



# Article Enhancing Precision with an Ensemble Generative Adversarial Network for Steel Surface Defect Detectors (EnsGAN-SDD)

Fityanul Akhyar <sup>1,2</sup>, Elvin Nur Furqon <sup>2,3</sup> and Chih-Yang Lin <sup>1,\*</sup>

- <sup>1</sup> Department of Electrical Engineering, Yuan Ze University, Taoyuan 320, Taiwan; fityanul@telkomuniversity.ac.id
- <sup>2</sup> School of Electrical Engineering, Telkom University, Bandung 40257, Indonesia; elvin@novaglobal.com.sg
   <sup>3</sup> Artificial Intelligence—High Performing Computer Division Nova Clobal Pte Ltd
- <sup>3</sup> Artificial Intelligence—High Performing Computer Division, Nova Global Pte Ltd.,
- Singapore 609966, Singapore Correspondence: andrewlin@saturn.yzu.edu.tw

Abstract: Defects are the primary problem affecting steel product quality in the steel industry. The specific challenges in developing detect defectors involve the vagueness and tiny size of defects. To solve these problems, we propose incorporating super-resolution technique, sequential feature pyramid network, and boundary localization. Initially, the ensemble of enhanced super-resolution generative adversarial networks (ESRGAN) was proposed for the preprocessing stage to generate a more detailed contour of the original steel image. Next, in the detector section, the latest state-of-the-art feature pyramid network, known as De-tectoRS, utilized the recursive feature pyramid network technique to extract deeper multi-scale steel features by learning the feedback from the sequential feature pyramid network. Finally, Side-Aware Boundary Localization was used to precisely generate the output prediction of the defect detectors. We named our approach EnsGAN-SDD. Extensive experimental studies showed that the proposed methods improved the defect detector's performance, which also surpassed the accuracy of state-of-the-art methods. Moreover, the proposed EnsGAN achieved better performance and effectiveness in processing time compared with the original ESRGAN. We believe our innovation could significantly contribute to improved production quality in the steel industry.

Keywords: defect detection; generative adversarial network (GAN); recursive FPN; boundary localization

# 1. Introduction

Vision-based steel surface inspection system (detectors) are the key to maintaining the production quality of steel products. Even when a high-resolution camera is utilized to acquire images of the steel surface, some difficult features of defects affect the performance of detectors. More specifically, defects that are vague in appearance and tiny may be challenging to detect since they are shown against a wide background area [1]. Therefore, enhancing the quality of input images plays an important role in improving the localization performance of defect detectors. These two problems mean a preprocessing stage, such as recalibrating the pixel quality, is required before the steel image enters the detection stage. The output of this process might increase the difference between the defect and the background area.

# 1.1. The Benefecial of Super Resolution Generative Adversarial Networks

Currently, there are two main options to improve the sharpness of the image via pixel manipulation. Data augmentation is the first most practical way; for example, gamma correction and contrast-limited adaptive histogram equalization (CLAHE) [2]. Although easy to implement, this process increases the noise in the image and reduces the difference between the region of interest and the background. On the other hand, super-resolution is becoming another popular pixel restoration method. In essence, image super-resolution is



Citation: Akhyar, F.; Furqon, E.N.; Lin, C.-Y. Enhancing Precision with an Ensemble Generative Adversarial Network for Steel Surface Defect Detectors (EnsGAN-SDD). *Sensors* 2022, 22, 4257. https://doi.org/ 10.3390/s22114257

Academic Editor: Xinyu Li

Received: 25 April 2022 Accepted: 31 May 2022 Published: 2 June 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 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 (https:// creativecommons.org/licenses/by/ 4.0/). the process of increasing the resolution of an image from low resolution to high resolution. This method is quite useful in many applications, i.e., surveillance systems [3], medical imaging [4], and industry [5]. Performing facial recognition in security systems requires high-resolution images [6]. Moreover, generating MRI images requires a high resolution to ensure the accuracy of disease diagnosis [7]. The same goes for inspection systems in the industry, where a high resolution can assist with the precision of defect localization in the products [8]. To the best of our knowledge, enhanced super resolution generative adversarial networks (ESRGAN) is a state-of-the-art model for super resolution [9], which is a recent innovation in machine learning. GAN basically employs a deep network in conjunction with an adversarial network to produce higher resolution images. The generated realistic image produced by GAN is very pleasant to human eyes since it adopts the similarity index to measure the distance from the original image. GANs are made up of two networks: the generator and the discriminator. The generator component of a GAN learns to generate synthesis data by incorporating feedback from the discriminator. It learns to make the discriminator classify its output as real. GAN is more applicable to this case since it applies to unseen images in the latest models [10]. Furthermore, multi-generator GAN has significantly improved upon single-generator GAN [11,12]. The basic idea in the multi-generator is to share the network parameters and find the best generator through a probabilistic mechanism. This technique is also applied to real-world applications such as power systems [13], medication systems [14], and transportation systems [11].

## 1.2. The Benefecial of Feature Pyramid Networks

Feature pyramid networks (FPN) [15] have had greater success in object detection. The FPN structures allow the features to be extracted in different scales, which is the key to solving the challenges for small objects. Furthermore, several works have paid more attention to improving FPN performance. For instance, the proposed work, NAS-FPN [16], used neural architecture search to develop a new feature pyramid architecture in a novel scalable search space that covered all cross-scale connections. The newlyinvented architecture uses a combination of top-down and bottom-up connections to connect features at different scales. Moreover, the work in ref. [17] proposed a plugin for feature upsampling in the FPN structure called content-aware reassembly of features (CARAFE) to enlarge the receptive field. Instead of using interpolation and deconvolution, CARAFE employs an adaptive and optimized reassembly kernel in various locations to achieve better performance. Currently, recursive feature pyramid (RFP) [18] extends the functionality of FPN by incorporating extra feedback connections from the FPN layers into the bottom-up backbone layers. The RFP can serve as a backbone for an object detector that looks at the images twice or more by unrolling the recursive structure to a sequential implementation to enhance the generation of a meaningful representation of samples. In line with the successful works on FPN, anchoring methods also play an essential role in object detection. Region proposal network (RPN) [19] is a state-of-the-art anchoring method developed to predict the region proposal with a broader scale and aspect ratio. The anchor boxes proposed are served at multiple scales and aspect ratios in this scheme. The RPN predicts whether an anchor will be in the background or foreground, and refines the anchor accordingly. After discerning the exact box (location) of the foreground or object of interest, the current model [20] proposes the precise localization of the object. This methodology, named Side-Aware Boundary Localization (SABL), estimates an object's boundary with the bucketing scheme and fine regression. In addition to applying super-resolution GAN, most real-world applications, including manufacturing, benefit from FPN [21,22]. Moreover, the combination of FPN and RPN shows better performance on production in various industries [23–26].

