

Article

WID-YOLO11: Weld Surface Defect Detection via Frequency-Domain Feature Decomposition and Multi-Scale Dilated Attention

Zhiyuan Li ¹, Songsong Li ^{1,*}, Weining Li ¹, Meide Liu ¹, Shidong Li ¹ and Xiaoming Chen ^{2,*}

¹ School of Information Engineering, Dalian Ocean University, Dalian 116023, China

² School of Optoelectronic Engineering and Instrumentation Science, Dalian University of Technology, Dalian 116024, China

* Correspondence: lisongsong@dlou.edu.cn (S.L.); chen_xm@dlut.edu.cn (X.C.)

Abstract

Weld defect detection is challenged by significant variations in defect scale, complex defect morphology, and strong background interference. To address these issues, this study proposes WID-YOLO11, an improved detection model based on YOLO11n. First, a novel C3k2_LFA-WTConv module is developed to decompose feature maps into frequency subbands via learnable wavelet filters and adaptively reweight them through a Learnable Frequency-Attention (LFA) mechanism, enabling the network to learn task-adapted subband representations for different defect types, where each class exhibits distinct wavelet subband energy distributions. Second, an Improved Multi-Scale Dilated Local Attention (IMSDA) module is constructed by extending the dilation-rate set of the original MSDA from three to four values (1, 2, 3, and 4), expanding the maximum equivalent receptive field from 7×7 to 9×9 to improve multi-scale sensitivity to medium-scale defects such as porosity and spatter at low computational overhead. Finally, a DySample dynamic upsampling module is integrated into the neck network to adaptively learn sampling locations and preserve defect boundary details more effectively during multi-scale feature fusion. Evaluations on a publicly available weld defect dataset of 986 images show that WID-YOLO11 outperforms the YOLO11n baseline by 4.5, 10.4, and 3.4 percentage points in mAP@0.5, Recall, and mAP@0.5:0.95, respectively, confirming the effectiveness of the three proposed enhancements.

Keywords: weld defect detection; YOLO11n; wavelet transform convolution; dynamic upsampling; multi-scale dilated local attention

Academic Editors: Muhammad Farooq Siddique and Izaz Raouf

Received: 15 June 2026

Revised: 7 July 2026

Accepted: 8 July 2026

Published: 10 July 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Welding, a process that joins metal components through heating and pressure, is widely used in aerospace, automotive, shipbuilding, construction, energy, and machinery manufacturing [1], and is regarded as one of the most critical joining technologies in modern industry. As the load-bearing region connecting metallic components, weld quality directly governs the strength, stability, and service life of the overall structure. During welding, however, factors such as thermal stress, cooling rate, process parameters, and environmental conditions frequently give rise to defects including cracks, porosity, and spatter [2], which can substantially reduce joint strength and may ultimately lead to struc-

tural failure. Detecting weld surface defects during production is therefore of significant practical and research importance. Traditional inspection approaches such as radiographic testing [3] and ultrasonic testing [4] can reveal internal discontinuities, but they rely heavily on manual operation, face challenges in automated defect interpretation, and are difficult to deploy as real-time inline systems on production lines.

The increasing maturity of deep learning [5], especially the widespread deployment of CNNs [6], has opened new avenues for automated weld defect detection. By learning multi-level feature hierarchies directly from data in an end-to-end fashion [7], deep models can build discriminative representations without hand-crafted features, yielding marked gains in both detection accuracy and generalization. Within this field, object detectors are conventionally grouped into two families: two-stage pipelines such as Faster R-CNN [8] that first propose candidate regions and then refine their classification, and single-stage detectors typified by the YOLO series [9] that jointly localize and classify objects in one unified forward pass. Single-stage architectures are therefore more amenable to real-time industrial deployment owing to their lower inference latency and compact design. Within this framework, a number of YOLO-based weld defect detection methods have been proposed and are reviewed in the following section. However, weld defect inspection presents unique challenges that generic architectures are not specifically designed to address. Different defect types exhibit distinct subband energy distributions in the wavelet domain: crack regions concentrate the most energy in the low-frequency approximation (LL) subband, spatter exhibits comparatively higher contributions from the horizontal and vertical detail subbands (LH/HL), and porosity occupies an intermediate position. These class-conditioned differences motivate a design that decomposes feature maps into subbands and learns subband-specific attention weights. Consequently, attention mechanisms that operate solely in the spatial domain are unable to disentangle these frequency-domain differences, making it difficult to simultaneously enhance the discriminative features of all three defect types. Moreover, the substantial scale disparity among cracks, porosity, and spatter further challenges conventional attention designs that lack multi-scale local focus. Fixed-interpolation upsampling compounds these issues by smoothing defect boundaries and losing high-frequency edge details during multi-scale feature fusion. These considerations highlight the need for a detection framework that explicitly addresses frequency-domain decomposition, adaptive upsampling, and multi-scale local attention in a unified manner.

Motivated by the above observations, this study presents WID-YOLO11, a weld surface defect detection model derived from a targeted enhancement of YOLO11n [10]. We summarize the principal contributions of this paper below:

1. We construct a C3k2_LFA-WTConv module by redesigning the learnable wavelet filters of WTConv and introducing a Learnable Frequency-Attention (LFA) mechanism. Residual learning enables the filters to adapt to the frequency-domain distribution of weld defects, and the LFA module applies differentiated weights to the resulting subbands, exploiting the distinct subband energy patterns of each defect type.
2. We improve the MSDA module by extending the dilation-rate set from three values to four (1, 2, 3, and 4), expanding the maximum equivalent receptive field from 7×7 to 9×9 . This enhancement captures contextual information at different ranges, improving detection of medium-scale defects such as porosity and spatter while keeping computational cost low through localized attention.
3. We integrate the DySample dynamic upsampling module to replace conventional fixed-interpolation upsampling. By learning sampling-location offsets in an end-to-end manner, the upsampling process adaptively focuses on defect-rich regions, effectively preserving boundary details across different defect types during multi-scale pyramid aggregation.

Systematic experiments on a dedicated weld defect dataset demonstrate that the resulting model achieves strong detection performance, validating the individual contribution of each proposed component.

2. Related Work

2.1. YOLO-Based Methods for Weld Surface Defect Detection

Deep learning has gained substantial traction in the field of weld surface defect detection over recent years. Unlike conventional approaches that rely on manually engineered features, deep neural networks autonomously construct multi-level feature representations from large-scale image data, demonstrating strong adaptability to varied defect appearances and complex weld background textures. The following subsection surveys several YOLO-based detection methods that have been specifically developed for this task.

Gan et al. [11] improved YOLO11 by integrating a DWR module with the C3k2 backbone to reinforce feature extraction, embedding LSKA-based large-kernel separable attention into an enhanced SPPF module to emphasize key features, and adopting Inner-SIoU loss to refine bounding-box regression accuracy.

Chen et al. [12] developed a compact detector derived from YOLOv5s by substituting ShuffleNetv2 as the backbone and incorporating a modified CBAM attention module to enrich feature discrimination. A dedicated extra detection head targeting small and fine-grained defects was further appended to the network to boost sensitivity at coarse feature scales.

Hu et al. [13] proposed DualSPD-YOLO, integrating SPDConv modules into the YOLO backbone to strengthen small-target detection capability and introducing a TripletAttention mechanism to optimize cross-dimensional feature fusion across the channel, height, and width dimensions.

Wang et al. [14] introduced a frequency-domain feature modulation strategy into the YOLO11 framework, fusing spatial and frequency-domain representations to improve sensitivity to subtle weld defects. This integration of frequency-domain cues into the learning pipeline strengthens the network's capacity to discriminate subtle textural changes characteristic of small-scale weld anomalies.

Li et al. [15] introduced WaveletKernelNet (WKN), embedding a Continuous Wavelet Convolutional layer as the first stage of a standard CNN for industrial fault diagnosis, where scale and translation parameters are directly learned from raw signals to construct task-adaptive kernel banks that highlight defect-related components.

Chen et al. [16] proposed MRWA-Net for wafer defect recognition, substituting max-pooling with a trainable Discrete Wavelet Transform layer and inserting frequency-location attention modules to select discriminative wavelet-domain features at each resolution level.

Wang et al. [17] proposed CARAFE, a content-aware upsampling operator that generates instance-specific reassembly kernels conditioned on local feature content through a lightweight sub-network, enabling the upsampler to adapt its sampling pattern to the local spatial structure of each input feature map.

The YOLO family has established itself as a leading choice for industrial inspection tasks, owing to its streamlined single-stage pipeline and well-balanced accuracy–efficiency profile. The studies surveyed above confirm that task-specific modifications can substantially boost YOLO's applicability to weld surface defect scenarios. Nevertheless, key technical gaps persist: Existing frequency-domain methods typically rely on fixed wavelet bases or apply frequency attention without jointly optimizing the underlying decomposition. Content-adaptive upsampling methods such as CARAFE and DySample have been proposed, but their integration into lightweight YOLO-based detectors for weld inspection has not been extensively explored. These unresolved issues motivate the design of WID-YOLO11, which takes YOLO11n as its starting point. The following section provides a detailed description of the architecture.

2.2. Architecture of the YOLO11 Model

YOLO11, launched by Ultralytics (Los Angeles, CA, USA) in September 2024, represents the most recent advancement in the YOLO line of real-time object detectors. The family encompasses five model scales—n, s, m, l, and x—spanning a wide range of parameter counts and computational budgets. Each configuration preserves the high-throughput characteristic of the YOLO lineage while pushing forward both detection precision and generalization, making the family suitable for deployment scenarios with varying hardware constraints and latency requirements. Among these variants, YOLO11n is the smallest and fastest, comprising only about 2.6 M parameters and 6.3 GFLOPs, which makes it well-positioned for execution on resource-limited hardware and embedded inspection platforms. Such a profile is consistent with the strict efficiency and latency constraints encountered in industrial scenarios like weld defect detection. Its overall architecture is illustrated in Figure 1.

