



Semi-autonomous inspection for concrete structures using digital models and a hybrid approach based on deep learning and photogrammetry

Ali Mirzazade¹ · Cosmin Popescu^{1,2} · Jaime Gonzalez-Libreros¹ · Thomas Blanksvård¹ · Björn Täljsten¹ · Gabriel Sas¹

Received: 27 June 2022 / Accepted: 16 January 2023 / Published online: 30 January 2023
© The Author(s) 2023

Abstract

Bridge inspections are relied heavily on visual inspection, and usually conducted within limited time windows, typically at night, to minimize their impact on traffic. This makes it difficult to inspect every meter of the structure, especially for large-scale bridges with hard-to-access areas, which creates a risk of missing serious defects or even safety hazards. This paper presents a new technique for the semi-automated damage detection in tunnel linings and bridges using a hybrid approach based on photogrammetry and deep learning. The first approach involves using photogrammetry to reconstruct a 3D model. It is shown that a model with sub-centimeter accuracy can be obtained after noise removal. However, noise removal also reduces the point cloud density, making the 3D point cloud unsuitable for quantification of small-scale damages such as fine cracks. Therefore, the captured images are also analyzed using deep convolutional neural network (CNN) models to enable crack detection and segmentation. For this aim, in the second approach, the 3D model is generated by the output of CNN models to enable crack localization and quantification on 3D digital model. These two approaches were evaluated in separate case studies, showing that the proposed technique could be a valuable tool to assist human inspectors in detecting, localizing, and quantifying defects on concrete structures.

Keywords Bridge inspection · Computer vision · Photogrammetry · Damage detection · Damage segmentation · UAV

1 Introduction

Due to the age of the world's existing road infrastructure, damage and deterioration of existing concrete structures is becoming a major social and economic concern in many countries [1]. In fact, as of 2008, more than 35% of Europe's 220,000 bridges were over 100 years old, with only 11% being less than 10 years old [2]. At the same time, tunnel infrastructures are also becoming older, and road administrators require more capable analysis and monitoring systems to determine the health of this critical infrastructures.

The process of detecting damage on existing structures and evaluating their performance is known as Structural Health Monitoring (SHM) [3]. SHM involves the long-term observation of the physical and functional condition of the construction and identifies extent of deterioration from the previous inspections. Its main purpose is to gather information about issues such as concrete deterioration, steel rebar corrosion, water seepage, concrete cover delamination, spalling, deflection/settlement, cracks, and geometry. SHM inspections should be performed routinely to evaluate the condition of bridges, plan future structural interventions, and identify structures needing replacement [4]. Such inspections are usually based on field observations performed by a human inspector. However, they are time-consuming, and the collected data do not always provide adequate visualization of locations and/or the extent of defects [1]. The maximum period between direct visual inspections varies greatly from country to country, ranging from 6 months in Australia to 3 years in Germany, while the period between detailed visual/instrumented inspections ranges from 2 to 10 years.

✉ Ali Mirzazade
ali.mirzazade@ltu.se

¹ Structural and Fire Engineering, Department of Civil, Environmental and Natural Resources Engineering, Luleå University of Technology, Luleå, Sweden

² SINTEF Narvik AS, 8517 Narvik, Norway

In Sweden, inspections must be performed every two years, with a detailed inspection every 10 years [5].

Visual inspections rely heavily on the subjective assessments of individual inspectors, which influences their reliability and repeatability [6]. In addition, visual inspection usually does not give detailed results and must be complemented with the use of technological devices. Accordingly, a study by Graybeal et al. [7] found that at most 81% of visual inspections were assigned correctly, and Phares et al. [6] concluded that at least 48% of individual condition ratings from visual inspections were incorrect. There is, thus, a need for simple, inexpensive, and practical methods for monitoring defect propagation and geometric deviation in structures as an alternative to traditional visual inspections.

The evolution of unmanned vehicles and sensors technology has significantly improved the efficiency of structural health monitoring. Technologies such as UAV [8–11], laser scanning [4, 12–14], and photogrammetry [4, 8, 12, 13, 15, 16], leading to significant increases in the accuracy and quality of data collection for structural assessment. Modern structural analysis techniques generally do not rely on physical contact and can rapidly generate very large datasets whose analysis can provide highly accurate and reliable descriptions of a structure. These techniques have been enabled by the remarkable advances in computer power, data storage capacity, and camera sensors that have been made in recent decades. One important technique in terms of 3D visualization of structures is photogrammetry, which is a contactless optical sensing method that has received considerable attention due to its highly productive data acquisition, low cost, and ability to be used in almost any climate or environment [4, 17].

In addition to data collection, the logic behind defect detection is the other piece of puzzle leading to reach semi-autonomous structural inspection. For this aim, data driven approach offers a lot of benefits in automation, and one of its novels and rapidly emerging application is deep learning. Deep learning technique is gradually accepted by civil engineers as a powerful tool in various applications, especially for damage identification of structures. Machine learning technique is one of the most efficient techniques for damage identification, able to imitate intelligent human learning without following explicit instructions. Image-based damage detection using deep learning algorithms has emerged as a powerful technique for SHM in recent years [18, 19]. This method has achieved remarkable performance in the crack detection [20–25], road/pavement inspection [26–31], corrosion detection [32], and overall condition assessment [33] with multiple damage types [34, 35]. However, the introduction of this application was a classification approach for damage identification, presented by Rytter [36], and defined in four levels: (1) Detection: Determination whether or not the damage is existed in the structure,

(2) Localization: Determination of the geometrical location/position of detected damage, (3) Quantification: Extension and/or quantification of the severity of detected and localized damages, and (4) Prediction of the remaining service life (RSL) of the structure.

1.1 Image-based 3D reconstruction for civil infrastructure

Structure from Motion (SfM) is currently the most popular photogrammetric technique for generating a 3D model of a structure. It involves capturing numerous multi-view images of the structure of interest from the ground or the air, which can be done using affordable non-metric cameras. The resulting 3D models can be used to perform SHM on a computer, remotely, without the safety and time constraints associated with direct visual inspections. For a detailed explanation of the theoretical principles of image-based 3D reconstruction algorithms, interested readers are referred to the work of Remondino et al. [37].

Broome [38] has shown that there is no significant difference between accuracy of close-range photogrammetry (CRP) and terrestrial laser scanning (TLS); the distances measured using the two methods differed by only 0–7 mm. In addition, a cost benefit analysis showed that CRP is a far more cost-effective method overall because it requires substantially less expensive equipment. However, TLS was more accurate overall when performed by a skilled operator [38]. Point clouds used for infrastructure inspection must have sufficient accuracy and density to represent the kinds of small-scale visual details that inspectors look for during an inspection, which are often less than a millimeter in size. However, generated point clouds usually include missing data, inaccurate geometric positioning [39], surface deviations [40], and outlier-based noise [41]; each of these noise types is described in detail by Chen et al. [42]. In more detailed studies for small-scale defect detection by point clouds data, Valenca et al. [43] have shown that cracks with widths of 1.25 mm can be detected using a TLS scanner if the scanning parameters are set properly. However, in real built structures, it is difficult to monitor cracks at submillimeter scales because they may be covered by dirt and moisture stains [43]. Therefore, data acquisition should only be performed after briefly checking the site, removing dust from spaces and surfaces, clearing shady vegetation, and ensuring access to designated spaces.

Due to the clear advantages of remote structural inspection using photogrammetry, it has been investigated by several researchers. For instance, Jahanshahi and Masri [44] used computer vision and image processing algorithms to develop a technique for crack detection using 3D reconstruction. In addition to damage detection, geometrical deviations must be monitored during routine inspections. This

can be achieved by analyzing point cloud datasets, which are typically generated by laser scanning or photogrammetry. It has been shown that image-based 3D reconstruction is an inexpensive and efficient method for 3D reconstruction [45], although its achievable accuracy is lower than that of terrestrial laser scanning (TLS) [4]. However, Kwak et al. [46] showed that submillimeter precision in the estimation of vertical deflections and horizontal displacements could be achieved by combining two photogrammetric reconstruction techniques, namely image-matching-based reconstruction, and model-based image fitting. This approach enabled a high level of automation while delivering a root-mean-square error (RMSE) of 0.5 mm.

Another important advantage of image-based 3D reconstruction is its potential for automation [4, 17], which can be facilitated using unmanned aerial vehicles (UAVs) for image acquisition. Image acquisition using UAVs is valuable because it can facilitate data acquisition from both areas that are likely to be damage-prone and areas that are hard to access. Moreover, UAVs allow images of regions of interest (ROI) to be captured at close range (and, thus, at high resolution), leading to improved pattern recognition and more accurate models in terms of geometric dimensions. Close-range images also provide more information on local structural details and improve feature-matching processes. However, procedures based on close-range imaging have the drawback of requiring the capture and analysis of more images than alternatives based on longer-range imaging, leading to higher computational costs. To overcome this problem, the hierarchical Dense Structure-from-Motion (DSfM) method, which was designed for use with UAV imaging systems, was proposed by Khaloo and Lattanzi [47]. Khaloo et al. [48] described the use of UAVs to generate 3D models with sufficient accuracy to detect defects on an 85 m-long timber truss bridge. The UAV-based method was found to outperform laser scanning with respect to the quality of the captured point clouds, the local noise level, and the ability to render damaged connections. However, neither TLS nor conventional DSfM have yet proven capable of generating point clouds accurate enough to resolve structural flaws on scales of 0.1 mm (the minimum dimension of a hairline crack) while simultaneously capturing a structure's overall geometry.

