

# Multi-temporal crack segmentation in concrete structure using deep learning approaches

Said Harb<sup>1,\*</sup>, Pedro Achancaray<sup>1</sup>, Mehdi Maboudi<sup>1</sup>, Markus Gerke<sup>1</sup>

<sup>1</sup> Institute of Geodesy and Photogrammetry, Technische Universität Braunschweig, Germany - {s.harb,p.diaz,m.maboudi,m.gerke}@tu-braunschweig.de

**Keywords:** Crack segmentation, multi-temporal, structural health monitoring, concrete structure, deep learning, transformers

## Abstract

Cracks are among the earliest indicators of deterioration in concrete structures. Early automatic detection of these cracks can significantly extend the lifespan of critical infrastructures, such as bridges, buildings, and tunnels, while simultaneously reducing maintenance costs and facilitating efficient structural health monitoring. This study investigates whether leveraging multi-temporal data for crack segmentation can enhance segmentation quality. Therefore, we compare a Swin UNETR trained on multi-temporal data with a U-Net trained on mono-temporal data to assess the effect of temporal information compared with conventional single-epoch approaches. To this end, a multi-temporal dataset comprising 1356 images, each with 32 sequential crack propagation images, was created. After training the models, experiments were conducted to analyze their generalization ability, temporal consistency, and segmentation quality. The multi-temporal approach consistently outperformed its mono-temporal counterpart, achieving an IoU of 82.72% and a F1-score of 90.54%, representing a significant improvement over the mono-temporal model's IoU of 76.69% and F1-score of 86.18%, despite requiring only half of the trainable parameters. The multi-temporal model also displayed a more consistent segmentation quality, with reduced noise and fewer errors. These results suggest that temporal information significantly enhances the performance of segmentation models, offering a promising solution for improved crack detection and the long-term monitoring of concrete structures, even with limited sequential data.

## 1. Introduction

Infrastructures such as highways, roads, dams, harbors, and buildings are essential for the sustainability and efficiency of economic activities, ensuring public safety, and facilitating social interactions (Zhou et al., 2023). Many of these infrastructures are constructed from materials such as concrete, asphalt, or stone and face various challenges, including fatigue stress, cyclic loading, and the cumulative effects of time, along with increasing human and environmental pressure (Mohan and Poobal, 2018; König et al., 2022; Kheradmandi and Mehranfar, 2022). These factors can compromise structural integrity (Hamishehbar et al., 2022) and diminish the asset value of these constructions (Kheradmandi and Mehranfar, 2022). Consequently, there is a pressing need for effective Structural Health Monitoring (SHM) to maintain and prolong the lifespan of critical structures (Hamishehbar et al., 2022). This helps reduce maintenance and repair costs (Kheradmandi and Mehranfar, 2022; König et al., 2022) and prevents damage to humans and the environment (Mohan and Poobal, 2018).

Numerous structures worldwide are in poor condition and require continuous monitoring and damage detection systems. For instance, German seaports feature approximately 3000 km quay walls and 2500 km facilities along the federal waterways. Throughout 140 km of these quay walls, approximately 279 million tons of goods are handled, underscoring the significant role that these structures play in the German economy. However, the current state of quay walls and facilities is a concern. According to the Federal Waterways Engineering and Research Institute (BAW), 70% of quay walls and facilities on waterways are in adequate condition. This situation foreshadows the considerable maintenance efforts required to ensure continued transportation of essential goods on German waterways (Alamouri et al., 2024).

Looking across the Atlantic, a 2022 report highlighted that the United States has 2000 dams and over 46 000 bridges with structural deficits (König et al., 2022), and 11% of roads are classified as being in poor or mediocre conditions (Zhou et al., 2023). Similarly, in the United Kingdom, approximately 10% of roads are disrepaired (König et al., 2022). These examples illustrate the significant global need for infrastructure assessment and maintenance.

One of the earliest and most common indicators of structural degradation and reduced structural integrity is the presence of cracks on the surfaces of structures (König et al., 2022; Hamishehbar et al., 2022). Cracks begin at the microscopic level and continuously reduce the local stiffness of materials and create material discontinuities (Mohan and Poobal, 2018). If left untreated, these cracks grow in size, and the cost and effort required to repair them increase accordingly. To mitigate consequential safety issues, it is crucial for SHM to precisely assess the state of a structure and identify indicators of future damage such as cracks König et al. (2022).

Traditionally, manual visual assessment by an inspector has been the standard method for identifying cracks. However, new machine learning (ML) techniques, particularly deep learning (DL), have begun to change this landscape. Manual assessments are costly, labor-intensive, time-consuming, and require highly trained experts. Despite this training, human factors can lead to subjective results (Hamishehbar et al., 2022; Mohan and Poobal, 2018; Maboudi et al., 2021). For instance, inspectors may experience fatigue during assessments, exhibit inconsistencies in their evaluations, or receive inadequate training (König et al., 2022). Moreover, the manual inspection can pose hazards to inspectors owing to unsafe structures (Kheradmandi and Mehranfar, 2022). In some instances, the locations requiring inspection may be inaccessible to humans (König et al., 2022), or the inspection process could result in downtime for the structure

\* Corresponding author

(König et al., 2022; De Arriba López et al., 2024), leading to interruptions in traffic on roads, bridges, or tunnels (Kheradmandi and Mehranfar, 2022).

Given these disadvantages, automatic crack detection methods are required. Computer vision (CV) techniques, in conjunction with ML/DL, have proven to be effective for this purpose. In this subfield, image-based approaches have emerged as the most cost-effective methods because of the widespread availability of cameras (Hamishehbar et al., 2022). When mounted on UAVs, these systems can reach otherwise inaccessible locations without interrupting structure use (Mohan and Poobal, 2018).

