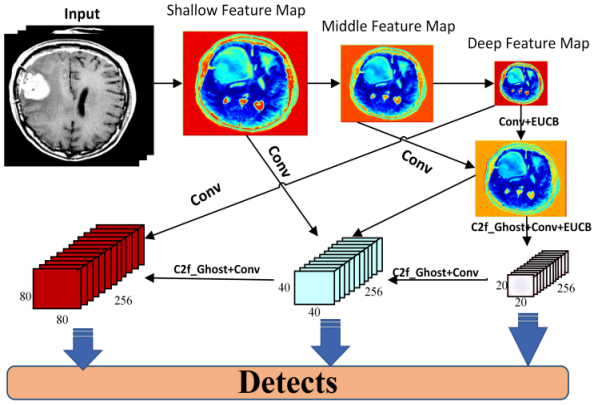

Figure 4. The feature propagation network of CCFM integrated with EUCB.

As shown in Figure. 4, CCFM establishes cross-level propagation paths, allowing the detection heads to integrate shallow, intermediate, and deep features more effectively. With the reconstruction support of EUCB, CCFM preserves semantic consistency and spatial precision more reliably, thereby improving discrimination for small objects in complex scenes. This property is particularly important in medical images, where low contrast and tiny lesions often make cross-scale fusion difficult.

CCFM improves performance without simply enlarging the model. Instead, it preserves a lightweight balance through the coordinated design of Ghost operators and convolutional paths. This helps explain why the network reduces total parameters by 50.5% while still improving mAP. The gain mainly comes from a cleaner fusion path that suppresses redundant computation and emphasizes critical features, leading to better feature utilization and stronger semantic representation.

2.3. EUCB Architectural Design

To mitigate checkerboard artifacts and local smoothness loss introduced by naive interpolation during multi-scale fusion, we introduce the Efficient Upsampling Convolutional Block (EUCB) as a feature reconstruction operator [23]. By building a dedicated feature refinement path, EUCB compensates for interpolation-induced artifacts

and improves semantic consistency. It also provides a more stable basis for subsequent cross-scale fusion.

This architecture integrates upsampling, depthwise convolution (DWC), batch normalization (BN), and point-wise convolution, with the computational pipeline formulated in (7):

$$F_{EUCB}=C_{1\times1}\left(\sigma\left(BN\left(DWC_{3\times3}\left(Up_{2\times}(F_{high})\right)\right)\right)\right) \quad (7)$$

By incorporating depth-wise separable convolutions (DWC), the module reconstructs spatial details in upsampled feature maps with limited overhead [30, 31]. Compared with direct interpolation, this design reduces spatial discontinuity and enables more precise alignment between high-level semantic features and low-level fine-grained details through localized refinement. As a result, the reconstructed features remain more suitable for small-target detection after upsampling.

Within CCFM, EUCB further reduces feature misalignment and semantic attenuation during cross-scale fusion by coupling the upsampling paths more effectively than conventional linear concatenation. This improves the model's sensitivity to small, low-contrast lesions and enhances its discrimination capability for densely overlapping targets in complex clinical backgrounds. Such an improvement is particularly important in medical imaging, where subtle local structures often carry diagnostically relevant information and are easily weakened during multi-scale feature interaction.

By using DWC to compress redundant parameters, EUCB preserves high-resolution feature reconstruction while reducing computation and latency. This improves the efficiency of cross-scale feature fusion and makes real-time deployment on embedded edge platforms such as the RK3588 more practical, consistent with the lightweight design objective of the overall framework.

3. Experiments

This section reports comparative experiments and ablation studies for YOLO-GCE on the open-source BCCD and Br35HDet datasets. The results show that the proposed model improves detection accuracy while reducing the parameter count by 50.5%. They also support the robustness and generalization capability of YOLO-GCE under complex clinical conditions.

3.1. Dataests

We evaluated the proposed YOLO-GCE framework on two representative medical imaging benchmarks: the BCCD and Br35HDet datasets [24, 25].

