remote sensing

MDPI

Article

Leveraging Saliency in Single-Stage Multi-Label Concrete Defect Detection Using Unmanned Aerial Vehicle Imagery

Loucif Hebbache [1,*], Dariush Amirkhani [1], Mohand Saïd Allili [1], Nadir Hammouche [1] and Jean-François Lapointe [2]

[1] Department of Computer Science and Engineering, University of Quebec in Outaouais , Gatineau, QC J8X 3X7 , Canada
[2] Digital Technologies Research Center, National Research Council Canada, Ottawa, ON K1A 0R6, Canadaa
* Correspondence: hebl08@uqo.ca

Abstract: Visual inspection of concrete structures using Unmanned Areal Vehicle (UAV) imagery is a challenging task due to the variability of defects' size and appearance. This paper proposes a high-performance model for automatic and fast detection of bridge concrete defects using UAV-acquired images. Our method, coined the Saliency-based Multi-label Defect Detector (SMDD-Net), combines pyramidal feature extraction and attention through a one-stage concrete defect detection model. The attention module extracts local and global saliency features, which are scaled and integrated with the pyramidal feature extraction module of the network using the max-pooling, multiplication, and residual skip connections operations. This has the effect of enhancing the localisation of small and low-contrast defects, as well as the overall accuracy of detection in varying image acquisition ranges. Finally, a multi-label loss function detection is used to identify and localise overlapping defects. The experimental results on a standard dataset and real-world images demonstrated the performance of SMDD-Net with regard to state-of-the-art techniques. The accuracy and computational efficiency of SMDD-Net make it a suitable method for UAV-based bridge structure inspection.

Keywords: one-stage concrete defect detection; saliency; deep learning; UAV imagery

Citation: Hebbache, L.; Amirkhani, D.; Allili, M.S.; Hammouche, N.; Lapointe, J.-F. Leveraging Saliency in Single-Stage Multi-Label Concrete Defect Detection Using Unmanned Aerial Vehicle Imagery. *Remote Sens.* **2023**, *15*, 1218. https://doi.org/10.3390/rs15051218

Academic Editors: Claudio Piciarelli, Danilo Avola, Alessio Mecca and Marco Cascio

Received: 2 January 2023
Revised: 14 February 2023
Accepted: 18 February 2023
Published: 22 February 2023

1. Introduction

Visual inspection to detect surface defects is an important task for maintaining the structural reliability of bridges. Failing to do so can lead to disastrous consequences, as shown by the recent collapse of the Morandi bridge [1]. According to public data, out of 607,380 existing bridges in the U.S., nearly 67,000 are classified as structurally deficient, whereas approximately 85,000 are considered functionally obsolete. According to the National Research Council of Canada , one-third of Canada's highway bridges have some structural or functional deficiencies and a short remaining service life [2]. Currently, the inspection task is often conducted manually by inspectors, which could be a time-consuming and, sometimes, a cumbersome and painstaking process. Recently, there has been a growing shift toward using Unmanned Aerial Vehicles (UAVs) to perform inspection tasks, specifically for bridges, due to the enormous benefits such as the ability to access and rapidly inspect remote segments of the structure [3]. Drones can dramatically reduce the inspection time while ensuring the safety of hard-to-reach sites. Since they are efficient, fast, safe, and cost-effective, transportation authorities in several countries have been starting to apply UAV-based bridge inspection techniques [4]. However, concrete defect detection in UAV imagery remains more challenging than general defect detection, due to perspective and scale variation, changing lighting conditions, and overlapping of defects [5].

Early vision methods for defect detection used image processing techniques to design low-level features for defect description [6,7]. The common pipeline of these traditional methods is to use handcrafted features (e.g., Histograms of Oriented Gradients (HOGs),

Gabor filters, Local Binary Patterns (LBPs)) to train an appropriate classifier (e.g., SVM, AdaBoost) and deploy the classifier in a sliding window fashion or by generating region proposals on the input image [8]. Some of these methods have proven their efficiency for detecting defects such as cracks and corrosion [9]. However, their performance has been shown on simple images only. On the other hand, they are poorly extendible to address the detection of other defect types in a single framework.

The advent of deep learning methods in the last decade has enabled state-of-the-art performance for visual recognition problems such as image classification and object detection. Convolutional Neural Networks (CNNs) have become popular for image classification and object detection. Inspired by biological systems, CNNs (or ConvNets) have a unique capability to learn hierarchical and robust features directly from the training data by alternating convolution, pooling, and non-linear activation operations on the input image. With several training datasets such as ImageNet [10], strong CNN-based architectures have been proposed for extracting advanced features, which have drastically increased the performance of deep learning methods for visual recognition problems such as object classification and detection. These developments have unleashed huge opportunities for applications such as monitoring and visual inspection for anomaly detection [11,12]. This has also led to the publishing of several benchmark datasets with annotations, facilitating model training and testing. One of the most-popular datasets is the COncrete DEfect BRidge IMage dataset (CODEBRIM) proposed by Mundt et al. [12], which exhibits multiple defects, including: crack, spalling, exposed reinforcement bar, efflorescence, and corrosion (oxidation stains). Other datasets include MCDS [13] and SDNET (crack detection) [6].

The availability of annotated data has spurred the research on concrete defect classification and detection using deep learning [14]. More specifically, two-stage object detection was proposed first for localising specific concrete defects such as cracks and spalling [3,15–17]. With the advent of single-stage and faster methods (e.g., SSD [18], YOLO [19]), several real-time defect detection methods have been proposed [16,20–23]. Most of the above models have targeted specific defects such as cracks or spalling, whereas their validation is generally performed on images containing single defects against uniform and defect-free backgrounds. Such images are generally obtained by preprocessing the original images to remove non-relevant parts of the background (e.g., bridge structure elements, paintings, artefacts, other defects, etc.). In typical UAV-based inspection scenarios, however, images can be acquired at different viewpoints, resolutions, and ranges to camera, creating a huge variability in defect appearance and size, which makes them hard to detect. In addition, small and low-contrast defects occupy generally a tiny portion of the images against dominant and non-uniform backgrounds, which increases the difficulty of their localisation. Finally, defects with similar classes can overlap at the same location, which can cause defect mislabelling (e.g., oxidation occurs often with reinforcement corrosion and spalling) [24]. These challenges can drastically decrease the performance of previous detection methods since they are dedicated to simple scenarios [16,20–23].

The use of visual attention in deep learning models has enabled a substantial improvement of image classification [25]. Attention is a property of the human visual system, which processes the scene by exploiting the selective focus on its salient parts, rather than processing it as a whole [26]. Recently, there have been several attempts to incorporate attention to improve the performance of CNNs in large-scale classification tasks. These methods use additional modules of the networks to highlight discriminative local regions, which improves the classification. Some models implement attention as residual blocks [27], channelwise weighting across different network branches [28], or a combination of several modules [29]. Despite its potential, the use of attention for concrete defects' recognition is limited. Recently, some methods have attempted to use attention to enhance concrete defect classification for cracks [30–33] or overlapping defects [34]. These methods have shown good success in recognising single or overlapping defects against a defect-free uniform background. However, their performance can be limited in UAV imagery, where the images can contain several defects against highly cluttered backgrounds. Using multi-label

defect detection in such images is more advantageous since it enables detecting multiple defects characterised potentially by an overlapping structure. Moreover, having a selective mechanism that allocates higher attention to defective regions can enable a better detection of small and low-contrast defects surrounded by cluttered backgrounds.

