# AI for Automatic Defects Detection on Large Civil Works: Our Adventure from the Database Creation to a First Efficient Pipeline

PIERRE STEPHAN[1], PAULINE LAVIRON-PETIT, SELIM BEN-ABDALLAH, THIERRY LACOUR[2], GUSTAVO MEDINA VASQUEZ[3] and QIUFENG ZHENG[4]

**ABSTRACT**

EDF, as a responsible operator, inspects and maintains the civil engineering structures of its nuclear power plants, which often exceed several tens of thousands of square meters in surface area. These inspections are carried out using drones, generating orthophotos of the surfaces. Manual analysis of these images to identify defects such as cracks, corroded bars, and spalling is costly and time-consuming. To address these issues, EDF is developing an AI-based tool for automated analysis to detect, locate, and measure these defects. The project includes a state-of-the-art review, creation of a training database, image preprocessing, data augmentation, neural network implementation, performance metrics selection, and optimization of the processing chain. The first results obtained and feedback on this image processing pipeline will be presented.

[1]Pierre Stephan, Pauline Laviron Petit, Sélim Ben-Abdallah, EDF Lab Chatou, 6 quai Watier, 78400 Chatou, France
[2]Thierry Lacour, EDF DTG, 12 rue Saint Sidoine, 69003 Lyon, France
[3]Gustavo Medina Vasquez, EDF UK R&D, 90 Whitfield Street, London, W1T 4EZ, UK
[4]Qiufeng Zheng, EDF R&D China, 12 F, Emperor Group Center, No. 12 ding, Jianguomenwai Avenue, Chaoyang District, Beijing 100022, PRC

## INTRODUCTION

Inspections of civil engineering structures, such as cooling towers, dams, and containments, are essential to ensure their integrity and safety. Traditionally, these inspections are carried out by civil engineering experts, often using drones to capture images of concrete surfaces. However, this process is costly, time-consuming, and prone to human error. Additionally, the quality of defect annotations, such as cracks and corrosion, can vary considerably, making it difficult to use this data to train artificial intelligence (AI) models.

Faced with these challenges, EDF set out to develop an automated solution for defect detection on concrete structures using AI-based algorithms. This study aimed to improve inspection efficiency and reduce associated costs. The main objective is to create an image processing pipeline capable of automatically detecting cracks on concrete surfaces using advanced semantic segmentation models.

In this study, we used the CrackSegFormer model, which combines a hierarchical encoder based on Transformers and a decoder consisting entirely of multilayer perceptrons (MLPs). This model was trained on images of cooling towers, annotated to include various types of defects. We also explored data pre-processing and augmentation techniques to improve annotation quality and model performance.

## PRELIMINARY WARNING

This article presents the first version of EDF's drone inspection image processing pipeline, focusing solely on automated crack detection. The objective is to share the applied approach and the acquired feedback, rather than presenting an optimized processing pipeline.

## METHODOLOGY

The methodology is illustrated in Figure 1. The first step is to create a database with high-quality annotations for segmentation, as existing open-source databases are not suitable. Next, the images are preprocessed to be formatted as required by the neural network, including data augmentation if necessary. The neural network is then trained with a portion of this database. Defects are predicted using the trained neural network. The performance of the predictions is evaluated against the ground truth database, although traditional criteria are often unsuitable for segmentation. Finally, the predictions are post-processed to extract useful values, such as the length and width of the cracks.

