Biomedical Signal Processing and Control 114 (2026) 109283

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc

BTDet: Towards lightweight and enhanced feature aggregation network for brain tumor detection

Yi Li [a], Huiying Xu [a,*], Xinzhong Zhu [a,b,d], Xiao Huang [c], Hongbo Li [d]

[a] *School of Computer Science and Technology of Zhejiang Normal University, Jinhua, Zhejiang, 321004, China*
[b] *Research Institute of Hangzhou Artificial Intelligence, Zhejiang Normal University, Hangzhou, Zhejiang, 311231, China*
[c] *College of Education of Zhejiang Normal University, Jinhua, Zhejiang, 321004, China*
[d] *Beijing Geekplus Technology Co., Ltd, Beijing, 100101, China*

ARTICLE INFO

Keywords:
Medical image processing
Brain tumor detection
Lightweight network
Feature aggregation

ABSTRACT

Accurate brain tumor detection in magnetic resonance imaging (MRI) is essential for early diagnosis, yet remains challenging due to the heterogeneous appearance and morphology of tumors. Although deep learning approaches have shown potential, their clinical applicability is often limited by high computational cost and restricted generalization capability. To address these issues, this study introduces BTDet, an efficient and lightweight detection framework that balances performance with computational efficiency. The model incorporates several design components: Reparameterized C2f GELAN (RCG) backbone combined with a Fast Spatial Pyramid Pooling Fusion (FSPPF) module to enhance feature extraction and semantic representation; C2f Squeeze and Excitation (CSE) attention mechanism and General Depthwise Separable Convolution (GSC) block to improve multi-scale feature fusion; and lightweight dual-head to maintain detection accuracy and inference speed. On the Br35H brain tumor dataset, BTDet achieves a $mAP@50{:}95$ of 0.753, surpassing the baseline by 2.45%, while requiring only 2.26M parameters and 6.0 GFLOPs. The framework also demonstrates strong cross-domain adaptability, improving accuracy by 5.4% on the LUNA16 lung nodule detection benchmark. These results indicate that BTDet offers a practical and resource-efficient solution suitable for real-world medical imaging applications.

1. Introduction

Brain tumors represent the most common type of neoplasm affecting the Central Nervous System (CNS), and their early detection is critical for improving patient survival rates and overall quality of life [1]. These tumors are among the most aggressive and life-threatening conditions, affecting both pediatric and adult populations, and are estimated to account for approximately 85%–90% of all primary CNS tumors [2]. Currently, Magnetic Resonance Imaging (MRI) serves as the primary imaging modality for the detection and diagnosis of brain tumors due to its superior soft tissue contrast and non-invasive nature [3]. However, MRI scans generate large volumes of high-dimensional data, placing a significant burden on radiologists who must manually interpret the images. Given the heterogeneous nature of brain tumors varying in size, shape, location, and intensity—manual interpretation is not only time-consuming but also prone to diagnostic inaccuracies and inter-observer variability. Although modern medical imaging technologies, such as MRI and Computed Tomography (CT), have substantially advanced tumor detection, there remains a pressing need for further improvement in diagnostic accuracy, particularly in terms of sensitivity and specificity. These challenges have spurred increasing interest in automated, AI-driven methods to assist in the accurate and efficient detection of brain tumors.

With the continuous advancement of medical imaging technology, the automatic detection and identification of brain tumors have become particularly critical in clinical diagnosis. Notwithstanding that this task still faces numerous challenges, including the diverse morphology of tumors, blurred boundaries, low contrast between tissues, and structural variations among different patients, all of which pose significant difficulties for image recognition. Additionally, the high cost of acquiring high-quality annotated data limits the generalization capability and robustness of traditional methods [4]. In recent years, the introduction of deep learning, particularly Convolutional Neural Networks (CNNs) [5], has significantly improved the accuracy and automation of medical image processing. Compared to traditional manual feature

* Corresponding author.
E-mail addresses: leeyee@zjnu.edu.cn (Y. Li), xhy@zjnu.edu.cn (H. Xu), zxz@zjnu.edu.cn (X. Zhu), huangxiao@zjnu.edu.cn (X. Huang), Jason.li@geekplus.com (H. Li).

https://doi.org/10.1016/j.bspc.2025.109283
Received 23 October 2024; Received in revised form 2 October 2025; Accepted 29 November 2025

1746-8094/© 2025 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

extraction methods, CNNs can automatically learn high-level features from large volumes of brain MRI images, enabling more efficient tumor recognition tasks [6–8]. Nevertheless, current models still face several challenges in practical applications, such as few-shot learning, poor interpretability, and inaccurate boundary processing.

Consequently, researchers have proposed various structural optimization strategies to enhance the model's representational capacity and robustness. Among these, U-Net and its variants (e.g., U-Net++ [9], Attention U-Net [10], Residual U-Net [11]) have emerged as mainstream approaches for medical image segmentation. These models leverage an encoder–decoder architecture to achieve multi-scale feature fusion, effectively preserving spatial information in images. To improve the model's focus on critical regions, attention-based methods such as Attention U-Net have been widely adopted, significantly enhancing segmentation accuracy in tumor regions. In recent years, the success of Transformer architectures in computer vision has garnered substantial attention in the field of medical image analysis. Vision Transformer (ViT) [12] and its variants (e.g., TransUNet [13], Swin-Unet [14]), which rely on self-attention mechanisms, excel at capturing long-range dependencies, making them particularly suitable for segmenting brain tumors with complex morphology and heterogeneity. These methods have demonstrated superior performance over traditional CNN-based models on multiple public datasets, further advancing the precision of automated diagnosis. Meanwhile, Graph Neural Networks (GNNs) [15,16] have also been introduced into brain tumor analysis tasks, leveraging graph-structured representations to model relationships between tumors and surrounding tissues, thereby improving multi-modal information fusion. By representing medical images as graph structures, GNNs can extract higher-level structural information, compensating for the limitations of conventional models in modeling spatial topological relationships.

Developing an advanced approach that simultaneously addresses accuracy, robustness, lightweight architecture, and transferability is crucial for establishing efficient and reliable automated brain tumor detection systems. In this work, we introduce an efficient method for brain tumor detection, referred to as BTDet, which aims to improve detection performance through a lightweight model pattern and feature aggregation techniques. To facilitate effective feature extraction from MRI images of brain tumor, we designed Reparameterized C2f GELAN (RCG) block, which serves as the foundational backbone characterized with lightweight paradigm. Furthermore, Fast Spatial Pyramid Pooling Fusion (FSPPF) module is developed to enhance the exploration of semantic information by multi-consecutive pooling and identity connections. The model's neck integrates C2f Squeeze and Excitation (CSE) attention and General Depthwise Separable Convolution (GSC) to reinforce the multi-scale feature fusion. Additionally, we demonstrated that the implementation of two lightweight detection heads are capable to achieve superior detection performance. The main contributions of this work are concluded as follows,

- We propose BTDet, a highly efficient brain tumor detector which achieves excellent detection performance while maintaining a favorable accuracy-model size trade-off.
- The BTDet architecture incorporates several key technical contributions: the RCG and FSPPF blocks for robust basic feature extraction; the CSE and GSC modules for advanced multi-scale feature integration; and a newly designed lightweight detection head that optimizes the trade-off between accuracy and inference efficiency.
- BTDet achieves state-of-the-art performance in brain tumor detection among mainstream algorithms, realizing a 2.45% increase in $mAP@50:95$ while maintaining a super-lightweight framework with only 2.26M parameters and 6 GFLOPs. Furthermore, experiments on the LUNA16 lung nodule dataset demonstrate its strong generalizability, where BTDet attains a 5.4% improvement in $mAP@50:95$ compared to the baseline. These consistent results across different medical imaging tasks underscore the robustness and broad applicability of the proposed detector.

2. Related works

2.1. Lightweight and real-time object detectors

Lightweight object detection algorithms are designed for deployment in resource-constrained environments, where computational power and memory are limited [17]. These models aim to reduce complexity and computational load while maintaining acceptable detection accuracy, enabling real-time performance on low-power devices. Common strategies include using efficient backbone networks like MobileNet [18], ShuffleNet [19], and FasterNet [20], which leverage techniques such as depthwise separable convolutions, pointwise operations, and channel blending. Real-time object detection focuses on rapidly identifying and localizing multiple targets within incoming image or video streams [21]. Its core advantage lies in high-speed processing, making it suitable for latency-sensitive applications. Representative algorithms include YOLO [22], which uses a single-stage pipeline to predict class and location simultaneously, drastically reducing inference time. SSD [23] detects objects at multiple feature map layers for better multi-scale detection, while RetinaNet [24] incorporates Focal Loss to address class imbalance, enhancing small object detection without compromising speed.

2.2. Brain tumor medical image processing

Brain tumors exhibit considerable heterogeneity in size, shape and location, making their detection and characterization particularly challenging [25]. As a result, extensive research has been conducted to improve the accuracy and robustness of brain tumor detection in medical imaging. For instance, RCS-YOLO [26] enhances detection performance by integrating reparameterized convolution with channel shuffle and a novel cascade feature fusion strategy. Alhussainan et al. [27] evaluated the robustness of various mainstream YOLO architectures for brain tumor detection, demonstrating their effectiveness in medical scenarios. Razzaghi et al. [28] proposed a multimodal deep transfer learning framework that incorporates domain adaptation techniques to bridge the distribution gap between training and testing MRI datasets, thereby improving detection performance.

In parallel, brain tumor segmentation plays a vital role in highlighting structural and pathological alterations in medical images, which is critical for accurate diagnosis, treatment planning, disease monitoring, and clinical research. EA-DFFTU-Net [8] addresses this task by introducing consecutive feature enhancement modules within a U-Net architecture to refine segmentation accuracy. Similarly, the Multi-scale Fractal Feature Network (MFFN) [7] enhances sensitivity and classification accuracy during segmentation by leveraging fractal features at multiple scales. Furthermore, Karthik et al. [29] proposed a unified framework combining attention-augmented convolutional networks, random forest classifiers, and U-Net models to simultaneously achieve high-accuracy multi-class classification and segmentation of brain tumors in MRI images.

Our proposed BTDet distinguishes itself by integrating a task-specific architectural design tailored for brain tumor detection in MRI images. Unlike general-purpose detection architectures, BTDet is optimized end-to-end for the biomedical context, achieving superior performance in both recall and precision, setting a new benchmark for clinical-grade tumor detection systems.

2.3. Transformer-based methods for brain tumor recognition

Transformer-based methods have achieved remarkable progress in brain tumor recognition, emerging as a significant research direction in medical image analysis. Compared to conventional CNN models, Transformers effectively capture global contextual information through self-attention mechanisms, addressing CNN's limitations in modeling long-range dependencies and identifying complex tumor boundaries.