In this paper, we investigated using attention to enhance concrete defect detection in the presence of the aforementioned challenging scenarios that typically arise in UAV-based inspection. We propose a fast and accurate model, coined the Saliency-based Multi-label Defect Detector (SMDD-Net), which leverages saliency and pyramidal feature extraction for the better detection and localisation of concrete defects. SMDD-Net integrates two modules in a single-stage pipeline: (1) the attention module, which extracts the global and local saliency to highlight regions of interest in the image, and (2) the detection module, which uses a Feature Pyramid Network (FPN) inspired by the RetinaNet model [35] to localise multiple and potentially overlapping defects. RetinaNet was chosen for its ability to detect defects at different scales and in the presence of imbalanced training datasets thanks to the use of the FPN and focal loss [36]. The two modules were integrated through residual skip connections to highlight regions of interest in the spatial representation of the pyramidal features. In other words, the attention module enables focusing the detection more on locally contrasted regions characterising general concrete defects. SMDD-Net has the ability to detect small and low-contrast defects, as well as localise defects in cluttered backgrounds, making it a suitable method for UAV-based inspection. Figure 1 (top) shows the pipeline of the SMDD-Net architecture, consisting of the attention and detection modules. The first module extracts the saliency features, whereas the second module optimises a multi-label loss function to detect overlapped defects. Our major contributions in this paper can be listed as follows:

- We propose the SMDD-Net architecture, which integrates attention in single-stage concrete defect detection. The attention module extracts global and local saliency maps, which highlight localised features for better detection of multiple defect classes in the presence of background clutter (e.g., artefacts, bridge structure elements, etc.). Contrary to detection methods that target single defect localisation against uniform backgrounds, SMDD-Net is capable of localising complex defects characterised by variable shapes, a small size, low-contrast, and overlap.
- We propose an attention module that is based on saliency extraction through gradient-based back-propagation of our feature extraction network. The back-propagation is performed via two paths: a global path, which highlights large-sized defect structures, and a local path, which highlights local image characteristics containing small and low-contrast defects. The two paths are fused using inter-channel max-pooling, and the output is added to the pyramidal features through residual skip connections.
- We demonstrate the performance of the SMDD-Net model on the well-known CODEBRIM dataset [12], which contains five classes of defects and several image examples with small, low-contrast, and overlapping defects. Our model leverages the benefits of the two detection paradigms: the high accuracy of two-stage detection and the high speed of one-stage detection. We compared also the performance of our model with state-of-the-art methods using several examples of real-world UAV images.

This paper is organised as follows: Section 2 discusses the related works. Section 3 presents the proposed method. Section 4 presents some experimental results for the validation. Finally, the paper ends with a conclusion and future work perspectives.

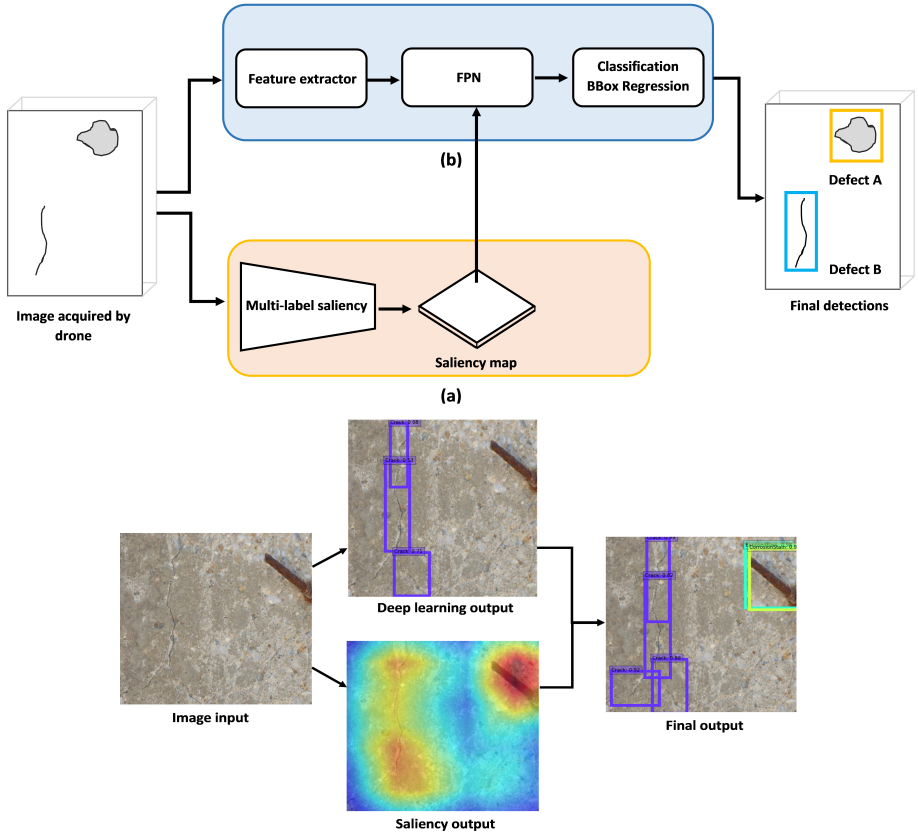

Figure 1. Pipeline of the proposed SMDD-Net method. On the top: (**a**) Saliency computation for the attention module; (**b**) multi-label one-stage concrete defect detection module. On the bottom: An example of our method depicting the benefits of using attention.

2. Literature Review

This section focuses on recent papers on concrete defect detection that apply bounding boxes to the localisation of defects. Following the evolution of object detection based on deep learning, the methods can be roughly divided into two categories: two-stage and one-stage concrete defect detection.

2.1. Two-Stage Concrete Defect Detection

Two-stage detection came first in the literature, where the detection process is divided into two steps: (1) region proposals and (2) bounding box regression and classification. Kim et al. [3] used a two-stage method to detect cracks by combining region proposals with rich features extracted by CNNs. The region proposals were extracted from the input images by a selective search, and the features were extracted by a CNN after image cropping and bounding box regression. However, the method can miss non-contrast cracks and requires heavy computation for feature extraction, which make it not easily applicable for real-time inspection. Haciefendioglu et al. [37] used Faster RCNN to detect road cracks. Because the final prediction is made using a single deep layer feature map, it is difficult to detect defects at different scales. Another limitation of this method is that it is proposed for single defect detection (cracks). Yao et al. [38] proposed a deep-learning-based method to detect bugholes. They used the inception module [39] to detect small-size defects and address the problem of a limited number of labelled examples in the training data. They also studied the effects of illumination and shadows on the detection accuracy. Wei et al. [40] developed a method based on Mask-RCNN for concrete surface bughole detection.However, most of these methods have been tested on cropped images, and they require generally a high computation time since they involve two separate stages in their detection pipeline.

Kang et al. [41] presented a technique for automatic detection, localisation, and quantification of cracks. Faster-RCNN was used to provide bounding boxes for cracks, which were then segmented to ensure pixel-level crack localisation. Finally, the segmented cracks were assessed for their thicknesses and lengths. Moreover, to increase the robustness of their method, the authors used a variety of complex backgrounds under varying environmental conditions. Mishra et al. [42] developed a two-stage automated method, based on YOLOv5, for the identification, localisation, and quantification of cracks on general concrete structures.In the first stage, cracks were localised using bounding boxes, whereas in the second stage, the length of the cracks, reflecting the damage severity, was determined. The main limitation of this work was that the method was tested on very simple cases involving cropped and close-range images, on which cracks can be easily located.In other words, it cannot be easily used for real-time inspection applications.

Xu et al. [43] developed a modified method for the detection and localisation of multiple seismic damages of reinforced concrete structures (i.e., cracking, spalling, rebar exposure, and rebar buckling). The Region Proposal Network (RPN) uses CNN features to define initial bounding boxes for damage, which are then refined using Fast-RCNN. One limitation of this method is that the RPN is trained by extracting all region anchors in the mini-batch from a single image. Because all samples from a single image may be correlated, it is possible that the network takes a long time to reach convergence. Li et al. [44] proposed a unified model for concrete defect detection and localisation developed using Faster-RCNN. The model takes an image and computes feature maps using the shared bottom layers and then applies the defect detection network at the top layers. However, combining methods will incur a large computational cost and will not be suitable for real-time applications. Wan et al. [45] presented a method based on vision transformers for concrete defect detection. However, it is computationally intensive, which makes it not applicable for real-time applications.

