ADAPTATION OF DEEPLAB V3+ FOR DAMAGE DETECTION ON PORT INFRASTRUCTURE IMAGERY

M. Scherff[1], F. Hake[1], H. Alkhatib[1]

[1] Geodetic Institute, Leibniz Universität Hannover, Germany - scherff,hake,alkhatib@gih.uni-hannover.de

KEY WORDS: Image segmentation, Damage Detection, Deep Learning, Optimization, Supervised, Weakly Supervised

ABSTRACT:

Regular inspection and maintenance of infrastructure facilities are crucial to ensure their functionality and safety for users. However, current inspection methods are labor-intensive and can vary depending on the inspector. To improve this process, modern sensor systems and machine learning algorithms can be deployed to detect defects based on rapidly acquired data, resulting in lower downtime. A quality-controlled processing chain allows to provide hence informed uncertainty assessments to inspection operators. In this study, we present several Deeplab V3+ models optimized to predict corroded segments of the quay wall at JadeWeserPort, Germany, which is a dataset from the 3D HydroMapper research project. Our models achieve generally high accuracy in detecting this damage type. Therefore, we examine the use of a Region Growing-based weakly supervised approach to efficiently extend our model to other common types in the future. This approach achieves about 90 % of the results compared to corresponding fully supervised networks, of which a ResNet-50 variant peaks at 55.6 % Intersection-over-Union regarding the test set’s corrosion class.

1. INTRODUCTION

Infrastructure buildings must meet high construction standards and require regular maintenance throughout their complete life cycle to ensure safe and reliable usage for its users like pedestrians, workers or even machines. Scheduled inspections are performed to detect emerging defects that could affect the facility’s functionality and stability, supplemented by additional inspections as needed. National legislation, inspired by norm standards, typically regulates this process. Commissioned inspection companies compile reports detailing damaged parts or sections and providing a complete infrastructure assessment, allowing for informed maintenance decisions.

Besides commonly known infrastructure objects as bridges or tunnels, this concept has also to be applied for port facilities or similar building objects such as watergates or canals. However, the traditional approach of performing inspections by hand, with only basic tools for documentation, causes significant downtime for the facility, especially in underwater sections where divers face complicated visual conditions. This downtime leads to high costs for the facility operator. To shorten this process step, an automated inspection process should be introduced, which rely on the capabilities of today’s high accurate sensors. An example of such capabilities can be found within the 3D HydroMapper research project, in which (Hesse et al., 2019) developed a swimming multi-sensor system (MSS) sensing the environment below and above the water surface. By that approach, the complete facility can be captured within several minutes or a few hours depending on its size, allowing to detect for instance regular appearing corrosion spots in the so-called splash zone more objectively afterwards. Therefore, various software systems can assist building inspectors digitally by providing data-driven predictions and optionally uncertainty information for damaged areas in order to assess the infrastructure knowledgeably.

In this study, we aim to enhance the damage detection process by adapting the Deeplab V3+ segmentation network (Chen et al., 2018) to analyze an image corrosion dataset obtained from the 3D HydroMapper project. Initially, we establish a baseline model, and explore different combinations of backbone networks and loss functions to create a robust feature-rich network. Subsequently, we conducted an extensive performance analysis and examined the impact of introducing weakly supervised ground truth images obtained through the Region Growing algorithm during the training process.

The structure of the paper is as follows: Chapter 2 gives a brief overview on related studies to damage detection of infrastructure buildings, with a focus on corrosion investigations. In Chapter 3, we describe the methodology used for the dataset acquisition and preparation. Chapter 4 presents the process of obtaining the best performing segmentation model in detail. Subsequently, we provide and discuss the evaluation results of the validation and test sets. Finally, Chapter 6 presents the conclusion and an outlook of further investigations.

2. RELATED WORK

The field of infrastructure inspection can nowadays be approached using a wide variety of sensors or tools. Among these, cameras are an excellent acquisition instrument that can effectively examine building surfaces due to their universal and simple deployment. Because of the progress achieved in image processing and computer vision, especially in the last decade, early damage detection became a feasible goal. This particular task has been investigated recently i. a. by (Duy et al., 2020), (Katsamenis et al., 2020) and (Tanveer et al., 2022). They were able to detect different damage types to a certain extent by means of Deep Learning (DL) segmentation models on concrete and metal surfaces, respectively. However, some damage types are more recognizable than others due to their colour, shape or size. While corroded segments can be unambiguously distinguished from the concrete quay wall and background in general at a certain severity level, their shape and size can vary significantly, which may lead to erroneous detections. On the other hand, for instance cracks can have more similar properties overall, but individual examples can be overlooked due to their small width

or low resolution. Nevertheless, DL models like Convolutional Neural Networks (CNN) are currently the dominant objective in this research field and have a high potential to classify, detect, or even segment these damaged areas correctly due to their self-learning feature capabilities. Hence, this image processing strategy is a promising approach to digitally assist the inspection operator effectively.