Morgenthala et al. [8] presented a framework for automated UAV-based condition assessment inspections of bridges that encompasses flight path planning, structural surface model reconstruction, and surface defect detection. Additionally, Chen et al. [42] provided a case study of UAV bridge inspection by 3D reconstruction and discussed quality evaluation mechanism for 3D point cloud.

Once generated, 3D models can be analyzed using deep convolutional neural networks (CNNs) to enable semi-automated detection, localization, and quantification of existed

defects. For example, Mirzazade et al. [10] proposed a CNN-based workflow for detecting and measuring joint openings in the abutments of a trough bridge and mapping them onto a 3D model generated by photogrammetry. The method performed well but was highly reliant on a prepared dataset for model training in both defect classification and segmentation tasks. This training dataset consisted of a large set of images of defects similar to those being inspected for in the structure of interest.

1.2 Damage detection using Convolutional Neural Networks (CNNs)

Image-based crack detection using CNN algorithms has emerged as a powerful technique for SHM in recent years [18, 19]. CNN-based image analysis methods have achieved remarkable performance in the detection of cracks [20–25], multiple damage types [26, 27], and overall condition assessment [33]. On these studies, four deep CNN architectures mostly have been used for classification purpose: VGGNet-19 [49], Inception v3 [50], GoogleNet [51], and ResNet-50 [52]. Mirzazade et al. [11] compared these four CNNs in terms of accuracy, loss, computation time, model size, and architectural depth, obtaining the results summarized in Fig. 1 [11]. Briefly, InceptionV3 achieved the highest accuracy with the prepared dataset, but its computation time was almost two times that of GoogleNet [11].

Many groups working on defect localization have approached the problem of image-based crack detection by splitting images and treating it as a sub-image classification problem. For example, Zhang et al. [27] used over 500 pavement images to train a ConvNet model that successfully recognized road cracks in a square image patch with dimensions of 99×99 pixels. Using a similar strategy, Kim et al. [53] proposed a transfer-learning network based on pre-trained R-CNN to detect cracks in a concrete bridge. Inspired by the famous CNN image classification model AlexNet [54], Kim and Cho [53] introduced an algorithm that classifies sub-images with dimensions of 227×227 pixels into four

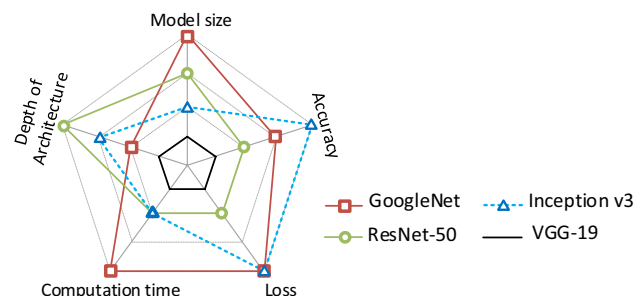


Fig. 1 Damage detection performance of four CNN architectures. In all cases, the 1st and 4th levels correspond to the worst and best performance with respect to the indicated items, respectively [11]

classes: cracks, structural joints, plants, and intact surfaces. This multi-class model and the large window size yielded a significant improvement in identification accuracy. Moreover, the use of an overlap window made it possible to narrow the crack-containing region [53]. All the studies mentioned above used the sliding window method whereby a full image of a concrete surface is divided into sub-images on which image classification is subsequently performed, allowing sub-images containing cracks to be selected and analyzed to locate the damaged area and quantify its size.

Defect quantification using CNN requires semantic segmentation of the detected crack-containing sub-images. Semantic segmentation is an important task in computer vision whose goal is pixel-wise classification. This is typically achieved using end-to-end networks consisting of two cooperative sub-networks (encoding and decoding) that classify each pixel of the image and then segment the image into distinct components based on the classifications. In the context studied here, the components of interest are detected defects.

In this work, CNNs were used to detect and segment damaged areas so that their dimensions could be determined. A tool with these capabilities could be used by inspectors to monitor damage propagation over a structure's service life. The task of pixel-wise damage detection using CNNs can be divided into two sub-tasks: (1) splitting images into multiple

smaller sub-images and performing CNN classification in each for damage localization, and (2) applying semantic segmentation on detected cracked sub-images for pixel-wise damage detection.

2 Methodology and inspection strategies

2.1 Proposed methodology

This study aims to develop new solutions for monitoring the condition of existing structures. Such solutions will allow defects to be identified at the earliest possible stage, making it possible to perform maintenance while minimizing traffic disturbances. To this end, a new approach involving generating a digital structure model followed by a semi-automated defect detection is presented. This will provide detailed information on the condition of the whole structure. Figure 2 presents an overview of the approach. First, image data are collected from different perspective (angles) on the structure. In the second step, these images are subjected to preprocessing and quality enhancement (e.g., background removal, brightness and blurring analysis) that may be needed for image-based 3D model reconstruction (see the work of Mirzazade et al. [10] for details), and photogrammetry is used to generate a 3D model of the structure (also

Fig. 2 Overview of the autonomous damage detection and quantification workflow



called digital model). In the third step, generated point cloud by photogrammetry technique is analyzed for geometrical deviation assessment. In this study, another 3D point cloud of the structure is generated by laser scanning to verify the accuracy of image-based point cloud. The point cloud serves as a reference for geometric verification of the photogrammetric model generated in step two.

Finally, in steps four and five, two Convolutional Neural Networks (CNN) are used for autonomous damage detection and pixel-wise segmentation to quantify the extent of the detected cracks in collected images. These two tasks have five distinct steps: (1) data acquisition, (2) dataset preparation, (3) designing and training CNNs, (4) damage detection and localization by splitting images into sub-images that are classified into “Crack” or “No Crack” areas using the CNN classifier, and (5) crack segmentation, mapping cracks onto a 3D model, and crack quantification. In step 5, pixel-wise crack segmentation is performed by applying U-Net semantic segmentation on 2D images that are then stitched together to reconstruct a 3D model and generate an orthophoto of

the defect-containing areas. The segmented crack can then be measured by determining the orientation of the camera’s position for each photo relative to the crack. Figure 3 shows a flowchart of the procedure.

2.2 Scientific novelty

This paper presents a pilot study that was conducted with the aim of developing inexpensive and easily deployed vision-based non-contact defect quantification solutions that (i) are suitable for field applications and (ii) could improve the inspection, monitoring, and assessment of existing bridges. Such solutions could improve the accuracy and efficiency of bridge inspections by eliminating human error, allowing damage to be detected at an early stage. This in turn will enable preventative and remedial measures to be implemented in a timely fashion and make it easy to generate historical records showing the progress of a structure’s deterioration.

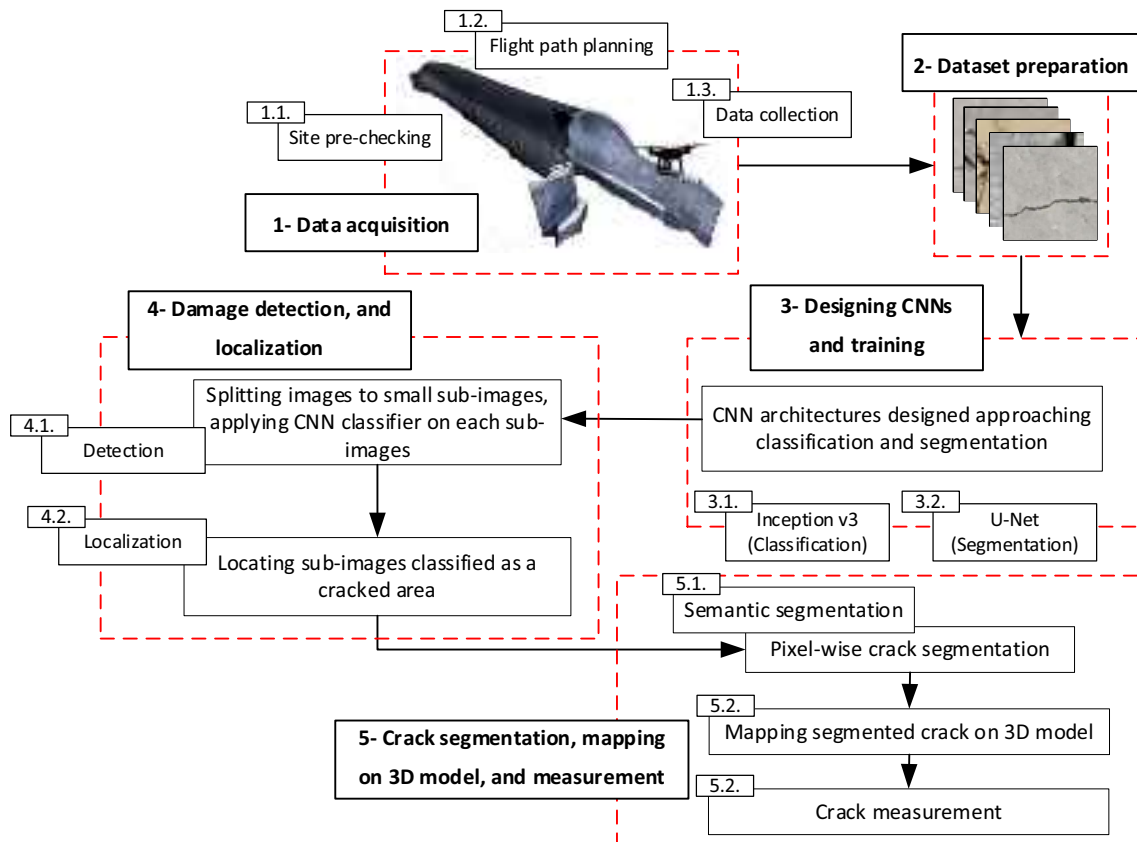


Fig. 3 Workflow of the proposed method for semi- automated damage detection and quantification

