Probability of detection curve for the automatic visual inspection of steel bridges

ce/papers Proceedings in civil engineering

Ernst & Sohn
A Wiley Brand

ORIGINAL ARTICLE

eurosteel AMSTERDAM 2023

Ernst & Sohn
A Wiley Brand

Probability of detection curve for the automatic visual inspection of steel bridges

Andrii Kompanets[1,2] | Davide Leonetti[1,2] | Remco Duits[2,3]| Johan Maljaars[1,4] |H.H. (Bert) Snijder[1]

Correspondence

Andrii Kompanets
Eindhoven University
of Technology,
Groene Loper 3,
Eindhoven, 5612 AE,
a.kompanets@tue.nl

[1] Eindhoven University of Technology, Department of the Built Environment, Eindhoven, The Netherlands
[2] Eindhoven Artificial Intelligence Systems Institute, Eindhoven, The Netherlands
[3]Eindhoven University of Technology, Department of Mathematics and Computer Science, Eindhoven, The Netherlands
[4] TNO, Delft, The Netherlands

Abstract

The damage tolerant design philosophy is based on periodical inspections and provides safe bridge operation preventing fatigue cracks from growing up to a critical size. It is possible to optimize the management costs of a bridge throughout its life by designing the inspection frequency, which depends on the capabilities of the inspection method. However, such optimization requires knowledge about the performance of inspections that are envisioned to take place during the use of the bridge. Regular visual inspections is the most frequently applied type of inspection of bridge structures. Recent advances in computer vision technologies provide a strong basis for the development of automatic damage detection systems that can support regular visual inspection, thus increasing the reliability of the inspection. Several automatic crack detection systems have been developed in the past years. However, the performances of such systems have not been evaluated in the way as for traditional non-destructive inspection methods, i.e. in terms of probability of detection curves and detectability limits. This restricts the applicability of automatic visual inspections for inspection planning and damage tolerant design. This paper proposes an encoder-decoder neural network for segmentation of cracks on images of steel bridges. A probability of detection curve is calculated for this neural network.

Keywords

Bridge inspection, Computer vision, Probability of detection

1 Introduction

Periodical inspections using non-destructive evaluation (NDE) methods ensure safe bridge operation. During bridge inspections, deterioration of the structure, such as fatigue cracks or corrosion, are identified, and any defective parts can be repaired or replaced as necessary. Timely detection of a defect largely depends on the performance of the NDE method used and the probability of detection (PoD) curve serves as a standard approach to characterize the performance of an NDE method. Often, the probability of a flaw, e.g. crack, being detected is plotted as a function of the flaw size, e.g. crack length. This relationship is called a PoD curve.

PoD curves have a variety of practical applications. Apart from solely NDE methods performance assessment [1,2], PoD curves are used for bridge management planning and fatigue reliability assessment. For example, in [3], a PoD curve was combined with crack growth models and a fatigue reliability model, to establish a procedure for structural performance assessment and management planning for steel bridges. Also, the estimate of the fatigue life [4] and the structural safety [5] of a bridge after inspection can be updated using PoD curves. Finally, in [6], a rough estimate of the probability of detection based on experience of experts was used to derive safety factors for the fatigue design of bridge structures. So far, regular visual inspection is the mostly used type of NDE in bridge inspections [7], even though its reliability is low [8] and it induces considerable financial expenses [9]. With the mentioned drawback of regular visual inspections in mind, much attention has been paid to the development of automatic visual inspection methods [10], that potentially can increase the reliability of bridge inspections and reduce their costs [11]. Such an automatic bridge inspection system would consist of an image/video acquisition setup, e.g. camera mounted on a drone, and a computer vision system for automatic damage detection.

In [12], automatic bridge inspection using unmanned aerial vehicles (UAV) was studied. UAV can easily carry a camera to remote parts of bridges, which can be difficult to be accessed by a human inspector. However, due to safety considerations it is not always possible to get closer than 1 meter to a bridge surface [13]. Alternative robotic systems were considered in [14]. For example, climbing

https://doi.org/10.1002/cepa.2585

robots can be used to access a closed bridge structure that would be impossible to access with UAV. However, it may be inefficient to inspect an entire bridge structure using such a robot.

To fully employ all possible benefits that an automatic inspection system can bring, its performance has to be integrated into fatigue reliability analysis and optimal inspection planning strategies. This integration requires a PoD curve to be determined.

To the best of the authors' knowledge, this paper presents a first attempt to generate a PoD curve for an automatic crack detection system on images of steel bridges. It should be emphasized that the study considers only the performance of the computer vision algorithm for crack detection and not the whole inspection system also including an image acquisition setup. This is because images used for the PoD determination were collected by a hand-held camera, during regular bridge inspection. Images collected in such a way may be different from images collected by an automatic inspection system such as a camera installed on UAV. The main contribution of this paper is to present:

- A two stage training strategy to train a neural network for crack segmentation on images of steel bridges with reduced false positive rate;
- A statistically underpinned approach to measure the performance of an automatic visual crack detection algorithm using the PoD curve and false positive rate.

In Section 2 of this paper, a computer vision algorithm for crack detection based on an encoder-decoder neural network with a modified U-net architecture and a dataset for its training are described. Further, a neural network training strategy is proposed that allows to reduce the fraction of false positive crack detections. Section 3 gives the methodology of deriving a PoD curve for the developed automatic crack detection algorithm using the maximum likelihood estimate method (MLE). Section 4 describes the results of the PoD curve derivation for automatic visual crack detection and compares the obtained PoD curve with PoD curves for regular visual inspection.

2 Crack detection using deep learning

A variety of deep learning approaches have been applied to the crack detection problem [15]. These approaches can be classified into four groups:

- *image classification approach*, which classifies an image or its part according to the presence or absence of a crack;
- *crack localization approach*, which utilizes algorithms such as Mask R-CNN [16] or YOLO [17] to localize a crack on an image, draw a so-called bounding box around the crack, determining the crack location;
- *crack segmentation approach*, which allows classifying each pixel of the image as either crack or background using neural network architectures such as U-Net [18];
- *combined approach*, where a segmentation algorithm is applied to a part of the whole image where cracks are localized by the localization algorithm [19].