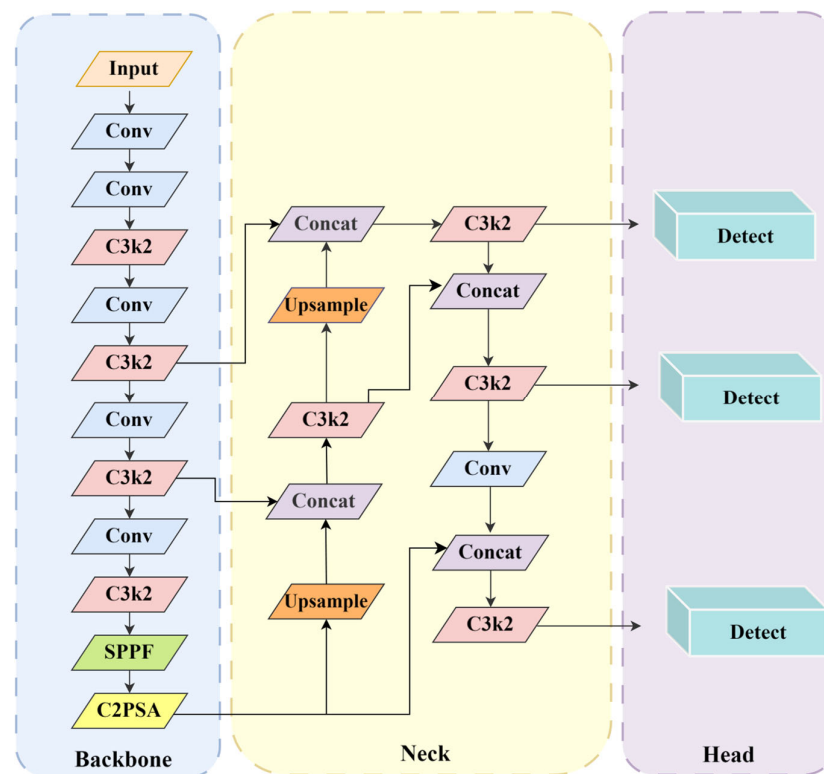


Figure 1. Architecture of the YOLO11n network model.

Following the standard Backbone–Neck–Head paradigm, YOLO11n strikes a favorable balance between model compactness and detection capability. In the Backbone, a refined CSPDarknet serves as the feature extractor. A series of strided convolutional layers progressively reduces the spatial resolution of the input while broadening the receptive field to capture increasingly abstract representations. The C3k2 block with variable kernel configurations endows the network with strong multi-scale perceptual capacity without sacrificing gradient propagation efficiency. Multi-scale contextual cues are then consolidated by an SPPF module through parallel max-pooling at multiple window sizes. A C2PSA partial self-attention block appended at the backbone output selectively amplifies task-relevant regions and suppresses background clutter at modest computational cost. In the Neck, a bidirectional PAN-FPN pathway fuses representations across network depths by interleaving top-down semantic flow with bottom-up spatial detail recovery, allowing the network to draw on complementary information from both high-level and low-level

representations. Interspersed C3k2 blocks further refine the aggregated features at each resolution level, and channel-spatial attention steers the network toward the most informative locations within each feature map. In the head, independent convolutional branches handle classification and localization respectively under an anchor-free design that directly regresses box centers and dimensions, removing the need for manually defined anchor configurations. Predictions are produced from three output strides—P3/8, P4/16, and P5/32—covering small, medium, and large targets respectively. Each branch processes features through C3k2 and CBS layers before jointly outputting coordinates, confidence scores, and class distributions for end-to-end inference. Regarding the loss function, an enhanced CIoU regression loss is combined with Distribution Focal Loss (DFL) to jointly optimize box geometry and localization precision. For label assignment, a Task-Aligned Assigner dynamically allocates positive and negative samples via a joint classification–IoU criterion, promoting stable and efficient convergence. Through careful co-design of channel widths, block depths, and overall computation, YOLO11n delivers competitive detection accuracy within a compact footprint, making it well suited for latency-sensitive tasks such as industrial defect inspection.

3. Materials and Methods

3.1. WID-YOLO Model Architecture

Weld surface defect detection poses three interconnected difficulties: pronounced scale disparity, highly varied appearances, and cluttered backgrounds. To tackle these issues, this study develops WID-YOLO11 by introducing three targeted modifications to YOLO11n. First, we replace the original C3k2 module in the backbone and neck with a newly designed C3k2_LFA-WTConv module. Through learnable wavelet transform convolution and a Learnable Frequency-Attention (LFA) mechanism, feature maps are decomposed into four frequency subbands and adaptively reweighted according to the distinct subband energy patterns of each defect type. This enhances multi-scale frequency-domain feature extraction by exploiting the class-conditioned subband differences among defect types. Second, at the detection layer, the Improved Multi-Scale Dilated Local Attention (IMSDA) module is proposed. Four parallel dilated local-attention branches with dilation rates of 1, 2, 3, and 4 extract contextual information at different ranges, improving detection of medium-scale defects (e.g., porosity and spatter) with low computational complexity. Third, in the upsampling layers of the neck network, the DySample dynamic upsampling module replaces fixed-interpolation upsampling. Sampling-location offsets are jointly optimized with the network, enabling the upsampling process to focus on defect-critical regions and substantially reducing boundary detail loss across different defect types and scales during feature pyramid fusion. The complete network architecture of WID-YOLO11 is presented in Figure 2.

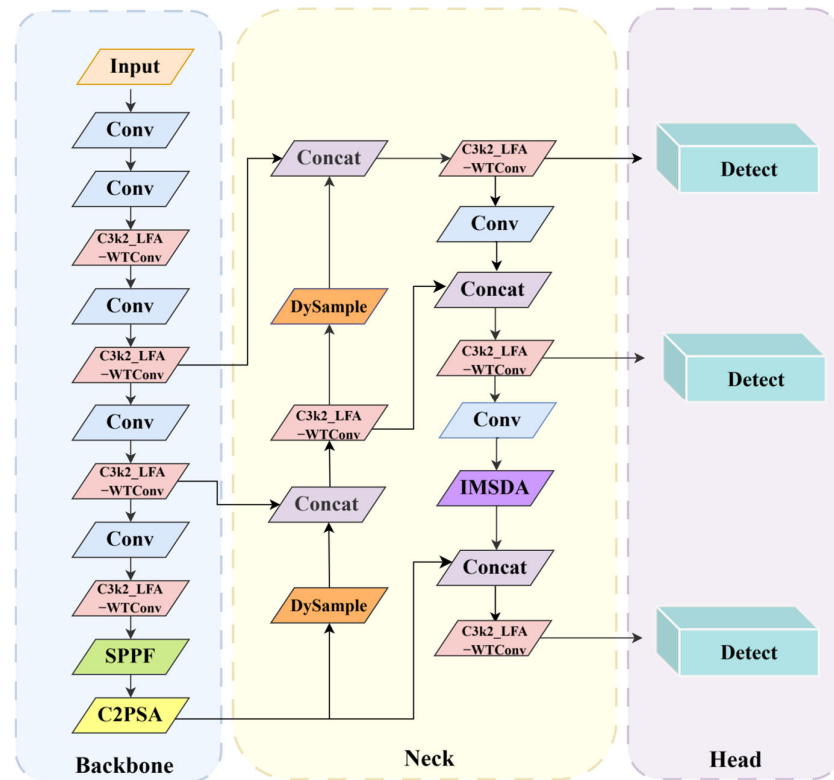


Figure 2. Architecture of the WID-YOLO11 network model.

3.2. C3k2_LFA-WTConv Module

Different types of weld defects exhibit distinct characteristics in the frequency domain: our subband energy analysis reveals that crack regions show the most concentrated LL (low-frequency approximation) energy, with minimal detail-subband contributions; spatter exhibits the highest relative LH/HL energy, reflecting its irregular texture; and porosity occupies an intermediate position. Standard convolution works solely in the spatial domain, making it incapable of capturing these class-conditioned subband differences.

To overcome this limitation, this study introduces the C3k2_LFA-WTConv module. Specifically, the second convolutional layer (cv2) inside the C3k2 Bottleneck is substituted with a redesigned LFA-WTConv (Learnable Frequency-Attention Wavelet Transform Convolution). Figure 3 compares the original C3k2 Bottleneck (left) with the resulting Bottleneck_LFA-WTConv (right) after substitution.

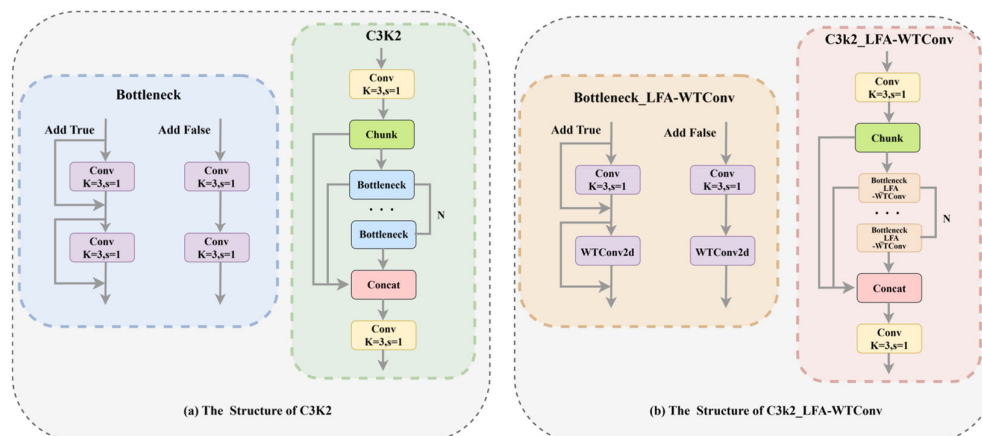


Figure 3. Architecture comparison of C3k2 and C3k2_LFA-WTConv.

As shown in Figure 4, LFA-WTConv uses two parallel branches—one spatial, one wavelet—whose outputs are fused by element-wise addition. The spatial-domain path (upper branch) routes the input features through a depthwise 5×5 convolution (DWConv) and a subsequent learnable scaling module (Base Scale) to extract local spatial patterns. The wavelet-domain path (lower branch) performs specialized refinement of frequency-domain information. The input is first decomposed into four frequency subbands (LL, LH, HL, HH) by a learnable wavelet transform. The resulting tensor of shape $[B, C, 4, H/2, W/2]$ is then reshaped to $[B, 4C, H/2, W/2]$ and passed through a Wavelet Conv and a scaling module (Wavelet Scale) for frequency-domain feature extraction. The tensor is subsequently reshaped back to $[B, C, 4, H/2, W/2]$ and fed into the LFA (Learnable Frequency-Attention) module for adaptive subband reweighting. Finally, an inverse wavelet transform reconstructs the frequency-domain features back into the spatial domain. By fusing the outputs of both paths via element-wise summation, the module simultaneously achieves local spatial-domain feature extraction and multi-scale frequency-domain decomposition.