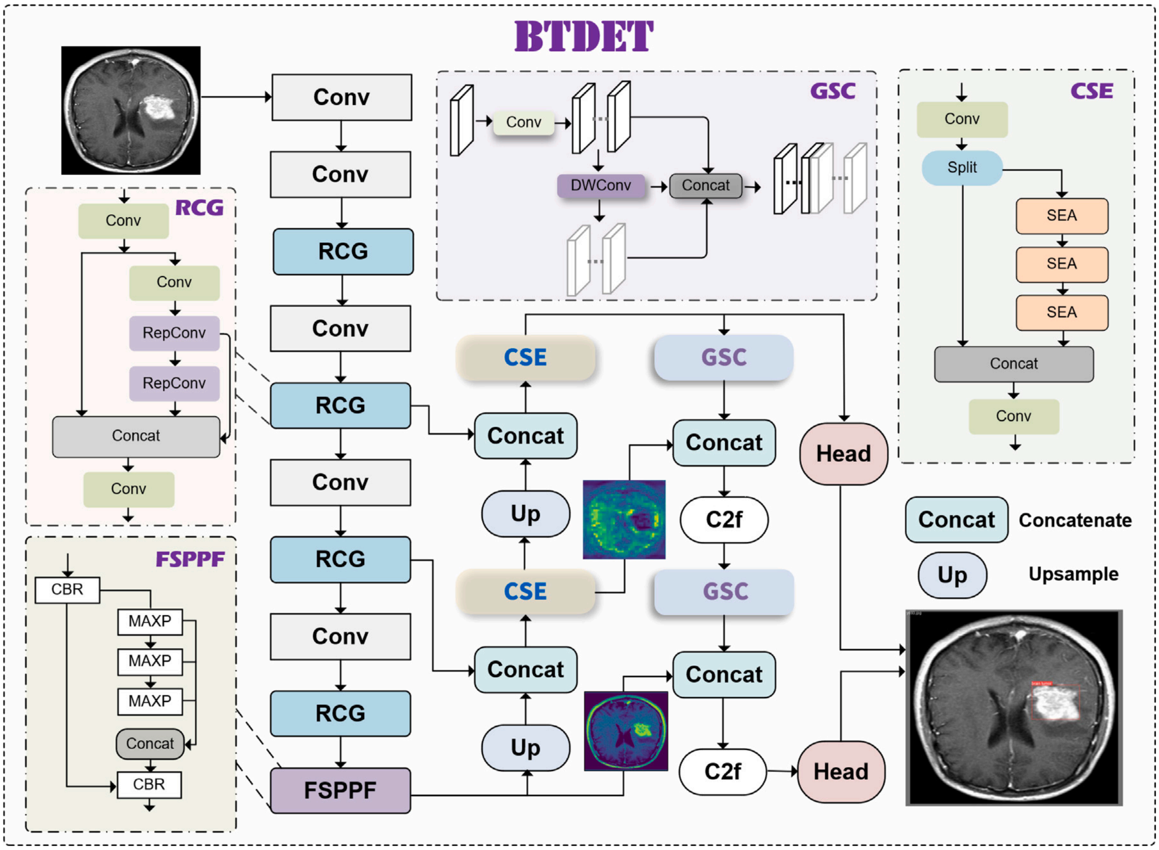

Fig. 1. Overview architecture of our proposed BTDet networks. RCG and FSPPF serve for high-efficiency feature exploration. CSE and GSC collaborate together for improving multi-scale feature fusion. Two lightweight detection heads for final brain tumor detection.

TransBTS [30] integrates CNN's local feature extraction with Transformer's global modeling, excelling in feature fusion from multimodal MRI data. UNETR [31] employs a pure Transformer encoder for end-to-end segmentation, eliminating manual feature engineering. Beyond segmentation, recent efforts such as Swin-Unet [14] and TransMed [32] have adapted hierarchical transformers or hybrid architectures to better model multi-scale tumor structures. However, a major challenge remains: transformers typically require large amounts of annotated data and high computational resources, which are not always feasible in medical settings. Furthermore, lightweight Transformer architectures, and transfer learning strategies have improved model generalizability and clinical applicability, highlighting Transformers' promising potential for automated brain tumor analysis. Compared to Transformer-based models, BTDet achieves state-of-the-art detection accuracy with a significantly more compact fully convolutional network architecture while maintaining excellent model scalability, thereby meeting the requirements for real-time brain tumor detection.

3. Methods

3.1. Overview of BTDet network

In this work, we employ the leading one-stage object detector YOLOv8, as our baseline for its optimal detection accuracy and inference speed. The YOLOv8 algorithm features five model categories: N, S, M, L, and X. We choose the smallest model, YOLOv8-N, for its satisfactory parameters and competitive results. The overall architecture of BTDet is depicted in Fig. 1 and consists of several key components: the basic convolutional layer, RCG blocks and the FSPPF module, which collectively serve as the backbone for efficient feature extraction. The architecture integrates CSE and GSC modules to enhance the fusion of multi-scale features. Ultimately, two lightweight detection heads are employed to execute the final classification and localization for brain tumor.

3.2. Reparameterized efficient aggregation backbone

The development of efficient and lightweight networks is crucial for achieving rapid, energy efficient and cost effective real-time image processing. These networks are particularly advantageous for deployment on resource constrained devices. Therefore, We designed RCG to realize fast and efficient feature extraction. The consecutive RCG blocks serve as the foundational backbone of BTDet, as illustrated in Fig. 3. The architecture of the RCG divides the input features into two distinct branches: the first branch (cross stage connection) facilitates the direct flow of information as an identity for concatenation, while the other (partial branch) comprises the RepGELAN modules. The RCG blocks are constructed using a parameterized paradigm [33] that enhances feature extraction while simultaneously reducing computational costs. Additionally, the architecture employs the GELAN [34] style to promote rapid gradient convergence.

The GELAN in Fig. 2 is a multi-branch lightweight architecture with flexible designed transition towards input data [34], aimed at enhancing the efficiency of object detection tasks. This architecture is founded on the basis of the Efficient Layer Aggregation Network (ELAN) [35] Fig. 2 and the Cross Stage Partial Network (CSPNet [36] 2. In comparison to ELAN, GELAN demonstrates the ability to utilize various computational blocks due to its generalization capabilities. Conversely, ELAN primarily mitigates gradient loss by minimizing the transition layer. CSPNet improves the model's learning capacity through cross-stage feature fusion, while also alleviating computational bottlenecks and reducing memory expenses. GELAN effectively strikes a balance

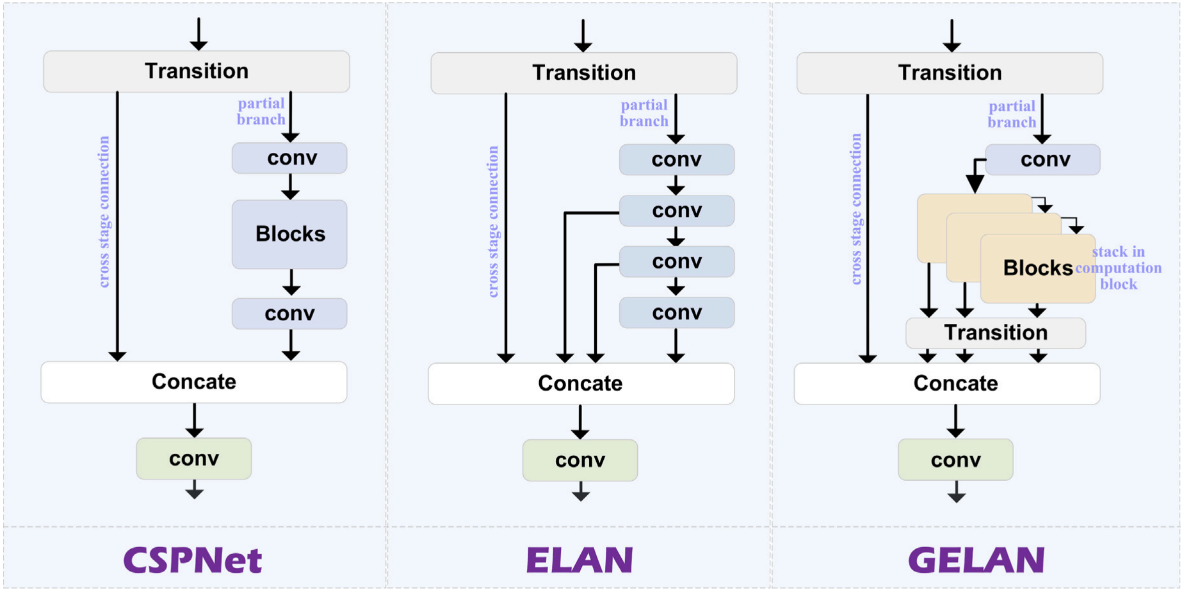

Fig. 2. Structure of CSP-style designed networks. CSPNet mitigates redundant gradient information in deep networks by partitioning feature maps and merging their gradient paths. ELAN addresses the issue of gradient degradation through a streamlined transition structure. GELAN maintains high accuracy while adhering to a lightweight design.

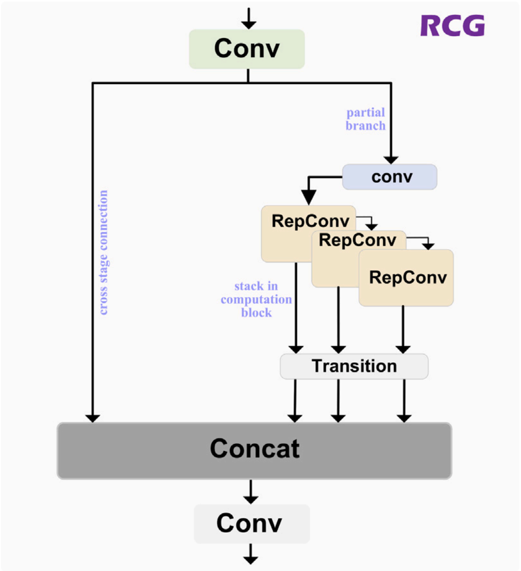

Fig. 3. Detailed structure of RCG block.

between lightweight design and high accuracy, making it particularly well-suited for real-time object detection tasks.

For an input $\mathbf{X} \in \mathbb{R}^{C\times H\times W}$, where C, H and W denotes the channel, height and width of $\mathbf{X}$, the whole process of RCG computation can be expressed by,

$$\mathbf{X}_\alpha, \mathbf{X}_\beta = Split(Conv(\mathbf{X})) \tag{1}$$

$$\begin{aligned} \mathbf{X}_\rho &= RepConv(Conv(\mathbf{X}_\beta)) \\ \mathbf{X}_\epsilon &= RepConv(\mathbf{X}_\rho) \\ \mathbf{X}_\gamma &= RepConv(\mathbf{X}_\epsilon) \\ \mathbf{X}_\eta, \mathbf{X}_\mu, \mathbf{X}_\nu &= T(\mathbf{X}_\rho, \mathbf{X}_\epsilon, \mathbf{X}_\gamma) \end{aligned} \tag{2}$$

$$\mathbf{X}_{out} = Concat[\mathbf{X}_\alpha, \mathbf{X}_\eta, \mathbf{X}_\mu, \mathbf{X}_\nu] \tag{3}$$

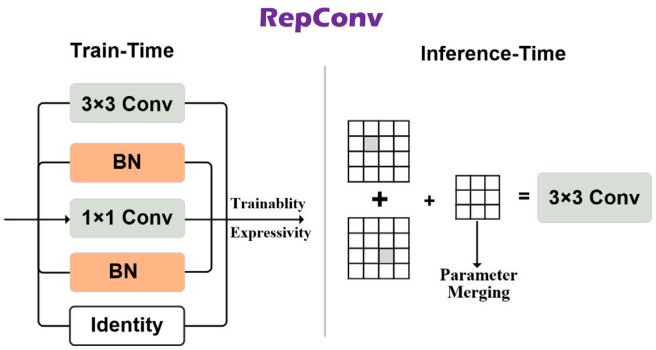

Fig. 4. Sketch of RepConv architecture.

where $Conv$ denotes the convolutional operation with kernel size 3 × 3, $Split$ divides the input data $\mathbf{X}$ into two branches equally along channel, $Concat$ means stacking the all the outputs of multi-branches together along the channel dimension. T denotes the general convolution transition.