2.2. One-Stage Concrete Defect Detection

Teng et al. [46] and Deng et al. [21] used one-stage YOLOv2, pre-trained on ImageNet, for crack detection. These works showed good performance for real-time crack detection on close-range and cropped images. However, they are not easily extendible to other types of defects. Cui et al. [20] improved the one-stage detector YOLOv3 [47] to detect erosion. As a pre-trained model, they used Darknet53, which uses the Mish activation function. However, the method has poor performance when images have different aspect ratios. In addition, the method is limited to single defect detection. Zhang et al. [48] also used YOLOv3 for detecting multiple concrete bridge defects, including cracks, pop-outs, spalling, and exposed bars. The model was pre-trained on the MS-COCO dataset. However, this approach has issues detecting defects at different scales.

Wu et al. [49] used the YOLOv4 model for crack detection. A pruning strategy was employed to overcome the issue of over-parameterised CNNs, as well as to increase the detection speed. Wang et al. [50] developed an automated one-stage concrete defect detection method that was composed of two parts: the EfficientNetB0 backbone network and the detector. To increase the detection accuracy, the detector gathers feature information from three scales and merges low-level and high-level features through an up-sampling procedure. However, they used small, cropped and close-range images and used them only for two types of defects (cracks and exposed bars). Jiang et al. [22] used improved YOLOv3 for multiple defect detection in concrete bridges. They combined EfficientNet-B0 and MobileNetV3 pre-trained on MS-COCO as a baseline and depthwise separable convolutions. The method did not show high performance when dealing with different scales. Kumar et al. [51] used YOLOv3 for detecting spalling and cracks. The method was validated only on cropped and close-range images. Zou et al. [8] used YOLOv4 to detect defects such as cracks, spalling, and exposed/buckled rebar. They employed depthwise separable convolutions to decrease the computational costs, which boosted the detection speed.

All the previous methods achieved a noticeable success when tackling single defect detection such as cracks, but they lost efficiency when dealing with other types of defects. Indeed, developing a unique model that can deal with all defect classes is hard to achieve. Defect detection and classification, particularly in UAV-based inspection, pose several challenges. First, even if defects can be categorised into different classes, there is huge intra-class variability due to varying illumination conditions and viewpoint and scale changes [14]. On the other hand, some defect classes can have a huge overlap (e.g., oxidation, exposed bar, spalling), which can cause defect mislabelling.

3. Proposed Method

The core idea in this paper is to improve defect detection and localisation by leveraging object detection with saliency. This was motivated by the fact that defects are usually characterised by their local contrast with regard to their defect-free background surface. Combining saliency and object detection is intuitively appealing to focus attention on parts with the highest local contrast for region proposals where potential defects can be located. Saliency can also enhance the feature representation of low-contrast defects (e.g., cracks, efflorescence), which will enhance their region proposal scores for detection. The pipeline of the SMDD-Net method is depicted in Figure 2, which is composed of two main modules: (a) the attention module and (b) the multi-label one-stage concrete defect detection module. The attention module was designed to enhance the feature representation by putting more emphasis on parts of the image containing local discontinuities. In other words, the saliency map will pinpoint parts of the image to enable the generation of region proposals on the concrete surface with the highest defect potential. The second module scans the region proposals to ascertain the defects through bounding box regression and classification. The latter was fine-tuned for multi-label defect detection using the CODEBRIM dataset [12]. In what follows, we describe each module separately by giving an example of the implementation of each module, as shown in Figure 2, where Grad-CAM [52] and RetinaNet [35] were used as the baselines for implementing our architecture.

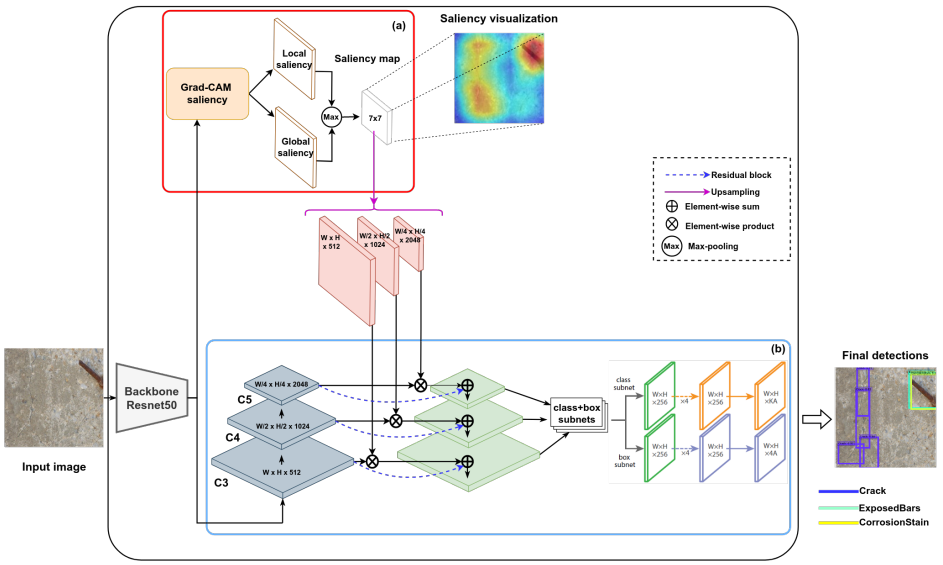

Figure 2. Pipeline of the proposed SMDD-Net method: (**a**) attention module enhancing regional feature representation for defect detection; (**b**) the multi-label one-stage defect detection module uses the RetinaNet model.

3.1. Saliency for Defect Region Proposals

The purpose of saliency, which is based on cognitive studies of visual perception, is to enable identifying regions that exhibit local contrast with regard to their surroundings.

This problem is very much aligned with the one of concrete defect detection, where defects usually exhibit some contrast with regard to the immediate defect-free concrete surface. While not every salient region constitutes a defect, most defects exhibit some degree of local saliency. Thus, computing saliency and using it to draw attention to defective regions can be very useful to reduced false negatives (e.g., due to small-sizes and low-contrast defects), which is a common problem in concrete defect detection [53].

Early methods for saliency detection were based on hand-crafted features such as local histograms, calculated on the region support such as superpixels, and used contrast between the features' local statistics [54] or graph ranking [55] to assign saliency scores to pixels. These methods have been successful at distinguishing compact salient regions without requiring extensive training, but they cannot be readily applied to distinguish defects. First, some defects such as cracks are hard to describe using local statistics. Second, defects do not always occupy compact regions, but rather, come with different sizes and shapes and are sometimes even fragmented into different parts. By using deep learning, local contrast patterns characterising saliency can be detected by training an appropriate model. For example, Hou et al. [56] introduced short connections to a CNN architecture that have the role of extracting holistic local discontinuities. Selvaraju et al. [52] proposed the *Gradient-weighted Class Activation Map* (Grad-CAM) method, which is aimed at producing a coarse localisation map highlighting the important regions of the image causing the prediction of a concept. Grad-CAM can be applied to a wide variety of CNN model families and correctly identifies the general region of the object that is predicted by the model.

In order to search for regions in the input image that can potentially contain defects, the SMDD-Net method integrates a local version of Grad-CAM to highlight regions in the image that are responsible for defect class activations. For this purpose, the input image is subdivided into several overlapping parts, where a saliency map is calculated separately for each part (see Figure 3b for an illustration). In order to maximise the defect detection rate, the maximum saliency is retained in each part of the division to produce the final saliency map of the input image.