To the best of the authors' knowledge, as stated by König et al. (2022), there is a lack of research regarding the use of multi-temporal data for CV tasks related to crack detection. In this context, we focus on the semantic segmentation of cracks using multi-temporal data, which is compared to approaches that utilize mono-temporal data. In that regard, a multi-temporal dataset and a corresponding serialized mono-temporal dataset were created. Moreover, we aim to address the research gap in multi-temporal crack propagation data for semantic segmentation. Additionally, the development of a multi-temporal dataset can provide a foundation for future research involving multi-temporal data. The contributions of this study can be summarized as follows:

- Development of mono- and multi-temporal models for crack segmentation.
- Assessment of mono- and multi-temporal models based on performance metrics.

## 2. Related works

Crack detection techniques can be based on RGB, infrared, ultrasonic, laser, and other types of images (Mohan and Poobal, 2018). In this study, we focused on camera-based detection, similar to the dataset used in our experiments. Many review papers categorize automatic crack detection methods into two fundamental types: rule-based and data-driven approaches (Kheradmandi and Mehranfar, 2022; Gong et al., 2024; Hamishehbar et al., 2022). Rule-based methods rely on edge information of cracks, morphological operations, and thresholding techniques. These methods may struggle to perform adequately on less structured and more complex data because of their rigidity and inability to adapt to different contexts. By contrast, data-driven approaches outperform rule-based approaches in managing complex scenes by learning useful feature representations from training data, particularly DL-based methods. As a result, they tend to be more accurate, easier to automate (Gong et al., 2024) and computationally more efficient. Consequently, this study focused on exploring DL approaches.

**Semantic Segmentation of Cracks:** DL approaches can be applied to various tasks related to crack analysis, with the most significant being: crack classification (presence of cracks in the image, e.g., transverse or longitudinal cracks), crack segmentation (distinguishing between crack and background pixels), and crack detection (localization of cracks). An overview of this can be seen in Figure 1. Semantic segmentation can be extended to instance segmentation, which aims to classify and localize each pixel of an individual crack (König et al., 2022). Most studies have emphasized crack segmentation because it provides the most precise localization and extent of cracks at the pixel level,

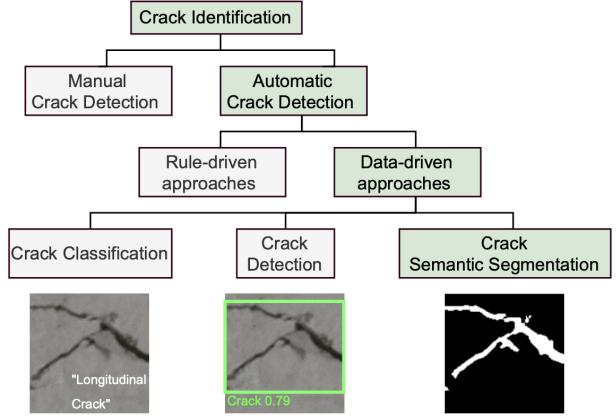


Figure 1. Overview of methods to analyze cracks.

making it particularly valuable for SHM (Hamishehbar et al., 2022; Gong et al., 2024; Zhou et al., 2023).

Semantic segmentation methods can be divided into two main categories: hybrid and pure segmentation. The former integrates classification or detection techniques with image processing methods, whereas the latter focuses solely on segmentation without these additional components. Given the scope of this study, we focus on the latter. According to Hamishehbar et al. (2022), pure segmentation architectures can be further divided into two fundamental types: encoder-decoder and non-encoder-decoder architectures. The encoder-decoder structure is a prominent paradigm in segmentation because it allows the use of various backbones to extract features (Gong et al., 2024). This architecture emphasizes the extraction of contextual information and multi-scale features, rather than the depth of the network. Typically, this is achieved by concatenating features from feature maps of different resolutions, thereby enabling the combination of high-level contextual information with low-level localization features (Gong et al., 2024). The encoder module is then responsible for extracting features from the input image while downsampling it by strided convolutions or attention mechanisms. The decoder module concatenates the information from the encoder and upsamples the extracted features to generate pixel-level predictions (Zhou et al., 2023). The encoder can be substituted by any model architecture that produces hierarchical feature maps. The most commonly used backbones, as noted by Gong et al. (2024) and Hamishehbar et al. (2022), include VGG (Simonyan and Zisserman, 2015), ResNet (He et al., 2016), and transformers (Vaswani et al., 2017).

**Multi-temporal data:** The limited availability of multi-temporal crack propagation datasets has resulted in most crack segmentation research focusing on mono-temporal images (using datasets such as *CrackTree* (Zou et al., 2012), *CFD* (Shi et al., 2016), *CRKWH100* (Zou et al., 2019), *DeepCrack* (Liu et al., 2019), and *Crack500* (Yang et al., 2020), ignoring the crack propagation effect over time. On the other hand, remote sensing (RS) exploits spatio-temporal features due to its access to satellite data from different locations around the world at different times. In this context, RS approaches can be effectively extended to crack detection, as both fields utilize similar data types, namely high-resolution images.

Numerous methods have been utilized in DL models for crack segmentation, with the most notable being those that rely on Convolutional Neural Networks (CNN) and the attention mechanism (Gong et al., 2024; Kheradmandi and Mehranfar, 2022).

In the first category, a distinction can be made between 1D (along the time dimension), 2D (along the spatial dimensions), and 3D (along the spatial and time dimensions) convolutions, depending on the type of relationships learned by the model (Zhong et al., 2019). The attention mechanism can function as the sole mechanism within a transformer architecture (as a feature extractor) or can be integrated into encoder-decoder architectures alongside CNN-based networks (using spatial or channel attention). This mechanism processes data from a global perspective and assigns weights based on the learned significance of information. Although the attention mechanism offers advantages over CNNs in capturing long-range dependencies, it may exhibit limitations in localization capabilities because of its large receptive field (Gong et al., 2024; Zhong et al., 2019).