3 Field deployment

3.1 Selected case studies

3.1.1 Kedkejokk tunnel

The Kedkejokk tunnel is a concrete arch tunnel built in 1906. It is located in the Riksgränsen region (at km 1533 + 175 according to the nomenclature of Trafikverket, the Swedish Transport Authority), near the Norwegian border in northern Sweden. As shown in Fig. 4, it crosses a narrow stream of water. The tunnel width is 4.0 m and its total length is 41.2 m. It is located on a hillside, and there are steep slopes between the tunnel's foundation and existed railway. The surrounding area is sparsely vegetated with scattered small trees and bushes. The tunnel was surveyed on two consecutive days, during which the weather conditions varied from cloudy to sunny. Because the tunnel under the railway infrastructure was dark, images were captured with the camera in AV mode and the ISO was set automatically based on the light environment. It was necessary to use a shutter time of almost 20 s to capture sufficient light, which made the process time-consuming.

Fig. 4 Photos of Kedkejokk tunnel



Fig. 5 Photos of Juovajokk bridge



3.1.2 Juovajokk bridge

The Juovajokk bridge is a simply supported bridge made from reinforced concrete that was built in 1902 in the vicinity of Abisko in northern Sweden (at km 1504 + 915 using the Trafikverket nomenclature). Its superstructure was replaced in 1960, and it spans a stream as shown in Fig. 5. The bridge's span and width are 5.5 m and 3.8 m, respectively. The surrounding area is densely vegetated and there are steep slopes behind the abutments that made it difficult to balance a tripod for image acquisition. During the survey, some areas under the bridge were frozen, making the surface slippery. It took around 3.5 h to completely survey the bridge during a partially sunny morning with temperatures below 0 °C.

3.2 Data acquisition

The equipment used for TLS was a long-range RIEGL VZ-400 3D terrestrial laser scanner (see Fig. 6a). This 3D scanner operates on the time-of-flight principle and can perform measurements at distances between 1.5 and 600 m with a nominal accuracy of 5 mm at 100 m. It uses near-infrared laser wavelengths with a laser beam divergence of

Fig. 6 Data acquisition equipment. **a** RIEGL VZ-400, 3D terrestrial laser scanner, **b** Canon EOS6D Mark II digital camera with lenses



Table 1 Interior orientation of the sensor used for CRP

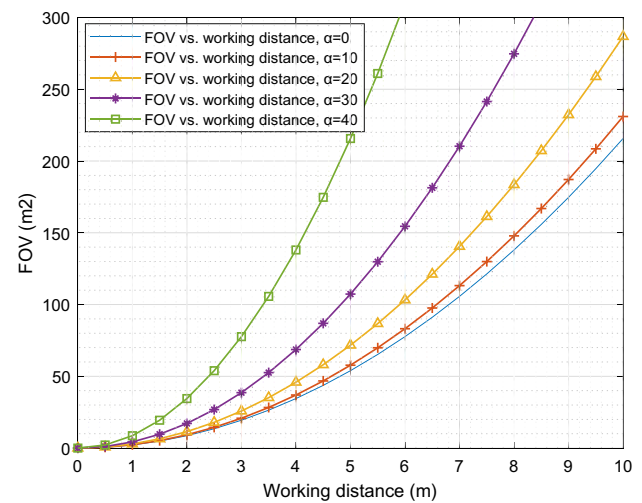
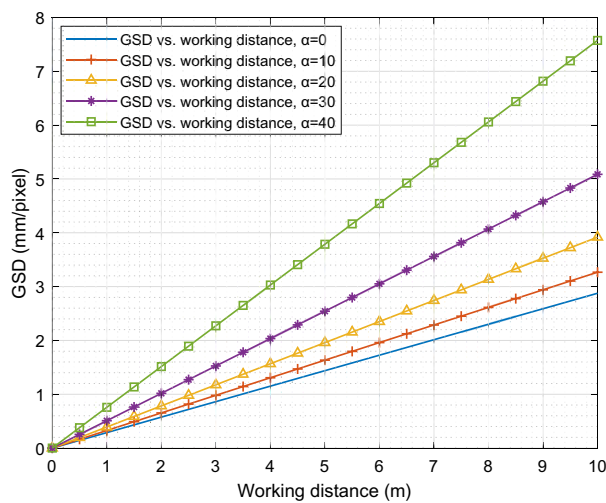
Lens	24 mm	20 mm
Aperture	f/2.8–f/22	f/2.8–f/22
Diffraction-limited aperture	f/9.3	
Closest focusing distance	0.66/0.20 m	0.25 m (0.82 ft)
Horizontal viewing angle	74°	84°
Diagonal viewing angle	84°	94°
Vertical viewing angle	53°	62°
Filter diameter	58 mm	72 mm
Focal length	24 mm	20 mm
Pixel dimensions	6240 × 4160	
Pixel size	5.75 μ m	
Sensor size	35.9 × 24.0 mm	
Shutter speed	30–1/4000 s in 1/3 stop increments	
Frame rate	29.97 fps	

0.3 milliradians (mrad), corresponding to a beam diameter increase of 30 mm per 100 m of distance. The instrument's maximum vertical and horizontal scan angle ranges are 100° and 360°, respectively. The raw TLS data, i.e., point clouds captured from multiple scans, were post-processed (registered and geo-referenced) using the Leica Cyclone software package, which automatically aligns the scans and exports the point cloud in various formats for further processing.

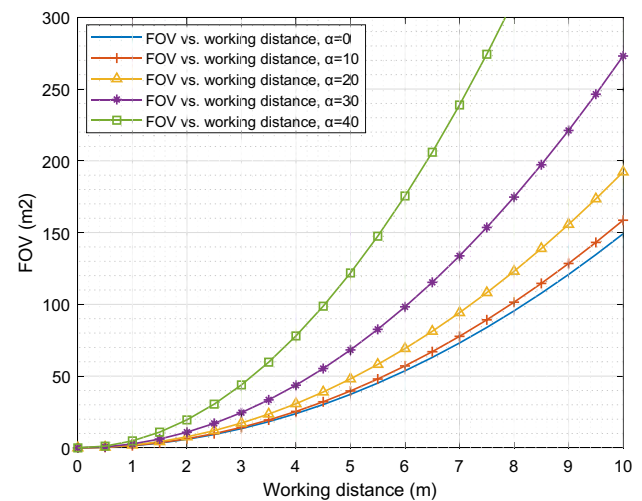
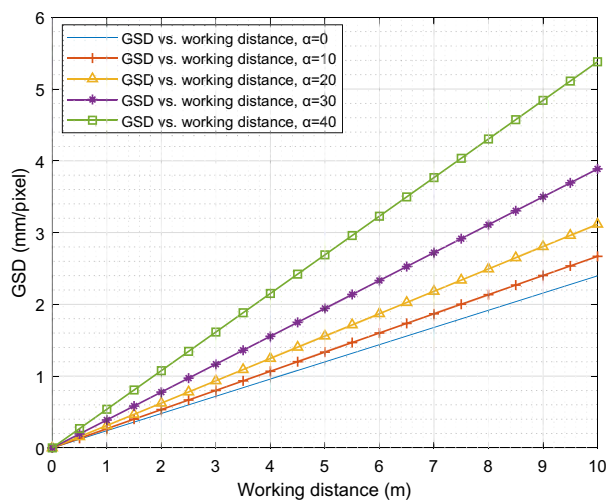
The equipment used for CRP (Fig. 6b) consisted of a Canon EOS6D Mark II DSLR camera with a full-frame complementary metal–oxide–semiconductor (CMOS) optical sensor giving a resolution of 12.8 megapixels. The camera was equipped with Canon EF 24 mm and 20 mm wide-angle prime lenses; its interior orientation is specified in Table 1.

Image acquisition is the first step in image-based 3D reconstruction. The acquired images were fed into a commercial SfM software package, Agisoft PhotoScan Pro (LLC, 2017) that simultaneously determines the camera's interior orientation and defines parameters relating to its exterior orientation, such as the camera's angle and the working distance relative to the scanned object. Images of the two case studies were captured from several points of view corresponding to different working distances between the camera and the bridge; the working distance is arguably the variable with the greatest impact on data quality. The information presented in Table 1 was used to calculate the Ground Sampling Distance (GSD) and Field of View (FOV) as functions of the working distance for both lenses (see Fig. 7a, b). The GSD is the distance between the centers of two consecutive pixels on the target surface. Smaller GSD values correspond to higher resolutions and are therefore preferred. However, in practice it is also necessary to consider the FOV value because a larger FOV minimizes the number of images that must be captured (see the work of Chen et al. [42] for further details).

