Scientific Reports

https://doi.org/10.1038/s41598-026-40457-y

Article in Press

SFD-YOLO for small-object fragment impact detection in warhead target-plate testing

Received: 1 December 2025

Accepted: 12 February 2026

Published online: 16 February 2026

Cite this article as: Liu H., Ding Y., You W. *et al.* SFD-YOLO for small-object fragment impact detection in warhead target-plate testing. *Sci Rep* (2026). https://doi.org/10.1038/s41598-026-40457-y

Huaqi Liu, Yonghong Ding, Wenbin You & Yaning Li

SFD-YOLO for small-object fragment impact detection in warhead target-plate testing

Huaqi Liu[1], Yonghong Ding[2], Wenbin You[1,*], and Yaning Li[2]

[1]School of Electrical and Control Engineering, North University of China, Taiyuan 030051, China
[2]State Key Laboratory of Extreme Environment Optoelectronic Dynamic Measurement Technology and Instrument, North University of China, Taiyuan 030051, China
[*]youwenbin@nuc.edu.cn

ABSTRACT

To address missed detections and limited efficiency in small fragment-impact recognition on target plates during warhead testing, this paper proposes small fragment detection YOLO (SFD-YOLO), a task-oriented detection framework built upon YOLOv11. An improved Spatial–Channel Reconstruction C3k2 (SCC3k2) module is integrated into the backbone to suppress redundant responses in both spatial and channel dimensions, enhancing the representation of weak micro-scale impact cues. To improve sensitivity to extremely small targets, we introduce an additional micro-object detection head and adopt an Asymptotic Feature Pyramid Network (AFPN) for progressive multi-level feature alignment and fusion, which strengthens feature consistency across pyramid levels. In addition, a Lightweight Adaptive Extraction (LAE) module is employed to replace standard convolutions, reducing model complexity while maintaining effective feature extraction. To comprehensively evaluate performance in realistic testing scenarios, we construct a multi-scene target-plate dataset from a series of static explosion experiments, covering both penetrative fragment holes and non-penetrative impact marks. Experimental results demonstrate that SFD-YOLO achieves 98.1% mAP@0.5 and 69.7% mAP@0.5:0.95, outperforming the YOLOv11 baseline by 2.7% in mAP@0.5 and 6.8% in mAP@0.5:0.95, at 135 FPS with only 2.15M parameters. Moreover, robustness evaluations under image degradations indicate that SFD-YOLO maintains more stable detection performance than the baseline. The proposed method provides a high-precision real-time solution for fragment lethality evaluation and shows potential for broader applications such as metal surface defect inspection.

Introduction

Fragments are one of the key metrics for evaluating warhead lethality[1], as their velocity, flight angle, and spatial distribution directly reflect the munition's destructive performance[2]. Currently, fragment characterization methods fall into two categories: contact-based measurements[3] and non-contact testing[4]. Contact methods are widely used due to their straightforward and intuitive response to experimental outcomes. In static explosion trials, the most widely used procedure is the target-plate method[5], whereby fragments produced by detonation strike a calibrated plate and parameters such as impact locations, hit density, and hole diameters are recorded to quantify destructive effects. However, manual measurement of hole distributions[6] on target plates suffers from low statistical efficiency and high labor demands. Harsh conditions in static-blast test ranges, such as strong winds, dust, and other adverse weather, further hinder observers' visibility and introduce data-collection errors, ultimately compromising assessment accuracy. Therefore, applying object detection techniques to fragment-hole detection on target plates holds significant potential for improving real-time monitoring, data analysis, and operational effectiveness evaluation[7].

Currently, research that combines artificial intelligence with industrial inspection has concentrated mainly on conventional defects and surface fault detection, while studies applying object detection algorithms to the identification of fragment holes on target plates remain scarce[8]. Detecting fragment holes on target plates presents several challenges: the holes are very small, densely distributed, and it is difficult to acquire large datasets of target-plate inspection scenarios[9]. With the advancement of computer vision[10], various approaches have been proposed, broadly divided into image-processing-based methods and deep-learning-based methods, such as classification[11] and segmentation[12]. Notably, multi-threshold binarization has shown effectiveness for representing small objects under complex imaging conditions[13]. However, image-processing-based algorithms suffer from limited generalization and robustness to specific defect types[14], as each target requires manual feature engineering and parameter tuning, resulting in high computational cost that fails to meet real-time industrial requirements[15]. In contrast, deep neural networks provide powerful feature extraction capabilities and have achieved higher accuracy in detection tasks. Consequently, many researchers have shifted their focus from traditional feature-engineering techniques to deep-learning-based detection[16]. In numerous application domains, the You Only Look Once (YOLO) series of algorithms has gained widespread

adoption in object detection tasks due to its high inference efficiency and competitive accuracy[17]. Its single-stage architecture is particularly advantageous in scenarios that require real-time performance[18]. Based on these strengths, this work employs a YOLO-based detection framework for efficient fragment localization. Nonetheless, the abundance of small targets in target-plate inspection scenarios continues to expose YOLO's limitations in small-object detection[19].

In this paper, we present small fragment detection-YOLO (SFD-YOLO), a high-precision yet lightweight model for fragment-hole detection on target plates. In the feature-extraction stage of a YOLOv11n backbone, we integrate the SCC3k2 module, which combines spatial reconstruction and channel reconstruction units—and append a dedicated small-object detection head. An Asymptotic Feature Pyramid Network (AFPN) is then employed to progressively fuse low-, middle-, and high-level feature representations, thereby reinforcing semantic consistency across different scales. To address data scarcity and varying weather conditions, we augment our dataset with affine transforms and synthetic fog and dust effects, yielding a multi-condition target-plate fragment dataset. Experimental results demonstrate that training on this enhanced dataset reduces model parameters and computational complexity while maintaining high detection accuracy.The main contributions of this article are as follows.

1. We propose a novel SFD-YOLO model that integrates the SCC3k2 module and a lightweight adaptive extraction (LAE) unit, resulting in both higher detection precision and a reduced parameter count.

2. We optimize the YOLOv11 detection head by adding a P2 small-object layer and employing an Asymptotic Feature Pyramid Network (AFPN) to strengthen multi-scale feature fusion and small-fragment recognition.

3. We construct a custom target-plate fragment dataset of 4565 images captured in static-blast test ranges and augmented with synthetic fog and dust to cover diverse weather conditions.

4. We demonstrate the superiority of SFD-YOLO through comprehensive comparison and ablation experiments. The proposed model achieves an mAP of 98.1%, with only 2.15M parameters and 11.8 GFLOPs, and reaches 135 FPS, outperforming mainstream detectors in both accuracy and efficiency.

The remainder of this paper is organized as follows. Section II reviews related work on target-plate and small-object detection. Section III describes the SFD-YOLO architecture in detail. Section IV presents our experimental setup and compares results on the custom dataset against other methods. Finally, Section V concludes and outlines future research directions.

Related Work

Introduction to the Object Detection Algorithm

Object detection is the task of identifying instances of semantic object classes in an image and localizing them with bounding boxes[20]. In recent years, deep convolutional neural networks (CNNs) have dramatically improved detection performance. Modern CNN-based detectors are generally divided into two categories: two-stage methods and one-stage methods. Two-stage detectors first generate candidate object regions and then classify each one[21], whereas one-stage detectors skip the proposal step and directly predict object locations and labels in a single network pass[22].

R-CNN is an early two-stage detection framework that first applied deep learning to object detection. It generates candidate regions and applies a convolutional neural network to extract features from each region individually[23]. Its slow inference speed led to the development of Fast R-CNN for improved efficiency. Fast R-CNN improved efficiency by computing a convolutional feature map for the entire image and then pooling features for each region[24]. Faster R-CNN further improved both accuracy and speed by introducing a Region Proposal Network (RPN), which generates candidate regions and shares convolutional features with the detection network[25].

One-stage models tend to be much faster, making them well-suited for real-time applications. For example, SSD uses a single CNN to predict offsets and confidences for a fixed set of default boxes over multiple feature-map scales[26]. CenterNet detects objects by identifying their center points and key corner positions, enabling accurate localization without the need for anchors[27]. DETR performs object detection by leveraging object queries and positional encodings to model global relationships in an end-to-end manner[28]. one-stage models are well suited for real-time tasks, with the YOLO series standing out for its balance of speed and accuracy. Through continuous improvements, YOLO has become a widely adopted solution in fast and efficient object detection applications.

Application of the Fragments Detection Algorithms

Early fragment detection methods were primarily based on manual measurements or traditional image processing techniques, which often lacked robustness, scalability, and adaptability to complex scenarios[29]. With the development of deep learning, particularly object detection algorithms, various approaches have been proposed to automate the detection and classification of fragment impacts on target plates and related test surfaces.

For target detection of fragment perforations on target plates, Lei[30] achieved fragment recognition through joint training of a Region Proposal Network (RPN) and Fast R-CNN; however, the method suffers from slow detection speed and limited utilization of shallow features. Wei[31] replaced the SSD backbone with DenseNet and introduced both channel and spatial attention mechanisms, resulting in improved detection performance for small fragments. He[32] proposed an improved YOLOv5-based algorithm for detecting fragment clusters from warheads by incorporating coordinate attention mechanisms and optimizing the upsampling module to reduce information loss in small-object features. However, detection accuracy remains an area for further improvement. Gao[33] introduced a DSA-EF semantic segmentation network with an ECA-based channel attention mechanism that considers inter-channel correlations of multi-scale target features, although the model incurs high computational cost. At present, research specifically targeting fragment hole detection on target plates remains limited, and there is still substantial room for improvement in both detection accuracy and efficiency.

Small Object Detection Methods

Deep learning–based object detection methods typically rely on backbone networks to extract high-level features. However, in target plate images, small fragment features may occupy only a limited number of pixels, making them difficult to detect. To address this, researchers have proposed various multi-scale feature fusion techniques. Zhang[34] introduced ACNet to reduce feature loss during downsampling and enhance context representation. Tang[35] developed pyramidBox to improve contextual awareness using pyramid anchors and a context-sensitive structure. Guo[36] proposed AugFPN to align semantic and detailed features via supervised fusion. Zhang[37] presented FFCA-YOLO, an efficient detector that enhances feature representation and context awareness. It incorporates an improved BiFPN neck to aggregate multi-scale feature maps and strengthen the semantic representation of small targets. These approaches demonstrate that improving the quality of multi-scale feature fusion can significantly enhance small object detection performance.

In addition, data augmentation methods help improve model generalization. Chen[38] proposed Stitcher, which assembles smaller image patches to enhance the presence of small targets during training. Yu[39] introduced a Scale Matching strategy to reduce size discrepancies by adaptively cropping objects based on scale.

