algorithms

MDPI

Article

ID-MSNet: An Enhanced Multi-Scale Network with Convolutional Attention for Pixel-Level Steel Defect Segmentation

Mohammadreza Saberironaghi, Jing Ren and Alireza Saberironaghi *

Department of Electrical, Computer and Software Engineering, Ontario Tech University, Oshawa, ON L1G 0C5, Canada; mohammadreza.saberironaghi@ontariotechu.net (M.S.); jing.ren@ontariotechu.ca (J.R.)
* Correspondence: alireza.saberironaghi@ontariotechu.ca

Abstract

Automated pixel-level detection of steel surface defects is a critical challenge in manufacturing quality control, complicated by the variation in defect size and shape, low contrast with background textures, and the diversity of defect patterns. This paper proposes ID-MSNet, an enhanced version of the UNet3+ architecture, designed specifically for the segmentation of three common steel surface defect types: inclusions, patches, and scratches. The proposed architecture introduces three targeted modifications: (1) a multi-scale feature learning module (MSFLM) in the encoder that uses dilated convolutions at multiple rates to capture contextual features across different scales, combined with DropBlock regularization and batch normalization to improve generalization; (2) an improved down-sampling (IDS) module that replaces standard max-pooling with learnable strided convolutions fused via 1×1 convolution, preserving richer feature representations; and (3) a convolutional block attention module (CBAM) integrated into the skip connections to selectively focus the model on spatially and channel-wise relevant defect regions. Experiments on the publicly available SD-saliency-900 dataset demonstrate that ID-MSNet achieved an 86.19% mIoU, outperforming all compared state-of-the-art segmentation models while using only 6.7 million parameters—approximately 75% fewer than the original UNet3+. These results establish ID-MSNet as a strong and efficient baseline for steel surface defect segmentation, with potential applicability to automated quality inspection in broader manufacturing contexts.

Keywords: defect detection; surface defect detection; steel defect detection; defect segmentation; UNet3+; attention mechanism; deep learning

Check for updates

Academic Editor: Igor Aizenberg

Received: 8 March 2026
Revised: 24 March 2026
Accepted: 26 March 2026
Published: 9 April 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) license.

1. Introduction

Surface defects in manufactured goods compromise both product quality and financial efficiency [1–3]. Despite advancements, a large portion of the manufacturing sector still relies on human-led manual inspections for detecting these defects. Such manual processes are not only time-intensive but also fall short in terms of efficiency [4]. Due to these limitations, there has been an escalating focus on automated methods for surface defect detection, which are becoming increasingly effective in industrial contexts.

Strip steel plays a crucial role in various industries, including construction engineering, aerospace, and automotive production. Assessing the quality of this material is vital for ensuring the longevity of the end products in these sectors. Recently, a number of

computational vision and machine learning techniques have been put forward to automate the process of detecting surface defects in steel [5–7]. However, as depicted in Figure 1, there are still three primary categories of challenges that any proposed model for defect detection is anticipated to address:

(a) Varied defect types: Steel surfaces can exhibit a wide range of defect types such as corrosion, pitting, and scratches. The diversity in defect forms necessitates an adaptable detection approach capable of identifying multiple types of irregularities.
(b) Background similarity: The background textures on steel surfaces can closely resemble defects, leading to challenges in accurate identification. Advanced image processing techniques are required to discern between actual defects and similar-looking background textures.
(c) Varying defect sizes: Defects on steel surfaces can vary significantly in size, from microscopic inclusions to larger dents. A multi-scale detection approach is essential to accurately identify defects across this broad size spectrum.

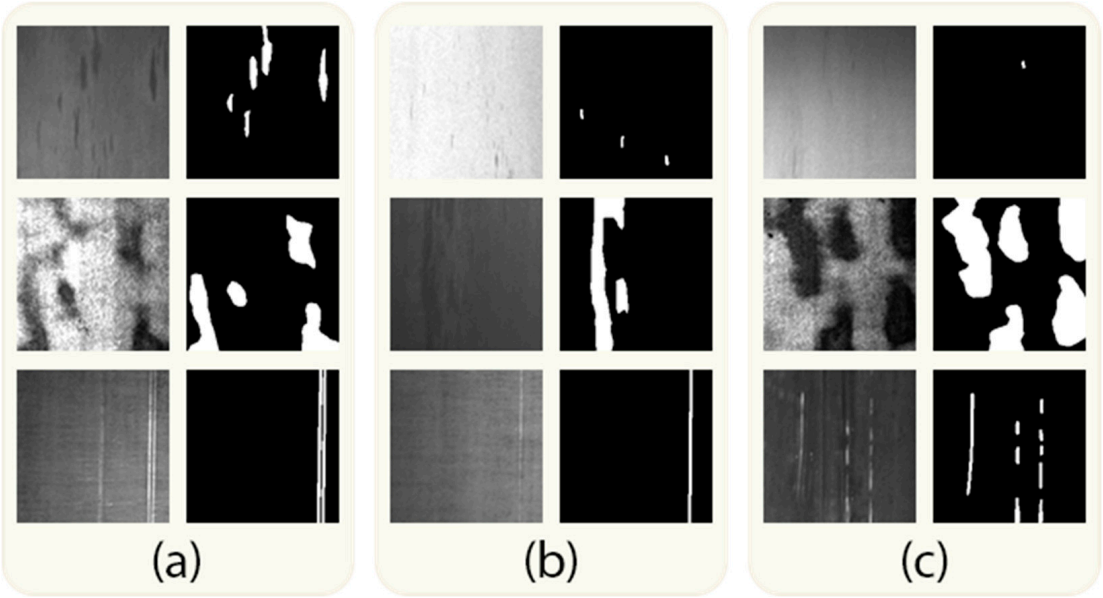


Figure 1. Challenges in defect detection on steel surfaces (white regions in the ground truth signify defect locations): (**a**) varied defect types, (**b**) background similarity, and (**c**) varied defect sizes.

Recent research emphasizes the use of computer vision models and techniques for detecting surface flaws in strip steel. These approaches can broadly be classified into two categories: traditional methods and those based on deep learning. In the traditional methods category, there are statistical, spectral, and model-based techniques. Specifically, statistical methods identify imperfections by examining the distribution of pixels between the defect area and the background texture [2]. Ref. [8] investigated the pixel distribution in hot-rolled steel defects to optimize segmentation thresholds for detection. Additionally, Ref. [9] used a localized, adaptive threshold to address non-uniform surface features caused by scales. While statistical methods perform well on high-contrast test images, they struggle with the low-contrast images often found in production lines. The low luminance makes defect boundaries indistinct, challenging the effectiveness of these methods. To address this, Ref. [10] combined Gabor filters with morphological features to detect pinholes in defective steel. Likewise, Ref. [11] used optimized Gabor filters to find the minimal energy separation between defects and the background, aiding in segmentation. Furthermore, Ref. [12] introduced a wavelet-based algorithm to isolate defects from textured backgrounds. Spectral methods are effective but require extensive fine-tuning of parameters like Gabor

and wavelet filters. To bypass the challenges of parameter tuning, various model-based techniques have been introduced. For instance, Ref. [13] employed a Markov random field model to reconstruct sheet-metal images for easier analysis. The model-based approaches struggle to identify defects with gradual intensity changes or low contrast, regardless of being supervised or semi-supervised.

Consequently, traditional methods often struggle with challenging surface defects, while deep learning-based approaches show superior performance due to their ability to automatically extract key features. For example, Ref. [14] proposed a specialized VSD network for steel defect classification, outperforming standard architectures like VGG19 and ResNet. Moreover, Ref. [15] used a semi-supervised learning approach with generative adversarial networks to generate unlabeled defect samples for improved classification. While these classification-centric methods deliver high accuracy rates, they do not provide detailed information on the precise location and shape of the defects. To address this gap, research has been undertaken on automated defect detection techniques. For example, Ref. [16] employed deformable convolutions in place of traditional convolutions within Faster R-CNN, thereby enhancing its capability to detect smaller object defects. However, while such object detection-based methods can accurately locate defects, the boundary and shape of the defect still cannot be obtained. To overcome this issue, Ref. [17] introduced a saliency detection network that uses channel-weighted and residual decoder blocks for more precise defect identification. While semantic segmentation and saliency detection have improved edge shape capture, more research is needed to refine detection boundaries, enhance feature sets, and bolster model robustness.