An advantage of the crack segmentation approach, in comparison to the localization and classification approach, is its ability to not only detect and localize cracks but also to provide enough information enabling crack size estimation. The advantage of the segmentation approach over the mixed approach is that the mixed approach involves the sequential application of two neural networks, with the first network responsible for localization and the second network for segmentation. Thus, two separate neural networks have to be trained on datasets, specifically annotated for each of them. The primary limitation of the segmentation approach is its high computational complexity. This arises from the fact that the segmentation neural network often has more weights than the localization network due to the inclusion of an additional decoder which almost doubles the size of the neural network, thus increasing training and inference time. Additionally, segmentation must be applied to the entire image, as opposed to just a part of it as in the combined approach, further increasing the inference time. In this study, the segmentation approach is employed to carry out labelling of each pixel on an image, thus determining whether it belongs to a crack or to the background. This choice can be partly explained by a peculiarity of the image dataset available for this

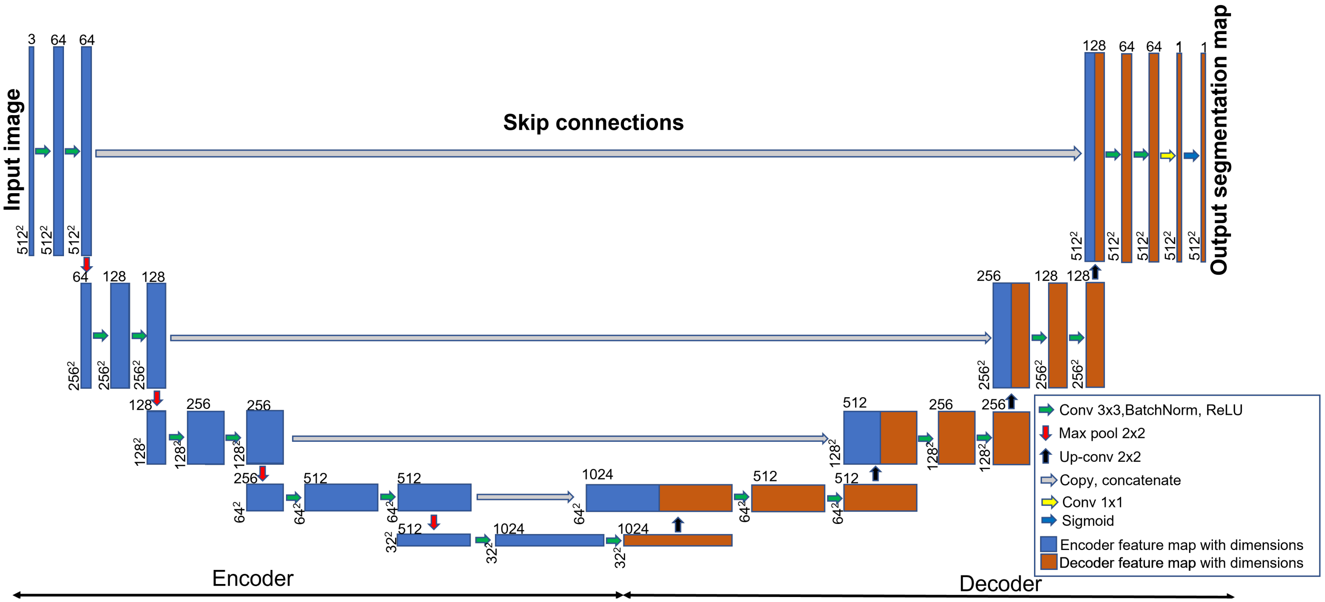

Figure 1 Proposed modified U-net architecture of the neural network

study which is described in Section 2.2. The majority of images with cracks in this dataset contains markings from inspectors that were made to indicate the cracks. Localization and classification neural networks may use information from the entire image, and if trained on this dataset, the neural network could learn to rely on these markings. A neural network that relies on markings for crack detection could have a low performance during actual application because then it would be necessary to detect cracks which were not marked by the inspector in advance. Further in the text, this problem referred as "the marking problem". Meanwhile, the segmentation neural network uses local image information as described in Section 2.2, thus eliminating the described marking problem. However, lack of global information can lead to a high false positive rate as explained in Section 4.

2.1 Architecture

An encoder-decoder neural network with a modified U-net architecture is used in this study, being the same as the one proposed in [20], however, only the encoder-decoder part is used without additional postprocessing steps. Background on the methods and terminology used in this section can be found in [21]. The architecture of the neural network is summarised in Figure 1. The used encoder-decoder network takes images of size 512x512 pixels as input and gives a segmentation map of the same size with each pixel marked as belonging to a crack or to the background. The encoder part of the network consists of four encoding blocks. Each encoding block consists of two 3x3 convolutions each followed by batch normalization and a rectified linear unit (ReLU) activation function. A 2x2 max pooling operation with a stride of 2 is applied to the output of each block to down-sample feature maps. Four of such blocks encode image information into a 32x32x1024 feature map. Further, this feature map is decoded using the decoder part of the network, which consists of the same blocks as the encoder part, but then applying a 2x2 upsample operation to the output of each decoder block instead of a max pooling operation. Long-range skip connections between encoder and decoder parts help to preserve fine-grain spatial information, available at earlier stages of the encoder, but gradually diminish due to the application of max pooling. A 1x1 convolution with sigmoid activation function is applied to the output of the last decoder block to assign values in the range between 0 and 1 to each pixel. Finally a thresholding is applied with a threshold value equal to 0.5 to produce a binary segmentation map with values 0 (background) and 1 (crack) assigned to each pixel.

2.2 Dataset

Roughly 16000 images were collected, mostly with 4608x3456 resolution (and a few images with lower resolution), that were taken by inspectors during regular inspections of steel bridges. These images were obtained in situ, with the objective of conducting bridge inspection and maintenance. The images were captured at varying distances from the bridge, ranging from less than 1 meter up to 10 meters, as estimated based on a visual assessment. To train and validate the proposed neural network, 755 images are selected where crack edges are identifiable. The ground truth pixel-wise labelling of these images is made using a semi-automatic tool available in [22] and described in [23], which is built based on a geometric tracking algorithm [24] and specifically designed for the crack image pixel-wise labelling purpose.

The selected images are randomly divided into training and validation datasets containing 80% and 20% of the images, respectively. As mentioned above, the proposed neural network takes as input images of size 512x512 pixels, because a bigger input image size would imply an unreasonably high computational load. To account for the difference between image size in the dataset and neural network required input size, a special training data loader algorithm is developed that selects patches of pixel size 512x512 from images, as illustrated in Figure 2. The criterions underlying the selection of a patch are explained in Section 2.3. To process the entire image, the neural network is applied in a sliding window manner. This means that the entire image is partitioned into patches so that each pixel is represented in at least one of the patches. After that, each patch is processed separately and a segmentation map of the entire image is constructed.