In the study by (Duy et al., 2020), corroded areas on electric poles are detected using several architectural modified segmentation networks ranging from hundred thousands to ten millions trainable parameters. Three models were evaluated with and without prior background removal, achieving high Intersection-over-Union (IoU) scores for the pole and background class, but struggled to detect corrosion reliably. To overcome this limitation, the dataset was extended with a focus on the corrosion class, and the Mask R-CNN architecture (He et al., 2017) was investigated to partition these segments based on their severity level. But this network type was already examined with common CNNs by (Katsamenis et al., 2020) in the same year achieving 71 % F1 score on another corrosion dataset. In their experiments, Mask R-CNN outperforms the other candidates significantly. Their purposely developed boundary refinement approach, a data projection method dividing the model prediction in confident and fuzzy regions, effects the quantified results of this model barely. In addition, (Tanveer et al., 2022) evaluated the performance of various low parameter segmentation CNNs compared to the Deeplab V3+ architecture with ResNet as backbone network. The group trained these on a concrete dataset containing the damage types cracks, efflorescence, rebar exposure and spalling and found on average a relative maximum discrepancy of around 17 % in terms of F1 score to the reference. All these groups use Cross-entropy loss for the classification purpose in model training and no extensive optimization or evaluation was performed. Moreover, only a single model instance was learned, which limits the uncertainty investigation. Here comes (Nash et al., 2022) into play. This group trained different models of modified HRNetV2 (Sun et al., 2019) networks by Variational Inference, Monte Carlo Dropout and an Ensemble method on a further corrosion dataset. These Bayesian transformed models were evaluated especially in terms of aleatoric and epistemic uncertainty. In our study, we combine these concepts to provide an informed overview of different modified Deeplab V3+ models trained on an optimal loss function that is selected from a pool of distribution- and region-based error terms using our self-labeled port dataset. In contrast, (Hake et al., 2023b) focused only on single trained model instances and restricted architecture changes with the smaller and differently structured 3D HydroMapper dataset.

The most precise network structure is iteratively trained on simulated weakly supervised label images generated once by a Region Growing algorithm (Adams and Bischof, 1994) before any model training begins. However, (Huang et al., 2018) proposed a more sophisticated approach by using this algorithm on seed cues generated by a classification network in their segmentation pipeline. Their Deep Seeded Region Growing method works hence on image-level label information. We aim to estimate still the expected performance decline for this straight-forward approach when the manual labeling effort is significantly reduced in our investigation. These findings will guide our approach to further datasets containing other damage types that will be fed to the same model architecture to support the infrastructure inspection process. It is currently intended to mark a small region in the damage segment instead of bordering it completely by a polygon if that procedure is proven successfully. Due to the nature of the current dataset, we cannot efficiently benefit from bounding box or class activation map approaches.

3. DATASET

3.1 Data acquisition

The dataset used in this work was specifically acquired within the 3D HydroMapper project to aid the development of AI-assisted software for future port inspections based on MSS data. This image sequence shows the quay wall of the JadeWeserPort in Wilhelmshaven (Germany) in an overlapping manner. We observe similar coverage of up to 80 % between subsequent captures. Occasionally, the perspective is adjusted during the acquisition process by flying closer to the wall. This provides a wide variety of real-world information for the detection models to be trained on. The complete dataset was recorded on a single day using a controlled drone equipped with a Canon EOS 5D Mark III reflex camera, hovering or flying over the water a few meters in front of the wall. For this project the focal length was fixed at 35 mm and the resolution of pre-processed images are either 5760 x 3840 or cropped to 4608 x 3456 pixels. The dataset consists of 1300 high-resolution digital images of the infrastructure, where the object pixel size varies approximately between 2 and 5 mm depending on the individual capture. Due to this, we can find fine or arising defects and achieve accurate results in terms of position and shape of the damaged segments. This allows us to retrace the growing process over time and modify the inspection period or maintenance measures accordingly to related predictions. The labeling process neglected damages besides corrosion, such as cracks, efflorescence, or spalling due to their seldom occurrence. Therefore, our trained models cannot detect them yet. However, (Hake et al., 2023a) have already demonstrated an AI-assisted approach to detect geometrical related damage types in heightfields extracted from point cloud data. Currently, we had images of an efflorescence-damaged watergate available, which were taken by means of a tripod at several locations within this facility. Hence, these are not suitable for the current inspection concept based on a swimming or consistent moving sensor platform.

3.2 Generate ground truth images

To train the supervised segmentation models, we required labeled images indicating which pixels or segments correspond to the corrosion class. Initially, our own Matlab program was used to obtain these ground truth images, which converts manually drawn polygons successively into a 8-bit black/white (b/w) image. Later on, we used the service of Supervise.ly (Supervisely contributors, 2023) to complete our currently considered dataset consisting of 84 image-label pairs. Thereon, we can incorporate model predictions directly into the labeling process. These image pairs are uniformly distributed over the quay wall, showing a big portion of the total infrastructure, because overlapping areas were avoided for the most part. The ground truth images are based on our knowledge in this field and the gained experience over the time of this procedure, but have not been validated yet by experts. However, due to investigations in handling label noise in the DL context especially by (Nash et al., 2019), particular for corrosion detection, we expect no significantly improved corrosion detection capabilities. Nonetheless, we can not ensure that the achieved metric scores listed in chapter 5 correspond to the true values, especially with the provided number of digits.