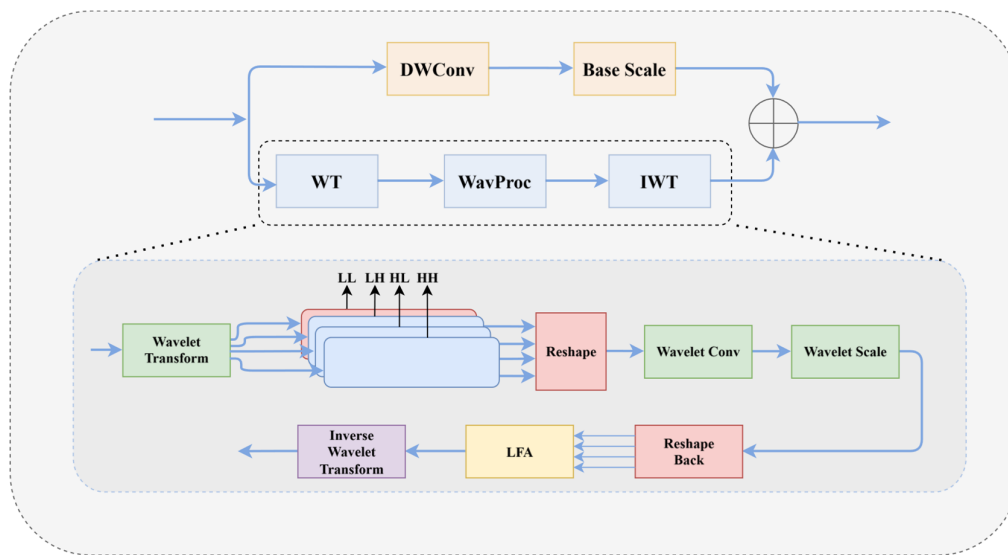


Figure 4. Architecture of LFA-WTConv.

It should be noted that the wavelet transform used in the wavelet-domain path of Figure 4 is not based on a fixed wavelet basis. Instead, a learnable wavelet filter mechanism is introduced. The original WTConv [18] employs a fixed db1 (Haar) wavelet basis whose filter parameters are frozen during training and therefore cannot adapt to the frequency-domain feature distribution of specific tasks. This study adopts a residual learning strategy, superimposing trainable residual filters on top of the fixed db1 basis:

$$F = F_{db1} + \alpha \cdot F_{residual} \quad (1)$$

where $F_{db1} \in \mathbb{R}^{C \times 1 \times K}$ denotes the fixed depthwise (grouped) convolutional filter bank based on the Daubechies-1 (db1) wavelet, with C channels and kernel size $K=2$, and groups $= C$. These parameters are frozen during training. $F_{residual} \in \mathbb{R}^{C \times 1 \times K}$ is a trainable residual filter tensor with the same shape as F_{db1} , initialized from $\mathcal{N}(0, 0.1^2)$, serving as the residual term. $\alpha \in \mathbb{R}$ is a learnable scaling coefficient initialized to $\alpha_0 = 0.1$, and is hard-clamped to the interval $[0, 0.3]$ during both the forward pass and gradient computation:

$$\alpha_c = \text{clamp}(\alpha, 0.0, 0.3) \quad (2)$$

This constraint is a strict architectural bound, not a soft regularization term, ensuring that the learned filter cannot deviate arbitrarily from the db1 basis. The 2D discrete wavelet transform is implemented as two successive separable 1D depthwise convolutions

(row decomposition followed by column decomposition), producing four frequency subbands [LL, LH, HL, HH] per level. Both the forward wavelet transform (WT) and the inverse wavelet transform (IWT) employ independent LearnableWaveletFilters instances: the forward instance is initialized from the exact db1 analysis filter, and the inverse instance is initialized from the exact db1 synthesis (reconstruction) filter, each adapted by its own residual term under the same $[0, 0.3]$ clamp constraint. Although independently learned WT and IWT filters do not guarantee strict perfect reconstruction in a signal-compression sense, WTConv2d operates as a feature extractor within a deep learning pipeline rather than as a signal codec; near-perfect reconstruction is fully sufficient for this purpose. The empirically measured α after full training converges to 0.0577 (approximately 19% of the constraint upper bound), confirming that the filter-bank perturbation remains small and that the learned filter bank retains the near-orthogonal properties of the db1 basis.

Through end-to-end training, the filters can adaptively learn the optimal decomposition strategy suited to the frequency-domain distribution of weld defects while preserving the fundamental decomposition properties of the db1 basis, thereby more effectively capturing the class-conditioned subband distributions of different defect types.

Following the Reshape Back operation in the wavelet-domain path, the four resulting subbands are forwarded into the LFA (Learnable Frequency-Attention) module, whose internal layout is depicted in Figure 5. A limitation of vanilla WTConv lies in its uniform treatment of all four subbands, ignoring the fact that different defect categories exhibit distinct subband energy distributions. Specifically, crack regions show the most concentrated LL energy with the lowest relative high-frequency contributions; spatter shows the highest relative LH/HL contributions; and porosity occupies an intermediate distribution. Uniform subband weights fail to leverage these class-specific differences, motivating the learnable frequency attention mechanism. The LFA module first concatenates the four subbands into a tensor of shape $[B, 4C, H/2, W/2]$. The spatial dimensions are squeezed via global pooling to yield a channel descriptor. This descriptor is processed by a bottleneck with two 1×1 convolutions separated by ReLU, and the output is passed through a Sigmoid to generate attention weights. The input tensor is then reweighted by multiplying with these weights. The recalibrated tensor is finally reshaped back to $[B, C, 4, H/2, W/2]$ and routed back to the main pathway, allowing the network to learn task-aware importance for each frequency subband. The above procedure is formulated as:

$$z = \text{GAP}(x) \quad (3)$$

$$A = \sigma \left(\text{Conv}_2(\text{ReLU}(\text{Conv}_1(z))) \right) \quad (4)$$

$$x' = x \odot A \quad (5)$$

Here, $x \in \mathbb{R}^{B \times 4C \times (H/2) \times (W/2)}$ denotes the unfolded subband feature tensor after wavelet decomposition; $\text{GAP}(\cdot)$ performs global average pooling and outputs the channel descriptor z ; Conv_1 and Conv_2 are 1×1 convolutions that respectively compress and restore the channel dimension within the bottleneck; $\text{ReLU}(\cdot)$ and σ refer to the rectified linear unit and Sigmoid activations; A corresponds to the learned per-subband attention vector; and $x' \in \mathbb{R}^{B \times 4C \times (H/2) \times (W/2)}$ is the resulting frequency-domain feature map after adaptive reweighting.

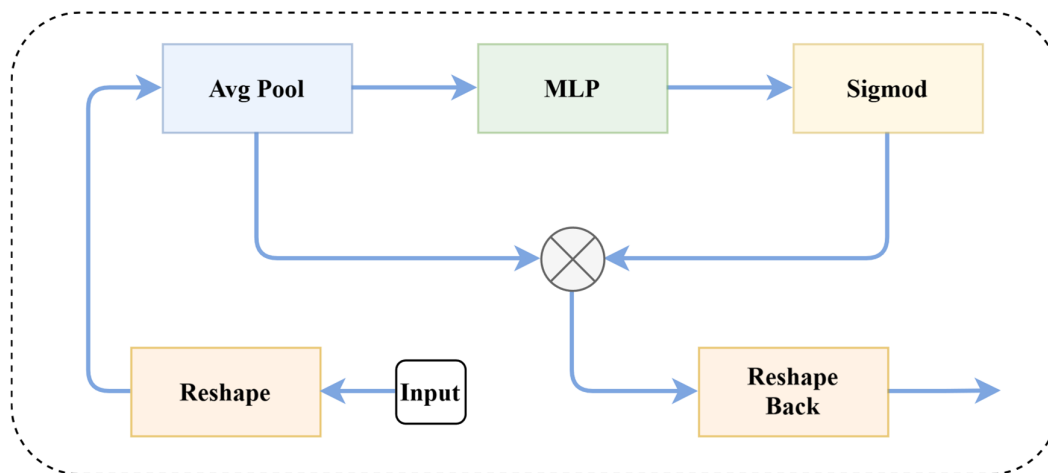


Figure 5. Architecture of the LFA frequency-domain attention module.

3.3. IMSDA Multi-Scale Dilated Local Attention Module

Weld defects exhibit pronounced multi-scale characteristics: cracks may be only a few pixels wide yet extend over considerable lengths; the diameters of porosity range from a few pixels to several tens of pixels; and spatter varies even more widely in size and shape. Existing attention mechanisms typically adopt either global attention or simple channel-spatial attention. Global attention incurs a computational complexity of $O(H^2 \cdot W^2)$, which is prohibitively expensive for high-resolution weld images. Channel-spatial attention, on the other hand, lacks targeted focus on multi-scale local regions and struggles to simultaneously enhance features for defects with markedly different scales, such as cracks, porosity, and spatter. Drawing on the Multi-Scale Dilated Attention (MSDA) [19] mechanism as a foundation, this work improves the MSDA module, referring to the resulting design as IMSDA, and embeds it at the detection layer to overcome the above limitations.

Architecturally, IMSDA is organized as four parallel dilated local-attention branches operating at distinct dilation rates $r \in \{1, 2, 3, 4\}$, which yield equivalent receptive fields of 3×3 , 5×5 , 7×7 , and 9×9 . Given an input feature map X , a single 1×1 convolution first projects X into query (Q), key (K), and value (V) tensors; these tensors are subsequently divided evenly along the channel axis, with each chunk dispatched to one of the four branches. Within each branch, multi-head dilated local attention is computed independently, as depicted in Figure 6. The branches with small dilation rates are responsible for capturing fine-grained cues such as crack edges, whereas those with larger dilation rates aggregate contextual information from wider neighborhoods, which is beneficial for porosity and spatter that occupy broader regions. After parallel computation, the four branch outputs are concatenated channel-wise and integrated through a linear projection, producing the multi-scale enhanced feature representation.

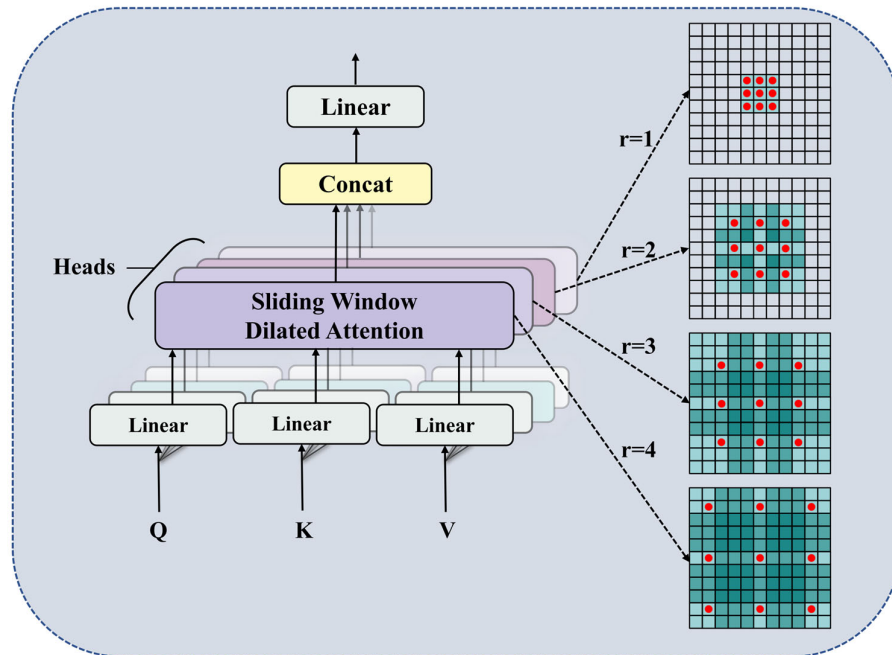


Figure 6. Architecture of the multi-head sliding-window dilated attention module.

