POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

EXECUTIVE SUMMARY OF THE THESIS

Automated Crack Width Measurement in 3D Models: A Photogrammetric Approach with Image Selection

TESI MAGISTRALE IN MECHANICAL ENGINEERING – INGEGNERIA MECCANICA

AUTHOR: HÜSEYİN YASİN ÖZTÜRK

ADVISOR: PROF. EMANUELE ZAPPA

ACADEMIC YEAR: 2024-2025

---

1. Introduction

Cracks in critical infrastructure, such as bridges and pavements, jeopardize structural integrity, and accelerate deterioration. Accurate crack width measurement is essential for assessing safety, prioritizing maintenance, and preventing catastrophic failures. However, conventional inspection methods reliant on visual surveys or telescopic tools are inefficient, subjective, and hazardous, particularly for large or inaccessible structures [1]. Traditional 2D imaging techniques further struggle to provide spatially accurate measurements due to their inability to account for geometric distortions, scale, or three-dimensional context [2].

This work addresses these challenges by introducing an automated, photogrammetry-driven system that integrates multiple images analysis with 3D modelling and deep learning. The system reconstructs high-fidelity 3D meshes of structures using optimal images selected based on camera orientation and distance, ensuring minimal perspective distortion. Crack detection is automated through deep learning algorithms, which outperform traditional methods by generalizing across diverse crack morphologies and lighting conditions [3]. Detected cracks are projected onto the 3D model, enabling precise width measurements in a spatial context and overcoming the planar limitations of 2D imaging.

Conventional crack detection methods, which rely on 2D images, are inherently limited by distortions caused by parallax, occlusions, and surface curvature. For instance, cracks on curved surfaces may appear to vary in width across images due to differences in camera orientation, compromising measurement reliability. To address these limitations, this study introduces a novel approach where crack edges identified in 2D images are projected onto a 3D mesh reconstructed via photogrammetry using Agisoft Metashape. This ensures measurements are based on the physical geometry of the structure, eliminating perspectival distortions inherent to individual images.

Images are systematically chosen based on their attitude to the crack surface, including the angle between the camera's optical axis and the local surface, as well as the distance from the camera to the target. This selective strategy ensures that images with near-orthogonal viewing angles and optimal working distances are prioritized, enhancing crack visibility and segmentation accuracy.

2. Preliminary Procedures

Image Acquisition on Site

A structured image acquisition process was implemented to ensure accurate 3D reconstruction and crack assessment. A high-resolution digital camera (5 MP or higher) was used, avoiding ultra-wide-angle and fisheye lenses to minimize distortions. Key camera parameters, ISO level, exposure time, and aperture size, were adjusted to balance image quality and clarity. Circular targets were placed around or on the object, with measured distances between them providing a reference for scene scaling. High overlap between images ensured successful alignment in Metashape, while additional close-up images captured fine details for crack assessment. The process adhered to best practices to produce high-fidelity 3D models suitable for both reconstruction and crack analysis.

Manual Crack Measurements on Site

To validate the results of the automated crack detection system, manual measurements were conducted on-site using a microscope camera and a crack gauge. The microscope camera allowed for detailed examination of cracks, capturing high-resolution images that revealed intricate details often missed by conventional methods. A crack gauge was positioned within the field of view to provide a known reference distance for measurement calibration.

A custom measurement tool was developed to load these images and enable the manual definition of crack widths in pixel units. The tool compared the pixel dimensions of the cracks to the known distance on the crack gauge, ensuring accurate measurements. Multiple measurements were taken at different locations along each crack to obtain a distribution of crack widths, and statistical methods were applied to these results to facilitate comparison with the automated measurements from the proposed method.

While cracks may appear as simple gaps between two surfaces when viewed from a distance, the microscope camera revealed more complex details, such as spalled regions resulting from material loss. In this study, the focus was specifically on the crack regions, excluding spalled areas to ensure accurate and consistent measurements. This manual validation process provided a reliable benchmark for evaluating the performance of the automated crack detection system.