After calculating the GSD and FOV, an appropriate Working Distance (WD) and tilt angle (α) can be selected to match the surveying objectives for image collection. The graphs presented in Fig. 7a, b show that at any given working distance, the 24 mm lens gave a lower GSD (and, thus, a higher resolution leading to the capture of more detail) than the 20 mm lens. However, its FOV is lower than that of the 20 mm lens, making it necessary to acquire and process more images to cover all the surfaces of the structure. Consequently, more processing time and resources are needed to generate the 3D model when scanning with the 24 mm lens. To strike an optimal balance between processing time and resolution, a zoom lens can be used for data acquisition. A zoom lens is a mechanical assembly of lens elements whose focal length can be varied to maintain a consistent GSD at different working distances, which is



a. GSD and FOV as functions of the working distance for the 20 mm lens



b. GSD and FOV as functions of the working distance for the 24 mm lens

Fig. 7 **a** GSD and FOV as functions of the working distance for the 20 mm lens. **b** GSD and FOV as functions of the working distance for the 24 mm lens

important when performing hierarchical Dense Structure from Motion (DSfM). The collected images were used for 3D model generation as described below.

4 3D model generation

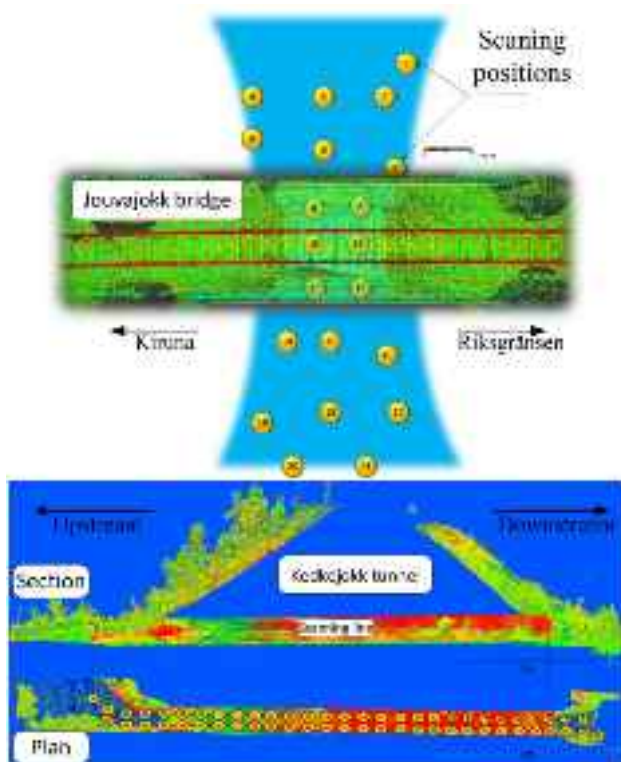
Both bridges were scanned on cloudy to sunny days and 3D point clouds were successfully generated from both the TLS and CRP scanning data. Table 2 shows the scanning duration, hardware, and software used for this purpose as well as the main challenges encountered during inspection and

modeling in each case study. Figure 8 shows the locations where the camera was set up for CRP data acquisition in each case. Good performance was achieved when using CRP due to its easy set-up and high productivity.

Both generated point clouds were imported into Autodesk ReCap to extract measurements of the bridges' structural elements (see Fig. 9). Measurements of the bridges' structural elements can be found in as-built drawings to obtain ground truth as a reference model; however, in Kedrejokk tunnel, due to the performed repairment and new installed lining, the existed as-built is totally different compared with current condition. Therefore, TLS model

Table 2 Scanning details for the Kedkejokk tunnel and Juovajokk bridge

	Kedkejokk tunnel	Juovajokk bridge
Time duration	<ul style="list-style-type: none"> • 12 h (CRP) • 2 h (TLS) 	<ul style="list-style-type: none"> • 3 h and 30 min (CRP) • 1 h (TLS)
Hardware	<ul style="list-style-type: none"> • Canon EOS6D digital camera • Canon EF 24 mm, and 20 mm wide-angle prime lens • RIEGL VZ-400 	
Software	<ul style="list-style-type: none"> • Agisoft PhotoScan Pro (LLC, 2017) • Leica Cyclone software package 	
Distance to construction	1.5–2 m	2–4 m
Difficulties; inspection	<ul style="list-style-type: none"> • Light condition • Narrow space • Rush flow and low temperature of water 	<ul style="list-style-type: none"> • Hard to access areas • Slippery areas due to frozen water
Difficulties; Modeling	<ul style="list-style-type: none"> • Changing in environmental light due to long image acquisition period of time 	<ul style="list-style-type: none"> • Light reflection on pattern-free and smooth surfaces

**Fig. 8** Photogrammetric scanning positions for the Juovajokk bridge and Kedkejokk tunnel

considered as an updated and more reliable ground truth for both case studies, Table 3. Therefore, last column of Table 3 shows higher geometric deviation for the Kedkejokk tunnel compared with that of Juovajokk bridge, which is due to the geometry of tunnel, which is longer in axis direction versus others, in addition to the poor light conditions in tunnel. The need for suitable light conditions and scale bars for large structures (especially those

that are long in one direction like the Kedkejokk tunnel) to minimize geometric deviation are notable weaknesses of the CRP method when compared to TLS. The effects of geometric deviations in assessments of existing concrete bridges were discussed in detail by Mirzazade et al. [17]. Overall, the image-based generated 3D models provided information, with less than 1% error, for bridge inspection, especially in hard-to-access areas, while presenting minimal risks and safety issues. In the next part of the paper, the Juovajokk bridge is used as a case study for 3D model quality assessment because it has a variety of surfaces, materials, and geometric shapes that present different challenges in 3D model reconstruction. In addition, the Kedkejokk tunnel is examined as a case study for autonomous damage detection, segmentation, and quantification due to its poor lighting and hard-to-access areas present significant difficulties for human inspectors.

4.1 Data quality evaluation

The Structure-from-Motion process (SfM) starts with image acquisition, determines the interior orientation, and defines parameters relating to the exterior orientation of the camera, such as camera angle and work distance, relative to the scanned object. However, generated model usually include missing data, inaccurate geometric positioning, surface deviations, and outlier-based noise. Fig. 10 shows a confidence model illustrating the reliability of the image-based point cloud for each part of the model from Juovajokk bridge; warm colors indicate noisy parts while cold colors indicate areas where the confidence in the generated point cloud is relatively high.

Fig. 9 Measurements of bridge structural elements by CRP (left), and laser scanning (right)



Table 3 Accuracy of 3D models generated using TLS and CRP

	Component	As-built dimension (mm)	TLS		CRP		TLS vs. CRP (Ref.: TLS) %ΔL
			(mm)	%ΔL	(mm)	%ΔL	
Juovajokk bridge	Span	5500	5432	− 1.2%	5456	− 0.80%	0.44%
	Width (deck)	3800	3805	0.13%	3810	0.26%	0.13%
Kedkejokk tunnel	Width ^a	4000	2750	−	2774	−	0.87%
	Length ^a	41,200	44,014	−	44,156	−	0.32%

^aOriginal values before repair are as-built dimension, but after lining replacement, tunnel width is changed

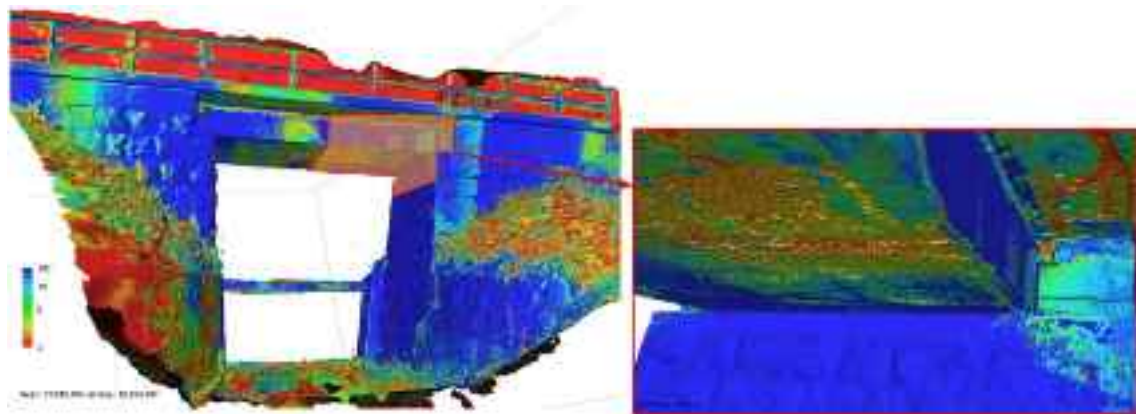


Fig. 10 Confidence model of the Juovajokk Bridge. Blue parts have reliable triangulation while red parts have comparatively high noise levels

4.1.1 Incomplete data

In 3D model reconstruction, missing data give rise to areas with poor overlap, especially for slim or narrow

parts of the structure such as struts (Fig. 11) or cables, because of a lack of sufficient features for image matching.