Reparameterization Convolution (RepConv) is a technique that transforms a multi-branch convolutional architecture during training into a single-branch convolutional structure at inference time, aiming to maintain performance while improving computational efficiency. The core idea is to merge multiple convolutional kernels and branches into an equivalent single kernel through structural reparameterization, shown in Fig. 4. Below is its mathematical formulation,

Multi-Branch Structure During Training. Consider a parallel architecture (e.g., RepVGG [33]) with the following branches: **Main Branches**: A 3 × 3 convolution with kernel weights $\mathbf{W}^{(3)} \in \mathbb{R}^{C_{out}\times C_{in}\times 3\times 3}$ and bias $b^{(3)} \in \mathbb{R}_{\mathtt{out}}$. **Identity Branch**: A skip connection, equivalent to 1 × 1 convolution with kernel $\mathbf{W}^{(1)} = \mathbf{I}$ (identity matrix) and bias b (usually zero). **Residual Branch**: A 1 × 1 convolution with kernel weights $\mathbf{W}^{(1)} \in \mathbb{R}^{C_{out}\times C_{in}\times 1}$ and bias $b^{(1)} \in \mathbb{R}^{C_{out}}$. The output is the sum of all branches:

$$\begin{aligned} \mathbf{y}_{train} &= Conv(\mathbf{x}, \mathbf{W}^{(3)}, b^{(3)}) \\ &+ Conv(\mathbf{x}, \mathbf{W}^{(1)}, b^{(1)}) \\ &+ Identity(\mathbf{x}) \end{aligned} \tag{4}$$

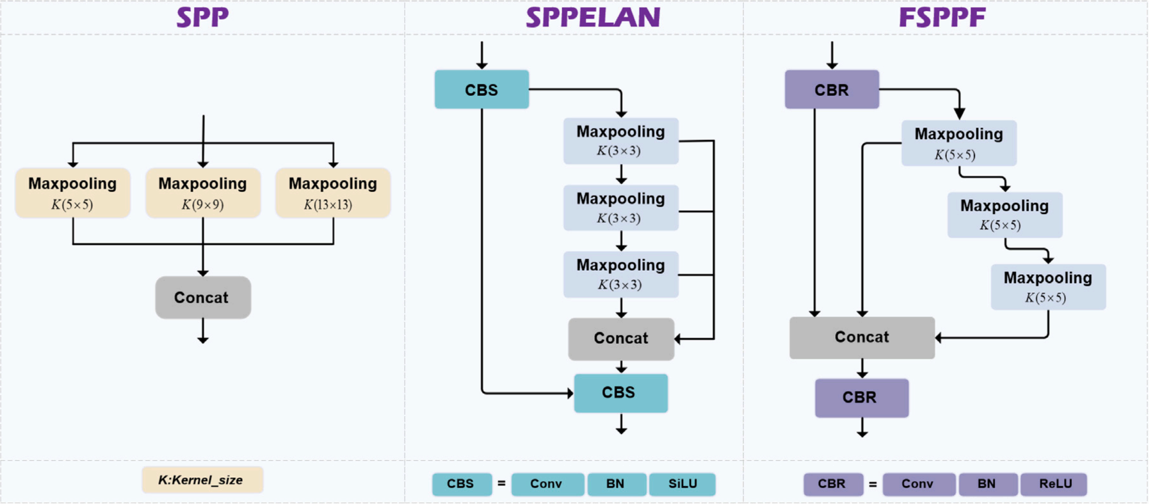

Fig. 5. Structure of semantic enhancement modules. Leveraging a spatial pyramid pooling structure, the SPP collects multi-scale information via pooling and concatenation operations. SPPELAN enhances multi-scale feature processing by integrating consecutive max-pooling blocks. FSPPF leverages stacked 5 × 5 pooling and multi-path routing for improved semantic feature discovery in deep networks.

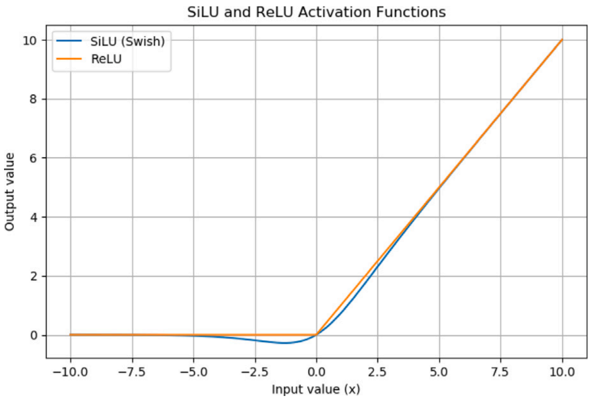

Fig. 6. Schematic diagram of SiLU and ReLU activation Function.

Equivalent Single-Branch Structure at Inference. The multi-branch structure is merged into a single 3 × 3 convolution: Zero-pad the 1 × 1 kernel $\mathbf{W}^{(1)}$ to 3 × 3, denoted as $\hat{\mathbf{W}}^{(1)}$. Represent the identity mapping as a 1 × 1 identity matrix $\mathbf{I}$, zero-paded to 3 × 3 (denoted as $\hat{\mathbf{I}}$). The **Kernel Fusion**, **Bias Fusion** and **Inference Output** are represented as follows,

$$\mathbf{W}^{merge} = \mathbf{W}^{(3)} + \hat{\mathbf{W}}^{(1)} + \hat{\mathbf{I}} \tag{5}$$

$$b^{merge} = b^{(3)} + b^{(1)} + b^{identity} \tag{6}$$

$$\mathbf{y}_{infer} = Conv(\mathbf{x}, \mathbf{W}^{merge}, b^{merge}) \tag{7}$$

Through this reparameterization, inference requires only a single convolutional computation, significantly reducing computational overhead and enhancing the model's adaptability in practical deployment scenarios.

3.3. Fast spatial pyramid pooling semantic enhancement

Semantic information is fundamental to achieving a comprehensive understanding of brain tumor region for its complex representation. It enables precise differentiation between tumor boundaries and healthy tissues, which is critical for accurate diagnosis and treatment planning. This understanding is crucial for attaining a profound level of image comprehension. To achieve sufficient semantic information feature extraction in the deep layers, we designed FSPPF block, as is shown in Fig. 5. Spatial Pyramid Pooling (SPP) [37] is a technique employed in CNN with different large maxpooling kernels to facilitate the processing of input images of varying dimensions. This method enables adaptive processing of inputs of arbitrary sizes by executing pooling across sub-regions of multiple scales. Spatial Pyramid Pooling Efficient Layer Aggregation Networks (SPPELAN) builds upon the fast gradient forward flow established by ELAN, utilizing three consecutive maxpooling blocks with kernel size of 3 × 3 to facilitate multi-scale feature processing.

Our FSPPF differs from SPPELAN with regards to kernel size and information flow path and activation function. Specifically, within the FSPPF architecture, three hierarchically arranged 5 × 5 max-pooling layers are strategically cascaded to achieve graduated receptive field expansion, facilitating the extraction of high-level semantic features while maintaining spatial coherence in tumor boundary delineation. Besides, ReLU activation function was employed to replace original SiLU counterpart inside SPPELAN. ReLU is computationally simpler and more suitable for vision tasks with single and simple features as shown in Fig. 6, while the computational complexity of SiLU is high because of the exponential operations involved, resulting inferior performance compared with ReLU.

Give the input data $\mathbf{X} \in \mathbb{R}^{C\times W\times H}$, the calculation process of FSPPF can be expressed by,

$$\begin{aligned} \mathbf{X}_{\sigma}, \mathbf{X}_{\theta} &= Split(CBR(\mathbf{X})) \\ \mathbf{X}_{\theta_1} &= MP(\mathbf{X}_{\theta}) \\ \mathbf{X}_{\theta_2} &= MP(MP(MP(\mathbf{X}_{\theta}))) \\ \mathbf{X}_{output} &= CBR(Concat[\mathbf{X}_{\sigma}, \mathbf{X}_{\theta_1}, \mathbf{X}_{\theta_2}]) \end{aligned} \tag{8}$$

where CBR denote the Convolution layer, BatchNormalization and ReLU activation function. $Split$ means the input $\mathbf{X}$ are divided into two branches equally along channel dimension, MP is the general maxpooling operation with fixed kernel size of 5 × 5.

The design of the FSPPF semantic enhancement module enables the network to perform in-depth analysis of the spatial distribution characteristics of brain tumor lesions, thereby significantly improving the model's semantic understanding of pathological regions. This multi-scale feature fusion mechanism not only optimizes computational efficiency but also generates more discriminative feature representations, providing more precise localization and grading information for clinical diagnosis.

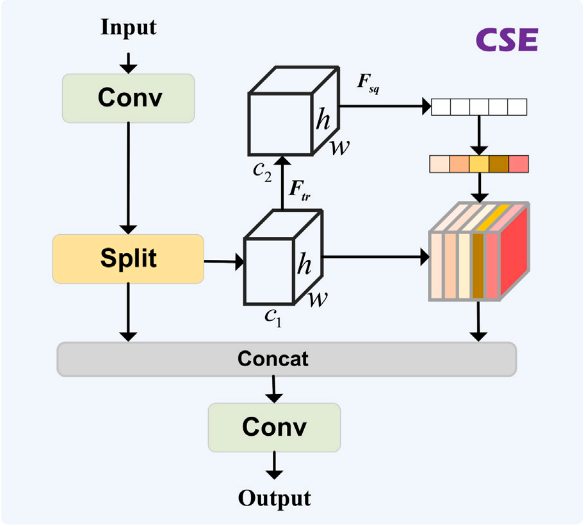

Fig. 7. Detailed architecture of the CSE integrating SEA and partial connection.

3.4. Attention and channel shuffle enhanced feature fusion

The Neck module constitutes an essential architectural component in modern object detection frameworks, serving as the feature processing bridge between convolutional feature extractors (Backbone) and task-specific decoders (Head). Its principal function involves strategic feature aggregation from different pyramidal levels, thereby preserving both high-level semantic information and low-level spatial details essential for accurate detection.

We propose an enhanced collaborative fusion framework that synergistically integrates the CSE with GSC operations for optimized multi-scale feature fusion. The framework capitalizes on the complementary characteristics of different feature scales: higher-resolution feature maps preserve richer spatial details that are crucial for small object localization, while lower-resolution features contain more discriminative semantic representations that facilitate robust object classification. This dual-path architecture enables simultaneous enhancement of both localization precision and recognition accuracy.