To adapt the Deeplab V3+ structure for the corrosion segmentation in the beginning by means of cross validation, and evaluate the performance of the various models methodically correct afterwards, we divide the created dataset randomly into six subsets à 14 images each. Four of them form the train set, and each of the remaining ones forms the validation and test sets, receptively. The validation set determines the optimal parameter state of any model based on the corresponding lowest epoch-wise loss value, and the test images act as independent evaluation measures. An example of the aforementioned image-label pairs is illustrated in Figure 1. As visualized by the ground truth image (black: damage-free/background, white: corrosion), corroded segments represent the minority class of this segmentation task. Because damage detection has to deal regularly with highly imbalanced datasets due to its underlying nature, models will by design favor the majority class to minimize the prediction error or maximize the total accuracy. To reorder the model's behaviour in terms of equal class detection abilities, data processing itself can be modified or the optimization process of the considered learning method has to be adjusted. In this study, we focus mainly on the second point by selecting more robust loss functions with (additionally) integrated class weights.

3.3 Simulating weakly supervised images

For the weakly supervised approach, we label the same images as before by simulating a starting region for each corroded segment. These regions are subsequently extended using the Region Growing algorithm. The quality of this process is evaluated using the original label images serving as reference. This procedure was chosen primarily because of the labor-intensive and time-consuming nature of manual labeling, which is especially true for the here investigated segmentation task. By using a brush annotation tool to mark only a small fraction or region, rather than labeling the entire segment pixel-accurately with a polygon, the process can be significantly sped up. Based on our own experience with the port dataset, labeling a single $\geq$ 16 megapixel image can take up to 30 minutes, even one with only a few damaged areas requires between five to ten minutes. Assuming that the suggested brush method reduces the manual effort by only 50 %, it can still result in time savings of 1-2 working days for the small image dataset we considered. This scales proportionally to the true labeling duration, which may vary between individual samples due to the number of damaged segments, their variable sizes and shapes. The subsequent Region Growing processing can be conducted after working hours to minimize utilization of the available computing capacity, or directly after labeling a single image to reduce the total processing time. On the other hand, this approach can also lead to a higher number of image and ground truth pairs in the same time period, resulting in a more robust trained segmentation network. However, a major disadvantage of the chosen algorithm in particular is that it is not suitable for all kinds of damage types or every segmentation class in general. For Region Growing to approximate fully supervised labels qualitatively, the true damage segments must have homogeneous colours and must clearly distinguish themselves from the background and surrounding classes. While this may be true on average for corrosion and efflorescence in this domain, it does not to concrete spalling, for example.

The version of Region Growing used here is based on standardizing each colour channel c independently to determine whether an adjacent pixel should be added to the growing start region. A pixel x_i is added if the maximum standardized absolute deviation from the expected colour is within a given threshold t. Equation 1 defines the deviation:

$$\Delta x_{max} = max\left(\left|\frac{x_i - \mu_c}{\sigma_c}\right|\right) \leq t \tag{1}$$

Here, Δx_{max} represents the deviation, μ_c and σ_c contain the channel-wise statistical quantities, and the max function selects the largest element of the standardized RGB vector $\in \mathbb{R}^{\#}$. The threshold is set to a strict value of 1.25 to prevent overgrowing damage segments, especially for this pile quay wall and to provide the segmentation network with mostly correctly classified pixels, even if they represent only a small fraction of the true damaged areas. We believe that it is better to add initial missing corrosion pixels along the way than to include a large number of incorrect background pixels in the weakly supervised damage segments. To counteract wrongly generated corrosion segments during training, we use Conditional Random Field (CRF) implementation[1] by (Krähenbühl and Koltun, 2012) in each iteration. The threshold parameter is not specifically tuned to optimize certain metrics based on the original labeled data but is selected by visually comparing different outcomes for a small set of images. At the end of the algorithm, we apply equation 1 once again to every pixel within the grown region to possibly remove outliers. The starting segments for Region Growing are created by a randomly orientated ellipse that satisfies a certain overlapping condition with the reference label. The initial set semi-axes are dependent on the segment's size and are shrunken every time to the condition is not fulfilled. Using this method, the weak labels were evaluated in order to gain insight into the model's performance in the iterative training process. The initial weak labels achieved IoU scores of 92.4 and 36.2 % for the background/undamaged and corrosion class, respectively, with respect to the manual images. In Figure 1, we present the original RGB and both labeled ground truth images for a visually appealing sample of the test dataset.

4. SEGMENTATION NETWORKS

4.1 Baseline model

The Deeplab V3+ architecture is widely recognized for its impressive segmentation capabilities, and its basic modules, such as spatial pyramid pooling (SPP) and encoder-decoder structures, which are broadly applied even outside the core AI research community. The network captures object boundaries to some extent by gradually upsampling concatenated low- and high-level feature maps, and detects variable-sized objects with ease thanks to the Atrous SPP (ASPP) module, which encodes multi-scale contextual information (Chen et al., 2018). Generally, object boundaries can be refined using post-processing steps, but in this work, we orient ourselves on feature attention mechanisms ((Azad et al., 2020), (Hsu et al., 2022), (Zeng et al., 2020)) within the encoder part to achieve this. Such mechanisms recalibrate the functional relationship represented by the model to rely more on meaningful feature extractions. We aim to mimic this behaviour by modifying the dilation rates within ASPP's convolutions. By backpropagating the prediction error through the network during training, the feature maps or responsible filter blocks should be modified accordingly to boost detection capabilities of certain spatial resolutions slightly. This

[1] https://github.com/lucasb-eyer/pydensecrf

Figure 1. Example of a cropped RGB image showing the east-facing sheet pile wall of the JadeWeserPort and corresponding weakly and manually supervised ground truth images (top to bottom).

approach is expected to be beneficial due to the limited variability of occuring damaged areas in the particular port images we use, and for further related datasets that will be captured by similar systems to (Hesse et al., 2019). In other words, the damaged areas on quay walls or buildings can be closely sensed with high-resolution, allowing the ASPP module, regradless of the damage size, to segment the border area of different classes with high quality by taking optimized local context into account.