**Mono- and Multi-temporal models:** One of the first models for semantic segmentation was the Fully Convolutional Network (FCN) (Shelhamer et al., 2016), which first reduces the size of the input image through convolution and pooling layers (encoder) and then upsamples the features again through deconvolutional layers (decoder) to generate a segmentation map for all pixels. Similarly, SegNet employs an encoder-decoder structure, where pooling indices from the encoder are used during upsampling in the decoder. Later, U-Net (Ronneberger et al., 2015) further developed this structure by adding skip connections between the encoder and decoder to preserve spatial information, being successfully applied for crack segmentation (König et al., 2022).

Çiçek et al. (2016) introduced a 3D U-Net, where 3D convolutional layers are employed to segment volumetric images. This model can be easily extended to multi-temporal data, where the third dimension is interpreted not as depth, but as the temporal domain. Xu et al. (2019) presented a Long Short-Term Memory (LSTM) multi-modal U-Net, which is a U-Net with multiple encoder paths designed to process multi-modal data for the segmentation of brain tumor data. An attempt was made to increase the learning capabilities of the model by not fusing the feature extraction of different channels. The decoder concatenates information from every encoder path and produces a segmentation mask. This multi-modal U-Net is applied to different slices of volumetric multi-modal input data. The output segmentation masks create a sequence of segmentation masks based on the depth of the image. This sequence is sequentially input into Convolutional LSTM (ConvLSTM) networks to capture the sequence dependencies. Therefore, volumetric data is fully segmented. Rustowicz et al. (2019) employed a 2D U-Net in combination with a ConvLSTM for crop type segmentation using multi-modal and multi-temporal satellite data. The model has a U-Net type structure, where the encoder first extracts the features of each channel independently using CNNs, and these features are each fed into a ConvLSTM. Zhou et al. (2023) proposed the Swin UNETR model, which also employs a U-Net architecture with a Swin Transformer as a backbone in the encoder, enhanced to process volumetric data. Feature extraction of the backbone relies solely on the attention mechanism, whereas CNNs are only employed within residual blocks in skip connections to refine the features. The decoder concatenates features from the feature maps with different resolutions and upsamples them to the size of the input data. Therefore, volumetric or multi-temporal data can be fully segmented. Because this model extracts features using the attention mechanism and can segment multi-temporal data in the original input size, we selected this model for multi-temporal crack segmentation. Moreover, to the best of the authors' knowledge, attention

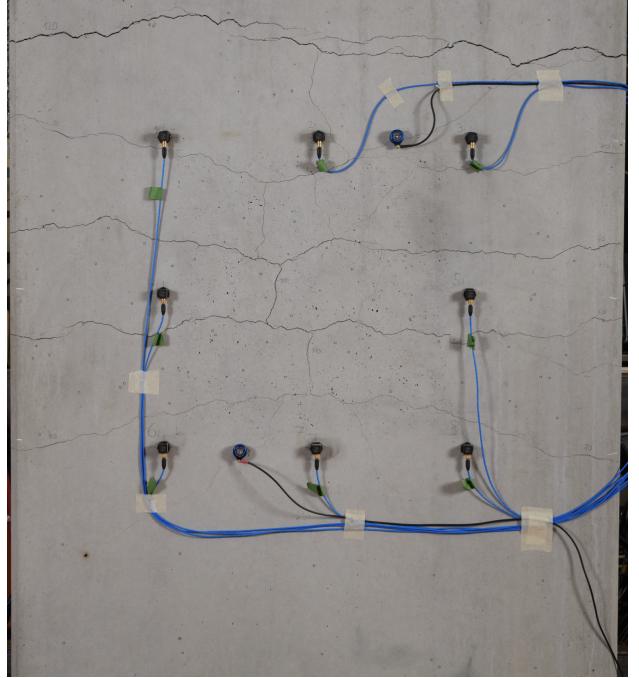


Figure 2. Concrete block in the last epoch. This is the last stage of crack propagation on the concrete block. The preceding 24 images show the incremental crack propagation through time.

mechanisms have not yet been applied to multi-temporal crack segmentation.

### 3. Materials and Methods

To address the aforementioned research question, a multi-temporal dataset was created, and a Swin UNETR model was selected to serve as the foundation for a comparative analysis of mono- and multi-temporal approaches.

#### 3.1 Dataset

A temporal sequence of images depicting crack propagation is required to create a multi-temporal dataset for crack segmentation. For this purpose, a setup was constructed involving a vertical concrete slab with an area of  $2 \text{ m}^2$  and thickness of 15 cm (see Figure 2), which was reinforced and positioned upright. A force was applied to the center of the slab, with the load incrementally increasing by 20 kN at each load stage, referred to as epochs, until the maximum load was reached. Eight epochs were recorded before the concrete slab failed. During each epoch, high-resolution RGB images were captured using a mounted camera positioned at a specified distance (Backhaus et al., 2024). This study used 25 images from the final epoch to generate training data. These images are in temporal order and provide a detailed view of the entire concrete block, with the progression of cracks between the images visible upon visual inspection.