In this study, we introduce an enhanced version of UNet3+ model [18], enriched with multiscale feature learning and attention mechanisms, specifically designed for automated steel defect segmentation. The ID-MSNet model, consisting of 6.7 million parameters, has yielded promising results largely attributable to its simplified but effective structure. This stands in contrast to the original UNet3+ model [18], which encompasses around 26.9 million parameters. Furthermore, comparative evaluations indicate that our model outperforms existing state-of-the-art approaches in terms of performance.

Our major contributions are summarized as follows:

(a) We introduced a novel encoder structure that incorporates varying dilation rates and DropBlock regularization to capture features at multiple scales, improving the model's adaptability and robustness.
(b) We augmented the traditional max-pooling operation in UNet3+ with strided Conv2D layers, followed by concatenation and a 1×1 convolution, to retain critical feature information and enhance predictive power.
(c) We Integrated the convolutional block attention module (CBAM) into the skip connections of UNet3+, refining the model's focus on features crucial for defect detection.
(d) In experimental studies, the proposed ID-MSNet model showed superior performance compared to previous studies and state-of-the-art models. In addition, the proposed structure is shown to be effective in detecting surface defects such as varied defect types, background similarity, and varied defect sizes.

The rest of the paper has been organized as follows. Section 2 provides a review of related work in the area of steel defect detection. Section 3 presents sub-modules used in the proposed ID-MSNet architecture. Section 4 presents the dataset, evaluation metrics, implementation details, and numerical and visual results of our experimental studies. Section 5 presents ablation experiments, comparison of the proposed model with the previous studies, and an analysis of failure cases. Finally, we conclude the work in Section 6.

2. Related Work

Initial efforts in the literature to identify surface defects largely relied on basic image processing techniques such as thresholding, morphological operations, and Fourier transforms [12]. For instance, Ref. [19] employed basic image processing tasks like noise filtering, gradients, and morphological operations to determine optimal thresholding values, achieving better outcomes compared to other thresholding models like OTSU. Moreover, Ref. [13] introduced a dynamic thresholding technique based on pixel intensity probability distribution, which showed promise in defect identification when tested on industrial steel images from a hot rolling machine. Such thresholding-focused approaches [13,19] are notably susceptible to noise due to their reliance on pixel intensity distributions. On a related note, Ref. [20] identified defects on uniform surfaces using the KNN classifier, improving results by incorporating elementary image processing tasks like edge detection and morphological processing. Ref. [21] proposed an entity sparsity pursuit (ESP) method for defect detection, using local binary patterns (LBPs) for feature extraction. Post-extraction, the image is segmented into perceptual and compact super-pixels, with defects subsequently identified through the ESP algorithm. In a different approach, Ref. [22] presumed the surface to be homogeneous and then isolated defect zones deviating from this homogeneous, repetitive pattern. While these studies have generated effective outcomes, they suffer from the limitations inherent in manual feature extraction techniques: they are time-consuming and offer limited generalizability. Moreover, these models often require recalibration for deployment on diverse surfaces.

Recently, deep learning-based image segmentation techniques have gained prominence, owing to their high precision, computational efficiency, and ease of use. Unlike traditional methods that require manual feature extraction and separate classification, deep learning unifies these processes, performing feature extraction and defect identification simultaneously. These models are capable of automatically generating the required features from the training data, thus eliminating the need for expert-designed feature sets. Prominent among semantic segmentation architectures are encoder–decoder frameworks such as SegNet [23] and U-Net [22], as well as pyramid pooling structures like Deeplabv3+ [20] and PSPNet [24]. Building upon these foundational architectures, numerous enhancements have been introduced to improve model robustness and segmentation accuracy. These advancements can be broadly classified into three categories: methods based on attention-centric techniques, feature aggregation, and conditional random field (CRF)-based methods.

Feature aggregation is pivotal in boosting the accuracy of models in semantic segmentation tasks. Currently, there are mainly two types of feature aggregation techniques: multi-scale and multi-level. Multi-scale feature aggregation methods like the inception block [21] achieve feature fusion by concatenating features extracted using various sizes of convolution kernels—namely 1×1, 3×3, and 5×5. This approach allows the model to capture semantic information at different scales, providing a richer representation of the input image. On the other hand, multi-level feature aggregation techniques focus on fusing high-level and low-level features. This is generally achieved through upsampling and downsampling operations to harmonize the differing resolutions of feature maps. This allows the model to benefit from both the finer-grained details and the broader context within the image. For instance, Ref. [25] designed a robust feature pyramid complemented by a tailored pyramid pooling module to fully exploit multi-level semantic features, yielding promising results particularly in the context of remote sensing image segmentation. FPN is another critical approach in multi-level feature aggregation. It consists of a bottom-up and a top-down pathway, leveraging convolution kernels and concatenation operations to extract and fuse features effectively. Although both multi-scale and multi-level feature

aggregation techniques have improved model accuracy, they come with drawbacks. The primary challenges include an increase in the number of network parameters and a decrease in detection efficiency due to the additional computational complexity. In summary, while feature aggregation methods have achieved progress in the accuracy of semantic segmentation tasks, optimizing them to be both efficient and effective is an ongoing area of research. Inspired by the human vision system, where attention mechanisms allow us to focus on relevant information while ignoring the irrelevant, attention-based methods in deep learning have become increasingly popular. These approaches generally fall into three categories of channel attention, spatial attention, and non-local attention. Ref. [26] introduced the SE module to analyze the interdependencies between feature channels and accordingly reweight them, focusing more on the most informative channels. Ref. [27] used residual skip connections to emphasize high-frequency features while disregarding the low-frequency ones. However, this method mainly focuses on channel-level features, ignoring spatial relationships in the image. To overcome the limitations of channel-only attention, a CBAM was designed to consider both channel and spatial dimensions simultaneously [28]. Ref. [29] extended CBAM by incorporating self-attention mechanisms based on non-local auto-correlations, simplifying some of the operations required in CBAM. Although these attention-based models have shown improvements in feature extraction and representation, they are not without downsides: non-local-based methods, in particular, add a considerable number of parameters to the network. This increases computational requirements, potentially making real-time applications more challenging. Thus, while attention mechanisms offer a promising avenue for enhancing the feature representation capabilities of semantic segmentation models, more work is needed to balance their benefits against the costs in terms of computational efficiency and complexity. One of the limitations of conventional semantic segmentation models is the inability to capture sufficient scene-level semantic context, making it challenging to delineate object boundaries clearly. To address this, many research works have utilized conditional random fields (CRFs) in the post-processing stage to refine segmentation masks: Ref. [30] introduced a novel approach where a convolutional neural network (CNN) is trained to construct the potentials of superpixels. These potentials are then refined using CRF parameters that have been optimized by a structured support vector machine (SVM). The method has shown promise in producing improved segmentation masks.

3. Proposed Model

In this work, we introduce an enhanced version of the UNet3+ model, designed specifically for steel defect detection tasks. Our novel model incorporates three main modifications aimed at improving feature extraction, downsampling operations, and skip connections. The general structure of the ID-MSNet model is given in Figure 2.

The major components used in these parts are:

- In the encoder part of the architecture, we integrate a multiscale feature learning module (MSFLM). This module is designed to capture multi-scale contextual information, and it is further strengthened by the inclusion of DropBlock, a regularization technique that improves the model's generalization capabilities.
- We re-engineer the downsampling operations traditionally carried out by max-pooling layers in the original UNet3+ model. In our improved version, we adopt a hybrid approach that fuses max-pooling with strided convolutions. The resulting features from both operations are then concatenated and subjected to a 1×1 convolution, enhancing the feature representation power of the model.
- We incorporate a convolutional block attention module (CBAM) into the skip connections of the architecture. This module refines the feature maps by focusing on

essential regions, thus making the model more sensitive to the nuances of defects in steel surfaces.