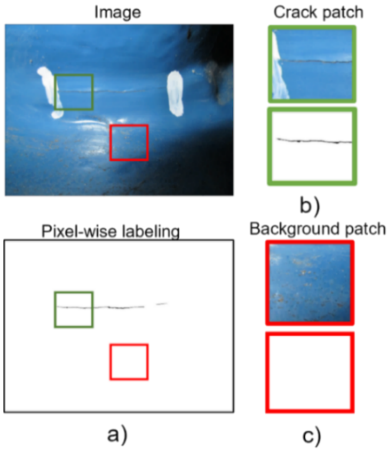

Figure 2 Image of a crack and example of patches used for training and evaluation of the neural network. a) Example of an image of a crack in a steel bridge with its ground truth annotation; b) Example of a patch containing a crack and its ground truth annotation; c) Example of a patch not containing a crack and its ground truth annotation.

2.3 Training with false positive reduction strategy

Commonly, to train a segmentation neural network, patches of the necessary size (here we used a patch size of 512x512 pixels) that contain some part of a crack are randomly taken to compose a training dataset. Often, random background patches that do not contain cracks are also provided [25]. False positive detection is recognized as a significant problem for crack detection algorithms [19]. With the dataset used in this work, the common training strategy trains a neural network with an extremely high false positive rate, when applied to the entire image. Practically this means that in any processed image, independent on the presence of cracks, a significant portion of pixels is marked as a crack. This can be seen in Figure 3, where for images containing a crack (Figure 3a) the segmentation maps (Figure 3c) are produced using a neural network trained with the common training strategy. Comparison of these segmentation maps (Figure 3c) with

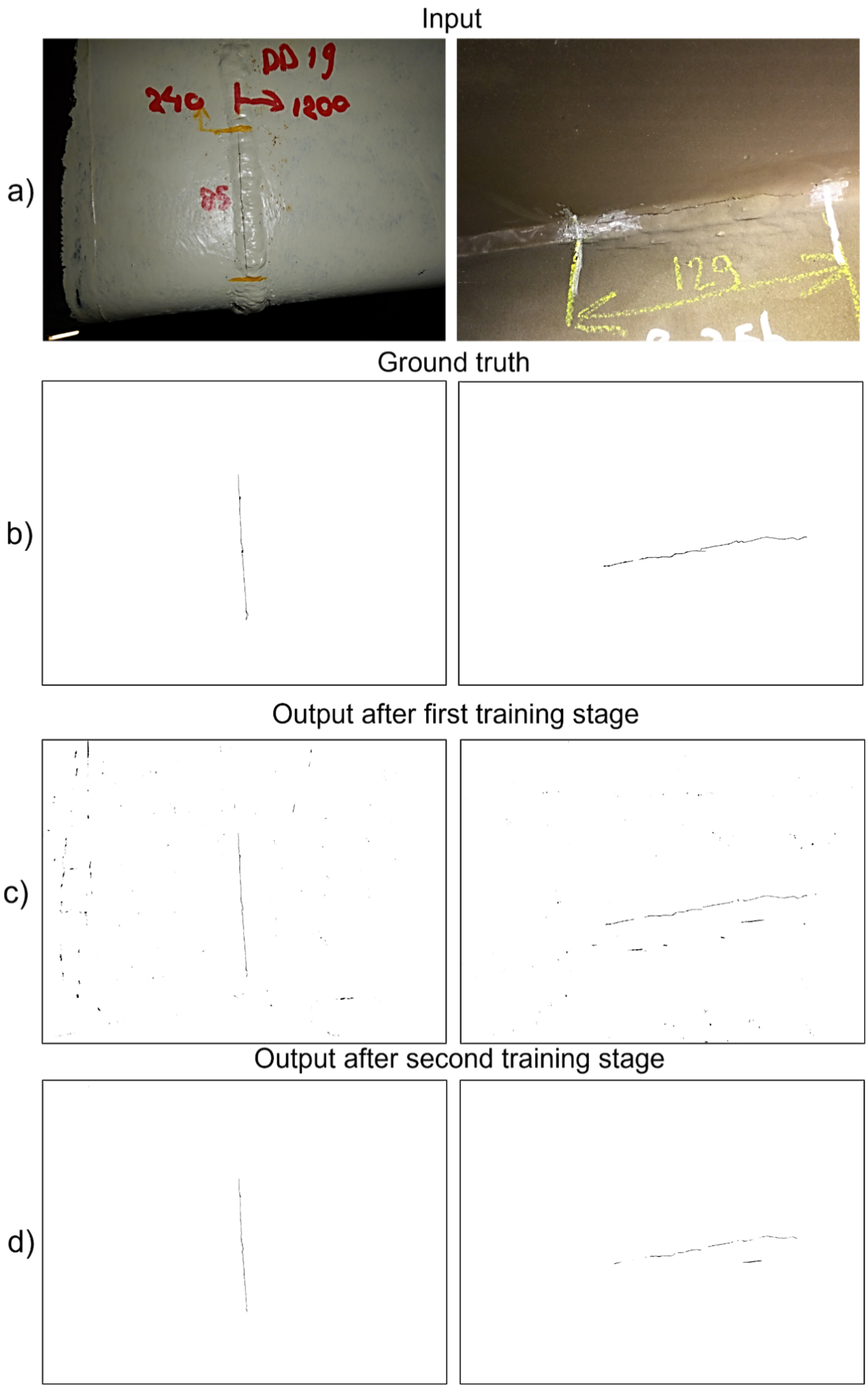
Input
a)
DD 19
240
1200
85
129
Ground truth
b)
Output after first training stage
c)
Output after second training stage
d)

Figure 3 a) Images of cracks in a steel bridge structure; b) ground truth crack annotation; c) encoder-decoder network outputs after the first training stage; d) encoder-decoder network outputs after the second training stage the ground truth crack annotation (Figure 3b) can reveal false positive pixels. The high false positive rate can be explained by the fact that cracks often appear as a dark line on an image and a neural network trained with the common training strategy learns to classify most of such dark lines as crack. However, the images available for this study contain several dark lines that are not a crack (e.g. shadow from protruding structural elements, gaps in connections between two parts of a structure). Often these non-crack image elements that look like a crack are located many pixels away from a crack, meaning they will not often occur in the training crack patches. As a result, a neural network trained using only the crack patches does not "see" enough training examples of the crack-like-looking image elements during the training. Thereby, it is not able to distinguish them from cracks. The problem remains even if random background patches are used for training.