To improve segmentation results, we investigate different dilation rates of the corresponding Atrous convolution operation. The combined pyramid pooled features regain spatial context when merged with low-level features from the feature extraction network (alias backbone), primarily consisting of object boundaries. In the decoder, two 4x upsampling operations reconstruct the original image size, and contiguous segments are produced. To prevent information loss in the network structure, we apply some convolutions to combat the simplicity of linear interpolation. We examine whether adding additional blocks (0, 1 and 2) in between the Upsampling layers can significantly improve the segmentation result. For the dilation rates we consider the single to triple amount of 3, 6 (default), 12 and 24 for the Atrous convolutions. These hyperparameters are tested with three different models: VGG-16 (Simonyan and Zisserman, 2014) (Batch normalizations included), ResNet-50 (He et al., 2015) (version 1.5), and EfficientNet V2 (Tan and Le, 2021) (Small). VGG-16 has the most basic structure due to its early development in 2014 and consists of around 16 million of the lowest number of trainable parameters. The other two models are built with more advanced concepts in mind, such as Residual blocks, Neural Architecture Search optimized model structure, and Stochastic Depth, but can still be considered as rather basic or small DL models, with approximately 23 and 20 million parameters, respectively. We individually modify the models such that the output stride is set to 16 by removing the fifth MaxPooling layer or setting the step size of the last strided convolution(s) to one. Furthermore, we freeze the initial layers up to Deeplab’s low-level feature output to preserve the extensively learned parameters from the ImageNet database (used weight version 1 according to the torchvision (Marcel and Rodriguez, 2010) documentation). While the purpose of this work is to assist inspection operators, fast inference times are not necessarily required. However, to prevent overfitting or long training duration for new acquired datasets in the future, we have not examined significantly larger or more complex backbones.

To train our models, we initially divide the image-label pairs in (non-)overlapping patches of 512x512 pixels. This approach allowed the models to be fed with rich semantic context that exceeded their output layer’s receptive field, enabling them to learn from different perspectives of cut-off corrosion segments located near the edges. We use a 25 % overlap for the train dataset but nothing is utilized otherwise. The available 24 GB VRAM of the GeForce RTX 3090 and later 4090 is fully utilized by setting the batch size to 8 (for the EfficientNet) or 12. For an efficient training advancement, the initial learning rate is determined to be 5e-4 and 1e-3, respectively, by the most negative slope of a learning rate finder curve.

To counteract the imbalance between undamaged and corroded areas, we use image augmentation methods such as affine transformation, colour jitter and Gaussian blur, with individual associated probabilities and class weights (1:1 for baseline determination to emphasize architecture changes effectively) within the different considered loss functions. To optimize the network architecture robustly, we employ k-fold Cross-Validation with the four aforementioned training subsets extensively. Each combination of dilation rates and number of additional convolutions is trained on three of these subsets and subsequently evaluated on the validation set to determine the optimal model parameter state based on the loss value. Through this procedure, the maximum number of epochs has never exceeded 30 for all types of model trainings.

In our experiments, we employ the combination of distribution- and region-based loss terms in the form of Categorical Cross-Entropy and IoU loss. To identify the optimal model structure, we calculate the total uncertainty (entropy of softmax output), accuracy, F1 and IoU scores based on the validation patches. The average uncertainty of this subset ranges from 0.06 to 0.12, while the total accuracy varies only between 95.2 and 96.1 %. The class-specific F1 and IoU scores reveal the details in the detection capabilities on the other hand. Overall, undamaged areas are recognized with more than 97 and 94 %, respectively, with deviations of around 1 % across all experiments. The most significant performance differences occur in the corrosion class. The F1 and IoU scores for this class vary by 8 % between the best and worst performing Deeplab network, with upper bound of 59 and 42 %, respectively. By counting the best performing models for both parameter sets individually, we determined that no additional convolution leads to a performance boost, and the standard dilation rates of 3 and 12 are almost equally useful for the segmentation task. Therefore, we changed the kernel spaces within the Atrous convolutions to 5, 10 and 20 to combine these ASPP blocks roughly. Thereby, we focus primarily on a local field of view in order to detect segment boundaries accurately, but find more deviating and rich features at the same time due to the variable dilation rates. To further improve the baseline model, we consider a wide range of modified loss functions with class weights, and evaluate all three backbone networks.

4.2 Training process of (weakly) supervised models

In the first stage of finetuning, we trained the modified Deeplab V3+ architecture with several pretrained backbones and overall five different loss terms on the complete training dataset. We evaluated them equally to the previous step to determine the optimal loss function for each feature extraction network. After that, we adjusted the hyperparameters related to the cost function to improve the model performance. We then used the separate testing set to independently assess the best performing Deeplab V3+ models among the feature extraction networks (s. chapter 5). The top architecture-loss pairs are then used to train the Deeplab model on the weakly supervised dataset in an iterative process, whereby the training and validation sets are created by the model predictions post-processed with a Potts model. We changed the standard deviations regarding pixel distance and RGB difference slightly compared to the default values listed at Pydense package's Github page (s. footnote 1) to account for the properties of our dataset. In particular, we set s_{xy} and s_{rgb} of the bilateral term to 40 and 10, respectively. This enables predicted corrosion segments to be extended more strictly based on the colour difference to nearby pixels, which is necessary for the corrosion segments onto the specific pile quay wall. We stooped when the IoU scores of the validation set do not improve compared to the last iteration. The initially trained models change the datasets extensively, leading to little overlapping segments. For instance, when the complete weakly supervised dataset is deployed, the first model achieves around 30 % IoU for the corrosion class. Subsequently, the other models benefit from more coherent samples post-processed by a CRF approach, gradually refining the damage segments. This entire procedure restricts the final model from learning self-produced label noise. We provided a detailed visualization of this segmentation approach with respect to the validation and test set in subsection 5.2.