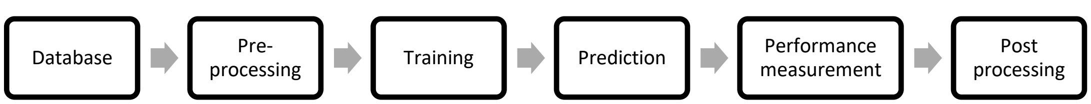



Figure 1. Proposed Pipeline

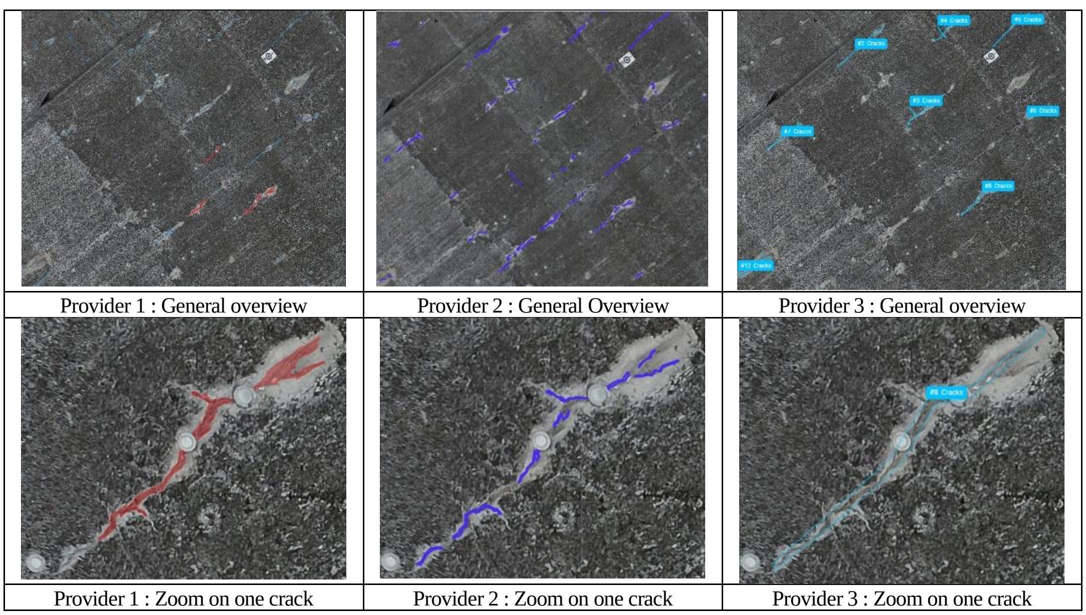
## ABOUT THE DEFECT DATABASE

### Content

Cooling towers are structures with surfaces ranging from 40,000 to 60,000 m². Their inspections are currently carried out by drones equipped with high-performance cameras. The images are captured approximately 7 meters from the surface over its entirety and are post-processed after the inspection. This post-processing involves flattening the captured images, known as orthophoto.

### Feedback on the creation of the database and the search for service providers

To create a quality database, we enlisted a company specializing in image annotation. Several selection criteria were applied: cost, annotation quality, annotation platform quality, and responsiveness. After signing an NDA, each company received 10 images and annotation guidelines. The contacted companies were responsive, but we noticed differences in the precision and number of annotated cracks (Figure 2). Billing methods also varied: some required an annual subscription, while others charged per annotation or per image. The annotation platforms differed in terms of design and image loading times. The annotation process lasted 2 months. It is crucial to agree on a method for reviewing images to provide feedback and highlight missing annotations. Reviewing annotations is time-consuming, so it is essential to ensure that the guidelines are well understood by the annotators.



Provider 1 : General overview | Provider 2 : General Overview | Provider 3 : General overview

Provider 1 : Zoom on one crack | Provider 2 : Zoom on one crack | Provider 3 : Zoom on one crack

Figure 2. Comparison of Annotations Made by Three Different Providers

## Database Creation

To facilitate easy annotation, the orthophoto was divided into images of 10k pixels by 10k pixels. After this preprocessing, three cooling towers were manually re-annotated precisely to enable their use in segmentation neural networks. Three classes of defects were used: cracks, corrosion, concrete detachment. The contents of the database are detailed in TABLE I and II.

It is noted that the database contains a significant number of crack annotations, but far fewer examples of corrosion and concrete detachment. It is also observed that defects represent only a tiny fraction of the pixels (0.7%) in the database images, with the remaining pixels corresponding to healthy concrete. Furthermore, in EDF's database, only 27,279 images out of 278,331 contain defects. A comparison of this database with those available and identified as open source is provided in Table III [1] [2] [3] [4] [5]. Unfortunately, it is not possible to compare the number of annotations present in each.