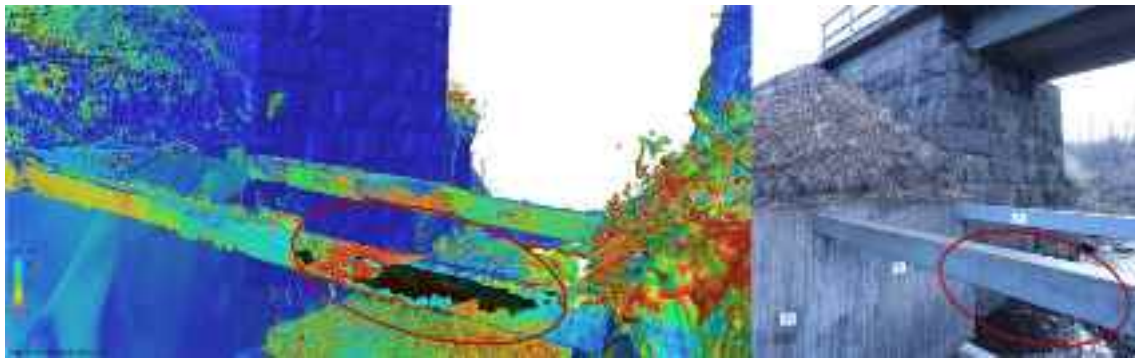


Fig. 11 An example of a missing data problem resulting from a lack of features on the struts

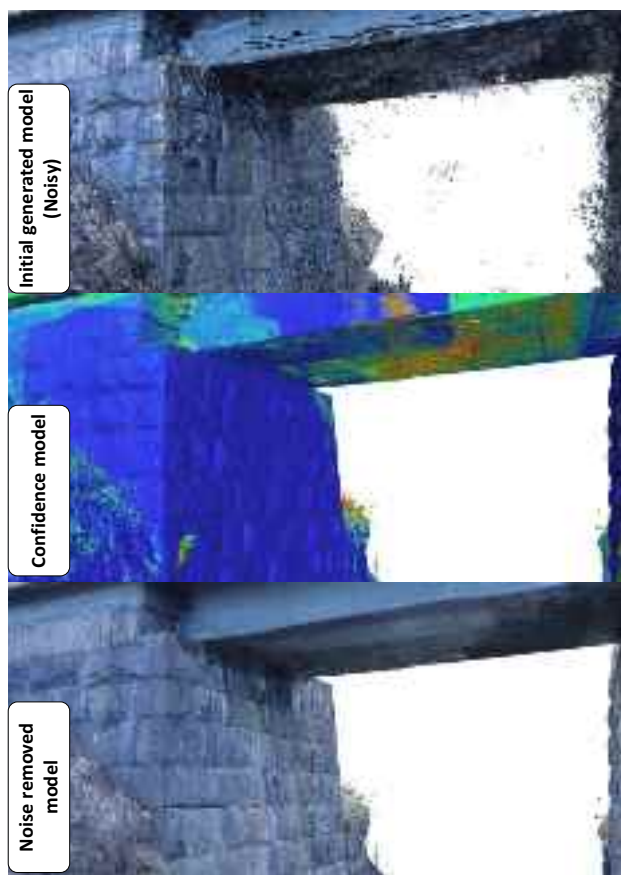


Fig. 12 Removal of outlier noise from the initially generated model by Agisoft PhotoScan Pro (Agisoft LLC, 2017)

4.1.2 Outlier noise and surface deviation

Outlier noise usually appears around the boundary of the structure because textureless backgrounds (like the sky) tend to confuse 3D reconstruction approaches. For example, the area underneath the bridge in Fig. 12 is poorly reconstructed because the reconstruction algorithm treats the background

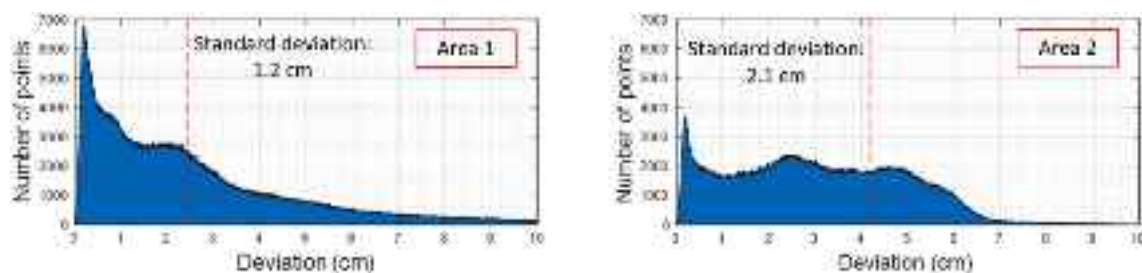
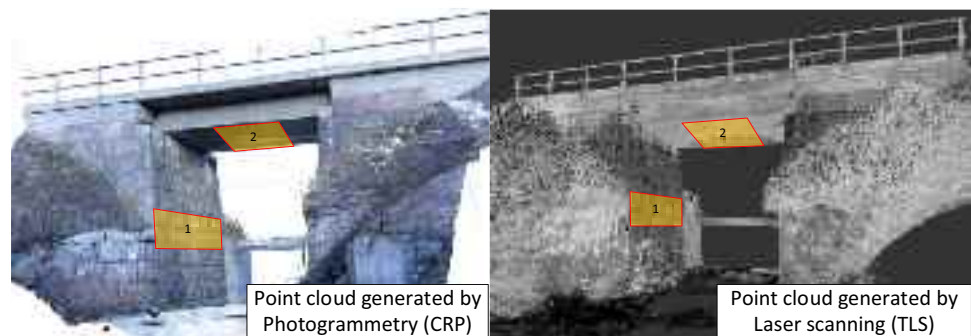
(sky) as part of the front object (bridge). Furthermore, since the camera failed to fully observe the area beneath the beam, many outliers appear around the border. Those outlier points will affect subsequent surface reconstructions and generate floating artifacts around the object. In addition, shadows and large tilt angles can weaken or hide surface textures, making this part noisier. Commercial software tools such as Agisoft PhotoScan Pro (Agisoft LLC, 2017), which was used in this work, include outlier noise removal as part of the standard rendering procedure (see Fig. 12).

Assessment by the naked eye reveals little difference in quality between the final rendered CRP and TLS models. Table 4 lists the point cloud densities achieved with each scanning methods for the two areas shown in Fig. 13.

These results showed that a higher point-cloud density does not necessarily yield a more detailed model with a higher resolution. Despite the high point-cloud density achieved using CRP, the resulting model had surface deviations, came from outlier noises, that had to be mitigated during post-processing. Figure 14 shows the point cloud deviation of the photogrammetric model relative to the laser scanning model for the two areas shown in Fig. 12. These graphs were generated by aligning the CRP data to the TLS data using the iterative closest point (ICP) algorithm of Besl and McKay [55], and calculating the distance between specific points in each set. For each point in the CRP data set, a search was performed to find the nearest neighbor point in the TLS data set, and the offset distance between these two points was recorded. As shown by the graphs in Fig. 14, the standard deviation of the resulting cloud-to-cloud distance maps was highest for the area underneath the bridge (area 2), which is mostly due to this area's poor lighting and the lack of distinct features in the captured images. Outlier noise can be removed by applying a threshold for detecting outlier points with excessive deviations. As discussed previously [17], a reasonable threshold for this purpose is equal to the mean cloud-to-cloud distance plus/minus twice the standard deviation.

Table 4 Resolutions achieved by CRP and TLS for two different areas of the Juovajokk bridge

Area	Intended area (m ²)	Reconstruction method	Number of points	Local point density (points/m ²)
1-Abutment	8.23	TLS	1,297,370	157,639
		CRP	1,883,544	228,863
2-Underneath	13.97	TLS	1,342,556	96,102
		CRP	1,391,184	99,583

Fig. 13 Areas of the abutment and underneath of the bridge used to evaluate the point cloud densities of the CRP and TLS models**Fig. 14** Absolute cloud-to-cloud deviations between the CRP and TLS point clouds, using the TLS cloud as a reference, for areas 1 (abutment) and 2 (underneath) of the Juovajokk bridge**Table 5** Absolute numbers of deviating points in the CRP clouds, percentage of surface deviation, and local point-cloud density after noise removal for areas 1 and 2 of the Juovajokk bridge when applying different noise removal thresholds

Threshold for surface deviation noises	Absolute number of deviating points		Percentage of surface deviation noise		Local point density after noise removal (points/m ²)	
	Area 1 (1,883,544 points)	Area 2 (1,391,184 points)	Area 1	Area 2	Area 1	Area 2
Up to 2 cm	938,778	943,934	49.8%	67.8%	114,889	32,065
Up to 2.4 cm	793,496	839,322	42.1%	60.3%	132,511	39,534
Up to 4 cm	438,727	459,156	23.3%	33.0%	175,537	66,720
Up to 4.2 cm	411,144	418,073	21.8%	30.0%	178,970	69,708
Up to 5 cm	313,327	238,415	16.6%	17.1%	190,871	82,554

Therefore, taking the TLS point cloud as the ground truth, all points for which the absolute cloud-to-cloud distance was more than twice the standard deviation were filtered out as surface deviation noise. Finally, the degree of surface deviation in the two studied areas was calculated as

the ratio of the number of filtered points in each area to the total number of points in the same area (see Table 5).