The creation of mono- and multi-temporal datasets can be divided into three phases: labeling, post-processing, and aggregation. In the labeling phase, because manual annotation of all images is tedious and extremely time-consuming, a set of three images was manually annotated and used to train a segmentation model (auxiliary U-Net) to assist the annotation process of the remaining 25 images. The post-processing phase involves

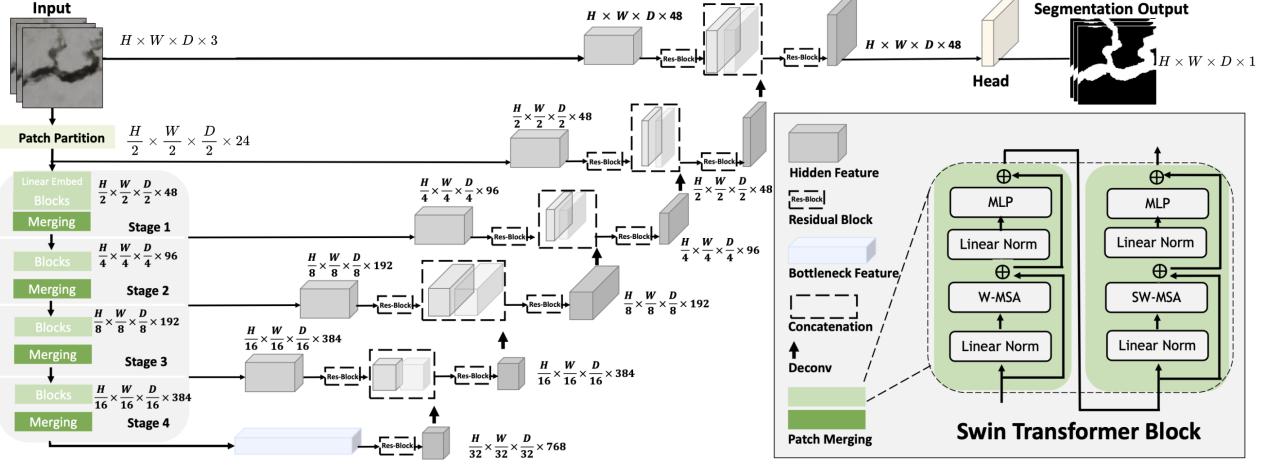


Figure 3. Swin UNETR architecture. A 3D Swin Transformer is used as a feature extractor, and in the decoder, the feature maps are concatenated and upsampled to the original input size. Modified from Hatamizadeh et al. (2022).

the removal of false positives (pencil drawings, concrete pores, shadows, and cables) and segmentation errors (isolated groups of pixels smaller than five pixels) in the auxiliary U-Net predictions. Then, morphological operations were applied (closing with a  $3 \times 3$  kernel), followed by manual cleaning to refine the crack annotations. Finally, the aggregation phase consists of fulfilling the model requirements for training, as the multi-temporal model requires sequences of 32 images, but only 25 are available. To address this, seven images at evenly spaced intervals were duplicated to expand the sequence to 32 images.

To create the mono-temporal dataset, patches of  $128 \times 128$  were extracted from the image sequences and their corresponding annotations (targets), resulting in a total of 180 224 image-target pairs. The multi-temporal dataset contains a stack of all images, resulting in 5 632 multi-temporal image-target pairs. As expected, there were several samples without cracks. To balance the dataset, all multi-temporal samples with cracks were included, and samples without cracks were randomly selected to achieve a 2:1 ratio of images with cracks to those without cracks. The training, validation, and test sets were created at the following proportions: 60%, 20%, and 20%, respectively. To generate the mono-temporal dataset and ensure that both models have access to exactly the same data, the multi-temporal samples within the splits were deserialized. Tables 1 and 2 show the dataset statistics.

Sample size	Crack image ratio (%)	Crack pixel ratio (%)
All	43 392	40.1
Train	26 016	40.1
Val	8 672	39.1
Test	8 704	41.3

Table 1. Mono-temporal dataset. The deserialization of the multi-temporal dataset leads to a decreased crack image ratio in the mono-temporal dataset.

### 3.2 Methodology

The methodology employed in this study is as follows. First, the datasets were created and annotated as previously mentioned. Second, a multi-temporal model was trained to perform the segmentation task on the datasets. Finally, the assessment of the multi-temporal approach will be quantitative, based on metrics,

Sample size	Crack image ratio (%)	Crack pixel ratio (%)
All	1 356	66.7
Train	813	66.5
Val	271	66.8
Test	272	66.9

Table 2. Multi-temporal dataset. The sample size of the multi-temporal dataset is 32 times smaller than in the mono-temporal dataset because of sequential stacking.

and qualitative, based on a visual inspection of the segmentation results.

We selected the Swin UNETR model for multi-temporal crack segmentation to extract features from the spatial and temporal domains using its attention mechanisms.

**Swin UNETR:** The Shifting Window U-Net Transformer (Swin UNETR) was introduced by Hatamizadeh et al. (2022) and extends the Shifting Window Transformer (Swin Transformer) (Liu et al., 2021) to perform semantic segmentation on three-dimensional inputs. This is achieved by combining a Swin Transformer adapted for three-dimensional inputs with a U-Net architecture. In Figure 3 the architecture of Swin UNETR is depicted. The input tokens are patches of pixels, which are embedded into a feature dimension  $C$  in the input layer. The Swin Transformer then serves as a feature extractor in the encoder path, whereas the decoder path reassembles the original input shape and predicts each voxel using the feature maps produced by the Swin Transformer. The Swin Transformer enables dense prediction tasks for the transformer architecture, which is not feasible in the preceding Vision Transformer (ViT) (Dosovitskiy et al., 2021), owing to its quadratic computational cost.

The computational complexity of the Swin Transformer is reduced to a linear order by applying the attention mechanism only in local windows instead of the whole image and shifting the window configuration to introduce the global context in two consecutive Swin Transformer blocks. The attention mechanism is embedded within the window-masked self-attention and shifted-window masked self-attention modules in transformer encoder blocks, which were presented by Vaswani et al. (2017) in the original transformer. Patch-merging layers decrease the resolution of feature maps in each stage, thereby creating a hier-

archical structure of feature maps that can be used for dense prediction tasks.

#### 4. Experimental Setup and Results

In order to compare the mono- and multi-temporal approach this section will show the different experiments conducted and draw a conclusion based on the results of the experiments.