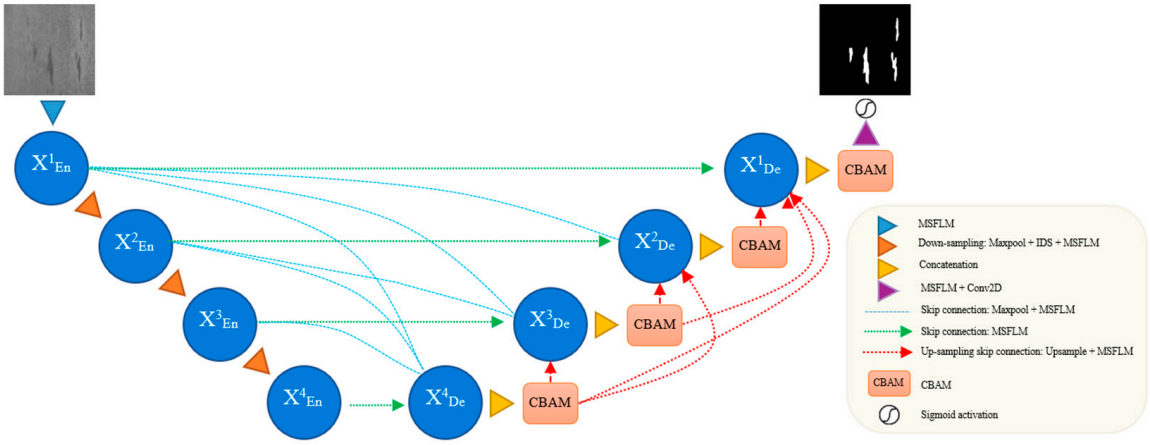


Figure 2. The architecture of the ID-MSNet.

These modifications collectively contribute to the improved performance of our ID-MSNet model over the original UNet3+ and other state-of-the-art models in steel defect detection tasks.

3.1. MultiScale Feature Learning Module (MSFLM)

Convolutional neural network (CNN)-based architectures, particularly in the field of defect detection, have long grappled with the trade-off between complexity and performance. Existing architectures like standard U-Net or its variants primarily use a sequence of convolutional layers, ReLU activations, and batch normalization to extract features. Researchers have often sought to enhance these architectures by increasing the number of layers to capture more complex and semantic features [31–33]. However, this comes with its own set of challenges:

(a) Vanishing Gradient Problem:
The addition of more layers, while aiding in feature extraction, exacerbates the vanishing gradient problem, making it difficult for the network to learn effectively.

(b) Computational complexity:
As each convolutional operation adds new parameters to be learned, the computational cost of these deep networks becomes increasingly prohibitive.

(c) Data scarcity:
In the domain of defect detection, datasets are often limited in size [28]. This paucity of data makes it challenging to train deep networks, which are parameter-heavy and require large amounts of data for effective training.

To address these challenges, this study introduces the multiscale feature learning module (MSFLM), which is incorporated into a UNet3+ based architecture. The multiscale feature learning module (MSFLM) is designed to capture contextual information at different scales to improve the defect detection capability of our UNet3+ based model. The MSFLM module replaces the standard convolutional block in the original UNet3+ architecture. The primary objective is to capture multi-scale features from the input using dilated convolutions, and then fuse them to form a more comprehensive feature representation. The structure is illustrated in Figure 3.

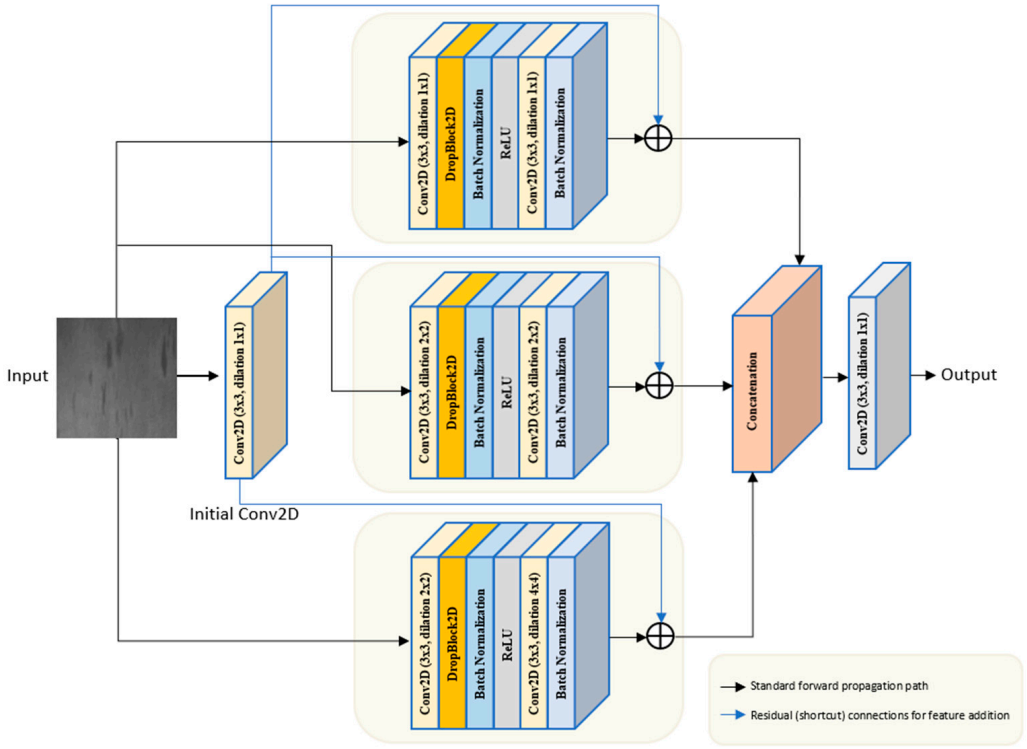


Figure 3. The structures of the multiscale feature learning module (MSFLM).

As can be seen in the Figure 3. The MSFLM consists of three components:

3.1.1. Initial Convolution Layer

The input is first passed through a convolutional layer with a 1×1 kernel size, generating a feature map that acts as the base for the subsequent operations. This is denoted as initial conv2D in the figure.

3.1.2. Dilated Convolution Blocks

MSFLM consists of three primary blocks, each consisting of a set of Conv2D layers with varying dilation rates. Each block applies dilated convolutions to the input with the objective to capture features at different scales without increasing the receptive field size or the number of parameters drastically. DropBlock regularization is integrated within each block to enhance generalization and fortify the model's robustness. Alongside this, batch normalization and ReLU activation functions are employed in every pathway. These not only normalize the feature distributions but also introduce non-linearity into the system. Additionally, residual connections play a crucial role. After the generation of feature maps post each block, they are element-wise added to the initial feature map, which is a technique known as residual connection. This step is essential in combating the vanishing gradient problem, thereby improving the network's capacity to learn effectively.

3.1.3. Feature Concatenation

Finally, the output feature maps from each block are concatenated along the channel axis, followed by a ReLU activation function. This results in a composite feature map that embodies multi-scale information.

By incorporating multi-scale feature learning in the encoder part of our UNet3+ architecture, we aim to improve the model's ability to detect various types of defects in steel with high accuracy and robustness.

3.2. Improved Downsampling Module (IDS)

Traditional downsampling techniques often rely on pooling layers, such as max-pooling or average pooling, to reduce the spatial dimensions of an image. While effective for computational reduction, these methods have two significant limitations: (1) They often discard valuable spatial information, leading to a potential loss in performance, especially when dealing with complex tasks like object detection or segmentation; and (2) They are not learnable, meaning that the pooling layers do not adapt during the learning process, possibly leading to sub-optimal solutions. To counter these challenges, especially the loss of valuable feature information during downsampling, this study utilized the improved downsampling (IDS) module. The improved downsampling (IDS) module is devised to perform the downsampling operation more efficiently than traditional methods. While conventional downsampling often employs pooling layers, the IDS module leverages a 3×3 convolution with strides of 2×2, followed by batch normalization and ReLU activation. This architecture not only reduces the spatial dimensions of the feature maps, but also learns representations through the convolution operation, thereby making the feature map more expressive for subsequent layers.

3.3. Convolutional Block Attention Module (CBAM)

Skip connections in CNN-based architectures like U-Net and its variants play an essential role in transmitting low-level features to deeper layers, helping in more accurate reconstruction of the output. However, conventional skip connections transmit all the features equally without prioritizing. This indiscriminate mixing can sometimes dilute the more critical features, leading to sub-optimal performance in tasks like image segmentation or object detection. To address this challenge, we employed the convolutional block attention module (CBAM) [28] designed specifically to enhance the quality of skip connections of the ID-MSNet model.

By acting on both channel and spatial aspects of feature maps, CBAM enhances the quality of the information transmitted through skip connections. This focus on essential channels and locations results in a more effective feature transmission, ultimately boosting the performance of our ID-MSNet model. Figure 4 demonstrates the structure of CBAM.