Table 5 shows the degree of surface deviation in the CRP clouds for areas 1 and 2 when applying different noise removal thresholds. The cells with gray shading show the

results obtained when applying the suggested threshold of twice the standard deviation for the area under consideration. As expected, the local point density in both areas of the CRP cloud after noise removal is lower than that of the TLS cloud, and lowering the threshold to increase noise removal exacerbates this difference. This means that while these CRP point clouds could be used to monitor geometric deviations of the structure, they cannot be used for small-scale damage detection due to their low point density. However, this is not a severe limitation because damage detection using deep CNN is performed using image data rather than a 3D point cloud.

A representative cloud-to-cloud distance map, to illustrate outlier noises, is shown in Fig. 15. In this figure, points with

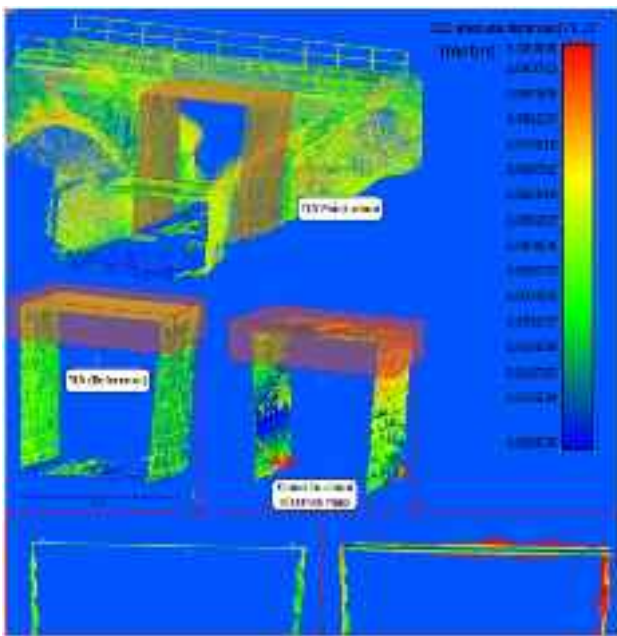


Fig. 15 A cloud-to-cloud absolute distance map for the region underneath the bridge

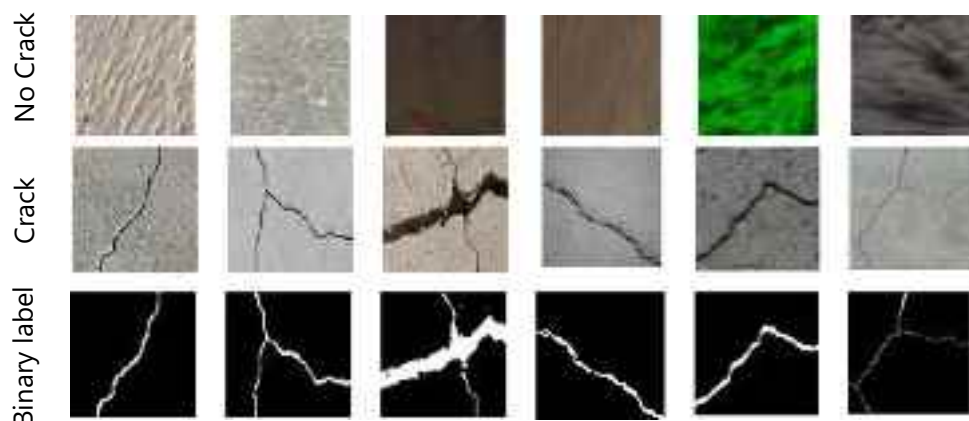
warmer colors have higher deviations. Surface deviations can be filtered out by applying a threshold based on the absolute cloud-to-cloud distance.

5 Damage detection

The Kedkejokk tunnel was used as a case study to test the capabilities of deep learning algorithms for crack detection and quantification in hard-to-access areas with poor lighting. Then after data acquisition, which is discussed in Sect. 3.2, datasets for training CNNs must be pre-processed. The training dataset used in this work consists of 40,000 images with RGB channels (227×227 pixels each) divided into two classes (Crack and No Crack) comprising 20,000 images each. The dataset was generated by Özgenel et al. [56] from 458 high-resolution images (4032×3024 pixels) of concrete surfaces at the METU campus; the high-resolution images were converted into a much larger set of smaller images using the method proposed by Zhang et al. [27]. Two models were trained using this dataset: (1) a classification CNN model based on the Inception V3 architecture, and (2) a semantic segmentation with an end-to-end CNN based on the U-Net architecture that was trained using binary segmented crack images. Some representative images from the dataset are shown in Fig. 16.

Before training, the datasets were divided into training and validation groups comprising 70% and 30% of the total data, respectively. The datasets were augmented with randomly cropped and rescaled ($1 \times$ to $2 \times$) images to enable learning of important features at different scales and positions. Random rotation (between 0° and 90°) and random mirroring in both X and Y directions were also performed. Data normalization was then performed to avoid undesired bias due to the inclusion of high-frequency information. This ensured that the data frequency was normally distributed with a mean of 0 and a variance of 1.

Fig. 16 Representative images from the dataset prepared by Özgenel et al. (2018) [56] that was used for training both CNNs



Using the result of previous studies by authors [10, 11] and based on obtained iterative optimization of gradient descent for similar dataset, optimized hyperparameters and needed training epochs to avoid underfitting were considered. Therefore, fifteen training epochs were performed for the classification CNN model based on Inception V3, while 5 and 8 training epochs were performed for the segmentation models based on U-Net and SegNet, respectively. The learning rate was assumed to be 0.001 and the mini-batch size was set to 128 and 1 images for image classification and segmentation, respectively. The verification frequency was set to 20 iterations, and training was performed using an Intel® Core™ i9-9880H CPU running at 2.30 GHz.

5.1 Crack localization

All of the captured images were divided into sub-images with dimensions of 227×227 pixels and those containing cracks were detected by the CNN classification model. Bounding boxes were then drawn around sub-images identified as potentially cracked regions. A trade-off must be struck when determining the size of the sub-images; a smaller size will increase the precision of localization, but as the image becomes more finely divided, the information content of the sub-images declines, creating a risk that they may lack sufficient information to confidently determine whether damage is present. For each sub-image, an overlap region with a thickness of 15 pixels in the vertical and horizontal directions was defined to avoid risk of missing data on the borders of the sub-images.

In general, increasing the number of training epochs increases the training accuracy and, thus, reduces the training loss. However, too many training epochs may cause the model to overfit the training data. In other word, model does not learn the training dataset and memorizes that. Therefore, validation accuracy is found for each epoch to investigate whether it overfits or not. Figure 17 shows how the training accuracy and training loss varied with the number of epochs when training the InceptionV3 CNN-based classifier. After training, the model's confusion

matrix was generated using the test dataset (which comprised 30% of the total dataset) to determine the number of true and false positive and negative predictions that were obtained (see Fig. 17).

5.2 Crack segmentation

Semantic segmentation is an important task in computer vision whose goal is pixel-wise segmentation. This is typically performed using an end-to-end network consisting of two cooperative sub-networks (encoding and decoding) classifying individual pixels as Crack or No Crack areas. A previous study [10] comparing the performance of U-Net [57] and SegNet [58] for semantic segmentation of small-scale block openings concluded that U-Net offered better performance. In this work, both models were tested for crack segmentation. Figure 18 shows the corresponding training accuracy and loss graphs.

As found in the earlier study, U-Net achieved a higher overall accuracy than SegNet and required fewer training epochs to achieve good accuracy. Figure 19 shows the pixel-wise crack segmentation results obtained using the U-Net and SegNet models after 5 and 8 training epochs, respectively. Because the U-Net model achieved better crack segmentation performance even with less training epochs, it was applied to all sub-images that were found to contain cracks using the classifier model to segment the defects. Final images were then generated by merging the processed sub-images.

Since crack quantification was performed by counting pixels, it was important to verify the quality of the semantic segmentation by matching it to ground truth data. Table 6 shows the metrics used to evaluate crack segmentation by U-Net and SegNet with respect to the ground truth.