##### 4.1 Experimental Setup

Three experiments were conducted by modifying the size and transformations of the training set. The first experiment examined the effect of a reduced sample size, where the mono-temporal training and validation datasets for the U-Net were matched to the smaller size of the multi-temporal dataset. The second and third experiments focused on data augmentation (DA) by applying five transformations (horizontal and vertical flips, brightness, contrast, and blurring) with a probability of 0.5. The second experiment applied DA to expand the multi-temporal SwinUNETR dataset, whereas the third applied DA to the mono-temporal U-Net dataset.

As baselines, the following models were used: a mono-temporal U-Net (31M parameters) and a multi-temporal SwinUNETR (15.7M parameters), both implemented using PyTorch (Paszke et al., 2019) and trained on a Tesla P100 GPU. We employed early stopping with a patience of 20 epochs, monitoring the validation Intersection over Union (IoU) for the crack class, along with a step learning rate scheduler that adjusted the learning rate by a factor of 0.1 if the validation loss for the class crack does not decrease for 10 epochs. For the batch size, a value of 4 was chosen. Both models processed  $128 \times 128$  images owing to memory constraints. SwinUNETR was taken from the Medical Open Network for AI (MONAI) (Cardoso et al., 2022) framework and has an input channel size of 3, an output channel size of 1, and a feature size  $C = 24$ ; otherwise, the configuration is the default one described in the MONAI documentation.

#### 4.2 Results

**4.2.1 Quantitative analysis:** Table 3 summarizes the results obtained on the test set in the three experiments and the baseline models (BL) in terms of different quantitative metrics: IoU, Precision (P), Recall (R), and F1-score (F1). DS and DA are the downsampled dataset and dataset with data augmentation, respectively. We can see that the BL multi-temporal Swin UNETR outperformed all mono-temporal U-Net models. BL Swin UNETR achieved the highest metric scores, with an IoU of 82.7% and an F1-score of 90.5%. Swin UNETR DA (experiment 2) performed only slightly worse than BL Swin UNETR, a difference that may be attributed to the inner randomness of each model, given that the metrics are closely aligned. Notably, the recall is higher in the DA model compared to the BL model. Overall, both models exhibited similarly strong performance on the test set, with balanced P and R, indicating robust segmentation capabilities.

The best-performing U-Net on the test set was the BL U-Net model, which achieved an IoU of 76.7% and an F1-score of 86.2%. A similar trend is observed in the Swin UNETR models, where the model trained with DA performs marginally worse than the BL model on the original dataset. Notably, the F1-score of the BL model was only 0.63 percentage points higher than that of the DA U-Net from experiment 3. In contrast,

Exp.	Model	DS	DA	Metrics (%)			
				IoU↑	P↑	R↑	F1↑
BL	U-Net	—	—	76.7	87.0	86.6	86.8
BL	Swin UNETR	—	—	<b>82.7</b>	<b>91.9</b>	89.2	<b>90.5</b>
1	U-Net	✓	—	71.9	80.5	87.1	83.7
2	Swin UNETR	—	✓	82.5	91.2	<b>89.6</b>	90.4
3	U-Net	—	✓	75.7	86.0	86.3	86.2

Table 3. Results overview. Metrics obtained by the selected methods in the test set. BL: baseline, DS: downsampled dataset, DA: dataset with data augmentation.

the U-Net from experiment 1, which was trained on a downsampled (DS) set (denoted as DS U-Net), showed a larger performance gap compared to both the BL model and the DA U-Net. The IoU and F1-score for the DS U-Net are both more than 3 percentage points lower than those of the DA U-Net and BL models. Furthermore, P and R were balanced for all U-Net experiments, except for experiment 1. In experiment 1, R was the highest among all U-Net experiments on the original dataset; however, this model exhibited a lower P compared to P across all other U-Net experiments. In summary, the results indicated that the multi-temporal approach consistently outperformed the mono-temporal approach across the test set. This suggests that the multi-temporal model effectively learned the temporal dependencies between sequential images, leading to improved segmentation results.

**4.2.2 Qualitative analysis:** In addition to the metrics, it is crucial to visually inspect the predictions of both the mono- and multi-temporal models to assess whether the multi-temporal model successfully learns temporal dependencies, in contrast to the mono-temporal model. For this purpose, the predictions of the baseline models were used.

A brief visual inspection of numerous examples revealed that the multi-temporal Swin UNETR exhibits superior segmentation quality, aligned with the expectations set by the metrics. This improved performance was observed in several aspects. First, the predictions made by the Swin UNETR were smoother and contained less noise than those of the U-Net. This suggests that the Swin UNETR effectively learned the temporal dependencies associated with crack propagation, resulting in greater confidence when consistently classifying the same region of an image as a crack. Second, although U-Net captures the general shape of the cracks, it introduces a substantial number of artifacts into its predictions. This inconsistency is particularly evident in Figure 4, where Swin UNETR demonstrates greater consistency in predicting the crack shape and accurately classifying regions according to crack propagation. In contrast, U-Net's predictions vary significantly from image to image, primarily because of the noisy outputs. Consequently, more segmentation holes appear in the U-Net predictions, which are significantly less prevalent in those from Swin UNETR. While U-Net generally captures the edges of cracks well, it often lacks the same level of detection quality at the center of the cracks.

U-Net and Swin UNETR demonstrated similar performance on early stage cracks, as illustrated in Figure 5, where both models produce consistent predictions that accurately depict the target. However, in the later stages of crack propagation, as shown in the subsequent images of Figure 5, U-Net's predictions become less consistent and exhibit numerous holes not present in the target. In contrast, the Swin UNETR maintains a high level of consistency with no holes in its segmentation.

