Journal of Electronic Research and Application, 2025, Volume 9, Issue 5
http://ojs.bbwpublisher.com/index.php/JERA
ISSN Online: 2208-3510
ISSN Print: 2208-3502

Lightweight Multi-Object Detection for Construction Sites Based on YOLO-World

Bing Chen*

School of Computer Science and Technology, Taiyuan Normal University, Taiyuan 030619, China

**Author to whom correspondence should be addressed.*

Abstract: Addressing the current issues in construction site detection algorithms—such as missed detections, false positives, and high model complexity—caused by occlusions and scale variations in dense environments. This paper proposes a lightweight multi-object detection model for construction sites based on YOLO-World, named the LCS-YOLO model, to achieve a balance between detection efficiency and accuracy. We propose the RGNet (Re-parameterization GhostNet) module, which integrates re-parameterized convolutions and a multi-branch architecture. This approach addresses the issue of information redundancy in intermediate feature maps while enhancing feature extraction and gradient flow capabilities. Combined with the adaptive downsampling module ADown (Adaptive Downsampling), it better captures image features and achieves spatial compression, reducing model complexity while enhancing interaction between images and text. Experiments demonstrate that the LCS-YOLO model outperforms other comparison models in overall performance, achieving a balance between accuracy and efficiency.

Keywords: Dense; LCS-YOLO; RGNet; ADown; Re-parameterized

Online publication: October 17, 2025

1. Introduction

In the construction industry, to ensure worker safety, workers are required to wear basic protective gear such as hard hats and reflective vests. Therefore, ensuring workers wear safety equipment is a key aspect of construction site safety management. If workers lack sufficient safety awareness, manual inspections of protective gear usage become necessary. However, this approach is not only inefficient but also incurs extremely high labor costs. Early methods of using sensors to detect whether workers were wearing safety equipment were limited by high costs and operational inconvenience [1–3].

With the advancement of image processing technologies, particularly the progress in deep learning and convolutional neural networks [4–6], construction site detection has experienced rapid development. The YOLO [7] series, as a representative of object detection, has undergone continuous iterations and integrated new technologies

such as FPN [8], PANet [9], and RepVGG [10] to optimize detection speed and accuracy. Additionally, Yi *et al.* [11] employed YOLOv5 as the baseline model and added an extra detection head to capture small object features, thereby enhancing the recognition capability for small objects. However, this also resulted in an increase in the number of model parameters.

Although the aforementioned research has achieved promising results in detection, for construction site safety equipment inspection, reducing redundant information to enhance feature extraction and minimizing model complexity remain critical factors in detection. This paper proposes the RGNet module, combined with the adaptive downsampling module ADown [12], to introduce the LCS-YOLO model. This model effectively reduces redundant information while enhancing feature extraction and gradient flow capabilities, achieving a balance between accuracy and efficiency.

In summary, the main contributions of the LCS-YOLO model proposed in this paper are as follows:

(1) The RGNet module proposed for backbone networks integrates re-parameterized convolutions and multi-branch structures. This approach addresses information redundancy in intermediate feature mappings within the C2F module while enhancing feature extraction and gradient flow capabilities.

(2) The introduction of the ADown module replaces the original downsampling module, enabling more effective extraction of image features and spatial compression. This approach reduces model complexity while enhancing interaction between images and text.

2. Related work

Dense pedestrian detection typically involves a trade-off between feature extraction requirements and computational resource constraints, particularly in edge devices or real-time applications where model compactness and efficiency are paramount. We found that YOLO-World [13] tends to introduce significant information redundancy when processing image features via C2F. The original downsampling primarily employed convolution operations with a stride of 2, which tended to filter out fine-grained features. This approach may limit generalization capabilities in complex scenarios such as multi-scale objects. Additionally, the model's complexity was relatively high due to the introduction of visual-language modeling, making deployment on construction sites more challenging.