As illustrated in Figure 7, the local attention computation within each branch proceeds as follows. The query Q is first rearranged via multi-head reshape (Reshape Q). For each spatial position, the keys K and values V are extracted from a sparse local neighborhood at a dilation rate of r . Specifically, k^2 positions are sampled within a $k \times k$ ($k = 3$) window centered at each location with spacing r . Scaled dot-product attention is then computed between Q and the unfolded K , followed by Softmax normalization to obtain attention scores, which perform a weighted aggregation over the unfolded V . Finally, a reshape operation restores the multi-head output to the spatial feature map. The computational complexity of this procedure is $O(k^2 \cdot H \cdot W)$, substantially lower than the $O(H^2 \cdot W^2)$ of global attention.

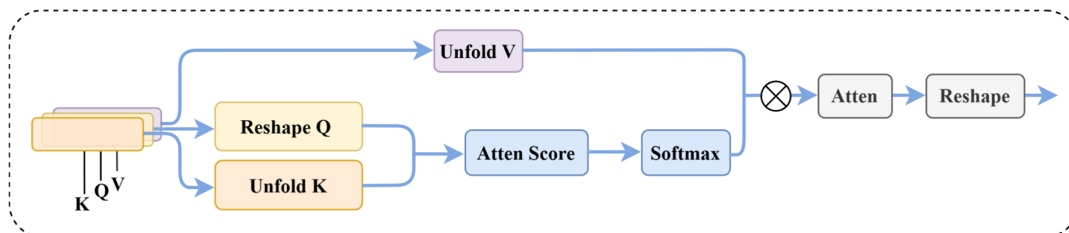


Figure 7. Architecture of a single-branch local attention unit in IMSDA.

The process is formalized as follows:

$$[Q_r, K_r, V_r] = \text{Split}(W_{qkv}X), \quad r=1, 2, 3, 4 \quad (6)$$

where W_{qkv} denotes the weight of the 1×1 projection convolution, and r is the branch index (simultaneously serving as the dilation rate of each branch). Within the r branch, K_r and V_r are unfolded in a $k \times k$ local window at dilation rate r , and attention weights are computed for weighted aggregation:

$$a_{r,n,j} = \text{Softmax}_j(s^{-1/2} \cdot Q_{r,n} \cdot K_{r,n,j}^T), \quad O_{r,n} = \sum_{j=0}^{k^2-1} a_{r,n,j} \cdot V_{r,n,j} \quad (7)$$

where s is the head-dimension scaling factor, n denotes the spatial position, and $j \in \{0 \dots k^2 - 1\}$ is the local window position index. $O_{r,n}$ is the aggregated output of the branch at position n . The per-branch outputs are subsequently stacked along the channel dimension and aggregated through a learned linear transformation:

$$Y = W_{proj}([O_1; O_2; O_3; O_4]) \quad (8)$$

where Y is the final module output, W_{proj} is the linear projection weight, and $[\cdot; \cdot]$ denotes concatenation along the channel dimension. By employing four parallel local-attention branches with different dilation rates, IMSDA achieves differentiated enhancement of multi-scale defect features—including cracks, porosity, and spatter—while maintaining a low computational cost of $O(k^2 \cdot H \cdot W)$.

3.4. DySample Dynamic Upsampling Module

In the neck feature-fusion process of YOLO-series detectors, upsampling is required to restore high-level, low-resolution feature maps to the spatial resolution of low-level feature maps. Conventional fixed interpolation methods (e.g., nearest-neighbor, bilinear) are non-learnable and tend to produce blurry or aliased edges. While transposed convolution is learnable, it is prone to introducing checkerboard artifacts and incurs higher computational overhead. Crucially, the discriminative features of weld defects—such as crack contours and porosity boundaries—are predominantly concentrated in local edge regions. Non-adaptive, fixed-kernel upsampling cannot dynamically attend to these information-dense localities, which often leads to the smoothing or loss of critical edge detail during the fusion process.

To address this issue, this study introduces the DySample dynamic upsampling module [20] in the upsampling layers of the neck network, whose structure is shown in Figure 8. This module follows a “Learn then PixelShuffle” (LP) paradigm: given an input feature map $X \in \mathbb{R}^{C \times H \times W}$, two parallel 1×1 convolutions (each with $2gs^2$ output channels, where g is the number of groups and s is the upsampling scale factor) separately predict a sampling offset(X) and a dynamic range factor $\text{scope}(X)$. The offset is element-wise multiplied by the Sigmoid-activated range factor and scaled by a fixed coefficient of 0.5, then added to a predefined uniform initial grid p_0 to obtain the dynamic sampling coordinates p . The coordinates are subsequently rearranged by PixelShuffle from low-resolution space to high-resolution space. Finally, GridSample bilinear interpolation samples the input features at the dynamic coordinates, directly producing a high-resolution output feature map of size $C \times 2H \times 2W$. The dynamic sampling process is formalized as:

$$p = p_0 + 0.5 \cdot \text{offset}(X) \cdot \sigma(\text{scope}(X)) \quad (9)$$

$$Y = \text{GridSample}(X, p) \quad (10)$$

where p_0 is the predefined uniform initial position grid, $\text{offset}(X)$ is the output of the offset prediction convolution, $\sigma(\text{scope}(X))$ is the dynamic scaling weight obtained by applying Sigmoid to the scope prediction, p is the dynamically adjusted sampling coordinate, $\text{GridSample}(\cdot, \cdot)$ denotes bilinear interpolation sampling at the given coordinates, and Y is the upsampled output. Compared with fixed interpolation, DySample dynamic upsampling module learns sampling locations end-to-end to adaptively focus on feature-dense regions such as defect edges, effectively preserving critical details including crack contours and porosity boundaries. The range-factor branch dynamically modulates the offset magnitude via Sigmoid: during early training (when scope is initialized to 0 and

sigmoid outputs 0.5), the model predominantly uses conservative offsets, gradually increasing sampling freedom as training proceeds. The grouped sampling mechanism further allows different channel groups to learn distinct offset patterns, enhancing the network's ability to differentially preserve features of multiple defect types.

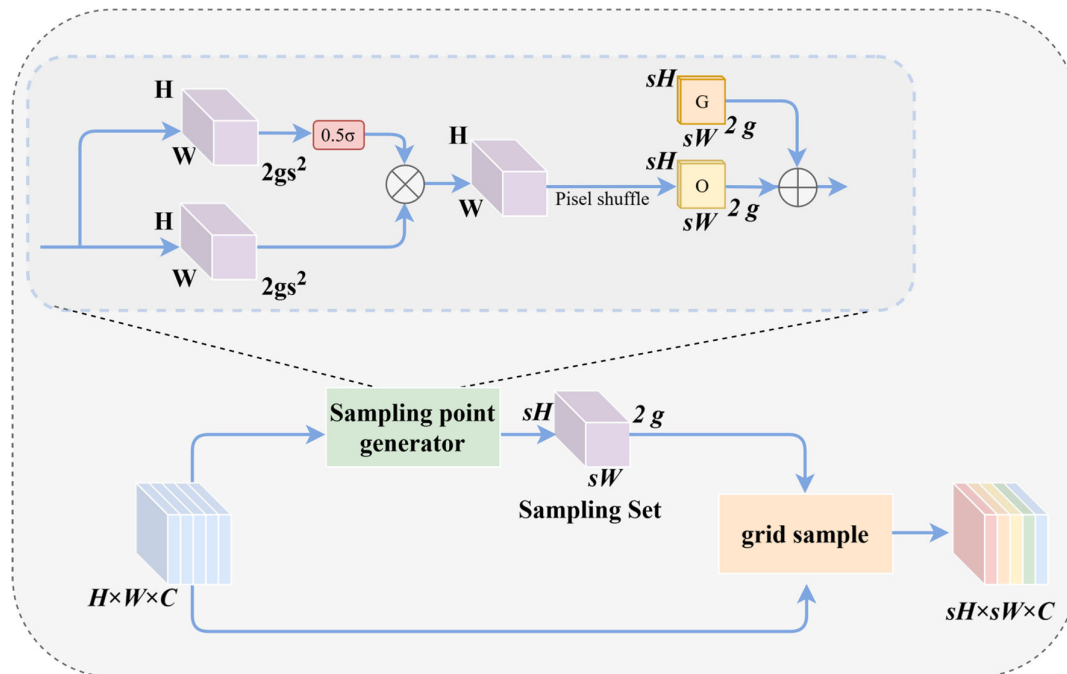


Figure 8. Architecture of the DySample dynamic upsampling module.

4. Experimental Studies and Results Analysis

4.1. Experimental Environment

The experimental platform ran Windows 10 (Microsoft Corporation, Redmond, WA, USA) with Python 3.8.20 (Python Software Foundation, Wilmington, DE, USA). The deep learning framework was PyTorch 2.4.1 (Meta AI, New York, NY, USA), with GPU acceleration provided by CUDA 12.4 and cuDNN 8.7.0 (NVIDIA Corporation, Santa Clara, CA, USA). The host machine was equipped with an Intel Core i5-13490 CPU (Intel Corporation, Santa Clara, CA, USA) and an NVIDIA RTX 3070 Ti GPU (8 GB VRAM, NVIDIA Corporation, Santa Clara, CA, USA).

All models were trained using the same set of hyperparameters listed in Table 1. Default pretrained weights provided by Ultralytics were used for all models. All other training settings, including augmentation policies, learning rate schedule, warm-up strategies, and loss weights, follow the default configuration of Ultralytics YOLO (version 8.3.83).

Table 1. Summary of the training hyperparameters.