3D Scene Reconstruction in Metashape

The 3D reconstruction in Agisoft Metashape started with feature detection and matching across overlapping images. Camera positions were estimated, and tie points were manually cleaned to focus on the object. An initial mesh was created for image masking, refining it to remove background elements. Depth maps were generated, producing a dense point cloud that was manually refined for accuracy.

The final 3D mesh, derived from the refined point cloud, provided a high-quality surface representation. Texturing was optional for visualization.

3. Automated Crack Detection and Measurement

Crack Detection and Projection Algorithm

The developed Python script introduces a systematic and automated approach for crack detection, segmentation, and measurement within the Metashape environment. Leveraging user-defined parameters, advanced image processing techniques, and 3D point projection, the algorithm streamlines the analysis of cracks in structural surfaces. It provides a robust framework for identifying crack features, measuring their dimensions, and exporting data for further evaluation. This automated solution enhances efficiency and accuracy in structural health monitoring, offering a reliable tool for crack analysis.

Camera Selection

In photogrammetry, multiple images are captured from varying distances and orientations to reconstruct a 3D model of an object. However, not all images are suitable for crack detection and width measurement. Images taken from excessive distances often lack the resolution necessary to accurately capture fine crack details, while those not perpendicular to the surface introduce errors in crack detection and projection algorithms. To address these limitations, the proposed methodology selectively utilizes images that are relatively close to the surface and captured orthogonally to it.

Since Metashape does not provide direct tools to determine the relative distance and orientation between the camera and the object's surface, a custom algorithm was developed. The algorithm calculates these parameters by leveraging the camera positions and orientations computed during the 3D reconstruction process. To estimate the distance, the image centre is selected as a reference point. This point is projected onto the 3D surface, with the camera centre serving as the first endpoint and the projected point as the second. The Euclidean distance between these two points is then computed to determine the camera-to-surface distance.

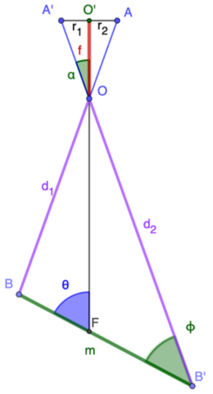

Figure 1: The resulting geometry formed by projecting two quadrants onto an imaginary planar surface

For orientation estimation, a local surface plane is defined using an imaginary small circle with radius r centred on the image. Four quadrant points on this circle are projected onto the 3D surface using the pinhole camera model, with two points aligned vertically and two horizontally. The distances between the camera centre and these projected points are calculated, yielding four distance values per image. Using these distances (d_1 and d_2) and the camera's focal length f, two similar triangles are constructed to determine the angle θ between the camera plane and the surface for each axis as seen in Figure 1. The orientation angle is derived using geometric relationships and trigonometric functions, as detailed in:

$$|m| = \sqrt{|d_1|^2 + |d_2|^2 - 2|d_1||d_2|\cos(2\alpha)} \tag{1a}$$

$$\phi = \arcsin\frac{\sin(2\alpha)|d_1|}{|m|} \tag{1b}$$

$$\theta = \phi + \alpha \tag{1c}$$

$$\alpha = \arctan\frac{r}{f} \tag{1d}$$

The smaller of the two orientation angles and the computed distance are used for further analysis. If the projection fails to intersect the 3D mesh, the algorithm returns "-1" for both distance and orientation, prompting the user to decide whether to include or exclude the image.

Binary Crack Segmentation

The core of the crack detection system lies in the automation of crack identification from images, leveraging advancements in computer vision and deep learning. Traditional methods, such as threshold-based techniques (e.g., Otsu's method) and hand-crafted feature approaches (e.g., morphological operators, wavelet filters), have laid the groundwork for crack detection but face limitations in handling complex crack patterns, uneven lighting, and noise. These methods often require manual intervention and struggle to capture the full variability of crack characteristics.

In contrast, deep learning-based approaches, particularly encoder-decoder architectures like UNet, FPN, PSPNet, and DeepLabV3, have revolutionized crack detection by enabling end-to-end training and multi-scale feature extraction. These models, supported by backbone networks such as ResNet and VGG, incorporate advanced techniques to achieve precise pixel-level segmentation. Given the need for accurate crack width measurement, which demands pixel-level precision, segmentation models were identified as the most suitable choice for this study.