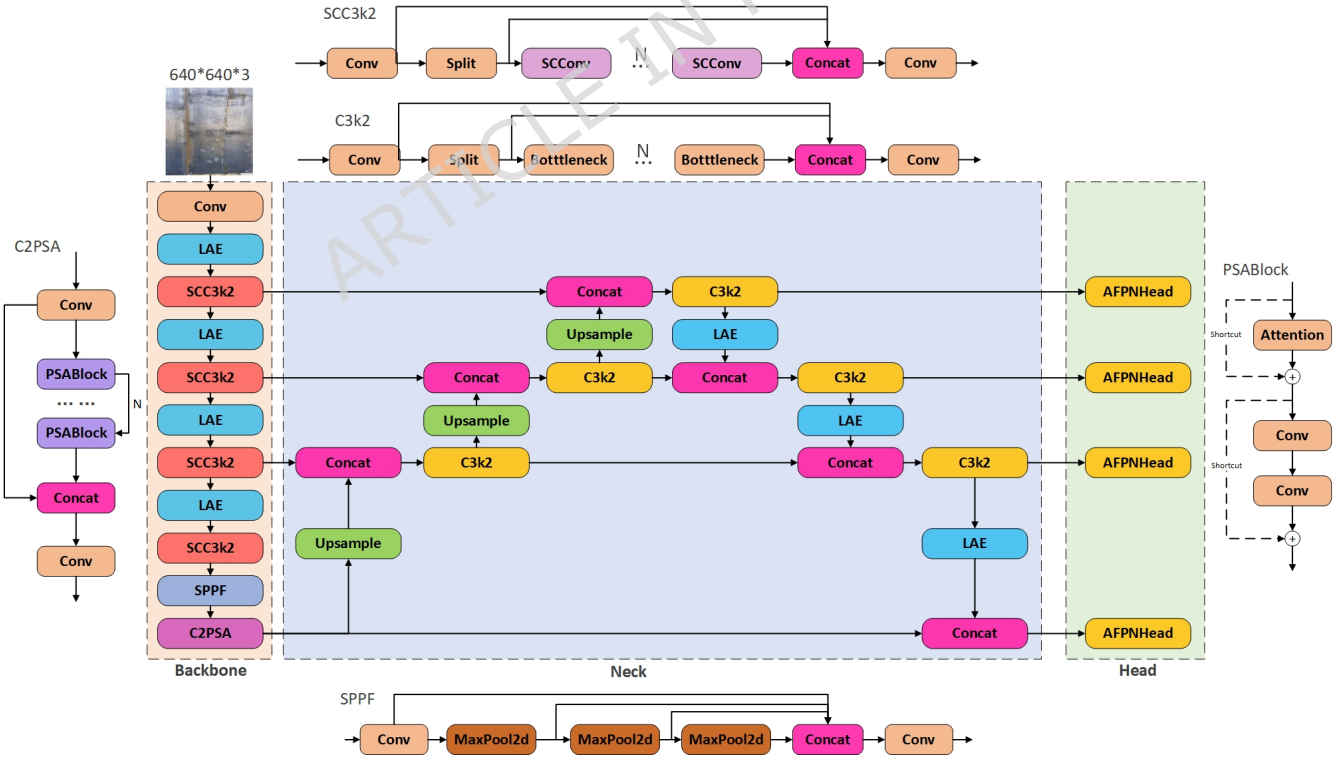

Figure 1. Model architecture diagram of SFD-YOLO.

Methodology

Network Architectrue of SFD-YOLO

Based on existing studies, detecting fragment holes on target plates presents considerable challenges owing to their small size, low pixel resolution, and dense spatial distribution[33]. In this paper, we introduce SFD-YOLO, a model specifically designed for fragment-hole detection. Fig. 1 illustrates the overall architecture of SFD-YOLO, which extends the YOLO series with several key improvements to ensure real-time inference speed while markedly enhancing both accuracy and robustness for tiny targets. The network is divided into a backbone, a neck, and detection heads: the backbone extracts multi-scale features from the input target-plate image, the neck fuses these features, and the detection heads employ an efficient feature-fusion strategy to produce object classification and bounding-box regression outputs.

First, we redesign the backbone feature extraction by introducing the SCC3k2 module to enhance spatial–channel representation while suppressing redundant information, improving sensitivity to weak and tiny impact patterns. Second, we incorporate a lightweight adaptive extraction (LAE) module to reduce computation with minimal loss of discriminative details, enabling real-time deployment. Third, unlike the standard YOLO head that detects on three scales (P3–P5), we extend the detection pyramid to four scales (P2–P5) and adopt an AFPN-based head to strengthen cross-scale feature fusion and semantic consistency. These architectural modifications jointly improve detection accuracy and robustness on small, densely distributed fragment perforations and non-penetration marks while maintaining a compact model size and high inference speed.

Within the backbone, to address the particular difficulty of detecting small fragment holes, we integrate an enhanced C3k2 module (denoted SCC3k2), which combines a Spatial Reconstruction Unit (SRU) and a Channel Reconstruction Unit (CRU). By reconstructing features along both spatial and channel dimensions, SCC3k2 effectively suppresses redundant information and improves the extraction of fine-grained details. To further streamline feature extraction and reduce model complexity, we replace standard convolutional layers with a Lightweight Adaptive Extraction (LAE) module, thereby lowering computational cost and boosting inference speed.

In the neck and detection-head stages, we enhance the original structure with a dedicated small-object detection layer and an Asymptotic Feature Pyramid Network (AFPN) to address the multi-scale, diminutive nature of fragment holes. AFPN progressively integrates low-, mid-, and high-level features, narrowing semantic gaps between scales and strengthening the model's responsiveness to tiny targets. Ultimately, SFD-YOLO achieves high-precision localization and classification of fragment holes by aggregating outputs from four parallel detection heads.

The remainder of this section details the design principles and implementation of each improved module, as well as experimental validation on the target-plate fragment-hole detection task.

Spatial and Channel Reconstruction C3k2 Module

During fragment impact inspection, the targets are extremely small, irregularly shaped, and often exhibit weak contrast against reflective metallic backgrounds. In such cases, missed detections are frequently caused by insufficient preservation of fine-grained spatial cues and redundant channel responses in early feature extraction, rather than by the lack of high-level semantics.

To enhance discriminative feature extraction for tiny fragment marks, we propose an improved SCC3k2 module based on the original C3k2 block, as illustrated in Fig. 2. The main modification is to replace the conventional *Bottleneck* unit inside C3k2 with an *SC_Bottleneck* consisting of two SCConv layers. This design preserves the original CSP-style split–transform–concat topology while introducing spatial and channel reconstruction, enabling the network to better capture subtle edges and low-contrast textures of micro-scale penetration holes and non-penetration impact marks.

SCConv integrates two complementary components (Fig. 3): a Spatial Reconstruction Unit (SRU) that suppresses spatial redundancy and improves local consistency, and a Channel Reconstruction Unit (CRU) that reduces channel-wise redundancy and enhances discriminative responses. Such dual-dimensional reconstruction is particularly beneficial for fragment-hole inspection, where target cues are weak and background interference is strong.

Considering that shallow feature maps usually contain fewer channels with limited redundancy, while deeper layers tend to be more redundant, we adopt a selective replacement strategy: SCC3k2 is applied only to key backbone stages for feature extraction, whereas standard bottleneck structures are retained in later feature fusion blocks to avoid over-filtering multi-scale information. This strategy provides a favorable trade-off between representation capability and computational cost.

The SRU adopts a separation-and-reconstruction strategy to reduce redundant features along the spatial dimension, as shown in Fig. 4. By compressing irrelevant information, SRU enables convolution layers to focus more effectively on salient detail regions, which is important for suppressing background reflections and noise patterns on metallic target plates.

$$X_{out} = GN(X) = \gamma \frac{X - \mu}{\sqrt{\sigma^2 + \varepsilon}} + \beta \tag{1}$$

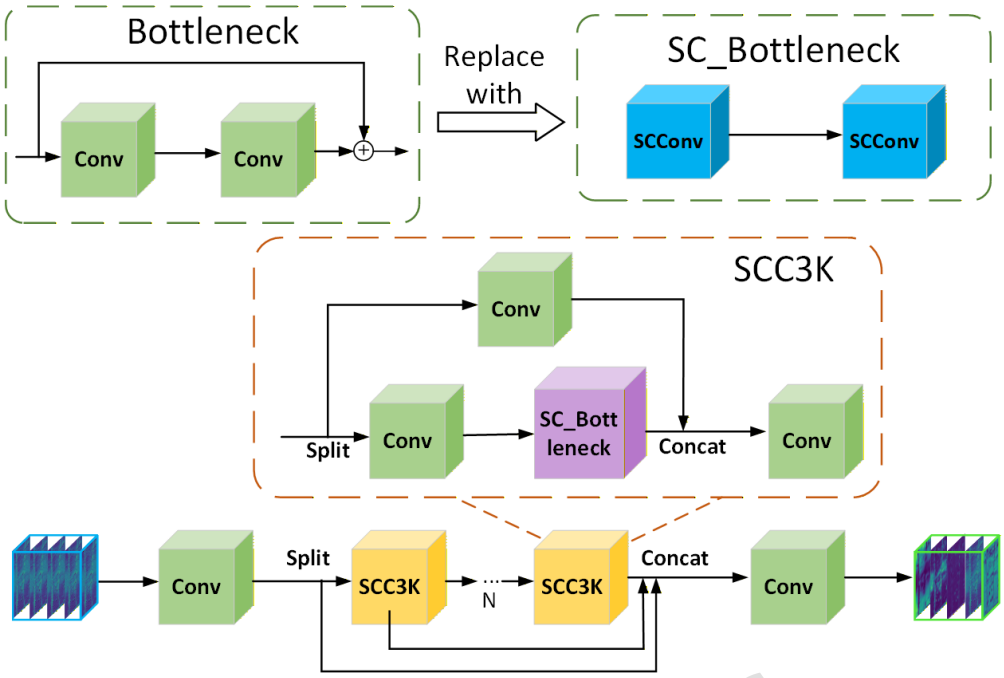

Figure 2. Architecture of the proposed SCC3k2 module.

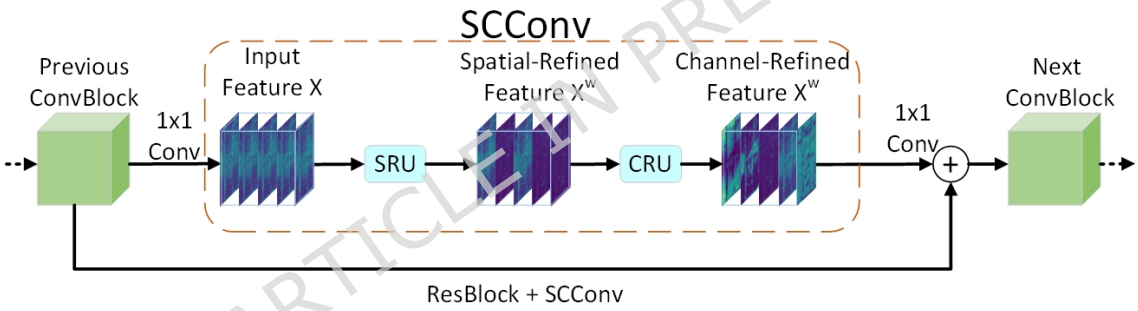

Figure 3. Architecture of SCConv, consisting of a Spatial Reconstruction Unit (SRU) and a Channel Reconstruction Unit (CRU).

where μ and σ are the mean and standard deviation of X, ε is a small constant for numerical stability, and γ and β are trainable affine parameters.

$$W_\gamma = \{w_i\} = \frac{\gamma_i}{\sum_{j=1}^{C} \gamma_j},\ i, j = 1, 2, \cdots, C \tag{2}$$

$$W = Gate(Sigmoid(W_\gamma(GN(X)))) \tag{3}$$

In the separation stage, informative and less informative spatial feature maps are distinguished using the scaling factors γ in group normalization, followed by a sigmoid-based gating mechanism. Two weighted components are obtained: informative (X_1^w) and redundant (X_2^w).

$$\begin{cases} X_1^w = W_1 \otimes X, \\ X_2^w = W_2 \otimes X, \\ X_{11}^w \oplus X_{22}^w = X^{w1}, \\ X_{21}^w \oplus X_{12}^w = X^{w2}, \\ X^{w1} \cup X^{w2} = X^w \end{cases} \tag{4}$$

In the reconstruction stage, SRU performs cross-reconstruction to enhance information flow while suppressing spatially redundant responses, resulting in a more expressive spatial representation for tiny impact marks.

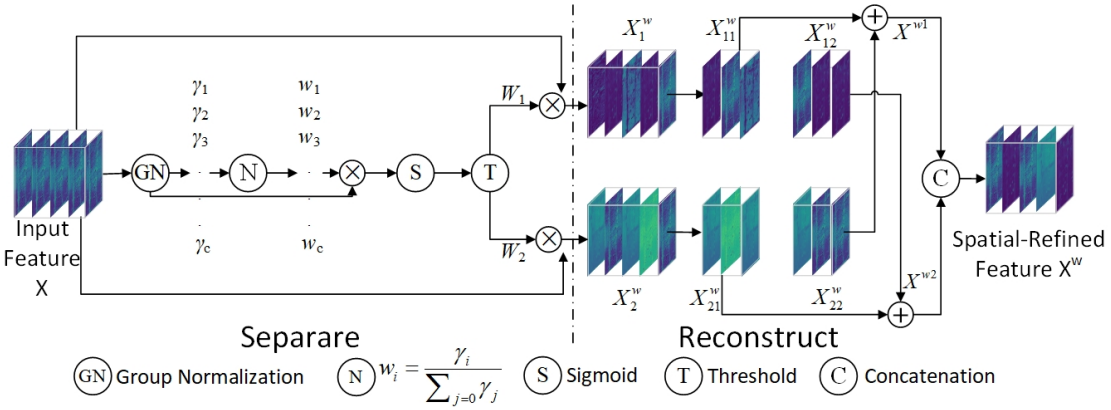

Figure 4. Architecture of Spatial Reconstruction Unit (SRU).