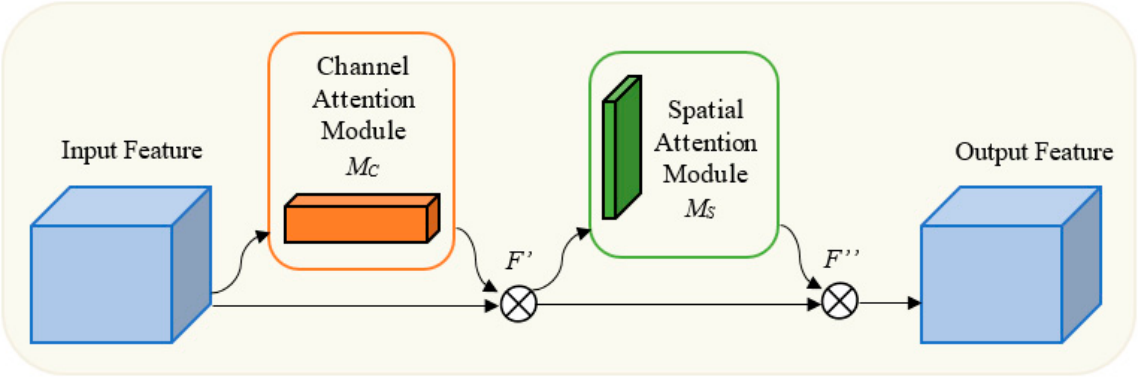


Figure 4. Convolutional block attention module (CBAM) structure.

The overall attention process can be summarized as:

$$F' = Mc\ (F) \otimes F \quad (1)$$

$$F'' = Ms\ (F') \otimes F' \quad (2)$$

where the symbol ⊗ denotes element-wise multiplication. F' is the outcome when the feature map is multiplied by the channel attention map, and F'' is the result when F' is further multiplied by the spatial attention map, producing the final output.

3.3.1. Channel Attention Module

The channel attention module plays a crucial role in isolating channels with relevant information for feature extraction. To minimize data loss during this process, the module employs both global average pooling and global max pooling layers to squeeze the feature map spatially. As illustrated in Figure 5, the global average pooling layer captures the general features, while the global max pooling layer focuses on the variations within the feature map. The synergistic combination of these two layers outperforms either layer used in isolation.

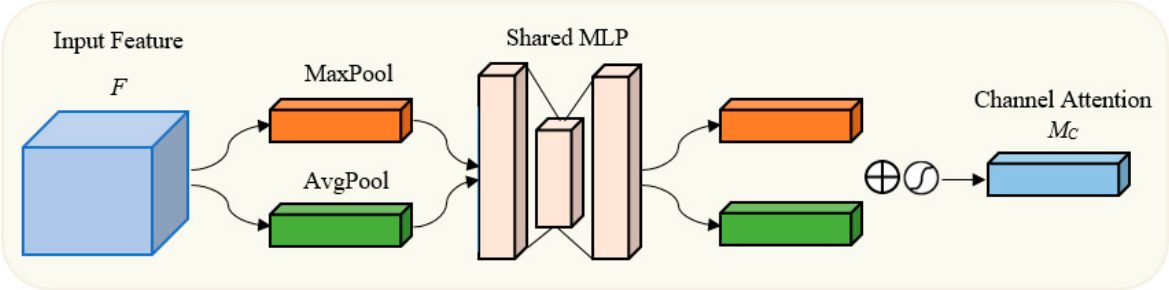


Figure 5. Channel attention module.

Following the pooling layers, the squeezed feature maps *FCavg* and *F*Cmax are passed through a shared multi-layer perceptron (MLP) network consisting of a single hidden layer. The MLP operates at a predetermined compression ratio to lower computational complexity and the number of parameters. A sigmoid activation function is then applied to produce the channel attention map $MC(F) \in RC \times 1 \times 1$, the process of which is as follows:

$$\begin{aligned} MC(F) &= \sigma(MLP(Avgpool(F)) + MLP(Maxpool(F))) \\ &= \sigma(W1(W0(FCavg)) + W1(W0(FCmax))) \end{aligned} \quad (3)$$

In this scenario, *FCavg* and *F*Cmax are the feature maps obtained after applying the global average pooling and global max pooling layers, respectively. The sigmoid function is represented by σ. The weight matrices W0 and W1 of the MLP have dimensions RC/r × C and RC × C/r, respectively, where r is the compression ratio.

3.3.2. Spatial Attention Module

As shown in Figure 6, the spatial attention module is shown to focus more on certain regions of the feature map that are more responsive compared to what the channel attention module targets. For the feature maps outputted by the channel attention module, both the global average pooling and global max pooling layers are employed to squeeze the feature maps into two 2D maps, *FSavg* and *F*Smax, along the channel dimension. This is conducted to accentuate regions of the feature map that contain crucial information.

After concatenating the two 2D feature maps, *FSavg* and *F*Smax, an effective feature map is formed, which then undergoes a convolution operation. Subsequently, a sigmoid function is applied to this convolved feature map to compute the spatial attention map, denoted as MS (F) with dimensions 1 × H × W.

$$\begin{aligned} Ms(F) &= \sigma(f7 \times 7([AvgPool(F); MaxPool(F)])) \\ &= \sigma\ (f7 \times 7([FSavg; FSmax])) \end{aligned} \quad (4)$$

In this setup, *FSavg* and *FSmax* are the channel-dimension-squeezed feature maps, while σ signifies the sigmoid function. The introduction of CBAM into the CNN brings an attention mechanism that enables both the channel and spatial modules to work collaboratively. These modules learn to focus on key information across both channel and spatial dimensions, thereby adaptively redistributing feature weights for improved performance.

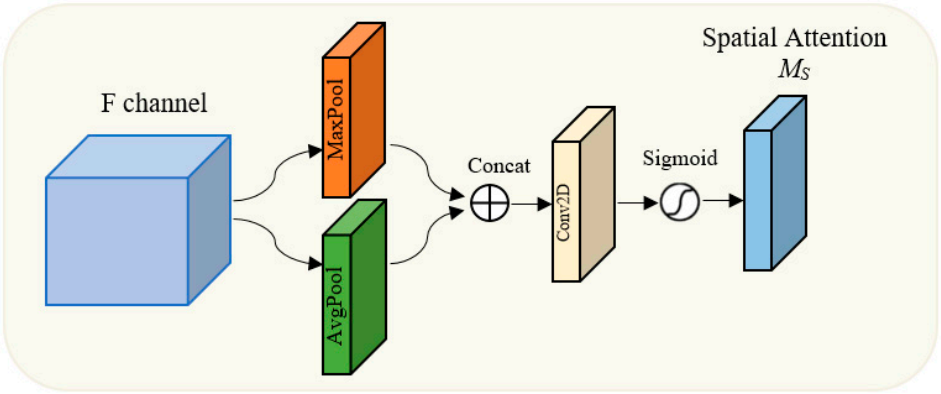


Figure 6. Spatial attention module.

4. Experimental Works and Results

In this section, we have organized the content to provide a comprehensive understanding of our research methodology. Details about the dataset used for this study are provided in Section 4.1. Section 4.2 explains the evaluation metrics used to assess the model's performance. The implementation and training particulars, including hyper-parameter settings, are covered in Section 4.3. Finally, Section 4.4 presents the experimental results, which include both visual and quantitative evaluations.

4.1. Dataset

In our experiments, we make use of the publicly available SD-saliency-900 [17] dataset as the benchmark for our proposed model. This dataset includes 900 steel surface defect images, segmented into three primary categories: inclusion, patches, and scratches. These categories correspond to standard metallurgical defect classifications: inclusions are localized, compact defects (point-type); scratches are elongated linear marks (line-type); and patches are extended area imperfections (surface-type). A selection of images from this dataset can be seen in Figure 7. Each category comprises 300 images, all of which have a default resolution of 200 $\times$ 200 pixels. In addition to the images, the dataset includes pixel-level labels for each defect type, allowing for training and a comprehensive evaluation of our ID-MSNet model. For this study, we standardized the dataset by resizing all the original images to a 224 $\times$ 224 pixel resolution, ensuring that they aligned with the UNet3+ architecture. The dataset was then partitioned into training, validation, and test sets, the details of which are provided in Table 1.

Table 1. Overview of the SD-saliency-900 dataset utilized in our experiments.