In the LCS-YOLO model, we propose the RGNet module, which integrates re-parameterized convolutions [14] and a multi-branch architecture. This addresses the information redundancy present in the intermediate feature maps of the C2F module, thereby enhancing feature extraction and gradient flow capabilities. The introduction of the ADown module addresses the issue of fixed subsampling in the original model, which tends to filter out fine-grained features. It simultaneously acquires image features and performs spatial compression to reduce computational load, thereby enhancing interaction between images and text. The overall structure of the model is shown in **Figure 1**.

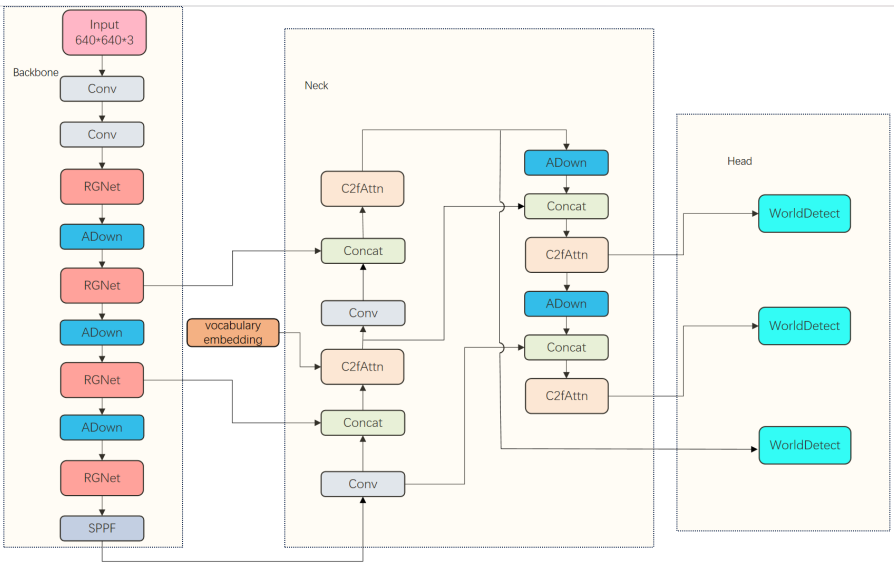

Figure 1. Overall structure

2.1. RGNet model

In the backbone network, when performing feature mapping with feature maps P3, P4, and P5 obtained at different scales, ensuring detection accuracy inevitably leads to redundant computations and underutilization of features. This results in a high computational load for the model, which can cause errors during subsequent object detection tasks, particularly when detecting dense crowds. We propose the RGNet module, which abandons the original Bottleneck module. To compensate for the performance loss resulting from discarding residual blocks, we employ re-parameterized convolutions on the gradient flow branch to enhance feature extraction and gradient propagation capabilities. Reduce the number of parameters and computational complexity while preserving the expressive power of key features, as shown in **Figure 2**.

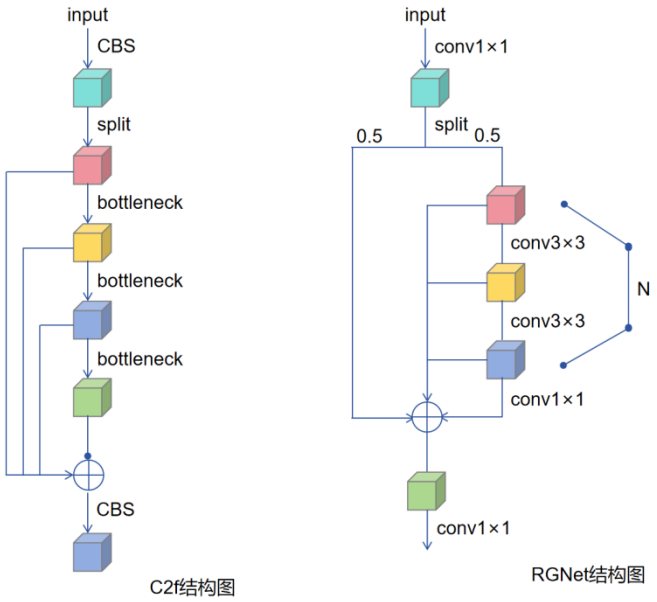

Figure 2. C2f and RGNet structures

RepConv primarily reduces computational complexity by fusing convolution and normalization operations. Standard convolution is expressed as in Equation 1:

$$Y=X\cdot W+b \tag{1}$$

Here, X represents the input feature map, W denotes the convolutional weights, and b is the bias term.