#### 1.3. The Major Contributions and Novelty of the Proposed EnsGAN-SDD

Motivated by the success of enhanced super-resolution generative adversarial networks (ESRGAN) and feature pyramid networks (FPN) in many real-world applications, this work proposed GAN for the preprocessing stage to improve the quality of steel images before inspection by defect detectors. Moreover, this work aimed to integrate the state-of-the-art ESRGAN in the preprocessing stage and the recursive feature pyramid networks in the detectors stage to develop a new steel surface inspection system. Although GAN is commonly used in industrial applications [27–29], to the best of our knowledge, the concept of multi-generator GAN has never previously been applied. Furthermore, in this work, we applied the concept of multi-generator GAN to steel manufacturing by proposing a novel EnsGAN to enhance the output quality and processing speed of conventional ESRGAN. The current multi-generator GAN [12,30-32] uses the whole image as the input network. In contrast, the proposed multi-generator in EnsGAN improves upon previous works by splitting operations on the input image, which significantly enhances the processing speed. Meanwhile, we improved defect localization precision by adding the Side-Aware Boundary Localization into the detector's architecture. Our proposed work, named EnsGAN-SDD, significantly improves the accuracy of the steel surface inspection system compared with state-of-the-art methods like the R-CNN Family [19,33], YOLO variants [34], and transformer networks [35].

# 2. Related Works

Super-resolution generative adversarial network (SRGAN) [36] uses a deep neural network combined with an adversary network to generate higher resolution images. It employs a perceptual loss function that includes an adversarial loss and a content loss. Specifically, a high-resolution image (HR) is downsampled to a low-resolution image (LR) during the training. Then, using a GAN generator, LR images are upsampled to super-resolution images (SR). Finally, it uses a discriminator to distinguish the HR images. Moreover, the backpropagation uses the GAN loss to train the discriminator and generator by employing a perceptual loss function comprising an adversarial loss and a content loss. The adversarial loss directs the solution to the natural image manifold by employing a trained discriminator network to distinguish between super-resolved images and original photo-realistic images. Rather than relying on similarity in pixel space, the proposed work employed a content loss motivated by perceptual similarity. To improve visual quality, this study [9] enhanced three critical components of SRGAN architecture, adversarial loss, and perceptual loss by proposing Enhanced SRGAN (ESRGAN). Specifically, Residual-in-Residual Dense Block (RRDB) without batch normalization was introduced as the basic network building unit. Furthermore, relativistic GAN [37] enables the discriminator to predict relative reality rather than absolute value. Finally, using the features prior to activation, ESRGAN improves perceptual loss and provides stronger supervision for brightness consistency and texture recovery. Furthermore, Real-ESRGAN [10] applies the powerful ESRGAN to a practical restoration application trained on purely synthetic data. Specifically, a high-order degradation modeling process was introduced to simulate complex real-world degradations better. Ringing and overshoot artifacts were also taken into account during the synthesis process. In addition, a U-Net discriminator with spectral normalization improves discriminator capability and stabilizing training. Some of the the-state-of-the-art methods [11,12] propose using not only a single generator, but also multiple generators in GAN. Specifically, various generators were combined as a mixture of the probabilistic model, and one was selected as the best generator to yield the final output.

A feature pyramid network (FPN) [15] is a feature extractor that takes a single-scale image of any size as the input and outputs proportionally sized feature maps at multiple levels in a fully convolutional manner. This process is separate from the backbone convolutional architectures. Therefore, the FPN can be used as a generic solution for building feature pyramids inside deep convolutional networks for tasks like object detection [19,33,34,38–40]. The construction of the pyramid applies a bottom-up pathway and a top-down pathway. Specifically, the bottom-up pathway is the backbone ConvNet's feedforward computation, which computes a feature hierarchy consisting of feature maps at various scales with a scaling step. On the other hand, higher resolution features are

produced by the top-down pathway, which upsamples spatially coarser but semantically stronger feature maps from higher pyramid levels. These features are then enhanced with features from the bottom-up pathway through lateral connections. Each lateral connection merges feature maps of the same spatial size from the bottom-up and top-down pathways. For instance, the R-CNN family, such as Faster R-CNN [19] and Cascade R-CNN [33], as well as free anchor methods like Foveabox [39], always use the FPN inside their structures. The YOLO Family, including YOLOv4 [34], YOLOv5 [41], and YOLOX [38], as well as the most recent pyramid vision transformer or PVTv2 [40], utilized enhancement of feature pyramid networks as the primary structural component. Moreover, the latest feature pyramid model, DetectoRS [18], proposed recursive feature pyramid network (RFP) to incorporate extra feedback connections from Feature Pyramid Networks into the bottom-up backbone layers, which significantly improves object detection performance.

To hypothesize object locations, state-of-the-art object detection networks [18,19,33,39] rely on region proposal algorithms. A region proposal network (RPN) [19] shares full-image convolutional features with the detection network, allowing for near-free region proposals. It is a fully convolutional network that predicts object bounds and objectness scores at each position at the same time. The RPN was trained inside the end-to-end network to generate high-quality region proposals. Furthermore, Side-Aware Boundary Localization (SABL) [20] was proposed for precise localization proposals in object detection. Each side of the bounding box is localized with a dedicated network branch. Empirically, this method observes that when they manually annotate a bounding box for an object, it is frequently easier to align each side of the box to the object boundary than to move the box while tuning the size. In response to this observation, SABL positions each side of the bounding box based on its surrounding context.