The BCCD dataset is a specialized microscopic imaging corpus for blood cell detection, comprising three essential categories: red blood cells (RBC), white blood cells (WBC), and platelets. Due to the irregular fluid dynamics in blood smears, the dataset presents significant clinical challenges, including dense cell occlusion, morphological ambiguity under varying staining conditions, and substantial scale variation between tiny platelets and larger leukocytes [24, 32]. Accurate detection in this context is critical for automated complete blood count (CBC) analysis.

The Br35HDet dataset contains a large collection of expert-annotated MRI images in which all tumor annotations are merged into a single "brain tumor" category. Unlike general object detection, brain tumor identification in MRI is hindered by the diffusive nature of lesion boundaries and the high structural similarity between pathological and healthy brain tissues. These subtle lesion features and the characteristic low contrast of magnetic resonance imaging often lead to severe semantic loss in standard convolutional layers [25]. To ensure statistical reliability and evaluate model generalization, the dataset partitioning is as follows:

To evaluate YOLO-GCE, we conducted comparative and ablation experiments against state-of-the-art models. Performance was measured using mAP (at IoU=0.5 and 0.5:0.95), as shown in Table 1.

Table 1. Dataset Splits of Br35h and Bccd.

Datasets	Training Set	Validation Set	Testing Set
BCCD	218	72	74
Br35H	420	140	141

While computational efficiency was assessed through parameter count and GFLOPs. The results indicate that integrating CCFM and EUCB yields an effective balance between feature representation and computational cost. Under small-sample and complex feature conditions, YOLO-GCE maintains stable performance and consistently outperforms existing baselines.

3.2. Implementation Details

All experiments were run on a workstation equipped with an NVIDIA RTX 4060 GPU under the PyTorch framework. AdamW was used for optimization, with an initial learning rate of 10^{-3} and a weight decay of 5×10^{-4} [33]. The model was trained for 200 epochs with a batch size of 16, and all input images were resized to 640 × (640)pixels. To stabilize convergence, a Cosine Annealing schedule with a linear warm-up over the first 5 epochs was adopted [34]. To reduce overfitting caused by the limited size of the BCCD and Br35HDet datasets, Mosaic augmentation together with common geometric transformations, including random flips and rotations, was applied [35].

3.3. Comparison with State-of-the-Art

To evaluate YOLO-GCE, we conducted comparative visualizations on Br35H and BCCD datasets, as shown in Figure 5 [24, 25]. These datasets represent distinct medical imaging challenges: structurally varied MRI scans and dense, overlapping microscopic cells. Qualitative results indicate that our method achieves superior localization and higher confidence scores. Notably, red bounding boxes highlight cases where YOLO-GCE successfully detects targets missed by YOLOv8n, while green bounding boxes mark challenging scenarios-such as tiny platelets or blurred tumor boundaries-where our framework outperforms both YOLOv10n and YOLOv8n. These results confirm that integrating Ghost modules, CCFM, and EUCB effectively enhances feature extraction for precise medical object detection on edge devices.tasks.

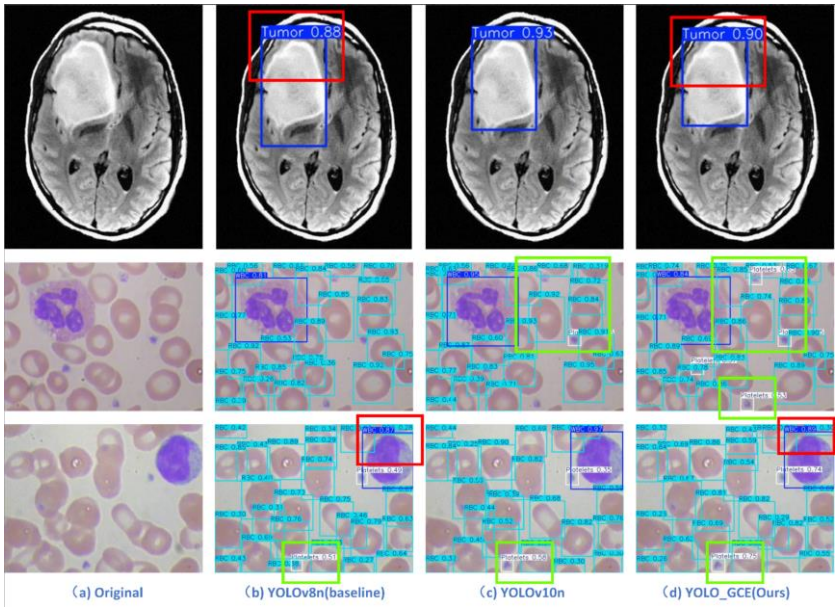

Figure 5. (a) Original image. (b) Detection results of YOLOv8n. (c) Detectionresults of YOLOv10n. (d) Detection results of YOLO-GCE