Dataset	Resolution	Images	Train	Validation	Test
SD-saliency-900 [17]	200 $\times$ 200	900	624	156	120

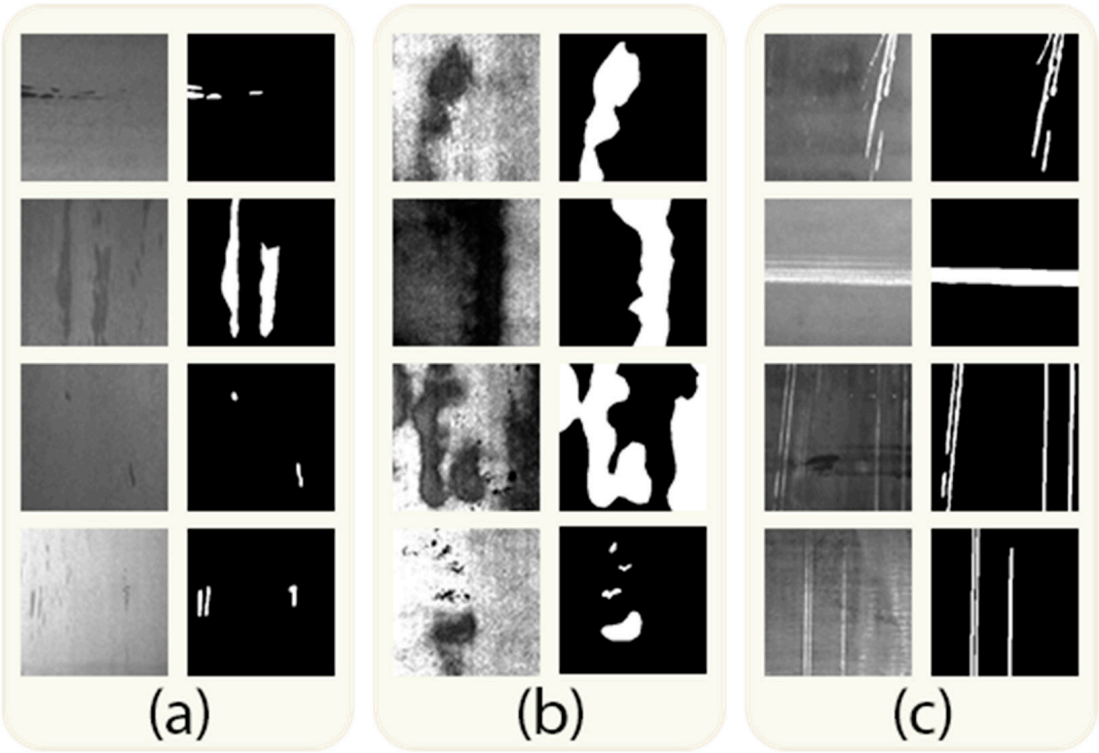


Figure 7. Sample images and their corresponding ground truths from the SD-saliency-900 dataset, showcasing the three primary categories of steel surface defects: (**a**) inclusion, (**b**) patches, and (**c**) scratches.

4.2. Evaluation Metrics

In evaluating our segmentation model's performance, we utilized a range of crucial metrics: precision, recall, F1-score, and mean intersection over union (mIoU). Precision examines the model's effectiveness in making accurate positive predictions for defect classification. Recall aims to minimize false negatives by assessing the model's capacity to spot all real defects. F1-score harmonizes precision and recall to offer a balanced overview of the model's accuracy and reliability. Mean intersection over union (mIoU) quantifies how well the model's predicted segmentation aligns with the actual ground truth labels, thus assessing its spatial accuracy. These metrics collectively offer a thorough assessment of how well the model performs in tasks related to steel defect detection. The equations for these metrics are detailed below:

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{5}$$

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \tag{6}$$

$$\text{F1} = 2 \times \text{Precision} \times \text{Recall}/(\text{Recall} + \text{Precision}) \tag{7}$$

$$\text{mIoU} = (\text{pr} \bigcap \text{GT})/(\text{pr} \bigcup \text{GT}) \tag{8}$$

In the equations used for performance evaluation, we use specific terms to clarify various outcomes during the segmentation process. When the model accurately detects and categorizes a defect, it is termed a true positive (TP). A false positive (FP) describes an incorrect defect classification. A false negative (FN) is an oversight, where a defect that should have been identified is missed. A true negative (TN) occurs when the model accurately determines that a region is not defective. These definitions are essential for interpreting Equations (5) and (6), which compute metrics like precision and recall. During

our empirical studies, the count of accurately predicted test samples matched the count of accurately labeled samples, signifying robust model performance.

4.3. Implementation Details and Training

In the following section, we delve into the hyperparameter selections optimized during the training phase. The architecture of our model was built on the TensorFlow and Keras frameworks, which are open-source platforms for deep learning. We utilized a batch size of 8 and conducted training over 100 epochs. Adam's optimizer was employed for network parameter updates. All computational experiments were carried out on an Ubuntu 20.04 system, equipped with an i9 9900 kf CPU and an RTX 2080 ti GPU. Additional computational capacity was provided by an NVIDIA 80 GB GPU card and 90 GB of RAM, operating within the Paperspace computing environment. In a practical deployment setting, the model would be integrated with industrial line-scan or area-scan cameras, with inference running on a GPU-equipped edge unit or server. The segmentation output would then feed into a downstream decision system to trigger quality control actions. Inference latency and memory footprint under production conditions were not benchmarked in this study and were identified as important directions for future evaluation.

4.4. Experimental Results

In this section, we present the results of our research, both in visual and numerical formats. Our model was compared against state-of-the-art models for performance evaluation, leveraging the SD-saliency-900 dataset. It is crucial to note that every model was subjected to the same training and testing conditions to ensure a balanced comparison. Uniform parameters were employed across the board during the training stage for all models examined in this research.

Figure 8 visually compares the sample ground-truth images from the test dataset of the SD-saliency-900 dataset with segmentation results from our model and other existing models. The original images and their ground-truth segmentations are shown in the first and second columns. Columns 3 to 8 showcase segmentation outcomes from models like UNet [22], LinkNet [34], FPN [35], ResUnet-a [36], Attention U-net [37], and PSPNet [24]. Our proposed UNet3+ model's results are highlighted in the ninth column. As illustrated in Figure 8, the SD-saliency-900 dataset encompasses a range of challenges in steel surface defect detection, including diverse types of defects, similarities in the background, and varying defect sizes. The first three rows in Figure 8 present sample images showcasing diverse types of defect problems. Based on the intersection over union (IoU) percentages obtained, our proposed model showed a demonstrable superiority across all three categories of defects—small inclusions, patches, and line-type scratches—when compared to state-of-the-art models. The performance of UNet and Attention U-Net varied significantly across defect types, suggesting that while they are versatile, they might not be the best-suited for all kinds of steel defects. LinkNet and FPN showed moderate performance but were outshone by more specialized architectures. Notably, FPN performed exceptionally well for small inclusion defects, highlighting its capability to capture multi-scale features effectively. ResUNet-a performed consistently strong across all defect types, but was still unable to beat our model. The residual connections seemed to help, but not enough to capture the highest IoU. PSPNet also showed strong performance, particularly with lines and patches, but was not as versatile as our model across different defect types. Consistently high performance across varied defect types proves the model's robustness and versatility. It is well-suited for dealing with different types of steel defects, whether they are inclusions, patches, or scratches. Rows 4, 5, and 6 of Figure 8 delve into the background similarity issue, highlighting the complexity of the defect detection task. Specifically, in the

challenging category of inclusion defects depicted in row 4, our model scored an impressive 90% IoU. This significantly eclipsed the second-best performer, LinkNet, by a 7 percentage point margin. For patches and scratches defects, our model consistently topped with IoUs of 89%, indicating its excellent capacity to handle a variety of defect types in complex backgrounds. While other models like LinkNet and FPN also showed strong results in certain categories, none displayed the across-the-board proficiency that our model did in this specific issue. In addressing the challenge of varied defect sizes, rows 7, 8, and 9 in Figure 8 reveal our model's exceptional performance. For small inclusion defects in row 7, our model registered an IoU of 87%, a striking leap from its closest competitor, FPN, which scored an IoU of 84%. This likely indicates a superior architectural configuration that excels in capturing features across different scales. In the category of patches defects, while other state-of-the-art models like LinkNet, FPN, and ResUnet-a did make it to the high-80s, our model pushed the bar even further with an IoU of 91%. The gap widened in the scratches category, where our model scored an IoU of 88%, notably higher than PSPNet's 86%. What stands out in these results is the model's ability to maintain this superiority across different types and sizes of defects. This suggests that our architecture has effectively learned to adapt to varied scales, likely incorporating multi-scale feature integration mechanisms more effectively than the compared models.

