Article

Not peer-reviewed version

Co-CrackSegment: A New Corporative Deep Learning Framework for Pixel-Level Semantic Segmentation of Concrete Cracks

Nizar Faisal Alkayem * , Ali Mayya , Xin Zhang , Lei Shen , Panagiotis G. Asteris , Qiang Wang , Maosen Cao *

Posted Date: 27 August 2024

doi: 10.20944/preprints202408.1801.v1

Keywords: semantic segmentation; crack identification; ensemble learning; deep learning; Co-CrackSegment

Article

Co-CrackSegment: A New Corporative Deep Learning Framework for Pixel-Level Semantic Segmentation of Concrete Cracks

Nizar Faisal Alkayem [1,2,*], Ali Mayya [3], Lei Shen [4], Xin Zhang [1], Panagiotis G. Asteris [5]， Qiang Wang [1] and Maosen Cao [6,*]

[1] College of Automation and College of Artificial Intelligence, Nanjing University of Posts and Telecommunications, Nanjing 210046, China
[2] College of Civil and Transportation Engineering, Hohai University, 210098, Nanjing, China
[3] Computer and Automatic Control Engineering Department, Faculty of Mechanical and Electrical Engineering, Tishreen University, Lattakia 2230, Syria
[4] College of Water Conservancy and Hydropower Engineering, Hohai University, Nanjing, 210098, China
[5] Computational Mechanics Laboratory, School of Pedagogical and Technological Education, 15122 Athens, Greece
[6] College of Mechanics and Engineering Science, Hohai University, 211100, Nanjing, China
* Correspondence: nizar.alkayem@njupt.edu.cn (N.F.A.); cmszhy@hhu.edu.cn (M.C.)

Abstract: In the era of massive construction, damaged and aging infrastructure are becoming more common. Defects, such as cracking, spalling, etc., are main types of structural damage that widely occur. Hence, ensuring the safe operation existing infrastructure through health monitoring has emerged as an important challenge facing structural engineers. In recent years, intelligent approaches, such as data driven machine and deep learning crack detection, gradually dominate over traditional methods. Among them, the semantic segmentation using deep learning models is a process of characterization of accurate location and portrait of cracks using pixel level classification. Most available studies rely on single model knowledge to perform this task. However, it is well-known that the single model might suffer from low variance and low ability to generalize in case of data alteration. By leveraging the ensemble deep learning philosophy, a novel corporative semantic segmentation of concrete cracks method called Co-CrackSegment is proposed. Firstly, five models, namely the U-net, SegNet, DeepCrack19, DeepLabV3-ResNet50, and DeepLabV3-ResNet101 are trained to serve as core models for the ensemble model Co-CrackSegment. To build the ensemble model Co-CrackSegment, a new iterative approach based on the best evaluation metrics, namely the dice score, IoU, pixel accuracy, precision, and recall metrics is developed. Results show that the Co-CrackSegment exhibits a prominent performance compared to core models and weighted average ensemble by means of the considered best statistical metrics.

Keywords: semantic segmentation; crack identification; ensemble learning; deep learning; Co-CrackSegment

MSC: 68U10

1. Introduction

Structural health monitoring (SHM) and damage identification play a crucial role in ensuring the safe operation and structural integrity of in-service infrastructure by the promptly diagnosis of structural condition [1,2]. SHM adheres to guarantee the continuous service of structures bearing internal and external loads and hazardous conditions [3,4]. These unwanted conditions can deteriorate structural elements and gradually lead to structural defects. Therefore, SHM serves as an essential management and maintenance framework for ensuring reliable infrastructure performance all over the expected their lifespan when subject to catastrophic events. Even if regular manual inspections might deliver good information about structural conditions, they are often time-consuming, rely on human evaluations, and are susceptible to human errors [5,6]. In consequence, the utilization of intelligent soft computing approaches has emerged as more reliable and convenient ways against the traditional methods. Among them, the convolutional neural networks (CNNs) are being implemented for the data-driven defect identification either by using 1D data as time-domain responses, or using 2D data as originally captured images or time-frequence plots. For instance, image datasets gathered from a structure's surface can be adapted to develop CNNs for image-based defect identification. A well-designed CNN can provide an effective tool for crack identification, oiling the wheels of early damage detection and eliminating the risks of catastrophic events. This image-based CNN tools delivers smooth monitoring by taking advantage of machine intelligence to verifiably deduce features from training instances in a ductile manner more efficient than manual condition assessments [7,8].

Most common supervised image-based crack identification using CNNs can be categorized into three main categories: (i) CNN-based classification methods directly try to recognize whether an image taken from structure surface contains cracks by means of binary crack classification, or more complexly multi-class crack classification [9,10]. The total image is given a classification label and the CNN learns to classify the images according to their labels. After training, the CNN classifier can be deployed for real-time crack identification without computationally expensive pixel level tackling. Nevertheless, such category of methods fails to characterize and localize cracks in most scenarios without proper additional image processing techniques [11,12]; (ii) Region-based methods are types of CNN-based tools that work on partial regions of the image and consider the cracks as objects needs to be detected. Methods such as sliding windows and You-Only-Look-Once (YOLO) [13] are common object detection tools that can be used to localize cracks by means of boundary boxes rather than tackling the full-scale images or pixel-level features. These tools require the use of manually annotated ground truth bounding boxes that deliver supervision for the CNN or basic references for optimizing object possibilities. The output after training is given as the boundary boxes that identify the cracks from the backgrounds with confidence probabilities. Although the object detection tools are faster to train and deploy, they suffer from drawbacks such as the misidentification of crack boundaries and background voids [14–16]; (iii) Semantic segmentation based on CNNs are modern pixel-level tools adhere to precisely characterize cracks pixels and isolate them from the background pixels. Semantic segmentation uses two main procedures, namely the down-sampling and up-sampling. The former reduces the spatial dimensions of feature maps and increases the number of filters/ channels, while the later increases the spatial dimensions of feature maps and reduces the number of filters/channels. In this case, the CNN assigns a probability of each pixel in the image according to its label, i.e. crack or non-crack, enabling to generate binary-class attribute maps. The main advantages of semantic segmentation are the ability to characterize crack morphology and give more visual information about crack size and orientation. However, they require the manual preparation of ground truth images to train the CNN and can be computationally expensive in both training and testing. Given the advantages of the state-of-the-art sematic segmentation of concrete cracks, this paper aims to provide effective CNN-based semantic segmentation methods able to overcome the current challenges in this field [15,17–19].

In recent years, CNN models such as Unet, SegNet, DeepLab, etc. have been successfully used for crack semantic segmentation. Although the single CNN model-based semantic segmentation has achieved major milestones in pixel-level crack identification, ensemble learning philosophy deliver renowned advantages by exploiting collective knowledge and diversity among ensemble core models. By training several CNN models either from the same type or different types and combine their predictions serves to improve the overall crack characterization and deliver more precise pixel crack/background probabilities. Therefore, ensemble learning can be more efficient than the single-model-based methods for concrete crack identification. Although traditional ensemble CNN models that involve the weighted average, bagging, stacking, and boosting have been used for image-based crack classification, they have been rarely applied in semantic segmentation of cracks. This is mainly because the pixel data are highly imbalanced and mostly belong to the background rather than the crack. Moreover, the pixel data is of high dimensions which make it difficult for metal earners to train and combine predictions and require high computational efforts, especially in cases of stacking and boosting. Moreover, the averaging in weighted average and bagging methods might blur the pixels in crack boundaries hiding their actual label probabilities. Therefore, it is of great importance to further adapt the ensemble learning philosophy for the purpose of crack semantic segmentation and propose more efficient methods similar to the current paper.

2. Literature Review, Research Gaps, and Contributions

Several research works investigated the application of CNNs for concrete crack semantic segmentation with main focus on single model-based approaches. For example, Arafin et al [20], developed a multi-stage strategy for classification and semantic segmentation of concrete defects with promising results. First, the classification of cracks and spalling defects was done using three CNNs, namely the InceptionV3, ResNet50, and VGG19 with reported 91% accuracy for InceptionV3. Also, the semantic segmentation was employed based on the Unet and PSPnet to identify defects' areas with average evaluation metrics score over 90%. In another work, Hang et al. [21] developed the AFFNet that used the ResNet101 as backbone and dual attention mechanisms for the semantic segmentation of concrete crack with higher mean intersection over union (IoU) metrics over 84%. Tabernik et al. [22] developed the SegDecNet++ for semantic segmentation of concrete and pavements cracks and enhanced classification-based segmentation reporting the dice score of 81%. Shang et al. [23] proposed a fusion-based Unet for the pixel-level identification of sealed cracks with an IoU over 84%. In another research [24], the multi-resolution feature extraction network (MSMR) was developed for the semantic segmentation of concrete cracks with a reported IoU over 82%. Minh Dang et al. [25] developed a semantic segmentation of sewer defects method by utilizing the DeepLabV3+ with various backbone networks and reported an accuracy of 97% and IoU of 68%. Another semantic segmentation model was developed by Joshi et al. [26], in which three sub-modules were incorporated and the transfer learning was utilized to improve the overall segmentation results. In addition, a multi-stage YOLO-based object detection and Ostu thresholding for crack quantification purpose was proposed by Mishra et al [27]. Another research was conducted by Shi et al. [28] in which they proposed what was called the multilevel contrastive learning CNN for crack segmentation. The developed approach incorporated a dual-training approach by using full image and image patches with prespecified sizes and the contrastive learning was then used to provide the final decision about the pixel labels. The overall reported IoU for different scenarios did not exceed the 70% for all tested datasets. Another research was done by Savino and Tondolo [29] in which the Deeplabv3+ networks were developed using weights initialization using transfer learning of different other networks with highest reported accuracy of over 91%. Hadinata et al. [30] developed a multi-class segmentation approach for three classes of crack, spalling and voids using the Unet and DeepLabV3+ with a mean reported IoU of around 60% using the Unet. Another approach for crack semantic segmentation using a hybrid deep learning approach based on class activation maps and encoder-decoder network was proposed by Al-Huda et al. [31]. By incorporating image processing methods and transfer learning, the proposed approach was able to provide a mean IoU of around 90%. In addition, Ali et al. [32] utilized the local pixel weighing approach with residual blocks for improving a CNN with encoder-decoder section with average accuracies over 98% for different scenarios. Kang et al. [33] utilized the faster RCNN to allocate crack boundaries and a modified tubularity flow field for segmentation with a mean average IoU of 83% was reported. Also, the crack semantic segmentation of nuclear containments was conducted using an improved Unet using the multi-feature fusion and focal loss. Compared to other approaches, the proposed approach achieved a better IoU value over 73%. From the studied literature, it can be seen that the developing and deploying of single model and its improved features or hybrid versions for the task of crack semantic segmentation is the common research trend worldwide. However, single models are often susceptible to low generalization abilities and might not recognize all underlying crack patterns. In addition, the high bias of the considered datasets might contribute to decline in performance of single-model-based approaches.

In machine learning applications, ensemble predictions often contribute to improve the individual model predictions especially when performance of individual models drops with data alterations [34–36]. In recent years, several attempts were devoted to implement ensemble learning for semantic segmentation applications [37–39]. Difficulties of high computational cost to train individual models to deal with pixel-level data make ensemble learning less favorable in case of semantic segmentation. Besides the semantic segmentation of cracks, several successful attempts were reported in the literature. For example, Bousselham et al. [40] developed an ensemble models based on single meta-learner by leveraging the multi- feature pyramid network for semantic segmentation which was tested using general benchmarking datasets. Nigam et al. [41] developed an ensemble deep learning semantic segmentation model by extracting knowledge by training individual models on separate data sources and fine-tuning after transfer learning to the intended domain with main dataset was drone-collected scenes image data. Also, three DeeplabV3 models trained using the firefly algorithms were ensembled by Zhang et al. [42] by applying model averaging for semantic segmentation of several benchmark datasets. In another research, Lee et al. developed an ensemble learning model by the progressive weighted of several core models and their backbones for segmentation of skin Lesion. In another work [43], three Unet models with different backbones are ensembled using the model averaging and further tuned by evolutionary algorithm for the retinal vessel segmentation. For crack semantic segmentation purposes, few research works were concerned with the use of ensemble learning. However, some few papers applied the ensemble learning for crack semantic segmentation such as, the work of Lee et al. [44] developed a meta-model architecture to synthesize an ensemble prediction of four models, namely the DeeplabV3, Unet, DeepLabV3+, and DANet with better results reported for the case of the meta-learner ensemble. Li and Zhao [45] attempted to ensemble five models, namely the PSPNet, Unet, DeepLabv3+, Segnet, PSPNet, and FCN-8s by using four softmax regression-based models. Amieghemen and Sherif [46] employed the weighted ensemble of four models, namely three Unets with different backbones and the PaveNet for the semantic segmentation of aerial images including pavement cracks. In another work, the fuzzy integral was used to ensemble three Linknet models with three different backbone architectures for the purpose of pavement crack segmentation [47]. Similar other research works can be found in [48–53]. A recent review article has indicated that the use of ensemble learning for semantic segmentation of concrete crack is less popular to minimize the overfitting and low variance of the deep neural networks models [54]. From the above literature, it is evident that research on crack semantic segmentation using ensemble learning is still premature and need further improvements.