Currently, deep learning-based methods have been widely adopted in many application tasks, including the development of vision-based inspection systems. For instance, the work in ref. [42] applied deep learning convolutional neural networks to determine defect areas on highway roads. Furthermore, the proposed system in ref. [43] used deep learning object detection to develop a defect inspection system for highway roads. Moreover, object detection for defect detection systems (detector) has been widely applied in the manufacturing process, including the steel industry [21–26]. Feature pyramid networks (FPN) and region proposal network (RPN) are the main detector components in these previous works. Specifically, the proposed approach in refs. [21,22] utilized FPN inside of a YOLO structure to extract multi-scale features from images of steel and aircraft products. The proposed framework in [23–26] also adopted FPN to acquire multi-scale features from images of steel, wood, solar cells, and electrical products combined with RPN to localize the defect regions accurately.

The proposed EnsGAN-SDD is mainly inspired by the literature reviewed above to construct the steel surface inspection system. First, ESRGAN was selected as the input image enhancement technique rather than Real ESRGAN since Real ESRGAN produces more noise when facing lower resolutions. Moreover, we proposed a novel EnsGAN that adopted the concept of the multi-Generator GAN to improve the performance of the baseline method in terms of accuracy and processing speed. Then, in the detection stage or steel defect detector (SDD), Side-Aware Boundary Localization (SABL) was integrated into the latest state-of-the-art feature pyramid network, or DetectoRS. The experiments on the common steel surface dataset and Severstal steel dataset showed the proposed work on top of the residual network, and the aggregate residual network backbone achieved significant improvement over the state-of-the-art object detection methods [19,33,34,38–40]. Moreover, the proposed EnsGAN SDD also met the criteria of inference time [44,45] in the steel surface inspection system, making it suitable for real-world scenarios in the steel industry.

#### 3. Proposed Methods

We proposed two processes for detecting steel defect areas: preprocessing and feature extraction, as shown in Figure 1. The preprocessing stage in our method, named the EnsGAN model, is integrated with the state of the art recursive feature pyramid network (DetectoRS) [18] on top of a residual network (ResNet) [46] or residual aggregation network (ResNext) [47], with Boundary Localization [20] as the localization and recognition network in the feature extraction stage. Moreover, in this work, we aimed to improve super resolution quality from the Enhanced Super-Resolution Generative Adversarial Network (ESRGAN) and reduce the latency of process inference for the steel image enhancement method. This section introduces the base architectures of the proposed EnsGAN (generator and discriminator) used in this research and our strategy to improve quality and reduce latency for the steel defect problem. The resulting image from the EnsGAN is used as the input for DetectorRS with the SABL prediction head [20] as the steel defect detector (SDD) to produce a highly accurate defect localization system for steel surface images.



**Figure 1.** Flow Process detecting steel defects image. In the first stage, we proposed the preprocessing model named EnsGAN to enhance the lower resolution of input image. The next stage is feature extraction that utilizes the combination of DetectoRS, which has Recursive FPN on top of the Residual Network (ResNet) or residual aggregation network (ResNext) backbone, and side aware boundary localization in the prediction head.

In the first step of the preprocessing procedure, the discriminator should be trained to classify an image as "High Resolution" or "Low Resolution". The generator will train to generate the super-resolution image and test it in the discriminator. The loss and information from the discriminator will be used in the generator to enhance the weight to create better resolution. For further detail, the Algorithm 1 shows the complete process flow from preprocessing through detection:

Algorithm 1 Algorithm of the proposed method						
1.	Procedure EnsGAN:					
2.	Train discriminator (VGG28) with HR and LR to identify HR/LR					
3.	Split each LR image into n pieces ( $n = 2, 4, 8,$ )					
4.	Initiate n generator (RRDB)					
5.	For $i = 1$ to iteration do					
6.	Generate HR with generator with each piece of LR image					
7.	Combine all result generators into 1 image according to positions					
8.	Check whether X <sub>generator</sub> is HR or LR using the discriminator					
9.	If $Y_{discriminator} = 1 do$					
10.	Pass					
11.	Else do					
12.	Update weight and bias in generator					
13.	Repeat training generator					
14.	Procedure Feature Extraction:					
15.	Convolution input image with RestNet-50					
16.	Localize enhancement using recursive feature pyramid (RFP)					
17.	Boundary localization using SABL					

# Algorithm 1 Cont.

- 18. Prepare the data images, convert HR to LR (use bicubic algorithm)
- 19. Prepare dataset in Pascal VOC format
- 20. Run Procedure EnsGAN
- 21. Run Procedure Feature Extraction

#### 3.1. Proposed EnsGAN

Our base model uses ESRGAN [9], which achieves better quality than previous superresolution models such as SRCNN [48], EDSR [49], SRGAN [36], and RCAN [50]. The ESRGAN model is an improvement on the model from SRGAN that removes batch normalization (BN) in the generator structure and uses the residual in residual dense block (RRDB), as depicted in Figure 2.



**Figure 2.** (a) ESRGAN improves the SRGAN with removing BN in the residual block. (b) The Residual in residual dense block (RRDB) and residual scaling parameter ( $\beta$ ) used at the SRGAN structure.

Removing batch normalization [49] can reduce memory usage by up to four times, achieve better performance than the standard ResNet structure model, and serve as the best simple strategy for sharpening and deblurring images [51]. In the generator structure, ESRGAN keeps the high-level SRGAN architecture. The main difference is that each block contains some convolution layer and activation functions using Leaky Rectified Linear Unit (LReLU) to prevent dead nodes [9]. Improvement can be seen by scaling down the residuals by multiplying a constant between 0 and 1 before adding them to the main path to prevent instability and using initialization when training has a slight variance.

For the discriminator, ESRGAN uses a relativistic discriminator to approximate the probability of an image being real or fake. A generator uses a linear combination of perceptual difference between real and fake images using the VGG-28 network architecture, pixel-wise absolute difference between real and fake images, and the relativistic average loss between real and fake images function during adversarial training. In standard discriminator GAN, the discriminator can be defined as:

$$f(\mathbf{x}) = \operatorname{Sigmoid}(\mathbf{R}(\mathbf{x})) \approx \sigma(\mathbf{R}(\mathbf{x}))$$
 (1)

where R(x) is a non-transformed layer, and D(x) is a discriminator [36]. A relative discriminator can be represented as:

$$D(x_{real}, x_{fake}) = \sigma(R(x_{real}) - E[R(x_{fake})])$$
(2)

$$D(x_{fake}, x_{real}) = \sigma(R(x_{fake}) - E_{real}[R(x_{real})])$$
(3)

where  $\mathbb{E}$  represents the operation of taking the average of the data in the mini-batch. The discriminator loss is then defined as:

$$L_D^{Ra} = -\mathbb{E}_{x_{real}}[\log(D_{Ra}(x_{real}))] - \mathbb{E}_{x_{fake}}[\log(1 - D_{Ra}(x_{fake}))]$$
(4)