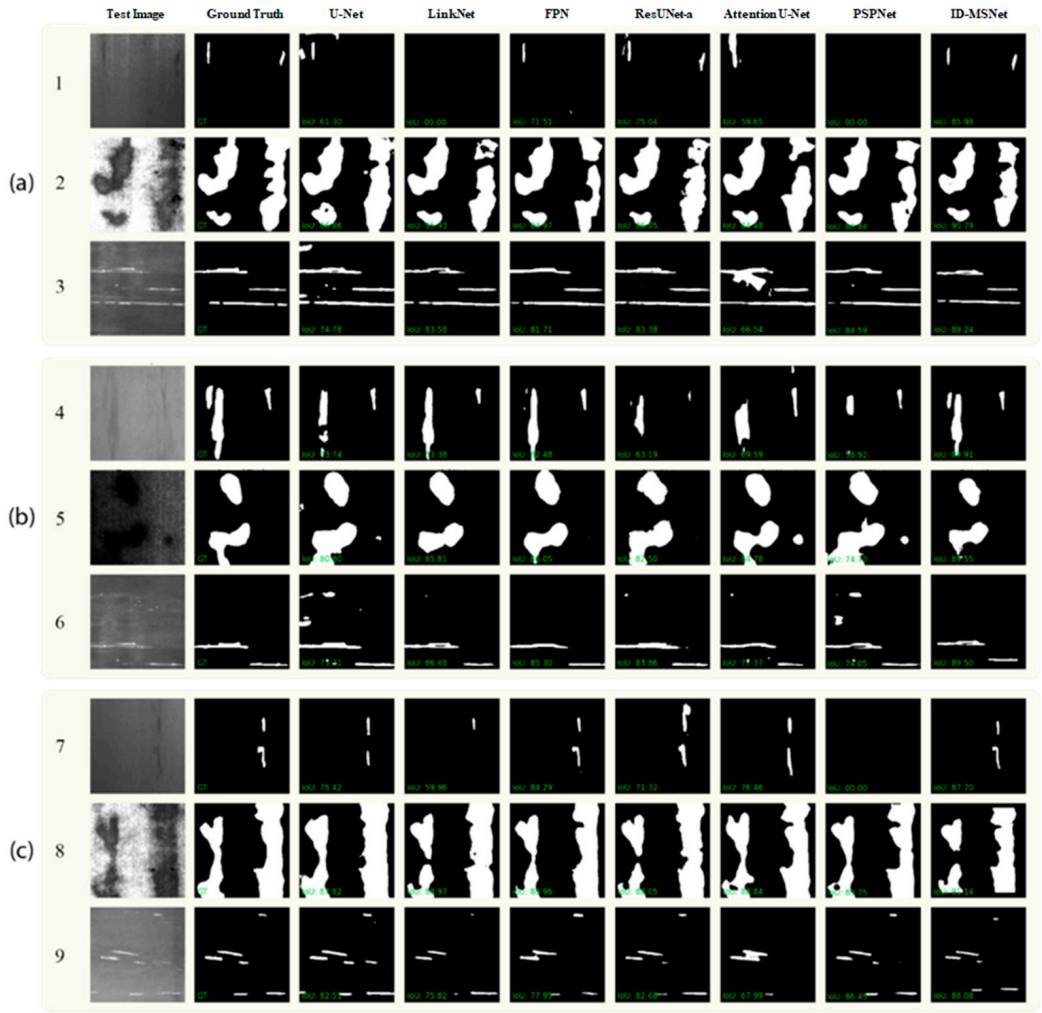


Figure 8. Visual outcomes from the sample test images from the SD-saliency-900 dataset, segmented into three categories: (**a**) diverse types of defects, (**b**) similarity in background, and (**c**) varying sizes of defects.

Table 2 presents the comprehensive performance metrics obtained from our experimental studies on steel defect detection. The proposed ID-MSNet model surpassed all state-of-the-art architectures with an F1-score of 87.91% and a remarkable mIoU of 86.19%. The closest competitor, LinkNet, achieved a commendable F1-score of 88.47% but fell behind in mIoU, registering only 83.99%. Although LinkNet demonstrated high precision, it did not maintain this advantage uniformly across all evaluation metrics, indicating possible limitations in handling the intricate challenges posed by varied defect types, sizes, and background similarities in steel defect detection. Interestingly, FPN, an architecture known for its ability to handle multi-scale information, recorded an F1-score of 86.10% and an mIoU of 82.40%. Despite its robust design, it did not outperform our proposed model. The ResUnet-a model, lauded for its residual connections and feature extraction capabilities, also trailed with an mIoU of 81.52%. Attention U-Net and PSPNet, although respected models, showed mIoUs of 78.21% and 80.26%, respectively, revealing their limitations in this specific application. The original UNet, despite being the pioneering architecture in this domain, posted the lowest mIoU of 76.89%. Its high recall rate implies sensitivity but does not translate into precise defect detection, thus reaffirming that newer architectures like our ID-MSNet are better adapted to contemporary challenges. Our proposed model combines architectural strengths from various SOTA models while introducing novel elements that enhance feature integration and representation. This is evidenced by its superior performance metrics, making it the most comprehensive and effective model for steel defect detection.

Table 2. Numerical test results of the SD-saliency-900 dataset.

Model	Precision	Recall	F1-Score	mIoU
UNet [22]	0.6936	0.9539	0.8032	0.7689
LinkNet [34]	0.8795	0.8900	0.8847	0.8399
FPN [35]	0.8489	0.8736	0.8610	0.8240
ResUnet-a [36]	0.8329	0.8811	0.8563	0.8152
Attention U-net [37]	0.7958	0.8614	0.8273	0.7821
PSPNet [24]	0.8149	0.8548	0.8344	0.8026
Proposed ID-MSNet	0.8626	0.8963	0.8791	0.8619

5. Discussion

This section is organized into three subsections for comprehensive analysis. In Section 5.1, we discuss the effectiveness of the primary components in the proposed network, examining how each contributes to the overall performance. Section 5.2 focuses on a review of previous studies related to the dataset. Finally, Section 5.3 delves into an analysis of the instances where the proposed model falls short or fails to accurately detect defects.

5.1. Ablation Studies

The aim of this ablation study is to rigorously evaluate the enhancements we made to the base UNet3+ model for the task of steel defect detection. These changes were systematically tested and their contributions quantified in terms of precision, recall, F1-score, and mean intersection-over-union (mIoU) using the publicly available SD-saliency-900 dataset. The results of these ablation experiments are summarized in Table 3, which reveals that each modification incrementally improves the model's performance metrics.

Table 3. Ablation experimental results of the SD-saliency-900 dataset.

Methods	Precision	Recall	F1-Score	mIoU
UNet3+	0.8543	0.8903	0.8720	0.8486
UNet3 + CBAM	0.8870	0.8480	0.8670	0.8503
UNet3 + CBAM + IDS	0.8717	0.8810	0.8763	0.8587
UNet3 + CBAM + IDS + MSFLM	0.8637	0.8921	0.8785	0.8561
UNet3 + CBAM + IDS + MSFLM + DropBlock	0.8626	0.8963	0.8791	0.8619

In our study, each modification to the UNet3+ model progressively improved its performance, effectively addressing the complexities of steel defect detection. For example, adding CBAM led to a notable boost in precision but incurred a slight dip in recall. Nevertheless, the mIoU increased, underscoring CBAM's ability to focus the model on essential features. Subsequently, the integration of IDS balanced out both precision and recall while also elevating the mIoU, indicating that this approach is effective in retaining more detailed information during the downsampling process. Further incorporation of MSFLM resulted in a considerable improvement in recall without sacrificing much in terms of precision or mIoU. This makes MSFLM a valuable component, especially for detecting a range of steel defects that vary in size and complexity. Finally, the addition of DropBlock slightly reduced precision but yielded higher recall and F1-score values. This trade-off suggests that DropBlock enhances the model's generalization capabilities, resulting in a more balanced performance across metrics. To sum up, our fully modified UNet3+ model, which incorporates all of these improvements, achieved a state-of-the-art mIoU score of 86.19%. This affirms the efficacy of our comprehensive approach to addressing the multi-faceted challenges posed by steel defect detection.

5.2. Comparison of the Proposed Model with the Previous Studies

In this study, we conducted a comprehensive comparison between our proposed ID-MSNet model and the performance scores achieved in prior research studies. These assessments were based on the SD-saliency-900 dataset, and the comparative results are shown in Table 4.

Table 4. Results of previous studies using SD-saliency-900 dataset.