According to the aforementioned literature survey, the main motivations and contributions of the current paper can be given as:

- Most available studies relied on individual model prediction to perform the crack semantic segmentation. Nevertheless, it is well-known that the individual model might suffer from low variance and low generalization ability in case of data alteration.
- To overcome the overfitting of crack image data, many studies focus on various hybridizations or modifications of existing models as well as transfer learning which still do not incorporate the knowledge of multiple learning to perform the concrete semantic segmentation task.
- Crack semantic segmentation underlies several problems, particularly when dealing with complex and highly contaminated image backgrounds, blurring, shadows, etc. Therefore, it is necessary to improve the existing identification method and include novel techniques.
- The ensemble learning is a very effective method to improve the performance of individual learners by combining their knowledge using some well-established methods, such as weighted averaging, stacking, bagging, and boosting.
- For pixel-level semantic segmentation especially in case of crack images, the abovementioned ensemble learning methods are less popular among researchers. This is mainly due to problems related to computational cost and difficulties in optimizing ensemble learning parameters.
- The traditional weighted average ensemble learning for pixel-level semantic segmentation might suffer from pixel blurring of crack boundaries resulting high bias of predicted crack map than the ground truth.

- It is well-known that Pixel-level semantic segmentation is of high spatial correlation features which do not highly suit the independent sampling of supervised learning. Moreover, as most pixels belong to background and to crack area, class imbalance is inevitable in pixel level crack detection. These two reasons make the use of traditional ensemble learning such as boosting and stacking difficult.
- Hence, it is of great significance to improve the existing ensemble learning methods for pixel-level semantic segmentation, especially when considering crack images that naturally include various background contaminations.
- By leveraging the ensemble deep learning philosophy, a novel corporative semantic segmentation of concrete cracks method called Co-CrackSegment is proposed.
- Five models, namely the U-net, SegNet, DeepCrack19, and DeepLabV3 with ResNet50, and ResNet101 backbones are trained to serve as core models for the Co-CrackSegment.
- To build the corporative model, a new iterative approach based best evaluation metrics, namely the dice score, IoU, pixel accuracy, precision, and recall metrics is developed.
- Finally, a detailed numerical and visual comparisons between the Co-CrackSegment and the core models as well as the weighted average ensemble learning model is presented.

The remainder of the paper is outlined as: (i) The proposed method of semantic segmentation of surface cracks is presented in section 3; (ii) The results and discussion of implementation of the proposed Co-CrackSegment with overall evaluation and comparison are illustrated in section 4; (iv) Finally, the conclusions of this work are presented in section 5.

3. Materials and Methods

In this section, a full description on the adopted datasets for semantic segmentation as well as the core deep learning models used in the Co-CrackSegment model is presented. Moreover, an overview on the proposed Co-CrackSegment model including the iterative optimal evaluation metric-based ensemble approach is given in details.

3.1. Crack Semantic Segmentation Framework.

The overall common crack semantic segmentation comprises six key stages that can be summarized as follows: (i) image data gathering; (ii) image pre-processing and ground truth image dataset construction; (iii) semantic segmentation model architecture and training algorithm determination; (vi) semantic segmentation model training and testing; (v) crack map identification; and (vi) results reporting. Raw crack image dataset is preliminary collected from a considered structure such as flying drones, camera holders, climbing robots, etc. After that, the dataset should undergo some data pre-processing procedures, in which the dataset is undergone image cropping, scaling, augmentation, labeling, normalization, etc. Thereafter, the ground truth images of the dataset are built which provide the main comparison tool inside the semantic segmentation models. Then, the dual dataset of pre-processed images and their ground truths are divided into training and testing subsets. Subsequently, the semantic segmentation model design and parameters as well as the training method are determined. Then, the semantic segmentation model is trained until approaching a good accuracy. After training the model, the model is evaluated and the crack maps are determined. Finally, the final spatial locations of cracks are reported. The overall semantic segmentation can be realized in Figure 1.

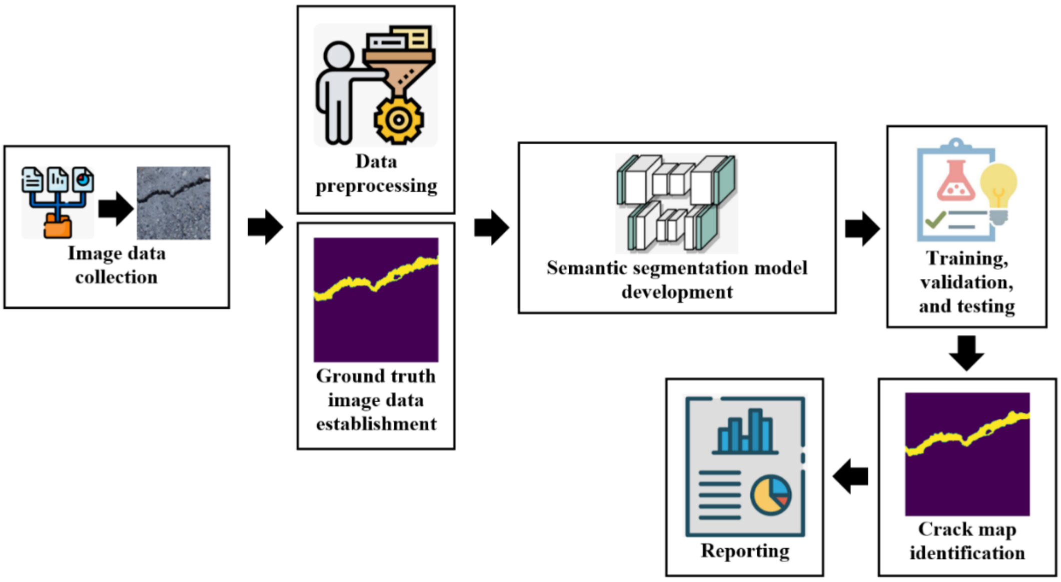


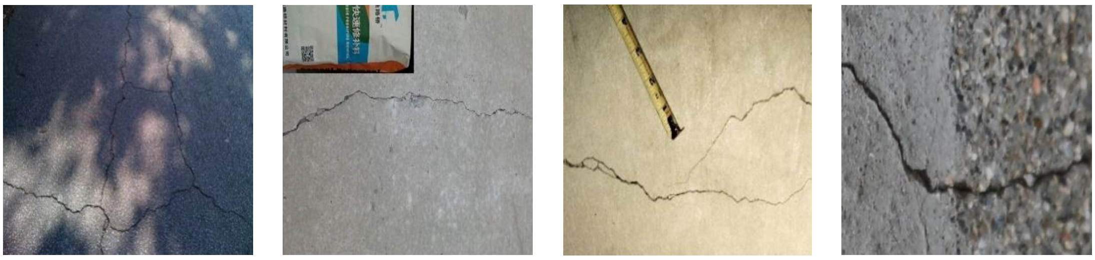
Image data collection
Data preprocessing
Ground truth image data establishment
Semantic segmentation model development
Training, validation, and testing
Crack map identification
Reporting

Figure 1. The general crack semantic segmentation framework.

3.2. Datasets

In this research, two public datasets are adopted. The first one is the famous DeepCrack dataset which was developed by Liu et al. [55]. The DeepCrack dataset is composed of 537 concrete and Asphalt images with their manually annotated ground truth images. This dataset is divided into 85% for training and 15% for testing. The dataset includes many challenging aspects such as cracks with shadows, cracks with foreign objects, spalling, complex background, cracks with rust and marks, etc. Some representative images and their ground truth masks are presented in Figure 2.

Another larger and well-known dataset for crack segmentation is the Rissbilder dataset [56,57], which contains 3249 training and 573 testing images divided into 85% for training and 15% for testing. The dataset includes wall images taken by a climbing robot. The data is very challenging containing shadows, illumination, foreign objects, crack-like scaling, crack-like background texture, thin cracks with dirty background, etc. The utilization of this dataset helps to provide complex instances to the developed semantic segmentation methods and verify their performances. Some image samples and their ground truth masks are presented in Figure 3.

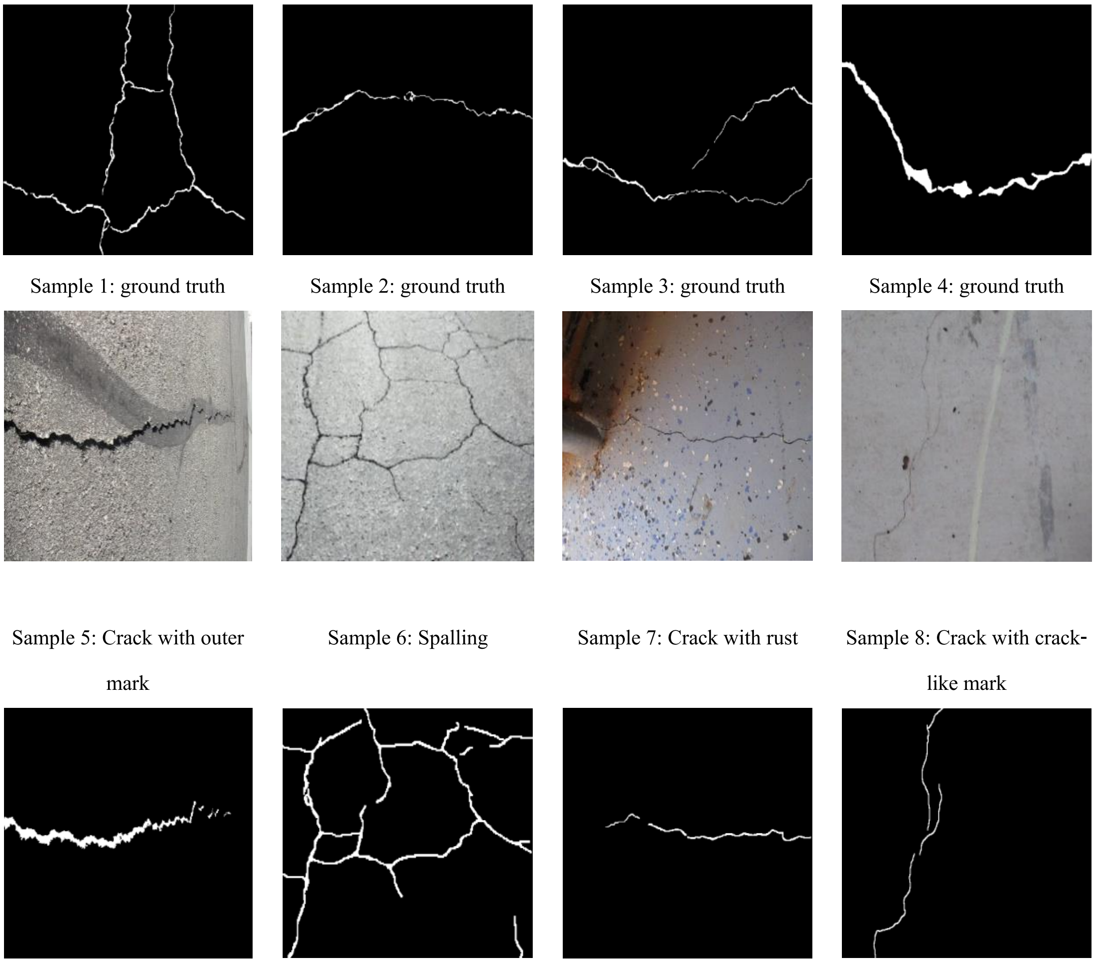


Sample 1: Crack with shadow

Sample 2: Single crack with a foreign object

Sample 3: Multiple cracks with a foreign object

Sample 4: Crack with complex background and crack-like scaling



Sample 1: ground truth

Sample 2: ground truth

Sample 3: ground truth

Sample 4: ground truth

Sample 5: Crack with outer mark

Sample 6: Spalling

Sample 7: Crack with rust

Sample 8: Crack with crack-like mark

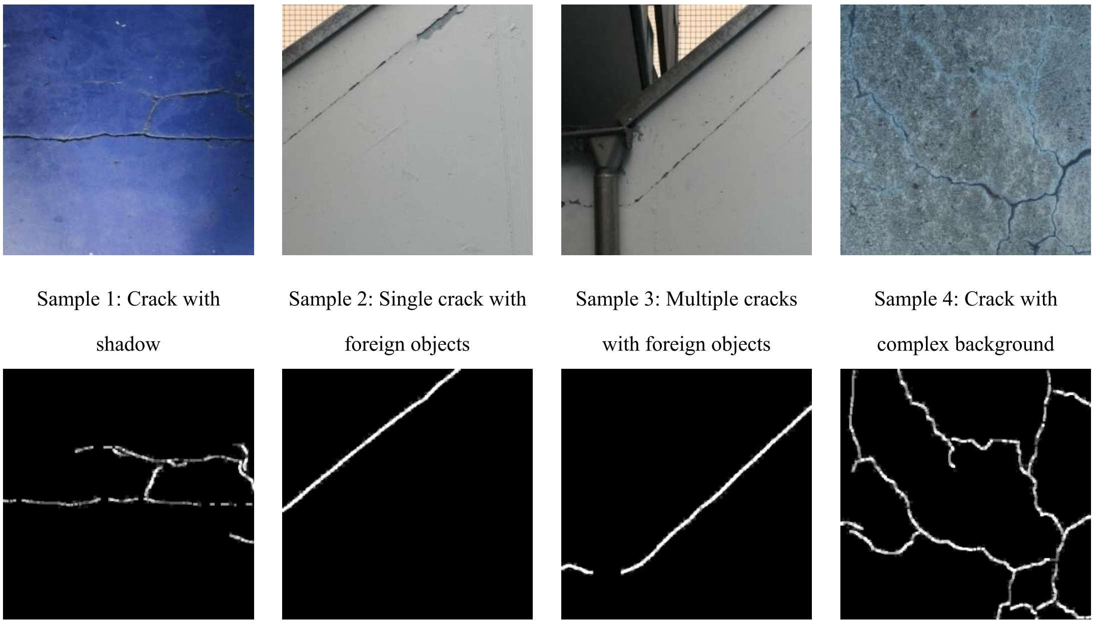
Sample 5: ground truth

Sample 6: ground truth

Sample 7: ground truth

Sample 8: ground truth

Figure 2. Sample images of the DeepCrack dataset [55].