The attention mechanism computes adaptive spatial or channel-wise weights to prioritize diagnostically significant regions in brain tumor imaging, this selective feature enhancement improves tumor boundary delineation and pathological feature extraction, particularly for glioma metastasis differentiation. CSE enhances feature representation by combining Squeeze and Excitation Attention (SEA) [38] and partial connection, enabling it to effectively capture complex inter-channel relationships, shown in Fig. 7. SEA is a channel attention strategy that enhances the quality of representations generated by neural networks by explicitly modeling the inter-dependencies among feature channels. The fundamental concept of the SEA mechanism is to enable the model to learn how to recalibrate the responses of feature channels based on global information. This approach enhances the model's expressive capacity without a substantial increase in the number of parameters. For the input $\mathbf{X} \in \mathbb{R}^{C \times H \times W}$, the output of CSE can be formulated as,

$$\begin{aligned} \mathbf{X}_{\lambda}, \mathbf{X}_{\xi} &= Split(CBR(\mathbf{X})) \\ \mathbf{X}_{\tau} &= CSE(\mathbf{X}_{\xi}) \\ \mathbf{X}_{out} &= Concat[\mathbf{X}_{\lambda}, \mathbf{X}_{\tau}] \end{aligned} \tag{9}$$

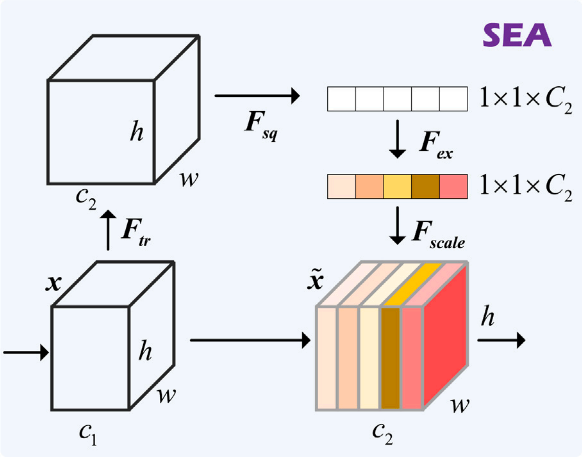

Fig. 8. SEA with dynamic channel refinement to enhance feature representation.

The SEA architecture consists of two main steps: **Squeeze** and **Excitation**, which is shown in Fig. 8. For squeeze, this step compresses the spatial information of each channel into a single value through global average pooling to obtain a global representation of each channel, the input $\mathbf{X} \in \mathbb{R}^{C \times H \times W}$, which can be represent as,

$$\mathbf{u}_c = \mathbf{v}_c * \mathbf{X} = \sum_{s=1}^{C'} \mathbf{v}_c^s * \mathbf{x}^s \tag{10}$$

$$\mathbf{z}_c = F_{sq}(\mathbf{u}_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j) \tag{11}$$

where $u_c \in \mathbb{R}^{H \times W}$ represents 2D feature map of cth channel. F_{sq} compresses the spatial dimensions $H \times W$ into channel descriptors $\mathbf{z}_c \in \mathbb{R}^C$ through Global Average Pooling (GAP).

Excitation operation employs a fully connected neural network with nonlinear activation functions to optimize the weights associated with each channel. Initially, the weights are downscaled through a fully connected layer, followed by the application of the ReLU activation function. Subsequently, the weights are upscaled by another fully connected layer, culminating in the extraction of the final weights via a sigmoid function, which can be defined as,

$$\mathbf{s} = F_{ex}(\mathbf{z}_c, \mathbf{W}) = \sigma(\mathbf{W}_2 \delta(W_1 \mathbf{z}_c)) \tag{12}$$

$$\mathbf{X}_{output} = F_{scale}(\mathbf{u}, \mathbf{s}_c) = \mathbf{s}_c \mathbf{u}_c \tag{13}$$

$\mathbf{F}_{ex}$ learns the channel dependencies by connection layers. F_{scale} adjusts the channel number depended on the hyper-parameter r. σ and δ denote sigmoid function and ReLU activation function.

The structure of GSC block is shown in Fig. 9, which aims to enhance the efficiency and accuracy of neural networks, particularly in tasks for small object detection [39]. GSC attains more efficient feature extraction by integrating the operations of Standard Convolution (SC) and Depthwise Separable Convolution (DWConv) [40], while also incorporating a feature shuffling process for enhancing the representation of the network by changing the ordering of the feature map channels.

The DWConv shown in Fig. 10, was represents an efficient convolution operation utilized in computer vision, which decomposes the conventional convolution process into two distinct operations: depthwise convolution and pointwise convolution. This decomposition substantially decreases both the computational load and the number of parameters within the model.

DWConv offers significant advantages in computer vision and medical image analysis and reduced parameter count enhances robustness on small medical datasets, depthwise convolution focuses on spatial correlations (e.g., tumor boundaries), while pointwise convolution learns inter-channel semantic relationships (e.g., multi-modal MRI

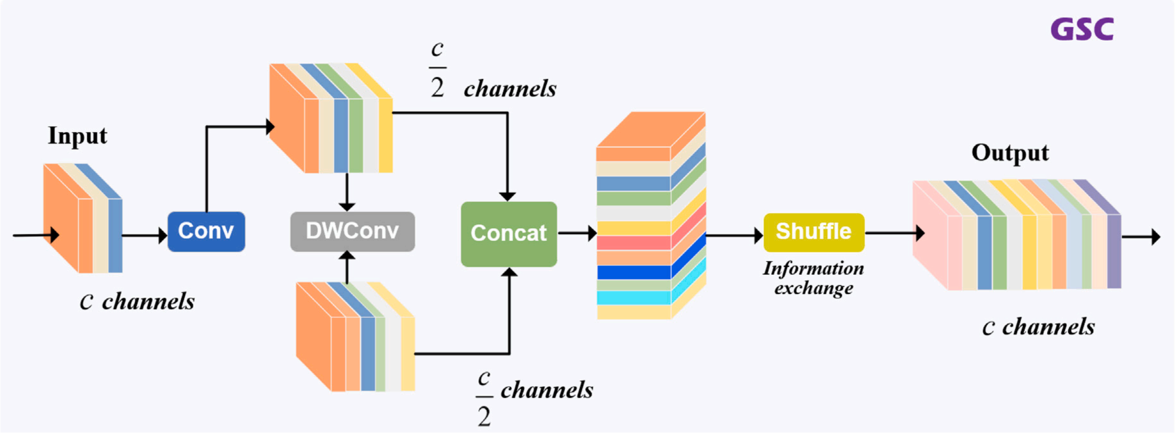

Fig. 9. The GSC architecture is a hybrid design that fuses standard convolution, DWConv, and channel shuffle operations for highly efficient feature extraction.

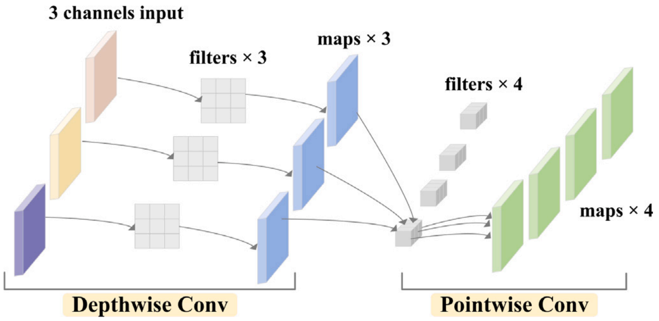

Fig. 10. DWConv consists of two sequential steps: depthwise convolution and pointwise convolution.

fusion). Additionally, DwConv can significantly reduce the computational parameters of a model, thereby further enhancing its lightweight characteristics.

The total parameters of a standard convolution can be expressed,

$$\mathcal{P}_{std} = K \times K \times C_{in} \times C_{out} \tag{14}$$

where K is the kernel size of convolution usually with 3 × 3, C_{in} and C_{out} are the input channel and out channel separately.

The parameters of DWConv can be divided into two steps,

(1) depthwise convolution:

$$\mathcal{P}_{dw} = K \times K \times C_{in} \tag{15}$$

(2) pointwise convolution:

$$\mathcal{P}_{pw} = 1 \times 1 \times C_{in} \times C_{out} \tag{16}$$

Total parameters:

$$\begin{aligned} \mathcal{P}_{DWConv} &= \mathcal{P}_{dw} + \mathcal{P}_{pw} \\ &= K^2 \times C_{in} + C_{in}C_{out} \end{aligned} \tag{17}$$

The parameter reduction ratio can be expressed,

$$\eta = \frac{\mathcal{P}_{DWConv}}{\mathcal{P}_{std}} = \frac{1}{C_{out}} + \frac{1}{K^2} \tag{18}$$

According to [39] suggested, we have placed the CSE blocks to position at the early stage of fusion layers for better results, while the effective GSC block can be embedded at the head's entrance. That is because that shallow networks are saturated with low-level semantic information, rendering the fusion function of attention modules largely unnecessary.

The calculation of GSC can be defined as,

$$\begin{aligned} \mathbf{X}_{\varsigma}, \mathbf{X}_{\varrho} &= Conv(\mathbf{X}) \\ \mathbf{X}_{\phi} &= DWConv(\mathbf{X}_{\varrho}) \\ \mathbf{X}_{output} &= F_{Shuffle}(Concat[\mathbf{X}_{\varsigma}, \mathbf{X}_{\varrho}]) \end{aligned} \tag{19}$$

where $DWConv$ means the depthwise separable convolution and $F_{shuffle}$ denotes the channel shuffle operations for facilitating information interaction.

The combination CSE and GSC blocks are devoted to ameliorating the information flow at the bottom of feature pyramid. This process not only shortens the transmission path of information fusion, but also produces fine-grained target patterns for fusing stage network, increasing the feature pyramid architecture's detection capacity and generating complementary features knowledge for final detection. Meanwhile, they also collaborate to facilitate the multi-scale feature fusion with local and global contextual information, thereby improving the performance of brain tumor detection.

4. Experiments

4.1. Dataset and settings

To evaluate the proposed BTDet, we employed brain tumor detection Br35H [41], this dataset for brain tumor detection comprises 3000 annotated MRI slices (T1/T2-weighted) with balanced binary classification (1500 normal vs. 1500 tumor cases), collected from diverse clinical sources. Besides, the Br35H dataset features clinically representative tumor distributions (frontal/temporal lobes, cerebellum) with early-stage lesions ($\geqslant$ 3 mm), providing uniformly formatted 256 × 256 resolution MRI slices (T1/T2-weighted) that exhibit mild artifacts while presenting significant diagnostic challenges due to intra-class morphological/intensity heterogeneity and moderate class imbalance (60% high-grade gliomas). The dataset's standardized format supports rapid prototyping but lacks pixel-level annotations and multi-modal sequences, suggesting complementary use with BraTS for advanced studies. For brain tumor detection task, we employ 500 images for training BTDet and 201 images for validation according the original project roles.

We also tested BTDet on the LUNA16 [42], which is a medical imaging dataset dedicated to lung nodule detection. The LUNA16 dataset is selected from the larger LIDC-IDRI [43] dataset and contains 888 low-dose CT scans of the lungs, as well as annotations by four radiologists of 1186 lung nodules with nodule diameters ranging from 3 to 30 millimeters. We split the dataset with ratio of 8:2 for training and testing, with 948 and 238 respectively to verify the effectiveness on lung nodule detection task.

Table 1
Hyper parameters and configurations of BTDet training.