Epochs	Batch Size	Learning Rate	Momentum	Optimizer	Image Size	Random Seed
300	16	0.01	0.937	SGD	640×640	42

4.2. Dataset

The dataset used in this study consists of 986 original images sourced from a publicly available Kaggle weld-defect benchmark (Kaggle Inc., San Francisco, CA, USA) [21], covering three defect categories (Crack, Porosity, Spatters). All images were re-annotated from scratch by two NDT-qualified engineers into three defect categories (Crack, Porosity, Spatters) in accordance with the ISO 6520-1 [22] weld-defect classification standard; the

original Kaggle labels were discarded. For ambiguous cases, a third senior engineer was consulted to reach a consensus; cases that remained unresolvable were excluded. Images originally labelled as Good Weld with no visible defect were retained as defect-free background samples (i.e., negative samples without bounding-box annotations). The images were partitioned at the original-image-ID level (group-wise split, seed = 42) into training, validation, and test subsets at a 7:2:1 ratio, yielding 689, 196, and 101 source images, respectively. To prevent information leakage, all augmented versions of an original image were strictly confined to the same subset. Augmentation was applied exclusively to the training subset after partitioning, expanding it from 689 to 2650 images. The validation and test subsets were not augmented. Detailed dataset statistics are provided in Tables 2 and 3. Example samples from each defect category are illustrated in Figure 9.

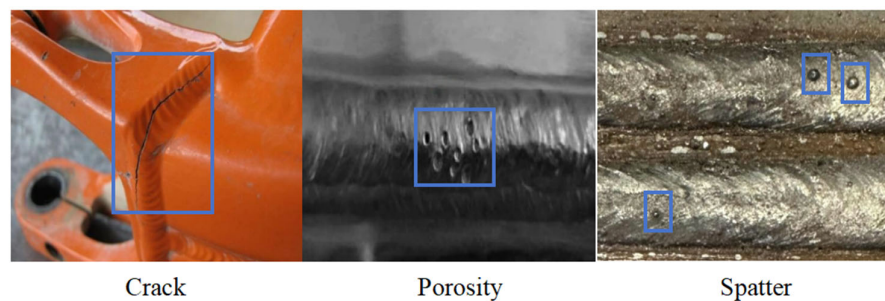


Figure 9. Sample images from the weld defect dataset.

Table 2. Dataset statistics by split.

Subset	Original Images	Augmented Images	Positive (with Annotation)	Background (Defect-Free)	Total Instances	Avg. Boxes/Image (Augmented Image)
Train	689	2650	1914	736	34,393	13.0
Val	196	196	121	75	1842	9.4
Test	101	101	60	41	759	7.5
Total	986	2947	2095	852	36,994	12.6

Table 3. Per-class instance distribution.

Subset	Crack	Porosity	Spatters
Train	1254	12,907	20,232
Val	75	813	954
Test	43	347	369
Total	1372	14,067	21,555

4.3. Evaluation Metrics

Model performance is assessed along two dimensions: detection accuracy and computational efficiency. Accuracy is measured by Precision (P), Recall (R), and mean Average Precision (mAP), while computational cost is measured by the total parameter count (Params) and the number of floating-point operations per forward pass (GFLOPs).

Among them, P reflects the reliability of the predictions: given all bounding boxes predicted as defects, it indicates the fraction that genuinely correspond to defects. A higher P therefore implies fewer false alarms in the inspection results. P is formulated as:

$$P = \frac{TP}{TP+FP} \quad (11)$$

where TP counts the predictions correctly matched to ground-truth defects, and FP counts the predictions that are mistakenly produced on defect-free regions.

Complementing P , the metric R characterizes detection completeness, defined as the proportion of actual defect instances that the model correctly detects. A higher R corresponds to fewer omitted defects, which is particularly critical in industrial weld inspection where missed defects may lead to safety risks. R is computed as:

$$R = \frac{TP}{TP+FN} \quad (12)$$

where FN represents the ground-truth defects that fail to be detected. Both P and R take values in $[0, 1]$, and a value closer to 1 indicates stronger performance on the corresponding aspect.

To jointly reflect both precision and recall across all operating thresholds, the Average Precision (AP) for each category is computed as the area enclosed by its Precision-Recall curve. Let AP_i denote the AP of the i -th defect category. Aggregating AP across all N defect categories yields the mean Average Precision (mAP), which serves as the primary accuracy indicator:

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (13)$$

where N denotes the number of defect categories. Consistent with standard object detection evaluation protocols, two variants are reported: mAP@0.5, computed at a fixed IoU threshold of 0.5, which primarily reflects category recognition ability; and mAP@0.5:0.95, averaged over IoU thresholds ranging from 0.5 to 0.95 at 0.05 intervals, which additionally penalizes imprecise localizations.

Beyond accuracy, computational overhead is quantified by two efficiency indicators: Params, denoting the total count of trainable weights in the network, and GFLOPs, representing the number of floating-point operations consumed by one forward pass on a 640×640 image. Smaller values for both metrics indicate a more compact and inference-friendly model.

4.4. Module Effectiveness Study

Per-Module Contribution Analysis

To quantify the independent contribution of each proposed module, a controlled ablation study was conducted following a single-variable principle: modules were added to the YOLO11n baseline one at a time so that the performance change attributable to each could be measured in isolation. For conciseness, C3k2_LFA-WTConv and DySample dynamic upsampling are abbreviated as CLW and DS, respectively, throughout the tables. The corresponding results are presented in Table 4.

Table 4. Incremental Performance Gains of Each Proposed Module.

YOLO11n	CLW	DS	IMSDA	mAP@0.5 (%)	Params (M)	GFLOPs	R (%)	mAP@0.5:0.95 (%)
√	-	-	-	59.0	2.58	6.3	51.3	31.7
√	√	-	-	63.2	2.83	7.6	56.1	35.6
√	√	√	-	64.5	2.91	7.8	61.1	37.2
√	√	-	√	62.5	2.90	7.8	58.2	35.2
√	-	√	√	64.2	2.66	6.5	58.9	35.1
√	√	√	√	63.5	2.91	7.8	61.7	35.1

As reported in Table 4, each of the three introduced modules brings a measurable benefit over the YOLO11n baseline, which yields 59.0% mAP@0.5, 51.3% Recall, and 31.7% mAP@0.5:0.95. The standalone addition of CLW improves mAP@0.5 to 63.2% (+4.2 pp) and mAP@0.5:0.95 to 35.6% (+3.9 pp), confirming the effectiveness of learnable wavelet-

based frequency-domain decomposition for weld defect feature extraction. Further stacking DS on top of CLW raises mAP@0.5 to 64.5% (+1.3 pp over CLW alone) and Recall to 61.1%, indicating that DS complements frequency-domain enhancement by improving boundary preservation during multi-scale feature fusion.

When IMSDA is added to form the full WID-YOLO11 model, mAP@0.5 settles at 63.5%, slightly below the CLW + DS combination (64.5%). This small difference likely reflects a partial functional overlap between the multi-scale dilated attention in IMSDA and the multi-scale representations already captured by CLW + DS. Importantly, the full configuration achieves the highest Recall among all variants (61.7%), suggesting that IMSDA helps recover additional true defects that may otherwise be missed.

Taken together, the full WID-YOLO11 model delivers consistent gains over the baseline across all three metrics (+4.5 pp mAP@0.5, +10.4 pp Recall, +3.4 pp mAP@0.5:0.95), while incurring only a moderate increase in parameters (+0.33M) and GFLOPs (+1.5). The three modules collectively strike a sound accuracy–efficiency trade-off.

4.5. Benchmarking Against Alternative Approaches

4.5.1. Comparison of Wavelet-Based C3k2 Modules

To isolate the effect of the learnable wavelet design, we compare YOLO11n, YOLO11n with the original WTConv, and YOLO11n with the proposed C3k2_LFA-WTConv. All three configurations were trained under identical conditions, and results are reported as mean $\pm$ standard deviation over three independent runs (seeds 42, 43, 44). The results are summarized in Table 5.

Table 5. Performance comparison of wavelet-based C3k2 modules (mean $\pm$ std over three runs).

Model	mAP@0.5 (%)	Params (M)	GFLOPs	R (%)	mAP@0.5:0.95 (%)
YOLO11n	61.28 $\pm$ 1.81	2.58	6.3	56.33 $\pm$ 4.05	33.25 $\pm$ 1.10
YOLO11n+C3k2_WTConv	62.24 $\pm$ 0.88	2.80	7.5	55.38 $\pm$ 3.21	33.87 $\pm$ 0.39
YOLO11n+C3k2_LFA-WTConv	62.37 $\pm$ 1.22	2.83	7.6	56.16 $\pm$ 0.38	35.64 $\pm$ 0.52

As shown in Table 5, replacing the standard C3k2 with WTConv brings a moderate improvement in mAP@0.5 (+0.96 pp) but only a marginal gain in mAP@0.5:0.95 (+0.62 pp), at the cost of additional parameters and computation. The proposed LFA-WTConv achieves the highest scores on all three accuracy metrics. While the mAP@0.5 advantage over WTConv is small (0.13 pp), LFA-WTConv improves mAP@0.5:0.95 by 1.77 pp over WTConv and by 2.39 pp over the baseline, indicating more precise localization. The parameter and GFLOPs increases relative to WTConv are negligible (+0.03M and +0.1 GFLOPs). The LFA module contributes to this improvement by learning to weight the four subbands adaptively, exploiting the distinct subband energy patterns of different defect types, without requiring substantial extra capacity.

4.5.2. Evaluation of Dilation Strategies in MSDA

To determine the optimal dilation-rate configuration, we compare the original MSDA (dilation rates 1, 2, 3), an expanded variant (1, 2, 3, 5), and the proposed IMSDA (1, 2, 3, 4). All modules are integrated into the detection layer of the YOLO11n baseline under identical conditions, and results are reported as mean $\pm$ standard deviation over three independent runs (seeds 42, 43, 44). The results are summarized in Table 6.

Table 6. Performance comparison of MSDA with different dilation-rate sets (mean $\pm$ std over three runs).

Model	mAP@0.5 (%)	Params (M)	GFLOPs	R (%)	mAP@0.5:0.95 (%)
YOLO11n	61.28 $\pm$ 1.81	2.58	6.3	56.33 $\pm$ 4.05	33.25 $\pm$ 1.10
YOLO11n+MSDA (1, 2, 3)	61.62 $\pm$ 0.44	2.69	6.6	54.20 $\pm$ 1.80	33.45 $\pm$ 0.13
YOLO11n+MSDA (1, 2, 3, 5)	59.95 $\pm$ 0.93	2.65	6.5	54.93 $\pm$ 0.18	33.09 $\pm$ 1.97
YOLO11n+IMSDA (1, 2, 3, 4)	62.90 $\pm$ 1.32	2.65	6.5	57.77 $\pm$ 1.60	35.38 $\pm$ 0.62

As shown in Table 6, the original MSDA with dilation rates (1, 2, 3) brings a modest improvement over the baseline, increasing mAP@0.5 by 0.34 pp to 61.62% and mAP@0.5:0.95 by 0.20 pp to 33.45%, with a parameter increase of 0.11M. The expanded variant with dilation rates (1, 2, 3, 5) actually degrades performance below the baseline, with mAP@0.5 dropping to 59.95% and mAP@0.5:0.95 to 33.09%. This suggests that introducing an excessively large dilation rate ($r = 5$) disrupts the local attention mechanism and introduces irrelevant background context.