TABLE I. DATABASE

| Class | Number of annotations | Building 1 | Building 2 | Building 3 | Average annotation size (px) | % of the total surface |
|---|---|---|---|---|---|---|
| Cracks | 83,033 | 29,243 | 29,583 | 24,207 | 3,996 | 0.33 |
| Corrosion | 3,113 | 1,187 | 1,389 | 537 | 1,136 | 0.0035 |
| Concrete detachment | 3,014 | 1,636 | 853 | 525 | 12,508 | 0.037 |

TABLE II. NUMBER OF IMAGES WITH AND WITHOUT CRACKS

| Building | Total number of 512*512px images | Number of 512*512px images with cracks | Number of 512*512px images without cracks |
|---|---|---|---|
| Building 1 | 140,790 | 8,845 | 131,945 |
| Building 2 | 86,640 | 10,102 | 76,538 |
| Building 3 | 50,901 | 8,332 | 42,569 |
| Total | 278,331 | 27,279 | 251,052 |

TABLE III. OPEN-SOURCE DATABASES

| Database | Content description | Nb of images in eq. 512*512 px | Original Image size (in pixels) | Number of cracks annotations | Number of annotations of corroded bars | Number of annotations of spallation | Annotation type | Ref |
|---|---|---|---|---|---|---|---|---|
| EDF | Cooling tower | 278 331 | 10k*10k | 83 033 | 3 113 | 3 014 | Segmentation | |
| CODEBRIM | Bridges | 85 000 | 6k*4k max | 2 507 | 1 507 | 1 898 | Bounding boxes | [1] |
| Crackforest | Roads | 118 | 320*480 | 118 | 0 | 0 | Segmentation | [2] |
| SDNet2018 | Pavements, walls, bridges | 14 023 | 256*256 | 8 484 | 0 | 0 | Classification | [3] |
| Crack500 | Pavements | 6 000 | 2k*1.5k | 500 | 0 | 0 | Segmentation | [4] |
| CrackSeg9K | Pavements Civil works | 9 255 | 400*400 | NA | NA | NA | Segmentation | [5] |

## IMAGE PROCESSING PIPELINE

### Image preprocessing

The preprocessing of images involved sorting the images and formatting them for the neural network input. Images with large black areas or various objects (ladders and platforms) were excluded to retain only concrete surfaces, thus avoiding errors in the deep learning model. Next, the 10k x 10k pixel images were resized to 512 x 512 pixels, a format suitable for the SegFormer neural network. This resizing is necessary due to GPU memory limitations and recommendations for RGB images on 24GB GPUs. It has to be noted that no data augmentation has been used in the pipeline presented in this paper.

### The Neural Network : CrackSegFormer

The CrackSegFormer architecture by Li et al. [6] has been integrated into our pipeline. Based on the SegFormer architecture introduced by Xie et al. [7], SegFormer is an efficient semantic segmentation framework that unifies Transformer-based encoders and multilayer perceptron decoders. SegFormer includes a hierarchically structured Transformer encoder that produces multi-scale features and a lightweight multilayer perceptron decoder that aggregates information from different layers, thus combining local and global attention to provide powerful representations. Li et al. [6] subsequently proposed CrackSegFormer, a model for crack detection on concrete and asphalt surfaces. CrackSegFormer combines Cross-Entropy and Dice loss functions to improve the detection of fine cracks.

### Neural Network Training

The training dataset was built from images with cracks and with a resolution of 512*512 pixels. The repartition between training, validation, and test datasets was a 70 / 20 / 10% respectively. The model was trained on 100 epochs. TABLE IV shows this repartition for building 1 that is used in the following.

TABLE IV. NUMBER OF IMAGES IN THE AUGMENTED AND UNAUGMENTED SETS FOR BUILDING 1

| Dataset tag | Number of images (total) | Nb. Of images: Training set | Nb. Of images: Validation set | Nb. Of images: Test set | Train & validation set from: | Test Set Origin |
|---|---|---|---|---|---|---|
| Dataset 1on1 | 8,845 | 6,193 | 1,771 | 881 | Building 1 | Building 1 |

TABLE V. PREDICTION SCORES WITH AND WITHOUT AUGMENTATION

| Prediction | Precision | Recall | F1-score | IoU | Mean IoU |
|---|---|---|---|---|---|
| Cracks | 0.385 | 0.321 | 0.350 | 0.212 | 0.596 |