All the models are in general trained using Stochastic Gradient Descent with Nesterov Momentum and L2 regularization, with associated parameters μ and λ set to 0.9 and 1e-5, respectively, in the Pytorch framework (Paszke et al., 2019). The learning rate is halved when the validation loss does not decrease for three epochs in a row, and Batch Normalization layer momentum starts at 0.2 and gradually decreases to help the non-frozen network section adapt quickly to the task. The explored loss functions include Categorical Cross-Entropy, IoU loss, their combination (compound loss), and the base variants modified by the Focal term or parameter $\gamma = 2$ that shifts the model's attention to hard-to-classify pixels (Lin et al., 2017). Additionally, class weights of 1:3 were added to each error term. We observed significant performance improvements with the Focal IoU loss networks. Further Deeplab models were examined by incrementally increasing the parameter up to 5 for each backbone individually. The aforementioned weakly supervised model was finally trained with ResNet-50 and Focal IoU loss with $\gamma = 4$, and the impact of uncertain ground truth images was investigated by varying their fraction within the dataset between 60 to 100 %. Manually labeled images remained unchanged during each of these training runs to simulate real conditions.

5. EVALUATION AND DISCUSSION

5.1 Supervised models

The performance of the baseline Deeplab V3+ model is evaluated on the dedicated validation dataset after training it with moderate basic backbones on various distribution- and region-based loss functions. Each image patch, which lacks surrounding context, is processed by the individual segmentation networks and analyzed quantitatively using metrics related to total uncertainty and confusion matrix. In this supervised context, its evident that VGG-16 performs significantly worse than the other feature extraction networks on average due to the lower output mapping capabilities. However, the default Focal IoU loss variant ($\gamma = 2$) is marginally ahead of the corresponding EfficientNet V2 network. Additionally, we observe that region-based exclusive cost functions exhibit up to one magnitude superior uncertainty scores across all models, indicating their superiority for segmentation tasks with imbalanced datasets. We report only the corrosion class-related F1 and IoU scores in detail in this step of model optimization, as the background detection rates are comparable to those in subsection 4.1 with differences of around 3 %. Given the imbalance of this particular dataset, the total accuracy values can be neglected completely. Overall, the remaining quantities for the different Deeplab models are visualized in Figure 2.

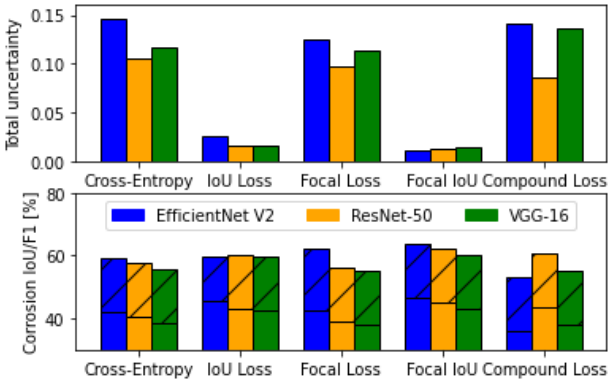

Figure 2. Total uncertainties and corrosion class related metrics of Deeplab V3+ models with respect to validation set consisting of different backbones and trained on various loss functions. The hatched bar sections indicate the difference between IoU and F1 score.

Due to the excellent predictive performance achieved using the Focal IoU loss, we conducted further experiments to optimize the γ parameter for each backbone network separately. The best-performing parameter value was then applied for the first time on the testing set. To obtain complete images instead of smaller patches for visualization or later applications, we used a blending method described in (Chevallier, 2017). This method constructs smooth and continuous segments across patch edges by means of a window function that combines several orientated model predictions. These original sized images were than analyzed using a metric designed to assess agreement with complete corroded segments in the ground truth data. This is particularly helpful because inspection operators are primarily interested in the presence of damaged areas at a certain location, rather than their exact shape and/or size. The resulting scores originate from False Positive and Negative corrosion pixels, and can therefore be understood as supportive for the detection task. To make it more comprehensible: the remaining percentages to 100 represent fully made-up or overlooked damage segments. These metrics are presented in Table 1, alongside common quality measures. In total, we trained five different segmentation network instances by randomly reinitializing the parameters (except the pretrained backbone) to determine the

aleatoric and epistemic uncertainties individually. The total uncertainty is derived by the entropy of the composed Deep Ensemble, while epistemic uncertainty is derived by mutual information, and aleatoric uncertainty is equivalent to their difference. Therefore, the computed result(s) may deviate slightly from the results the single model shown in Figure 2.