And the adversarial loss for a generator in the symmetrical form can be written as:

$$L_{G}^{Ra} = -\mathbb{E}_{x_{real}}[log(1 - D_{Ra}(x_{real}))] - \mathbb{E}_{x_{fake}}[log(D_{Ra}(x_{fake}))]$$
(5)

where:

$$D_{Ra}(x) \begin{cases} \text{sigmoid} (R(x) - \mathbb{E}_{x_{fake}} R(x_{fake})) & \text{if } x \text{ is real} \\ \text{sigmoid} (R(x) - \mathbb{E}_{x_{real}} R(x_{real})) & \text{if } x \text{ is fake} \end{cases}$$
(6)

We present our model to achieve better results and time consumption for steel defect images. Our idea is to use an ensemble multi-Generator rather than a single one in the general ESRGAN model. The input data used for training are a part of the split image, as shown in Figure 3. Specifically, we used a split array image to implement the splitting operation. Next, the proposed EnsGAN inputs the generator using a split low-resolution image, as shown in Figure 4.

$$X_{LR} = \begin{bmatrix} 1 & 4 & 5 & 8 & 4 & 6 \\ 6 & 2 & 3 & 5 & 9 & 1 \\ 2 & 7 & 9 & 8 & 6 & 8 \\ 4 & 1 & 2 & 3 & 7 & 6 \\ 5 & 1 & 0 & 5 & 1 & 5 \\ 2 & 1 & 2 & 3 & 3 & 7 \end{bmatrix} \rightarrow \left( \begin{bmatrix} 1 & 4 & 5 \\ 6 & 2 & 3 \\ 2 & 7 & 9 \end{bmatrix}_{LR1}, \begin{bmatrix} 8 & 4 & 6 \\ 5 & 9 & 1 \\ 8 & 6 & 8 \end{bmatrix}_{LR2}, \begin{bmatrix} 4 & 1 & 2 \\ 5 & 1 & 0 \\ 2 & 1 & 2 \end{bmatrix}_{LR3}, \begin{bmatrix} 3 & 7 & 6 \\ 5 & 1 & 5 \\ 3 & 3 & 7 \end{bmatrix}_{LR4} \right)_{X_{LR}}$$
$$\left( \begin{bmatrix} 1 & 4 & 5 \\ 6 & 2 & 3 \\ 2 & 7 & 9 \end{bmatrix}_{LR1}, \begin{bmatrix} 8 & 4 & 6 \\ 5 & 9 & 1 \\ 8 & 6 & 8 \end{bmatrix}_{LR2}, \begin{bmatrix} 4 & 1 & 2 \\ 1 & 2 \end{bmatrix}_{LR3}, \begin{bmatrix} 3 & 7 & 6 \\ 5 & 3 & 7 \end{bmatrix}_{LR4} \right)_{X_{LR}}$$

**Figure 3.** Example of splitting an array image into a four-piece image, then inputting it into Generator EnsGAN.



**Figure 4.** Proposed EnsGAN model with n generators and a discriminator. As illustrated in Figure 3, splitting an array image or Image LR from one single image and inputting it into a multi-Generator  $(G_1, G_2, \ldots, G_n)$  then uses the ensemble technique to reconstruct the output Image HR.

# 3.2. Detection Baseline

In the proposed steel defect detector (SDD), we utilized DetectoRS as a detection baseline. This complex deep learning model has two main cores: recursive feature pyramid (RFP) and switchable atrous convolution (SAC). The RFP in DetectoRS, as shown in Figure 5, was placed at the micro-level, which incorporates extra feedback connections in bottom-up

prediction layers prediction layers prediction lavers top-down lavers bottom-up lavers

backbone layers, and SAC was placed at the macro-level, which convolves the features with different atrous rates and gathers the result using switch functions.

Figure 5. Visualization of recursive feature pyramid networks (RFP). The feedback connections are used to learn more deeply about the features' information.

In feature pyramid networks, the output feature  $\delta_i$  can be defined by:

$$\delta_{i} = \boldsymbol{\alpha}_{i}(\delta_{i+1}, \mathbf{x}_{i}), \text{ where } \mathbf{x}_{i} = \boldsymbol{\beta}_{i}(\mathbf{x}_{i-1}), \tag{7}$$

where  $\beta_i$  denotes the i-th stage of the bottom-up backbone, and  $\alpha_i$  denotes the i-th topdown FPN operation. The backbone, equipped with FPN, outputs a set of feature maps  $\{\delta_i \mid i = 1, \dots, S\}$ , where S is the number of the stages,  $\mathbf{x}_0$  is the input image, and  $\delta_{s+1} = 0$ . After adding the recursive feature pyramid (RFP) for feedback connections, the output feature  $\delta_i$  will be defined by:

$$\boldsymbol{\delta}_{i} = \boldsymbol{\alpha}_{i}(\boldsymbol{\delta}_{i+1}, \boldsymbol{x}_{i}), \text{ where } \boldsymbol{x}_{i} = \boldsymbol{\delta}_{i}(\boldsymbol{x}_{i-1}, \boldsymbol{\vartheta}_{i}(\boldsymbol{\delta}_{i})), \tag{8}$$

where  $\vartheta_i$  denotes the feature transformations before connecting them back to the bottom-up backbone. To implement the recursive operation, we unroll it to a sequential network, i.e.,  $\forall i = 1, ..., S; t = 1, ... T.$ 

$$\boldsymbol{\delta}_{i}^{t} = \boldsymbol{\alpha}_{i}^{t} \left( \boldsymbol{\delta}_{i+1}^{t}, \boldsymbol{x}_{i}^{t} \right), \text{ where } \boldsymbol{x}_{i}^{t} = \boldsymbol{\beta}_{i}^{t} \left( \boldsymbol{x}_{i-1}^{t}, \boldsymbol{\vartheta}_{i}^{t} \left( \boldsymbol{\delta}_{i}^{t-1} \right) \right).$$
(9)

where T is the number of unrolled iterations, and superscript t is used to denote operations and features at the unrolled step t. The detection baseline using DetectoRS with ResNeXt backbone, ResNext, performs very well in extracting the deeper feature information. Moreover, the RFP has a robust network because the output of the FPN is brought back to each stage of the bottom-up backbone thought-feedback connection, which means there is a double reading of the feature information from the defect image.