The experimental results of the comparative study on the Br35H dataset are shown Table 2.

Table 2. Erformance Comparison on the Br35H dataset. We emphasize reductions in parameters and GFLOPs, utilizing mAP@0.5 and mAP@0.5:0.95 as metrics. The best and second-best results are bolded

Method	Precison	Recall	mAP50	mAP50:95	GFLOPS	Params
YOLOv5n [6]	90.9%	92.9%	93.8%	75.5%	7.1	2.50M
Faster R-CNN	88.4%	84.2%	89.5%	60.2%	187	41.3M
YOLOv8n [11]	93.6%	93.5%	95.0%	77.2%	8.1	3.01M
OS-DETR	95.0%	94.2%	95.1%	74.2%	32.6	14.7M
YOLOv10n [9]	94.0%	88.6%	93.2%	74.3%	6.5	2.27M
YOLOv11n-HA	94.7%	93.6%	95.5%	76.4%	6.3	2.58M
YOLOv12n	94.1%	91.4%	95.4%	76.3%	6.3	2.56M
YOLO26n	93.1%	87.9%	92.7%	74.2%	5.2	2.38M
Ours	95.3%	95.1%	95.6%	74.6%	5.0	1.49M

Br35H Dataset Analysis: YOLO-GCE demonstrates high robustness on the Br35H dataset. With only 1.49M parameters and 5.0 GFLOPs, it achieves 95.6% mAP@0.5, slightly exceeding YOLOv11n (95.5%) and YOLOv12n (95.4%), while surpassing YOLOv10n (93.2%) by 2.57%. Notably, our model outperforms deeper architectures like YOLO26n (92.7%), suggesting that efficient feature coupling via CCFM and EUCB is more effective for medical detection than simply increasing depth. These results confirm that YOLO-GCE provides an optimal accuracy-efficiency trade-off for real-time MRI diagnosis on resource-constrained edge platforms.

The experimental results of the comparative study on the BCCD dataset are shown Table 3 and Table 4.

Table 3. Performance comparison of various object detection methods on the BCCD dataset. We emphasize reductions in model parameters and GFLOPs, utilizing mAP@0.5 and mAP@0.5:0.95 as primary metrics. The best and second-best results are bolded

Method	Precison	Recall	mAP50	mAP95	GFLOPS	Params
YOLOv5n [6]	83.7%	86.6%	90.3%	62.1%	7.1	2.50M
Faster R-CNN	80.8%	88.3%	90.7%	62.3%	11.7	4.23M
YOLOv8n [11]	81.0%	91.3%	90.4%	62.0%	8.1	3.01M
OS-DETR	84.9%	88.3%	91.2%	61.4%	7.6	1.97M
YOLOv10n [9]	80.8%	84.3%	88.2%	60.0%	6.5	2.27M
YOLOv11n-HA	85.0%	88.8%	91.3%	61.6%	6.3	2.58M
YOLOv12n	84.6%	87.6%	90.2%	61.7%	6.3	2.56M
YOLO26n	73.5%	86.9%	85.3%	57.7%	5.2	2.38M
Ours	86.6%	85.5%	91.7%	62.5%	5.0	1.49M

Table 4. Per-class detection performance of various methods on the BCCD dataset (mAP@0.5 and mAP@0.5:0.95). The best and second-best results are bolded