Sample 1: Crack with shadow

Sample 2: Single crack with foreign objects

Sample 3: Multiple cracks with foreign objects

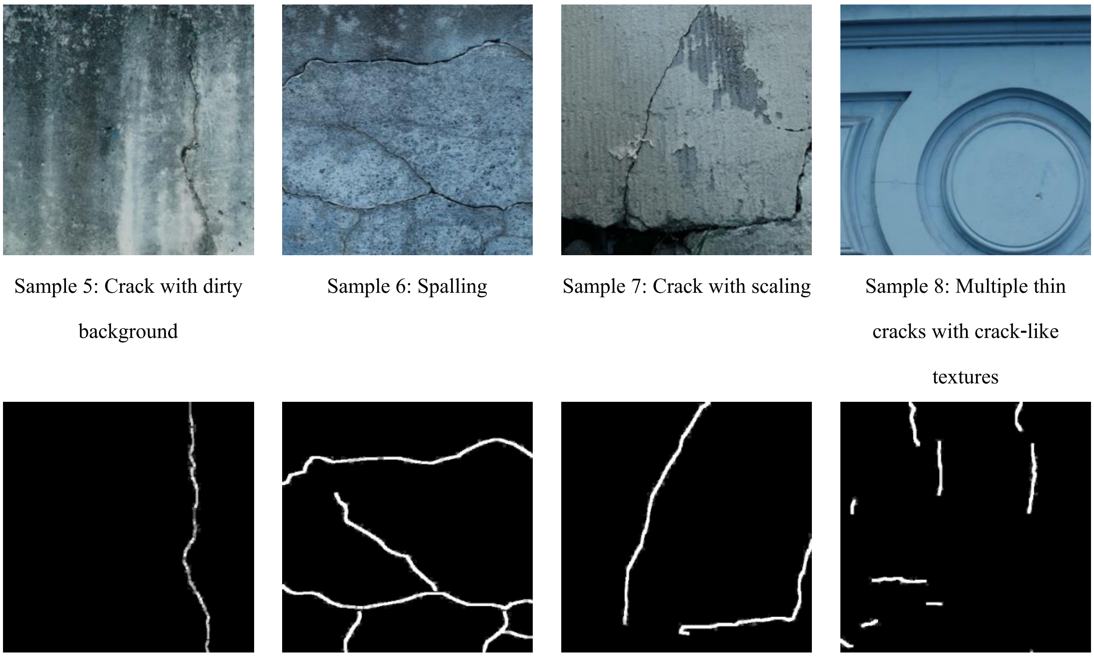
Sample 4: Crack with complex background

Sample 1: ground truth

Sample 2: ground truth

Sample 3: ground truth

Sample 4: ground truth



Sample 5: Crack with dirty background

Sample 6: Spalling

Sample 7: Crack with scaling

Sample 8: Multiple thin cracks with crack-like textures

Sample 5: ground truth

Sample 6: ground truth

Sample 7: ground truth

Sample 8: ground truth

Figure 3. Sample images of the Rissbilder dataset [56].

3.3. The Core Models

3.2.1. The U-net.

One of the most famous semantic segmentation deep CNN is the U-net which was originally proposed by Ronneberger [58] for medical imaging segmentation applications. However, the U-net has been utilized in various semantic segmentation projects afterwards. The U-net was named due to it architecture that takes the portrait of U-frame of encoder-decoder paths. The encoder path grasps the semantic features of the image by applying the basic CNN module, in which the image is subject to downsampling by employing Conv. And Pooling operations. Nevertheless, the encoder part endeavors to precisely recuperate spatial features by applying a set of upsampling and Conv. operations. The high order feature maps from the encoder are merged with the upsampled feature maps using skip connections, which permits to efficiently recover the low-level spatial features. The main merit of the U-net is the capability to recuperate and merge local and global features using the encoder-decoder pair and skip connections that help to deliver accurate pixel-level classification even with small datasets. The accurate identification of crack boundaries makes very effective for semantic segmentation applications, especially in case of crack semantic segmentation. The architecture of U-net can be observed in Figure 4.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 27 August 2024 doi:10.20944/preprints202408.1801.v1

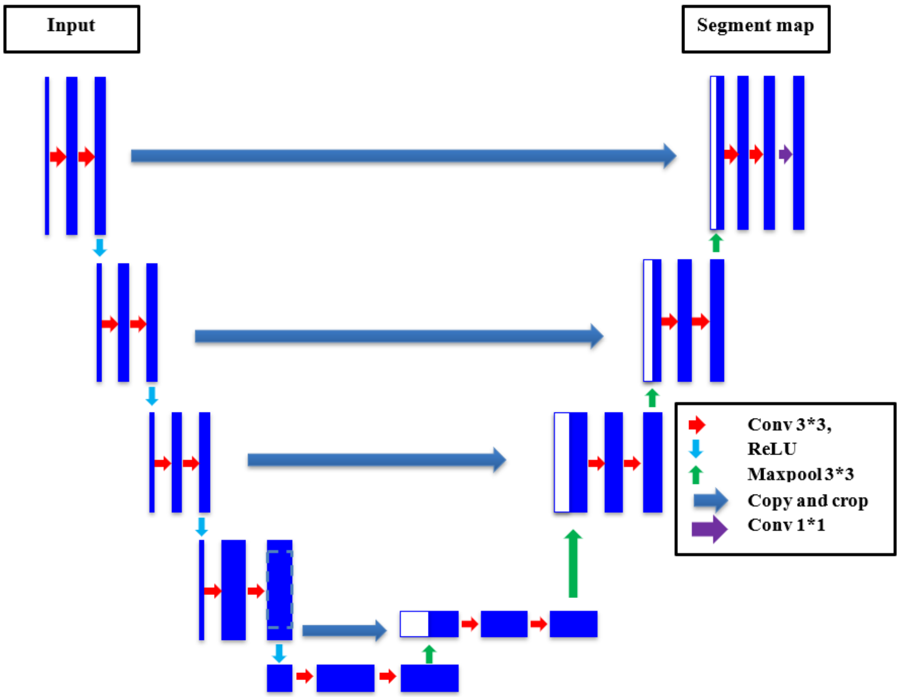


Figure 4. The U-net architecture.

3.2.2. The SegNet

Another well-known network architecture that is used for semantic segmentation tasks is the SegNet, which is a type of CNN with encoder-decoder pair model originally proposed for road scene segmentation. The main tool of the SegNet was the transfer learning of the VGG-16 architecture into the decoder module to recuperate pixel-level special and semantic features employing the Conv. and pooling operations of the VGG-16. The innovative point of the SegNet was the design of its decoder, in which the unpooling operation was proposed to unsample low-level features coming from the encoder. The indices of the maxpooled features taken from the encoder are utilized to upsample the higher features helping to maintain the boundaries and spatial features, which makes it very suitable for crack semantic segmentation applications. The main implemented operations in the SegNet are the full convolutions instead of fully-connected layers which enables it to process inputs of various dimensions, the skip connections that contribute to combine the high-level features from the encoder with the upsampled features of the decoder which boosts the semantic segmentation accuracy, and the use of transfer learning VGG-16 helps to keep the minimum number of training parameters and contributes to lower computational efforts. The overall merits of SegNet have made it an excellent choice for semantic segmentation of cracks, hence it is adopted in this work as a main core model. The design of SegNet can be realized in Figure 5.

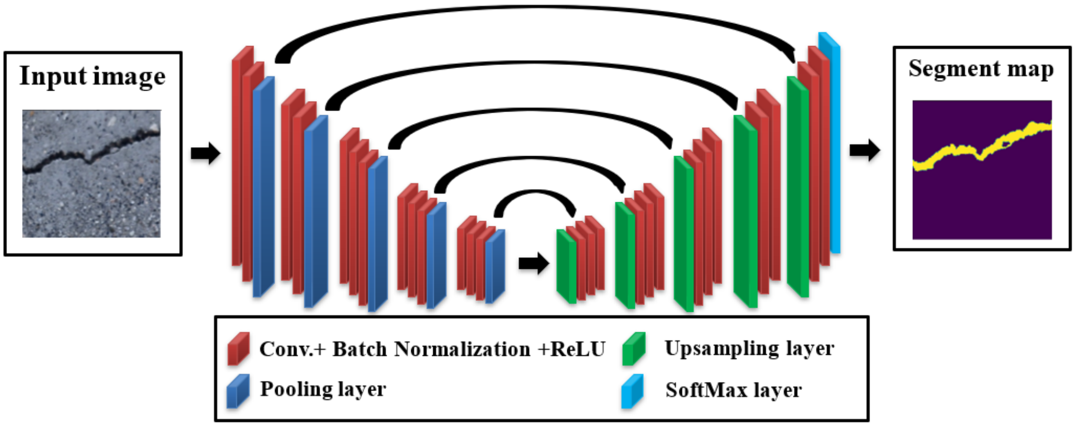


Figure 5. The SegNet architecture.

3.2.3. DeepCrack19

A well-established CNN model which was designed particularly for concrete crack identification and segmentation is the DeepCrack19 [55]. This model leverages the popular encoder-decoder pair structure with skip connections similar like other semantic segmentation models. The encoder part that does the downsampling is made of a VGG-19 [59] model previously trained using the ImageNet dataset. It is composed of a 19 Conv. layers with 5 maxpool layers. The decoder branch uses the same idea of upsampling and skip connections to merge the encoder resulted upsampled low-level feature maps and the unsampled high-level feature maps to provide a concise semantic segmentation of cracks. The training process of this network elaborates a double loss function, namely the cross entropy and the dice loss to optimize pixel level classification of minor cracks. It is well-known that predictions from lower order Conv. layers efficiently maintain crack boundaries but are susceptible to noise, however deeper layers are robust against noise but might not be able to keep concrete boundaries. DeepCrack19 proposed a compromise solution to solve this problem by introducing the guided filtering operation in which the model generates a binary crack mask from a fused prediction of various Conv. layers and then utilizing the output of Conv.1 & 2 layers as a guiding tool. Thereafter, the guided filtering is implemented to deliver the final classification. The overall DeepCrack19 model can be well-understood in Figure 6.

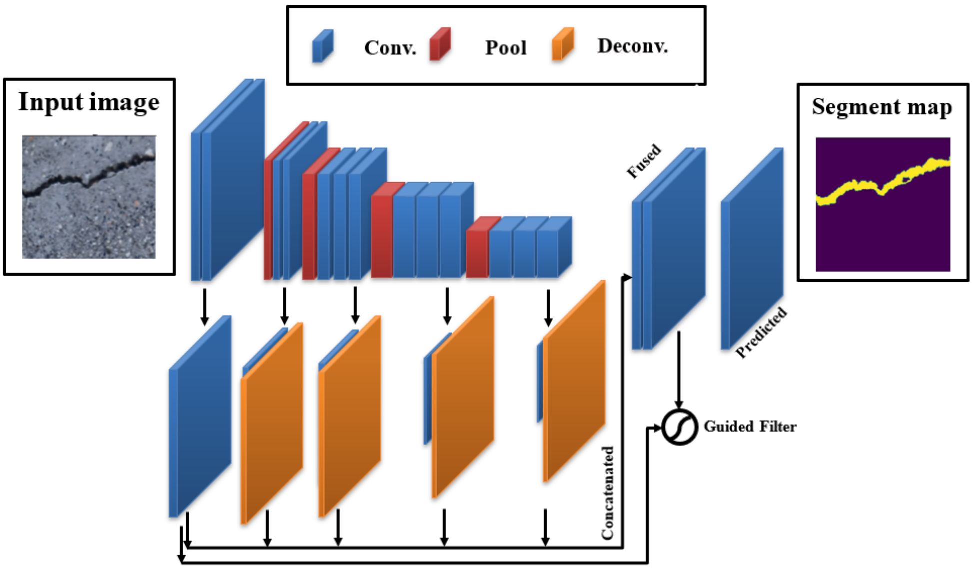


Figure 6. The DeepCrack19 architecture.