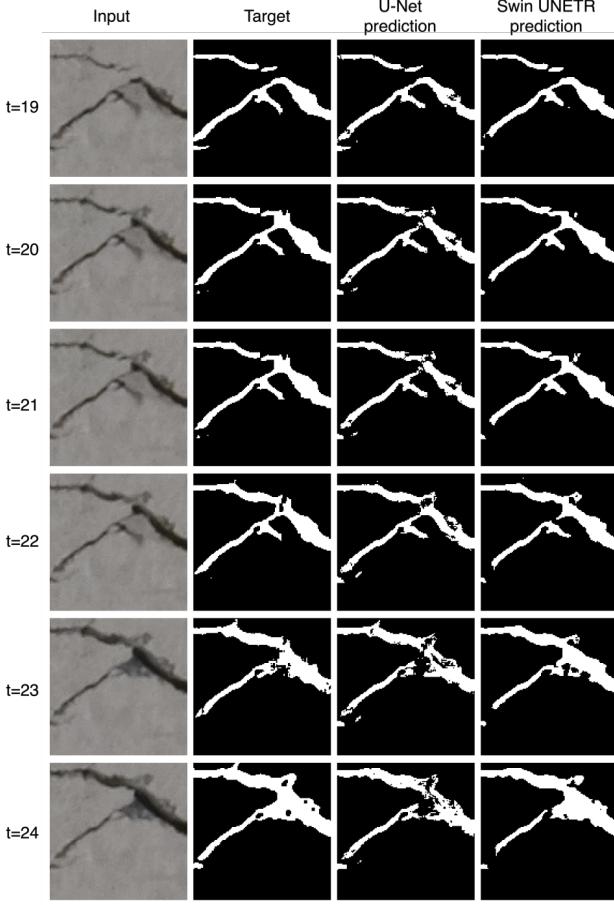


Figure 4. Comparison of U-Net and Swin UNETR sequential predictions. The Swin UNETR produces smoother and more consistent predictions containing less noise than the U-Net.

**4.2.3 Evaluation of false positives:** On the concrete slab, there were several distinct challenges for both models to handle, such as pencil-drawn numbers, sensors, cables, thin and low-contrast cracks, and cavities which all proved to be potential sources for false positive classifications. By generating the prediction of the entire image of the concrete slab, we can see whether the models learned to mitigate these challenges.

The most significant segmentation error is the labeling of the numbers drawn on the concrete with a pencil as cracks. Visual inspection of the input images revealed 18 visible numbers in a single image. By examining the segmentation masks of both models on the complete images, it is possible to count the numbers labeled as cracks by each model, providing insight into whether the models can differentiate numbers from cracks. In the final segmentation mask of the Swin UNETR, nine numbers are visible, whereas U-Net identifies 17 numbers as cracks in the figure. This significant difference underscores the interpretation that the multi-temporal Swin UNETR has learned that these numbers do not possess typical crack features such as propagation over time, and are therefore not cracks. In contrast, U-Net relies more on less sophisticated features, such as the shape and color of the numbers, to determine if they represent cracks. Some examples of these challenges are presented in Figure 6. The pencil-drawn number is completely recognized as a crack by U-Net, as well as parts of the sensor and cable. On the other hand, Swin UNETR managed to overcome these challenges; in particular, even thin and low-contrast cracks were

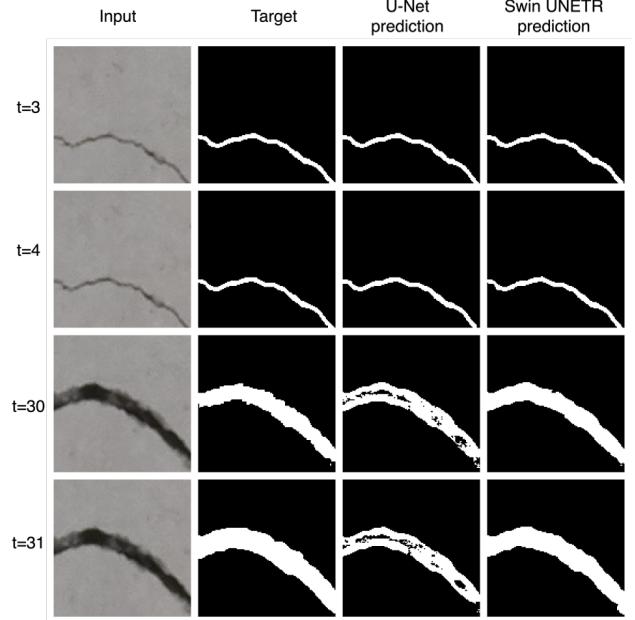


Figure 5. Early stage vs late stage crack segmentation. Both models perform well on early stages cracks, but only the Swin UNETR manages to maintain adequate segmentation quality in later stage cracks.

partly segmented. Cavities in the concrete proved to be a source of noise in the segmentation masks, as both models classified some cavities as cracks. However, U-Net does this to a larger extent than Swin UNETR.

Next, there are 10 sensors with cables attached to the concrete, which present segmentation challenges for the models. Both models often misclassified the edges of these sensors and cables as cracks. In examining the segmentation masks, U-Net misclassified the edges of nine out of ten sensors as cracks, whereas Swin UNETR only did this for one sensor. The dark color of the sensors contributes to this issue because their edges are sometimes mistaken for cracks. The cables attached to the sensors pose an additional challenge to both models. Their long, thin, and darker appearance makes it difficult to distinguish them from the actual cracks. In this case, Swin UNETR outperformed U-Net. U-Net incorrectly classified most of the darker cables as cracks, resulting in a significant false positive. In contrast, the Swin UNETR did not cause this error. However, it misclassifies the edges of the blue cables as long, thin cracks. The final prominent error observed is the misclassification of cavities, which appear as small dark holes in concrete. A comparison of the segmentation masks reveals that Swin UNETR has significantly fewer misclassified cavities than U-Net, which is reflected in the higher IoU of Swin UNETR.

The segmentation masks clearly show that Swin UNETR is more effective at identifying thin cracks than U-Net. However, it is important to note that the performance of both models for thin and low-contrast cracks is lower than that for more common cracks.