This is because such crack-like image patterns rarely appear in background patches, and most of the background patches capture plain surfaces without any crack-like elements as in Figure 2c. The number of these crack-like image elements shown to a neural network during training depends on their presence on images, number of training patches and the patches size relative to the images size. With images available for this study and with the chosen patch size, the crack and background patches do not capture enough of these crack-like image elements to teach a neural network to distinguish them from cracks.
To solve this problem and to reduce the number of false calls a two-stage training strategy is proposed:

- The first stage uses the common training approach. Only the crack patches are shown to a neural network during its training. Afterwards, locations from entire images where the trained neural network gives false positive crack detections are identified.
- In the second stage, the training of the neural network continues, but in this stage, together with crack patches, background patches taken from false positive locations are shown. Also, in this stage, the loss function is modified to impose higher loss for false positive mistakes than for false negative ones.

A common approach is employed to train the proposed neural network in the first stage of the proposed training strategy. Patches are randomly selected in such a way that at least a few pixels annotated as a crack are present in them. However, a modification is introduced in this common approach: normally, a predefined number of crack patches are generated and stored beforehand to be shown during the training of a neural network, whereas "on-the-fly" patch generation is introduced in the proposed approach. Each time a training example is required, a random image is chosen, and the patch taken from this image is used as a training example. Such "on-the-fly" patch generation practically eliminates the possibility of overfitting, because there is no patch that is shown more than twice during training. One possible drawback of the proposed method is that "on-the-fly" patch generation could increase the training time due to the necessity to generate patches each time a neural network update is done. But parallelization of the patch generation procedure with the neural network parameters update process allows to reduce this time increase effect.

A Dice loss function [26] is used for optimization of the neural network:

$$D_{loss} = 1 - DSC = 1 - \frac{2 \cdot TP}{\sum_j^M p_j^2 + \sum_j^M g_j^2} \qquad (1)$$

where, TP means True Positive segmentations. The sums in the denominator runs over M pixels of the predicted binary segmentation map p_j and of the ground truth binary segmentation map g_j. The neural network training is done using "ADAM" optimizer having parameters β_1=0.9 and β_2=0.999 (default parameters for PyTorch implementation) and a learning rate equal to 0.001 which is multiplied by 0.99 for each training epoch. The batch size was set to 8. The convergence of the proposed neural network is achieved after 200 epochs. Reference [21] gives detailed explanation of the optimization algorithm, and the terminology used in this paragraph.

After the neural network is trained in the first training stage described above, the false positive segmentations are identified. The entire images from the dataset are segmented in a sliding window manner and the produced segmentation maps are compared with ground truth annotations. This comparison enables to recognise the pixels of the segmentation maps produced by the neural network, which are marked as a crack but are background according to the ground truth annotations, i.e. false positive pixels. The recorded false positive locations are needed for the second training stage. It should be noted that false positive segmentations within a 10-pixel range from ground truth annotated cracks are not regarded as false positives, to account for the fact that many cracks do not have clear boundaries on the images. The 10-pixel range (in images with sizes up to 4608x3456 pixels) was chosen as an acceptable deviation.

In the second training step, 50% of the training patches are crack patches and the others are background patches. Moreover, the background patches are taken so that the false positive pixels, recorded as described above, are present in these background patches. Further, in the second training stage, instead of the Dice loss function, the Tversky loss function [27] is used:

$$T_{loss}(\alpha_1, \alpha_2) = 1 - \frac{TP}{TP + \alpha_1 \cdot FP + \alpha_2 \cdot FN} \qquad (2)$$

where, TP means True Positive segmentations, FP means False Positive segmentations, and FN means False Negative segmentations. By adjusting the parameters α_1 and α_2 it is possible to outbalance penalization of the neural network for making false positive and false negative mistakes. In the second stage of the proposed training strategy, α_1 is set to 0.9 and α_2 to 0.1 in order to provide higher penalties for false positive mistakes of the neural network than for false negative mistakes.

The proposed neural network trained from scratch with the unbalanced Tversky loss function and with background patch generation may get stuck in a local minimum of the optimization space, with each output segmentation map being a "whole-white" segmentation map (meaning there will be no pixels marked as a crack). To avoid this local minimum, in the second stage of the proposed training strategy, the neural network parameters are initialized with values that were trained at the first training stage. In other words, the second training stage that now also includes background patches taken from false positive locations continues the neural network training and it does not train from scratch.

3 PoD derivation

Experimental methods to determine PoD curves are given in [28, 29, 30]. In order to derive the PoD curve, specimens with a known flaw size should be examined with the NDE method under consideration. Multiple of such measurements on different specimens should be done to provide statistically meaningful data. Specimens and inspection conditions should represent those that occur in real applications as much as possible. The variability of different factors that may affect inspection outcomes should be fairly represented in the PoD curve determination experiments. Besides measurements on flawed specimens, unflawed specimens should also be inspected to provide data for the estimation of a false positive rate. It is a rule of thumb [30] to provide a ratio of unflawed to flawed specimens of 3 to 1.

To collect data for the PoD derivation for the developed computer vision system, 100 images are selected among the available 16000 on which the actual crack length was marked (for example as in Figure 3a). The selection contains some of the images used to train the neural network with identifiable crack edges. Images in which a crack is not visible, but markings written by the inspector indicate the presence of a crack, are also included into the selection. Additionally, 300 images not containing any signs of crack are added.

Since the automatic crack detection algorithm proposed in this study outputs a segmentation map, it is necessary to choose a method to transfer the segmentation map into a single scalar or binary value that would represent an NDE output. This is necessary to allow determination of the PoD curve according to standard approaches [28, 29, 30]. It is decided to use the number of crack pixels in the output segmentation map as a quantitative output of the automatic visual crack detection algorithm. Another approach would be to calculate the crack length from the segmentation maps. However, this approach is not employed in this study, because this would require an additional post-processing step and information about distance between the crack and the camera in the moment when the image is taken.