Batch normalization is defined as shown in Equation 2:

$$\begin{aligned}\hat{x}_i &= \gamma\cdot\frac{x_i-\mu}{\sqrt{\sigma^2+\varepsilon}}+\beta \\ &= \frac{\gamma}{\sqrt{\sigma^2+\varepsilon}}\cdot x_i+\left(\beta-\frac{\gamma\cdot\mu}{\sqrt{\sigma^2+\varepsilon}}\right)\end{aligned} \tag{2}$$

The combined convolution formula is shown in Equations 3 and 4:

$$w_{BN}=\frac{\gamma}{\sqrt{\sigma^2+\varepsilon}},b_{BN}=\beta-\frac{\gamma\cdot\mu}{\sqrt{\sigma^2+\varepsilon}} \tag{3}$$

$$\begin{cases}w=w_{BN}\cdot w_{conv}\\ b=w_{BN}\cdot b_{conv}+b_{BN}\end{cases} \tag{4}$$

γ and β represent learnable parameters. μ denotes the mean of the input, σ^2 indicates the variance of the input, and ε serves as a smoothing term to prevent the denominator from becoming zero.

Drawing inspiration from GhostNet, mainstream CNNs exhibit extensive redundancy in their intermediate feature maps. Despite the high correlation among these features, leveraging them can enhance the richness and robustness of feature representations, thereby improving detection performance. RGNet generates partially redundant feature maps through channel splitting and lightweight convolution operations. The main branch extracts features via re-parameterized convolutions, followed by multi-branch lightweight convolutions to extract deep features from redundant information, culminating in feature concatenation. As shown in Equation 5:

$$Y=\mathrm{Conv}_{1\times1}\left(\mathrm{Concat}\left(\mathrm{Conv}_{1\times1},\{\mathrm{Conv}_{3\times3}^{(i)}\}_{i=1}^{n},\mathrm{Conv}_{1\times1}(\cdot)\right)\right) \tag{5}$$

$\{\mathrm{Conv}_{3\times3}^{(i)}\}_{i=1}^{n}$ represents a series of stacked 3×3 convolution operations, serving as the final output after pooling.

The RGNet module enhances feature extraction and gradient diversion through re-parameterized convolutions and multi-branch structures. By employing scaling factors to control module size, it effectively reduces computational load and parameter count while preserving feature expressiveness, making it well-suited for multi-object detection in construction sites.

2.2. ADown

In detection tasks, small-sized targets are often embedded within complex backgrounds, exhibiting distinct scale differences and texture discontinuities. In YOLO-World, downsampling is primarily achieved using convolutions with a stride of 2,although highly efficient, this fixed structural approach inevitably loses some detailed information, particularly the characteristics of small targets or the edge details of objects. ADown employs a dual-branch architecture to simultaneously perform feature extraction and edge information preservation, forming a lightweight downsampling mechanism that integrates convolution and pooling for joint modeling. This addresses

the limitation of fixed downsampling in the original model, which tends to filter out fine-grained features. It achieves spatial compression while capturing image features, thereby reducing computational complexity and facilitating interaction between images and text (**Figure 3**).

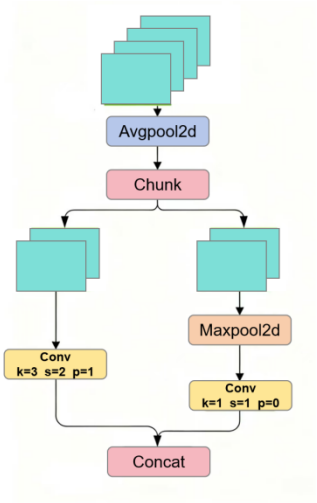

Figure 3. ADown structure

First, the input feature map undergoes average pooling to achieve global compression. Subsequently, it is divided equally along the spatial dimension. One branch performs downsampling and local context modeling through a stride-2 convolution, as shown in Equation 6:

$$F_1=\mathrm{Conv}_{3\times3}^{s=2,p=1}(X_1) \tag{6}$$

The kernel size is 3×3 with 1-based padding.