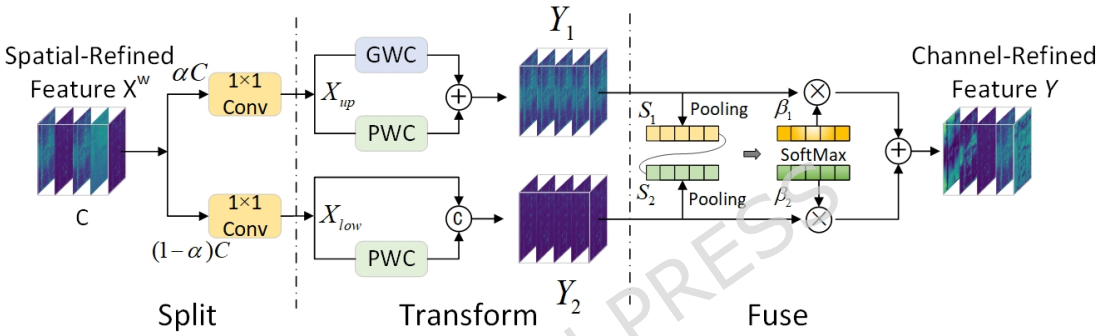

Figure 5. Architecture of Channel Reconstruction Unit (CRU).

The CRU employs a split–transform–merge approach to minimize channel redundancy and strengthen discriminative channel responses, as shown in Fig. 5.

First, the input feature map is split into two parts along the channel dimension. The upper branch combines group-wise convolution (GWC) and point-wise convolution (PWC) to capture diverse channel interactions efficiently, while the lower branch uses lightweight 1×1 convolutions and feature reuse to preserve fine-grained details. Finally, both branches are fused using a lightweight attention mechanism to adaptively balance their contributions:

$$Y_1 = M^G X_{up} + M^{P_1} X_{up} \tag{5}$$

$$Y_2 = M^{P_2} X_{low} \cup X_{low} \tag{6}$$

$$\beta_1 = \frac{e^{s_1}}{e^{s_1} + e^{s_2}}, \quad \beta_2 = \frac{e^{s_2}}{e^{s_1} + e^{s_2}}, \quad \beta_1 + \beta_2 = 1 \tag{7}$$

$$Y = \beta_1 Y_1 + \beta_2 Y_2 \tag{8}$$

where s_1 and s_2 are attention scores generated from global descriptors of the two branches. The coefficients β_1 and β_2 are automatically learned during training, enabling adaptive channel fusion without manual tuning.

Overall, SCC3k2 improves feature representation for tiny fragment marks by jointly reducing spatial and channel redundancy, providing a stronger backbone foundation for subsequent multi-scale fusion and detection head optimization.

Lightweight Adaptive Extraction Module

In target-plate fragment-hole detection, the objects of interest are extremely small, sparsely distributed, and often exhibit weak visual responses under complex surface textures and illumination variations. These characteristics make conventional convolutional operations prone to suppressing fragment-hole features during downsampling and feature fusion.

To address this issue, we adopt the Lightweight Adaptive Extraction (LAE) module as a task-oriented enhancement for low-level feature stabilization rather than as a generic multi-scale extraction block[40]. In SFD-YOLO, LAE is selectively

integrated into both the backbone and neck to reinforce fragile fragment-hole responses and suppress background interference caused by metallic reflections and surface irregularities.

As illustrated in Fig. 6, LAE adaptively extracts information from multi-scale feature representations and dynamically modulates feature responses according to local contextual cues. This adaptive behavior allows the network to emphasize informative micro-scale structures, such as fragment-hole boundaries, while attenuating irrelevant background patterns.

Algorithm 1 LAE Module

Require: Input feature $X \in \mathbb{R}^{H\times W\times C}$
Ensure: Output feature X_{out}

Lightweight Extraction Branch

$F_L \leftarrow \text{GroupedConv}(X,\ \text{groups} = C/16)$
$F_L \leftarrow \text{Rearrange}(F_L)$
$F_L \leftarrow \text{DimensionMapping}(F_L)$

Adaptive Extraction Branch

$G \leftarrow \text{AvgPool}(X)$
$F_A \leftarrow \text{Conv}_{1\times 1}(G)$
$F_A \leftarrow \text{DimensionMapping}(F_A)$
$F_A \leftarrow \text{Rearrange}(F_A)$
$W \leftarrow \text{Softmax}(F_A)$

Fusion

$F \leftarrow F_L \odot W$
$X_{\text{out}} \leftarrow X + F$
return X_{out}

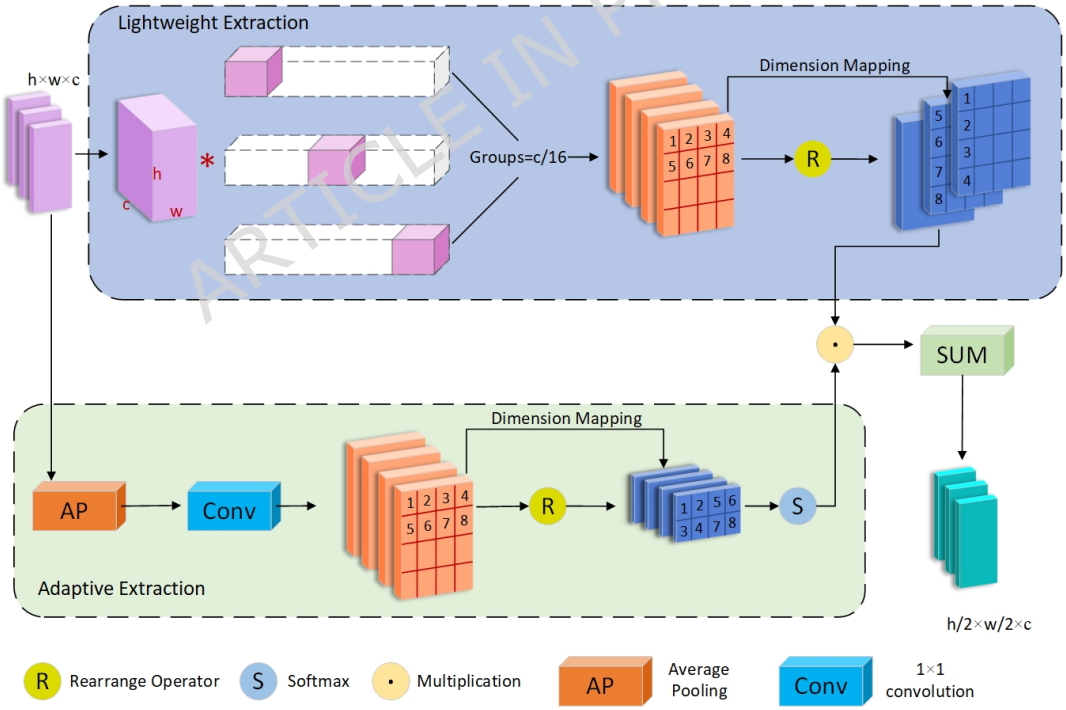

Figure 6. Architecture of Lightweight Adaptive Extraction module.

Structurally, LAE adopts a dual-branch design consisting of a lightweight extraction branch and an adaptive refinement branch. The lightweight branch employs group convolution to reduce computational cost while preserving essential spatial–channel interactions. The adaptive branch utilizes average pooling followed by a 1×1 convolution and softmax-based weighting to refine local feature responses in a context-aware manner.

By incorporating group convolution with N groups, the parameter count is reduced to approximately $1/N$ of standard convolution. A dimension-mapping operation reorganizes spatial information into the channel domain, mitigating information loss during downsampling. The outputs of the two branches are fused through element-wise multiplication and residual addition, achieving a balance between computational efficiency and representational robustness.

Within the SFD-YOLO framework, LAE serves as a complementary component that cooperates with the SCC3k2 backbone and the AFPN-based detection head. Rather than acting as an independent innovation, LAE contributes to a system-level design that stabilizes micro-scale feature responses and improves detection robustness under challenging target-plate conditions, thereby contributing to a balanced trade-off between detection accuracy and computational efficiency.

Detection Head Improvement

In fragment-hole detection on target plates, especially for micro-scale perforations and non-penetration impact marks, multi-scale feature fusion plays a decisive role in detection accuracy. These targets usually occupy only a limited number of pixels and may appear with weak contrast on reflective metallic surfaces. As a result, the detector must preserve fine-grained spatial cues while maintaining sufficient contextual information.

The baseline YOLOv11n employs a standard Feature Pyramid Network (FPN)-style neck to fuse multi-level features for objects of different sizes. Although effective in general scenarios, direct top-down fusion may introduce semantic inconsistency between pyramid levels. This issue becomes more evident for tiny and low-contrast fragment holes, where shallow details can be easily diluted by stronger but coarser high-level semantics.

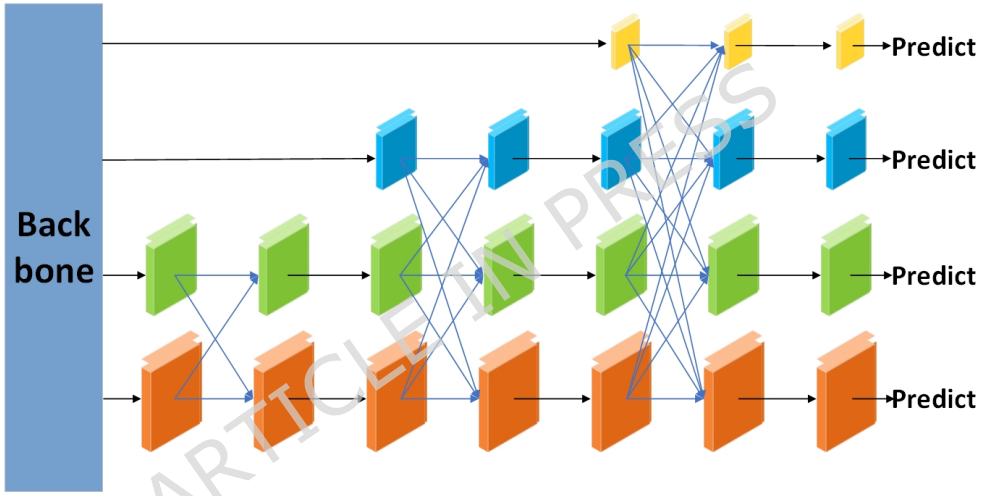

Figure 7. Architecture of Asymptotic Feature Pyramid Network (AFPN).

To address this problem, we introduce an Asymptotic Feature Pyramid Network (AFPN) as a task-adapted fusion head in SFD-YOLO. As illustrated in Fig. 7, AFPN adopts a staged and progressive fusion strategy, which gradually aligns low-level spatial details with higher-level contextual representations. Compared with conventional FPN, this asymptotic aggregation reduces the semantic gap across pyramid levels and helps retain localization-sensitive information that is critical for micro-target detection.

More importantly, unlike the original AFPN design that targets pixel-level labeling and general multi-class detection[41], our objective is to enhance robustness for point-like targets under complex backgrounds (e.g., reflective artifacts and surface stains). Therefore, AFPN is used here as a scale-alignment and feature-consistency module to better support fragment-hole localization in a constrained detection setting.

In addition to feature fusion, we further enhance small-object sensitivity by introducing a dedicated micro-object detection head. Specifically, we extend the original three-head prediction (P3–P5) in YOLOv11n by adding an extra high-resolution detection head at P2, forming a four-head detection scheme (P2–P5). This modification explicitly decouples micro-scale fragment detection from larger-scale feature aggregation, enabling the network to exploit finer spatial resolution for tiny hole patterns.