# 3.3. Boundary Localization

To improve the defect localization accuracy, we add Side-Aware Boundary Localization (SABL) [20] to replace the prediction head on the DetectoRS. SABL is a methodology for precise localization in object detection where each side of the bounding box is respectively localized with a dedicated network branch. There is a two-step localization scheme that first predicts a range of movement through bucket prediction and then pinpoints the precise position within the predicted bucket. Figure 6 shows that it first looks for the correct bucket, i.e., the one in which the boundary is located. Fine regression is then performed by predicting offsets using the selected bucket's centerline as a coarse estimate. This scheme enables very precise localization even in the presence of large variances in displacements.





**Figure 6.** Visualization of Side Aware Boundary Localization (SABL). To replace the proposal: the bucketing estimation is used to predict the buckets candidate and employs the fine regression to acquire the final bounding box (bbox) prediction.

#### 4. Experiments

For the experiments, we used the common steel surface defect dataset provided by Kaggle, which consists of 12,568 images. This dataset has been used for the real industrial application scenarios [52–54]. According to the original source (https://www.kaggle.com/ c/severstal-steel-defect-detection, accessed on 24 April 2022), this dataset has four class categories: Class 1, Class 2, Class 3, and Class 4, respectively. Specifically, the study in ref. [54] explained that Class 1 has pitted surface defect conditions, Class 2 has crazing defect conditions, Class 3 has scratch defect conditions, and Class 4 has patch defect conditions. A clear presentation of the defect conditions can be found in Figure 7. Moreover, we trained the dataset using an Nvidia RTX 3090 Ti 24GB with a splitting ratio of 80% of images for the training, 10% for validation, and 10% for the testing. Originally, this dataset used pixel-wise annotation labels. However, in this experiment, we converted the label into a bounding box using the Pascal VOC annotation format [55] to reduce the annotation cost and make the system more suitable for real-world scenarios.

Specifically, in the generator of the proposed preprocessing EnsGAN, for RRDB Network, we used 2D Convolution with a 3 × 3 kernel size and 1 × 1 strides with the same padding, and 2D Transpose Convolution with a 3 × 3 kernel size and 2 × 2 strides with the same padding; for the activation function, we used Leaky ReLU with  $\alpha$  = 0.2. Furthermore, in VGG28 Architecture, we used Convolution 2D with a 3 × 3 kernel size with the same padding, Leaky ReLU activation function with  $\alpha$  = 0.2, and fully connected layers with 100 neurons. In the localization and recognition network experiments, we used a single image as the batch size with stochastic gradient descent as the training optimizer. Moreover, the initial learning rate was set to 0.001, and the decay rate step was 0.0001 in epochs 16 and 19 of the 40 epochs. In addition, we only used one general data augmentation method, random flipping, to increase the diversity feature of the input image.

Finally, in this section, we discuss the results of our scope comparison for super resolution with ESRGAN [9] and Real-ESRGAN [10] in the preprocessing stage. Furthermore, to offer a complete view of performance results for the proposed method, we used the state-of-the-art object detection models that are feature pyramid network-based. From the YOLO family, these include: YOLOv4 [34], YOLOv5 [41], YOLOX [38]; from the free anchor method: FoveaBox [39]; from the transformer network: PVTv2 [40]; and from the R-CNN family: Faster R-CNN [19], and Cascade R-CNN [33]; for comparison with our proposed inspection system, EnsGAN-SDD.

#### 4.1. Results and Discussion of Proposed Preprocessing

In this subsection, we demonstrate the performance of our proposed method in the preprocessing stage. This experiment tried to sample the best split number ranging from 2 up to 32 based on the Peak Signal-to-noise Ratio (PSNR) to measure the quality between the original and the reconstructed image. We used three generated defect images produced by Real-ESRGAN, ESRGAN, and proposed EnsGAN to perform the quantitative comparison.

Moreover, we cropped the specific regions in these three defect samples images and have shown them in Figure 8 as Image 1 on the top, Image 2 in the middle, and Image 3 on the bottom for the qualitative comparison. As shown in Table 1, the objective results of the proposed EnsGAN surpass the quality of the original Real-ESRGAN and ESRGAN with the maximum PSNR numbers on the three images of 42.005, 39.278, and 43.135 when the split ratio was equal to 16. Moreover, Table 2 shows the benefit of our proposed EnsGAN in terms of processing speed efficiency. As the split ratio increased, the processing time simultaneously decreased. As shown in this table, for the three images, Real-ESRGAN and ESRGAN took 11.927, 12.473, 11.638, and 11.687, 11.723, 11.656 s, respectively. Our proposed method with a split ratio of 16 showed highly superior performance of 7.937, 8.250, and 8.123 s.



(**d**)

**Figure 7.** The Severstal steel defect dataset has four defect classes marked with different colors. (a) Class 1 has the pitted surface defect condition; (b) class 2 has the crazing defect condition; (c) class 3 has the scratches defect condition; and (d) class 4 has the patches defect condition.

For the qualitative comparison or subjective comparison performance, Figure 8 shows the original sample defect image and the generated defect sample images of results produced by Real-ESRGAN, ESRGAN, and the proposed EnsGAN. The Figure shows that the adversarial image of the proposed method enhanced the resolution quality, especially in the defect region marked by the red and yellow boxes. Moreover, the green box shows that the proposed method has better noise reduction on the original images and attained a smoother texture than ESRGAN did. On the other hand, the output of Real-ESRGAN enhanced the resolution, but unfortunately increased the noise because of aliasing issues. The problem arises because the traditional degradation model, which consists of a blur, downsampling, noise, and JPEG compression, leads to insufficient ability to model real-world degradations.

Based on the discussion above, we conclude that the proposed super-resolution method EnsGAN performs well in all aspects compared with the state-of-the-art methods. As we know, good sample data are very important for the neural network model. There-



fore, the quality of the produced image and efficiency of the speed performance from our proposed method can benefit the performance of the steel surface inspection system.