IMSDA with dilation rates (1, 2, 3, 4) achieves the best performance across all accuracy metrics. It improves mAP@0.5 by 1.62 pp over the baseline to 62.90%, and raises mAP@0.5:0.95 by 2.13 pp to 35.38%. Compared with the original MSDA, IMSDA gains 1.28 pp in mAP@0.5 and 1.93 pp in mAP@0.5:0.95. Recall also reaches the highest value among all configurations (57.77%). The parameter increase relative to the baseline is only 0.07 M, confirming that the additional dilation branch adds negligible overhead. The improvement can be attributed to the expanded receptive field (from 7×7 to 9×9), which enhances the capture of medium-scale defect structures without introducing excessive dilation that would compromise local attention.

4.5.3. Upsampling Module Comparison

To assess the contribution of the DySample dynamic upsampling module (DS), three alternative strategies—bilinear interpolation (Bilinear), transposed convolution (ConvTranspose) [23] and CARAFE—were each substituted into the neck upsampling layers of the YOLO11n baseline, under the same experimental settings. The outcomes are reported in Table 7.

Table 7. Comparison of Various Upsampling Modules.

Model	mAP@0.5 (%)	Params (M)	GFLOPs	R (%)	mAP@0.5:0.95 (%)
YOLO11n	59.0	2.58	6.3	51.3	31.7
YOLO11n+Bilinear	60.2	2.58	6.3	52.4	32.4
YOLO11n+ConvTranspose	62.1	2.91	8.0	55.9	34.8
YOLO11n+CARAFE	59.2	2.72	6.8	54.1	32.3
YOLO11n+DS	62.7	2.60	6.3	52.1	34.7

As shown in Table 7, DS achieves the best overall performance, outperforming the baseline by 3.7 pp in mAP@0.5 and 3.0 pp in mAP@0.5:0.95, with only 0.02M additional parameters. Bilinear interpolation and CARAFE yield modest gains (+1.2 and +0.2 pp in mAP@0.5, respectively). ConvTranspose improves mAP@0.5 by 3.1 pp but increases parameters by 0.33M. Overall, DS provides the most favorable trade-off between accuracy and efficiency.

4.5.4. State-of-the-Art Model Comparison

WID-YOLO11 is further evaluated against a set of lightweight detectors on the same weld defect dataset under identical conditions. The comparison includes RT-DETR-R18 [24] and YOLOv5n [25], YOLOv8n [26], YOLOv10n [27], YOLO11n, YOLO12n [28], and

YOLO13n [29]. All models are benchmarked on six metrics—mAP@0.5, mAP@0.5:0.95, Recall, Params, GFLOPs, and FPS. The results are summarized in Table 8. All models were trained and evaluated using the identical dataset splits, input resolution (640×640), augmentation pipeline, and number of training epochs (300). Evaluation was performed with the same Ultralytics validation script (v8.3.83). YOLO12n and YOLO13n were sourced from their respective official implementations; all FPS values were measured on an RTX 3070 Ti (8 GB) GPU with PyTorch FP32 and batch size = 1. A full-pass warm-up run was performed first and excluded from timing; the reported speed is the average over a complete pass on the test set (101 images). Latency includes preprocessing, inference, and postprocessing (including NMS).

Table 8. Comparison with State-of-the-Art Models.

Model	mAP@0.5 (%)	Params (M)	GFLOPs	R (%)	mAP@0.5:0.95 (%)	FPS
RT-DETR-R18	56.1	25.45	78.0	51.5	29.3	106.9
YOLOv5n	62.5	2.50	7.1	55.2	35.3	389.3
YOLOv8n	58.6	3.01	8.1	54.6	31.8	275.7
YOLOv10n	56.2	2.70	8.2	52.6	31.6	444.2
YOLO11n	59.0	2.58	6.3	51.3	31.7	232.6
YOLO12n	57.3	2.58	6.3	54.1	30.3	304.9
YOLO13n	59.8	2.45	6.1	57.0	34.7	233.0
Ours	63.5	2.91	7.8	61.7	35.1	322.6

As shown in Table 8, WID-YOLO11 achieves the highest scores across all evaluated metrics among the compared detectors. Specifically, it outperforms the YOLO11n baseline by 4.5 pp in mAP@0.5, 10.4 pp in Recall, and 3.4 pp in mAP@0.5:0.95. It also surpasses the best competing YOLO variant (YOLOv5n) by 1.0 pp in mAP@0.5 and by 6.5 pp in Recall. RT-DETR-R18, despite its larger parameter count (25.45 M vs. 2.91 M), lags behind WID-YOLO11 by 7.4 pp in mAP@0.5 and 10.2 pp in Recall. These results demonstrate that WID-YOLO11 offers a favorable accuracy–efficiency trade-off among lightweight detectors for weld defect inspection.

4.6. Qualitative Analysis of Detection Results

4.6.1. Frequency-Domain Analysis of Defect Characteristics

To provide empirical support for the frequency-domain design motivation and to verify that the proposed learnable wavelet mechanism operates as intended, two analyses are conducted.

Per-Class Wavelet Subband Energy Distribution

Figure 10 presents the normalized wavelet subband energy distribution (variance ratio) computed on ground-truth defect bounding-box crops from the test set, decomposed using a db1 single-level 2D DWT. Crops smaller than 8×8 pixels or truncated at image boundaries were excluded to avoid boundary artifacts. All three defect types are dominated by the LL (low-frequency approximation) subband, as expected for natural image patches. However, the distributions are not identical. Crack regions exhibit the most concentrated LL energy (0.9547 ± 0.0356 , $n = 43$), with the lowest relative contributions from the detail subbands (LH: 0.0162 ± 0.0123 , HL: 0.0264 ± 0.0270 , HH: 0.0027 ± 0.0047). Spatter regions show the highest relative contributions in LH and HL (0.0621 ± 0.0342 and 0.0534 ± 0.0259 , respectively, $n = 71$), with LL = 0.8740 ± 0.0558 , reflecting their irregular, multi-directional texture. Porosity occupies an intermediate position across all subbands (LL: 0.9036 ± 0.0593 , LH: 0.0405 ± 0.0289 , HL: 0.0487 ± 0.0319 , HH: 0.0072 ± 0.0101 , $n = 80$). Although the inter-class differences are moderate in absolute magnitude, they are statistically

consistent across samples (see error bars in Figure 10) and confirm that the three defect classes present distinguishable subband energy signatures. This provides an empirical justification for the differentiated subband weighting in LFA: because the classes differ in how energy is distributed across subbands, learning class-adaptive weights over these subbands can improve discrimination.

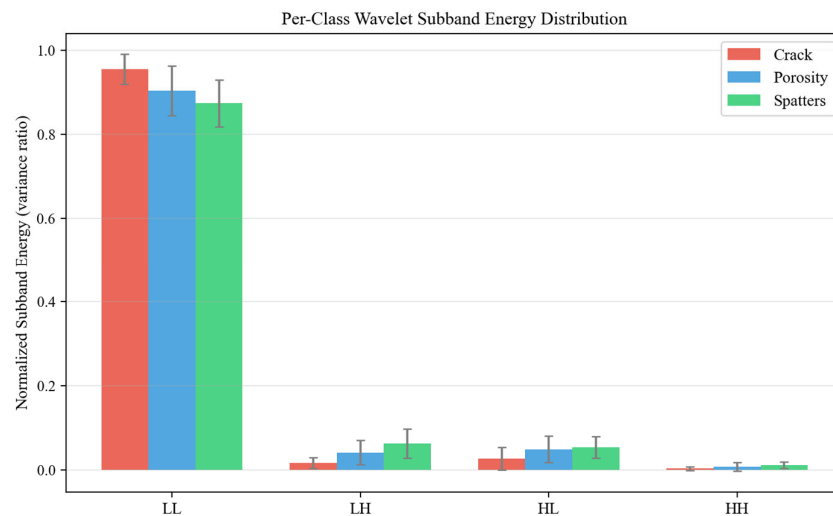


Figure 10. Subband energy distribution by defect class.

Per-class wavelet subband energy distribution (normalized variance ratio), computed on ground-truth defect bounding-box crops from the test set using db1 single-level 2D DWT decomposition. Error bars denote $\pm$ one standard deviation across samples. All classes are LL-dominant; crack regions show the most concentrated LL energy, while spatter exhibits the highest relative LH/HL contributions. Sample sizes: Crack $n = 43$, Porosity $n = 80$, Spatters $n = 71$.

Learned Filter Responses

Figure 11 compares the original fixed db1 filters with the filters learned after full training for the four subbands (LL, LH, HL, HH). The learned filters retain the clear directional structure of the db1 basis, with a residual scaling coefficient $\alpha = 0.0577$ (approximately 19% of the constraint upper bound of 0.3). This confirms that the learnable wavelet layer preserves the fundamental frequency-decomposition properties of db1 while introducing small, task-specific adjustments during training.

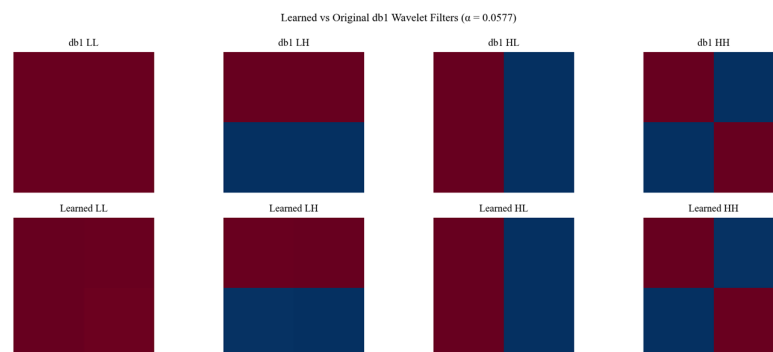


Figure 11. Learned vs. original db1 wavelet filters.

Comparison of original db1 wavelet filters (top row) and learned filters after full training (bottom row) for the four subbands (LL, LH, HL, HH). Residual scaling $\alpha = 0.0577$, confirming that the db1 decomposition properties are preserved under the bounded residual adaptation.

4.6.2. False-Positive Analysis on Defect-Free Weld Images

At the standard operating thresholds (confidence = 0.25, IoU = 0.45), the false-positive behaviour of both the baseline and the proposed WID-YOLO11 on the 41 defect-free images is summarised in Table 9.