Training Hyperparameters			
Parameter	Value	Parameter	Value
CPU	i7-13700KF	amp	False
GPU	Nvidia RTX 4090	works	8
cuda	11.7	optimizer	SGD
epochs	150	momentum	0.937
framework	pytorch 2.1	learning_rate	0.01
image_size	640	weight_decay	5e−4
batch_size	16	warmup_epochs	3

The detailed implementation information and hyperparameters for training BTDet is shown in Table 1. Data augmentation was systematically employed during network training to enhance model generalizability through geometric transformations and intensity variations including Mosaic Augmentation, Mixup Augmentation, Random Perspective and Hue-Saturation-Value Color-Space Augmentation, particularly. By introducing diverse image transformation, it mitigates model overfitting to specific training features while improving resilience to clinical image variations, significantly boosting detection accuracy for small tumors.

4.2. Evaluation metrics

We employ five criteria to assess and compare the detection performance of our proposed BTDet: Precision Rate (P), Recall Rate (R), mean Average Precision (mAP) at IoU thresholds from 0.50 to 0.95, number of Parameters (params), Floating Point Operations (FLOPs), and Frames Per Second (FPS). These metrics provide a comprehensive evaluation of the performance of BTDet in brain tumor and lung nodule detection task.

$$P = \frac{TP}{TP + FP} \tag{20}$$

$$R = \frac{TP}{TP + FN} \tag{21}$$

$$AP_i = \int_0^1 P_i(R_i)dR_i \tag{22}$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{23}$$

A threshold plays a crucial role in evaluating the accuracy of detection systems. Specifically, the True Positive (TP), False Positive (FP), and False Negative (FN) bounding box samples are essential metrics. Average Precision (AP) quantifies the area under the Precision-Recall (P-R) curve, indicating model performance. mAP aggregates the average precision across all categories, providing a comprehensive assessment of the detection framework.

4.3. Experimental analysis

Table 2 lists the overall comparison of our proposed BTDet with YOLO state-of-art series algorithms on the Br35H dataset. It is apparent that YOLOv5-N realizes the detection 0.684 at strict detection evaluation metric $mAP@50:95$ and 227 inference speed with only 1.7M parameters and 4.1 GFLOPs model complexity. As for larger model scale for YOLOv5-S and YOLOv5-N, they achieve detection accuracy increase at $mAP@50:95$ 0.686 and 0.703, while hold more parameters and model complexity and the model running speed gradually descends. BTDet obtains detection improvement at $mAP@50:95$ by 8.19% and 10.09% compared with YOLOv6-S and YOLOv7. Compared with baseline model YOLOv8-N, our BTDet also boosts detection performance by 2.45% at $mAP@50:95$ and also accomplishes the overall improvement at $Precision$, $Recall$ and $mAP@50$ evaluation metrics, with only 2.26M model parameters and 178 FPS inference speed, indicating the superior characteristic of BTDet in Brain tumor task. YOLOv9-C realizes the 0.73 at $mAP@50:95$, 3.15% detection accuracy lower that BTDet, with over 200 GFLOPs model complexity and 94 FPS inference speed, implying this kind of detector is incompetent when deployed in real applications. Compared with latest released YOLO detector YOLOv10, our BTDet also exhibits leading detection performance with regard in most evaluation standards. It is noteworthy that the smaller model size YOLOv10-N maintains the best detection accuracy at $mAP@50$ and $mAP@50:95$ compared with its larger version YOLOv10-S and YOLOv10M, the reason may come from that larger models suffers from complicated architecture design which is not efficient for simple texture or feature patterns for MRI brain tumor images. Compared with latest leading YOLO variants YOLO11, YOLOv12 and YOLOv13, under comparable model sizes, BTDet still exhibits significantly superior performance across different evaluation metrics. BTDet accomplishes 0.966 and 0.753 detection accuracy at $mAP@50$ and $mAP@50:95$ and 178 FPS inference speed with only 2.26M parameters and 6 GFLOPs model complexity, exhibiting high detection performance trade-off compared with the mainstream YOLO series algorithms, demonstrating that BTDet is competent to operate fast and precise brain tumor detection.

Table 3 lists detection results of BTDet with current mainstream object detectors. The representative two-stage methods Faster RCNN and Mask RCNN reach the detection accuracy $mAP@50:95$ 0.586 and 0.584 respectively, with more than 40 million parameters, indicating this kind of detector with heavy model size and not efficient for this medical brain tumor image detection. For classic one-stage detectors, SSD300 and SSD512 achieve the $mAP@50:95$ 0.647 and 0.653, they also hold numerous parameters with high complexity, not suitable for edge devices deployment. Experiments have been also conducted to compare with latest research transformer-based detector RT-DETR, which achieves 0.703 at $mAP@50:95$, there is still larger accuracy gap than BTDet, this kind of result may stems from that transformer-based method needs to benefit from large scale dataset and Br35H used in this task is not suitable for RT-DETR. It is notable that BTDet achieves 0.753 at the overall detection evaluation metric $mAP@50:95$, realizing excellent detection performance and lightweight design with only 2.26M parameters and 6 GFLOPs model complexity.

Experiments have been conducted to verify the choice of light detection heads. The results are shown in Table 4. For the baseline model, it employs 3 heads to realize brain tumor localization and classification, whose scales from larger to small are 80 × 80, 40 × 40 and 20 × 20, reaching 0.735 detection accuracy at $mAP@50:95$ with 3M parameters and 207 FPS running speed. First, we have made a attempt on 4 heads for detection by adding a larger heads DH2 160 × 160 on the base 3 heads. Then, we found the detection accuracy at $mAP@50:95$ and FPS descends dramatically, which imply that larger scale head may become a burden for final detection. Similarly, by inserting smaller size head DH6 20 × 20 also earned unpleasant results both on accuracy and inference stage. Then, we have tried several combination 2 heads for detection. Specifically, by utilizing DH3 80 × 80 and DH5 20 × 20 realized the best detection accuracy at $mAP@50:95$ 0.736 with 214 FPS and relatively lower model scale, compared with other choices such as DH3+DH4, DH4+DH5 and DH5+DH6, accomplishing satisfying trade-off between detection performance and inference speed. This experimental study demonstrates the importance choosing the proper detection head when dealing with the MRI medical images.

Table 5 presents experimental results of the FSPPF module with SPP-style methods for semantic exploration in the deep layers of backbone network. In advanced semantic understanding, it is becoming increasingly important to understand the causal relationships behind image content, which helps to improve the interpretive and generalization capabilities of the model. It can be seen that multi-maxpooling based SPP achieves 0.732 $mAP@50:95$ with only 7.4 GFLOPs model complexity, which is not efficient for lightweight paradigm detection.

Table 2
Overall comparison with the YOLO series algorithms on Br35H.

Methods	$Precision$	$Recall$	$mAP@50$	$mAP@50{:}95$	$Params.(M)$	$FLOPs(G)$	FPS
YOLOv5-N [17]	0.917	0.891	0.925	0.684	1.7	4.1	227
YOLOv5-S [17]	0.936	0.891	0.939	0.686	7	15.8	213
YOLOv5-M [17]	0.912	0.905	0.94	0.703	20.85	47.9	185
YOLOv6-N [44]	0.764	0.777	0.94	0.72	4.62	11.3	179
YOLOv6-S [44]	0.73	0.768	0.924	0.696	18.5	45.2	157
YOLOv7_Tiny [45]	0.934	0.91	0.942	0.678	6.01	13	156
YOLOv7 [45]	0.953	0.9	0.947	0.684	36.48	103.2	72
YOLOv8-N [46]	0.917	0.933	0.949	0.735	3	8.1	207
YOLOv8-S [46]	0.936	0.905	0.951	0.735	11.12	28.2	184
YOLOv9-C [34]	0.901	0.915	0.945	0.73	50.7	236.6	94
YOLOv10-N [47]	0.93	0.863	0.926	0.707	2.69	8.2	199
YOLOv10-S [47]	0.879	0.861	0.921	0.697	8	24.4	195
YOLOv10-M [47]	0.91	0.841	0.917	0.682	16.45	63.4	170
YOLO11-N [48]	0.916	0.925	0.944	0.917	2.58	6.3	183
YOLO11-S [48]	0.925	0.925	0.95	0.721	9.4	21.3	171
YOLOv12-N [49]	0.954	0.925	0.951	0.73	2.55	6.3	133
YOLOv12-S [49]	0.911	0.91	0.944	0.73	9.23	21.2	123
YOLOv13-N [50]	0.913	0.905	0.938	0.728	2.4	6.2	107
YOLOv13-S [50]	0.956	0.886	0.951	0.735	9	20.7	106
BTDet	0.945	**0.95**	**0.966**	**0.753**	2.26	6	178

Table 3
Experimental results with the mainstream algorithms on Br35H dataset.

Methods	$Precision$	$Recall$	$mAP@50$	$mAP@50{:}95$	$Params.(M)$	$FLOPs(G)$	FPS
Faster RCNN [51]	0.816	0.947	0.93	0.586	41.52	91.4	115
Mask RCNN [52]	0.747	0.923	0.928	0.584	41.17	144.5	113
SSD300 [23]	0.442	0.455	0.893	0.647	34.3	154	268
SSD512 [23]	0.44	0.458	0.894	0.653	35	154.4	161
RetinaNet [24]	0.426	0.47	0.938	0.631	37.74	95.6	116
FCOS [53]	0.28	0.342	0.713	0.307	31.83	206.51	79.4
RT-DETR [54]	0.858	0.871	0.919	0.703	31.98	103.4	61
BTDet	**0.945**	**0.95**	**0.966**	**0.753**	**2.26**	**6**	178

Table 4
Experimental study of choices of detection heads. DHα denotes the number of detection heads. DH3: the design of baseline model which has 3 heads for final detection.

Methods	$Precision$	$Recall$	$mAP@50$	$mAP@50{:}95$	$Params.(M)$	$FLOPs(G)$	FPS
DH3	0.917	0.933	0.949	0.735	3	8.1	207
+ DH2	0.953	0.9	0.955	0.729	2.92	12.2	166
+ DH6	0.934	0.915	0.951	0.732	4.78	8.1	203
DH3 + DH4	0.935	0.876	0.939	0.725	1.99	7.3	220
DH4 + DH5	0.943	0.899	0.935	0.729	3.29	6.9	196
DH5 + DH6	0.926	0.933	0.951	0.734	5.59	6.1	264
DH3 + DH5	0.934	0.91	**0.956**	**0.736**	2.78	7.4	214

Table 5
Comparison of semantic feature extraction block with proposed FSPPF.

Methods	$Precision$	$Recall$	$mAP@50$	$mAP@50{:}95$	$Params.(M)$	$FLOPs(G)$	FPS
SPPF [46]	0.917	0.933	0.949	0.735	3	8.1	207
SPPELAN [35]	0.928	0.898	0.955	0.731	3.27	7.8	229
SPPCSPC [36]	0.939	0.935	0.957	0.735	4.39	8.7	214
RFB [55]	0.946	0.896	0.949	0.728	2.94	7.5	197
ASPP [56]	0.889	0.917	0.946	0.739	4.84	9	217
SPP [37]	0.941	0.925	0.955	0.732	2.78	7.4	225
FSPPF	0.945	0.926	**0.957**	**0.74**	**2.78**	**7.4**	221

Atrous Spatial Pyramid Pooling (ASPP) is a deep learning technique for extracting multi-scale features, commonly used in semantic segmentation tasks. It contains components such as 1×1 convolution, dilated convolution and pyramidal maxpooling branches, where features with different receptive fields are obtained by various expansion rates of convolution. Notably, ASPP achieves the best detection accuracy 0.739 at $mAP@50{:}95$ whereas its parameters and model complexity became a burden for model deployment. Our FSPPF employs larger maxpooling size 5×5 than SPPELAN 3×3 and also concatenated all the branches output before next $ConvBNReLU$ process, whichi shorten the information flows to a certain extent. FSPPF realizes the 0.74 detection accuracy compared with most semantic information methods, and also offers 221 FPS running speed with the lowest model complexity 7.4 GFLOPs, which meet our lightweight and effective quality for building backbone.