Original

Real ESRGAN

ESRGAN

EnsGAN

**Figure 8.** From left to right (zoom in for the best view): the original image and the generated image produced by Real-ESRGAN, ESRGAN, and the proposed EnsGAN, respectively. The red and yellow boxes show that the proposed EnsGAN does a better job of improving the resolution quality of the defect region. Moreover, the green box shows that the proposed method has better noise reduction on the original images and attains a smoother texture than ESRGAN does.

# 4.2. Results and Discussion of Proposed Inspection System

In the experiments on the localization and recognition network highlighted in this section, we used recall (REC), average precision (AP), and mean average precision (mAP) to align with the quantitative evaluation in Pascal VOC object detection [55]. First, Table 3 shows the ablation improvement of the proposed method. We used two state-of-the-art models, known as residual network 50 (ResNet-50), as the light backbone, and residual aggregation network 101 (X-101) as the deeper backbone for the DetectoRS as the defect detector's baseline. With the first ResNet-50 backbone, the mAP accuracy of the system was 77.1%. When we added EnsGAN to the system, the accuracy increased to 78.8%. After Side-Aware Boundary Localization (SABL) was substituted for the general prediction head, accuracy improved to 79%. Furthermore, we demonstrated the proposed system in a deeper network backbone. The initial mAP accuracy of the DetectorRS on top of X-101 was 78.5%. After we integrated the proposed EnsGAN and SABL, the mAP improved to 79.2% and 80.4%, respectively.

12 of 18

**Table 1.** Comparison of Real-ESRGAN, ESRGAN, and proposed EnsGAN for PSNR results on three sample images from the Severstal steel defect dataset. The defect samples in these three images are shown in Figure 8 in the following order: the top is Image 1, the middle is Image 2, and the bottom is Image 3.

Input Image	<b>Real-ESRGAN</b>	ESRGAN	EnsGAN (2×)		
Image 1	21.476	41.895	41.932		
Image 2	28.174	39.205	39.225		
Image 3	29.539	43.048	43.079		
Input Image	Real-ESRGAN	ESRGAN	EnsGAN (4×)		
Image 1	21.476	41.895	41,949		
Image 2	28.174	39.205	39.233		
Image 3	29.539	43.048	43.081		
Input Image	Real-ESRGAN	ESRGAN	EnsGAN (8×)		
Image 1	21.476	41.895	41.953		
Image 2	28.174	39.205	39.254		
Image 3	29.539	43.048	43.083		
Input Image	Real-ESRGAN	ESRGAN	EnsGAN (16×)		
Image 1	21.476	41.895	42.005		
Image 2	28.174	39.205	39.278		
Image 3	29.539	43.048	43.135		
Input Image	Real-ESRGAN	ESRGAN	EnsGAN (32×)		
Image 1	21.476	41.895	41.789		
Image 2	28.174	39.205	39.182		
Image 3	29.539	43.048	42.975		

**Table 2.** Comparison of EnsGAN, ESRGAN, and Real-ESRGAN for speed process on Severstal Steel defect dataset. The defect samples of these three images are shown in Figure 8 in the following order: the top is image 1, the middle is image 2, and the bottom is image 3.

Input Image	<b>Real-ESRGAN</b>	ESRGAN	EnsGAN ( $2\times$ )	
Image 1	11.927s	11.687s	11.210s	
Image 2	12.473s	11.723s	11.137s	
Image 3	11.638s	11.656s	11.199s	
Input Image	Real-ESRGAN	ESRGAN	EnsGAN (4×)	
Image 1	11.927s	11.687s	10.548s	
Image 2	12.473s	11.723s	10.497s	
Image 3	11.638s	11.656s	10.511s	
Input Image	Real-ESRGAN	ESRGAN	EnsGAN (8×)	
Image 1	11.927s	11.687s	9.572s	
Image 2	12.473s	11.723s	9.611s	
Image 3	11.638s	11.656s	9.557s	
Input Image	Real-ESRGAN	ESRGAN	EnsGAN (16×)	
Image 1	11.927s	11.687s	7.937s	
Image 2	12.473s	11.723s	8.250s	
Image 3	11.638s	11.656s	8.123s	
Input Image	Real-ESRGAN	ESRGAN	EnsGAN (32×)	
Image 1	11.927s	11.687s	6.950s	
Image 2	12.473s	11.723s	6.944s	
Image 3	11.638s	11.656s	6.877s	

Improvement	REC-1	REC-2	REC-3	REC-4	AP-1	AP-2	AP-3	AP-4	mAP
Baseline R-50	0.814	0.800	0.878	0.922	0.663	0.727	0.806	0.890	0.771
+ EnsGAN	0.791	0.850	0.883	0.953	0.647	0.807	0.799	0.898	0.788
+ SABL	0.791	0.850	0.863	0.969	0.668	0.813	0.775	0.904	0.790
Baseline X-101	0.802	0.850	0.883	0.984	0.653	0.765	0.803	0.922	0.785
+ EnsGAN	0.826	0.850	0.888	0.938	0.705	0.774	0.812	0.876	0.792
+ SABL	0.849	0.850	0.883	0.922	0.727	0.801	0.800	0.889	0.804

Table 3. The ablation improvement results of the proposed EnsGAN-SDD.

Table 4 further shows the proposed system's objective performance over the state-ofthe-art object detection models. The proposed EnsGAN-SDD on top of ResNet-50 improved the accuracy of the YOLO family: YOLOv4, YOLOv5, and YOLOX, anchor free method: Foveabox; transformer network: PVTv2; and R-CNN family: Faster R-CNN and Cascade R-CNN with a margin of 18.2%, 18.9%, 13.8%, 12.4%, 7.7%, 13.7%, and 11.5%, respectively. Figure 9, in turn, shows how accurate the proposed steel surface inspection system was in localizing four types of each defect class in the Severstal steel defect dataset. Moreover, Figure 10 shows the proposed system's capabilities in multi-class defect localization. These results show that the proposed method obtained high performance in detecting challenging steel product defects. In addition, the proposed system achieved 11.6 frames per second (FPS) in a single GPU Nvidia RTX 3090 Ti on top of the ResNet-50 backbone. This shows that EnsGAN-SDD can be applied in the steel industry, which requires a processing speed above 10 FPS, as explained in refs. [44,45].