3.2.4. The DeepLabV3 with Backbones

Based on DeepLab model and developed by Google, DeepLabV3 is a relatively new deep learning model for semantic segmentation [60]. The DeepLabV3 employed several new techniques to improve prediction. The main novelty of this model was the replacement of convolution operation with the dilated or atrous convolution to recuperate the multiscale features without adding extra computational efforts. The atrous convolution implements dilation rates between the filter values, which efficiently increases the filter field of observation without altering its size. In addition, the DeepLabV3 uses what is called the atrous spatial pyramid pooling to perform a multiscale object segmentation which uses a parallel approach incorporating atrous operations with various dilation rates. In addition, the DeepLabV3 utilizes a simple type of decoder module to purify the semantic segmentation outcomes, which can be very useful in case of segmenting crack boundaries. Furthermore, the bilinear sampling and concentration is done by the decoder which merges both low- and high-level feature maps, which in turn helps to recover the accurate spatial information of crack boundaries. The DeepLabV3 has the feature of coupling various backbone architectures similar to U-net. In this work, two ResNet models which are the ResNet50 and ResNet101, are merged within the DeepLabV3 for the purpose of semantic segmentation of structural cracks. The use of ResNet50 backbone holds the advantage of the powerful residual and skip connections to overcome the gradient vanishing problem during the training process. In addition, the use of ResNet101 as a backbone helps to improve the accuracy because the ResNet101 is double in depth of ResNet50 and can be useful to perform better extractions of the complex crack pixel features. However, the use of DeepLabV3/ResNet101 is more computationally expensive comparing especially when using large datasets. The architecture of DeepLabV3 with backbones can be seen in Figure 7.

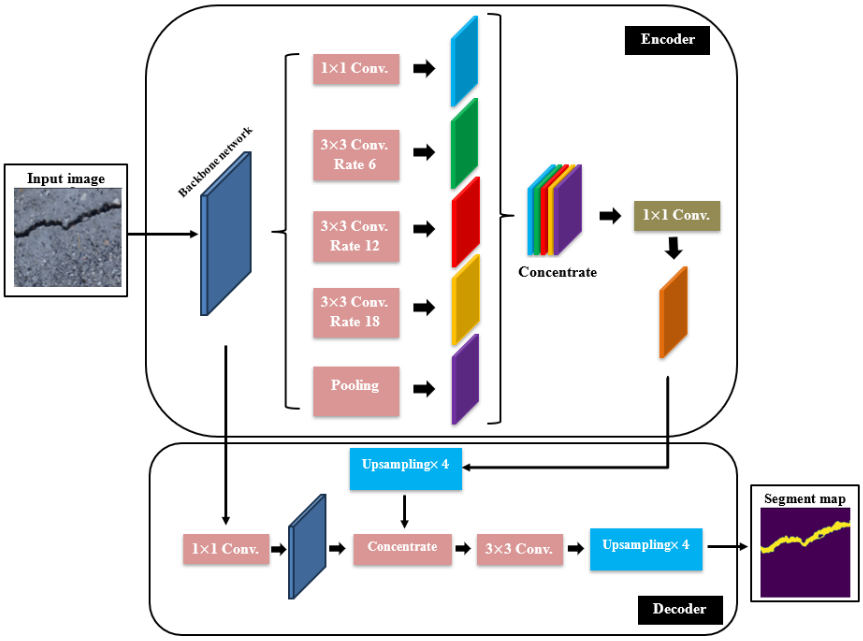


Figure 7. The DeepLabV3 with backbones architecture.

3.4. Training Procedure

To train the core models, the input crack images are resized to 448×448 pixels RBG images and normalized the pixel values between 0 and 1. Data augmentation is applied to improve the variability of the images by using the random rotation, horizontal and vertical flipping, normalization and random color jittering. The core models are evolved by considering a batch size of 8 and with 40 epochs. The dice loss loss function is implemented to measure the difference between the predicted and ground truth masks. Other related parameters can be observed in Tables 1 and 2 with respect to dataset1 and dataset2, respectively.

Table 1. The training parameters of dataset1.

Epochs	40
Loss function	Dice loss
batch_size	8
Initial Learning rate	1e-3
weight_decay	5e-5
Classification layer activation function	Sigmoid
Input image dimensions	448*448*3
Data augmentation operations	Normalization Random rotation Horizontal flip Vertical flip Random color jittering
Optimizer	Adam

Table 2. The training parameters of dataset2.

Epochs	40
Loss function	Dice loss
batch_size	8
Initial Learning rate	1e-3
weight_decay	5e-5
Classification layer activation function	Sigmoid
Input image dimensions	448*448*3
Data augmentation operations	Normalization Random rotation Horizontal flip Vertical flip Random color jittering
Optimizer	Adam

3.5. Evaluation Metrics

The main sets of evaluation metrics for semantic segmentation are categorized under overlapping metrics, in which the semantic segmentation model measures the overlap of pixels between the original image segmentation map and ground truth. To compute the semantic segmentation evaluation metrics, the confusion matrix of pixel-based segmentation mask is starting point. The confusion matrix is composed of true (TP) and false positives (FP) as well as false (FN) and true negatives (FN). The main overlap metrics are the Dice and intersection over union (IoU) metrics, in which a one prediction corresponds a full overlapping, whereas a zero-prediction associated with an absence of overlapping between the predicted mask and ground truth. The Dice score and IoU can be calculated as follows:

$$Dice(a,b) = \frac{2\|a \cap b\|}{\|a\| + \|b\|}, \tag{1}$$

and

$$IoU(a,b) = \frac{\|a \cap b\|}{\|a \cup b\|}. \tag{2}$$

Or in terms of confusion matrix, the Dice score and IoU can be calculated as follows:

$$Dice = \frac{2TP}{2TP + FP + FN}, \tag{3}$$

and

$$IoU = \frac{TP}{TP + FP + FN}. \tag{4}$$

Furthermore, the Rand score (pixel accuracy) is the number of correct pixel predictions (TP and TN) divided by the total number of pixel predictions as in the following equation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \quad (5)$$

In addition, the precision and recall are popular evaluation metrics in semantic segmentation. Precision measures how often predictions for the positive class are correct in the segmentation result, while Recall represents how well the semantic segmentation model detects all positive pixels in the segmentation result. The precision and recall can be calculated as follows:

$$Recall = \frac{TP}{TP + FN}, \quad (6)$$

and

$$Precision = \frac{TP}{TP + FP}. \quad (7)$$

Finally, the mAP which is the mean average precision or the average value of precision across all classes is utilized. The mAp can be given as follows:

$$mAP = \frac{1}{|classes|} \sum_{c=1}^{K} \frac{|TP_C|}{|TP_C| + |FP_C|}, \quad (8)$$

where k is the number of classes in the segmentation problem.

3.6. The Proposed Method

Crack semantic segmentation is often a challenging task, particularly when dealing with complex and contaminated image backgrounds. The available identification approaches require improvements and advanced techniques must be employed. Group learning or ensemble learning are common tools that improve single classifiers by combining their predictions. However, these group learning tools are less common to be applied for pixel-level semantic segmentation [54], especially for crack images due to computational costs and difficulties in optimizing ensemble learning parameters for pixel level evaluation. Therefore, it is of high significance to boost existing ensemble learning methods for pixel-level semantic segmentation for crack images. To address this issue, a novel cooperative crack semantic segmentation method, called Co-CrackSegment is proposed. This method takes advantage of ensemble deep learning philosophy by developing a new iterative approach based on the optimal evaluation metrics. Five Co-CrackSegment frameworks using the optimal dice score (Co-CrackSegment/dice), optimal IoU (Co-CrackSegment/IoU), optimal pixel accuracy (Co-CrackSegment/Pixel_Acc), optimal precision (Co-CrackSegment/Precision), and optimal recall (Co-CrackSegment/Recall) are developed and compared. To construct the group learner, five models, namely the U-net, SegNet, DeepCrack19, and DeepLabV3 with ResNet50, and ResNet101 backbones are trained to serve as core models for the Co-CrackSegment. The N trained core models are inserted in a model list and an external archive that stores the best model metrics. Each testing image is fed to the trained models and the evaluation metrics are computed subsequently including the current evaluation metrics. Thereafter, the external archive is altered and the model evaluation metrics are stored if a better evaluation metric score is achieved for each of the Co-CrackSegment frameworks. Finally, the overall iterative method is terminated after the best trained models' metrics are stored. The overall approach can be realized in Figure 8 as well as the following steps:

1. Load N trained semantic segmentation models in the model_list.
2. Choose one Co-CrackSegment framework, namely Co-CrackSegment/dice, Co-CrackSegment/IoU, Co-CrackSegment/Pixel_Acc, Co-CrackSegment/Precision, or Co-CrackSegment/Recall.
3. Set best_evaluation_metric_score to -1, and best_model_metrics to an empty matrix.
4. For each test image do the following:
    (a) For each current_model in the model_list (N times)

(b) Set the trainer.model to the current_model.
(c) Evaluate current_model with test image and compute the segmentation prediction output.
(d) Compute the overall evaluation metric scores including the current_evaluation_metric_score of the test image (current_model_metrics).
(e) If (current_evaluation_metric_score >best_ evaluation_metric_score)
i. best_ evaluation_metric_score = current_ evaluation_metric_score
ii. best_model_metrics= current_model_metrics
iii. Add trainer.model to the evaluation results matrix.

5. Show the results

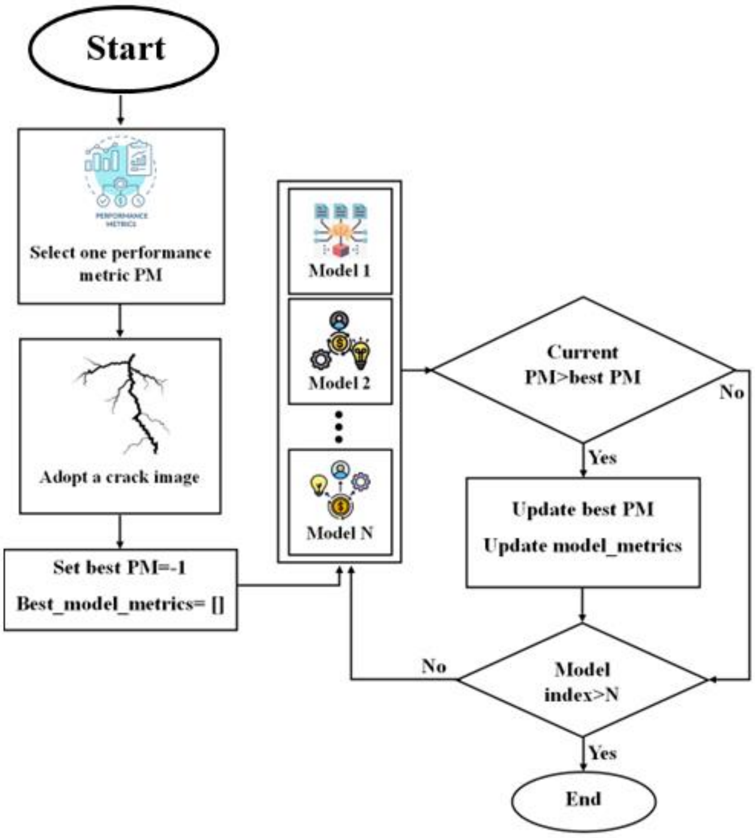


Figure 8. The developed corporative Co-CrackSegment semantic segmentation approach.

In addition to the aforementioned Co-CrackSegment framework the ensemble using the weighted average method employs the following steps as:

The ensemble method is based on the trained semantic segmentation models and consists of the following steps:

1- Load N trained semantic segmentation models in the model_list.
2- Set current_model to model1, and model_outputs to an empty matrix.
3- For each test image do the following:
For each current_model in the model_list
a. Compute the prediction of the current model.
b. Multiply predictions by the weight of the model
weighted_predictions = predictions * weights[j]
c. Add the weighted predictions to the list
model_outputs.append(weighted_predictions).
4- Perform weighted average sum:
ensemble_output = (sum(model_outputs) >= 0.5)

5- Compute metrics for the ensemble output
6- Show the results

Research manuscripts reporting large datasets that are deposited in a publicly available database should specify where the data have been deposited and provide the relevant accession numbers. If the accession numbers have not yet been obtained at the time of submission, please state that they will be provided during review. They must be provided prior to publication.

Interventionary studies involving animals or humans, and other studies that require ethical approval, must list the authority that provided approval and the corresponding ethical approval code.

4. Results and Discussion

This section presents the overall outcomes of the pixel level semantic segmentation of surface crack two paradigms, namely the results of the core models and the proposed Co-CrackSegment frameworks.

4.1. Performances of the Core Models

In this study, five independent core models namely the DeepLabV3 with ResNet50 (DLV3/ResNet50) and ResNet101 (DLV3/ResNet101), U-net, SegNet, and DeepCrack19 have been trained and tested using the two considered datasets for the purpose of pixel level semantic segmentation of surface cracks. The aim of training those aforementioned models is to develop strong core classifiers to be utilized inside the ensemble learning Co-CrackSegment frameworks. The five trained models are evaluated and by considering the parameter sets considered in Tables 1 and 2 corresponding to dataset1 and dataset2, respectively. The dice loss, percentage of pixel accuracy (%), IoU (%), precision (%), recall (%), mAP (%), and the iteration per second values are taken into account as main evaluation and comparison metrics. Results of training the core models can be observed in Tables 3 and 4 for dataset1 and dataset2, respectively. In addition, the statistical results are drawn in Figures 9 and 10 for dataset1 as well as Figures 11 and 12 for dataset2. By studying the tabulated results, it is clear that the U-net has achieved the best performance by means of loss, pixel accuracy, IoU, dice, and mAP in case of dataset1. It is also worth mentioning that the DLV3/ResNet50 and DLV3/ResNet101 have also achieved good performances when compromising the overall evaluation metrics. Furthermore, the iteration per second score of the DeepCrack19 makes it more competitive as a computationally efficient model. Moreover, in case of dataset2 and similar to dataset1, the U-net has also achieved the best performance by means of loss, accuracy, IoU, recall, dice, and mAP. Also, the DeepCrack19 has shown better computational performance than the other models when considering the number of iterations per seconds.