Another approach first extracts features from edges and regions with strong responses through max pooling operations, followed by downsampling via convolutions with a stride of 1, as shown in Equation 7:

$$F_2=\mathrm{Conv}_{1\times1}^{s=1,p=0}\left(\mathrm{MaxPool}_{2\times2}(X_2)\right) \tag{7}$$

Finally, the outputs from both branches are concatenated using Concat to form a downsampling mechanism that combines multi-scale feature representation with gradient stability.

This module effectively minimizes feature information loss during the downsampling stage while maintaining a lightweight design, enhancing the ability to preserve multiple targets and improving the discrimination accuracy of subsequent detection branches.

3. Experiment

3.1. Configuration and datasets

Our model was developed based on Ultralytics. Experiments were conducted using NVIDIA RTX 5080 GPUs, employing the open-source CLIP text encoder with pre-trained weights to encode input text.

The construction site dataset we employed comprises five categories: nohat, helmet, people, reflect, and workcloth. It consists of 2,204 training images and 551 validation images. During training, the initial learning rate

was set to 0.002, with a weight decay of 0.05. The model was trained for 100 epochs using a batch size of 32.

3.2. Test metrics

To evaluate and improve algorithm metrics, we employ common evaluation indicators [15] such as mean average precision (mAP), number of parameters (Params), and floating-point operations per second (FLOPs). These metrics reflect both model performance and resource consumption.

mAP is a key indicator to evaluate the performance of the object detection model, mAP is mainly divided into mAP@0.5 and mAP@0.5–0.95, mAP@0.5 indicates that the average accuracy of each class is calculated when the IOU threshold is 0.5, and then the average value of all categories is taken, and mAP@0.5–0.95 represents the average mAP value under different IOU thresholds (0.5–0.95, step size is 0.05). Params is expressed as the memory occupied by the model, usually in millions (M). FLOPs are used to measure the computational complexity of a model, usually measured in GFLOPs (billions of floating-point calculations per second).

3.3. Comparative experiments

To further validate the detection performance of our proposed improved algorithm, we conducted comparative experiments using multiple popular object detection models on the construction site dataset. The experimental results are shown in **Table 1**.

Table 1. Comparative experiments

Model	mAP@0.5%	mAP@0.5–0.95%	Params/M	FLOPs/10^9
Yolov5	80.4	55.3	2.65	7.37
Yolov8	83.0	57.8	3.15	8.7
Yolov10 [16]	79.5	54.1	2.30	6.7
Yolov11 [17]	82.8	58.2	2.62	6.6
YOLO-Worldv8	82.4	58.0	3.69	10.0
Ours	83.0	58.9	2.85	8.0

As shown in **Table 1**, compared to YOLO-Worldv8s, the improved algorithm achieves a 22.8% reduction in parameters and a 20% reduction in computational complexity, mAP@0.5% and mAP@0.5–0.95% improved by 0.6% and 0.9%, respectively. Compared to other mainstream YOLO series models, it maintains comparable parameters and FLOPs while achieving a 3.5% improvement over YOLOv10 and a 0.2% improvement over YOLOv11 in mAP@0.5. The improved algorithm proposed in this paper demonstrates enhanced efficiency by achieving greater accuracy while reducing the number of parameters and computational complexity. This performance advantage stems from the judicious introduction of innovative modules and the optimized design of the overall structure.

3.4. Ablation experiment

To further validate the algorithm's effectiveness, we conducted a series of ablation experiments to investigate the specific impact of each module on model performance. The experimental results are shown in **Table 2**.