## 5. Conclusions

This study examined the effect of multi-temporal data on the performance of semantic segmentation for crack images. The Swin UNETR was trained on a multi-temporal crack dataset,

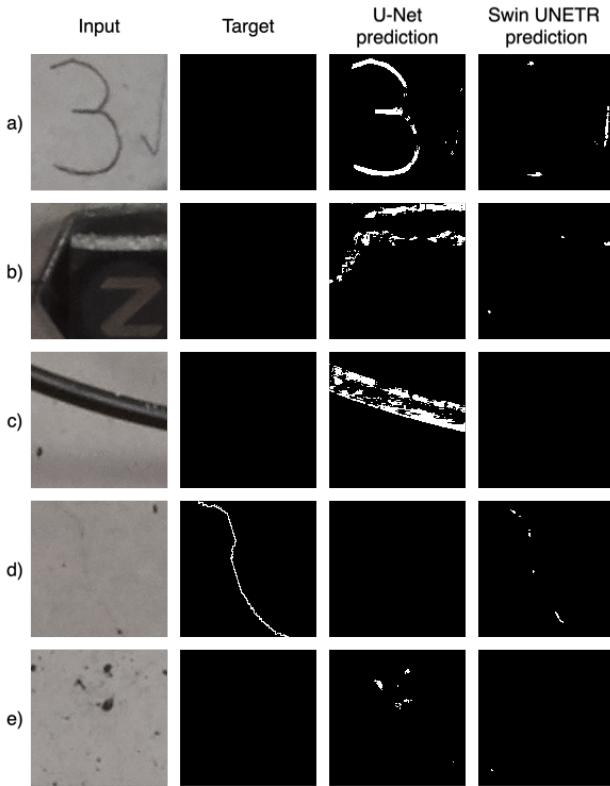


Figure 6. Comparison of segmentation errors. a) Pencil drawings, b) Sensor, c) Cable, d) Thin and low contrast crack, e) Cavities

whereas a U-Net was trained on the deserialized version of the same dataset for comparison. Swin UNETR achieved an Intersection over Union (IoU) of 82.72% and an F1-score of 90.52%, whereas U-Net resulted in an IoU of 76.69% and an F1-score of 86.81% on the test set. This indicates that the multi-temporal approach outperformed the mono-temporal approach, even when utilizing only half the parameters compared to U-Net.

The multi-temporal Swin UNETR demonstrated superior segmentation quality compared with the mono-temporal U-Net. Swin UNETR predictions exhibited greater consistency, reduced noise, and enhanced ability to distinguish cracks from visually similar features in the images. By leveraging the entire time-series data, the multi-temporal model significantly improved its performance in ambiguous image regions, resulting in more confident predictions. In contrast, mono-temporal U-Net encountered several challenges. It struggled to segment larger cracks without introducing artifacts, and often produced noisy predictions. Furthermore, U-Net frequently failed to differentiate between actual cracks and visually similar features, highlighting its limitations in this specific application. Access to temporal information proved to be a crucial advantage for the Swin UNETR, enabling it to make informed decisions based on the evolution of features over time. This temporal context allowed the model to better interpret ambiguous areas and reduce false positives, resulting in more accurate and reliable crack segmentation.

**Acknowledgment:** The authors would like to thank the DFG (German Research Foundation) for the support within the project “Optical 3D-Bridge-Inspect” – Project Number 501682769 as part of DFG’s Priority Programme 2388 “Hundred plus”.

## References

- Alamouri, A., De Arriba López, V., Achancaray Diaz, P., Backhaus, J., Gerke, M., 2024. High-resolution data capture and interpretation in support of port infrastructure maintenance. *44. Wissenschaftlich-Technische Jahrestagung der DGPF in Remagen: Stadt, Land, Fluss - Daten vernetzen*, 32, DPGA, Remagen, 269–278.
- Backhaus, J., De Arriba López, V., Maboudi, M., Bestmann, U., Gerke, M., 2024. Combining UAV-based photogrammetry and structured light scanning to support the structural health monitoring of concrete structures. *e-Journal of Nondestructive Testing*, 29(7). <https://doi.org/10.58286/29797>.
- Cardoso, M. J., Li, W., Brown, R., Ma, N., Kerfoot, E., Wang, Y., Murrey, B., Myronenko, A., Zhao, C., Yang, D., Nath, V., He, Y., Xu, Z., Hatamizadeh, A., Myronenko, A., Zhu, W., Liu, Y., Zheng, M., Tang, Y., Yang, I., Zephyr, M., Hashemian, B., Alle, S., Darestani, M. Z., Budd, C., Modat, M., Vercauteren, T., Wang, G., Li, Y., Hu, Y., Fu, Y., Gorman, B., Johnson, H., Genereaux, B., Erdal, B. S., Gupta, V., Diaz-Pinto, A., Dourson, A., Maier-Hein, L., Jaeger, P. F., Baumgartner, M., Kalpathy-Cramer, J., Flores, M., Kirby, J., Cooper, L. A. D., Roth, H. R., Xu, D., Bericat, D., Floca, R., Zhou, S. K., Shuaib, H., Farahani, K., Maier-Hein, K. H., Aylward, S., Dogra, P., Ourselin, S., Feng, A., 2022. MONAI: An open-source framework for deep learning in healthcare. *arXiv (Cornell University)*. <https://doi.org/10.48550/arXiv.2211.02701>.
- De Arriba López, V., Maboudi, M., Achancaray, P., Gerke, M., 2024. Automatic non-destructive UAV-based structural health monitoring of steel container cranes. *Applied Geomatics*, 16(1), 125–145. <https://doi.org/10.1007/s12518-023-00542-7>.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *International Conference on Learning Representations*. <https://openreview.net/pdf?id=YicbFdNTTy>.
- Gong, H., Liu, L., Liang, H., Zhou, Y., Cong, L., 2024. A state-of-the-art survey of deep learning models for automated pavement crack segmentation. *International Journal of Transportation Science and Technology*, 13, 44–57. <https://doi.org/10.1016/j.ijtst.2023.11.005>.
- Hamishehbar, Y., Guan, H., So, S., Jo, J., 2022. A Comprehensive Review of Deep Learning-Based Crack Detection Approaches. *Applied Sciences*, 12(3), 1374. <https://doi.org/10.3390/app12031374>.
- Hatamizadeh, A., Nath, V., Tang, Y., Yang, D., Roth, H. R., Xu, D., 2022. SWIN UNETR: SWIN Transformers for Semantic Segmentation of Brain Tumors in MRI images. 272–284. [https://doi.org/10.1007/978-3-031-08999-2\\_22](https://doi.org/10.1007/978-3-031-08999-2_22).
- He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. <https://doi.org/10.1109/CVPR.2016.90>.
- Kheradmandi, N., Mehranfar, V., 2022. A critical review and comparative study on image segmentation-based techniques for pavement crack detection. *Construction and Building Materials*, 321, 126162. <https://doi.org/10.1016/j.conbuildmat.2021.126162>.