**Table 4.** Comparison results of the proposed EnsGAN-SDD with the state-of-the-art object detection models.

Model	Baseline	REC-1	REC-2	REC-3	REC-4	AP-1	AP-2	AP-3	AP-4	mAP
YOLOv4	CSPDarknet	0.895	0.850	0.949	0.922	0.443	0.615	0.664	0.710	0.608
YOLOv5	CSPDarknet	0.837	0.900	0.921	0.906	0.423	0.693	0.607	0.682	0.601
YOLOX	CSPDarknet	0.802	0.900	0.921	0.828	0.529	0.628	0.745	0.707	0.652
FoveaBox	R-50	0.826	0.750	0.905	0.953	0.574	0.450	0.771	0.870	0.666
PVTv2	PVTv2-B5	0.895	0.850	0.934	0.953	0.579	0.637	0.779	0.856	0.713
Faster RCNN	R-50	0.756	0.850	0.879	0.891	0.531	0.746	0.636	0.700	0.653
Cascade RCNN	R-50	0.802	0.850	0.878	0.891	0.608	0.714	0.684	0.696	0.675
EnsGAN-SDD (our)	R-50	0.791	0.850	0.863	0.969	0.668	0.813	0.775	0.904	0.790
EnsGAN-SDD (our)	X-101	0.849	0.850	0.883	0.922	0.727	0.801	0.800	0.889	0.804



Figure 9. Sample of defect localization results for each class of the Severstal dataset.



Figure 10. Sample of multi-class defects localization results of the Severstal steel dataset.

# 5. Conclusions

The proposed EnsGAN-SDD can offer a solution to improve steel surface inspection systems. EnsGAN achieved a good PSNR on the steel defect dataset and had a low processing time relative to ESRGAN and Real-ESRGAN. Moreover, when it was integrated with robust detectors or DetectoRS and Side-Aware Boundary Localization (SABL) as steel defect detectors (SDD), the localization accuracy significantly improved over those of state-of-the-art object detection models while maintaining the necessary processing speed to meet standards for steel surface inspection systems. Furthermore, in the steel defect detection, but also in the difficulty of examples, e.g., the tiny defect size in Class 1 or pitted surface defect conditions, which have lower accuracy than other classes. Future work must extend the proposed preprocessing stage to address this issue.

**Author Contributions:** Conceptualization, F.A. and C.-Y.L.; methodology, F.A.; software, E.N.F.; validation, F.A., E.N.F. and C.-Y.L.; formal analysis, C.-Y.L.; investigation, F.A.; resources, C.-Y.L.; data curation, F.A.; writing—original draft preparation, F.A.; writing—review and editing, C.-Y.L.;

visualization, E.N.F.; supervision, C.-Y.L.; project administration, F.A.; funding acquisition, C.-Y.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by the Ministry of Science and Technology, Taiwan, under Grant MOST 110-2221-E-155-039-MY3, and Grant MOST 111-2923-E-155-004-MY3.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

**Data Availability Statement:** Author's datasets. Available online: https://www.kaggle.com/ competitions/severstal-steel-defect-detection/data (accessed on 24 April 2022).

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

# References

- Akhyar, F.; Lin, C.-Y.; Kathiresan, G.S. A Beneficial Dual Transformation Approach for Deep Learning Networks Used in Steel Surface Defect Detection. In Proceedings of the 2021 International Conference on Multimedia Retrieval, Taipei, Taiwan, 21–24 August 2021; pp. 619–622.
- Buslaev, A.; Iglovikov, V.I.; Khvedchenya, E.; Parinov, A.; Druzhinin, M.; Kalinin, A.A. Albumentations: Fast and flexible image augmentations. *Information* 2020, 11, 125. [CrossRef]
- Jung, S.; Kim, J. Adaptive and stabilized real-time super-resolution control for UAV-assisted smart harbor surveillance platforms. J. Real-Time Image Process. 2021, 18, 1815–1825. [CrossRef]
- 4. Greenspan, H. Super-resolution in medical imaging. Comput. J. 2009, 52, 43–63. [CrossRef]
- 5. Chen, X.; Zhai, G.; Wang, J.; Hu, C.; Chen, Y. Color guided thermal image super resolution. In 2016 Visual Communications and *Image Processing (VCIP)*; IEEE: Piscataway, NJ, USA, 2016; pp. 1–4.
- Rasti, P.; Uiboupin, T.; Escalera, S.; Anbarjafari, G. Convolutional neural network super resolution for face recognition in surveillance monitoring. In Proceedings of the International Conference on Articulated Motion and Deformable Objects, Palma de Mallorca, Spain, 13–15 July 2016; pp. 75–184.
- 7. Du, X.; He, Y. Gradient-guided convolutional neural network for MRI image super-resolution. Appl. Sci. 2019, 9, 4874. [CrossRef]
- 8. Song, H.; Yang, Y. Super-resolution visualization of subwavelength defects via deep learning-enhanced ultrasonic beamforming: A proof-of-principle study. *NDT E Int.* **2020**, *116*, 102344. [CrossRef]
- 9. Wang, X.; Yu, K.; Wu, S.; Gu, J.; Liu, Y.; Dong, C.; Qiao, Y.; Loy, C.C. Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.
- Wang, X.; Xie, L.; Dong, C.; Shan, Y. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 1905–1914.
- Dendorfer, P.; Elflein, S.; Leal-Taixé, L. MG-GAN: A Multi-Generator Model Preventing Out-of-Distribution Samples in Pedestrian Trajectory Prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 13158–13167.
- Ghosh, A.; Kulharia, V.; Namboodiri, V.P.; Torr, P.H.; Dokania, P.K. Multi-agent diverse generative adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8513–8521.
- 13. Adiban, M.; Safari, A.; Salvi, G. Step-gan: A step-by-step training for multi generator gans with application to cyber security in power systems. *arXiv* 2020, arXiv:2009.05184.
- Zhang, A.; Su, L.; Zhang, Y.; Fu, Y.; Wu, L.; Liang, S. EEG data augmentation for emotion recognition with a multiple generator conditional Wasserstein GAN. *Complex Intell. Syst.* 2021, 1–13. [CrossRef]
- 15. 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.
- 16. Ghiasi, G.; Lin, T.-Y.; Le, Q.V. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7036–7045.
- 17. Wang, J.; Chen, K.; Xu, R.; Liu, Z.; Loy, C.C.; Lin, D. Carafe: Content-aware reassembly of features. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 3007–3016.
- 18. Qiao, S.; Chen, L.-C.; Yuille, A. DetectoRS: Detecting Objects with Recursive Feature Pyramid and Switchable Atrous Convolution. *arXiv* 2020, arXiv:2006.02334.
- Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* 2015, 28, 91–99. [CrossRef]
- Wang, J.; Zhang, W.; Cao, Y.; Chen, K.; Pang, J.; Gong, T.; Shi, J.; Loy, C.C.; Lin, D. Side-aware boundary localization for more precise object detection. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2020; pp. 403–419.
- Guo, Z.; Wang, C.; Yang, G.; Huang, Z.; Li, G. MSFT-YOLO: Improved YOLOv5 Based on Transformer for Detecting Defects of Steel Surface. Sensors 2022, 22, 3467. [CrossRef]