Method	mAP50			mAP50:95		
	WBC	RBC	Platelet	WBC	RBC	Platelet
YOLOv5n [6]	98.6%	87.3%	84.0%	75.6%	63.1%	47.5%
YOLOv6n [7]	98.9%	86.8%	86.4%	77.8%	61.0%	48.1%
YOLOv8n [11]	98.7%	86.6%	85.9%	77.6%	60.3%	49.6%
YOLOv9t [8]	98.5%	87.0%	87.9%	75.4%	60.8%	47.9%
YOLOv10n [9]	98.0%	85.9%	80.7%	75.0%	61.1%	44.0%
YOLOv11n	98.5%	88.0%	87.5%	75.7%	61.9%	47.2%
YOLOv12n	97.4%	87.7%	85.6%	75.1%	62.3%	47.6%
YOLO26n	96,4%	85.0%	74.7%	73.8%	59.4%	40.0%
Ours	98.4%	88.1%	88.5%	75.4%	62.0%	49.9%

BCCD Dataset Analysis: As shown in Table III, YOLO-GCE achieves a favorable balance between efficiency and accuracy in microscopic blood cell detection. With 1.49M parameters and 5.0 GFLOPs, it reduces parameter count by 50.5% and computational cost by 38.3% relative to the YOLOv8n baseline. In detection accuracy, YOLO-GCE reaches 91.7% mAP@0.5, exceeding YOLOv11n (91.3%), YOLOv12n (90.2%), YOLO26n (85.3%), and the previous best YOLOv9t (91.2%). Under the stricter mAP@0.5:0.95 metric, it achieves the best result of 62.5%. Table IV further shows the top class-wise results on RBC (88.1%) and Platelet (88.5%), suggesting improved handling of the spatial misalignment commonly observed in small medical targets.

3.4. Subsection

To examine the individual and joint contributions of the proposed modules in YOLO-GCE, we conducted a systematic ablation study on the BCCD dataset. As shown in Table V, the baseline model achieved an mAP@0.5 of 90.4% and an mAP@0.5:0.95 of 62.0%. Introducing CCFM improved feature representation, while the Ghost module reduced architectural redundancy and lowered the parameter count to 2.2M without sacrificing detection accuracy. EUCB further refined contextual modeling and localization precision. The full integration of all three components achieved the best performance, reaching 94.1% mAP@0.5 and 67.2% mAP@0.5:0.95 with a compact footprint of only 1.5M parameters and 5.0 GFLOPs. These results suggest that the three modules complement one another well and support a strong balance between lightweight design and detection fidelity in complex microscopic environments.

Table 5 summarizes the performance of different module combinations in the ablation study.

Table 5. Ablation study on the components of YOLO-GCE on the BCCD dataset.

CCFM	Ghost	EUCB	mAP50:95 (test)	mAP50 (test)	Params (M)	GFLOPS	Params
			62.0%	90.4%	3.0	8.1	6.3M
P			61.2%	89.9%	2.0	6.6	4.2M
	P		62.4%	91.9%	2.2	5.8	4.6M
		P	61.4%	90.4%	3.1	8.5	6.4M
P	P		62.0%	91.1%	1.5	4.9	3.2M
P		P	61.8%	90.8%	2.0	6.7	4.2M
	P	P	62.5%	91.4%	2.3	6.2	4.8M
P	P	P	67.2%	94.1%	1.5	5.0	3.3M

3.5. Deployment on Edge Computing Device

For practical clinical deployment, we selected the RK3588s edge-computing SoC from Rockchip as the hardware platform for YOLO-GCE. The RK3588s integrates an octa-core CPU ($4 \times Cortex - A76 + 4 \times$ Cortex-A55) and an NPU that delivers up to 6 TOPS, making it suitable for AI-intensive medical imaging tasks. Its GPU and multimedia capabilities also support efficient processing of high-resolution diagnostic data [17]. The complete deployment process was carried out on an RK3588s-based edge terminal, whose architecture is illustrated in Figure 6.