To implement this design efficiently, we build an AFPNHead module that integrates AFPN-based progressive fusion with Adaptive Spatial Feature Fusion (ASFF) re-weighting. The overall architecture is shown in Fig. 8. In AFPNHead, the feature maps from P2–P5 are first aligned by lightweight convolution layers, and then fused through cascaded ASFF blocks to adaptively aggregate spatial contributions from different scales. This dynamic fusion suppresses noisy or conflicting responses

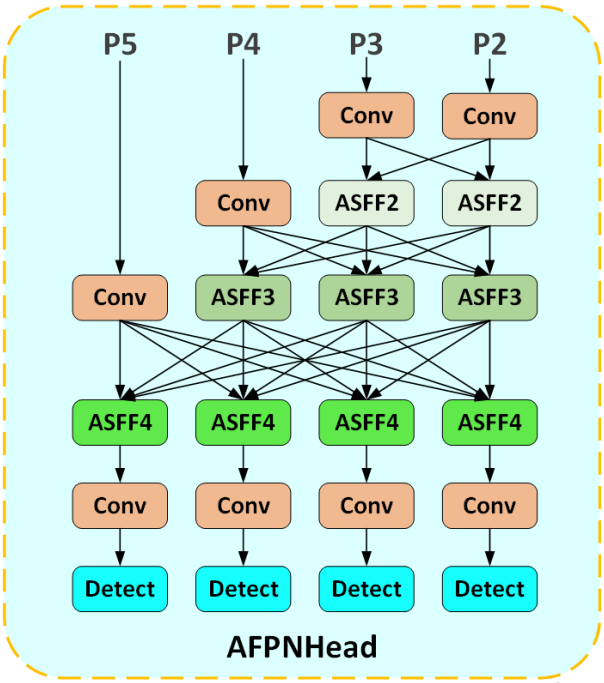

Figure 8. Architecture of AFPNHead.

and emphasizes discriminative impact regions, which is beneficial for target-plate imagery with illumination variations and reflective interference.

As a result, AFPN serves as an effective scale-alignment mechanism between the enhanced backbone features and the task-specific detection heads, improving the discrimination between penetrative fragment holes and non-penetration impact marks under complex target-plate conditions. The feature fusion process at each level can be formulated as:

$$y_{ij}^{l} = \alpha_{ij}^{l} \cdot x_{ij}^{1\to l} + \beta_{ij}^{l} \cdot x_{ij}^{2\to l} + \gamma_{ij}^{l} \cdot x_{ij}^{3\to l}. \tag{9}$$

where $x_{ij}^{k\to l}$ denotes the feature map from pyramid level P_k resized and aligned to level P_l, and α_{ij}^{l}, β_{ij}^{l}, and γ_{ij}^{l} are adaptive spatial weights learned by ASFF such that $\alpha_{ij}^{l} + \beta_{ij}^{l} + \gamma_{ij}^{l} = 1$.

Experiment

Establishment of the Target Plate Fragment Dataset

In static warhead detonation experiments, target plates are typically arranged at specified distances around the warhead with defined angular coverage to collect fragment impact information. Through static detonation tests, the numbers of fragment penetration holes and non-penetrative impact marks formed on the target plates can be statistically analyzed, enabling the evaluation of key performance indicators such as fragment quantity, spatial distribution characteristics, and impact density. These measurements provide direct evidence for warhead casing assessment, design verification, and final acceptance.

Fig. 9 illustrates a typical layout of a static warhead detonation experiment. Taking the blast center as the origin, target plates are symmetrically arranged along a circular perimeter with a radius of approximately 8 m, while the observation bunker is located about 500 m away from the blast center. The circular target region consists of three sector-shaped areas, each assembled from ten equivalent target plates. Each equivalent target plate has a length of 1.2 m, a height of 2.6 m, and a thickness of 6 mm. Within each sector, the plates are spliced to form a continuous arc-shaped target plate of approximately 12 m in length, with an angular separation of about 36° between adjacent sectors.

After static detonation, fragments impact the target plates and produce characteristic perforation patterns. High-resolution images of the target plates are then acquired for subsequent image-based analysis. Similar experimental configurations and data acquisition procedures can be found in recent studies on fragment localization and spatial distribution measurement[42].

Due to the constraints of static explosion experiments, large-scale data acquisition is challenging. To address this issue, we constructed a dedicated target-plate fragment dataset based on a series of previous static-blast trials. Each fragment impact instance was manually annotated and categorized into one of two classes: *penetrative fragment holes* and *non-penetrative*

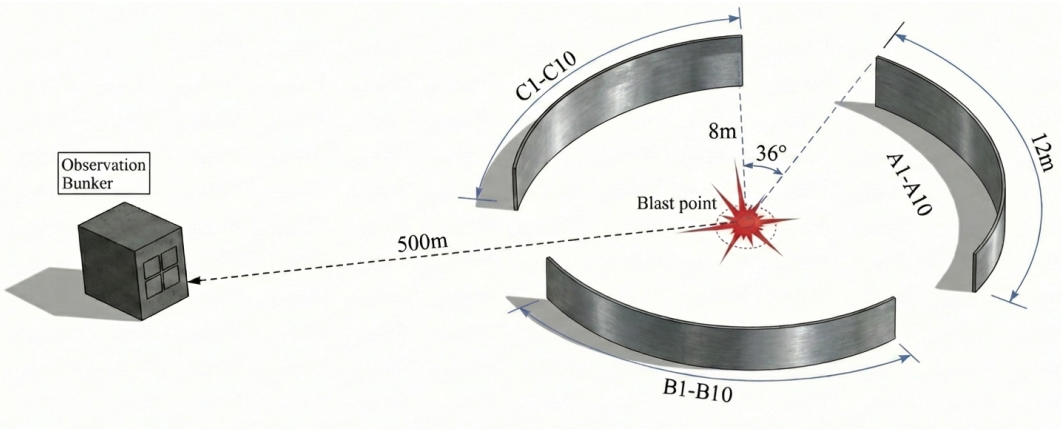

Figure 9. Layout of the static explosion experiment.

impact marks. These two categories correspond to different damage mechanisms and are of practical significance for lethality assessment.

As illustrated in Fig. 10, representative fragment impact holes in the target-plate images are extremely small and occupy only a limited number of pixels. The majority of fragment holes exhibit bounding-box sizes on the order of tens of pixels, typically ranging from approximately 25×25 to 45×45 pixels. To facilitate visual understanding, zoomed-in views of representative fragment holes are provided together with pixel-size annotations and a physical scale bar. These images were acquired in an

Figure 10. Examples of fragment impact holes with pixel-level size annotations and physical scale reference.

outdoor static detonation test range. Due to on-site installation constraints, some target plates may exhibit slight tilt angles relative to the camera optical axis, which can lead to minor variations in surface reflectivity. To mitigate illumination bias, all images were captured under natural diffuse daylight conditions without additional directional light sources, and the camera was positioned approximately front-facing to the target plate surface with stable exposure settings. Given that illumination quality and uniformity can affect surface defect detection[43], we further evaluate the robustness of the proposed system under blur, contrast variation, and illumination change conditions during the experimental phase.

To alleviate the limited dataset size and improve generalization performance, data augmentation techniques were employed, including affine transformations and the superposition of synthetic fog and dust effects, as Fig. 11 illustrated

The final augmented dataset contains 4565 images with an average resolution of 2777×2188 pixels. In total, 23,576 fragment impact instances were annotated, of which penetrative fragment holes account for 76.5% and non-penetrative impact

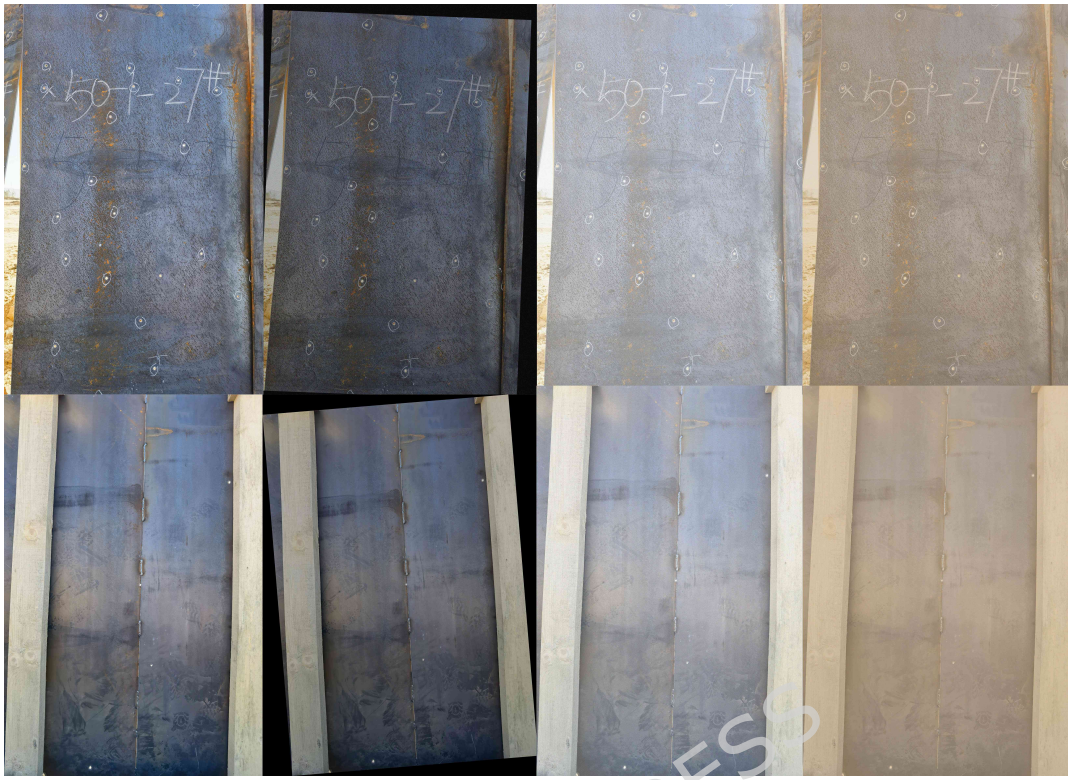

Figure 11. Target plate fragment detection dataset, including foggy and dusty weather scenarios.

marks account for 23.5%. Fig. 12(a) shows the spatial distribution of fragment impact centroids, while Fig. 12(b) illustrates the distribution of object sizes, indicating that the dataset is dominated by small-sized targets.

Following the definition in Krishna[44], an object is considered a small target if its bounding box occupies less than 1% of the image area. According to this criterion, 89.4% of the annotated fragment impacts in the dataset fall into the small-target category. Prior to training, all images were resized to a uniform resolution to reduce computational burden. The dataset was divided into a training set of 3195 images, a validation set of 913 images, and a test set of 457 images.

Design of Experiments

Experiments were conducted on a cloud platform provisioned with an Intel Core i9-13980 CPU, 32 GB of RAM, and an NVIDIA RTX 4090 GPU. The model was developed in PyTorch 2.0.0 framework, with CUDA 11.8 employed to accelerate training operations. During training, input images were resized to 640 × 640 pixels, and the model was trained for 250 epochs using the SGD optimizer with a batch size of 12. All other hyperparameters were kept consistent with those of the YOLOv11 baseline. During training, automatic mixed precision (AMP) was disabled. All inference speed measurements were conducted using FP32. precisionTable 1 provides the corresponding hyperparameter configurations.

We evaluate SFD-YOLO and comparative methods using four primary metrics: precision, recall, mean average precision (mAP), F1 score and frames per second (FPS). Precision and recall are defined as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{10}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{11}$$

where TP is the number of true positives, FP is the number of false positives, and FN is the number of false negatives. A detection is counted as a true positive if the Intersection over Union (IoU) between the predicted box and the ground truth exceeds a predefined threshold.

Average Precision (AP) quantifies the area under the precision–recall curve for each object category and reflects the overall balance between detection accuracy and recall. In the MS-COCO evaluation protocol, mean average precision (mAP) is computed by averaging AP values over multiple Intersection-over-Union (IoU) thresholds from 0.5 to 0.95, thereby evaluating detection performance under increasingly strict localization criteria.