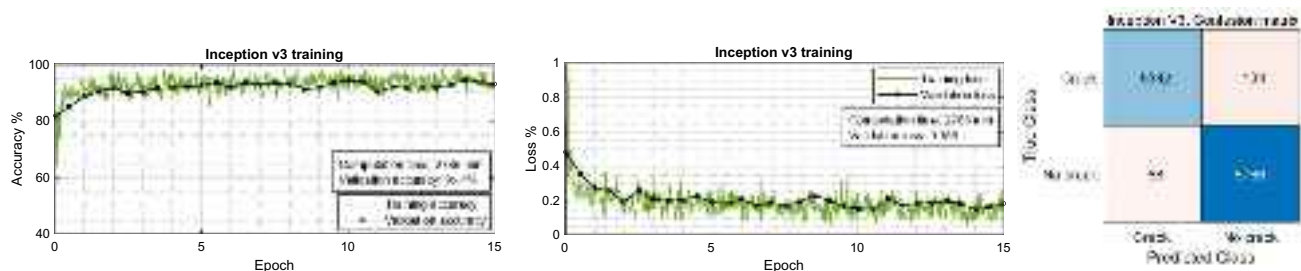


Fig. 17 Accuracy and loss curves for the training and validation of the Inception V3 classifier model and the confusion matrix obtained after applying the trained model to the test dataset

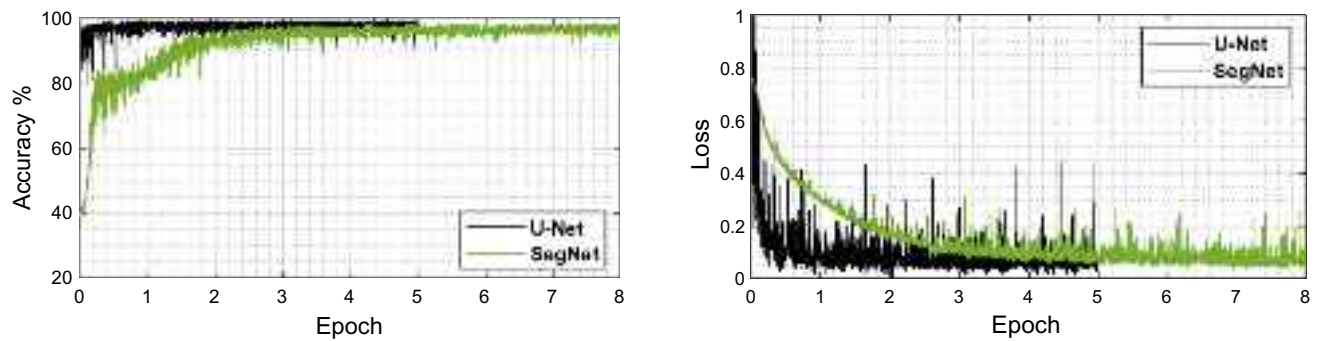


Fig. 18 Accuracy and loss training curves for semantic segmentation using U-Net and SegNet

Fig. 19 The performance of the two trained CNNs in crack segmentation; pixels classified as parts of cracks are shown in red

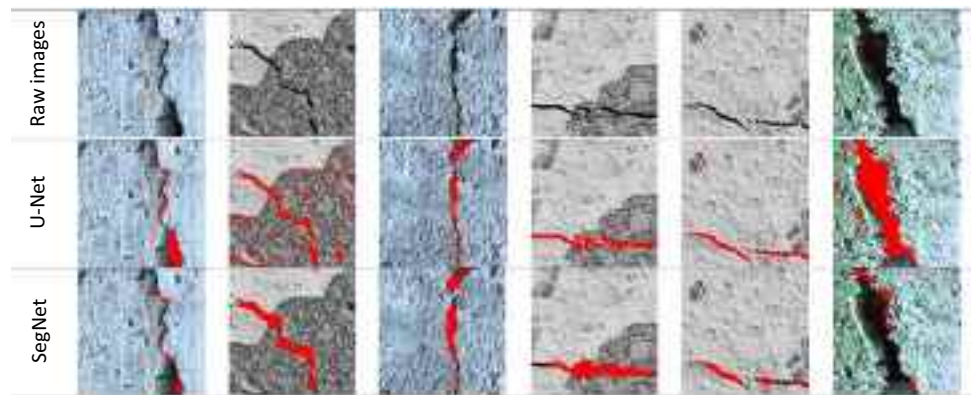


Table 6 Accuracies of the U-Net and SegNet segmentation models compared to the ground truth

Architecture	Training epochs	Global Accuracy	Mean Accuracy	Mean IoU	Weighted IoU	Mean BF Score
U-Net	5	0.97347	0.79777	0.68244	0.95677	0.82509
SegNet	8	0.88893	0.79681	0.51817	0.86612	0.64191

5.3 Workflow evaluation and crack quantification

The developed workflow was tested by applying it to a new high-resolution image (6240×4160 pixels) of the Kedke-jokk tunnel, captured in an area that would be difficult for a human inspector to access. Figure 20 shows the cracks detected by the CNNs, with damaged areas enclosed in red bounding boxes and segmented cracks shown using red pixels. Both false positives and false negatives were obtained. False positives (indicated by filled red boxes) occurred in areas containing crack-like patterns, while false negatives (indicated by dashed yellow boxes) are areas that contain clearly visible cracks whose shapes and scales differ from those of the cracks included in the dataset used to train the Inception V3 classifier model. To avoid false positives, a deeper CNN could be trained to extract deeper features from the training dataset. The incidence of false negatives could be reduced by further augmentation of the training dataset

with randomly rescaled images to enable the detection of cracks with different scales and shapes. Figure 21 shows the precision of crack segmentation achieved with U-Net which is needed for crack quantification.

After crack detection and segmentation, realistic 3D coordinates of the detected defects are extracted to measure the dimensions of the segmented cracks. For this purpose, an orthomosaic image or orthophoto must be generated, providing a photorealistic representation of the region of interest (ROI) from which crack dimensions can be measured. Orthophotos can be generated from images captured from different perspectives using collinearity equations [59]. The distance of the camera from the surface and the ground sampling distance (GSD) can then be calculated, allowing the dimensions of the cracks to be determined by counting the numbers of crack pixels in the horizontal and vertical directions. Figure 22 illustrates this process as performed in a controlled environment.

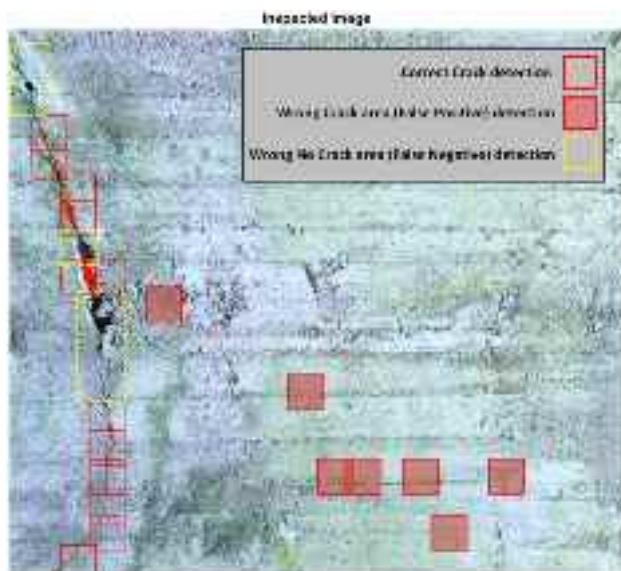


Fig. 20 Cracks detected and segmented by the studied CNNs

Using the approach described above, a 3D model of the Kedkejokk tunnel was generated by the hierarchical Dense Structure-from-Motion (DSfM) method, with an elevated resolution in the defected area. Cracks in this area were then segmented with the trained CNNs and an orthophoto was generated to measure the width of the cracks. Figure 23 shows the results obtained at three points along a detected crack whose coordinates were recorded on the digital model to serve as documented data from the autonomous inspection. A good accuracy was achieved, although it should be

noted that the accuracy of the results increases with the GSD of the captured images.

6 Conclusions and contributions

This paper introduces a method for semi-automated bridge inspection based on the generation of a photogrammetric 3D model followed by CNN-based crack detection, localization, and quantification. Case studies on two existed concrete structure were performed to evaluate and refine these two processes.

The Juovajokk bridge served as the case study for the first process, i.e., photogrammetric 3D model generation. Major challenges in this process were discussed, ranging from digital image acquisition to model quality evaluation, as well as parameters that influence model quality. It was concluded that:

1. Close-range photogrammetry (CRP) offers several benefits compared to conventional monitoring methods, namely:
 - Safe remote monitoring of difficult to access areas.
 - High productivity while providing data with sufficient accuracy for reliable analysis.
 - Easy set-up requiring comparatively little operator skill.
2. During data acquisition, it is vital to determine the optimal distance for image acquisition; the acquired images should be of sufficiently high resolution to permit the