## RESULTS AND PERFORMANCE MEASUREMENT

The initial performance metrics used to evaluate the predictions are precision, recall, F1-score, IoU, and mean-IoU. With a precision of 38.5% and a recall of 32.1%, the results obtained according to these metrics are not satisfactory based on a strictly numerical analysis (TABLE V). However, we noticed that these metrics are not representative of the alignment between the predictions provided by the algorithm and the actual needs of the end user. When we submitted the results to the end users of the algorithm, they indicated that, even if they were sometimes imperfect, they were already satisfied with the predictions. Thus, although the obtained metrics are low, they must be supplemented by visual analysis to confirm or refute them.

## DISCUSSION

### Reference Database

The construction of the database was outsourced with strict defect annotation requirements. Despite this, it is impossible to obtain a perfect database. In fact :

- Not all defects can be found by humans and annotated with certainty.
- The exact contours of cracks are difficult to determine due to the limitations of the images in terms of quality and so pixel size.
- The quality of annotations can be subjective and probably also varies depending on the annotators' fatigue. Verifying annotations is also time-consuming (5 minutes per image).

Therefore, it is important to take this into account when analyzing the results based on usual metrics, which are based on an imperfect ground truth. It was not uncommon to find that many predictions were more accurate than those made by human annotators (particularly regarding the width of the cracks). Moreover, some cracks were found by the AI algorithm and not by humans.

### Database Unbalance

The defect database is highly imbalanced, firstly in terms of the number of defects per class and secondly in terms of the number of healthy concrete pixels compared to the number of pixels corresponding to a defect. Regarding the lack of certain types of defects, data augmentation could be beneficial.

### Data Augmentation

Data augmentation is used to address the imbalance in databases. Although not presented in this paper, we found that classical techniques (rotations, flips) did not significantly improve the quality of crack predictions. However, this conclusion may not apply to the other types of defects that we didn’t talk about in this paper i.e. corrosion and concrete spallation. Moreover, techniques for creating artificial defects generated by AI seem promising and deserve to be tested.

### The Quest for the Perfect Training Database

EDF inspects numerous cooling towers at different production sites. For now, the pipeline has only been tested on one building, with training based on this same building. However, the ultimate goal is to be able to use this pipeline on a larger set of buildings, whose surface appearance will be different. Therefore, we need to verify the necessary conditions to achieve a pipeline that provides accurate predictions regardless of the building being processed. For this, an appropriate training database will be the subject of future studies.

### Performance Metrics

We noticed during the evaluation of our pipeline's performance that the metrics commonly used to assess the performance of segmentation algorithms were not correlated with the opinions of our end users. Specifically:

- They are penalizing because they are based on mathematical formulas that take into account the class of each pixel. However, the database constituting the ground truth on which these metric formulas are based is itself imperfect. Figure 3 provides an example of the differences between the predicted crack mask and the one annotated by humans, thus constituting the ground truth. TABLE III explains the meaning of the pixel colors. It is noted that the differences in predictions are located at the edges of the crack. In the case presented here, the prediction was better than the ground truth. Of course, not all errors stem from this observation, but it helps visualize the problem posed by the imperfection of the ground truth database.
- The metrics do not correspond to the notion of performance estimated by the end user, who is more focused on the number of missed defects, the length, and the width of the correctly predicted cracks.

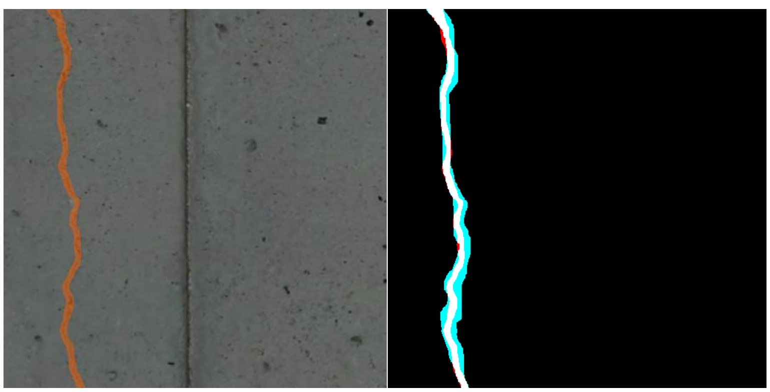


Figure 3 – Crack Prediction (on the left) and Visualization of the Correctly/Incorrectly Classified Pixels (on the right)