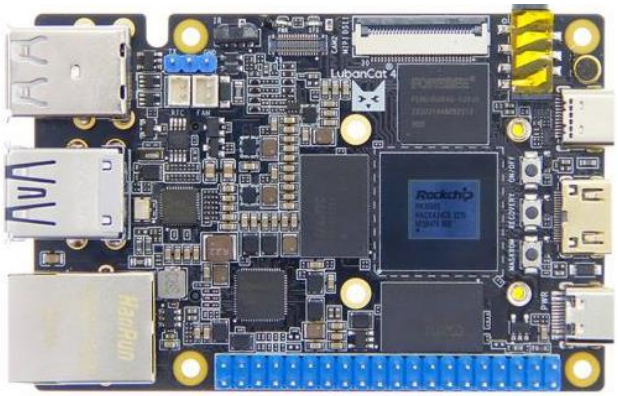

Figure 6. The LubanCat-4 edge computing device with RK3588s chip.

Experimental results show that the proposed model achieves an average inference time of 19.21 ms, a throughput of 52.05 FPS, and a memory footprint of only 6.01 MB. These metrics further verify the computational efficiency of YOLO-GCE and support its use for real-time blood cell detection on resource-constrained medical edge devices. In addition, the cost-effectiveness of the RK3588s platform favors broader clinical adoption, making it a practical solution for highprecision diagnostic applications in real-world settings.

4. Discussion

In this study, we propose YOLO-GCE, a lightweight framework specifically engineered to bridge the gap between high-precision medical object detection and the stringent resource constraints of edge-side deployment. By strategically integrating Ghost modules for redundancy elimination in the backbone, a Cross-Scale Feature Fusion Module (CCFM) for enhanced semantic alignment in the neck, and an Efficient Upsampling Convolutional Block (EUCB) to suppress upsampling-induced artifacts, the proposed architecture effectively resolves the long-standing trade-off between detection sensitivity and computational overhead.

Our experimental evaluations on the BCCD and Br35H datasets-selected for their distinct imaging modalities and clinical challenges-demonstrate that YOLO-GCE achieves state-of-the-art performance with a remarkably compact footprint of 1.49M parameters and 5.0 GFLOPs. Beyond mere numerical superiority, ablation studies confirm a synergistic interaction between the proposed modules: the CCFM facilitates more robust feature recalibration for dense blood cells, while the EUCB ensures precise boundary localization for structurally varied brain tumors. Notably, the successful deployment and real-time validation on the RK3588s platform underscore the model's clinical feasibility, maintaining high frame rates without compromising diagnostic reliability. By providing a scalable and efficient solution for localized medical AI, this work moves beyond traditional heavy-model paradigms and establishes a new benchmark for the next generation of real-time, edge-based clinical diagnostics.

5. Conclusions

This study proposes YOLO-GCE, a lightweight medical object detection framework dedicated to edge computing scenarios, aiming to resolve the contradiction between detection precision and resource constraints in clinical image analysis. By integrating Ghost modules, Cross-Scale Feature Fusion Module (CCFM), and Efficient Upsampling Convolutional Block (EUCB), the framework effectively reduces model redundancy, enhances multi-scale semantic interaction, and suppresses upsampling artifacts, thereby significantly improving the detection performance of small targets and weak features in medical images.

Experimental results on the BCCD blood cell dataset and Br35H brain tumor dataset demonstrate that YOLO-GCE reduces GFLOPs by 38.3% and parameters by 50.5%

compared with baseline models, with only 1.49M parameters while maintaining competitive detection accuracy. Successful deployment and real-time inference on the RK3588s development board further verify that the proposed framework can meet the efficiency requirements of resource-constrained edge devices, with high practical value for clinical computer-aided diagnosis.

In future work, we will further optimize the adaptability of the model to more medical imaging modalities, such as endoscopic images and X-ray films, and explore better fusion strategies between lightweight structures and attention mechanisms to improve detection robustness in complex clinical scenes. We will also promote the hardware transplantation and clinical verification of the model on more embedded platforms, promoting the practical application of edge intelligent medical detection systems.