Table 1. Results of best Deeplab V3+ segmentation network with respect to test set

Metric	Score
Total uncertainty (aleatoric, epistemic)	0.03492 (0.03491, 1e-5)
Total accuracy [%]	95.7
F1 Score [%] (backg., corr.)	97.7, 71.5
Pixel-wise IoU [%] (backg., corr.)	95.5, 55.6
Corrosion supporting FP and FN [%]	80.6, 86.3

The determined uncertainties are in-line with the other Deep Ensembles in terms of their relative relation. On average, the individual Deeplab models among the frameworks vary by 1e-6 to 1e-5, while the images themselves are responsible for almost the complete amount. This implies that the pretrained backbones, combined with the available but general small dataset, do not hinder the segmentation networks from learning corrosion damage appropriately, as aleatoric uncertainty remains unaffected by more training images. However, this information does not provide a detailed overview of the corrosion predictions in the images themselves. Looking at the other scores, we see that the non-damaged or background areas are classified with an accuracy of 95 % or more. The model struggles with precisely identifying the corrosion areas, but the value is as good as (Katsamenis et al., 2020) found on their available dataset. Due to the domain difference, we conclude that the optimization steps were necessary to achieve this level of performance with the restricted network architecture and overall available resources.

To get insight into the prediction results and their uncertainties in relation to the manually labeled images, we provide colour-coded corrosion segments and greyscale images below for the same scene as shown in Figure 1. In the top image, green, blue and red colors represent True Positive, False Positive and False Negative detections, respectively, while the brightness level indicates regions with higher uncertainty in the bottom images. The individual greyscale values are computed by converting an image-level scaled float value to an 8-bit unsigned integer, ensuring the full spectrum is always occupied despite the significant differences indicated by the tabular values. This example confirms that the trained models usually differ within a narrow buffer around the manually labeled damage segments.

The trained model often misses or wrongly detects corrosion near the areas that we identified as damaged with high confidence manually. Our integrated metrics reflect this by showing values above 80 %. Inspection operators dealing with manual classification on a daily basis can benefit from this information, despite other metrics indicating lower performance. If we also provide them additionally for example with the uncertainty images, they can adjust the segment boundaries using a prospect software package as part of a digitally assisted procedure.

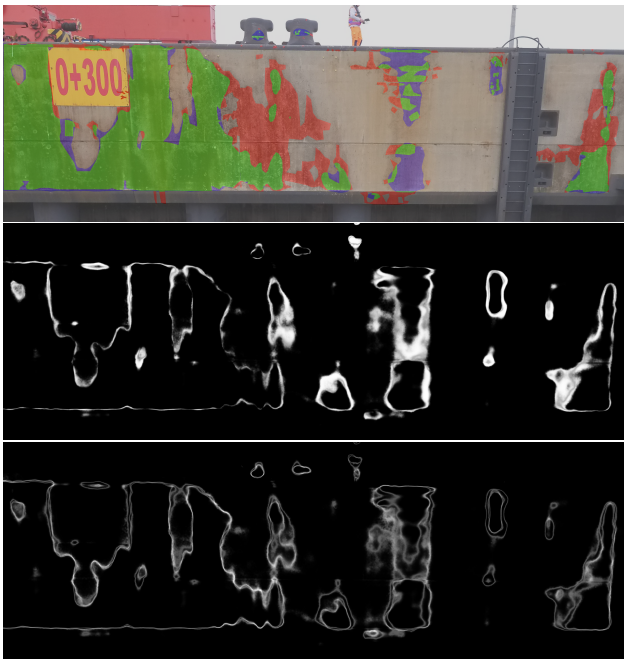

Figure 3. ResNet-50 Deeplab V3+ model’s cropped blending prediction with post-processed colour-coding according to corrosion segments from ground truth data (green: True Positive, blue: False Positive, red: False Negative). Aleatoric and epistemic uncertainty calculated from a related five instance Deep Ensemble network.

5.2 Weakly Supervised model

The best-performing segmentation network, Deeplab V3+ ResNet-50, in the supervision context is then demonstrated in our simple designed training pipeline. This pipeline starts with ground truth images generated from a strict Region Growing algorithm applied to the challenging JadeWeserPort dataset. Due to the lack of accurate label images over the complete iterative training process, performance degradation is inevitable. However, the time-consuming labeling effort is significantly reduced. This gives researchers the possibility to obtain a larger dataset, if necessary, in order to improve model’s generalization. In the following Figure 4, we show the performance development for different fractions of weakly label images used within the training process.

We draw here for the first time conclusions based on the agreement of blended patch predictions to the test set regarding the individual model training procedures. Especially the initial training phase can be highlighted, because the corresponding Region Growing labeled images achieved only 24.3 and 39.1 % in terms of corrosion IoU and F1 score, respectively. The final model of each training pipeline was determined by the highest IoU damage value of the individual evolved validation set, requiring 3, 2, 3, 3 and 2 iterations in descending order of the weakly label fraction. According to these results, the models iteratively trained on the complete weakly supervised dataset were the most successful, while the optimal Deeplab networks learnt from 70-90 % weak labels had a 5-10 % lower performance in terms of corrosion IoU. The dataset with the largest fraction of manual labels performed worst, with an additional 10 % drop in performance. This outcome was expected, as the strategy of finding the best model relys on monotonically increasing metric scores, which is equivalent to efficient conver-