Such signal responses for 100 images containing cracks and 300 images without cracks are plotted in Figure 4. The number of pixels classified as a crack is taken regardless of the pixel positions relative to each other on the segmentation maps. However, depending on the application case of the automatic crack detection algorithm, false positive pixels of the segmentation maps may be treated differently. As explained in Section 2.2, images of cracks often contain crack-like elements that can be misrecognized by the neural network as a crack. These false positive segmentations may or may not be included in the total number of crack pixels that contributes to the signal response, when used for the PoD determination. During the PoD curve determination the effect of the false positive segmentations on the algorithm signal response should be the same as during actual application. In this study, two application cases are considered. In the first application case the automatic visual crack detection algorithm is designed to be used as a "black box" that takes as input an image of a bridge and gives the decision about whether a crack is detected or not. In this application case, the false positive segmentations are added to the algorithm signal response and the same should be done when conducting measurements for the PoD determination. For this application case a PoD curve affected by false positive segmentations. In the second application case, the automatic visual crack detection algorithm is applied to take as input an image of a bridge and to output a segmentation map. To generate PoD curve for this application case it would make sense not to add false positive pixels to the signal response during the PoD experiments, because this would better represent conditions of this application case. For this case a PoD curve not affected by false positive segmentations is generated.

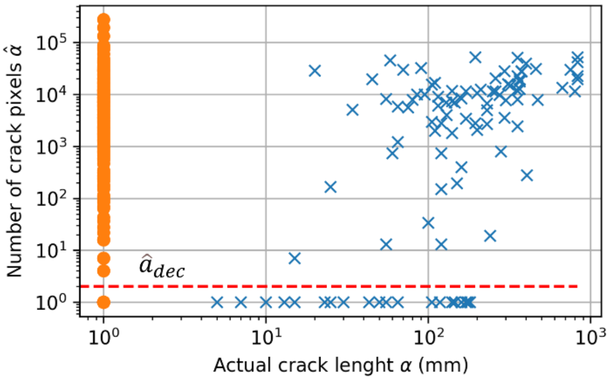

Figure 4 The number of pixels marked as crack by the neural network plotted against the actual crack length. Blue crosses represent measurements on images with cracks. Orange dots show the measurements on images without cracks. The dashed red line indicates a threshold decision level.

Two approaches for PoD curve determination exist based on the nature of the output of an NDE method [28, 29, 30]. The "a versus $\hat{a}$" approach is often applied to NDE methods that give a quantitative signal response "$\hat{a}$"(here referring to the number of pixels) correlated with the actual flaw size "a" that is being measured(in this case the length in mm). In contrast, the "hit/miss" approach deals with a binary output of an NDE method (flaw detected/not detected). Since the number of pixels classified as a crack is chosen as the crack detection system signal response, the "a versus $\hat{a}$" method for PoD curve calculation is first considered for this study. However, the "a versus $\hat{a}$" PoD curve calculation approach is built upon the requirement of data homoscedasticity, which implies uniform and normal scatter of measurement noise irrespective of the crack length. The data obtained in this study do not satisfy this requirement. This can be observed in Figure 4, where the horizontal scatter of the blue crosses varies depending on the crack length. So, the signal response data is further transformed into "hit/miss" data by applying a decision threshold value $\hat{a}_{dec}$. The crack is detected if the signal response (number of pixels marked as a crack) is above $\hat{a}_{dec}$. The produced "hit/miss" data affected by false positive segmentations is shown in Figure 5, where each blue cross represents a single measurement applied to one of the 100 selected images, with 1 for a detected crack (hit) and 0 for a crack being missed (miss). Similarly, Figure 6 shows the "hit/miss" data not affected by false positive segmentations.

Furthermore, a mathematical model is chosen and fitted to the available data [28,29,30]. A generalized linear model is selected with a logistic link function:

$$PoD(a|\theta_0,\theta_1) = \frac{\exp\left(\theta_0 + \theta_1 \cdot \log(a)\right)}{1 + \exp\left(\theta_0 + \theta_1 \cdot \log(a)\right)} \qquad (3)$$

where θ_0 and θ_1 are parameters that have to be fit to the data and a is a variable representing the actual crack length. In words, the $PoD(a|\theta_0,\theta_1)$ equals to the probability of detection of a crack, given the logarithmic crack length $\log(a)$ as covariate values in the logistic regression.

A maximum likelihood estimate (MLE) method is used for setting the parameters θ_0 and θ_1, where the log-likelihood $\log(L(\theta_0,\theta_1|a))$ of the model is calculated as in [28]:

$$L(\theta_0,\theta_1|a) = \prod_{i=1}^{N} PoD(a_i|\theta_0,\theta_1)^{Z_i} \cdot \left(1 - PoD(a_i|\theta_0,\theta_1)\right)^{1-Z_i} \Rightarrow$$

$$\log\left(L(\theta_0,\theta_1|a)\right) = \sum_{i=1}^{N} Z_i \cdot \log\left(PoD(a_i|\theta_0,\theta_1)\right) + (1-Z_i)\cdot \log\left(1 - PoD(a_i|\theta_0,\theta_1)\right) \qquad (4)$$

with Z_i being the "hit/miss" result of crack detection in the i −th image, and a_i the actual crack length in image $i \in \{1,\dots,N\}$ that is collected in $a = (a_i)_{i=1}^{N}$. In the experiments, $N = 100$ represents the number of images with cracks used for the PoD experiments. The optimization of θ_0 and θ_1 is done using the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm implemented in the scipy library of the Python programming language.

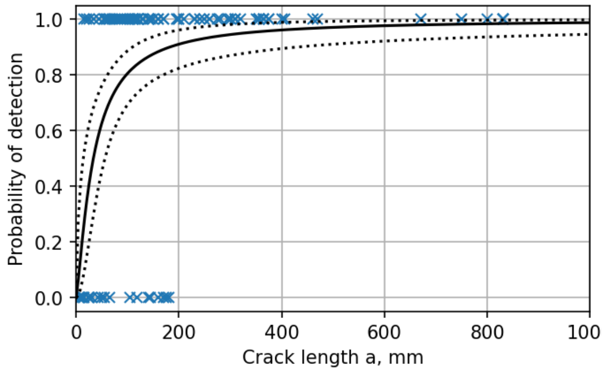

Figure 5 Hit/miss data affected by the false positive segmentations produced from the described measurement and the PoD curve (solid line). The upper and lower bounds of the 95% confidence interval of the PoD curve are shown with dotted lines.