Table 2. Ablation experiment

Worldv8	RGNet	ADown	mAP@0.5%	mAP@0.5–0.95%	Params/M	FLOPs/10^9
√			82.4	58.0	3.69	10.0
√	√		82.3	57.5	3.27	8.7
√		√	83.9	59.8	3.28	9.3
√	√	√	83.0	58.9	2.85	8.0

As shown in **Table 2**, introducing each module yields varying performance improvements over the baseline YOLO-Worldv8 model. After incorporating RGNet attention, the model's accuracy remains largely unchanged while reducing parameters by 11% and computational complexity by 13%. After introducing the ADown module, the model parameters were reduced by 11%, computational complexity decreased by 7%, and mAP@0.5–0.95% improved by 1.8%. Ultimately, our model achieved a 0.9% increase in mAP@0.5–0.95% while reducing parameters by 22.8% and computational complexity by 20%, striking a balance between accuracy and lightweight performance.

3.5. Visualization experiments

To thoroughly evaluate the adaptability and robustness of the LCS-YOLO model in real construction site scenarios, this paper selected complex background and dense crowd scenes to conduct a visual comparative analysis, as shown in **Figure 4**.

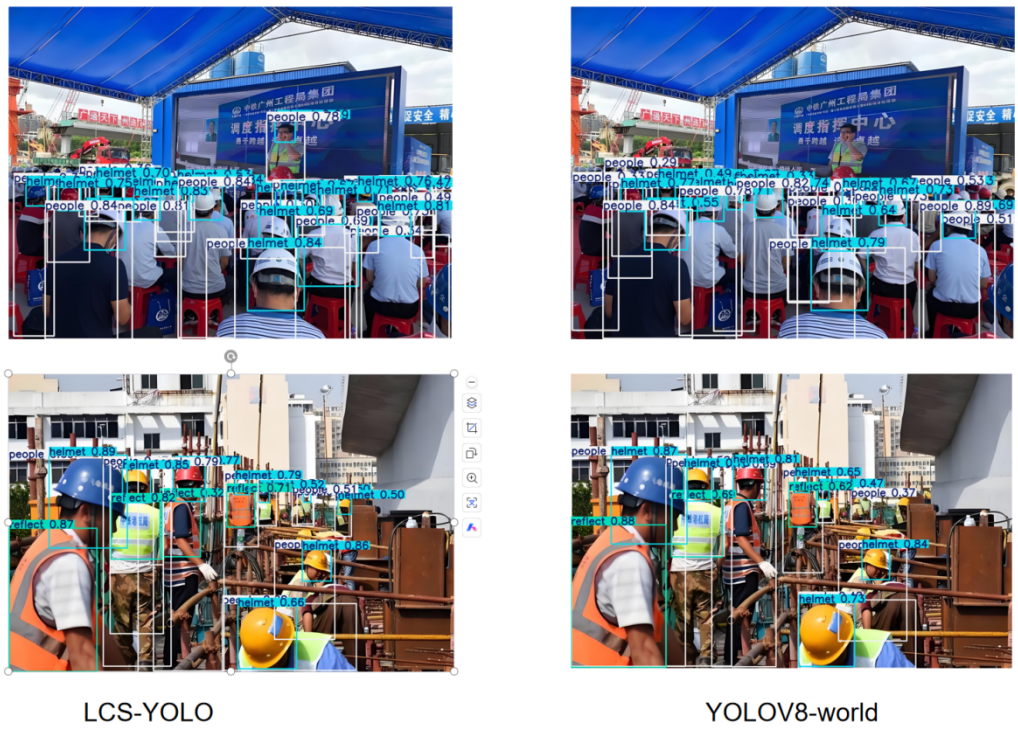

Figure 4. Visual comparison chart

Figure 4 demonstrates the detection performance of LCS-YOLO and YOLOV8-World in complex backgrounds and dense crowd scenes. The figure reveals that YOLOV8-World exhibits missed detections when objects are too small. The results indicate that LCS-YOLO achieves greater lightweight efficiency while demonstrating superior object perception and background suppression capabilities.