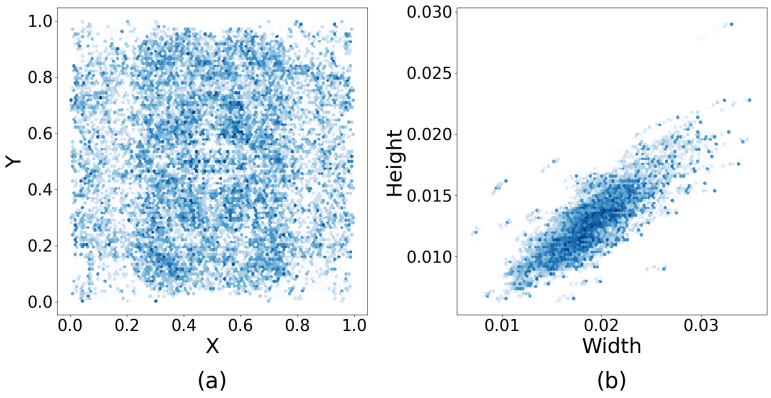

Figure 12. Visualization of dataset distribution. (a) Center Point Distribution. (b) Object Size Distribution.

Parameter name	**Setting**
Input image size	640×640
Epochs	250
Batch size	12
Workers	8
Initial learning rate (lr_0)	0.01
Optimizer	SGD
Momentum	0.937
Weight decay	0.0005
Learning rate schedule	Cosine annealing
Data augmentation	Mosaic

Table 1. Related training parameters.

In this work, we report AP at an IoU threshold of 0.5 (mAP@0.5) as well as mAP@[0.5:0.95] to characterize detection performance from both practical and stricter localization perspectives. For a detection task involving N object classes, the overall mAP is computed as the unweighted mean of the AP values across all classes:

$$\mathrm{AP}_i = \int_0^1 P_i(R_i)\,dR_i \tag{12}$$

$$\mathrm{mAP} = \frac{1}{N}\sum_{i=1}^{n} \mathrm{AP}_i \tag{13}$$

The F1 score combines precision and recall into a single metric by computing their harmonic mean:

$$F1 = \frac{2 \times \mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{14}$$

FPS denotes the average number of frames the model can process per second and depends on both the network's computational footprint and the hardware configuration. We also report Giga Floating-Point Operations (GFLOPs), which quantifies the total number of floating-point operations required for a single forward pass and serves as an indicator of model complexity.

In addition, we conducted a sensitivity analysis on the choice of optimizer. Specifically, SGD and AdamW were compared under identical training and inference settings on the proposed SFD-YOLO model. For AdamW, the learning rate and weight decay were adjusted to optimizer-appropriate values to ensure stable convergence, while all other configurations were kept unchanged. As shown in Table 2, SGD yields higher mAP and F1 scores on the target-plate dataset, whereas inference speed remains identical for both optimizers. Therefore, SGD was adopted as the default optimizer in this study.

Optimizer	$mAP_{0.5}$ (%)	$mAP_{0.5:0.95}$ (%)	F1 (%)	FPS
SGD	98.1	69.7	95.5	135
AdamW	96.7	68.3	93.0	135

Table 2. Sensitivity analysis of different optimizers on the proposed SFD-YOLO model.

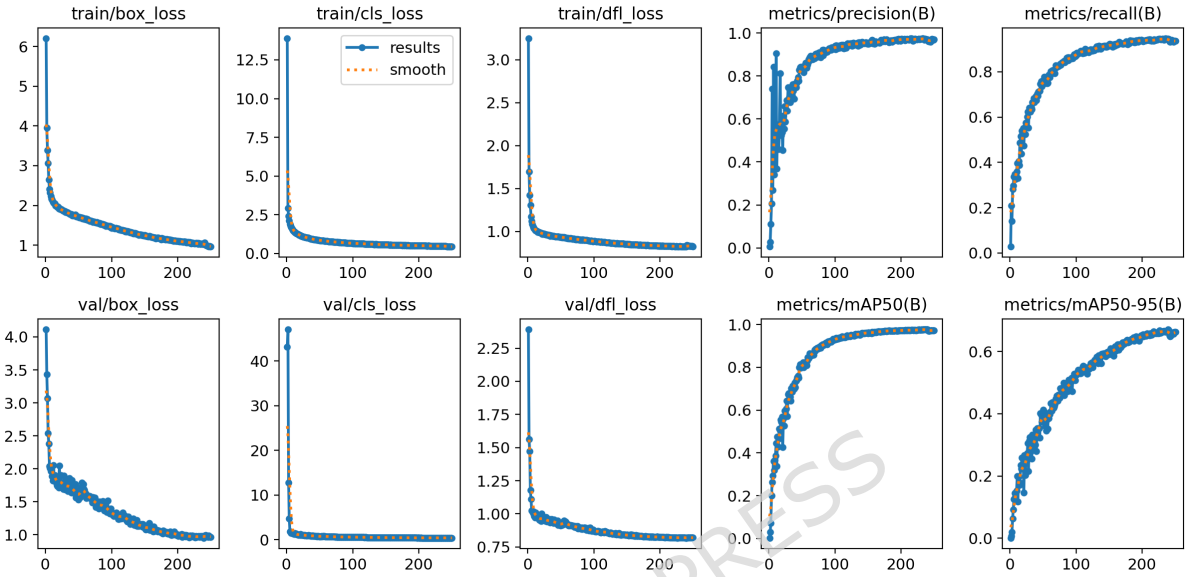

Figure 13. Model training results.

Result And Experimental Comparison

The model was trained on our custom dataset, and the training dynamics are illustrated in Fig. 13. Convergence was achieved within 250 epochs, with the training loss decreasing rapidly during the first 70 epochs. The loss curves for the training and validation sets followed similar trends and remained closely aligned, indicating stable learning. Precision, recall, and mAP gradually improved and tended to plateau between epochs 190 and 250, at which point the model reached optimal performance.

For comparative evaluation, SFD-YOLO was benchmarked against SSD, Faster RCNN, RT-DETRv2s, YOLOv5n, YOLOv8n, YOLOv10n, YOLOv11n, and CACS-YOLO[45]. We used mAP and FPS as the primary metrics to assess both detection accuracy and inference speed. Due to resource constraints, all methods were evaluated on the same augmented target-plate fragment dataset. The results are summarized in Table 3.

As shown in Table 3, the proposed SFD-YOLO achieves the best overall performance among all compared methods on the target-plate dataset. It attains the highest mAP@0.5 of 98.1% and mAP@[0.5:0.95] of 69.7%, together with a precision of 97.1%, a recall of 93.9%, and an F1 score of 95.5%. In terms of per-class performance, SFD-YOLO achieves an AP of 98.8% for penetrative fragment holes and 97.4% for non-penetrative impact marks, indicating balanced detection capability across different fragment types. Meanwhile, SFD-YOLO maintains a compact architecture with only 2.15 million parameters and 11.8 GFLOPs, while achieving the fastest inference speed of 135 FPS.

Compared with classical detectors such as SSD and Faster R-CNN, SFD-YOLO improves mAP@0.5 by 20.8% and 17.7%, respectively, while increasing inference speed from 45 FPS and 16 FPS to 135 FPS, using only a small fraction of their model parameters. Relative to the transformer-based RT-DETRv2s, SFD-YOLO achieves higher mAP@0.5 (98.1% vs. 95.4%) and mAP@[0.5:0.95] (69.7% vs. 61.8%), while reducing FLOPs from 60.0 G to 11.8 G and improving inference speed from 78 FPS to 135 FPS.

When compared with lightweight YOLO-series models, SFD-YOLO further refines the trade-off between accuracy and efficiency. It outperforms YOLOv11n by 2.7% in mAP@0.5 and 2.6% in mAP@[0.5:0.95], while reducing the parameter count from 2.59 M to 2.15 M and increasing inference speed from 126 FPS to 135 FPS. Compared with the small-object–oriented CACS-YOLO, SFD-YOLO achieves comparable or slightly higher mAP@0.5 (98.1% vs. 97.8%) and a higher F1 score (95.5% vs. 94.4%), while reducing model parameters by more than 25 M and lowering computational cost by approximately six times, demonstrating its suitability for real-time and resource-constrained target-plate testing scenarios.

Model	P (%)	R (%)	AP_{frag} (%)	AP_{non} (%)	$mAP_{0.5}$ (%)	$mAP_{0.5:0.95}$ (%)	F1 (%)	Params (M)	FLOPs (G)	FPS
SSD	82.5	59.8	87.4	67.2	77.3	47.3	69.3	26.35	35.2	45
Faster-RCNN	78.2	75.1	88.2	72.6	80.4	52.6	76.6	134.62	180.5	16
RT-DETRv2s	94.3	91.4	96.5	94.3	95.4	61.8	92.8	20.43	60.0	78
YOLOv5n	91.2	90.1	94.9	89.7	92.3	57.3	90.6	2.19	4.5	121
YOLOv8n	94.9	92.4	97.0	94.6	95.8	59.5	93.6	3.21	8.7	131
YOLOv10n	95.9	93.5	96.4	95.8	96.1	64.8	94.7	2.71	6.7	106
YOLOv11n	94.8	92.7	97.2	93.6	95.4	62.9	93.7	2.59	6.5	126
CACS-YOLO	96.7	92.3	98.1	**97.5**	97.8	66.9	94.4	27.44	66.6	101
SFD-YOLO	**97.1**	**93.9**	**98.8**	97.4	**98.1**	**69.7**	**95.5**	**2.15**	**11.8**	**135**

Table 3. Performance comparison of different detection models on the target-plate dataset. AP_{frag} and AP_{non} denote per-class average precision for penetrative fragment holes and non-penetrative impact marks, respectively.

Three images were randomly selected and tested using different detection algorithms for comparative analysis, as shown in Fig. 14. In the first image, SSD, Faster-RCNN, RT-DETRv2s, YOLOv8n, YOLOv10n, and YOLOv11n all exhibited missed detections. Additionally, SSD and Faster-RCNN showed low confidence scores, while YOLOv5 and CACS-YOLO produced false detections. Specifically, YOLOv5 failed to effectively distinguish between penetration and non-penetration holes, and CACS-YOLO mistakenly identified parts of the background as holes. In contrast, the proposed SFD-YOLO successfully detected all target holes, correctly classified penetration and non-penetration types, and achieved the highest detection confidence.

In the second image, SSD, Faster-RCNN, RT-DETRv2s, and YOLOv5 all missed non-penetration fragment holes, while YOLOv10n, YOLOv11n, and CACS-YOLO showed duplicate detections. SFD-YOLO, however, avoided all of these issues.

In the third image, SFD-YOLO again achieved the best detection performance. Although CACS-YOLO detected all fragment holes, it incorrectly classified a background region as a non-penetration hole, while other algorithms still exhibited missed detections. Overall, these results clearly demonstrate that SFD-YOLO not only achieves state-of-the-art accuracy but also maintains real-time processing capability, providing a more practical and efficient solution for fragment-hole detection in warhead target-plate testing.

To qualitatively evaluate the robustness of the proposed method against common imaging degradations encountered in field acquisition, we generated multiple degraded variants from the same clean test image, including motion blur, contrast variation, and illumination change. Specifically, we synthesized motion blur using a linear kernel with size $k = 15$ and rotation angle $\theta = 45°$; high-contrast enhancement using a linear transformation $I' = \alpha I$ with $\alpha = 1.4$; low-contrast degradation using $I' = (I - 128) \cdot f + 128$ with $f = 0.6$; overexposure by increasing brightness with $I' = I + \Delta$ and $\Delta = +40$ and low-light conditions simulated by gamma-based darkening ($\gamma = 1.8$), followed by a global brightness reduction ($\Delta = -30$) and additive Gaussian noise ($\sigma = 5$) to mimic sensor noise under insufficient lighting. All variants were applied consistently to both YOLO11 and SFD-YOLO under identical inference settings.

Fig. 15 provides a qualitative robustness comparison under blur, contrast variation, and illumination changes. Overall, SFD-YOLO produces more consistent detections than the YOLO11 baseline, exhibiting generally higher confidence scores and clearer localization across the tested degradations. The improvement is particularly evident for the non-penetrative impact marks, which are visually subtle and therefore more sensitive to image quality degradation. Under blur, high-contrast, bright and dark conditions, YOLO11 shows less stable predictions for the non-penetrative class, including occasional misclassifications or spurious detections, whereas SFD-YOLO maintains correct detections with relatively high confidence. These results suggest that the proposed architecture enhances feature robustness to common imaging disturbances and improves the reliability of small and weak-texture target recognition in practical scenarios.