- König, J., Jenkins, M., Mannion, M., Barrie, P., Morison, G., 2022. What's Cracking? A review and analysis of deep learning methods for structural crack segmentation, detection and quantification. *arXiv (Cornell University)*. <https://doi.org/10.48550/arXiv.2202.03714>.
- Liu, Y., Yao, J., Lu, X., Xie, R., Li, L., 2019. DeepCrack: A deep hierarchical feature learning architecture for crack segmentation. *Neurocomputing*, 338, 139-153. <https://doi.org/10.1016/j.neucom.2019.01.036>.
- Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. <https://doi.org/10.1109/iccv48922.2021.00986>.
- Maboudi, M., Alamouri, A., De Arriba López, V., Bajauri, M. S., Berger, C., Gerke, M., 2021. Drone-based container crane inspection: concept, challenges and preliminary results. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, V-1-2021, 121–128. <https://doi.org/10.5194/isprs-annals-V-1-2021-121-2021>.
- Mohan, A., Poobal, S., 2018. Crack detection using image processing: A critical review and analysis. *Alexandria Engineering Journal*, 57(2), 787–798. <https://doi.org/10.1016/j.aej.2017.01.020>.
- 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. Z., 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. *arXiv (Cornell University)*, 32, 8026–8037. <https://doi.org/10.48550/arXiv.1912.01703>.
- Ronneberger, O., Fischer, P., Brox, T., 2015. U-NET: Convolutional Networks for Biomedical Image Segmentation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, 234-241. [https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28).
- Rustowicz, R. M., Cheong, R., Wang, L., Ermon, S., Burke, M., Lobell, D. B., 2019. Semantic Segmentation of crop type in Africa: A novel dataset and analysis of deep learning methods. *Computer Vision and Pattern Recognition*, 75–82. [https://openaccess.thecvf.com/content\\_CVPRW\\_2019/papers/cv4gc/Rustowicz\\_Semantic\\_Segmentation\\_of\\_Crop\\_Type\\_in\\_Africa\\_A\\_Novel\\_Dataset\\_CVPRW\\_2019\\_paper.pdf](https://openaccess.thecvf.com/content_CVPRW_2019/papers/cv4gc/Rustowicz_Semantic_Segmentation_of_Crop_Type_in_Africa_A_Novel_Dataset_CVPRW_2019_paper.pdf).
- Shelhamer, E., Long, J., Darrell, T., 2016. Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4), 640–651. <https://doi.org/10.1109/tpami.2016.2572683>.
- 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. <https://doi.org/10.1109/TITS.2016.2552248>.
- Simonyan, K., Zisserman, A., 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. *3rd International Conference on Learning Representations, ICLR*. <https://doi.org/10.48550/arXiv.1409.1556>.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I., 2017. Attention is All you Need. *arXiv (Cornell University)*. <https://doi.org/10.48550/arXiv.1706.03762>.
- Xu, F., Ma, H., Sun, J., Wu, R., Liu, X., Kong, Y., 2019. LSTM Multi-modal UNet for Brain Tumor Segmentation. 236-240. <https://doi.org/10.1109/ICIVC47709.2019.8981027>.
- Yang, F., Zhang, L., Yu, S., Prokhorov, D., Mei, X., Ling, H., 2020. Feature Pyramid and Hierarchical Boosting Network for Pavement Crack Detection. *IEEE Transactions on Intelligent Transportation Systems*, 21(4), 1525-1535. <https://doi.org/10.1109/TITS.2019.2910595>.
- Zhong, L., Hu, L., Zhou, H., 2019. Deep learning based multi-temporal crop classification. *Remote Sensing of Environment*, 221, 430–443. <https://doi.org/10.1016/j.rse.2018.11.032>.
- Zhou, S., Canchila, C., Song, W., 2023. Deep learning-based crack segmentation for civil infrastructure: data types, architectures, and benchmarked performance. *Automation in Construction*, 146, 104678. <https://doi.org/10.1016/j.autcon.2022.104678>.
- Zou, Q., Cao, Y., Li, Q., Mao, Q., Wang, S., 2012. Crack-Tree: Automatic crack detection from pavement images. *Pattern Recognition Letters*, 33(3), 227-238. <https://doi.org/10.1016/j.patrec.2011.11.004>.
- Zou, Q., Zhang, Z., Li, Q., Qi, X., Wang, Q., Wang, S., 2019. DeepCrack: Learning Hierarchical Convolutional Features for Crack Detection. *IEEE Transactions on Image Processing*, 28(3), 1498-1512. <https://doi.org/10.1109/TIP.2018.2878966>.
- Çicek, O., Abdulkadir, A., Lienkamp, S. S., Brox, T., Ronneberger, O., 2016. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. 424–432. [https://doi.org/10.1007/978-3-319-46723-8\\_49](https://doi.org/10.1007/978-3-319-46723-8_49).