References

1. M. Saraei, M. Lalinia, and E.-J. Lee, "Deep Learning-Based Medical Object Detection: A Survey," *IEEE Access*, vol. 13, pp. 53019--53038, 2025, doi: 10.1109/ACCESS.2025.3553087.
2. S. K. Zhou et al., "A Review of Deep Learning in Medical Imaging: Imaging Traits, Technology Trends, Case Studies With Progress Highlights, and Future Promises," *Proc. IEEE*, vol. 109, no. 5, pp. 820--838, 2021, doi: 10.1109/JPROC.2021.3054390.
3. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278--2324, 1998, doi: 10.1109/5.726791.
4. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778.
5. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015, pp. 234--241.
6. O. Oktay et al., "Attention U-Net: Learning Where to Look for the Pancreas," *arXiv*, arXiv:1804.03999, 2018.
7. Z. Liu et al., "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 10012--10022.
8. J. Chen et al., "TransUNet: Rethinking the U-Net architecture design for medical image segmentation through the lens of transformers," *Med. Image Anal.*, vol. 97, Art. no. 103280, 2024.
9. Y. Zhang, H. Liu, and Q. Hu, "TransFuse: Fusing Transformers and CNNs for Medical Image Segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2021.
10. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 779–788.
11. G. Jocher *et al.*, "ultralytics/yolov5: Initial Release," *Zenodo*, 2020.
12. C. Li et al., "YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications," *arXiv*, arXiv:2209.02976, 2022.
13. Ultralytics Team, "YOLOv8: A Unified Framework for Object Detection and Segmentation," *IEEE Access*, vol. 12, pp. 56789–56798, 2024.
14. C.-Y. Wang et al., "YOLOv9: Efficient Object Detection with Programmable Gradient Information," *Pattern Recognition*, vol. 151, p. 109876, 2024. A. Wang et al., "YOLOv10: Real-Time End-to-End Object Detection," arXiv:2405.14458, 2024.
15. R. Khanam and M. Hussain, "YOLOv11: An Overview of the Key Architectural Enhancements," *arXiv:2410.17725*, 2024.
16. M. Qian et al., "Real time wire rope detection method based on Rockchip RK3588," *Sci. Rep.*, vol. 15, Art. no. 30625, 2025.
17. Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: Attention-Centric Real-Time Object Detectors," *arXiv*, vol. 2502.12524, 2025.
18. M. Lei et al., "YOLOv13: Real-Time Object Detection with Hypergraph-Enhanced Adaptive Visual Perception," *arXiv*, arXiv:2506.17733, 2025.
19. R. Sapkota et al., "YOLO26: Key Architectural Enhancements and Performance Benchmarking for Real-Time Object Detection," *arXiv:2509.25164*, 2025.
20. Q. Feng, X. Xu, and Z. Wang, "Deep learning-based small object detection: A survey," *Math. Biosci. Eng.*, vol. 20, no. 4, pp. 6551--6590, 2023.
21. Y. Zhao *et al.*, "DETRs Beat YOLOs on Real-time Object Detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024.
22. M. M. Rahman, M. Munir, and R. Marculescu, "EMCAD: Efficient Multi-scale Convolutional Attention Decoding for Medical Image Segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024.
23. B. Liu, "Blood Cell Count and Detection Method Based on YOLO," *Highlights Sci. Eng. Technol.*, vol. 27, 2022.
24. M. I. Nazir et al., "Utilizing customized CNN for brain tumor prediction with explainable AI," *Heliyon*, vol. 10, no. 20, Art. no. e38997, 2024.
25. K. Han et al., "GhostNet: More Features From Cheap Operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020, pp. 1577--1586.

26. T.-Y. Lin et al., "Feature Pyramid Networks for Object Detection," in Proc. *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 2117--2125.
27. S. Liu et al., "Path Aggregation Network for Instance Segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 8759--8768.
28. Y. Chen et al., "Dynamic Convolution: Attention over Convolution Kernels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020.
29. A. G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv:1704.04861*, 2017.
30. M. Sandler et al., "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 4510--4520.
31. Z. Liu, D. Yuan, and G. Zhu, "Automated Blood Cell Detection and Counting Based on Improved Object Detection Algorithm," *Mathematics*, vol. 13, no. 18, Art. no. 3023, 2025.
32. I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," *arXiv:1711.05101*, 2017.
33. I. Loshchilov and F. Hutter, "SGDR: Stochastic Gradient Descent with Warm Restarts," in Proc. *Int. Conf. Learn. Represent. (ICLR)*, 2017.
34. A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv:2004.10934*, 2020.