In addition to the standard evaluation on clear images, we further investigate the robustness of the proposed detector under adverse weather conditions. Due to safety constraints and practical limitations in warhead testing environments, collecting a sufficiently large amount of real fog- or dust-degraded target-plate images is difficult. Therefore, we generate synthetic fog and dust degradations with multiple severity levels and evaluate all methods under identical inference settings.

Specifically, the fog effect is simulated by blending the input image with a white haze layer, while dust is generated by

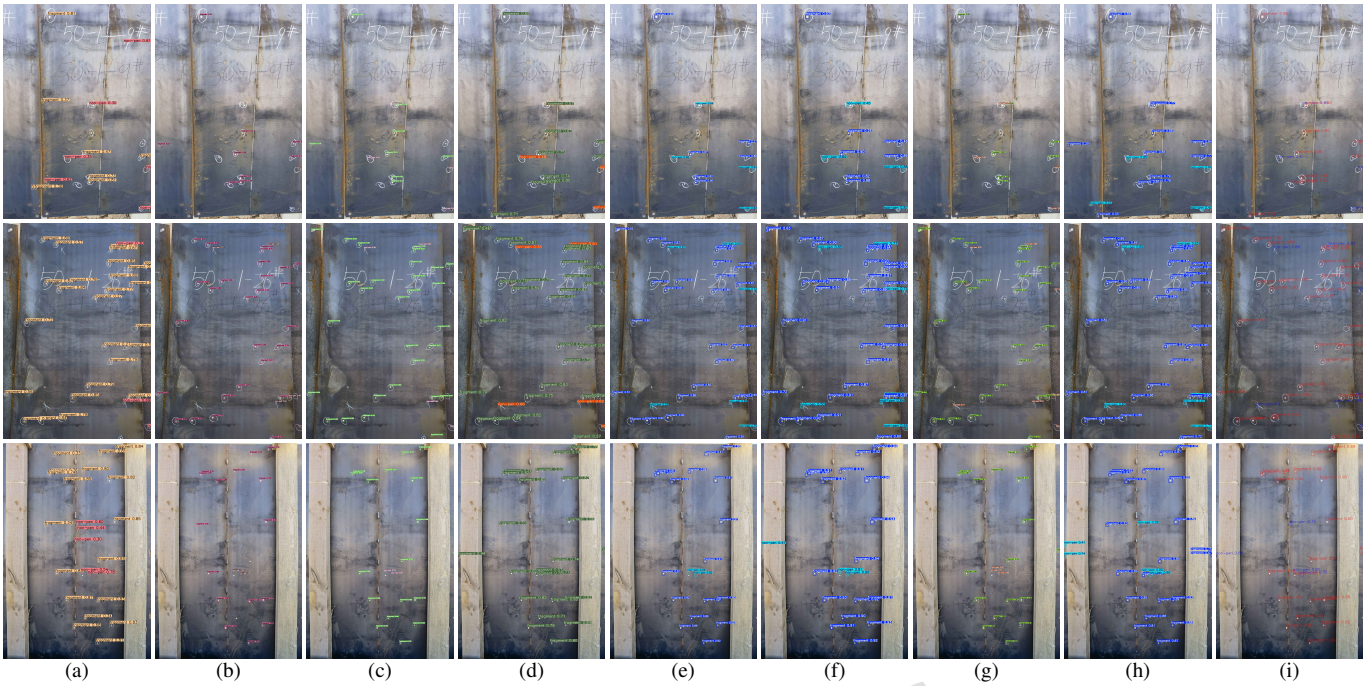

Figure 14. Visualization comparison of detection results using different algorithms. (a) SSD. (b) Faster-RCNN. (c) RT-DETRv2s. (d) YOLOv5n. (e) YOLOv8n. (f) YOLOv10n. (g) YOLOv11n. (h) CACS-YOLO. (i) SFD-YOLO.

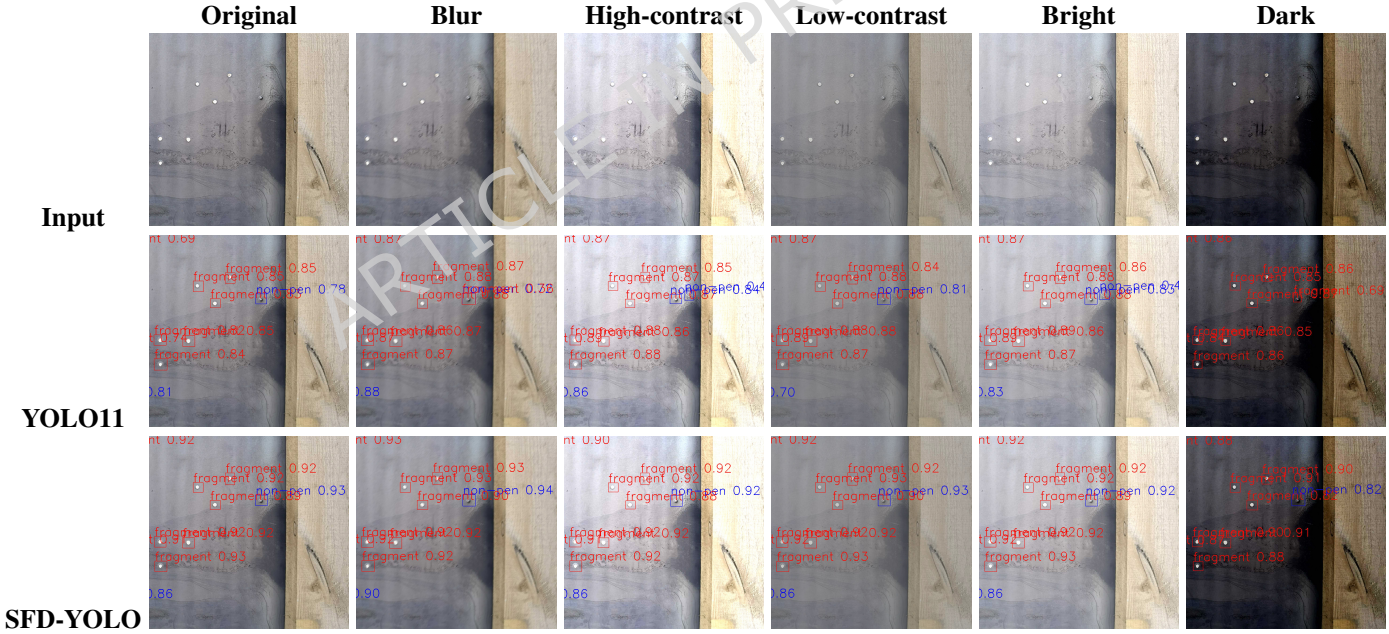

Figure 15. Qualitative robustness evaluation under blur, contrast variation, and illumination changes.

blending with a brown-toned layer. Gaussian noise is then added to mimic random particle scattering. Three degradation severity levels (light, moderate, and heavy) are considered for both fog and dust. The severity is controlled by the blending opacity (0.2–0.6) and the noise magnitude (Gaussian noise scale 5–15). These degraded samples are applied only to the test set to provide a stress-test evaluation of model robustness.

Fig. 16 presents a qualitative robustness comparison of YOLO11 and the proposed SFD-YOLO under synthetic fog and dust degradations with increasing severity. As the degradation level increases from light to heavy, YOLO11 exhibits more unstable predictions, especially for the non-penetration category, where missed detections and lower confidence scores become

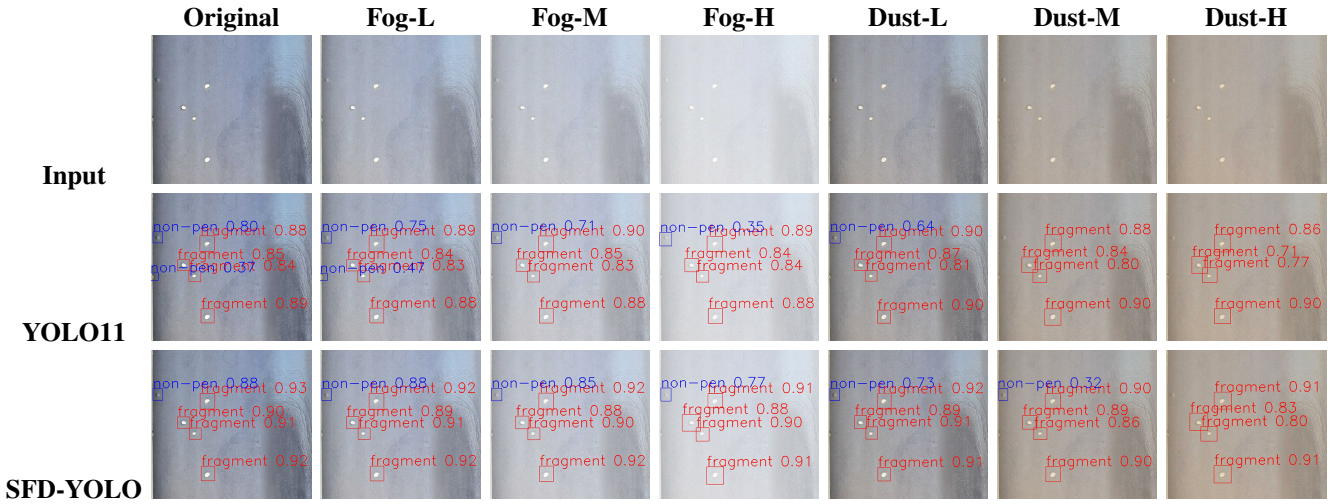

Figure 16. Qualitative comparison of detection results under synthetic dust and fog degradations.

more frequent. In contrast, SFD-YOLO maintains more consistent detection results across all conditions, producing tighter bounding boxes and generally higher confidence scores. Notably, even under moderate-to-heavy fog and dust, SFD-YOLO still preserves reliable recognition of non-penetration traces, indicating stronger feature robustness against haze-like blur and low-contrast disturbances. These observations align with the quantitative results reported in the following tables and robustness degradation curves.

Model	Condition	AP_{frag} (%)	AP_{non} (%)	$mAP_{0.5}$ (%)	$mAP_{0.5:0.95}$ (%)	F1 (%)
YOLOv11n	Clean	97.2	93.6	95.4	62.9	93.7
	Fog-L	96.1	90.0	93.1	61.1	90.2
	Fog-M	93.5	87.5	90.5	58.3	87.4
	Fog-H	90.0	83.7	86.9	53.6	82.2
	Dust-L	93.8	86.6	90.2	59.8	86.1
	Dust-M	90.2	77.2	83.7	55.5	82.1
	Dust-H	86.1	65.1	75.6	49.8	77.2
SFD-YOLO	Clean	98.8	97.4	98.1	69.7	95.5
	Fog-L	98.0	95.0	96.5	68.9	94.4
	Fog-M	96.5	91.5	94.0	64.5	90.6
	Fog-H	91.6	88.9	90.3	61.2	86.0
	Dust-L	96.4	90.8	93.6	67.0	92.2
	Dust-M	94.6	84.5	89.6	62.8	86.6
	Dust-H	90.4	72.1	81.3	58.1	80.5

Table 4. Quantitative robustness comparison under synthetic fog and dust degradations with different severity levels.

Table 4 reports the quantitative robustness results of YOLO11n and the proposed SFD-YOLO under synthetic fog and dust degradations with three severity levels (L/M/H). Overall, SFD-YOLO consistently outperforms YOLO11n across all conditions and exhibits a slower degradation trend as the interference becomes stronger.

Under fog degradation, YOLO11n decreases from 95.4% mAP@0.5 and 93.7% F1 on clean images to 86.9% and 82.2% at Fog-H. In contrast, SFD-YOLO maintains higher performance, dropping from 98.1% and 95.5% to 90.3% and 86.0%, respectively. Notably, SFD-YOLO achieves a clear advantage on the stricter mAP@0.5:0.95 metric, remaining at 61.2% under Fog-H compared with 53.6% for YOLO11n, indicating better localization stability under blur-like scattering effects.