The bounds of the 95% confidence interval of the PoD curve are calculated using the log-likelihood ratio test as described in [29]. Each combination of non-optimal parameters θ_0, θ_1 has a non-maximum likelihood value that can be calculated by the equation above. These parameters can be compared to maximum likelihood by the log-likelihood ratio (also called Wilk's likelihood ratio):

$$W(\theta_0,\theta_1|a) = -2 \cdot \log\left(\frac{L(\theta_0,\theta_1|a)}{L(\theta_0^{opt},\theta_1^{opt}|a)}\right) \sim\chi_1^2 \qquad (5)$$

where $\theta_0^{opt},\theta_1^{opt}$ are the parameters that give maximum likelihood, i.e. they are the maximum likelihood estimators for the parameters θ_0,θ_1. The log-likelihood ratio follows the χ_1^2 distribution under regularity assumptions and asymptotic analysis [31, 32]. It should be noted that in this hypothesis the W statistic is independent of N, which makes sense in view of the fraction in the logarithm. Therefore, parameters that are inside 95% confidence interval are enclosed between parameter combinations that provide the log-likelihood ratio W equal to a table constant of the χ_1^2 distribution: $\chi^2 = 3.841$[33](degrees of freedom are equal to 1 according to [30]). The confidence bounds calculation method is described in more detail in [30].

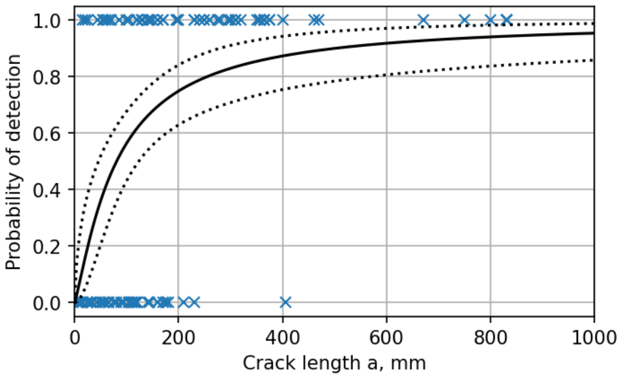

Figure 6 Hit/miss data *not* affected by the false positive segmentations produced from the described measurement and the PoD curve (solid line). The upper and lower bounds of the 95% confidence interval of the PoD curve are shown with dotted lines.

Each combination of θ_0,θ_1 within the region of 95% confidence gives a non-optimal PoD curve. An envelope of these non-optimal PoD curves provides the bounds of 95% confidence interval, which are plotted with dotted lines in Figure 5 and Figure 6. In Figure 5, the PoD affected by false positive segmentations is depicted, whereas Figure 6 shows the PoD *not* affected by false positive segmentations.

Finally, the false positive rate FPR is computed using the Clopper-Pearson method as described in [30]:

$$FPR = \left\{1 + \frac{n-x}{(x+1)\cdot F_{(1-a,\ 2x+2,\ 2n-2x)}}\right\}^{-1} \qquad (6)$$

where x is the number of false calls, n is the total number of measurements of the unflawed specimens, $F_{(1-a,\ 2x+2,\ 2n-2x)}$ is the F-statistics with degrees of freedom are equal to $2x+2$ and $2n-2x$, and level of significance equal to $1-\alpha$ with α being a required confidence level.

4 Results and discussion

Figure 7 compares the PoD curves for the automatic crack detection on images developed in this study with the PoD curves for regular visual inspection from Connor et al. [8], shown with the black dashed line, and from DNVGL-RP-C210 [34] (moderate access case), shown with the grey dashed line. It should be noted that in the study of Connor et al. [8], the PoD curve was calculated from data collected from accurately designed experiments providing realistic conditions for inspectors to inspect different components of steel bridge structures. Meanwhile, the data from DNVGL-RP-C210 are based solely on expert judgment and no experimental data were used [34]. Table 1 shows important points of the PoD curves providing a tool for simpler comparison. The symbols a_{50} and a_{90} show a crack length that will be detected with 50% and 90% probability respectively. The quantity denoted by $a_{90/95}$ represents an estimate of the length of a crack that is expected to be detected with 90% probability, based on the lower limit of the 95% confidence interval for the PoD curve.

The data from Table 1 and Figure 7 show that crack detectability with the automatic visual crack detection algorithm that includes false positive segmentation proposed in this work is roughly on the same level as crack detectability with regular visual inspection. The derived PoD curve affected by the false positive segmentations for the developed automatic crack detection algorithm is comparable to the one provided by DNVGL-RP-C210 for crack length values below 25 mm and performs better for bigger cracks. Compared to the PoD curve provided by Connor et al., the PoD curve affected by the false positive segmentations derived in this study has a lower probability of detection for any crack. The PoD curve for automatic visual crack detection that is not affected by false positive segmentations shows the lowest probability of detection for the entire range of crack length. The maximum difference between the determined PoD curve affected by false positive segmentations and the PoD curve not affected by false positive segmentations reaches almost 25% at a 100 mm crack length.

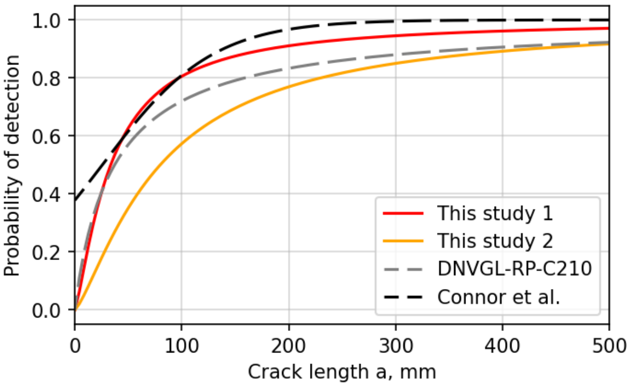

Figure 7 Comparison of PoD curves. "This study 1" refers to the PoD affected by false positive segmentations and "This study 2" to the PoD *not* affected by false positive segmentations.

The main limitation of the developed automatic visual inspection algorithm is a high false positive rate calculated using Equation (6). The proposed false positive reduction training strategy reduces the false positive rate from roughly 100% to 69%, but the problem of recognition of crack-like image features as a crack remains significant. This can be partly explained by the difference between the images on which the proposed neural network was trained and the images that were used for the false call rate estimation. As mentioned in Section 2.2, the available images were collected for inspection purposes and were not dedicated to the training of a robust neural network or to determine a PoD curve. Thus, images that contain cracks are often images of welded joints, while no-crack images used for false call rate estimation often contain structural elements different from those presented in crack images that often have more crack-like image elements. Another possible reason of the high false positive rate despite of the proposed false positive reduction training strategy, is the field of view of the neural network being restricted by the patch size. This problem also explains a low performance of the trained neural network when evaluated using the Dice score - DSC=0.6 according to Equation (1), when calculated on the patches generated from the validation dataset. Having this restricted field of view, it can be difficult to distinguish a crack-like image feature from a crack. This can be seen in Figure 8 where a patch with a crack and a patch with a dark line, that is similar in appearance to a crack but is not a crack, are shown. This hypothesis is supported by the fact that extending the proposed training strategy with a third stage similar to the second stage, does not lead to performance improvement. This restricted field of view can be solved by employing a combined crack detection approach shortly explained in section 2, where a neural network for localization has a global image view to identify the approximate crack location.