- Amosov, O.S.; Amosova, S.G.; Iochkov, I.O. Deep Neural Network Recognition of Rivet Joint Defects in Aircraft Products. Sensors 2022, 22, 3417. [CrossRef] [PubMed]
- Su, B.; Chen, H.; Chen, P.; Bian, G.; Liu, K.; Liu, W. Deep learning-based solar-cell manufacturing defect detection with complementary attention network. *IEEE Trans. Ind. Inform.* 2020, 17, 4084–4095. [CrossRef]
- He, Y.; Song, K.; Meng, Q.; Yan, Y. An end-to-end steel surface defect detection approach via fusing multiple hierarchical features. IEEE Trans. Instrum. Meas. 2019, 69, 1493–1504. [CrossRef]
- Zhao, B.; Dai, M.; Li, P.; Xue, R.; Ma, X. Defect detection method for electric multiple units key components based on deep learning. *IEEE Access* 2020, *8*, 136808–136818. [CrossRef]
- Shi, J.; Li, Z.; Zhu, T.; Wang, D.; Ni, C. Defect detection of industry wood veneer based on NAS and multi-channel mask R-CNN. Sensors 2020, 20, 4398. [CrossRef]
- Liu, J.; Wang, C.; Su, H.; Du, B.; Tao, D. Multistage GAN for fabric defect detection. *IEEE Trans. Image Process.* 2019, 29, 3388–3400.
   [CrossRef]
- Zhang, G.; Pan, Y.; Zhang, L. Semi-supervised learning with GAN for automatic defect detection from images. *Autom. Constr.* 2021, 128, 103764. [CrossRef]
- Chen, S.-H.; Kang, C.-H.; Perng, D.-B. Detecting and measuring defects in wafer die using gan and yolov3. *Appl. Sci.* 2020, 10, 8725. [CrossRef]
- Arora, S.; Ge, R.; Liang, Y.; Ma, T.; Zhang, Y. Generalization and equilibrium in generative adversarial nets (gans). In *International Conference on Machine Learning*; PMLR: London, UK, 2017; pp. 224–232.
- Hoang, Q.; Nguyen, T.D.; Le, T.; Phung, D. MGAN: Training generative adversarial nets with multiple generators. In Proceedings
  of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
- 32. Durugkar, I.; Gemp, I.; Mahadevan, S. Generative multi-adversarial networks. arXiv 2016, arXiv:1611.01673.
- Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
- 34. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* 2020, arXiv:2004.10934.
- Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 568–578.
- Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
- 37. Jolicoeur-Martineau, A. The relativistic discriminator: A key element missing. *arXiv* 2018, arXiv:1807.00734.
- 38. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* 2021, arXiv:2107.08430.
- Kong, T.; Sun, F.; Liu, H.; Jiang, Y.; Li, L.; Shi, J. Foveabox: Beyound anchor-based object detection. *IEEE Trans. Image Process.* 2020, 29, 7389–7398. [CrossRef]
- Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pvtv2: Improved baselines with pyramid vision transformer. *Comput. Vis. Media* 2022, *8*, 415–424. [CrossRef]
- 41. Jocher, G.; Nishimura, K.; Mineeva, T.; Vilariño, R. yolov5. Code Repository. 2020. Available online: https://github.com/ ultralytics/yolov5 (accessed on 24 April 2022).
- 42. Riid, A.; Louk, R.; Pihlak, R.; Tepljakov, A.; Vassiljeva, K. Pavement distress detection with deep learning using the orthoframes acquired by a mobile mapping system. *Appl. Sci.* **2019**, *9*, 4829. [CrossRef]
- Du, F.-J.; Jiao, S.-J. Improvement of Lightweight Convolutional Neural Network Model Based on YOLO Algorithm and Its Research in Pavement Defect Detection. Sensors 2022, 22, 3537. [CrossRef]
- 44. Sun, X.; Gu, J.; Tang, S.; Li, J. Research progress of visual inspection technology of steel products—A review. *Appl. Sci.* **2018**, *8*, 2195. [CrossRef]
- 45. Wan, X.; Zhang, X.; Liu, L. An improved VGG19 transfer learning strip steel surface defect recognition deep neural network based on few samples and imbalanced datasets. *Appl. Sci.* **2021**, *11*, 2606. [CrossRef]
- 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.
- Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; He, K. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1492–1500.
- Dong, C.; Loy, C.C.; He, K.; Tang, X. Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 2015, 38, 295–307. [CrossRef]
- Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K.M. Enhanced Deep Residual Networks for Single Image Super-Resolution. In Proceedings
  of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017.
- 50. 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.
- Nah, S.; Kim, T.H.; Lee, K.M. Deep Multi-scale Convolutional Neural Network for Dynamic Scene Deblurring. In Proceedings of the IEEE Conference on Computer Vision and pattern recognition, Salt Lake City, UT, USA, 18–23 June 2018.

- 52. Boikov, A.; Payor, V.; Savelev, R.; Kolesnikov, A. Synthetic data generation for steel defect detection and classification using deep learning. *Symmetry* **2021**, *13*, 1176. [CrossRef]
- Ivanovska, M.; Perš, J.; Tabernik, D.; Skočaj, D. Evaluation of Anomaly Detection Algorithms for the Real-World Applications. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 6196–6203.
- 54. Wang, S.; Xia, X.; Ye, L.; Yang, B. Automatic detection and classification of steel surface defect using deep convolutional neural networks. *Metals* **2021**, *11*, 388. [CrossRef]
- 55. Everingham, M.; van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]