Dust degradation causes more severe performance drops for both models, especially at high intensity. YOLO11n falls sharply to 75.6% mAP@0.5 and 77.2% F1 at Dust-H, while SFD-YOLO preserves 81.3% and 80.5%, respectively. The

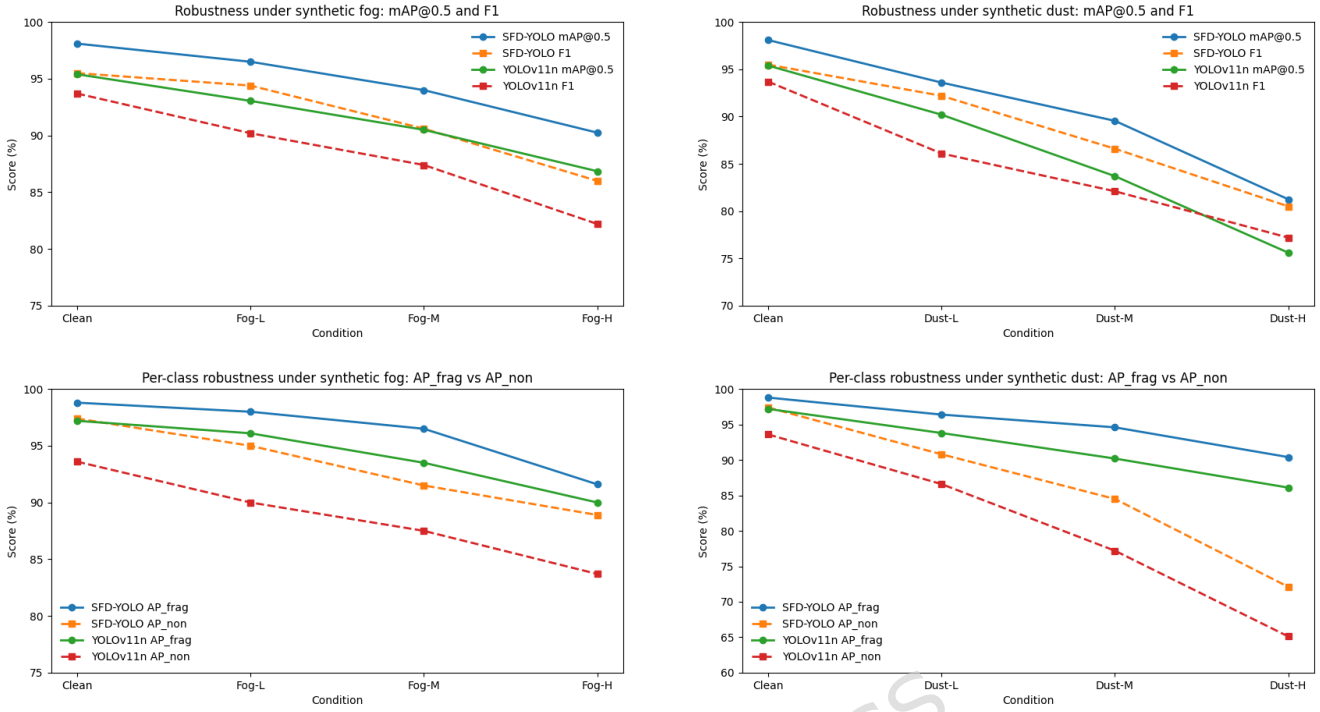

Figure 17. Performance degradation curves of YOLO11n and SFD-YOLO under synthetic fog and dust with three severity levels (L/M/H).

per-class results further reveal that the non-penetration category is more sensitive to adverse conditions. For example, AP_{non} of YOLO11n drops from 93.6% (clean) to 65.1% (Dust-H), whereas SFD-YOLO maintains a higher AP_{non} of 72.1%. These results demonstrate that SFD-YOLO improves robustness under both atmospheric scattering and particulate occlusion, particularly for the challenging non-penetration targets.

Fig. 17 shows the robustness comparison under synthetic fog and dust with three severity levels. SFD-YOLO achieves consistently higher mAP@0.5 and F1 than YOLO11n and degrades more slowly as the interference becomes stronger. Notably, under Dust-H, YOLO11n drops to 75.6% mAP@0.5, while SFD-YOLO still maintains 81.3%. The per-class results further indicate that AP_{non} is more sensitive to adverse conditions than AP_{frag}, and SFD-YOLO preserves higher AP_{non} under both fog and dust.

Ablation Study

We conducted an ablation study to quantify the individual and combined contributions of the proposed modules. In the YOLOv11-A variant, the SCC3k2 module is integrated into the YOLOv11 backbone; YOLOv11-B replaces the standard convolutions in the neck with the LAE module; and YOLOv11-C adds a P2 small-object detection head along with the AFPN for hierarchical feature fusion. To further evaluate the combined effects of these components, two additional configurations are tested: YOLOv11-D (SCC3k2 + LAE) and YOLOv11-E (SCC3k2 + LAE + AFPN head). The ablation results are summarized in Table 5

Table 5 reports the ablation experiments conducted to evaluate the contribution of each module in the proposed SFD-YOLO framework. The baseline YOLOv11 achieves a precision of 94.8%, recall of 92.7%, mAP@0.5 of 95.4%, and mAP@[0.5:0.95] of 62.9%, with an F1 score of 93.7% and 2.59M parameters. When the SCC3k2 module is introduced, the mAP@0.5 increases from 95.4% to 96.1% and mAP@[0.5:0.95] improves from 62.9% to 66.5%, while the parameter count is slightly reduced to 2.46M. This indicates that SCC3k2 effectively enhances spatial and channel feature representation by suppressing redundant information, leading to improved detection performance without increasing model complexity. Applying the LAE module alone significantly reduces the parameter count from 2.59M to 2.10M, confirming its effectiveness in lightweight feature extraction. However, this configuration results in a decrease in mAP@0.5 to 92.7% and mAP@[0.5:0.95] to 58.7%, suggesting that excessive lightweighting without sufficient high-level semantic support may weaken discriminative feature representation for small fragment holes. By contrast, introducing the AFPN head yields a substantial performance gain, with precision reaching 97.5%, mAP@0.5 increasing to 98.0%, and mAP@[0.5:0.95] rising to 68.6%, at the cost of a moderate increase in parameters.

Model	SCC3k2	LAE	AFPN head	P (%)	R (%)	$mAP_{0.5}$ (%)	$mAP_{0.5:0.95}$ (%)	F1 (%)	Params (M)
YOLOv11	×	×	×	94.8	92.7	95.4	62.9	93.7	2.59
YOLOv11-A	✓	×	×	95.2	92.4	96.1	66.5	93.8	2.46
YOLOv11-B	×	✓	×	91.3	90.6	92.7	58.7	90.9	2.10
YOLOv11-C	×	×	✓	97.5	92.3	98.0	68.6	94.8	2.71
YOLOv11-D	✓	✓	×	94.8	92.9	95.6	64.3	93.8	2.13
YOLOv11-E	✓	✓	✓	**97.1**	**93.9**	**98.1**	**69.7**	**95.5**	**2.15**

Table 5. Ablation study of individual modules in the proposed SFD-YOLO framework.

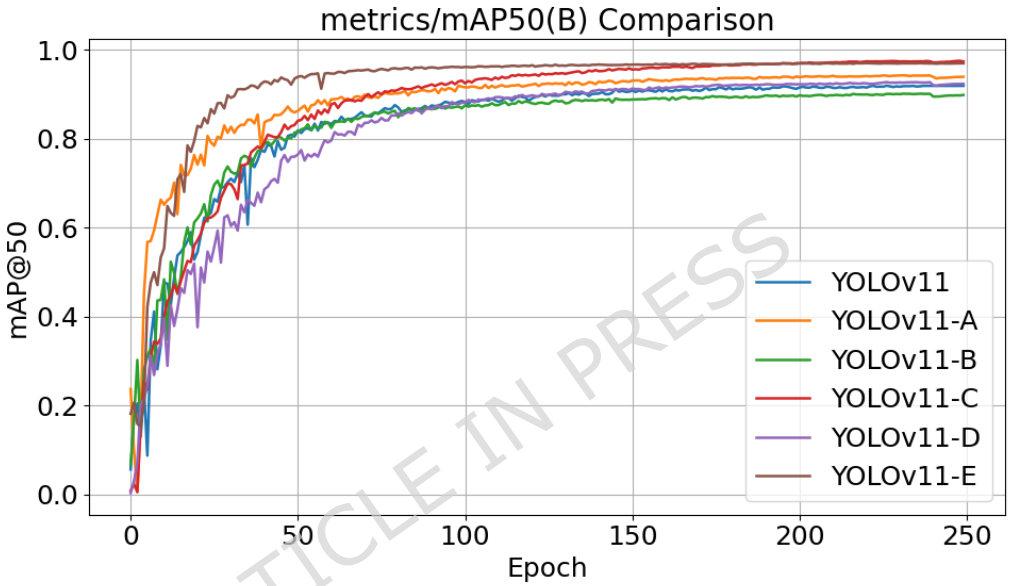

Figure 18. Ablation experiments of each module in SFD-YOLO.

This demonstrates that AFPN effectively strengthens multi-scale feature fusion and improves localization robustness for small and densely distributed targets. When SCC3k2 and LAE are combined, the model achieves a balanced improvement in accuracy and efficiency, with mAP@0.5 of 95.6%, mAP@[0.5:0.95] of 64.3%, and a reduced parameter count of 2.13M. Finally, the full configuration (YOLOv11-E), integrating SCC3k2, LAE, and AFPN, delivers the best overall performance, achieving 97.1% precision, 93.9% recall, 98.1% mAP@0.5, 69.7% mAP@[0.5:0.95], and an F1 score of 95.5%, while maintaining a compact model size of only 2.15M parameters.

These results demonstrate that SCC3k2, LAE, and AFPN play complementary roles within the SFD-YOLO architecture: SCC3k2 enhances feature representation, LAE improves computational efficiency, and AFPN strengthens multi-scale feature fusion. Their joint integration enables SFD-YOLO to achieve both high detection accuracy and lightweight design, which is critical for real-time fragment-hole detection on target plates.

Fig. 18 illustrates the mAP variation during training for different model variants. As shown, the baseline YOLOv11 converges steadily but reaches a lower final accuracy. Incorporating the SCC3k2 module (YOLOv11-A) significantly accelerates convergence and improves the final mAP, indicating enhanced feature representation through spatial and channel reconstruction. Replacing standard convolutions with the LAE module (YOLOv11-B) provides a moderate performance gain while reducing parameters, reflecting its efficiency in lightweight feature extraction. When the AFPN and small-object detection head are added (YOLOv11-C), the model achieves faster and more stable convergence, demonstrating the benefits of hierarchical feature fusion. The full SFD-YOLO configuration (YOLOv11-E) achieves the highest and most stable mAP across epochs, confirming that the combined modules effectively complement each other and jointly enhance detection accuracy and convergence stability.

Discussion

Despite the improved accuracy and robustness of SFD-YOLO, several limitations remain. First, the dataset was collected in outdoor static explosion test sites under relatively stable acquisition settings. Although synthetic blur, illumination, contrast changes, and fog, dust degradations were used for stress testing, these perturbations cannot fully represent real adverse environments, such as strong specular glare on metallic surfaces, spatially non-uniform scattering, and camera-dependent noise. Second, while the added P2 detection head enhances sensitivity to micro-scale impacts, the performance may still degrade under extremely severe blur or low-light conditions with heavy noise, where non-penetrative marks become difficult to distinguish and missed detections may occur.

Future work will focus on improving generalization in practical scenarios. We plan to collect more real adverse-weather samples (e.g., dust/smoke after detonation and strong backlighting conditions) to validate robustness beyond synthetic effects. In addition, degradation-aware training and feature-level robustness enhancement will be explored to maintain stable detection under extreme imaging conditions, without sacrificing real-time efficiency.

Conclusion

This paper presents SFD-YOLO, an enhanced YOLOv11-based framework designed to address missed detections and low throughput in small fragment impact detection on target plates during warhead testing. To construct a high-quality training corpus, multiple static-blast experiments were conducted to establish a target-plate dataset covering diverse scenes and fragment characteristics. Feature analysis indicates that accurate detection of small objects is critical for overall performance. To improve feature representation for such challenging cases, we introduce SCC3k2 into the backbone to strengthen spatial consistency and channel discrimination. Moreover, an additional micro-object detection branch is incorporated, forming a four-head detection scheme combined with an AFPN-based fusion strategy to better preserve fine-grained cues across scales. Lightweight Adaptive Extraction (LAE) is selectively integrated into the network to reduce redundancy and enhance efficiency.