Table 9. False-positive analysis on the 41 defect-free test images.

Model	FP Images	Total FP Boxes	FPPI	Mean FP Confidence
YOLO11n (baseline)	8/41	29	0.707	0.399
WID-YOLO11	9/41	23	0.561	0.386

While WID-YOLO11 introduces a false positive in one additional defect-free image (9 vs. 8), it reduces the total number of false-positive boxes from 29 to 23 and lowers FPPI from 0.707 to 0.561—a 20.7% reduction. This indicates that the proposed model produces sparser false alarms per normal image, which is advantageous in practice. The mean confidence of these false positives is also slightly lower (0.386 vs. 0.399).

We further examined the overall precision on the full test set (101 images: 60 with defects + 41 defect-free). At the default threshold, precision is 70.8% for the baseline and 65.8% for WID-YOLO11. This decrease is expected because the proposed model substantially raises recall from 51.3% to 61.7% and thus produces more predictions. Critically, all false positives from both models exhibit confidence below 0.41, meaning they can be almost entirely eliminated by a modest threshold increase with negligible impact on true positives. Taken together, these results confirm that WID-YOLO11 offers stronger false-positive suppression on normal weld surfaces and maintains a favourable precision-recall trade-off that can be further optimized through threshold tuning in practice.

4.6.3. Per-Class Detection Performance

All metrics in this section are computed on the independent test set (101 images), which was held out entirely during training and validation. Table 10 reports per-class Precision, Recall, AP@0.5, and AP@0.5:0.95 for both WID-YOLO11 and the YOLO11n baseline.

Table 10. Per-class detection performance on the independent test set.

Model	Class	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
YOLO11n	Crack	73.2	65.1	70.1	39.6
	Porosity	69.0	53.3	61.1	31.6
	Spatters	70.1	35.5	45.8	24.0
	All	70.8	51.3	59.0	31.7
WID-YOLO11	Crack	70.2	86.0	82.3	49.2
	Porosity	64.7	57.9	59.8	31.9
	Spatters	62.4	41.2	48.5	24.2
	All	65.8	61.7	63.5	35.1

WID-YOLO11 achieves the largest gains on Crack (AP@0.5: +12.2 pp; Recall: +20.9 pp), consistent with the frequency-domain design motivation. Spatters remains the most

challenging class for both models. The confusion matrix and per-class PR curves are shown in Figures 12 and 13.

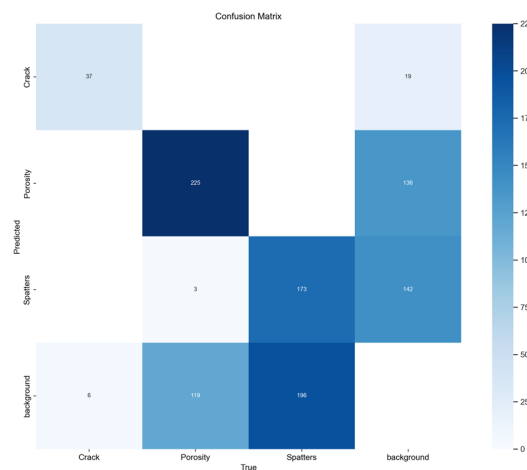


Figure 12. Confusion matrix of WID-YOLO11 on the independent test set.

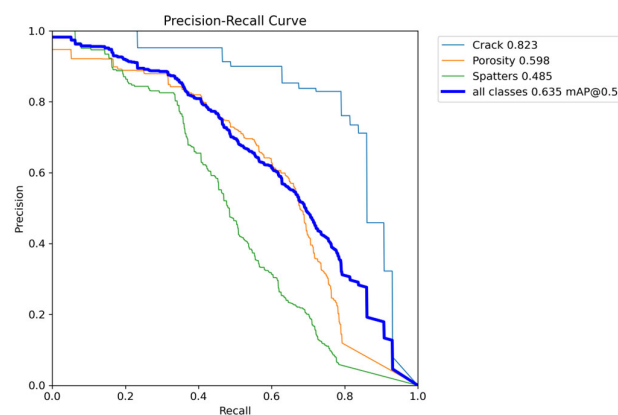


Figure 13. Per-class Precision–Recall curves of WID-YOLO11 on the independent test set.

Per-class metrics stratified by object size (small, medium, large) are not separately reported, as the dataset is dominated by medium-scale defects and the current evaluation pipeline does not output COCO-style size-specific AP by default. Size-stratified evaluation on more diverse datasets is left for future work.

4.6.4. Comparison of Detection Predictions

To provide a qualitative perspective on the proposed model's detection behaviour, three samples drawn from distinct defect scenarios are selected and presented in Figure 14. The figure is organized into three subfigures: (a) the input images, (b) the predictions generated by YOLO11n, and (c) the predictions generated by WID-YOLO11. Each subfigure contains three columns corresponding to three distinct defect scenarios: small-scale porosity with indistinct boundaries (left column), densely distributed spatter and porosity defects (middle column), and an elongated crack on a curved metallic surface (right column).

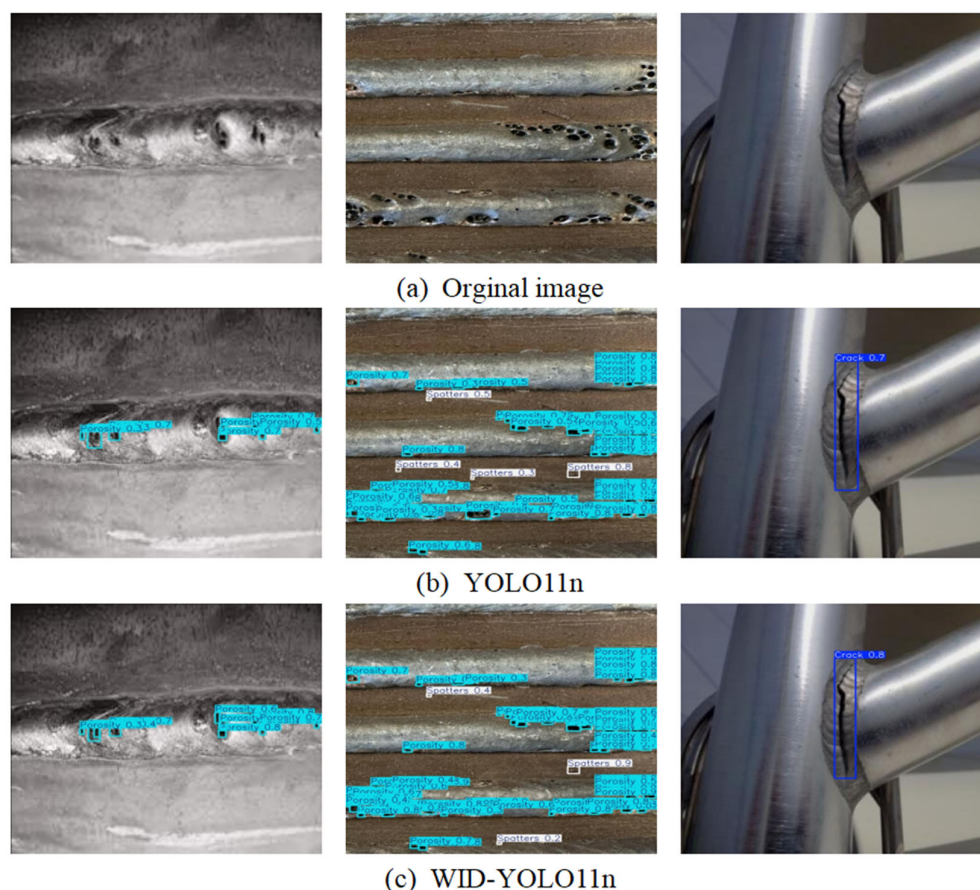


Figure 14. Detection results of YOLO11n and WID-YOLO11 on three representative defect samples.

As can be observed from Figure 14, WID-YOLO11 produces generally more stable detection results across different defect types. Taking the porosity sample in the left column as an example, compared with YOLO11n in sub-figure (b), WID-YOLO11 in sub-figure (c) achieves higher prediction confidence and successfully detects one tiny porosity defect in the middle of the weld seam that YOLO11n fails to identify, reflecting improved sensitivity to weak and small-scale targets. For the densely distributed defect sample (middle column), WID-YOLO11 obtains slightly higher confidence scores, focuses more precisely on the defects, and shows some improvement in suppressing background interference in cluttered regions. For the crack sample on the metallic surface (right column), both models can detect the target, while WID-YOLO11 produces a more reliable detection with higher confidence.

4.6.5. Activation Heatmap Comparison

Figure 15 visualizes the class activation maps for the same three representative samples to further examine the internal feature focus of each model. The figure is organized into three subfigures: (a) the input images, (b) the heatmaps from YOLO11n, and (c) the heatmaps from WID-YOLO11. The three columns correspond to the same defect scenarios as in Figure 14.

Warmer colors indicate regions of stronger model attention. As shown in Figure 15, WID-YOLO11 produces more concentrated activation responses at actual defect locations, whereas YOLO11n exhibits more dispersed attention patterns. This difference is particularly evident in the tiny porosity and densely distributed spatter cases, where WID-YOLO11 focuses more precisely on the defect regions. These observations, together with

the detection results in Figure 14, confirm that WID-YOLO11 achieves more reliable attention to defect areas across different defect types and background conditions.