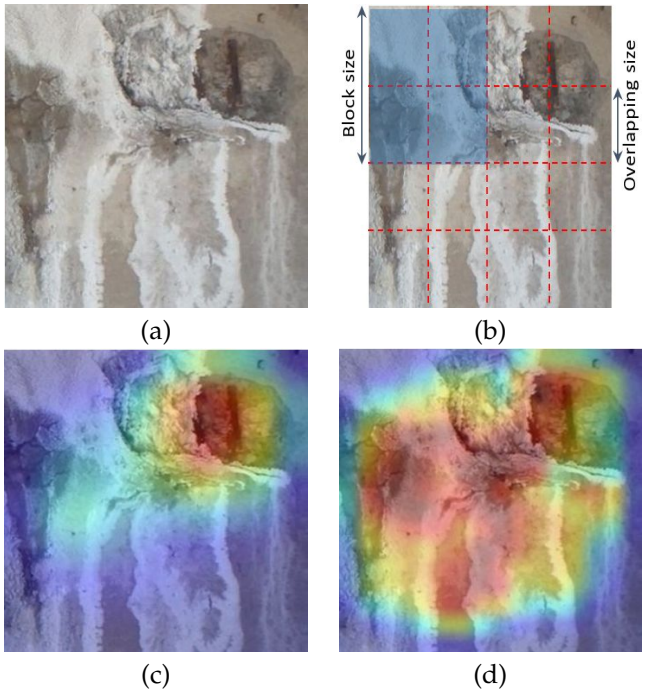

Figure 3. Illustration of the saliency extraction module for concrete defect detection: (**a**) input image, (**b**) overlapping block image subdivision, (**c**) global saliency extraction, and (**d**) local saliency extraction.

More formally, let us have K class defects and an input image I, which can contain one or several instances of these defect classes. We first subdivided the image into n overlapping parts $P_1, ..., P_n$, as shown in Figure 3b. The saliency induced on part P_i of the k-th feature map after activating defect class y_c is given by $\frac{\partial y_i^c}{\partial A_i^k}$, and the weighted combination of the forward activation maps is given by

$$L_i^c = ReLu(\sum_k \alpha_{ik}^c A_i^k) \quad \text{where} \quad \alpha_{ik} = \frac{1}{Z}\sum_p \sum_q \frac{\partial y_i^c}{\partial A_i^k(p,q)} \tag{1}$$

where α_{ik} are the neurons' importance weights and Z is the spatial size of the feature map (p and q are the spatial coordinates of the feature maps).

Moreover, to take into account correlation effects between defect classes (e.g., oxidation caused by exposed bar, which is itself caused by spalling), we built for class label c a subset of labels Q_c that might occur as a consequence of y^c (e.g., c = *spalling*, then Q_c = {exposed bar, oxidation}). The final activation map produced for part P_i is then given by the following formula:

$$S_i = ReLu\left(\max\left(\sum_k \alpha_{ik}^c A_i^k, Avg_{c' \in Q_c} \sum_k \alpha_{ik}^{c'} A_i^k\right)\right) \tag{2}$$

Finally, we built the complete local saliency map S_{local} for the whole image I by stitching the local saliency maps S_i generated for parts P_i into their corresponding positions in image I. For the overlapped parts between two patches, say P_i and P_j, we took the maximum saliency between S_i and S_j for the overlapped area. Furthermore, to enable a global saliency enhancement of the feature representation, another saliency map S_{global} was generated by feeding the whole image to Grad-CAM. Figure 3 illustrates the saliency extraction module on an image containing several defects spreading across the entire image. While the global saliency identified the most-salient defect on the right (exposed bar + spalling), the local saliency identified the efflorescence occupying a large portion of the image and the spalling on the left.

3.2. Multi-Label One-Stage Defect Detection

The concern of defect detection is not only to classify defects, but also to localise them inside bounding boxes. Note that the one-stage detection models are mainly focused on computational efficiency, which enables fast detection at the expense of limited accuracy [57]. Here, we tried to gain the benefits of the two worlds by simply narrowing the search area of the one-stage detector, thus enabling rapid and accurate defect detection at once. Using the computed saliency map, we generated region candidates around the salient parts to narrow the search space for defects, similar to two-stage object detection using deep learning [58].

In the majority of object detection methods such as RetinaNet, the bounding boxes are assigned a single label. For concrete defect detection, however, several defects can co-occur together at the same location. For example, exposed bar is often linked to corrosion or spalling defects. Likewise, cracks can be linked to efflorescence defects. To take into account this reality, we revisited the RetinaNet focal loss to implement a multi-label version by enabling assigning more than one label for each bounding box. The original Focal Loss (*FL*) for RetinaNet is an enhancement of the Cross-Entropy loss (CE), which suffers from an extreme foreground–background class imbalance problem due to the dense sampling of anchor boxes. Since there are hundreds of anchor boxes in each pyramid layer, only a few will be assigned to a ground-truth object, while the vast majority will be the background class, which can collectively overwhelm the model. To mitigate this problem, *FL* reduces the loss

contribution from easy examples and increases the importance of correcting misclassified examples. More formally, the *FL* for a given object is given by

$$FL(p_c) = -\alpha_c(1-p_c)^{\gamma} log(p_c) \tag{3}$$

where p_c is the estimated probability for the ground-truth class c. The weight α is used to mitigate the class imbalance problem, which could be set by the inverse class frequency or treated as a hyper-parameter to be set by cross-validation. The exponent γ is used to modulate the factor $(1-p_c)$ to the cross-entropy loss. The focusing parameter γ smoothly adjusts the rate at which easy examples are down-weighted. When $\gamma = 0$, *FL* is equivalent to CE, and as γ increases, the effect of the modulating factor increases.

To take into account the multi-label aspect for defect detection, we augmented the ground-truth by creating copies of the bounding boxes containing several defect labels. This will enable regressing different candidates in the region proposals to target the same ground-truth bounding box. The final effect of this is to produce tightly overlapping bounding boxes that can be labelled differently from each other. More formally, the *FL* for a given bounding box location having a set $\mathcal{L} = \{c_1, ...c_n\}$ of labels is given by

$$FL(p_{c_1}, .., p_{c_n}) = -\sum_{c_i \in \mathcal{L}} \alpha_{c_i}(1-p_{c_i})^{\gamma} log(p_{c_i}) \tag{4}$$

In the experiments of this study, we set α_{c_i} = 0.25 and γ = 2. For bounding box regression, we used the smooth L_1 loss defined in [59], which is less sensitive to outliers than the L2 loss. The final training loss was the sum of FL and the smooth L_1.

As illustrated in the implementation example of Figure 2, the detection module in the RetinaNet architecture uses features extracted from a ResNet50 [60] backbone fine-tuned on the CODEBRIM dataset [12]. From this backbone, three pyramidal features, F_3, F_4, and F_5, are extracted corresponding to the 3rd, 4th, and 5th convolutional blocks of the ResNet50 backbone. These features are combined with the saliency maps using the following formula:

$$F_i' = F_i \oplus (F_i \otimes MAX(\uparrow S_{local}, \uparrow S_{global})), i = 3, 4, 5 \tag{5}$$

where $\uparrow$ designates an up-sampling operation, $\oplus$ designates a skip connection adding a residual, and $\otimes$ designates pointwise feature multiplication. The new features F_i' are then fed to the FPN, followed by bounding box regression and classification sub-networks.

4. Experiments

To evaluate our method, we conducted experiments on bridge defect detection and compared our results with previous methods. Here, we briefly present the dataset and metrics used for the quantitative evaluation, as well as some important implementation details about our method.

The most-recent dataset that exists for bridge defect detection is the COncrete DEfect BRidge IMage Dataset (CODEBRIM) [12], which is composed of six classes: background (2490) and five defect classes: crack (2507), spallation (1898), exposed bars (1507), efflorescence (833), and corrosion stain (1559). The images were acquired at high-resolution using drones, then resized to fit the input resolution required by the RetinaNet baseline method. In order to make the model perform better, especially on small objects and with changing viewing conditions, we augmented this dataset using bounding box data augmentation techniques such as brightness, mosaic, and shear. Bounding-box-level augmentation generates new training data by only altering the content of a source images with the bounding boxes [61].

4.1. Implementation Details

For our the experiments, we used Google Colab Pro+, which provides 52 GB of RAM alongside 8 CPU cores and priority access to GPU P100. The CODEBRIM dataset was split into training, validation, and test sets with an approximate ratio of 70%, 20%, and 10%. We

trained our model along 30 epochs with 2370 iterations for each epoch, a batch size of 4, and fp16 mixed-precision training. Using transfer learning, ResNet50 was first pretrained on the CODEBRIM dataset, then used as the backbone for both the saliency (Grad-CAM) and detection modules. To help reduce False Positives (FPs), we added 10% background images to the dataset, with no objects (labels). In the inference phase, to select the best bounding box from the multiple predicted bounding boxes, we used the Non-Maximal Suppression (NMS) technique with a threshold equal to 0.45.