A modified version of a publicly available deep learning model [4], supporting both VGG16 and ResNet101 architectures, can be trained. For this work, ResNet101, with its 101 layers, was selected due to its superior performance on larger datasets. Training was conducted using publicly available crack datasets, including pavement and concrete crack images, with the latter manually annotated to ensure relevance. Images were resized into 448×448 pixels to match the model's input size. Training was performed on a MacBook Pro with an M1 Pro processor, using default hyperparameters and a batch size of 2 due to memory constraints. The model was trained until no further improvement in validation loss was observed, with the final model selected from epoch 24 as seen in Figure 2.

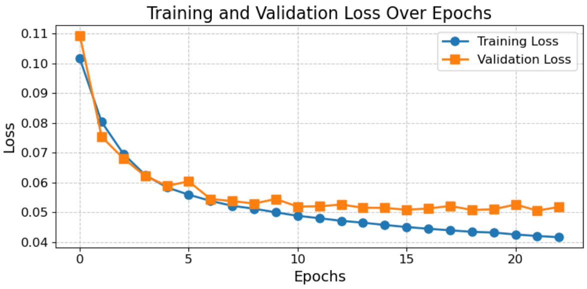

Figure 2: Training and Validation Loss Over Epochs

The trained model assigns a probability to each pixel, indicating its likelihood of belonging to a crack. An optimal confidence threshold of 0.4 was determined to balance crack detection accuracy and avoid over or underestimation.

For inference, two approaches were possible: down-sampling images to 448×448 pixels (followed by up-sampling the results) or using a sliding window with 50% overlap to process images at their native resolution. The latter approach was used because it preserves the original resolution and leverages the full detail level of the input images. To ensure accurate detection near patch boundaries, symmetric padding, which mirrors edges and corners to maintain consistent contextual information was applied, enhancing segmentation accuracy at image margins.

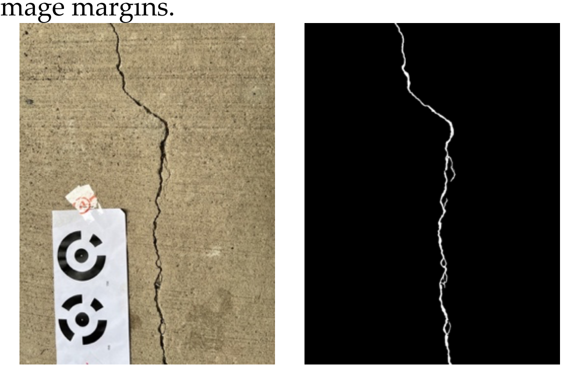

Figure 3: Original crack image with final binarization result

Post-processing steps, such as blob removal, were applied to eliminate small artifacts misclassified as cracks. Proposed binarization example can be seen in Figure 3. It is found that the method's performance declined for wider cracks or whose resolution approached the window size (448x448 pixels), where down-sampling segmentation yielded better results.

Selecting Crack Edge Twins

The next critical step involves defining crack edge points and their corresponding twin points to accurately measure crack width. An advanced algorithm proposed in prior research has been adapted and integrated into the current system [5]. The crack skeleton is generated using a thinning algorithm by Lee, which, when applied directly, often produces unwanted artifacts known as burrs due to the complex geometry of the crack. To address this, the Discrete Curve Evolution (DCE) method is employed to prune these artifacts by simplifying the crack contour. While simplification is necessary for effective pruning, excessive simplification can compromise geometric accuracy. Through trial and error, an optimal simplification level was determined for this study, balancing artifact removal with the preservation of essential geometric details.