Table 1 PoD characteristics of regular and automatic visual crack detection. $a_{90/95}$ characteristic was not reported in the referenced works (N/A stands for "not available")

PoD source	a_{50}, (mm)	a_{90}, (mm)	$a_{90/95}$, (mm)
Connor, et al. [8]	25.6	138.7	N/A
DNVGL-RP-C210 [34]	37.38	369.7	N/A
Automatic crack detection (this study 1, affected by false positive)	33.2	182.8	427.9
Automatic crack detection (this study 2, not affected by false positive)	78.9	427.9	1254.5

Another possible solution of this restricted field of view is to use a neural network with an attention mechanism, such as a transformer neural network, which allows to use global image information to segment a single local image patch. However, the usage of the global image information can bring another problem if used with the collected dataset: Images that contain cracks often also contain crack marks made by an inspector, for example as on Figure 3a. If a neural network that employs global image information is trained on such images, it can learn to rely on these markings for crack detection. This can reduce the performance of the crack detection algorithm for practical use, because in practice it is necessary to detect cracks which are not detected by an inspector in advance and which do not have any markings around it.

The images used in this study provide a limitation to the conclusion about the performance of possible automatic visual inspection systems. First of all, the used images were taken by inspectors and not by an image acquisition setup such as an UAV, as would be the case with a fully automatic visual NDE. If the camera to take images for automatic crack detection is installed on an UAV, structural elements and fatigue cracks may have a different appearance on images, leading to different segmentation outcomes.

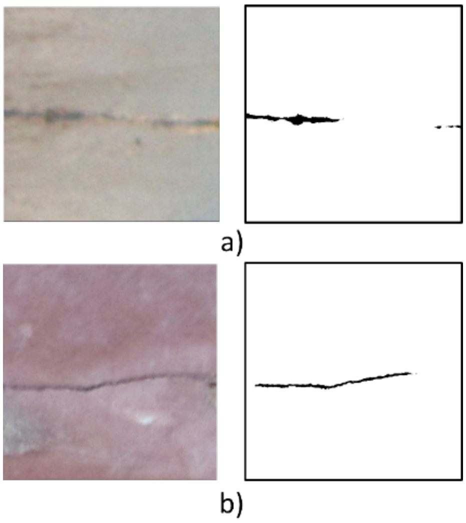

Figure 8 Background patches (512x512) and their segmentation maps obtained by the encoder-decoder network trained with the proposed two stage strategy: a) Background patch with a crack-like feature; b) Crack patch

A more sophisticated study would require images collected in a realistic environment that better represent conditions of automatic inspection. For example, if the PoD curve for automatic inspection with UAV is to be determined, then images collected using UAV are required. Moreover, in [8] different PoD curves were estimated for different structural components of a bridge. Thus, for a comprehensive study of automatic visual inspection, more images should be collected in a systematic way, capturing different structural elements and a variety of conditions, e.g. with and without corrosion.

In this study, the probability of detection is plotted as a function of the actual crack length as is usually done for visual inspection. This is done with the aim of comparing regular with automatic visual crack detection. However, for designing automatic visual inspection systems and procedures, it might be more informative to provide the probability of detection as a function of the crack length measured in pixels. This would better characterize the crack visibility and detectability on the image. Plotting the probability of detection as a function of a crack length in pixels would allow a more deliberate selection of a camera resolution for automatic visual NDE and the distance at which this camera should be held from the inspected surface to achieve the desired crack detectability.

Conclusions

An algorithm based on an encoder-decoder deep neural network for the segmentation of images of cracks in steel bridge structures is proposed. For reduction of false positive crack detection, a two-stage training strategy is designed to train a modified U-net neural network architecture.

Furthermore, the probability of detection curve is calculated for the proposed automatic crack detection algorithm, first, to check the feasibility of probability of detection curve determination for such an algorithm, and second, to compare its performance with that of regular visual inspection methods.

It can be concluded that an automatic crack detection system can augment current inspection practices, since its crack detection performance is comparable with that of regular visual inspection, while the frequency of automatic visual inspection could be increased thanks to lower costs. However, a high false positive rate provides the main limitation for the usage of the proposed method. Thus, the development of a robust automatic crack detection system requires further reduction of the high number of false positives which will be investigated in future work using more sophisticated geometric neural network architectures.

Acknowledgement

The authors would like to thank the Dutch bridge infrastructure owners "ProRail" and "Rijkswaterstaat", and "Nebest" engineering company for their support. The research is primarily funded by the Eindhoven Artificial Intelligence Systems Institute, and partly by the Dutch Foundation of Science NWO (Geometric learning for Image Analysis, VI.C 202- 031). Also, the authors would like to thank Gautam Pai for his help and advice related to the neural network implementation.

References

[1] Georgiou, G. A. (2007) *PoD curves, their derivation, applications and limitations*. Insight-Non-Destructive Testing and Condition Monitoring, 49(7), p. 409-414.

[2] Righiniotis, T. D. (2006) *A comparative study of fatigue inspection methods*. Journal of Constructional Steel Research, 62(4), p. 352-358.

[3] Kwon, K.; Frangopol, D. M. (2011) *Bridge fatigue assessment and management using reliability-based crack growth and probability of detection models*. Probabilistic Engineering Mechanics, 26(3), 471-480.

[4] Maljaars, J.; Vrouwenvelder, A. C. W. M. (2014) *Probabilistic fatigue life updating accounting for inspections of multiple critical locations.* International journal of fatigue, 68, p. 24-37.

[5] Leonetti, D.; Maljaars, J.; Snijder, H. H. (2021, September) *Influence of inspection on the safety of fatigue loaded welded cruciform steel joints–comparison of simplified and advanced crack growth models*. IABSE CONGRESS GHENT 2021: Structural Engineering for Future Societal Needs. International Association for Bridge and Structural Engineering, Ghent, pp. 1546-1554.

[6] Maljaars, J.; Leonetti, D.; Hashemi, B.; Snijder, H. B. (2022) *Systematic derivation of safety factors for the fatigue design of steel bridges*. Structural Safety, 97, 102229.