Table 3. Segmentation metrics of the trained individual models using dataset1.

Model	Loss	Pixel ACC%	IoU%	Precision%	Recall%	Dice%	mAP%	It/Sec
DLV3/ ResNet50	0.201	98.72	68.41	73.85	**90.92**	80	84.87	1.09
Unet	**0.178**	98.89	**71.15**	81.1	85.58	**82.2**	**84.9**	1.01
DeepCrack19	1.06	98.88	70.1	**81.31**	83.79	81.6	84.79	**1.31**
SegNet	0.191	98.78	69.4	79.23	85.48	81	84.522	1.17
DLV3/ ResNet101	0.185	**98.9**	70.2	80.38	84.6	81.7	84.69	1.11

Table 4. Segmentation metrics of the trained individual models using dataset2.

Model	Loss	Pixel ACC%	IoU%	Precision%	Recall%	Dice%	mAP%	It/Sec
DLV3/ResNet50	0.347	98.39	49.79	64.12	69.43	65.6	68.07	2.44
Unet	**0.33**	**98.5**	**51.2**	65.9	**70.46**	**67.04**	**69.36**	2.37

DeepCrack19	1.8	98.4	**51.2**	**66.24**	69.21	67.1	68.1	**3.06**
SegNet	0.339	98.4	50.22	65.1	69.93	66.1	68.35	2.66
DLV3/ResNet101	0.346	98.4	49.82	64.16	69.38	65.7	68.0	1.56

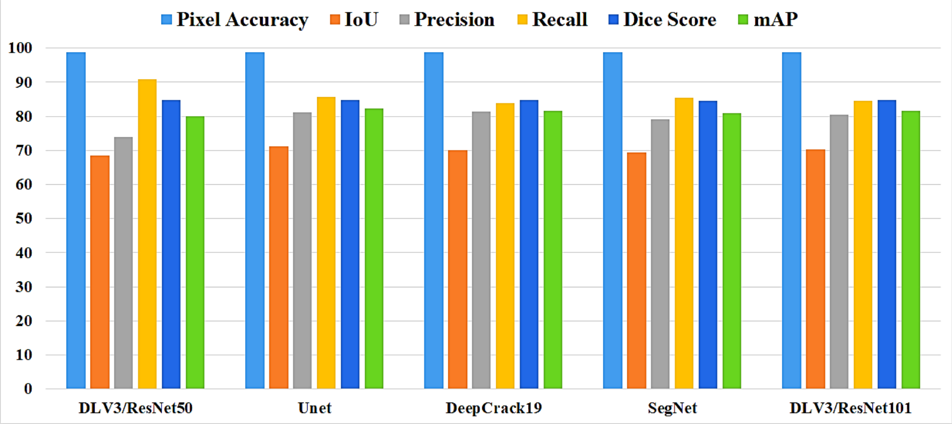


Figure 9. The evaluation metrics of the core models for dataset1.

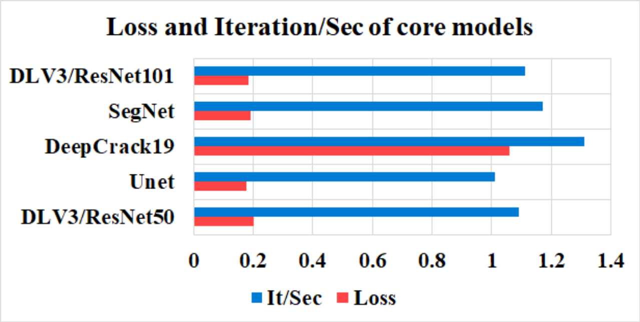


Figure 10. The losses and Iterations/Sec of the core models dataset1.

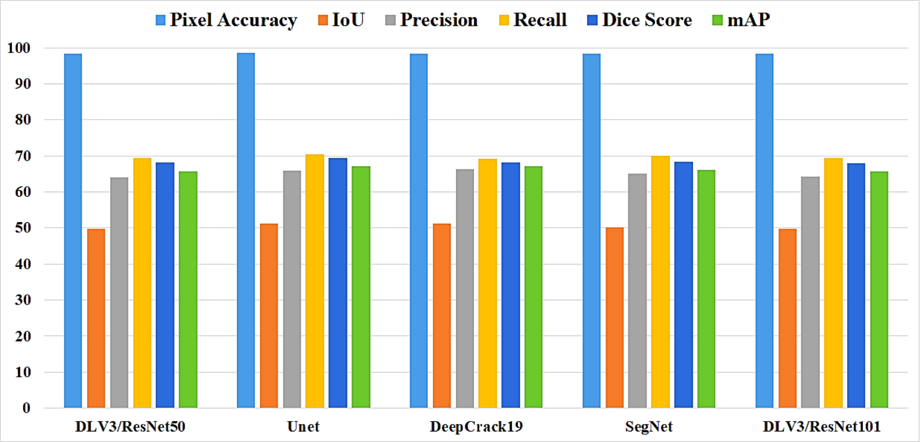


Figure 11. The evaluation metrics of the core models for dataset2.

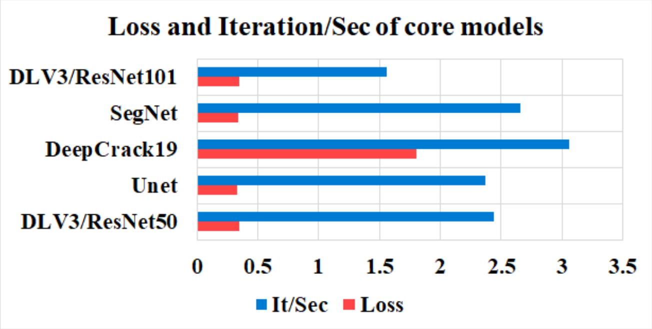


Figure 12. The losses and Iterations/Sec of the core models dataset2.

4.2. Performances of Co-CrackSegment

As it has been mentioned in section 3.5, the Co-CrackSegment takes advantage of group learning by developing an iterative approach based on the optimal evaluation metrics. The five trained deep learning semantic segmentation models are used core models inside the Co-CrackSegment. Thereafter, the Co-CrackSegment is executed by considering five frameworks using the optimal dice score (Co-CrackSegment/dice), optimal IoU (Co-CrackSegment/IoU), optimal pixel accuracy (Co-CrackSegment/Pixel_Acc), optimal precision (Co-CrackSegment/Precision), and optimal recall (Co-CrackSegment/Recall). The evaluation results of the Co-CrackSegment paradigms are presented in two styles as in Tables 5 and 6 as well as Figures 13 and 14 for dataset1 and dataset2, respectively. By studying the results, the Co-CrackSegment/dice and Co-CrackSegment/IoU have shown the best trade off scores comparing to other Co-CrackSegment frameworks. In addition, when comparing to weighted average method, most Co-CrackSegment frameworks outperformed the weighted average ensemble by means of all evaluation metrics. This is because the traditional weighted average ensemble learning for pixel-level semantic segmentation suffer from pixel blurring of crack boundaries due to average predictions resulting in high bias of predicted crack map than the ground truth. Furthermore, when comparing the results of core models with the Co-CrackSegment frameworks, it is clear that the group learning approach has boosted the performance of the individual models by means of all evaluation metrics. This proofs the efficiency of Co-CrackSegment approach for pixel level semantic segmentation of surface cracks.

Table 5. Segmentation metrics of the ensemble models using dataset1.

Model	Pixel_ACC	IoU	Precision	Recall	Dice	mAP
Co-CrackSegment/dice	99.03	**72.98**	82.22	86	**83.62**	**85.8**
Co-CrackSegment/IoU	99.038	**72.98**	82.24	86	**83.62**	**85.8**
Co-CrackSegment/Pixel_Acc	**99.042**	72.88	82.61	85.3	83.52	85.57
Co-CrackSegment/Precision	98.96	71.67	**83.29**	83.22	82.74	85.1
Co-CrackSegment/Recall	98.96	71.85	79.84	**87.31**	82.75	85.12
Weighted average	98.91	70.56	80.61	85.38	81.91	83.24

Table 6. Segmentation metrics of the ensemble models using dataset2.

Model	Pixel ACC	IoU	Precision	Recall	Dice	mAP
Co-CrackSegment/dice	98.52	**53.28**	67.37	71.925	**68.9**	70.31
Co-CrackSegment/IoU	98.52	**53.28**	67.41	71.88	**68.9**	70.31
Co-CrackSegment/Pixel_Acc	**98.536**	52.97	68.21	70.43	68.6	**70.5**
Co-CrackSegment/Precision	98.527	52.31	68.46	69.08	68.03	69.45
Co-CrackSegment/Recall	97.96	52.14	**64.72**	**73.33**	67.8	69.4
Weighted average	98.48	51.84	67.59	68.89	67.61	68.62

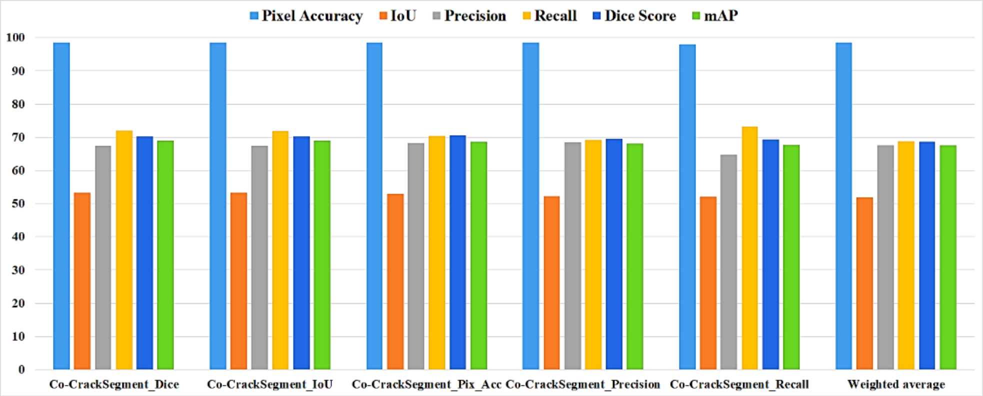


Figure 13. The evaluation metrics of the Co-CrackSegment frameworks for dataset1.

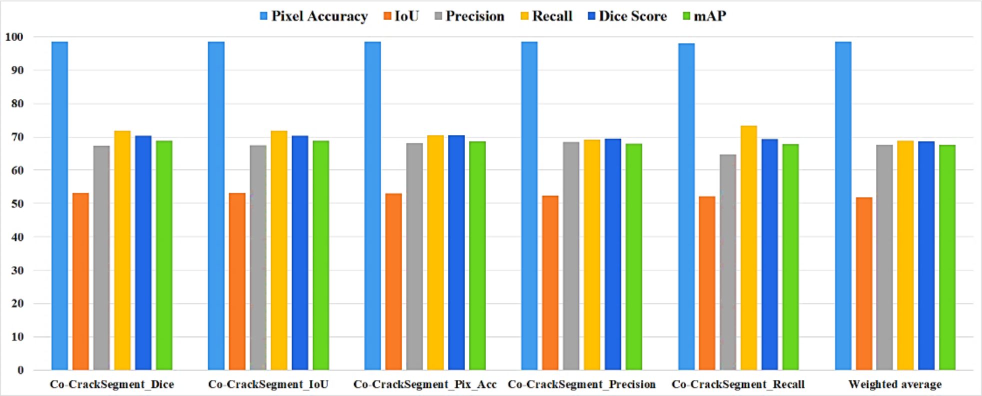


Figure 14. The evaluation metrics of the Co-CrackSegment frameworks for dataset2.

4.3. Visual Comparison and Discussion

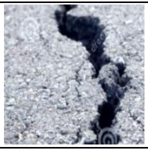
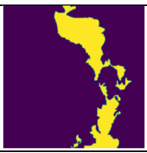
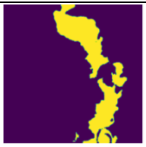
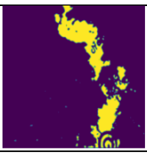
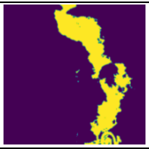
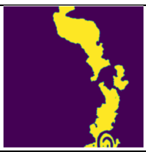
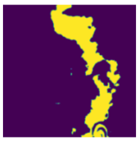
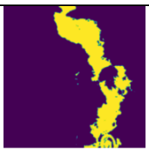
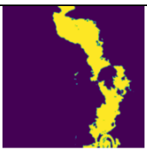
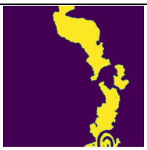
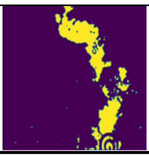
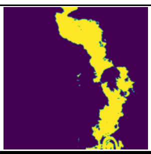
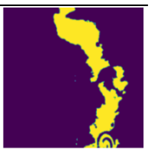
To give a better overview about the developed Co-CrackSegment approach, a detailed visual comparison between the different Co-CrackSegment frameworks as well as the core models are given in this section. Two groups of image samples from DeepCrack and Rissbilder dataset are tested as in Tables 7 and 8, respectively. The image sample groups contain several challenging aspects. The test group1 contains eight samples, in which test sample1 contains a wide discontinues crack with augmentation feature at the end of it.