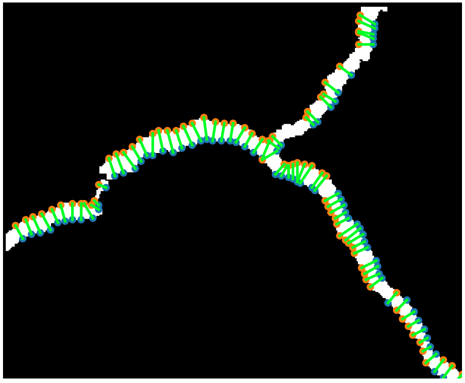

Figure 4: Crack twins selected with proposed method

Crack edges are detected using the Canny edge detection algorithm. With both the skeleton and edges identified, twin points representing the crack width are determined using a hybrid approach that combines two existing techniques: the shortest method and the orthogonal method. The shortest method selects edge points closest to the skeleton, while the orthogonal method uses a window around the skeleton point and identifies edges in a direction orthogonal to the skeleton's propagation [5].

To handle edge effects, a thin black border is added to the binary image. This border does not interfere with twin point selection, as the algorithm inherently avoids selecting points along these borders due to its reliance on orthogonality. The developed algorithm effectively identifies twin points, as demonstrated in Figure 4.

Projection Crack Edges on 3D

A key objective of this study is to utilize 3D width calculations to eliminate distortions caused by non-planar complex surfaces and to determine the 3D position of cracks for a comprehensive understanding of their location within large structures. To achieve this, the twin crack edges identified in the binary image are projected onto the 3D mesh model in Metashape. Overall projection process can be seen in Figure 5.

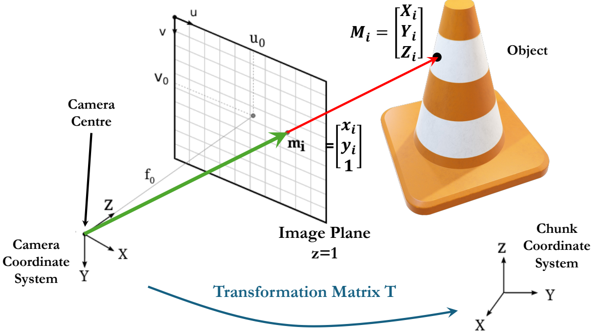

Figure 5: Proposed projection process on Metashape

In Metashape, three surface representations are available for projection: tie points, dense point clouds, and mesh models. Tie points and dense point clouds, while useful for initial reconstruction, represent the object as discrete points, leading to potential inaccuracies in projection. In contrast, the mesh model, composed of small triangular surfaces, allows for precise selection of any point on the surface, making it the preferred representation for crack edge projection. The projection process begins by identifying the pixel coordinates (u, v) of the crack edge relative to the image's top-left corner. These coordinates are undistorted using camera parameters and converted into a 3D point (x, y, z) in the camera coordinate system, with z set to 1. The 3D point is then expressed in homogeneous coordinates to facilitate transformations, including translation and rotation. Metashape internally computes the required transformation matrix, which converts the 3D point from the camera coordinate system to the chunk (world) coordinate system. This conversion can be presented mathematically as:

$$\begin{bmatrix} u \\ v \end{bmatrix} \rightarrow \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} x \\ y \\ 1 \\ 1 \end{bmatrix} = P_{camera} \qquad (2a)$$

$$P_{chunk} = T \cdot P_{camera} \qquad (2b)$$

A ray is defined from the camera position through defined point. Metashape's built-in function identifies the first intersection of this ray with the mesh surface, corresponding to the projected crack edge point on the 3D model. This process is repeated for both twin crack edges, enabling the identification of corresponding points on either side of the crack. The Euclidean distance between these points, scaled to reflect the actual proportions of the object, provides measurement of the crack width which can be seen in the Metashape user interface Figure 6.

If the ray corresponding to either twin edge fails to intersect the surface, the function returns a "Null" value, indicating that the crack does not exist on the target object.

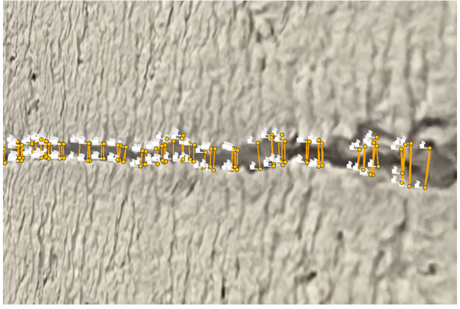

Figure 6 Estimated crack edges and widths in Metashape UI (User Interface). Points with white flags presents the crack edge while the yellow lines between the flags denotes the crack width

Exporting Results

While Metashape excels at photogrammetric processing, its interface lacks optimized tools for crack visualization. The textured mesh (exported as an .obj file with local coordinates) and the Python-generated .txt file of crack midpoints and widths are combined in CloudCompare, enabling enhanced 3D visualization and analysis of cracks."