[7] Rossow, M. (2012) *FHWA Bridge Inspector's Manual*.

[8] Connor, R. J.; Campbell, L. E.; Snyder, L.; Whitehead, J. M.; Lloyd, J. B. (2019) *Probability of Detection Study for Visual Inspection of Steel Bridges*. Full Project Report (No. FHWA/IN/JTRP-2019/22). Indiana. Dept. of Transportation.

[9] Agdas, D.; Rice, J. A.; Martinez, J. R.; Lasa, I. R. (2016) *Comparison of visual inspection and structural-health monitoring as bridge condition assessment methods*. Journal of Performance of Constructed Facilities, 30(3).

[10] Hamishebahar, Y.; Guan, H.; So, S.; Jo, J. (2022) *A comprehensive review of deep learning-based crack detection approaches*. Applied Sciences, 12(3), 1374.

[11] Dorafshan, S.; Thomas, R. J.; Maguire, M. (2018) *Fatigue crack detection using unmanned aerial systems in fracture critical inspection of steel bridges*. Journal of bridge engineering, 23(10), 04018078.

[12] Lin, J. J;, Ibrahim, A.; Sarwade, S.; Golparvar-Fard, M. (2021*) Bridge inspection with aerial robots: Automating the entire pipeline of visual data capture, 3D mapping, defect detection, analysis, and reporting.* Journal of Computing in Civil Engineering, 35(2), 04020064.

[13] Chu, H.; Wang, W.; Deng, L. (2022) *Tiny-Crack-Net: A multiscale feature fusion network with attention mechanisms for segmentation of tiny cracks.* Computer-Aided Civil and Infrastructure Engineering, 37(14), 1914-1931.

[14] Jo, B. W.; Lee, Y. S.; Kim, J. H.; Yoon, K. W. (2018) *A review of advanced bridge inspection technologies based on robotic systems and image processing.* International Journal of Contents, 14(3).

[15] Li, H.; Wang, W.; Wang, M.; Li, L.; Vimlund, V. (2022) *A review of deep learning methods for pixel-level crack detection*. Journal of Traffic and Transportation Engineering (English Edition), 9(6), p. 945-968.

[16] He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. (2017) *Mask R-CNN.* Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 2961-2969.

[17] Redmon, J.; Farhadi, A. (2018) *YOLOv3: An Incremental Improvement.*arxiv:1804.02767

[18] Ronneberger, O.; Fischer, P.; Brox, T. (2015) *U-net: Convolutional networks for biomedical image segmentation*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany: Springer International Publishing, pp. 234-241.

[19] Han, Q.; Liu, X.; & Xu, J. (2022) *Detection and location of steel structure surface cracks based on unmanned aerial vehicle images*. Journal of Building Engineering, 50, 104098.

[20] Dong, C.; Li, L.; Yan, J.; Zhang, Z.; Pan, H.; Catbas, F. N. (2021) *Pixel-level fatigue crack segmentation in large-scale images of steel structures using an encoder–decoder network.* Sensors, 21(12), 4135.

[21] Goodfellow, I.; Bengio, Y.; Courville, A. (2016) *Deep learning*. MIT press.

[22] Kompanets, A. (2023) Crack segmentation tool [online]. https://github.com/akomp22/crack-segmentation-tool.git [accessed on: 30 March. 2023]

[23] Kompanets, A.; Duits, R.; Leonetti, D.; van den Berg, N.; Snijder, H. B (2023) *Segmentation tool for images of cracks*. Proceedings of the 20th International Conference on Computing in Civil and Building Engineering (accepted for publication)

[24] Duits, R.; Meesters, S. P.; Mirebeau, J. M.; Portegies, J. M. (2018). *Optimal paths for variants of the 2D and 3D Reeds–Shepp car with applications in image analysis*. Journal of Mathematical Imaging and Vision, 60, p. 816-848.

[25] Tong, T.; Lin, J.; Hua, J.; Gao, F.; Zhang, H. (2021) *Crack identification for bridge condition monitoring using deep convolutional networks trained with a feedback-update strategy*. Maintenance, Reliability and Condition Monitoring, 1(2), p. 37-51.

[26] Milletari, F.; Navab, N.; Ahmadi, S. A. (2016, October) *V-net: Fully convolutional neural networks for volumetric medical image segmentation.* 2016 fourth international conference on 3D vision (3DV), pp. 565-571

[27] Salehi, S. S. M.; Erdogmus, D.; Gholipour, A. (2017) *Tversky loss function for image segmentation using 3D fully convolutional deep networks*. Machine Learning in Medical Imaging: 8th International Workshop, MLMI 2017, Held in Conjunction with MICCAI 2017, Quebec City, QC, Canada: Springer International Publishing, pp. 379-387

[28] Vicki, E. P. (1989) *Introduction to Quantitative Nondestructive Evaluation*. ASM metals handbook: nondestructive evaluation and quality control. American Society of Metals International ,17, pp. 689–701.

[29] US Department of Defense. (2009) MIL-HDBK-1823A-Nondestructive Evaluation System Reliability Assessment.

[30] Gandossi, L.; Annis, C. (2010) *Probability of detection curves: Statistical best-practices*. ENIQ report, 41.

[31] Pawitan, Y. (2013) *In all likelihood : statistical modelling and inference using likelihood*. Oxford University Press.

[32] Wilks, S. S. (1938*) The Large-Sample Distribution of the Likelihood Ratio for Testing Composite Hypotheses*. The Annals of Mathematical Statistics, 9(1), p. 60–62. http://www.jstor.org/stable/2957648

[33] Engineering tables/Chi-squered distribution [online]. Engineering Tables/Chi-Squared Distribution - Wikibooks, open books for an open world [accessed on: 5 April. 2023]

[34] DNVGL-RP-C210 (2015) *Probabilistic methods for planning of inspection for fatigue cracks in offshore structures* (DNV GL). Oslo, Norway.

[35] Carass, A.; Roy, S.; Gherman, A.; Reinhold, J. C.; Jesson, A.; Arbel, T.; Maier, O.; Handels, H.; Ghafoorian, M.; Platel, B.; Birenbaum, A.; Greenspan, H.; Pham, D. L.; Crainiceanu, C. M.; Calabresi, P. A.; Prince, J. L.; Roncal, W. R. G.; Shinohara, R. T.; Oguz, I. (2020) *Evaluating White Matter Lesion Segmentations with Refined Sørensen-Dice Analysis*. Scientific reports, 10(1), 8242.