From Table 7, it can be seen that the most reduced noise and closest crack map to ground truth has been achieved by the Co-CrackSegment/Pixel_Acc in case of Test sample 1. Test sample 2 contains a thin lateral crack with a blurry background which makes the pixel level identification challenging. Nevertheless, all the Co-CrackSegment frameworks have achieved very close crack map to the ground truth. Test sample 3 contains a wide crack with two challenging spots above and beneath. However, all the Co-CrackSegment methods have achieved very good matches with the ground truth image eliminating the background challenging spots. Image sample 4 contains a thin spalling with small voids in the background which contribute to highly noisy background. Except the tiny crack portion in the down-left corner, it can be seen that the Co-CrackSegment/Dice and Co-CrackSegment/IoU have given the best crack map images. Image sample 5 contains one wide crack with a repaired part in the middle as well as a very thin crack above it. It can be seen from the results that the Co-CrackSegment/Dice and Co-CrackSegment/IoU as well as Co-CrackSegment/Recall have achieved the best crack maps comparing to other methods. Test sample 6 includes a transverse crack with complex colored background and scaling in addition to bulges in the middle. It is reported that the Co-CrackSegment/Dice and Co-CrackSegment/IoU as well as Co-CrackSegment/Precision have achieved the best crack maps comparing to other methods. In Test sample 7, spalling cracks are distributed along the image with some voids in the background and very thin cracks around the main crack and the lower left part of the image. It can be observed that all the Co-CrackSegment methods have achieved relatively good crack maps with a tradeoff between the elimination of background voids and the thin crack portions.

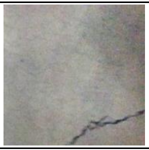
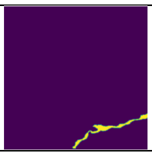
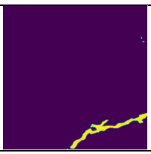
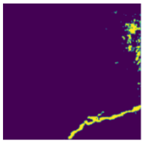
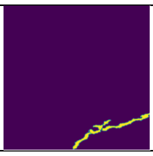
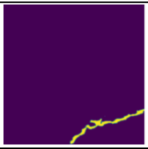
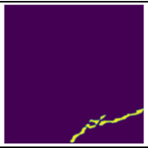
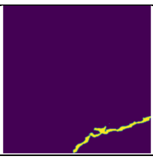
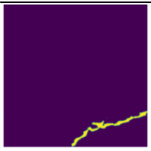
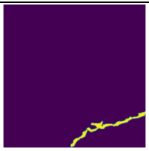
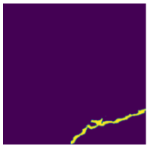
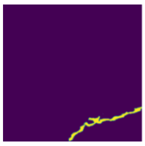
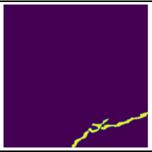
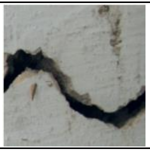
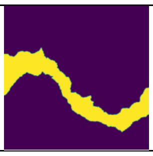
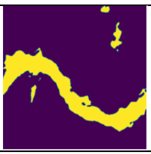
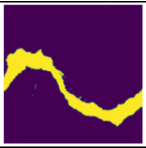
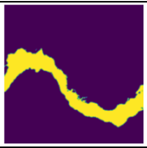
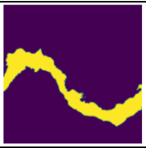
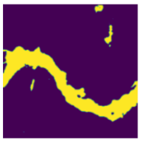
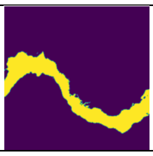
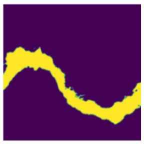
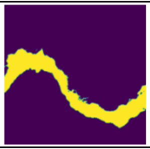
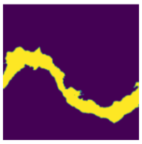
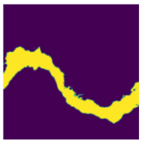
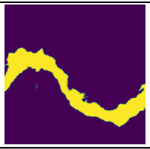
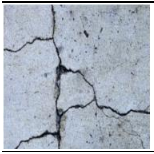
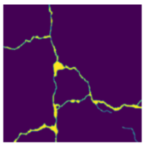
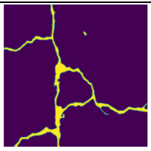
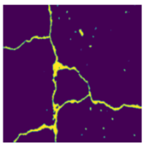
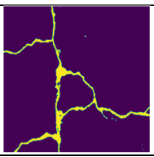
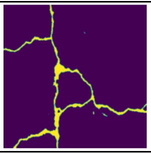
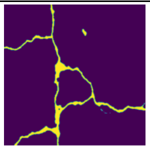
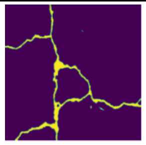
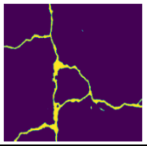
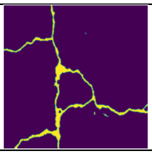
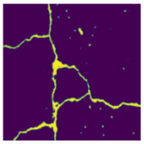
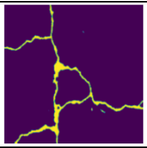
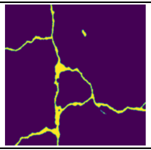
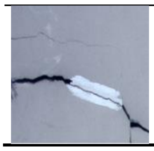
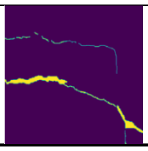
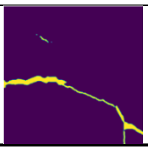
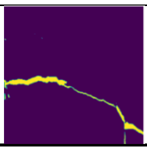
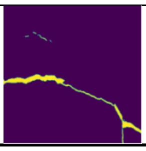
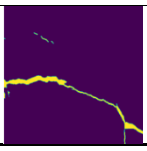
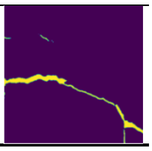
From Table 8, the Test sample 1 has three main cracks with thin ends. It can be observed that the Co-CrackSegment/Dice and Co-CrackSegment/IoU as well as Co-CrackSegment/Recall have achieved the best reduced noise and closet matches to original ground truth. Test sample 2 contains two main cracks with a crack like scaling at the left side. It can be reported that the Co-CrackSegment/Dice and Co-CrackSegment/IoU have delivered better crack maps comparing to other models with main advantage of reduced noise in the background. Test sample 3 contains a very thin vertical crack with scales in the background. It is reported that all the Co-CrackSegment as well as the weighted average have successfully eliminated the scaling positions and accurately located the crack area. In Test sample 4, only very minor lateral crack with complex color of the background and scaling like spots. Nevertheless, all the Co-CrackSegment methods have achieved very good matches with the ground truth image eliminating the background challenging spots. In Test sample 5, a spalling crack can be seen in the lower part of the image with a vertical crack along the image and a scaling region in the background. It can be seen that the Co-CrackSegment/Dice, Co-CrackSegment/IoU, and Co-CrackSegment/Pixel_Acc the weighted average models have delivered the best crack maps comparing to the original image and the ground truth. In Test sample 6, a horizontal crack with interconnected vertical crack can be observed as well complex crack like scaling in the background. It is reported that the CrackSegment/Dice, Co-CrackSegment/IoU, and Co-CrackSegment/Pixel_Acc have also given the best crack maps comparing to the original image and ground truth. In Test sample 7, a very thin crack tree with complex color and illumination of the background as well as crack-like scaling can be observed. All the Co-CrackSegment models and weighted average have delivered very excellent pixel level segmenation of the crack except the Co-CrackSegment/Recall that misclassified the pixel of the crack-like scaling. Finally, it is clear that the Co-CrackSegment/Dice and Co-CrackSegment/IoU frameworks have achieved the best performance comparing to other Co-CrackSegment frameworks and the weighted average method. This confirms the results presented in the previous discussion.

It is important to note that the test samples are randomly chosen image samples with very challenging feature maps. This cannot fully reflect the overall model performances that can be better observed from the statistical results. However, it can assist to provide a better visual analysis on the pixel level crack segmentation performances when the models are fed with complex and challenging images.

Table 7. Visual evaluation of the compared models using image samples of dataset 1.

Test sample 1	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						

doi:10.20944/preprints202408.1801.v1

Test sample 2	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						
Test sample 3	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						
Test sample 4	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						
Test sample 5	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						