4. Laboratory Test

To evaluate the crack detection system, lab tests used an Autoclaved Aerated Concrete (AAC) block split centrally to simulate a straight crack. An adjustable aluminium skeleton varied the crack width, and a laser sensor measured displacement to validate manual readings.

Targets were placed on the AAC block and surrounding desk to ensure accurate scaling of the 3D scene. Two cameras, an iPhone 13 Pro and a Nikon D5000, were used to capture images from various distances and angles, ensuring sufficient overlap for photogrammetric reconstruction. The images were processed in Agisoft Metashape to generate 3D scenes, and the proposed crack detection algorithm was applied to all images to analyse the impact of camera orientation and distance on measurement accuracy.

Manual crack measurements were conducted using a microscope camera and a crack gauge, providing a benchmark for comparison. The crack widths were measured at multiple points along the crack, and statistical methods were applied to compare these results with the automated measurements. To align the manual and estimated measurements, Principal Component Analysis (PCA) was used to reduce the three-dimensional positional data of the estimated measurements to a single axis representing crack propagation from 0% to 100%. This alignment allowed for a direct comparison between the manual and estimated measurements along the length of the crack.

The comparison process involved shifting the origin of the estimated measurements to one end of the crack and, in some cases, reversing the dataset to ensure proper alignment with the manual measurements. The mean difference and standard deviation between the manual and estimated measurements were calculated to evaluate the accuracy of the system. A low mean difference, coupled with a low standard deviation, indicated a strong correlation between the two datasets for each image.

Based on the analysis, an orientation threshold was established to enhance the reliability of the crack width measurements. It was determined that images captured with the camera's optical axis at an angle of more than 70 degrees relative to the surface produced the most reliable results. This threshold ensured that only images with near-orthogonal views of the crack were used, minimizing errors caused by perspective foreshortening and crack segmentation algorithm.

The Ground Sampling Distance (GSD), calculated based on camera parameters, was identified as another critical factor influencing measurement precision. Through systematic variation of image resolutions across selected images, an optimal GSD range of 0.09–0.15 mm/pixel was established to maximize crack detection precision It is found that ensuring that cracks were represented by 10 to 30 pixels in width is optimal range for this test. By implementing orientation and GSD thresholds, the system was able to pre-select images likely to yield accurate results, enhancing reliability.

5. Discussion

Key insights about proposed crack detection system include:

- Scalability challenges arise with very large images or cracks exceeding predefined processing window sizes, necessitating further training for complex conditions.
- Non-orthogonal camera angles introduce skeleton perspective misalignment, distorting crack geometry and measurement accuracy.
- Manual validation lacks positional precision and struggles with edge definition in spalled regions, requiring crack width distribution analysis.
- Smartphones offer accessibility for non-experts, while reflex cameras demand expertise but yield higher precision.

References

[1] G. Li, S. He, Y. Ju and K. Du, "Long-distance precision inspection method for bridge cracks with image processing," *Automation in Construction*, vol. 41, pp. 83-95, 2013.

[2] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei and H. Ling, "Feature Pyramid and Hierarchical Boosting Network for Pavement Crack Detection," *IEEE Transaction on Intelligent Transportation System*, vol. 99, pp. 1-11, 2019.

[3] S. Dorafshan, R. Thomas and M. Maguire, "Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete," *Construction and Building Materials*, vol. 186, no. 3, pp. 1031-1045, 2018.

[4] khanhha, "Crack Segmentation," 2020. [Online]. [Accessed 10 12 2024]. URL: https://github.com/khanhha/crack_segmentation.

[5] J. C. Ong, M.-Z. P. Ismadi and X. Wang, "A hybrid method for pavement crack width measurement," *Measurement*, vol. 197, 2022.