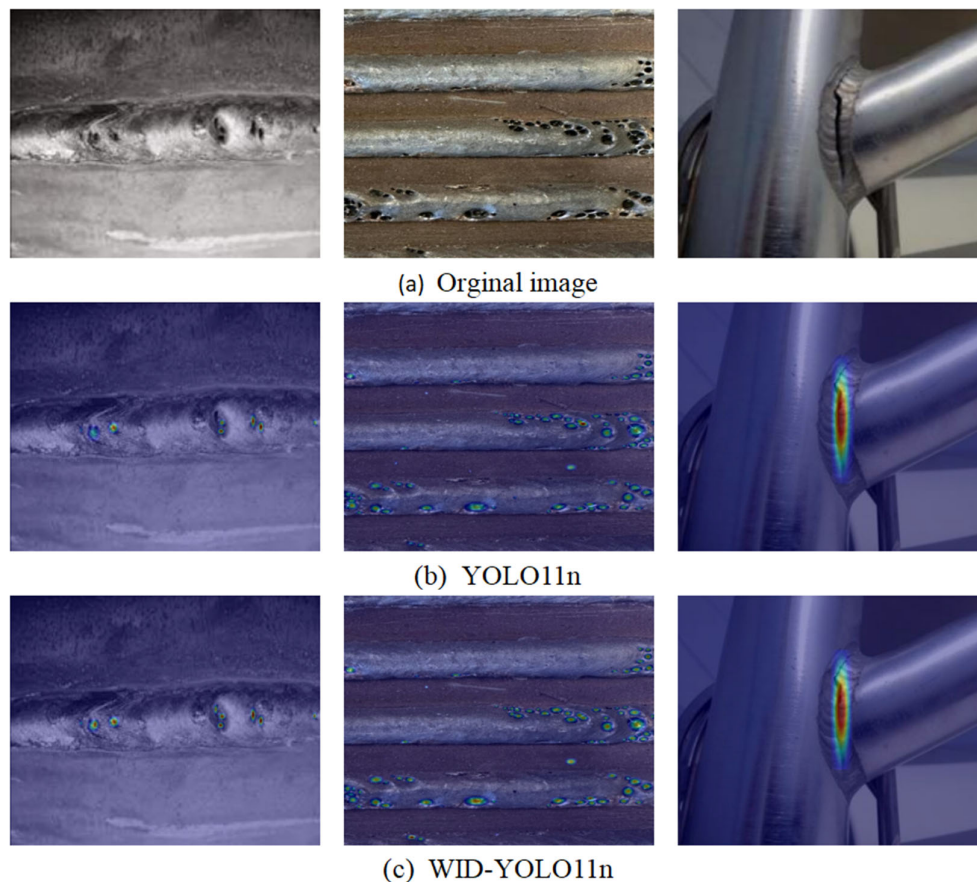


Figure 15. Activation heatmaps of YOLO11n and WID-YOLO11 on three representative defect samples.

5. Conclusions

Weld surface defect detection is complicated by pronounced scale variation, heterogeneous defect morphologies, and cluttered weld backgrounds. To tackle these difficulties, this study develops WID-YOLO11 by selectively augmenting YOLO11n with three complementary modules. The C3k2_LFA-WTConv module leverages learnable wavelet filters to split feature maps into frequency subbands and applies LFA attention to assign differentiated weights to each subband, leveraging the class-conditioned subband energy distributions of different defect types to improve discriminative feature extraction. The IMSDA module broadens the original three-rate dilation set to four rates (1, 2, 3, 4), pushing the maximum equivalent receptive field from 7×7 to 9×9 and enabling richer multi-scale local focus on medium-scale defects at a fraction of the cost of global attention. The DySample dynamic upsampling module supersedes fixed-interpolation upsampling by jointly learning content-aware sampling offsets, which concentrates the upsampling process on defect boundary regions and substantially reduces detail loss during feature fusion. On a publicly available weld defect dataset, WID-YOLO11 records gains of 4.5, 10.4, and 3.4 percentage points in mAP@0.5, Recall, and mAP@0.5:0.95 over YOLO11n, respectively, confirming the effectiveness of the three proposed enhancements.

Notwithstanding these encouraging results, some limitations warrant attention. The additional modules introduce a moderate rise in parameters (+0.33 M) and computation (+1.5 GFLOPs) compared with the YOLO11n baseline. The current evaluation is limited to

three surface defect categories on a single-source dataset; independent deployment testing on production lines has not been conducted. Subsequent research will investigate lightweight optimization techniques—including structured pruning and knowledge distillation—to shrink the model footprint without sacrificing detection accuracy. The current real-time evaluation is conducted on a desktop-grade RTX 3070 Ti GPU; validation on representative edge hardware is left for future engineering work.

Author Contributions: Conceptualization, Z.L. and S.L. (Songsong Li); Data curation, Z.L., W.L. and M.L.; Formal analysis, Z.L., W.L. and X.C.; Methodology, Z.L., S.L. (Songsong Li) and X.C.; Writing—original draft, Z.L.; Writing—review & editing, S.L. (Songsong Li) and X.C.; Software, Z.L. and W.L.; Validation, W.L., M.L., S.L. (Shidong Li) and X.C.; Supervision, S.L. (Songsong Li) and X.C.; Funding acquisition, S.L. (Songsong Li); Resources, S.L. (Songsong Li) and S.L. (Shidong Li). All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China (51778104) and the Department of Education of Liaoning Province (DL202005).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to privacy concerns.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. MaoBayır, M.; Yücel, E.; Kaya, T.; Yıldırım, N. Spot Welding Parameter Tuning for Weld Defect Prevention in Automotive Production Lines: An ML-Based Approach. *Information* **2023**, *14*, 50. <https://doi.org/10.3390/info14010050>.
2. Zuo, Y.; Li, W. Experimental and numerical analysis of mechanical properties of steel beam-column joints with initial defects. *KSCE J. Civ. Eng.* **2026**, *30*, 100466. <https://doi.org/10.1016/j.kscej.2025.100466>.
3. Hou, W.; Zhang, D.; Wei, Y.; Guo, J.; Zhang, X. Review on computer aided weld defect detection from radiography images. *Appl. Sci.* **2020**, *10*, 1878. <https://doi.org/10.3390/app10051878>.
4. Drinkwater, B.W.; Wilcox, P.D. Ultrasonic arrays for non-destructive evaluation: A review. *NDT E Int.* **2006**, *39*, 525–541. <https://doi.org/10.1016/j.ndteint.2006.03.006>.
5. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. <https://doi.org/10.1038/nature14539>.
6. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. <https://doi.org/10.1145/3065386>.
7. Zhang, Z.; Huang, X.; Wei, D.; Chang, Q.; Liu, J.; Jing, Q. Copper Nodule Defect Detection in Industrial Processes Using Deep Learning. *Information* **2024**, *15*, 802. <https://doi.org/10.3390/info15120802>.
8. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>.
9. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. <https://doi.org/10.1109/CVPR.2016.91>.
10. Jocher, G.; Qiu, J. Ultralytics YOLO11. 2024. Available online: <https://docs.ultralytics.com/models/yolo11> (accessed on 21 April 2026).
11. Gan, Z.; Wang, C.; Jia, Z. Improved YOLO11 model for weld surface defect detection. *Trans. China Weld. Inst.* **2026**, *47*, 22–32. (In Chinese).
12. Chen, Q.X.; Shi, Z.M.; Dai, L.; Li, H.X.; Zhang, F.X. A method for weld surface defect detection based on an improved YOLOv5s algorithm. *Trans. China Weld. Inst.* **2026**, *47*, 121–128. (In Chinese).
13. Xu, Z.Y.; Wang, C.; Lei, Y.J.; Yang, Y.; Yin, L.; Guo, Y.; Han, L.L.; Wang, Y. DualSPD-YOLO: A dual-stage SPD-enhanced network with TripletAttention for high-speed weld defect detection in thin-walled metal tubes. *Measurement* **2026**, *272*, 121001. <https://doi.org/10.1016/j.measurement.2026.121001>.

14. Wang, X.; Liu, T. High-precision weld defect detection based on improved YOLOv11. *Appl. Res. Comput.* **2025**, *42*, 3192–3200. (In Chinese).
15. Li, T.; Zhao, Z.; Sun, C.; Cheng, L.; Chen, X.; Yan, R.; Gao, R.X. WaveletKernelNet: An Interpretable Deep Neural Network for Industrial Intelligent Diagnosis. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 2302–2312. <https://doi.org/10.1109/TSMC.2020.3048950>.
16. Wei, Y.; Wang, H. Wavelet Integrated Attention Network with Multi-Resolution Frequency Learning for Mixed-Type Wafer Defect Recognition. *Eng. Appl. Artif. Intell.* **2023**, *121*, 105975. <https://doi.org/10.1016/j.engappai.2023.105975>.
17. Wang, J.; Chen, K.; Xu, R.; Liu, Z.; Loy, C.C.; Lin, D. CARAFE: Content-Aware ReAssembly of FEatures. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3007–3016. <https://doi.org/10.1109/ICCV.2019.00310>.
18. Finder, S.E.; Amoyal, R.; Treister, E.; Freifeld, O. Wavelet Convolutions for Large Receptive Fields. In *Proceedings of the European Conference on Computer Vision (ECCV)*; Springer: Cham, Switzerland, 2024; pp. 363–380. https://doi.org/10.1007/978-3-031-72949-2_21.
19. Jiao, J.; Tang, Y.M.; Lin, K.Y.; Gao, Y.; Ma, A.J.; Wang, Y.; Zheng, W.S. DilateFormer: Multi-Scale Dilated Transformer for Visual Recognition. *IEEE Trans. Multimed.* **2023**, *25*, 8906–8919. <https://doi.org/10.1109/TMM.2023.3243616>.
20. Liu, W.; Lu, H.; Fu, H.; Cao, Z. Learning to Upsample by Learning to Sample. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 2–6 October 2023; pp. 6004–6014. <https://doi.org/10.1109/ICCV51070.2023.00554>.
21. Sukma, A.W. Welding Defect Object Detection Dataset. Kaggle. 2025. Available online: <https://www.kaggle.com/datasets/sukmaadhijaya/welding-defect-object-detection> (accessed on 12 December 2025).
22. ISO 6520-1; Welding and Allied Processes—Classification of Geometric Imperfections in Metallic Materials—Part 1: Fusion Welding. International Organization for Standardization: Geneva, Switzerland, 2007. Available online: <https://www.iso.org/obp/ui/#iso:std:iso:6520:-1:ed-2:v1:en> (accessed on 21 April 2026).
23. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440. <https://doi.org/10.1109/CVPR.2015.7298965>.
24. Zhao, Y.; Lv, W.; Xu, S.; Wei, J.; Wang, G.; Dang, Q.; Liu, Y.; Chen, J. DETRs Beat YOLOs on Real-time Object Detection. In Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 17–21 June 2024; pp. 16965–16974. <https://doi.org/10.1109/CVPR52733.2024.01605>.
25. Jocher, G. YOLOv5. 2020. Available online: <https://github.com/ultralytics/yolov5> (accessed on 21 April 2026).
26. Jocher, G.; Chaurasia, A.; Qiu, J. Ultralytics YOLOv8. 2023. Available online: <https://docs.ultralytics.com/models/yolov8> (accessed on 21 April 2026).
27. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 10–15 December 2024; pp. 107984–108011. <https://doi.org/10.52202/079017-3429>.
28. Tian, Y.; Ye, Q.; Doermann, D. YOLOv12: Attention-Centric Real-Time Object Detectors. *arXiv* **2025**, arXiv:2502.12524. <https://arxiv.org/abs/2502.12524>.
29. Lei, M.; Li, S.; Wu, Y.; Hu, H.; Zhou, Y.; Zheng, X.; Ding, G.; Du, S.; Wu, Z.; Gao, Y. YOLOv13: Real-Time Object Detection with Hypergraph-Enhanced Adaptive Visual Perception. *arXiv* **2025**, arXiv:2506.17733. <https://arxiv.org/abs/2506.17733>.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.