Reference	Methods	mIoU	Dice	Accuracy	AUC	F1-Score
[38]	Chained atrous spatial pyramid pooling network	0.7820	-	-	-	-
[39]	Depth-wise separable convolution	-	0.8080	0.9542	-	-
[40]	Residual attention network, bidirectional convolutional long short-term memory	-	0.820	0.962	-	-
[41]	Nonconvex total variation, regularized RPCA with kernelization	-	-	0.8864	0.9255	0.5787
ID-MSNet	Multiscale feature learning, attention mechanisms	0.8619	0.8793	0.9737	0.9936	0.8791

Our findings indicate that ID-MSNet achieved a remarkable mIoU score of 0.8619%, marking it as the highest-performing model in this study. Second position achieved a competitive mIoU score of 0.820, closely following our model. A different method utilized

non-convex total variation regularized RPCA with kernelization, resulting in an accuracy of 0.8864 and a substantial AUC of 0.9255.

It is worth noting that in previous works on the SD-saliency-900 dataset, no model has achieved an mIoU score as high as ID-MSNet. Our results highlight the superior performance of our proposed architecture for steel defect detection.

In summary, our ID-MSNet model emerges as the superior model compared to all preceding studies on the SD-saliency-900 dataset, boasting the highest mIoU score among all evaluated methodologies.

5.3. Analysis of Failure Cases

In our proposed deep learning defect detection model, ID-MSNet, we achieved promising results in identifying steel defects on the SD-saliency-900 dataset. However, it is essential to conduct a thorough analysis of the cases where our model faced challenges and potential areas for improvement. In Figure 9, we present three illustrative cases where our model exhibited limitations in defect detection.

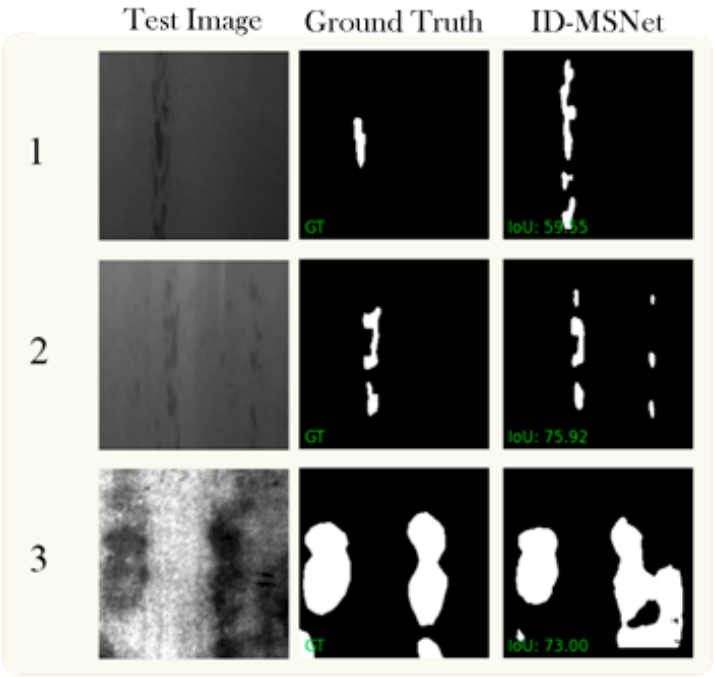


Figure 9. Sample challenging cases on the SD-saliency-900 dataset.

The first and second rows of images in Figure 9 depict instances where our model detected more defects than indicated in the ground truth. These situations occurred in regions that closely resembled defects in the images. Importantly, it is worth noting that the particular types of defects represented in the first two images were relatively rare in our dataset. The scarcity of such examples during training could contribute to the model's heightened sensitivity to these rare patterns. To address this challenge, future iterations of our model could benefit from additional rare defect samples during training, allowing the model to learn to distinguish these unusual patterns from benign variations more effectively. In the third row of Figure 9, we provide a sample of patches where our model achieved an intersection over union (IoU) of 73. However, it is evident that the model detected more regions as defects than what was annotated in the ground truth.

To address this challenge, future research could focus on enhancing the model's ability to differentiate between true defects and similar image structures, as well as implementing post-processing techniques to improve patch-level predictions. Moreover, expanding the

diversity of the training dataset with more examples of rare defects may lead to improved generalization and reduced false detections. The over-detection pattern in rows 1–2 may also reflect a texture bias, where the model responds to surface texture cues rather than true defect semantics. Investigating this using interpretability tools such as Grad-CAM to compare model activation maps against ground truth defect regions is a valuable direction for future work. As the field of deep learning for defect detection continues to evolve, addressing these challenges will contribute to more robust and accurate defect detection systems in industrial applications.

6. Conclusions

The quest for an efficient, accurate, and resource-effective model for steel surface defect detection is an ongoing endeavor, critical for enhancing the quality and reliability of manufacturing processes. This study introduces ID-MSNet, an enhanced version of the UNet3+ architecture, which integrates multiscale feature learning and attention mechanisms to tackle the challenges inherent in automated steel surface defect detection. Our primary contribution lies in the architectural modifications made to the encoder part of the UNet3+ model. Specifically, the introduction of the multiscale feature learning module enables the model to capture defect features at multiple scales, thus providing a more comprehensive understanding of complex defect patterns. The utilization of Conv2D layers with different dilation rates, in conjunction with DropBlock regularization and batch normalization, empowers the model with adaptive feature learning capabilities. Furthermore, the replacement of traditional max-pooling with a novel downsampling strategy using strided Conv2D layers aids in more effective feature extraction and representation. One of the distinguishing aspects of ID-MSNet is the incorporation of the convolutional block attention module (CBAM) into the skip connections. This innovative addition enhances the model's attention mechanism, enabling it to focus more effectively on relevant features for defect detection. These combined modifications result in a model that not only excels in terms of performance metrics but also achieves this with fewer network parameters. Experimental validation on the SD-saliency-900 dataset verified the efficacy of our approach. ID-MSNet outperformed existing state-of-the-art models by achieving a mean intersection-over-union (mIoU) score of 86.19%, thereby solidifying its capability to effectively identify steel surface defects with high precision. Several limitations of the current study merit attention. The SD-saliency-900 dataset covers only three defect categories and 900 images, which limits the scope of generalization claims. A single fixed dataset split was used to allow fair comparison with prior work; future studies should include cross-validation or repeated runs to provide stronger statistical evidence of robustness. Additionally, inference latency and memory footprint under deployment conditions were not benchmarked in this work and should be evaluated before drawing conclusions about production-line suitability.

As future work, we aim to evaluate ID-MSNet on larger and more diverse steel defect datasets, including datasets covering additional defect types such as dents and corrosion marks. A quantitative per-challenge breakdown of how each module contributes to performance on varied defect types, background similarity, and size variation would also strengthen the analysis and is planned for future extended work. We also intend to apply the model to other metallic and non-metallic surface inspection tasks to assess its broader generalizability. On the deployment side, a practical pipeline would involve computing defect area ratios or connected component statistics from the pixel-level segmentation output, which can then be mapped to tiered quality decisions (accept, flag, reject) according to configurable severity thresholds. This pathway requires validation on live production data before operational claims can be made. In sum, the ID-MSNet model presented in

this study establishes a strong and efficient baseline for automated steel surface defect segmentation, with results that support its further development for industrial quality inspection applications.

Author Contributions: Conceptualization, M.S., A.S. and J.R.; methodology, M.S.; software, M.S.; validation, M.S.; formal analysis, M.S.; investigation, M.S.; data curation, M.S.; writing—original draft preparation, M.S.; writing—review and editing, J.R. and A.S.; supervision, J.R. and A.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by a research grant from the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant RGPIN-2021-03893.

Data Availability Statement: The data used in this study are publicly available in the SD-saliency-900 dataset.