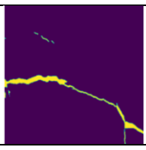
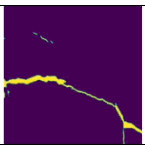
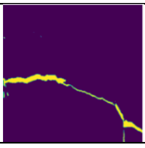
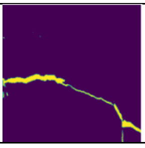
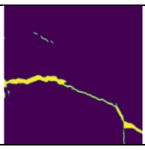
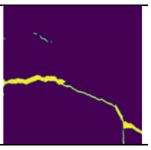
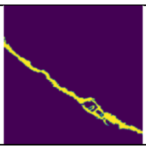
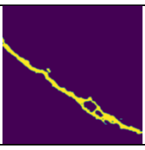
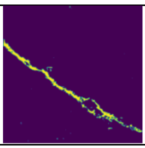
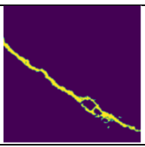
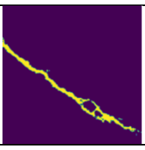
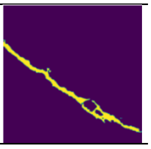
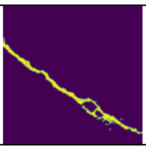
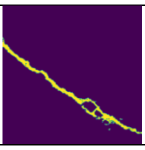
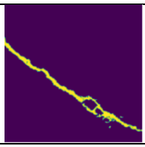
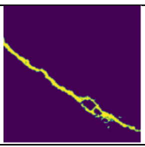
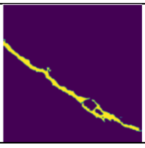
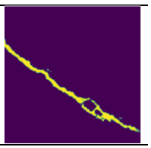
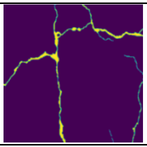
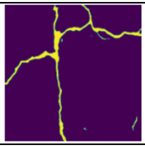
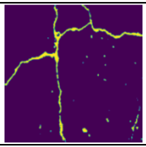
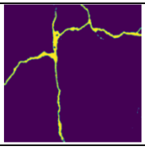
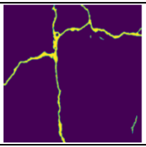
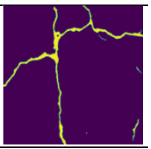
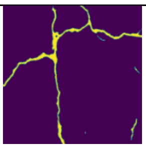
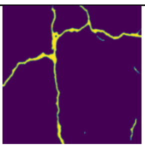
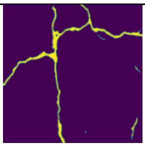
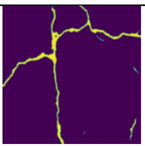
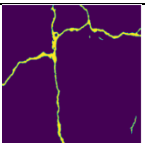
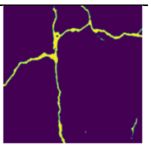
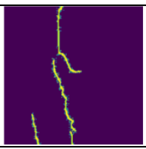
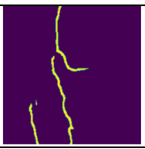
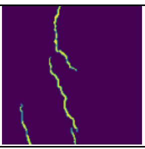
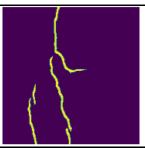
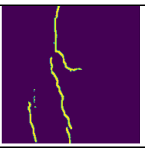
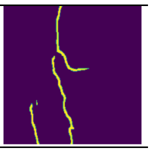
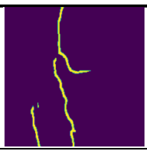
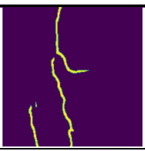
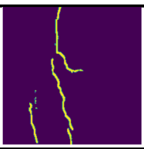
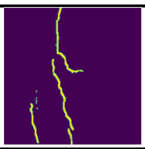
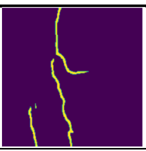
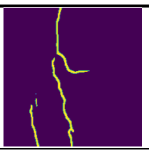
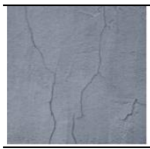
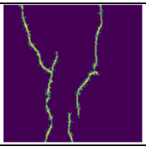
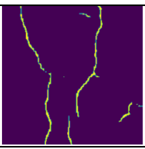
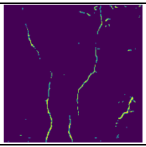
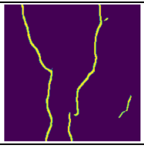
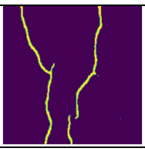
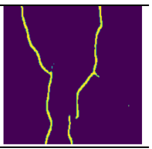
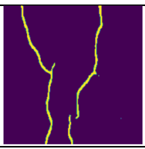
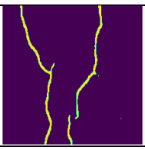
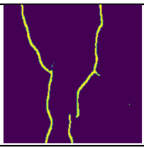
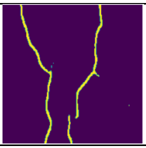
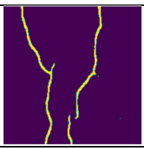
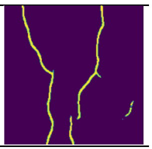
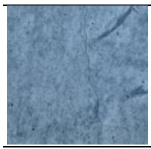
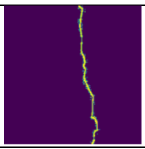
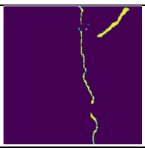
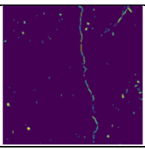
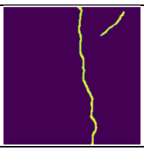
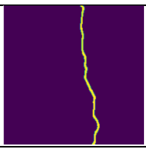
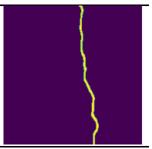
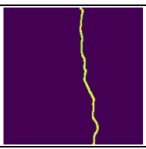
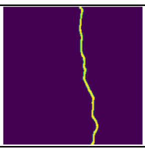
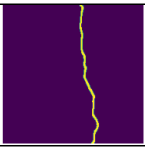
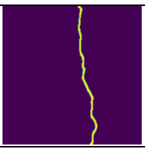
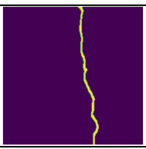
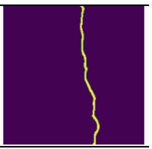
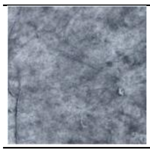
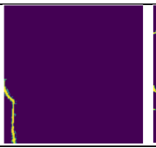
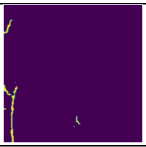
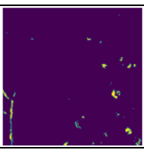
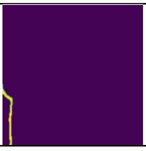
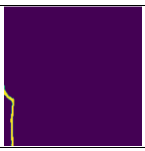
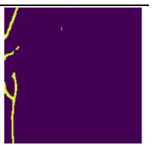
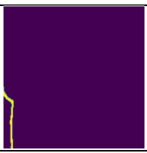
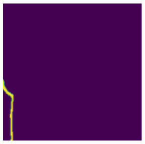
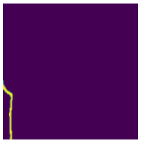
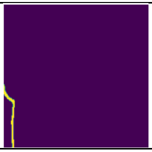
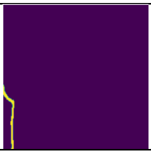
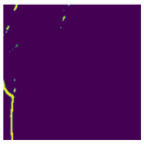
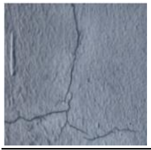
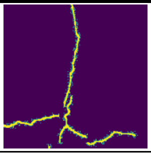
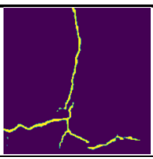
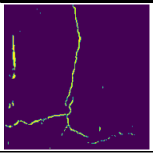
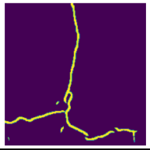
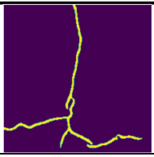
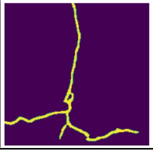
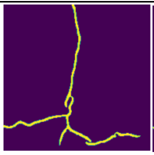
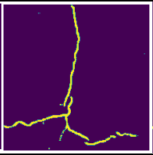
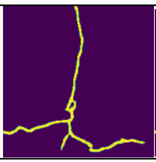
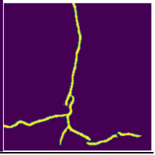
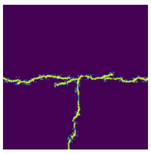
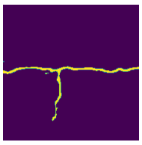
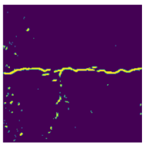
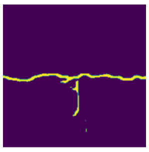
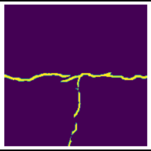
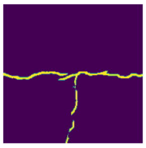
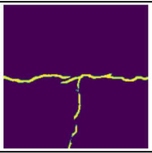
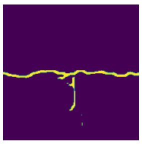
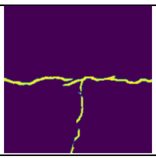
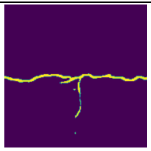
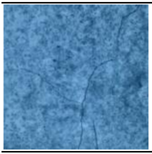
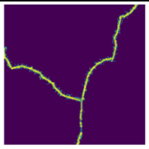
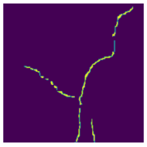
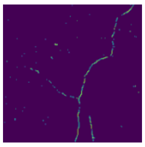
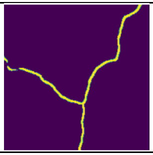
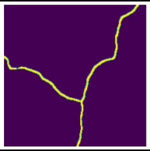
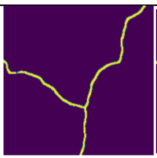
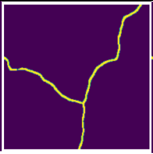
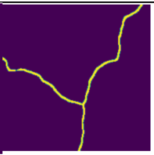
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						
Test sample 6	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						
Test sample 7	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						

Table 8. Visual evaluation of the compared models using image samples of dataset 2.

Test sample 1	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble

						
Test sample 2	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						
Test sample 3	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						
Test sample 4	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						
Test sample 5	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101

						
	Ensemble (Dice)	Ensemble (IoU)	Ensemble (Pixel_Acc)	Ensemble (Precision)	Ensemble (Recall)	Ensemble (weighted average)
						
Test sample 6	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						
Test sample 7	Ground Truth	DeepLabV3 Plus resnet50	DeepCrack19	SegNet	Unet	DeepLabV3 Plus resnet101
						
	Co-CrackSegment/ Dice	Co-CrackSegment/ IoU	Co-CrackSegment/ Pixel_Acc	Co-CrackSegment/ Precision	Co-CrackSegment/ Recall	Weighted average ensemble
						

5. Conclusion

In this research, a novel collaborative deep learning approach called Co-CrackSegment for the purpose of surface crack semantic segmentation has been proposed. For the purpose of constructing the Co-CrackSegment, five core models, namely the DeepLabV3/ResNet50, U-net, DeepCrack19, SegNet, and DeepLabV3/ResNet101 have been trained using two different datasets. Subsequently, the Co-CrackSegment has been tested by taking into account five frameworks using the optimal dice score (Co-CrackSegment/dice), optimal IoU (Co-CrackSegment/IoU), optimal pixel accuracy (Co-CrackSegment/Pixel_Acc), optimal precision (Co-CrackSegment/Precision), and optimal recall (Co-

doi:10.20944/preprints202408.1801.v1

CrackSegment/Recall). Comparisons have been made between the core models and the different Co-CrackSegment frameworks using the tabulated and visual aspects. Furthermore, challenging test images with complex patterns have been chosen to perform visual comparison between both the core models and the developed Co-CrackSegment models. The overall findings of this paper can be summarized as follows:

- Under the theme of the core models, it has been reported that the U-net has achieved a prominent performance by means of loss, pixel accuracy, IoU, dice, and mAP when trained using dataset1. It has also been observed that the DLV3/ResNet50 and DLV3/ResNet101 have also achieved high performances when compromising the overall evaluation metrics. Moreover, the iteration per second score of the DeepCrack19 has given it competitive advantages as a computationally efficient model. Moreover, when trained using dataset2 and similar to dataset1, the U-net has also achieved an outstanding by means of loss, accuracy, IoU, recall, dice, and mAP. Also, the DeepCrack19 has shown better computational performance than the other models when considering the number of iterations per seconds.
- When studying the proposed corporative semantic segmentation Co-CrackSegment approach, the Co-CrackSegment/dice and Co-CrackSegment/IoU have shown the best trade off evaluation scores comparing to other Co-CrackSegment frameworks. Furthermore, when comparing to weighted average method, most Co-CrackSegment frameworks outperformed the weighted average ensemble as well as the core models by means of all evaluation metrics. This is because the traditional weighted average ensemble learning for pixel-level semantic segmentation suffer from pixel blurring of crack boundaries due to average predictions resulting in high bias of predicted crack map than the ground truth. Furthermore, when comparing the results of core models with the Co-CrackSegment frameworks, it has been observed that the corporative learning approach has boosted the performance of the individual models by means of all evaluation metrics. This proofs the efficiency of Co-CrackSegment approach for pixel level semantic segmentation of surface cracks.
- When studying feeding the developed models with test samples that contain many challenges, such as crack-like scaling, foreign objects, this cracks, bulges, voids, spalling, etc. It has been reported that all the developed Co-CrackSegment approach for pixel level semantic segmentation of surface cracks have given very enhanced crack maps even in challenging cases. Also, the Co-CrackSegment/Dice and Co-CrackSegment/IoU frameworks have achieved the best performance comparing to other Co-CrackSegment frameworks and the weighted average method as well as the core models. This confirms the results presented in the previous discussion.
- Finally, several future improvements can be done to improve the proposed method. Firstly, the Co-CrackSegment approach can accept the insertion of any semantic segmentation model. This mainly due to its flexibility to add core models to its main framework. Moreover, the Co-CrackSegment method can be boosted by improving the utilized performance metrics to make a better trade-off between the original performance metrics that have already been used in its framework. Furthermore, the proposed Co-CrackSegment method can be further improved for multi-level semantic segmentation of structural surface defects. Lastly, the Co-CrackSegment can be easily adapted to be used in other semantic segmentation applications.

Supplementary Materials: The following supporting information can be downloaded at: The following supporting information can be downloaded at the website of this paper posted on Preprints.org, Figure S1: title; Table S1: title; Video S1: title.

Author Contributions: For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used "Conceptualization, N.F.A and A.M.; methodology, N.F.A and A.M.; software, N.F.A and A.M.; validation, N.F.A, A.M., X.Z, L.S, P.G.A. Q.W. and M.C.; formal analysis, N.F.A, A.M., X.Z, L.S, P.G.A. Q.W. and M.C.; investigation, N.F.A, and A.M.; resources, N.F.A, A.M., and M.C.; data curation, N.F.A, and A.M.; writing—original draft preparation, N.F.A and A.M.; writing—review and editing, N.F.A, A.M., Q.W. and M.C.; visualization, N.F.A, A.M., X.Z, L.S, P.G.A. Q.W. and M.C.; supervision, N.F.A, A.M., X.Z, L.S, P.G.A. Q.W. and M.C.; project administration, N.F.A, A.M., X.Z, L.S, P.G.A. Q.W. and M.C.; funding acquisition, N.F.A, A.M., X.Z, L.S, P.G.A. Q.W. and M.C. All authors have read and agreed to the published version of the manuscript.".

Funding: This work was supported by the Research Fund for International Young Scientists of the National Natural Science Foundation of China (No. 52250410359), the Natural Science Research Start-up Foundation of Recruiting Talents of Nanjing University of Posts and Telecommunications (No. NY223176), and the Jiangsu-Czech Bilateral Co-funding R&D Project (No. BZ2023011).

Data Availability Statement: We encourage all authors of articles published in MDPI journals to share their research data. In this section, please provide details regarding where data supporting reported results can be found, including links to publicly archived datasets analyzed or generated during the study. Where no new data were created, or where data is unavailable due to privacy or ethical restrictions, a statement is still required. Suggested Data Availability Statements are available in section "MDPI Research Data Policies" at https://www.mdpi.com/ethics.

Acknowledgments: Datasets used in this work are public datasets.