Table 6 shows the experimental results of mainstream lightweight architecture with combination of RCG blocks and FSPPF. It is known

Table 6
Experimental results of mainstream lightweight networks with RCG and FSPPF.

Methods	$Precision$	$Recall$	$mAP@50$	$mAP@50:95$	$Params.(M)$	$FLOPs(G)$	FPS
YOLOv8-N [46]	0.917	0.933	0.949	0.735	3	8.1	207
GhostNet [57]	0.952	0.896	0.892	0.736	1.7	6.1	227
MobileViT [58]	0.907	0.927	0.954	0.719	1.18	5.3	106
MobileNetv3 [59]	0.958	0.905	0.943	0.724	2.35	5.7	209
ShuffleNetv2 [60]	0.94	0.886	0.942	0.707	1.7	5	283
RCG+FSPPF	0.934	**0.94**	**0.946**	**0.747**	2.34	6.1	172

Table 7
Ablation study of the improved modules of BTDet on Br35H.

Scheme	Methods	$mAP@50$	$mAP@75$	$mAP@50:95$	$Params.(M)$	$FLOPs(G)$
A	baseline	0.949	0.9	0.735	3	8.1
B	A + LightHead	0.956	0.911	0.736	2.78	7.4
C	B + FSPPF	0.957	0.9	0.74	2.78	7.4
D	C + RCG	0.946	0.905	0.747	2.34	6.1
E	D + CSE + GSC	0.966	0.911	0.753	2.26	6

that lightweight architectures is particularly important for resource-constrained environments such as mobile platforms, embedded systems, and Things of Net (IoT) devices, which could maintain a relatively high level of accuracy while significantly reduce the amount of computation and the number of parameters in the model. It is clear that by utilizing repeat RCG and conv blocks attached FSPPF module as the backbone for basic feature extraction achieves the best result 0.747 at the overall detection evaluation metric $mAP@50:95$, which outperforms most famous lightweight networks, gaining 1.5%, 3.9%, 3.2% and 5.7% improvement compared with GhostNet, MobileViT, MobileNetv3 and ShuffleNetv2. RCG strengthens feature expression of conventional C2f blocks with GELAN archetype, which combines the features of CSPNet and ELAN, through well-designed gradient paths, allowing the network to propagate and aggregate feature information from different layers more efficiently.

Ablation study was conducted to verify the effectiveness of the designed modules in BTDet as shown in Table 7. When introduced the LightHead paradigm, the parameters and model complexity reduced 7.3% and 8.6% respectively compared with baseline method with detection accuracy 0.736 at $mAP@50:95$. Based on LightHead, the mAP was increased to 0.74 with the embedding of FSPPF in the deep layers owing the its contribution for semantic feature extraction. When RCG blocks take the place of the original backbone, the $mAP@50:95$ was boosted to 0.747, showcasing the productive feature exploration of the brain tumor complicated patterns. The synergistic design of CSE and GSC enhances multi-scale feature refinement across network layers, equipping the model with superior adaptability to object size changes and greater robustness in varied environments. As a result, BTDet attains $mAP@50$ and $mAP@50:95$ scores of 0.966 and 0.753 with merely 2.26M parameters and 6 GFLOPs, underscoring its effectiveness and efficiency in brain tumor detection.

To further validate the robustness of our proposed BTDet on other medical image processing task, we have also conducted experiments on the LUNA16 dataset, which is instrumental for developing and testing novel lung nodule detection algorithms, significantly advancing research in medical image analysis. Compared with YOLO series algorithms, which is shown in Table 8, BTDet achieves leading detection accuracy 0.328 at the overall evaluation metric $mAP@50:95$, 14.7%, 20.1% and 5.5% accuracy improvement than YOLOv5-M, YOLOv6-S and baseline YOLOv8-N. Besides, BTDet also follows the lightweight style design on LUNA16 dataset, with only 2.25M parameters and 6 GFLOPs model complexity and 263 FPS inference speed when testing lung nodule MRI images. For the YOLO new version methods, YOLOv9-C realized 0.325 accuracy at $mAP@50:95$ but the parameters are already up to 50M and GFLOPs over 200M, obviously, not capable to deploy on computational resource limited devices. This experiments validates the flexibility of BTDet on lung nodule detection task with notable performance among the main family of YOLO series algorithms.

Table 9 list the comparative results of BTDet with prevalent detection algorithms. For two-stage based method, Faster RCNN and Cascade RCNN achieved 0.262 and 0.282 at $mAP@50:95$ and the inference speed FPS are all below 100, significantly behind the accuracy reached by BTDet. As to the one-stage measures, SSD300 and SSD512 also realized unpleasant results. Transformer-designed detector RT-DETR behaved 0.694 and 0.302 at $mAP@50$ and $mAP@50:95$, 3.46% and 8.6% lower than BTDet. Although Sparse RCNN realized the best detection accuracy at $mAP@50:95$, it owes more than 100M model parameters and 157 GFLOPs complexity. Our BTDet only has 2.25M parameters and 6 GFLOPs complexity, saving 97.9% and 96.2% counterpart than SparseRCNN. BTDet not only realizes outstanding detection performance, but also features in model lightweight and fast inference speed compared with mainstream detection algorithms.

Table 10 list the ablation study of bounding box regression loss function of BTDet on LUNA16 dataset. The baseline model YOLOv8-N applies CIoU for accurate regression task, realized 0.317 at $mAP@50:95$ and 260 FPS. DIoU loss function does not performs well at accuracy and inference stage. GIoU extends the traditional IoU metric to provide a more comprehensive optimization objective by taking into account the distances between the predicted and true frames as well as their sizes. The GIoU loss function is particularly valuable because it addresses the limitations of IoU in certain situations, such as when the predicted and true frames do not overlap at all, where IoU cannot provide gradient information. Following the experimental results, we adjust the IoU loss function to GIoU when tested on LUNA16 dataset, which achieved the best detection performance 0.328 at $mAP@50:95$ and 263 FPS inference speed.

4.4. Visualization analysis of BTDet on brain tumor detection

For an intuitive understanding of BTDet real detection performance, we visualize several representative validation samples of Br35H in Fig. 11. It is noticeable that brain tumor presented in the figure all display irregular shapes and different locations. As shown in the last two row in the figure, detection confidence offered in the image of BTDet are generally higher than the baseline YOLOv8-N counterpart, demonstrating BTDet could process brain tumor images with comprehensive high-level performance due the effectiveness of the model design.

To further explore the recognition ability towards small tumor regions, we conducted a comparative Grad-CAM [67] visualization analysis between BTDet and baseline methods. As shown in Fig. 12, BTDet generates precise, compact heatmaps (the second row) that closely match tumor boundaries, while baseline (the first row) methods produce diffuse activation extending into healthy tissue. Furthermore, BTDet demonstrates a stronger response to brain tumor locations in heatmaps, enabling precise lesion localization. The proposed model

Table 8
Overall comparison with the YOLO series algorithms on LUNA16 dataset.

Methods	*Precision*	*Recall*	*mAP@50*	*mAP@50:95*	*Params.(M)*	*FLOPs(G)*	*FPS*
YOLOv5-N [17]	0.711	0.559	0.588	0.25	1.76	4.1	400
YOLOv5-S [17]	0.677	0.605	0.621	0.254	7	15.8	417
YOLOv5-M [17]	0.688	0.634	0.645	0.286	20.8	47.9	270
YOLOv6-N [44]	0.739	0.629	0.626	0.274	4.63	11.34	321
YOLOv6-S [44]	0.69	0.655	0.619	0.273	18.5	45.17	298
YOLOv7_Tiny [45]	0.66	0.588	0.593	0.24	6	13	270
YOLOv7 [45]	0.599	0.504	0.537	0.244	36.48	103.2	141
YOLOv8-N [46]	0.759	0.71	0.715	0.311	3	8.1	268
YOLOv8-S [46]	0.786	0.695	0.701	0.327	11.1	28.4	254
YOLOv8-M [46]	0.723	0.691	0.701	0.327	25.7	78.7	165
YOLOv9-C [34]	0.789	0.706	0.747	0.325	50.6	236.6	99
YOLOv10-N [47]	0.713	0.655	0.674	0.303	2.69	8.2	287
YOLOv10-S [47]	0.755	0.636	0.689	0.296	8	24.4	283
YOLOv10-M [47]	0.732	0.681	0.712	0.317	16.4	63.4	209
YOLO11-N [48]	0.753	0.693	0.714	0.322	2.58	6.3	291
YOLO11-S [48]	0.753	0.718	0.708	0.322	9.41	21.3	275
YOLOv12-N [49]	0.676	0.647	0.619	0.271	2.55	6.3	190
YOLOv12-S [49]	0.685	0.676	0.671	0.3	9.23	21.2	185
YOLOv13-N [50]	0.72	0.649	0.657	0.291	2.44	6.2	131
YOLOv13-S [50]	0.752	0.676	0.698	0.306	9	20.7	133
BTDet	0.761	**0.714**	**0.718**	**0.328**	2.25	6	263

Table 9
Comparison with the mainstream algorithms on LUNA16 dataset.