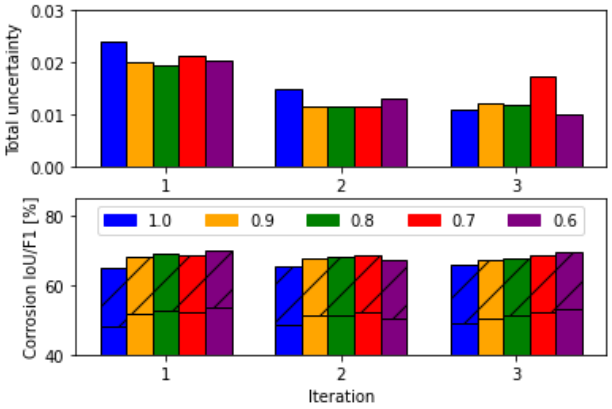

Figure 4. Total uncertainties and corrosion class related metrics of optimal Deeplab V3+ ResNet-50 with respect to test set trained on weakly supervised dataset. Different colours stand for the fraction of deployed weak labels. The hatched bar sections indicate the difference between IoU and F1 score.

gence. When an individual dataset contains inaccurate but obvious corrosion labels, the model can learn the underlying complexity more effectively. However, when a significant amount of the dataset consists of contradictory image pairs in terms of weak labels, there is no fast and straightforward solution. But the iteratively refined images may approximate the true damage segments step by step at different speeds. Hence, we can achieve supervised-like performance after several model trainings when a reasonable amount of accurate labels is available. However, we cannot measure such an advancement in general due to a lack of reference data. Nevertheless, we found a significant improvement between different models (with weak supervised fractions between 0.7 and 1.0) in terms of the labeled test set after a few iterations, as demonstrated in Figure 4. But how much labeling effort can be saved to achieve that? If we continue to calculate with the 50 % example from subsection 3.3, we would reduce the labeling effort for the complete training and validation dataset (70 images in total) by roughly 12 hours. For every accurately produced segmentation ground truth image taking approximately 10 - 15 more minutes, the model gains about 0.15 % IoU corrosion performance within an initial linear growing phase. Afterwards, the training may take more iterations than 3 to convergence, resulting in unpredictable performance capabilities, followed by a period in which a single model recognize weak labels confidently as outliers and take this into account for its parameter optimization automatically. Additionally, the determined total uncertainty over the test data is barely affected for the final models by the relative portion of manual labeled images. Therefore, we suggest that one should only label up to one third of small, imbalanced image datasets precisely to deploy the corresponding model in a short period of time. This will have a noticeable effect within the process chain and guarantees, based on our experiments, at least 90 % of the maximum achievable performance. This gap might be further closed by applying certain image augmentation methods and/or domain-specific loss functions.

In this paragraph, we present a detailed analysis of the performance of our best model on the test set, which provides insights into the potential of our method for real-world damage detection applications. We present the results using the same format as in the supervised setting, with Table 2 and Figure 5. Although most of the quantities in Deeplab model have

Table 2. Results of Deeplab V3+ segmentation network trained on 70 % weakly supervised images with respect to test set

Metric	Score
Total uncertainty (aleatoric, epistemic)	0.02248 (0.02248, 2e-6)
Total accuracy [%]	95.2
F1 Score [%] (backg., corr.)	97.4, 68.5
Pixel-wise IoU [%] (backg., corr.)	94.9, 52.1
Corrosion supporting FP and FN [%]	81.6, 75.0

only slightly changed compared to the fully supervised model, it tends to miss corrosion segments more often, resulting in a higher number of False Negative pixels (75.0 compared to 86.3 %). Conversely, the number of type 1 error for damaged segments decreases. These factors lead to relatively minor changes in the metric scores, but has a more visible impact on the prediction images themselves. Figure 5 illustrates the behaviour explicitly for the centrally located corrosion segments. In contrast, we observe finer, uncertain boundaries around the predicted corrosion segments, especially in aleatoric case, which is reflected in the lower scores in Table 2 compared to the fully supervised results. This difference can be explained by the iteratively trained models that learn from individual refined label images and build up confidence over time.

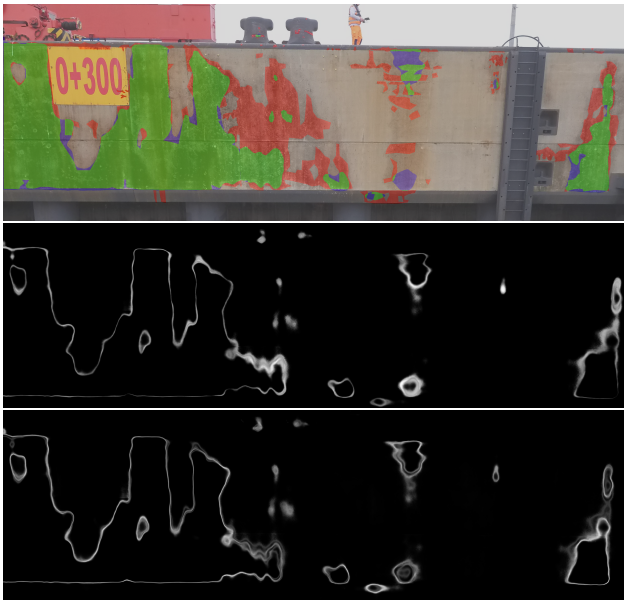

Figure 5. Cropped blending prediction by best ResNet-50 Deeplab V3+ model trained on 70 % weakly supervised images. Colour-coded in post-processing according to corrosion segments from ground truth data (green: True Positive, blue: False Positive, red: False Negative). Aleatoric and epistemic uncertainty calculated from an affiliated five instance Deep Ensemble network.