4.2. Evaluation Metrics

To evaluate our method's performance and compare it to the other models, we used the mean Average Precision (mAP) metric, representing the average of the AP of all classes. In brief, the AP is the enclosed area under the precision–recall curve drawn for a single class. The larger the area, the higher the AP value is and the better performance the method has for detection. The precision and recall metrics are defined as follows:

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

where TP, FP, and FN represent the number of True Positives, False Positives, and False Negatives, respectively. To classify the detections as a TP or FP, the Intersection Over Union (IOU) between the predicted and ground-truth bounding boxes was used. The $mAP@0.5:0.95$ was used to select the best weights of the model on the validation set, and the $mAP@0.5$ was used to evaluate the method on the test set.

4.3. Results Analysis

To show the importance of the attention module, we performed an ablation study using SMDD-Net for defect detection in four different scenarios: (1) SMDD-Net without the attention module, (2) SMDD-Net without global saliency, (3) SMDD-Net without local saliency, and (4) SMDD-Net without residual skip connections. This allowed us to see the advantages of employing saliency. The obtained AP values for each defect class, as well as the mAP value in the first scenario are presented in Table 1. The results of the remaining scenarios are presented in Table 2. According to the results of the second and third scenarios, the advantages of local and global saliency were complimentary. In other words, the modules contribute equally to detection. In the last scenario, with deleting residual blocks, the accuracy drastically decreased since the combination was performed through feature multiplication only. The saliency map extracted from the attention module may contain very small values, which can destroy the pyramidal feature through multiplication. This is reflected in the decreased accuracy for this scenario. In conclusion, all the components of the attention module were important to achieve a proven detection accuracy. Figure 4 shows the results of some detection samples by SMDD-Net. It should be noted that, in certain cases, the detection had a very high score (score = 100%). Note also that, even though the saliency highlighted regions other than the defects, the detection module did not output any false detections at these regions (see Examples 1, 3, and 4).

Table 1. Results of the SMDD-Net evaluation on the test set without using the attention module.

Classes	AP@0.5	mAP@0.5
Crack	0.98	
Spallation	0.96	
Efflorescence	0.80	0.88
Exposed bars	0.90	
Corrosion stain	0.74	

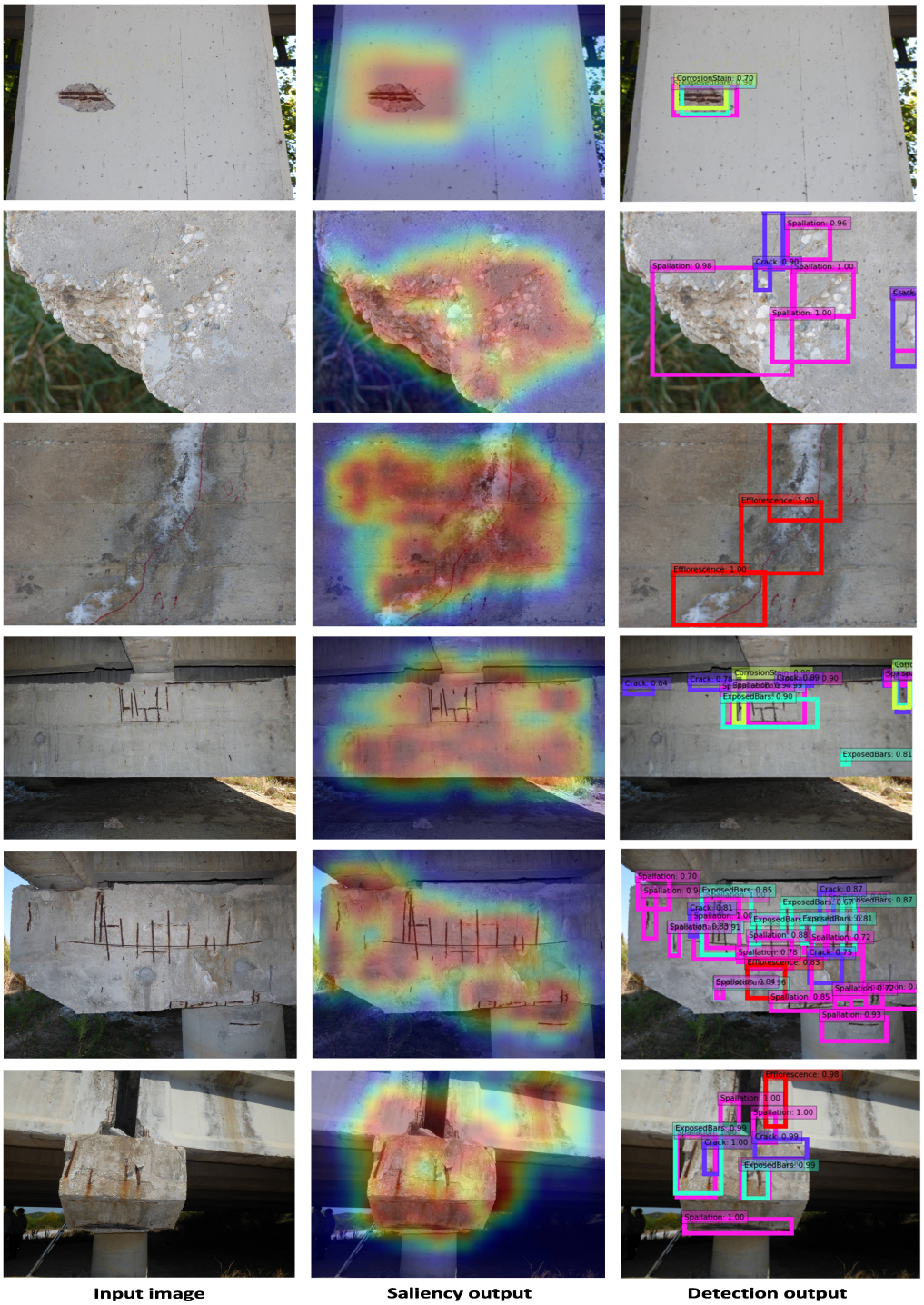

Figure 4. Detection samples from SMDD-Net.

Table 2. Ablation test results for SMDD-Net.

Scenarios	#Param.	mAP@0.5
SMDD-Net without attention module	36.5 M	0.88
SMDD-Net without global saliency	36.5 M	0.95
SMDD-Net without local saliency	36.5 M	0.93
SMDD-Net without residual block	36.5 M	0.46
SMDD-Net	36.5 M	0.99

To better assess the merits of the SMDD-Net method, we compared its performance against seven other methods [35,62–67]. Patel et al. [65] used an improved Faster-RCNN method by adding a multi-label loss function for concrete defect detection. They pointed out various elements that affect the network's accuracy, such as the bounding boxes' inability to accurately depict the complex shape of defect patches and further inaccuracies in the CODEBRIM database's annotation. Xiong et al. [66] used the pre-trained YOLO-v4 network on the MS COCO dataset and fine-tuned the entire model on the CODEBRIM training dataset for concrete defect detection in order to compare the visual inspections with the automated ones. The other methods were implementations of the baseline YOLOv5-l [67], YOLOv8-l [64], RetinaNet [35], YOLOX [62], and YOLOR [63]. Table 3 shows a comparison of the SMDD-Net method with these methods. Clearly, our method outperformed the other methods in terms of detection accuracy. Noticeably, the performance of the baselines YOLOR and YOLOX came in second and third, respectively, in rank behind the SMDD-Net method, but with a significant gap with regard to the latter (the gap was 7.3% with regard to YOLOX and 9.9% with regard to YOLOR). These results demonstrated, among other things, the benefit of using the attention module for boosting the local feature representation, which significantly enhanced the detection accuracy. Figure 5 shows a comparison of the validation curves for the implemented methods, demonstrating that SMDD-Net, RetinaNet, and YOLOX converged faster than the other methods.