Methods	*Precision*	*Recall*	*mAP@50*	*mAP@50:95*	*Params.(M)*	*FLOPs(G)*	*FPS*
Faster RCNN [51]	0.667	0.552	0.559	0.262	41.12	206.66	84
Cascade RCNN [61]	0.637	0.626	0.609	0.282	68.93	234.66	65
SSD300 [23]	0.609	0.504	0.433	0.155	23.75	34.27	199
SSD512 [23]	0.691	0.668	0.651	0.208	24.39	87.72	157
DCNv2 [62]	0.672	0.559	0.549	0.252	148.69	229.42	72
Sparse RCNN [63]	0.739	0.721	0.626	0.342	105.94	157	71
RT-DETR [54]	0.742	0.66	0.694	0.302	41.9	125.6	68
BTDet	**0.761**	0.714	**0.718**	**0.328**	**2.25**	**6**	**263**

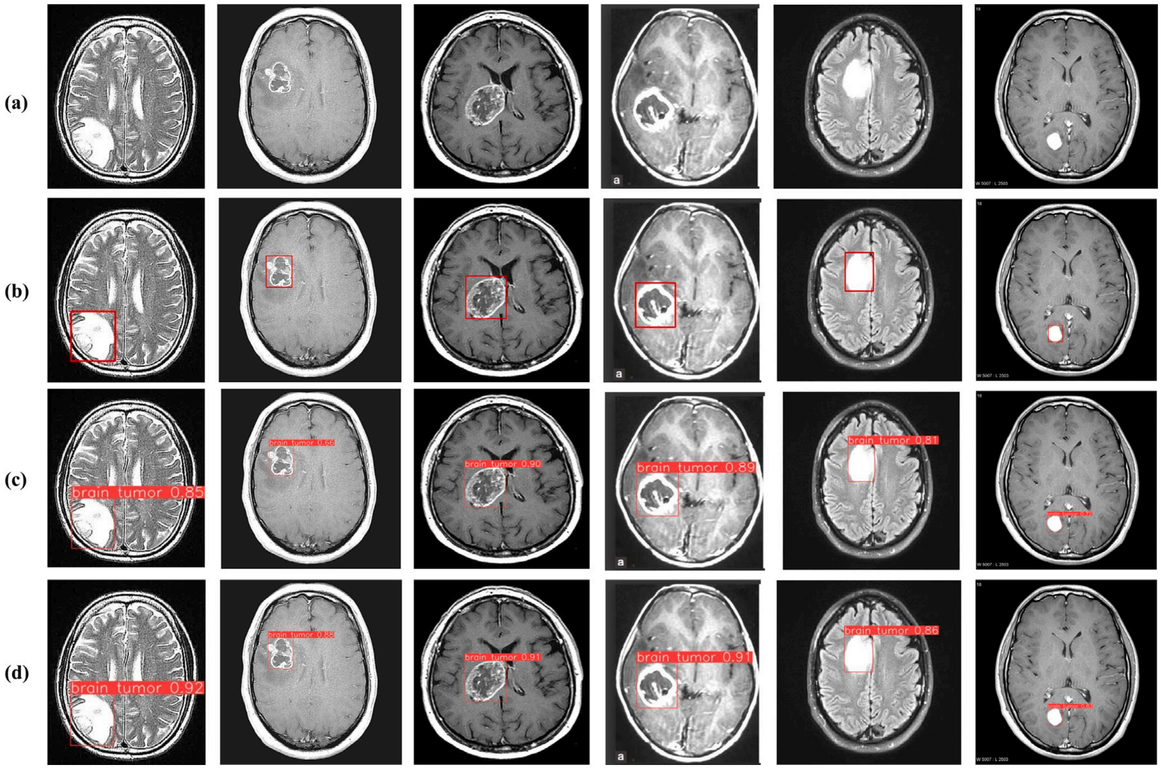

Fig. 11. Visualization of detection results of BTDet and YOLOv8n on Br35H. (a) Input Images. (b) Ground Truth. (c) Baseline. (d) BTDet.

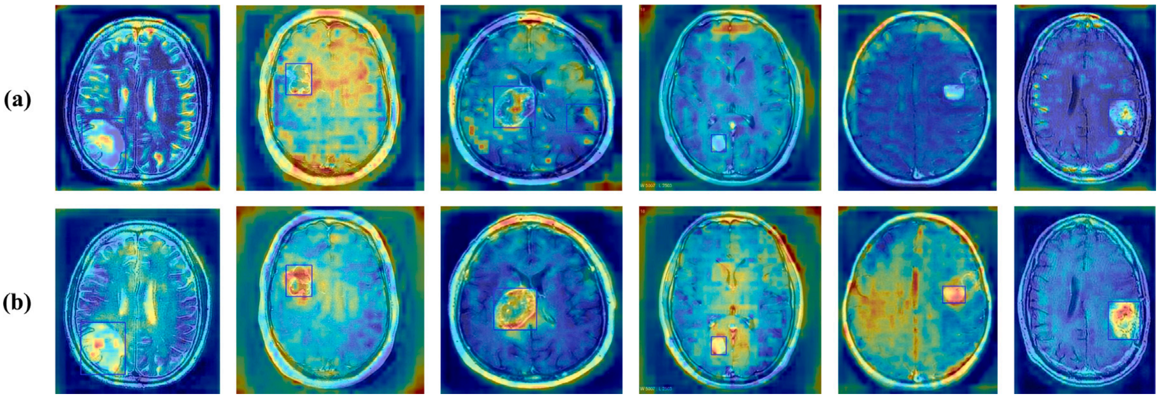

Fig. 12. Comparative analysis of heatmap by Grad-CAM between baseline method and BTDet on Br35H. (a) Baseline. (b) BTDet.

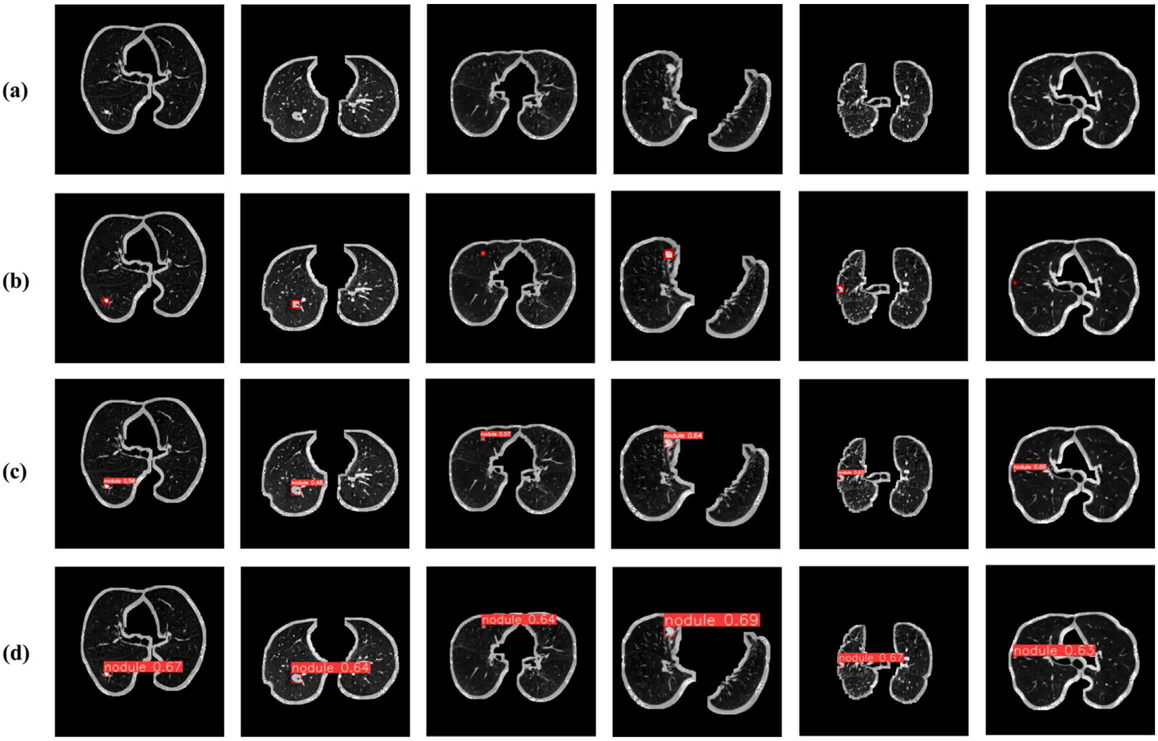

Fig. 13. Visualization of detection results of BTDet and YOLOv8n on LUNA16. (a) Input Images. (b) Ground Truth. (c) Baseline. (d) BTDet.

Table 10
Detection results of BTDet on LUNA16 under different IoU loss function settings.

Methods	*Precision*	*Recall*	*mAP@50:95*	*FPS*
CIoU [64]	0.755	0.698	0.317	260
DIoU [65]	0.733	0.727	0.313	231
GIoU [66]	0.761	0.714	0.328	263

maintains strong spatial correlation with radiologist annotations across all tumor sizes. Consequently, our architecture demonstrates significantly improved detection of smaller lesions through its multi-scale feature aggregation design, consistently localizing tumors that baseline methods miss or incorrectly fragment.

We also visualize several examples from LUNA16 dataset to verify the lung nodule detection performance of BTDet which shown in Fig. 13. From the sample images, we can understand that the percentage of lung nodules in MRI images is extremely small and surrounded by complex contextual feature information, which greatly boosts the difficulty for accurate nodule detection. Surprisingly, BTDet demonstrated excellent detection performance and was able to accurately localize and identify the position of lung nodules in the image at a high confidence level. BTDet is not only able to achieve a high level of detection performance on the brain tumor task, but also performs equally well on the lung nodule task, confirming that BTDet is robust and scalable and has great potential for application in the field of medical image processing.

5. Limitations

Despite achieving strong performance in brain tumor and lung nodule detection tasks, BTDet still has several limitations. First, the datasets used are relatively small and lack diversity, which may affect the model's generalizability across different clinical settings. Second, the detection accuracy for lesions with blurred boundaries or irregular shapes remains suboptimal. Third, BTDet has not yet been validated in real-world clinical environments, and the absence of interaction with medical professionals limits its practical applicability. Moreover, the model currently relies on 2D imaging and does not fully leverage 3D spatial information, potentially overlooking subtle lesions across slices. Finally, the structure of the detection head is fixed, making it difficult to adapt dynamically to different tasks. Future work should focus on improving data diversity, incorporating 3D modeling, conducting clinical validations, and enhancing structural flexibility to improve the practicality and adaptability of BTDet.

6. Conclusion

In this work, we present BTDet, an efficient and lightweight framework for brain tumor detection in medical images. The proposed model achieves an effective balance between detection accuracy and computational efficiency, demonstrating strong performance while maintaining low parameter counts and minimal computational overhead. BTDet integrates several design components to enhance detection capability: (1) an RCG block combined with a reparameterized GELAN backbone to facilitate gradient propagation and feature extraction; (2) an FSPPF module that enlarges the receptive field through large-kernel pooling for improved semantic representation in deeper layers; (3) a neck structure augmented with a CSE and GSC, supporting adaptive attention and efficient multi-scale fusion; and (4) two lightweight detection heads that further reduce inference cost. Extensive experiments show that BTDet achieves consistent performance gains across medical imaging tasks. On the Br35H brain tumor dataset, it attains a 2.45% improvement in $mAP@50:95$ over strong baselines. It also generalizes well to lung nodule detection on the LUNA16 dataset, where it achieves a 5.4% accuracy gain, confirming its cross-domain applicability. These results highlight BTDet as a practical and accurate detection solution that is suitable for real-world clinical scenarios.

Abbreviations

Main abbreviations are used in this manuscript:

CNNs	Convolutional Neural Networks
C2f	CSPDarknet53 to 2-Stage FPN
RCG	Reparameterized C2f GELAN
FSPPF	Fast Spatial Pyramid Pooling Fusion
SEA	Squeeze and Excitation Attention
CSE	C2f Squeeze and Excitation
GSC	General Depthwise Separable Convolution
ELAN	Efficient Layer Aggregation Network
GELAN	Generalized Efficient Layer Aggregation Network
DWConv	Depthwise Separable Convolution
P	Precision
R	Recall
mAP	mean Average Precision
FPS	Frame Per Second
FLOPs	Floating Point Operations
IoU	Intersection over Union

CRediT authorship contribution statement

Yi Li: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Huiying Xu:** Writing – review & editing, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Xinzhong Zhu:** Visualization, Validation, Supervision, Software, Resources, Project administration, Formal analysis, Data curation. **Xiao Huang:** Visualization, Validation, Methodology, Investigation, Formal analysis. **Hongbo Li:** Visualization, Validation, Software, Resources, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (62376252); Key Project of Natural Science Foundation of Zhejiang Province, China (LZ22F030003).

Data availability

Data will be made available on request.