4. Conclusion

This paper addresses the information redundancy in the intermediate feature maps of the C2F module within the YOLO-World model and the loss of effective information during fixed-stride dimensionality reduction, which hinders its efficient application in construction site detection. By introducing the RGNet module, which employs re-parameterized convolutions on the gradient flow branch, the model enhances both feature extraction and gradient flow capabilities. While reducing the number of parameters and computational complexity, the model preserves the expressive power of key features. The introduction of the ADown subsampling mechanism employs a dual-branch path design to simultaneously achieve feature extraction and edge information retention. This forms a lightweight subsampling approach that combines convolution and pooling modeling, enhancing the model's perception of multi-scale objects to facilitate interaction between images and text.

The improved model achieves a 7.9% reduction in parameters and a 2.8% decrease in computational complexity while maintaining comparable accuracy. This advancement significantly lowers model complexity without compromising precision, making it better suited for on-site detection requirements.

Disclosure statement

The author declares no conflict of interest.

References

[1] Kelm A, Laußat L, Meins-Becker A, et al., 2013, Mobile Passive Radio Frequency Identification (RFID) Portal for Automated and Rapid Control of Personal Protective Equipment (PPE) on Construction Sites. Automation in Construction, 3638–3652.

[2] Yuan F, Lin Z, Tian Z, et al., 2025, Bio-Inspired Hybrid Path Planning for Efficient and Smooth Robotic Navigation. Int J Intell Robot Appl.

[3] Liang B, Yuan F, Deng J, et al., 2025, Cs-pbft: A Comprehensive Scoring-Based Practical Byzantine Fault Tolerance Consensus Algorithm. J Supercomput, 81: 859.

[4] Yuan F, Huang X, Jiang H, et al., 2025, An xLSTM–XGBoost Ensemble Model for Forecasting Non-Stationary and Highly Volatile Gasoline Price. Computers, 14: 256.

[5] Girshick R, Donahue J, Darrell T, et al., 2014, Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Columbus, 580–587.

[6] Zhang K, Yuan F, Jiang Y, et al., 2025, A Particle Swarm Optimization-Guided Ivy Algorithm for Global Optimization Problems. Biomimetics (Basel), 10(5): 342.

[7] Sohan M, Sai Ram T, Reddy R, et al., 2024, A Review on YOLOv8 and Its Advancements, International Conference on Data Intelligence and Cognitive Informatics, Springer, Singapore, 529–545.

[8] Bochkovskiy A, Wang CY, Liao HYM, 2020, YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv. https://arxiv.org/abs/2004.10934

[9] Liu S, Qi L, Qin HF, et al., 2018, Path Aggregation Network for Instance Segmentation, Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, Salt Lake City, 8759–8768.

[10] Ding X, Zhang XY, et al., 2021, RepVGG: Making VGG-Style ConvNets Great Again, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.

[11] Zhang YJ, Xiao FS, Lu ZM, 2022, Helmet Wearing State Detection Based on Improved YOLOv5s. Sensors, 22(24): 9843.

[12] Fang S, Chen C, Li Z, et al., 2024, YOLO-ADual: A Lightweight Traffic Sign Detection Model for a Mobile Driving System. World Electric Vehicle Journal, 15(7): 323.

[13] Cheng T, Song L, et al., 2024, Yolo-World: Real-Time Open-Vocabulary Object Detection, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 16901–16911.

[14] Ding X, Zhang X, Ma N, et al., 2021, RepVGG: Making VGG-Style ConvNets Great Again, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 13733–13742.

[15] Jiang B, Luo R, Mao J, et al., 2018, Acquisition of Localization Confidence for Accurate Object Detection, Proceedings of the European Conference on Computer Vision (ECCV), 784–799.

[16] Wang A, Chen H, Liu LH, et al., 2024, YOLOv10: Real-Time End-to-End Object Detection, Proceedings of the 38th Annual Conference on Neural Information Processing Systems 2024, NeurIPS, Vancouver, 1–28.

[17] Khanam R, Hussain M, 2024, YOLOv11: An Overview of the Key Architectural Enhancements. arXiv. https://arxiv.org/abs/2410.17725