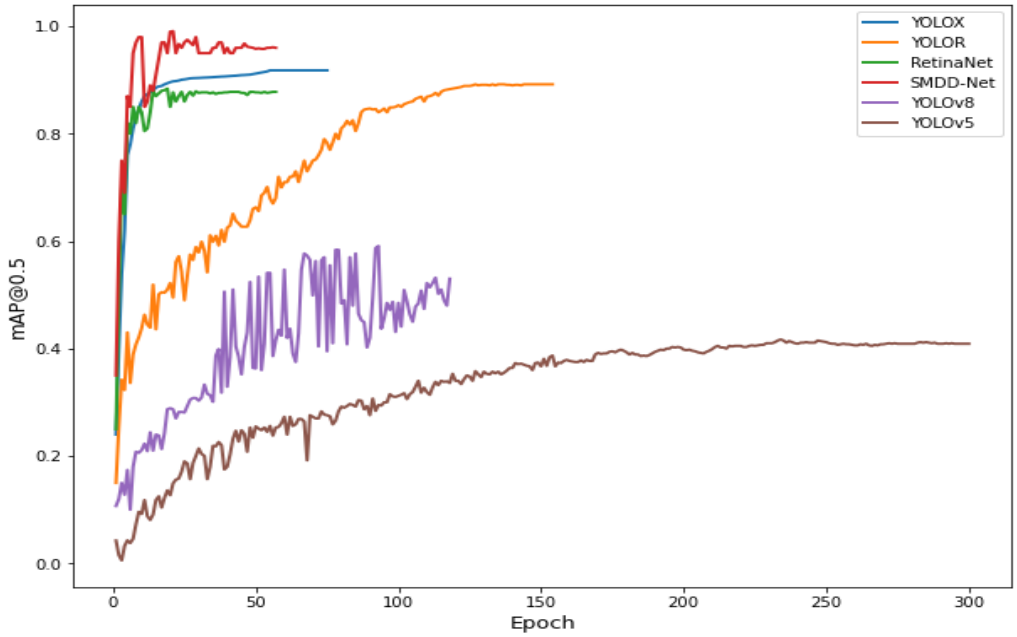

Figure 5. Comparison of the validation curves.

To qualitatively compare the implemented methods reported in Table 3, Figure 6 presents some illustrative examples of concrete defect detection. These examples we selected on purpose because they were particularly challenging. In the first example, the image was acquired at close range and contained two defects (a long crack on the left and corrosion on the right). Only SMDD-Net and YOLOR detected the corrosion. Note also that, due to the multi-label detection, the defect on the right was assigned another bounding box depicting an exposed bar, which was a valid detection. The second to fifth examples were acquired at a medium range. In the second example, the image contained several defects (two oxidation stains and a crack). SMDD-Net detected all these defects, whereas the other methods missed one or multiple defects. In the third example, the image contained various small cracks that were detected by SMDD-Net, but most of them were missed by the other methods. In the fourth example, the image contained a crack in the middle against a huge background part. Only SMDD-Net was able to detect it, as well as

some efflorescence. In the fifth example, the image contained an exposed bar and several small cracks, which can barely be seen even with the naked eye. SMDD-Net detected all the defects, but the other methods missed most of the cracks. Finally, in the sixth example, the image contained corrosion within a spalling area, as well as a significant part with efflorescence. Only SMDD-Net and YOLOv8 were successful in detecting the spalling and corrosion. These results demonstrated the efficiency of the SMDD-Net method in detecting defects in challenging scenarios.

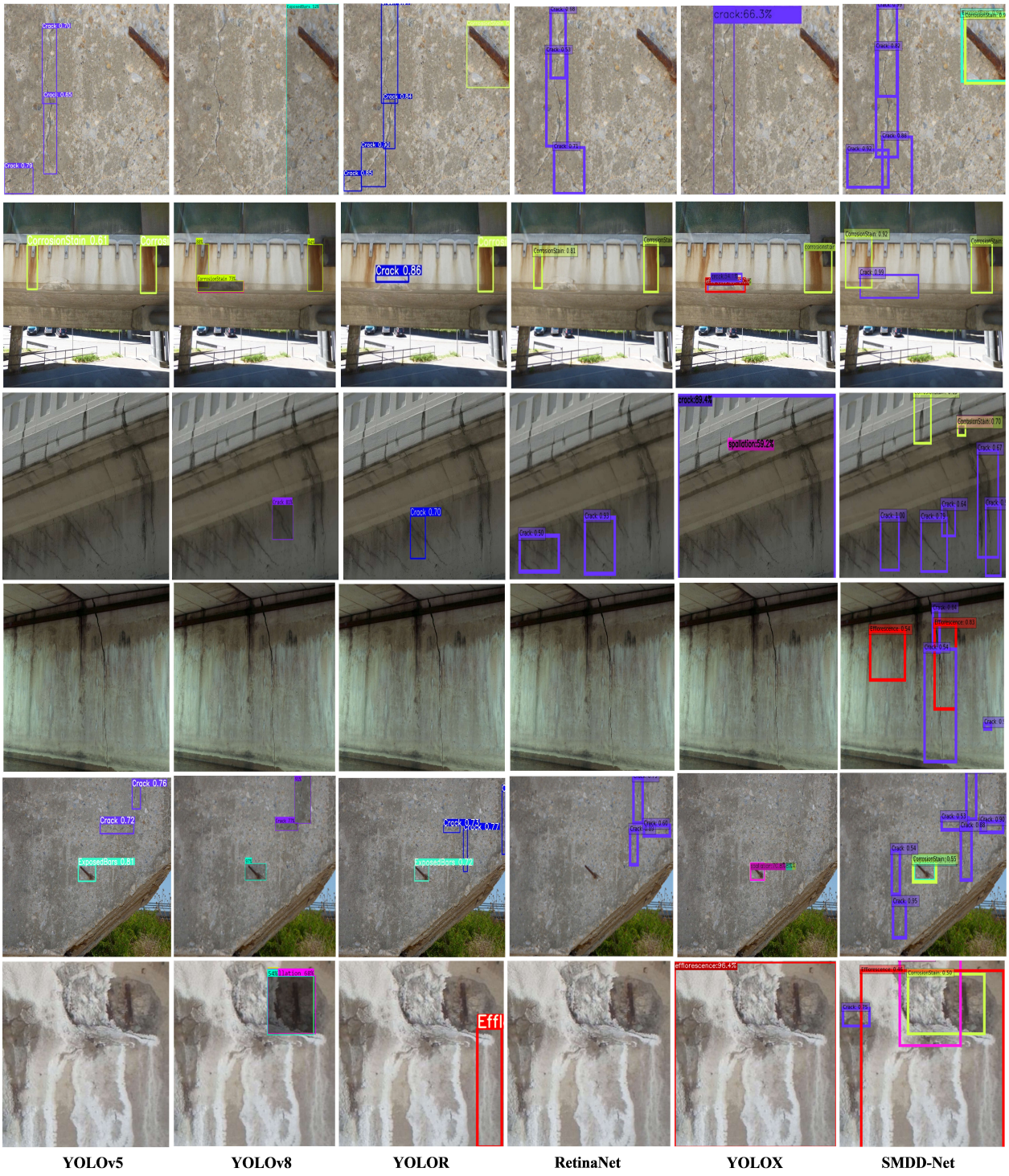

Figure 6. Visualisation of the compared results from concrete surface images.

Table 3. Comparison of SMDD-Net with other methods.

Method	#Param.	One-Stage	Two-Stage	Classification Head	Bounding Box Head	mAP@0.5 (%)	Speed (s)
Patel et al. [65]	74.4 M	-	✓	BCE [1] Loss	Smooth L1 Loss	91.2	0.14
Xiong et al. [66]	52.9 M	✓	-	BCE Loss	Smooth L1 Loss	22.7	0.03
YOLOv5-l [67]	46.1 M	✓	-	BCE Loss	Smooth L1 Loss	41.7	0.02
YOLOv8-l [64]	43.7 M	✓	-	BCE Loss	Smooth L1 Loss	59.6	0.02
RetinaNet [35]	36.5 M	✓	-	Focal Loss	Smooth L1 Loss	88.4	0.07
YOLOX-l [62]	54.2 M	✓	-	BCE Loss	Smooth L1 Loss	91.8	0.04
YOLOR-P6 [63]	36.9 M	✓	-	BCE Loss	L2 Loss	89.2	0.04
SMDD-Net	36.5 M	✓	-	Focal Loss	Smooth L1 Loss	99.1	0.11

[1] Binary Cross-Entropy.