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

References

1. Gu, C.; Lu, Y.; Chen, M.; Sun, G.; Ni, Z. A reweighting offset bin classification network for surface defect detection and location of metal components. *Measurement* **2022**, *187*, 110166. [CrossRef]
2. Luo, Q.; Fang, X.; Liu, L.; Yang, C.; Sun, Y. Automated Visual Defect Detection for Flat Steel Surface: A Survey. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 626–644. [CrossRef]
3. Ren, J.; Gabbar, H.A.; Huang, X.; Saberironaghi, A. Defect Detection for Printed Circuit Board Assembly Using Deep Learning. In Proceedings of the 2022 8th International Conference on Control Science and Systems Engineering (ICCSSE), Guangzhou, China, 14–16 July 2022; pp. 85–89. [CrossRef]
4. Saberironaghi, A.; Ren, J.; El-Gindy, M. Defect Detection Methods for Industrial Products Using Deep Learning Techniques: A Review. *Algorithms* **2023**, *16*, 95. [CrossRef]
5. Sime, D.M.; Wang, G.; Zeng, Z.; Peng, B. Deep learning-based automated steel surface defect segmentation: A comparative experimental study. *Multimed. Tools Appl.* **2023**, *83*, 2995–3018. [CrossRef]
6. Guo, B.; Wang, Y.; Zhen, S.; Yu, R.; Su, Z. SPEED:Semantic Prior and Extremely Efficient Dilated Convolution Network for Real-Time Metal Surface Defects Detection. *IEEE Trans. Ind. Inform.* **2023**, *19*, 11380–11390. [CrossRef]
7. Sime, D.M.; Wang, G.; Zeng, Z.; Wang, W.; Peng, B. Semisupervised Defect Segmentation With Pairwise Similarity Map Consistency and Ensemble-Based Cross Pseudolabels. *IEEE Trans. Ind. Inform.* **2023**, *19*, 9535–9545. [CrossRef]
8. Djukic, D.; Spuzic, S. Statistical discriminator of surface defects on hot rolled steel. *Image Vis. Comput.* **2007**, *1*, 158–163.
9. Yun, J.P.; Choi, S.; Kim, S.W. Vision-based defect detection of scale-covered steel billet surfaces. *Opt. Eng.* **2009**, *48*, 037205. [CrossRef]
10. Choi, D.-C.; Jeon, Y.-J.; Yun, J.P.; Kim, S.W. Pinhole detection in steel slab images using Gabor filter and morphological features. *Appl. Opt.* **2011**, *50*, 5122–5129. [CrossRef] [PubMed]
11. Yun, J.P.; Choi, S.; Kim, J.-W.; Kim, S.W. Automatic detection of cracks in raw steel block using Gabor filter optimized by univariate dynamic encoding algorithm for searches (uDEAS). *NDT E Int.* **2009**, *42*, 389–397. [CrossRef]
12. Liu, W.; Yan, Y. Automated surface defect detection for cold-rolled steel strip based on wavelet anisotropic diffusion method. *Int. J. Ind. Syst. Eng.* **2014**, *17*, 224–239. [CrossRef]
13. Gayubo, F.; Gonzalez, J.; de la Fuente, E.; Miguel, F.; Peran, J. On-line machine vision system for detect split defects in sheet-metal forming processes. In Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06), Hong Kong, China, 20–24 August 2006; pp. 723–726. [CrossRef]
14. Guan, S.; Lei, M.; Lu, H. A Steel Surface Defect Recognition Algorithm Based on Improved Deep Learning Network Model Using Feature Visualization and Quality Evaluation. *IEEE Access* **2020**, *8*, 49885–49895. [CrossRef]
15. He, Y.; Song, K.; Dong, H.; Yan, Y. Semi-supervised defect classification of steel surface based on multi-training and generative adversarial network. *Opt. Lasers Eng.* **2019**, *122*, 294–302. [CrossRef]
16. Zhao, W.; Chen, F.; Huang, H.; Li, D.; Cheng, W. A New Steel Defect Detection Algorithm Based on Deep Learning. *Comput. Intell. Neurosci.* **2021**, *2021*, 5592878. [CrossRef]
17. Song, G.; Song, K.; Yan, Y. EDRNet: Encoder–Decoder Residual Network for Salient Object Detection of Strip Steel Surface Defects. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 9709–9719. [CrossRef]
18. Huang, H.; Lin, L.; Tong, R.; Hu, H.; Zhang, Q.; Iwamoto, Y.; Han, X.; Chen, Y.-W.; Wu, J. UNet 3+: A Full-Scale Connected UNet for Medical Image Segmentation. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 1055–1059. [CrossRef]

19. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440. Available online: https://openaccess.thecvf.com/content_cvpr_2015/html/Long_Fully_Convolutional_Networks_2015_CVPR_paper.html (accessed on 15 August 2023). [CrossRef]
20. Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 801–818. Available online: https://openaccess.thecvf.com/content_ECCV_2018/html/Liang-Chieh_Chen_Encoder-Decoder_with_Atrous_ECCV_2018_paper.html (accessed on 12 September 2023).
21. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A.; Liu, W.; et al. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9. [CrossRef]
22. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*; Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2015; pp. 234–241. [CrossRef]
23. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef]
24. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid Scene Parsing Network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890. Available online: https://openaccess.thecvf.com/content_cvpr_2017/html/Zhao_Pyramid_Scene_Parsing_CVPR_2017_paper.html (accessed on 12 September 2023).
25. Gao, X.; Sun, X.; Zhang, Y.; Yan, M.; Xu, G.; Sun, H.; Jiao, J.; Fu, K. An End-to-End Neural Network for Road Extraction From Remote Sensing Imagery by Multiple Feature Pyramid Network. *IEEE Access* **2018**, *6*, 39401–39414. [CrossRef]
26. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141. Available online: https://openaccess.thecvf.com/content_cvpr_2018/html/Hu_Squeeze-and-Excitation_Networks_CVPR_2018_paper.html (accessed on 15 August 2023).
27. Zhang, Y.; Li, K.; Li, K.; Wang, L.; Zhong, B.; Fu, Y. Image Super-Resolution Using Very Deep Residual Channel Attention Networks. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 286–301. Available online: https://openaccess.thecvf.com/content_ECCV_2018/html/Yulun_Zhang_Image_Super-Resolution_Using_ECCV_2018_paper.html (accessed on 12 September 2023).
28. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19. Available online: https://openaccess.thecvf.com/content_ECCV_2018/html/Sanghyun_Woo_Convolutional_Block_Attention_ECCV_2018_paper.html (accessed on 12 September 2023).
29. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual Attention Network for Scene Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–21 June 2019; pp. 3141–3149. Available online: https://openaccess.thecvf.com/content_CVPR_2019/html/Fu_Dual_Attention_Network_for_Scene_Segmentation_CVPR_2019_paper.html (accessed on 12 September 2023). [CrossRef]
30. Liu, F.; Lin, G.; Shen, C. CRF learning with CNN features for image segmentation. *Pattern Recognit.* **2015**, *48*, 2983–2992. [CrossRef]
31. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556. [CrossRef]
32. Tan, M.; Le, Q. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In Proceedings of the 36th International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114. Available online: https://proceedings.mlr.press/v97/tan19a.html (accessed on 15 August 2023).
33. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. Available online: https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html (accessed on 15 August 2023).
34. Chaurasia, A.; Culurciello, E. LinkNet: Exploiting encoder representations for efficient semantic segmentation. In Proceedings of the 2017 IEEE Visual Communications and Image Processing (VCIP), St. Petersburg, FL, USA, 10–13 December 2017; pp. 1–4. [CrossRef]
35. Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125. Available online: https://openaccess.thecvf.com/content_cvpr_2017/html/Lin_Feature_Pyramid_Networks_CVPR_2017_paper.html (accessed on 12 September 2023). [CrossRef]
36. Diakogiannis, F.I.; Waldner, F.; Caccetta, P.; Wu, C. ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data. *ISPRS J. Photogramm. Remote Sens.* **2020**, *162*, 94–114. [CrossRef]

37. Oktay, O.; Schlemper, J.; Folgoc, L.L.; Lee, M.; Heinrich, M.; Misawa, K.; Mori, K.; McDonagh, S.; Hammerla, N.Y.; Kainz, B.; et al. Attention U-Net: Learning Where to Look for the Pancreas. *arXiv* **2018**, arXiv:1804.03999. [CrossRef]
38. Zheng, Z.; Hu, Y.; Zhang, Y.; Yang, H.; Qiao, Y.; Qu, Z.; Huang, Y. CASPPNet: A chained atrous spatial pyramid pooling network for steel defect detection. *Meas. Sci. Technol.* **2022**, *33*, 085403. [CrossRef]
39. Huang, Z.; Wu, J.; Xie, F. Automatic surface defect segmentation for hot-rolled steel strip using depth-wise separable U-shape network. *Mater. Lett.* **2021**, *301*, 130271. [CrossRef]
40. Yang, L.; Xu, S.; Fan, J.; Li, E.; Liu, Y. A pixel-level deep segmentation network for automatic defect detection. *Expert Syst. Appl.* **2023**, *215*, 119388. [CrossRef]
41. Wang, J.; Xu, G.; Li, C.; Wang, Z.; Yan, F. Surface Defects Detection Using Non-convex Total Variation Regularized RPCA with Kernelization. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 5007013. [CrossRef]

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.