References

[1] T. Saba, A.S. Mohamed, M. El-Affendi, J. Amin, M. Sharif, Brain tumor detection using fusion of hand crafted and deep learning features, Cogn. Syst. Res. 59 (2020) 221–230.

[2] C.P. Wild, B.W. Stewart, C. Wild, World Cancer Report 2014, World Health Organization Geneva, Switzerland, 2014.

[3] M.K. Abd-Ellah, A.I. Awad, A.A. Khalaf, H.F. Hamed, Two-phase multi-model automatic brain tumour diagnosis system from magnetic resonance images using convolutional neural networks, EURASIP J. Image Video Process. 2018 (1) (2018) 1–10.

[4] S.M. Alzahrani, Convattenmixer: Brain tumor detection and type classification using convolutional mixer with external and self-attention mechanisms, J. King Saud University-Computer Inf. Sci. 35 (10) (2023) 101810.

[5] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Commun. ACM 60 (6) (2017) 84–90.

[6] P. Smitha, G. Balaarunesh, C.S. Nath, A. Sabatini, Classification of brain tumor using deep learning at early stage, Meas.: Sensors 35 (2024) 101295.

[7] S.P. Jakhar, A. Nandal, A. Dhaka, A. Alhudhaif, K. Polat, Brain tumor detection with multi-scale fractal feature network and fractal residual learning, Appl. Soft Comput. 153 (2024) 111284.

[8] B. Jagadeesh, G.A. Kumar, Brain tumor segmentation with missing mri modalities using edge aware discriminative feature fusion based transformer u-net, Appl. Soft Comput. 161 (2024) 111709.

[9] Z. Zhou, M.M. Rahman Siddiquee, N. Tajbakhsh, J. Liang, Unet++: A nested u-net architecture for medical image segmentation, in: International Workshop on Deep Learning in Medical Image Analysis, Springer, 2018, pp. 3–11.

[10] O. Oktay, J. Schlemper, L.L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N.Y. Hammerla, B. Kainz, et al., Attention u-net: Learning where to look for the pancreas, 2018, arXiv preprint arXiv:1804.03999.

[11] Z.-L. Ni, G.-B. Bian, X.-H. Zhou, Z.-G. Hou, X.-L. Xie, C. Wang, Y.-J. Zhou, R.-Q. Li, Z. Li, Raunet: Residual attention u-net for semantic segmentation of cataract surgical instruments, in: International Conference on Neural Information Processing, Springer, 2019, pp. 139–149.

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, 2020, arXiv preprint arXiv:2010.11929.

[13] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A.L. Yuille, Y. Zhou, Transunet: Transformers make strong encoders for medical image segmentation, 2021, arXiv preprint arXiv:2102.04306.

[14] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, M. Wang, Swin-unet: Unet-like pure transformer for medical image segmentation, in: European Conference on Computer Vision, Springer, 2022, pp. 205–218.

[15] S. Mohammadi, M. Allali, Advancing brain tumor segmentation with spectral–spatial graph neural networks, Appl. Sci. 14 (8) (2024) 3424.

[16] M. Ravinder, G. Saluja, S. Allabun, M.S. Alqahtani, M. Abbas, M. Othman, B.O. Soufiene, Enhanced brain tumor classification using graph convolutional neural network architecture, Sci. Rep. 13 (1) (2023) 14938.

[17] G. Jocher, Yolov5 by ultralytics, 2020, http://dx.doi.org/10.5281/zenodo.3908559, https://github.com/ultralytics/yolov5.

[18] A.G. Howard, Mobilenets: Efficient convolutional neural networks for mobile vision applications, 2017, arXiv preprint arXiv:1704.04861.

[19] X. Zhang, X. Zhou, M. Lin, J. Sun, Shufflenet: An extremely efficient convolutional neural network for mobile devices, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6848–6856.

[20] J. Chen, S. h. Kao, H. He, W. Zhuo, S. Wen, C.-H. Lee, S.-H.G. Chan, Run, don't walk: chasing higher flops for faster neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 12021–12031.

[21] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, J. Chen, Detrs beat yolos on real-time object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 16965–16974.

[22] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, G. Ding, Yolov10: Real-time end-to-end object detection, 2024, arXiv preprint arXiv:2405.14458.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, Ssd: Single shot multibox detector, in: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, the Netherlands, October (2016) 11–14, Proceedings, Part I 14, Springer, 2016, pp. 21–37.

[24] T. Lin, Focal loss for dense object detection, 2017, arXiv preprint arXiv:1708.02002.

[25] B.M. Jebin, S.I. Shyla, K.N.S. Bel, C.J.J. Sheela, Brain tumor detection and classification using u-net and cnn with brain texture pattern analysis, Biomed. Signal Process. Control. 110 (2025) 108156.

[26] M. Kang, C.-M. Ting, F.F. Ting, R.C.-W. Phan, Rcs-yolo: A fast and high-accuracy object detector for brain tumor detection, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2023, pp. 600–610.

[27] N.F. Alhussainan, B. Ben Youssef, M.M. Ben Ismail, A deep learning approach for brain tumor firmness detection based on five different yolo versions: Yolov3–yolov7, Computation 12 (3) (2024) 44.

[28] P. Razzaghi, K. Abbasi, M. Shirazi, S. Rashidi, Multimodal brain tumor detection using multimodal deep transfer learning, Appl. Soft Comput. 129 (2022) 109631.

[29] A. Karthik, S.K. Sahoo, A. Kumar, N. Patel, P. Chinnaraj, L.P. Maguluri, A. Rajaram, et al., Unified approach for accurate brain tumor multi-classification and segmentation through fusion of advanced methodologies, Biomed. Signal Process. Control. 100 (2025) 106872.

[30] W. Wang, C. Chen, M. Ding, H. Yu, S. Zha, J. Li, Transbts: Multimodal brain tumor segmentation using transformer, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2021, pp. 109–119.

[31] A. Hatamizadeh, Y. Tang, V. Nath, D. Yang, A. Myronenko, B. Landman, H.R. Roth, D. Xu, Unetr: Transformers for 3d medical image segmentation, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 574–584.

[32] Y. Dai, Y. Gao, F. Liu, Transmed: Transformers advance multi-modal medical image classification, Diagnostics 11 (8) (2021) 1384.

[33] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, J. Sun, Repvgg: Making vgg-style convnets great again, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 13733–13742.

[34] C.-Y. Wang, I.-H. Yeh, H.-Y.M. Liao, Yolov9: Learning what you want to learn using programmable gradient information, 2024, arXiv preprint arXiv:2402.13616.

[35] C.-Y. Wang, H.-Y.M. Liao, I.-H. Yeh, Designing network design strategies through gradient path analysis, 2022, arXiv preprint arXiv:2211.04800.

[36] C.-Y. Wang, H.-Y.M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, I.-H. Yeh, Cspnet: A new backbone that can enhance learning capability of cnn, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 390–391.

[37] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (9) (2015) 1904–1916.

[38] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.

[39] H. Li, J. Li, H. Wei, Z. Liu, Z. Zhan, Q. Ren, Slim-neck by gsconv: A better design paradigm of detector architectures for autonomous vehicles, 2022, arXiv preprint arXiv:2206.02424.

[40] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.

[41] Hashira, Br35h :: brain tumor detection 2020 dataset, 2020, Kaggle, https://www.kaggle.com/datasets/ahmedhamada0/brain-tumor-detection.

[42] A.A.A. Setio, A. Traverso, T. De Bel, M.S. Berens, C. Van Den Bogaard, P. Cerello, H. Chen, Q. Dou, M.E. Fantacci, B. Geurts, et al., Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the luna16 challenge, Med. Image Anal. 42 (2017) 1–13.

[43] III S.G. Armato, G. McLennan, L. Bidaut, M.F. McNitt-Gray, C.R. Meyer, A.P. Reeves, B. Zhao, D.R. Aberle, C.I. Henschke, E.A. Hoffman, et al., The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans, Med. Phys. 38 (2) (2011) 915–931.

[44] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie, et al., Yolov6: A single-stage object detection framework for industrial applications, 2022, arXiv preprint arXiv:2209.02976.

[45] C.-Y. Wang, A. Bochkovskiy, H.-Y.M. Liao, Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 7464–7475.

[46] G. Jocher, A. Chaurasia, J. Qiu, Ultralytics YOLO, 2023, https://github.com/ultralytics/ultralytics.

[47] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, G. Ding, Yolov10: Real-time end-to-end object detection, 2024, arXiv preprint arXiv:2405.14458.

[48] R. Khanam, M. Hussain, Yolov11: An overview of the key architectural enhancements, 2024, arXiv preprint arXiv:2410.17725.

[49] Y. Tian, Q. Ye, D. Doermann, Yolov12: Attention-centric real-time object detectors, 2025, arXiv preprint arXiv:2502.12524.

[50] M. Lei, S. Li, Y. Wu, H. Hu, Y. Zhou, X. Zheng, G. Ding, S. Du, Z. Wu, Y. Gao, Yolov13: Real-time object detection with hypergraph-enhanced adaptive visual perception, 2025, arXiv preprint arXiv:2506.17733.

[51] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, in: Advances in Neural Information Processing Systems, vol. 28, 2015.

[52] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2961–2969.

[53] Z. Tian, C. Shen, H. Chen, T. He, Fcos: fully convolutional one-stage object detection, 2019, arxiv, arXiv preprint arXiv:1904.01355, 1904.

[54] W. Lv, S. Xu, Y. Zhao, G. Wang, J. Wei, C. Cui, Y. Du, Q. Dang, Y. Liu, Detrs beat yolos on real-time object detection, 2023, arXiv preprint arXiv:2304.08069.

[55] S. Liu, D. Huang, et al., Receptive field block net for accurate and fast object detection, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 385–400.

[56] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2017) 834–848.

[57] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, C. Xu, Ghostnet: More features from cheap operations, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 1580–1589.

[58] S. Mehta, M. Rastegari, Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer, 2021, arXiv preprint arXiv:2110.02178.

[59] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, et al., Searching for mobilenetv3, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1314–1324.

[60] N. Ma, X. Zhang, H.-T. Zheng, J. Sun, Shufflenet v2: Practical guidelines for efficient cnn architecture design, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 116–131.

[61] Z. Cai, N. Vasconcelos, Cascade r-cnn: Delving into high quality object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6154–6162.

[62] X. Zhu, H. Hu, S. Lin, J. Dai, Deformable convnets v2: More deformable, better results, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 9308–9316.

[63] P. Sun, R. Zhang, Y. Jiang, T. Kong, C. Xu, W. Zhan, M. Tomizuka, L. Li, Z. Yuan, C. Wang, et al., Sparse r-cnn: End-to-end object detection with learnable proposals, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 14454–14463.

[64] Z. Zheng, P. Wang, D. Ren, W. Liu, R. Ye, Q. Hu, W. Zuo, Enhancing geometric factors in model learning and inference for object detection and instance segmentation, IEEE Trans. Cybern. 52 (8) (2021) 8574–8586.

[65] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, D. Ren, Distance-iou loss: Faster and better learning for bounding box regression, in: Proceedings of the AAAI conference on artificial intelligence, vol. 34, 2020, pp. 12993–13000.

[66] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, S. Savarese, Generalized intersection over union: A metric and a loss for bounding box regression, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 658–666.

[67] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.