5. Conclusions

In this paper, a novel one-stage concrete defect detection method was proposed. This method leverages attention in the form of saliency to focus the detection on the most-important parts of the image. This had the benefit of boosting the overall detection accuracy, but also detecting non-contrasted and small defects, which were missed by most of the one-stage and two-stage detection methods. Our attention module was implemented by fusing local and global saliency maps extracted using back-propagation. To improve the feature representation for defect detection, the saliency features were combined with the pyramidal features. The experimental results demonstrated that the proposed method outperformed the other methods in terms of the detection accuracy, while being computationally efficient. It also enabled a better identification and localisation of overlapping defects. Although the results were shown for concrete bridge defect detection, SMDD-Net can be readily applied without major modifications to any other concrete structure defect detection task.

Author Contributions: Conceptualisation, L.H., M.S.A., J.-F.L. and D.A.; methodology, L.H., D.A. and M.S.A.; software, L.H., M.S.A. and N.H.; validation, L.H., M.S.A., D.A., N.H. and J.-F.L.; formal analysis, L.H. and M.S.A.; investigation, L.H., M.S.A., D.A. and J.-F.L.; resources, M.S.A. and J.-F.L.; data curation, L.H., D.A. and N.H.; writing—original draft preparation, L.H., M.S.A. and D.A.; writing—review and editing, L.H., M.S.A., D.A., N.H. and J.-F.L.; visualisation, L.H., D.A. and N.H.; supervision, M.S.A. and J.-F.L.; project administration, M.S.A. and J.-F.L.; funding acquisition, M.S.A. and J.-F.L. All authors have read and agreed to the published version of the manuscript.

Funding: This project was supported in part by collaborative research funding from the National Research Council of Canada's Artificial Intelligence for Logistics Program.

Data Availability Statement: The dataset supporting the reported results in this study can be found at https://zenodo.org/record/2620293#.Y69i_HbMK3D (accessed on 1 January 2023).

Conflicts of Interest: The funders had no role in the design of the study; in the collection, analyses, or interpretation of the data, nor in the writing of the manuscript. The authors declare no conflict of interest.

References

1. Calvi, G.M.; Moratti, M.; O'Reilly, G.J.; Scattarreggia, N.; Monteiro, R.; Malomo, D.; Calvi, P.M.; Pinho, R. Once upon a time in Italy: The tale of the Morandi Bridge. *Struct. Eng. Int.* **2019**, *29*, 198–217. [CrossRef]
2. Available online: https://nrc.canada.ca/en/research-development/products-services/technical-advisory-services/infrastructure-expertise-technology-assessment (accessed on 28 December 2022).
3. Kim, I.H.; Jeon, H.; Baek, S.C.; Hong, W.H.; Jung, H.J. Application of Crack Identification Techniques for an Aging Concrete Bridge Inspection Using an Unmanned Aerial Vehicle. *Sensors* **2018**, *18*, 1881. [CrossRef] [PubMed]
4. Mandirola, M.; Casarotti, C.; Peloso, S.; Lanese, I.; Brunesi, E.; Senaldi, I. Use of UAS for damage inspection and assessment of bridge infrastructures. *Int. J. Disaster Risk Reduct.* **2022**, *72*, 102824. [CrossRef]
5. Liu, W.; Quijano, K.; Crawford, M.M. YOLOv5-Tassel: Detecting tassels in RGB UAV imagery with improved YOLOv5 based on transfer learning. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *13*, 8085–8094. [CrossRef]
6. Dorafshan, S.; Thomas, R.; Maguire, M. Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete. *Constr. Build. Mater.* **2018**, *186*, 1031–1045. [CrossRef]

7. Jahanshahi, M.R.; Kelly, J.S.; Masri, S.F.; Sukhatme, G.S. A Survey and Evaluation of Promising Approaches for Automatic Image-Based Defect Detection of Bridge Structures. *Struct. Infrastruct. Eng.* **2009**, *5*, 455–486. [CrossRef]
8. Zou, Z.; Shi, Z.; Guo, Y.; Ye, J. Object Detection in 20 Years: A Survey. *arXiv* **2019**, arXiv:1905.05055v1.
9. Chen, C.; Seo, H.; Jun, C.; Zhao, Y. A potential Crack Region Method to Detect Crack Using Image Processing of Multiple Thresholding. *Signal Image Video Process.* **2022**, *16*, 1673–1681. [CrossRef]
10. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
11. Li, C.; Sohn, K.; Yoon, J.; Pfister, T. CutPaste: Self-Supervised Learning for Anomaly Detection and Localization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 9659–9669.
12. Mundt, M.; Majumder, S.; Murali, S.; Panetsos, P.; Ramesh, V. Meta-learning convolutional neural architectures for multi-target concrete defect classification with the concrete defect bridge image dataset. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–17 June 2019; pp. 11196–11205.
13. Hüthwohl, P.; Lu, R.; Brilakis, I. Multi-classifier for reinforced concrete bridge defects. *Autom. Constr.* **2019**, *105*, 102824. [CrossRef]
14. Feroz, S.; Abu Dabous, S. UAV-Based Remote Sensing Applications for Bridge Condition Assessment. *Remote Sens.* **2021**, *13*, 1809. [CrossRef]
15. Cha, Y.J.; Choi, W.; Suh, G.; Mahmoudkhani, S.; Büyüköztürk, O. Autonomous Structural Visual Inspection Using Region-Based Deep Learning for Detecting Multiple Damage Types. *Comput.-Aided Civ. Infrastruct. Eng.* **2018**, *33*, 731–747. [CrossRef]
16. He, Y.; Jin, Z.; Zhang, J.; Teng, S.; Chen, G.; Sun, X.; Cui, F. Pavement Surface Defect Detection Using Mask Region-Based Convolutional Neural Networks and Transfer Learning. *Appl. Sci.* **2022**, *12*, 7364. [CrossRef]
17. Huang, B.; Zhaom, S.; Kang, F. Image-based automatic multiple-damage detection of concrete dams using region-based convolutional neural networks. *J. Civ. Struct. Health Monit.* **2022**, 1–17. [CrossRef]
18. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.
19. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2016; pp. 779–788.
20. Cui, X.; Wang, Q.; Dai, J.; Zhang, R.; Li, S. Intelligent recognition of erosion damage to concrete based on improved YOLO-v3. *Mater. Lett.* **2021**, *302*, 130363. [CrossRef]
21. Deng, J.; Lu, Y.; Lee, V.C.S. Imaging-based crack detection on concrete surfaces using You Only Look Once network. *Struct. Monit.* **2021**, *20*, 484–499. [CrossRef]
22. Jiang, Y.; Pang, D.; Li, C. A deep learning approach for fast detection and classification of concrete damage. *Autom. Constr.* **2021**, *128*, 103785. [CrossRef]
23. Jiang, W.; Liu, M.; Peng, Y.; Wu, L.; Wang, Y. HDCB-Net: A Neural Network With the Hybrid Dilated Convolution for Pixel-Level Crack Detection on Concrete Bridges. *IEEE Trans. Ind. Inform.* **2021**, *17*, 5485–5494. [CrossRef]
24. Bhattacharya, G.; Mandal, B.; Puhan, N.B. Interleaved Deep Artifacts-Aware Attention Mechanism for Concrete Structural Defect Classification. *IEEE Trans. Image Process.* **2021**, *30*, 6957–6969. [CrossRef]
25. Kang, J.; Tariq, S.; Oh, H.; Woo, S.S. A Survey of Deep Learning-Based Object Detection Methods and Datasets for Overhead Imagery. *IEEE Access* **2022**, *10*, 20118–20134. [CrossRef]
26. Larochelle, H.; Hinton, G.E. Learning to combine foveal glimpses with a third-order Boltzmann machine. *Adv. Neural Inf. Process. Syst.* **2010**, *23*, 1243–1251.
27. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual Attention Network for Image Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6450–6458.
28. Zhang, H.; Wu, C.; Zhang, Z.; Zhu, Y.; Lin, H.; Zhang, Z.; Sun, Y.; He, T.; Mueller, J.; Manmatha, R.; et al. ResNeSt: Split-Attention Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, New Orleans, LA, USA, 19–20 June 2022; pp. 2735–2745.
29. Park, J.; Woo, S.; Lee, J.-Y.; Kweon, S. A Simple and Light-Weight Attention Module for Convolutional Neural Networks. *Int. J. Comput. Vis.* **2020**, *128*, 783–798. [CrossRef]
30. Pan, Y.; Zhang, G.; Zhang, L. A spatial-channel hierarchical deep learning network for pixel-level automated crack detection. *Autom. Constr.* **2020**, *119*, 103357. [CrossRef]
31. Qiao, W.; Liu, Q.; Wu, X.; Ma, B.; Li, G. Automatic Pixel-Level Pavement Crack Recognition Using a Deep Feature Aggregation Segmentation Network with a scSE Attention Mechanism Module. *Sensors* **2021**, *21*, 2902. [CrossRef]
32. Wan, H.; Gao, L.; Su, M.; Sun, Q.; Huang, L. Attention-Based Convolutional Neural Network for Pavement Crack Detection. *Adv. Mater. Sci. Eng.* **2021**, *2021*, 5520515 . [CrossRef]
33. Xiang, X.; Zhang, Y.; El Saddik, A. Pavement crack detection network based on pyramid structure and attention mechanism. *IET Image Process.* **2020**, *14*, 1580–1586. [CrossRef]
34. Bhattacharya, G.; Mandal, B.; Puhan, N.B. Multi-deformation aware attention learning for concrete structural defect classification. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *31*, 3707–3713. [CrossRef]
35. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp 2980–2988.