Experimental results demonstrate that SFD-YOLO achieves 98.1% mAP@0.5 and 69.7% mAP@0.5:0.95, outperforming the YOLOv11 baseline by 2.7% in mAP@0.5, while maintaining 135 FPS with only 2.15M parameters. These results confirm that the proposed architecture provides a favorable trade-off between accuracy and computational cost for real-time fragment-hole inspection. Future work will focus on strengthening robustness under more challenging acquisition conditions, further lightweight deployment, and extending the framework to other industrial surface inspection tasks.

Data availability

The target-plate fragment-hole dataset established in this study and the associated experimental data are available from the corresponding author upon reasonable request, subject to applicable safety and confidentiality constraints.

Code availability

The source code and scripts used to support the findings of this study are available from the corresponding author (or the first author) upon reasonable request.

References

1. Elveli, B. S. *et al.* Thin steel plates exposed to combined ballistic impact and partially confined airblast loading. *Eng. Fail. Analysis* **144**, 106943, DOI: https://doi.org/10.1016/j.engfailanal.2022.106943 (2023).
2. Li, H., Zhang, X., Zhang, X. & Gao, J. Warhead fragments' distribution measurement method by a multiscreen sensors' intersection test mechanism and equivalent target damage probability calculation. *IEEE Transactions on Instrumentation Meas.* **70**, 1–9, DOI: 10.1109/TIM.2021.3079557 (2021).
3. Gao, J. & Zhang, X. Fragment perforation spatial localization measurement method and calculation analysis by using photogrammetry technology. *IEEE Access* **11**, 102092–102102, DOI: 10.1109/ACCESS.2023.3312014 (2023).
4. Guildenbecher, D. R. *et al.* 3d optical diagnostics for explosively driven deformation and fragmentation. *Int. J. Impact Eng.* **162**, 104142, DOI: https://doi.org/10.1016/j.ijimpeng.2021.104142 (2022).
5. Chen, C., Shi, Q., feng You, Z., yu Ge, H. & Zhang, F. A method to determine the shell layout scheme for equipment battlefield damage tests under artillery fire. *Def. Technol.* **17**, 682–691, DOI: https://doi.org/10.1016/j.dt.2020.02.019 (2021).
6. Felix, D., Colwill, I. & Harris, P. Real-time simulation of a fragmenting explosion for cylindrical warheads. *The J. Def. Model. Simul.* **19**, 783–797, DOI: 10.1177/1548512921995560 (2022).

7. Zhang, G. *et al.* A novel detection method for warhead fragment targets in optical images under dynamic strong interference environments. *Def. Technol.* **43**, 252–270, DOI: https://doi.org/10.1016/j.dt.2024.08.008 (2025).

8. Hussain, M. Yolo-v1 to yolo-v8, the rise of yolo and its complementary nature toward digital manufacturing and industrial defect detection. *Machines* **11**, DOI: 10.3390/machines11070677 (2023).

9. Dong, Y., Yang, L., Jin, Z. & Wu, L. Experimental and numerical analysis of ballistic impact response of fiber-reinforced composite/metal composite target. *Compos. Struct.* **294**, 115776, DOI: https://doi.org/10.1016/j.compstruct.2022.115776 (2022).

10. Li, Y. Research and application of deep learning in image recognition. In *2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA)*, 994–999, DOI: 10.1109/ICPECA53709.2022.9718847 (2022).

11. Zhang, G. *et al.* A novel detection method for warhead fragment targets in optical images under dynamic strong interference environments. *Def. Technol.* **43**, 252–270, DOI: https://doi.org/10.1016/j.dt.2024.08.008 (2025).

12. Song, W. *et al.* Accurate extraction of fragmented field boundaries using classification-assisted and cnn-based semantic segmentation methods. In *2023 11th International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*, 1–4, DOI: 10.1109/Agro-Geoinformatics59224.2023.10233358 (2023).

13. Rusyn, B., Lutsyk, O., Kosarevych, R., Vysotska, V. & Lytvynenko, V. Features extraction from multi-spectral remote sensing images based on multi-threshold binarization. *Sci. Reports* **13**, 19655, DOI: 10.1038/s41598-023-46785-7 (2023).

14. Zhang, Y., Zhao, H., Li, X., Li, H. & Jin, J. Mers-net: A lightweight and efficient remote sensing image object detector. *IEEE Geosci. Remote. Sens. Lett.* 1–1, DOI: 10.1109/LGRS.2025.3569672 (2025).

15. Ghorai, S., Mukherjee, A., Gangadaran, M. & Dutta, P. K. Automatic defect detection on hot-rolled flat steel products. *IEEE Transactions on Instrumentation Meas.* **62**, 612–621, DOI: 10.1109/TIM.2012.2218677 (2013).

16. Li, R. & Shen, Y. Yolosr-ist: A deep learning method for small target detection in infrared remote sensing images based on super-resolution and yolo. *SIGNAL PROCESSING* **208**, DOI: 10.1016/j.sigpro.2023.108962 (2023).

17. He, M., Qin, L., Deng, X. & Liu, K. Mfi-yolo: Multi-fault insulator detection based on an improved yolov8. *IEEE TRANSACTIONS ON POWER DELIVERY* **39**, 168–179, DOI: 10.1109/TPWRD.2023.3328178 (2024).

18. Safaldin, M., Zaghden, N. & Mejdoub, M. An improved yolov8 to detect moving objects. *IEEE ACCESS* **12**, 59782–59806, DOI: 10.1109/ACCESS.2024.3393835 (2024).

19. Zhao, L., Zhi, L., Zhao, C. & Zheng, W. Fire-yolo: A small target object detection method for fire inspection. *SUSTAINABILITY* **14**, DOI: 10.3390/su14094930 (2022).

20. Zou, Z., Chen, K., Shi, Z., Guo, Y. & Ye, J. Object detection in 20 years: A survey. *PROCEEDINGS OF THE IEEE* **111**, 257–276, DOI: 10.1109/JPROC.2023.3238524 (2023).

21. Yao, Y. *et al.* On improving bounding box representations for oriented object detection. *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING* **61**, DOI: 10.1109/TGRS.2022.3231340 (2023).

22. Jiao, L. *et al.* A survey of deep learning-based object detection. *IEEE ACCESS* **7**, 128837–128868, DOI: 10.1109/ACCESS.2019.2939201 (2019).

23. Girshick, R., Donahue, J., Darrell, T. & Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 580–587, DOI: 10.1109/CVPR.2014.81 (2014).

24. Girshick, R. Fast r-cnn. In *2015 IEEE International Conference on Computer Vision (ICCV)*, 1440–1448, DOI: 10.1109/ICCV.2015.169 (2015).

25. Ren, S., He, K., Girshick, R. & Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis Mach. Intell.* **39**, 1137–1149, DOI: 10.1109/TPAMI.2016.2577031 (2017).

26. Liu, W. *et al.* Ssd: Single shot multibox detector. In Leibe, B., Matas, J., Sebe, N. & Welling, M. (eds.) *Computer Vision – ECCV 2016*, 21–37 (Springer International Publishing, Cham, 2016).

27. Duan, K. *et al.* Centernet: Keypoint triplets for object detection. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 6568–6577, DOI: 10.1109/ICCV.2019.00667 (2019).

28. Carion, N. *et al.* End-to-end object detection with transformers. In Vedaldi, A., Bischof, H., Brox, T. & Frahm, J.-M. (eds.) *Computer Vision – ECCV 2020*, 213–229 (Springer International Publishing, Cham, 2020).

29. Zhang, J., Sun, H. & Li, C. Study on the detection method of broken piece perforation and crater based on hyperbolic threshold segmentation. In *Second IYSF Academic Symposium on Artificial Intelligence and Computer Engineering*, vol. 12079, 556–560 (SPIE, 2021).

30. Lei, J., Wang, Z. & Li, J. Research on target detection of fragment image based on faster r-cnn. *Foreign Electron. Meas. Technol.* **40**, 70–74, DOI: 10.19652/j.cnki.femt.2002308 (2021).

31. Wei, Q., He, Z., Wang, Y. & Wang, J. Research on fragment identification of a static explosion field based on densenet and attention mechanism. *J. Ordnance Equip. Eng.* **44**, 259–265 (2023).

32. He, M., Wu, J., Liang, C., Hu, P. & Ren, Y. Few-shot warhead fragment group object detection based on feature reassembly and attention. *Opt. Precis. Eng.* **32**, 1929–1940 (2024).

33. Gao, J., Han, H. & Li, H. A semantic segmentation and distortion correction of fragment perforation small target. *IEEE Access* **12**, 82769–82781, DOI: 10.1109/ACCESS.2024.3412796 (2024).

34. Zhang, C. *et al.* Aggregation connection network for tiny face detection. In *2019 International Joint Conference on Neural Networks (IJCNN)*, 1–8, DOI: 10.1109/IJCNN.2019.8852083 (2019).

35. Tang, X., Du, D. K., He, Z. & Liu, J. Pyramidbox: A context-assisted single shot face detector. In Ferrari, V., Hebert, M., Sminchisescu, C. & Weiss, Y. (eds.) *Computer Vision – ECCV 2018*, 812–828 (Springer International Publishing, Cham, 2018).

36. Guo, C., Fan, B., Zhang, Q., Xiang, S. & Pan, C. Augfpn: Improving multi-scale feature learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020).

37. Zhang, Y. *et al.* Ffca-yolo for small object detection in remote sensing images. *IEEE Transactions on Geosci. Remote. Sens.* **62**, 1–15, DOI: 10.1109/TGRS.2024.3363057 (2024).

38. Chen, Y. *et al.* Dynamic scale training for object detection (2021). 2004.12432.

39. Yu, X., Gong, Y., Jiang, N., Ye, Q. & Han, Z. Scale match for tiny person detection. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1246–1254, DOI: 10.1109/WACV45572.2020.9093394 (2020).

40. Yu, Z. *et al.* Lsm-yolo: A compact and effective roi detector for medical detection. *arXiv preprint arXiv:2408.14087* (2024).

41. Yang, G., Lei, J., Tian, H., Feng, Z. & Liang, R. Asymptotic feature pyramid network for labeling pixels and regions. *IEEE Transactions on Circuits Syst. for Video Technol.* **34**, 7820–7829, DOI: 10.1109/TCSVT.2024.3376773 (2024).

42. Gao, J. & Zhang, X. Fragment perforation spatial localization measurement method and calculation analysis by using photogrammetry technology. *IEEE Access* **11**, 102092–102102, DOI: 10.1109/ACCESS.2023.3312014 (2023).

43. Maruschak, P. *et al.* Surface defects of rolled metal products recognised by a deep neural network under different illuminance levels and low-amplitude vibration. *Int. J. Adv. Manuf. Technol.* DOI: 10.1007/s00170-025-15942-9 (2025).

44. Krishna, H. & Jawahar, C. Improving small object detection. In *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)*, 340–345, DOI: 10.1109/ACPR.2017.149 (2017).

45. Cao, Z., Chen, K., Chen, J., Chen, Z. & Zhang, M. Cacs-yolo: A lightweight model for insulator defect detection based on improved yolov8m. *IEEE Transactions on Instrumentation Meas.* **73**, 1–10, DOI: 10.1109/TIM.2024.3453332 (2024).

Acknowledgements

The authors would like to thank all colleagues who provided helpful discussions and technical support.

Author contributions

H.L. conceived the study, developed the SFD-YOLO method, implemented the algorithms, and performed model training and data analysis. Y.D. contributed to dataset construction, experiment coordination, and validation of detection results. W.Y. supervised the research, provided methodological guidance, and oversaw project progress. Y.L. assisted with investigation, data curation, and performance evaluation. H.L. drafted the manuscript, and all authors revised and approved the final version.

Funding

This research was supported by the National Natural Science Foundation of China grant number 61701445, the Open Research Foundation of the Key Laboratory of North University of China grant number DXMBJJ202007, and the "Fundamental Research Program" of Shanxi Province grant number 20210302124200.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondences and requests for materials should be addressed to W.Y.