6. CONCLUSION

In this study, we developed a modified Deeplab V3+ model for segmentation of corrosion in infrastructure inspections. The model was optimized using 4-fold Cross-Validation on a corrosion dataset, and we found that the combination of ResNet-50 backbone and Focal IoU loss with $\gamma = 4$ (with integrated

class weights of 1:3) achieved optimal results for our imbalanced dataset. We restricted our optimization to moderate large feature extraction networks for efficient training on consumer-level hardware. Our DL model is designed to assist operators visually in future inspections, and we plan to retrain it on further datasets that contain also other types of damage. We also investigated an iterative weakly supervised approach based on Region Growing, which can reduce the manual effort required for ground truth labeling. We found that roughly one third of manual labels are sufficient to achieve around 90 % of the fully supervised detection capabilities. The produced Deeplab V3+ models are part of the automation process of infrastructure inspections to reduce on-site downtime and subjective assessments (for example under stress). In future research, we plan to evaluate the model on a wide range of infrastructure datasets and refine the weakly supervised approach possibly with existing concepts to train a more robust segmentation network. We also plan to provide possibly model predicted uncertainty maps to inspection operators for a justified decision.

Funding This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - NE 1453/5-1 and as part of the Research Training Group i.c.sens [RTG 2159].

REFERENCES

Adams, R., Bischof, L., 1994. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6), 641-647.

Azad, R., Asadi-Aghbolaghi, M., Fathy, M., Escalera, S., 2020. Attention deeplabv3+: Multi-level context attention mechanism for skin lesion segmentation. *ECCV 2020 Workshop on BioImage Computing*.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation.

Chevallier, G., 2017. Using a u-net for image segmentation, blending predicted patches smoothly is a must to please the human eye. `https://github.com/-Vooban/Smoothly-Blend-Image-Patches`.

Duy, L. D., Anh, N. T., Son, N. T., Tung, N. V., Duong, N. B., Khan, M. H. R., 2020. Deep learning in semantic segmentation of rust in images. *Proceedings of the 2020 9th International Conference on Software and Computer Applications*, ICSCA 2020, Association for Computing Machinery, New York, NY, USA, 129–132.

Hake, F., Lippmann, P., Alkhatib, H., Oettel, V., Neumann, I., 2023a. Automated damage detection for port structures using machine learning algorithms in heightfields.

Hake, F., Scherff, M., Neumann, I., Alkhatib, H., 2023b. Using semantic segmentation for the damage detection of port and marine infrastructures. Herbert Wichmann Verlag, accepted.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask r-cnn.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Deep residual learning for image recognition.

Hesse, C., Holste, K., Neumann, I., Hake, F., Alkhatib, H., Geist, M., Knaack, L., Scharr, C., 2019. 3D HydroMapper: Automatisierte 3D-Bauwerksaufnahme und Schadenserkennung unter Wasser für die Bauwerksinspektion und das Building Information Modelling. *Hydrographische Nachrichten*, 113, 26-29.

Hsu, C.-Y., Hu, R., Xiang, Y., Long, X., Li, Z., 2022. Improving the Deeplabv3+ Model with Attention Mechanisms Applied to Eye Detection and Segmentation. *Mathematics*, 10(15). https://www.mdpi.com/2227-7390/10/15/2597.

Huang, Z., Wang, X., Wang, J., Liu, W., Wang, J., 2018. Weakly-supervised semantic segmentation network with deep seeded region growing. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7014–7023.

Katsamenis, I., Protopapadakis, E., Doulamis, A., Doulamis, N., Voulodimos, A., 2020. Pixel-level corrosion detection on metal constructions by fusion of deep learning semantic and contour segmentation.

Krähenbühl, P., Koltun, V., 2012. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. https://arxiv.org/abs/1210.5644.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection.

Marcel, S., Rodriguez, Y., 2010. Torchvision the machine-vision package of torch. *Proceedings of the 18th ACM International Conference on Multimedia*, 1485-1488.

Nash, W. T., Powell, C. J., Drummond, T., Birbilis, N., 2019. Automated corrosion detection using crowd sourced training for deep learning.

Nash, W., Zheng, L., Birbilis, N., 2022. Deep learning corrosion detection with confidence. *npj Materials Degradation*, 6.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. Pytorch: An imperative style, high-performance deep learning library.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition.

Sun, K., Zhao, Y., Jiang, B., Cheng, T., Xiao, B., Liu, D., Mu, Y., Wang, X., Liu, W., Wang, J., 2019. High-resolution representations for labeling pixels and regions.

Supervisely contributors, 2023. Supervisely. `https://github.com/supervisely/supervisely`.

Tan, M., Le, Q. V., 2021. EfficientNetV2: Smaller Models and Faster Training. https://arxiv.org/abs/2104.00298.

Tanveer, M., Kim, B., Hong, J., Sim, S.-H., Cho, S., 2022. Comparative Study of Lightweight Deep Semantic Segmentation Models for Concrete Damage Detection. *Applied Sciences*, 12(24). https://www.mdpi.com/2076-3417/12/24/12786.

Zeng, H., Peng, S., Li, D., 2020. Deeplabv3+ semantic segmentation model based on feature cross attention mechanism. *Journal of Physics: Conference Series*, 1678, 012106.