36. Lin, T.-Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
37. Haciefendioglu, K.; Basaga, H.B. Concrete Road Crack Detection Using Deep Learning-Based Faster RCNN Method. *Iran. J. Sci. Technol.* **2022** , *46*, 1621–1633.
38. Yao, G.; Wei, F.; Yang, Y.; Sun, Y. Deep-Learning-Based Bughole Detection for Concrete Surface Image. *Adv. Civ. Eng.* **2019**, *2019*, 8582963. [CrossRef]
39. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
40. Wei, F.; Yao, G.; Yang, Y.; Sun, Y. Instance-level recognition and quantification for concrete surface bughole based on deep learning. *Autom. Constr.* **2019**, *107*, 102920. [CrossRef]
41. Kang, D.; Benipal, S.S.; Gopal, D.L.; Cha, Y.J. Hybrid pixel-level concrete crack segmentation and quantification across complex backgrounds using deep learning. *Autom. Constr.* **2020**, *118*, 103291. [CrossRef]
42. Mishra, M.; Jain, V.; Singh, S.K.; Maity, D. Two-stage method based on the you only look once framework and image segmentation for crack detection in concrete structures. *Archit. Struct. Constr.* **2022**, 1–18
43. Xu, Y.; Wei, S.; Bao, Y.; Li, H. Automatic seismic damage identification of reinforced concrete columns from images by a region-based deep convolutional neural network. *Struct. Control Health Monit.* **2019**, *26*, e2313. [CrossRef]
44. Li, R.; Yuan, Y.; Zhang, W.; Yuan, Y. Unified Vision-Based Methodology for Simultaneous Concrete Defect Detection and Geolocalization. *Comput.-Aided Civ. Infrastruct. Eng.* **2018**, *33*, 527–544. [CrossRef]
45. Wan, H.; Gao, L.; Yuan, Z.; Qu, H.; Sun, Q.; Cheng, H.; Wang, R. A novel transformer model for surface damage detection and cognition of concrete bridges. *Expert Syst. Appl.* **2023**, *213*, 119019. [CrossRef]
46. Teng, S.; Liu, Z.; Chen, G.; Cheng, L. Concrete Crack Detection Based on Well-Known Feature Extractor Model and the YOLOV2 Network. *Appl. Sci.* **2021**, *11*, 813. [CrossRef]
47. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
48. Zhang, C.; Chang, C.C.; Jamshidi, M. Concrete bridge surface damage detection using a single-stage detector. *Comput.-Aided Civ. Infrastruct. Eng.* **2020**, *35*, 389–409. [CrossRef]
49. Wu, P.; Liu, A.; Fu, J.; Ye, X.; Zhao, Y. Autonomous surface crack identification of concrete structures based on an improved one-stage object detection algorithm. *Eng. Struct.* **2022**, *272*, 114962. [CrossRef]
50. Wang, W.; Su, C.; Fu, D. Automatic detection of defects in concrete structures based on deep learning. *Structures* **2022**, *43*, 192–199. [CrossRef]
51. Kumar, P.; Batchu, S.; Swamy S.N.; Kota, S.R. Real-Time Concrete Damage Detection Using Deep Learning for High Rise Structures. *IEEE Access* **2021**, *9*, 112312–112331. [CrossRef]
52. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 618–626.
53. Jeong, E.; Seo, J.; Wacker, J. Literature Review and Technical Survey on Bridge Inspection Using Unmanned Aerial Vehicles. *J. Perform. Constr. Facil.* **2020**, *34*, 04020113. [CrossRef]
54. Cheng, M.M.; Mitra, N.J.; Huang, X.; Torr, P.; Hu, S.M. Global contrast based salient region detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *37*, 569–582. [CrossRef] [PubMed]
55. Filali, I.; Allili, M.S.; Benblidia, N. Multi-scale salient object detection using graph ranking and global–local saliency refinement. *Signal Process. Image Commun.* **2016**, *47*, 380–401. [CrossRef]
56. Hou, Q.; Cheng, M.-M.; Hu, X.; Borji, A.; Tu, Z.; Torr, P. Deeply Supervised Salient Object Detection with Short Connections. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3203–3212.
57. Tan, Z.; Nie, X.; Qian, Q.; Li, N.; Li, H. Learning to Rank Proposals for Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 261–318.
58. Liu, L.; Ouyang, W.; Wang, X.; Fieguth, P.; Chen, J.; Liu, X.; Pietikäinen, M. Deep Learning for Generic Object Detection: A Survey. *Int. J. Comput. Vis.* **2020**, *128*, 261–318. [CrossRef]
59. Girshick, R. Fast RCNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
60. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
61. Zoph, B.; Cubuk, E.D.; Ghiasi, G.; Lin, T.Y.; Shlens, J.; Le, Q.V. Learning Data Augmentation Strategies for Object Detection. In Proceedings of the 16th European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 566–583.
62. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
63. Wang, C.Y.; Yeh, I.H.; Liao, H.Y.M. You only learn one representation: Unified network for multiple tasks. *arXiv* **2021**, arXiv:2105.04206.
64. Jocher, G. YOLOv8. 2023. Available online: https://github.com/ultralytics/ultralytics (accessed on 14 February 2023).

65. Patel, R.A.; Steinmann, L.; Fehrenbach, J.; Fehrenbach, D.; Dehn, F. Convolution Neural Network-Based Machine Learning Approach for Visual Inspection of Concrete Structures. In Proceedings of the 1st Conference of the European Association on Quality Control of Bridges and Structures: EUROSTRUCT, Padova, Italy, 29 August–1 September 2021 ; pp. 704–712.
66. Xiong, R.; Liu, P.; Tang, P. Human Reliability Analysis and Prediction for Visual Inspection in Bridge Maintenance. In Proceedings of the ASCE International Conference on Computing in Civil Engineering 2021, Orlando, FL, USA, 12–14 September 2021; pp. 254–262.
67. Jocher, G. YOLOv5. 2020. Available online: https://github.com/ultralytics/yolov5 (accessed on 14 February 2023).