TABLE III. ENCODING RULES FOR PREDICTED CLASSES

| Pixel Color | Meaning |
| --- | --- |
| Black | Concrete pixel predicted as concrete |
| White | Crack pixel predicted as crack |
| Cyan | Crack pixel predicted as concrete |
| Red | Concrete pixel predicted as crack |

## CONCLUSION

EDF regularly inspects its civil engineering structures using drones equipped with cameras. Although the inspection takes a day or half a day for a cooling tower, the manual post-processing of the images can last several weeks. The use of AI could speed up this process. The article focuses on the experience gained during the construction of this pipeline and demonstrates the feasibility of automating image processing. Setting up an automated processing pipeline requires a high-quality database. Three cooling towers were re-annotated to allow the training and testing of neural networks (CrackSegFormer) to detect cracks. Creating a perfect database is very difficult, as it is complicated to ensure that all defects are seen and annotated at the pixel level by a human. The pre-processing of the images consisted of sorting them and formatting them for input into the SegCrackFormer neural network. The performance of the pipeline was measured using classic metrics (precision, recall, F1-score, IoU, mean IoU). These metrics proved to be severe and uncorrelated from the end user's point of view. For cracks, a metric based on the total length of cracking found or falsely added per image would probably be more suitable.

## FUTURE WORK

The encouraging results of this first phase of constructing the automated image processing pipeline for cooling tower inspections will be followed by new developments aimed at:

- Extending the pipeline to detect other defects such as corrosion and concrete spallation.
- Optimizing the training database for generalized use without having to annotate all towers beforehand.
- Testing the contribution of data augmentation and synthetic defects generated by AI.
- Improving the post-processing of predictions to extract useful information, such as the length and width of cracks.
- Comparing the performance of CrackSegFormer with other neural networks like Yolo or Mask RCNN.

## REFERENCES


1. Mundt, M., Majumder, S., Murali, S., Panetsos, P., & Ramesh, V. (2019). Meta-learning Convolutional Neural Architectures for Multi-target Concrete Defect Classification with the COncrete DEfect BRidge IMage Dataset. In *CVPR 2019*.
2. Shi, Y., Cui, L., Qi, Z., Meng, F., & Chen, Z. (2016). Automatic Road Crack Detection Using Random Structured Forests. *IEEE Transactions on Intelligent Transportation Systems*, 17(12), 3434–3445.
3. Dorafshan, S., Thomas, R. J., & Marc. (2018). SDNET2018: An annotated image data set for non-contact concrete crack detection using deep convolutional neural networks. *Architectural Engineering -- Faculty Publications*, 162.

4. Yang, F., Zhang, L., Yu, S., & Prokhorov, D. (2019). Feature Pyramid and Hierarchical Boosting Network for Pavement Crack Detection. *IEEE Transactions on Intelligent Transportation Systems*, PP(99), 1-11.
5. Kulkarni, S., Singh, S., Balakrishnan, D., Sharma, S., Devunuri, S., & Korlapati, S. C. R. (2022). CrackSeg9k: A Collection and Benchmark for Crack Segmentation Datasets and Frameworks. In *Computer Vision – ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VII* (pp. 179–195). https://doi.org/10.1007/978-3-031-25082-8_12
6. Li, H., Zhang, H., Zhu, H., Gao, K., Liang, H., & Yang, J. (2024). Automatic crack detection on concrete and asphalt surfaces using semantic segmentation network with hierarchical Transformer. *Engineering Structures*, 307, 117903.
7. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., & Luo, P. (2021). SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. In *35th Conference on Neural Information Processing Systems (NeurIPS 2021)*.
8. Mundt, M., Majumder, S., Murali, S., Panetsos, P., & Ramesh, V. (2019). Meta-learning Convolutional Neural Architectures for Multi-target Concrete Defect Classification with the COncrete DEfect BRidge IMage Dataset. In *CVPR 2019*.