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

References

1. N. F. Alkayem, M. Cao, Y. Zhang, M. Bayat and Z. Su, "Structural damage detection using finite element model updating with evolutionary algorithms: a survey," Neural Computing and Applications, vol. 30, no. 2, p. 389–411, 2018.
2. S. D. Nguyen, T. S. Tran, V. P. Tran, H. J. Lee, M. J. Piran and V. P. Le, "Deep Learning-Based Crack Detection: A Survey," International Journal of Pavement Research and Technology, vol. 16, p. 943–967, 2023.
3. P. M. Bhatt, R. K. Malhan, P. Rajendran, B. C. Shah, S. Thakar, Y. J. Yoon and S. K. Gupta, "Image-Based Surface Defect Detection Using Deep Learning: A Review," Journal of Computing and Information Science in Engineering, vol. 21, no. 4, p. 040801, 2021.
4. A. T. G. Tapeh and M. Z. Naser, "Artificial Intelligence, Machine Learning, and Deep Learning in Structural Engineering: A Scientometrics Review of Trends and Best Practices," Archives of Computational Methods in Engineering, vol. 30, pp. 115-159, 2023.
5. H.-T. Thai, "Machine learning for structural engineering: A state-of-the-art review," Structures, vol. 38, p. 448–491, 2022.
6. M. Cao, N. F. Alkaye m, L. Pan and D. Novák, "Advanced methods in neural networks-based sensitivity analysis with their applications in civil engineering," in Artificial neural networks: models and applications, Rijeka, Croatia, IntechOpen, 2016.
7. D. H. Nguyen and M. A. Wahab, "Damage detection in slab structures based on two-dimensional curvature mode shape method and Faster R-CNN," Advances in Engineering Software, vol. 176, p. 103371, 2023.
8. L. Yu, S. He, X. Liu, S. Jiang and S. Xiang, "Intelligent Crack Detection and Quantification in the Concrete Bridge: A Deep Learning-Assisted Image Processing Approach," Advances in Civil Engineering, vol. 2022, p. 1813821, 2022.
9. P. Kaewniam, M. Cao, et al. "Recent advances in damage detection of wind turbine blades: A state-of-the-art review," Renewable and Sustainable Energy Reviews, vol. 167, p. 112723, 2022.
10. S.-J. Wang, J.-K. Zhang and X.-Q. Lu, "Research on Real-Time Detection Algorithm for Pavement Cracks Based on SparseInst-CDSM," Mathematics, vol. 11, no. 5, p. 3277, 2023.
11. G. Yu and X. Zhou, "An Improved YOLOv5 Crack Detection Method Combined with a Bottleneck Transformer," Mathematics 2023, 11(10), 2377, vol. 11, no. 10, p. 2377, 2023.
12. T. S. Tran, S. D. Nguyen, H. J. Lee and V. P. Tran, "Advanced crack detection and segmentation on bridge decks using deep learning," Construction and Building Materials, vol. 400, p. 132839, 2023.
13. J. Zhang, Y.-Y. Cai, D. Yang, Y. Yuan, W.-Y. He and Y.-J. Wang, "MobileNetV3-BLS: A broad learning approach for automatic concrete surface crack detection," Construction and Building Materials, vol. 392, p. 131941, 2023.
14. N. F. Al kayem, L. Shen, A. Mayya, P. G. Asteris, R. Fu, G. D. Luzio, A. Strauss and M. Cao, "Prediction of concrete and FRC properties at high temperature using machine and deep learning: A review of recent advances and future perspectives," Journal of Building Engineering, vol. 83, p. 108369, 2024.
15. R. Fu, M. Cao, D. Novák, et al. "Extended efficient convolutional neural network for concrete crack detection with illustrated merits," Automation in Construction, vol. 156, p. 105098, 2023.
16. C. Xiong, T. Zayed and E. M. Abdelkader, "A novel YOLOv8-GAM-Wise-IoU model for automated detection of bridge surface cracks," Construction and Building Materials, vol. 414, p. 135025, 2024.
17. N. F. Al kayem, M. Cao and M. Ragulskis, "Damage Diagnosis in 3D Structures Using a Novel Hybrid Multiobjective Optimization and FE Model Updating Framework," Complexity, vol. 2018, p. 3541676 , 2018.
18. M. Cao, P. Qiao and Q. Ren, "Improved hybrid wavelet neural network methodology for time-varying behavior prediction of engineering structures," Neural Computing and Applications, vol. 18, pp. 821-832, 2009.

19. N. F. A lkayem and M. Cao, "Damage identification in three-dimensional structures using single-objective evolutionary algorithms and finite element model updating: evaluation and comparison," Engineering Optimization, vol. 50, no. 10, pp. 1695-1714, 2018.
20. P. Arafin, A. M. Billah and A. Issa, "Deep learning-based concrete defects classification and detection using semantic segmentation," Structural Health Monitoring, vol. 23, no. 1, p. 383–409, 2024.
21. J. Hang, Y. Wu, Y. Li, T. Lai, J. Zhang and Y. Li, "A deep learning semantic segmentation network with attention mechanism for concrete crack detection," Structural Health Monitoring, vol. 22, no. 5, pp. 3006-3026, 2023.
22. D. Tabernik, M. Šuc and D. Skočaj, "Automated detection and segmentation of cracks in concrete surfaces using joined segmentation and classification deep neural network," Construction and Building Materials, vol. 408, p. 133582, 2023.
23. J. Shang, J. Xu, A. A. Zhang, Y. Liu, K. C. Wang, D. Ren, H. Zhang, Z. Dong and A. He, "Automatic Pixel-level pavement sealed crack detection using Multi-fusion U-Net network," Measurement, vol. 208, p. 112475, 2023.
24. B. Chen, H. Zhang, G. Wang, J. Huo, Y. Li and L. Li, "Automatic concrete infrastructure crack semantic segmentation using deep learning," Automation in Construction, vol. 152, p. 104950, 2023.
25. L. M. Dang, H. Wang, Y. Li, L. Q. Nguyen, T. N. Nguyen, H.-K. Song and H. Moon, "Lightweight pixel-level semantic segmentation and analysis for sewer defects using deep learning," Construction and Building Materials, vol. 371, p. 130792, 2023.
26. D. Joshi, T. P. Singh and G. Sharma, "Automatic surface crack detection using segmentation-based deep-learning approach," Engineering Fracture Mechanics, vol. 268, p. 108467, 2022.
27. M. Mishra, V. Jain, S. K. Singh and D. Maity, "Two stage method based on the you only look once framework and image segmentation for crack detection in concrete structures," Architecture, Structures and Construction, vol. 3, p. 429–446, 2023.
28. P. Shi, S. Shao, X. Fan, Z. Zhou and Y. Xin, "MCL-CrackNet: A Concrete Crack Segmentation Network Using Multilevel Contrastive Learning," IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, vol. 72, p. 5030415, 2023.
29. P. Savino and F. Tondolo, "Civil infrastructure defect assessment using pixel wise segmentation based on deep learning," Journal of Civil Structural Health Monitoring, vol. 13, p. 35–48, 2023.
30. P. N. Hadinata, D. Simanta, L. Eddy and K. Nagai, "Multiclass Segmentation of Concrete Surface Damages Using U-Net and DeepLabV3+," Applied Sciences, vol. 13, p. 2398, 2023.
31. Z. Al-Huda, B. Peng, R. N. A. Algburi, M. A. Al-antari, R. AL-Jarazi and D. Zhai, "A hybrid deep learning pavement crack semantic segmentation," Engineering Applications of Artificial Intelligence, vol. 122 , p. 106142, 2023.
32. R. Ali, J. H. Chuah, M. S. A. Talip, N. Mokhtar and M. A. Shoaib, "Automatic pixel-level crack segmentation in images using fully convolutional neural network based on residual blocks and pixel local weights," Engineering Applications of Artificial Intelligence, vol. 104, p. 104391, 2021.
33. D. Kang, S. S. Benipal, D. L. Gopal and Y.-J. Cha, "Hybrid pixel-level concrete crack segmentation and quantification across complex backgrounds using deep learning," Automation in Construction, vol. 118, p. 103291, 2020.
34. C. Sha, C. Yue and W. Wang, "Ensemble 1D DenseNet Damage Identification Method Based on Vibration Acceleration," Structural Durability & Health Monitoring, vol. 17, no. 5, pp. 369-381, 2023.
35. V. Kailkhura, S. Aravindh, S. S. Jha and N. Jayanth, "Ensemble learning-based approach for crack detection using CNN," in Proceedings of the Fourth International Conference on Trends in Electronics and Informatics (ICOEI 2020), 2020.
36. Z. Fan, C. Li, Y. Chen, P. Mascio, X. Chen, G. Zhu and G. Loprencipe, "Ensemble of Deep Convolutional Neural Networks for Automatic Pavement Crack Detection and Measurement," Coatings, vol. 10, p. 152, 2020.
37. Y. Hong and S. B. Yoo, "OASIS-Net: Morphological Attention Ensemble Learning for Surface Defect Detection," Mathematics, vol. 10, p. 4114, 2022.
38. M. S. Barkhordari, D. J. Armaghani and P. G. Asteris, "Structural Damage Identification Using Ensemble Deep Convolutional Neural Network Models," Computer Modeling in Engineering & Sciences, vol. 134, no. 2, pp. 835-855, 2023.
39. A. A. Maarouf and F. Hachouf, "Transfer Learning-based Ensemble Deep Learning for Road Cracks Detection," in International Conference on Advanced Aspects of Software Engineering (ICAASE), Constantine, Algeria, 2022 .
40. W. Bousselham, G. Thibault, L. Pagano and A. Machireddy, "Efficient Self-Ensemble for Semantic Segmentation," arXiv , p. arXiv:cs.CV/2111.13280, 2022.
41. I. Nigam, C. Huang and D. Ramanan, "Ensemble Knowledge Transfer for Semantic Segmentation," in IEEE Winter Conference on Applications of Computer Vision, Lake Tahoe, NV, USA, 2018.

42. L. Zhang, S. Slade, C. P. Lim, H. Asadi, S. Nahavandi, H. Huang and H. Ruan, "Semantic segmentation using Firefly Algorithm-based evolving ensemble deep neural networks," Knowledge-Based Systems, vol. 277, p. 110828, 2023.
43. C. Lee, S. Yoo, S. Kim and J. Lee, "Progressive Weighted Self-Training Ensemble for Multi-Type Skin Lesion Semantic Segmentation," IEEE Access, vol. 10, pp. 132376-132383, 2022.
44. T. Lee, J.-H. Kim, S.-J. Lee, S.-K. Ryu and B.-C. Joo, "Improvement of Concrete Crack Segmentation Performance Using Stacking Ensemble Learning," Applied Sciences, vol. 13, p. 2367, 2023.
45. S. Li and X. Zhao, "A Performance Improvement Strategy for Concrete Damage Detection Using Stacking Ensemble Learning of Multiple Semantic Segmentation Networks," Sensors, vol. 22, p. 3341, 2022.
46. G. E. Amieghemen and M. M. Sherif, "Deep convolutional neural network ensemble for pavement crack detection using high elevation UAV images," Structure and Infrastructure Engineering, p. https://doi.org/10.1080/15732479.2023.2263441, 2023.
47. G. Cyganov, A. Rychenkov, A. Sinitca and D. Kaplun, "Using the fuzzy integrals for the ensemble based segmentation of asphalt cracks," Industrial Artificial Intelligence, vol. 1, p. 5, 2023.
48. Y. Chen, Y. Mo, A. Readie, G. Ligozio, I. Mandal, F. Jabbar, T. Coroller and B. W. Papież, "VertXNet: an ensemble method for vertebral body segmentation and identification from cervical and lumbar spinal X rays," Scientific Reports, vol. 14, p. 3341, 2024.
49. R. Bao, K. Palaniappan, Y. Zhao and G. Seetharaman, "GLSNet++: Global and Local-Stream Feature Fusion for LiDAR Point Cloud Semantic Segmentation Using GNN Demixing Block," IEEE SENSORS JOURNAL, vol. 24, no. 7, pp. 11610-11624, 2024.
50. D. Dais, I. E. Bal, E. Smyrou and V. Sarhosis, "Automatic crack classification and segmentation on masonry surfaces using convolutional neural networks and transfer learning," Automation in Construction, vol. 125, p. 103606, 2021.
51. R. Vij and S. Arora, "A hybrid evolutionary weighted ensemble of deep transfer learning models for retinal vessel segmentation and diabetic retinopathy detection," Computers and Electrical Engineering , vol. 115, p. 109107, 2024.
52. Z. Fan, C. Li, Y. Chen, P. D. Mascio, X. Chen, G. Zhu and G. Loprencipe, "Ensemble of Deep Convolutional Neural Networks for Automatic Pavement Crack Detection and Measurement," Coatings, vol. 10, p. 152, 2020.
53. K. S. Devan, H. A. Kestler, C. Read and P. Walther, "Weighted average ensemble based semantic segmentation in biological electron microscopy images," Histochemistry and Cell Biology, vol. 158, p. 447–462, 2022.
54. F. Panella, A. Lipani and J. Boehm, "Semantic segmentation of cracks: Data challenges and architecture," Automation in Construction, vol. 135 , p. 104110, 2022.
55. Y. Liu, J. Yao, X. Lu, R. Xie and L. Li, "DeepCrack: A deep hierarchical feature learning architecture for crack segmentation," Neurocomputing, vol. 338, pp. 139-153, 2019.
56. S. Kulkarni, S. Singh, D. Balakrishnan, S. Sharma, S. Devunuri and S. C. R. Korlapati, "CrackSeg9k: A Collection and Benchmark for Crack Segmentation Datasets and Frameworks," in Karlinsky, L., Michaeli, T., Nishino, K. (eds) Computer Vision – ECCV 2022 Workshops. ECCV 2022. Lecture Notes in Computer Science, vol 13807, Springer, Cham, 2022.
57. M. Pak and S. Kim, "Crack Detection Using Fully Convolutional Network in Wall-Climbing Robot," in Park, J.J., Fong, S.J., Pan, Y., Sung, Y. (eds) Advances in Computer Science and Ubiquitous Computing. Lecture Notes in Electrical Engineering, vol 715., Springer, Singapore. , 2021.
58. O. Ronneberger, P. Fischer and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science, vol 9351. , Springer, Cham, 2015.
59. K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv, p. arXiv:1409.1556, 2014.
60. L.-C. Chen, G. Papandreou, F. Schroff and H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation," arXiv, p. arXiv:1706.05587, 2017.

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