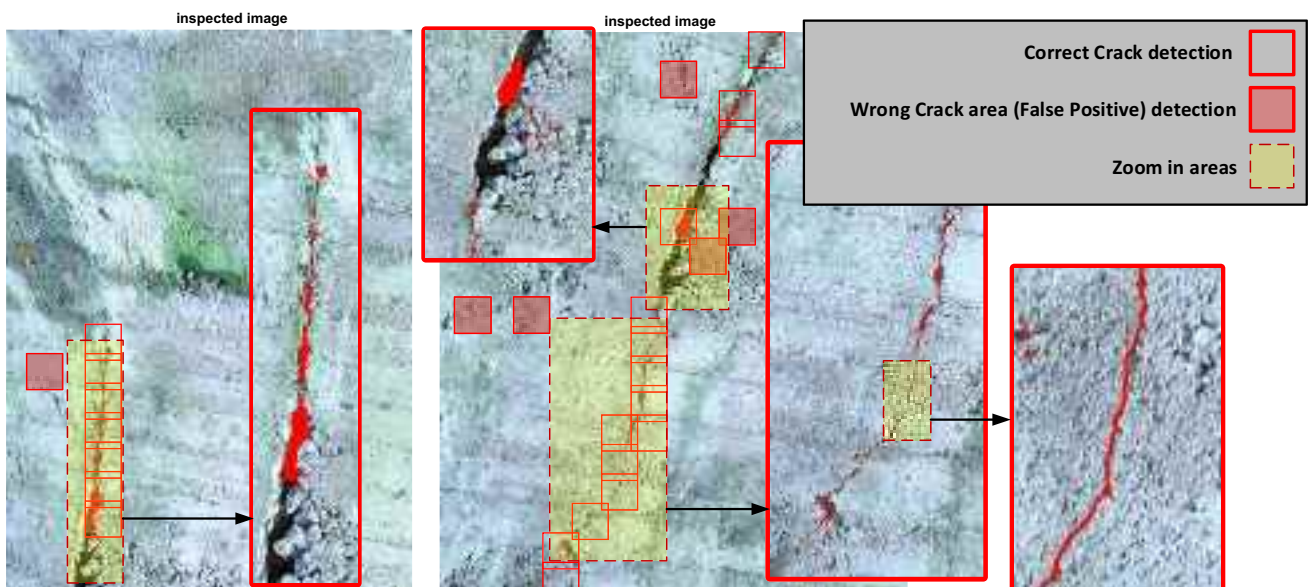


Fig. 21 Accuracy of crack segmentation using trained U-Net

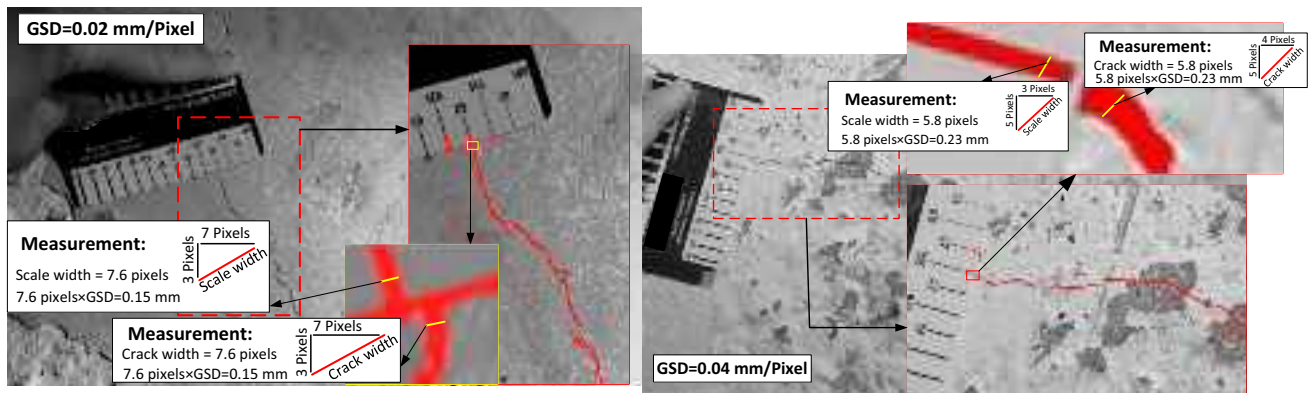
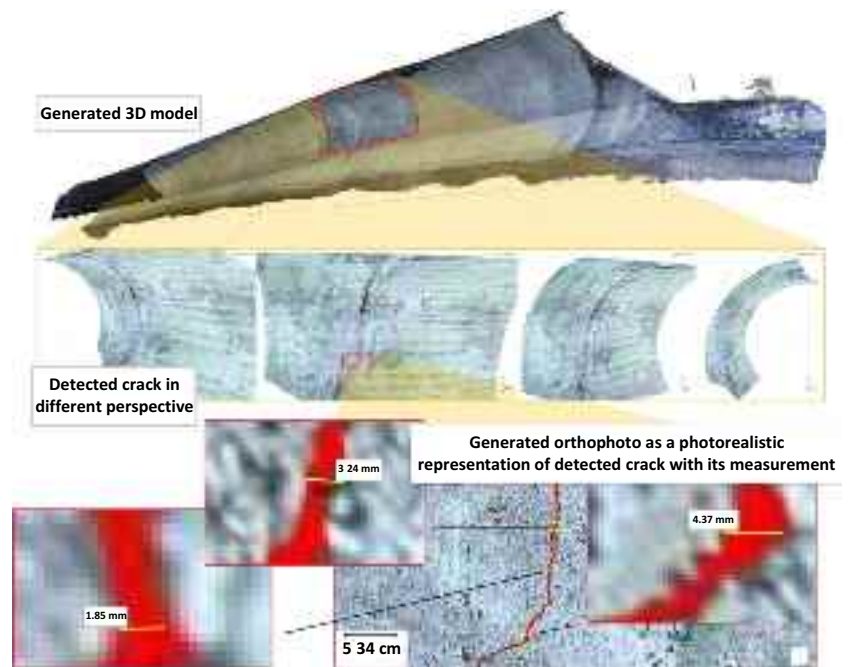


Fig. 22 Crack quantification in a controlled environment

Fig. 23 Crack quantification in a hard-to-access area of the Kedkejokk tunnel using the proposed method



detection of small-scale defects while capturing the fewest images possible. Resolution depends on the camera's calibration and interior orientation including the focal length and sensor size.

- Drawbacks of the photogrammetric approach include long post-processing times, computational cost, and greater noise when compared to laser scanning. Outlier noise removal discussed by applying an outlier removal threshold based on the point-to-point deviation between the photogrammetric point cloud and a reference point cloud. A threshold of twice the standard deviation provides acceptable accuracy while limiting the loss of point-cloud density.

The Kedkejokk tunnel served as the case study for the second process, i.e., semi-automated crack detection, localization, and segmentation using deep convolutional neural network models. In this step, the captured images were split into a set of sub-images that were classified into Crack and No Crack groups using a CNN classifier, InceptionV3 model. An end-to-end CNN with the U-Net architecture was then used to perform pixel-wise segmentation of the detected cracks within the sub-images. To increase accuracy and reduce computational cost, only the sub-images assigned to the "Crack" class by the classifier were subjected to this process. When this approach was applied to the case study, a crack in a hard-to-access area was successfully detected

and measured by pixel-wise mapping to an orthophoto. It was concluded that:

1. Generated datasets augmented and increased by random rescaling, horizontal/vertical flipping, changing the brightness/contrast/color, and random cropping of the included images. This improves the ability of the trained models to extract desired features under diverse conditions. However, it is important to minimize blurring in the captured images as this tends to cause loss of features.
2. The experimental results reveal that the size of the sub-images has an important effect on training times; smaller images contain fewer features than large ones and, thus, require far more iterations to reach convergence. However, the larger the images, the worse the precision of the boundary boxes around damaged areas. In addition, larger sub-images decreased the precision of semantic segmentation performed on those detected sub-images.
3. The proposed method has considerable potential in automated infrastructure inspection but some problems, due to background noise, remain to be overcome. The existence of noisy patterns such as shadows, dirt, and snow or water spots on surfaces makes crack detection very challenging, especially for the fine cracks.

Overall, while the semi-automated inspection technique proposed herein performs well, it clearly still requires supervision by a human inspector. A feedback system incorporating corrections supplied by expert inspectors could enable continuous improvement in the trained algorithm, allowing the proposed method to become an increasingly effective assistant for bridge inspectors that facilitates inspection of hard-to-access areas while also making inspection safer and more productive. Further research into the applications of reinforcement learning in autonomous inspection is, thus, warranted.

Acknowledgements The research has been carried out with funding from FORMAS, project number 2019-01515 and has been partly financed within the European Horizon 2020 Joint Technology Initiative Shift2Rail through contract no. 101012456 (IN2TRACK3). Any opinions, findings and conclusions expressed in this paper are those of the authors and do not necessarily reflect the views of funders.

Funding Open access funding provided by Lulea University of Technology.

Data availability The datasets generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes

were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Dabous SA, Yaghi S, Alkass S, Moselhi O (2017) Concrete bridge deck condition assessment using IR Thermography and Ground Penetrating Radar technologies. *Autom Constr* 81:340–354
2. Taljsten B, Elfgrén L (2008) Sustainable bridges—a European integrated research project—background overview and results. In: International conference on advanced composite materials in bridges and structures
3. Li H-N, Yi T-H, Ren L, Li D-S, Huo L-S (2014) Reviews on innovations and applications in structural health monitoring for infrastructures. *Struct Monitor Maint* 1(1):1–45
4. Popescu C, Taljsten B, Blanksvard T, Elfgrén L (2019) 3D reconstruction of existing concrete bridges using optical methods. *Struct Infrastruct Eng* 15(7):912–924
5. Lin Z, Pan H, Wang X, Li M (2019) Improved element-level bridge inspection criteria for better bridge management and preservation. Mountain-Plains Consortium, Fargo
6. Phares BM, Washer GA, Rolander DD, Graybeal BA, Moore M (2004) Routine highway bridge inspection condition documentation accuracy and reliability. *J Bridge Eng* 9(4):403–413
7. Graybeal BA, Phares BM, Rolander DD, Moore M, Washer G (2002) Visual inspection of highway bridges. *J Nondestruct Eval* 21(3):67–83
8. Morgenthal G, Hallermann N, Kersten J, Taraben J, Debus P, Helmrich M, Rodehorst V (2019) Framework for automated UAS-based structural condition assessment of bridges. *Autom Constr* 97:77–95
9. Xu Y, Turkan Y (2019) Br IM and UAS for bridge inspections and management. *Eng Constr Archit Manag*. <https://doi.org/10.1108/ECAM-12-2018-0556>
10. Mirzazade A, Popescu C, Blanksvard T, Taljsten B (2021) Workflow for off-site bridge inspection using automatic damage detection—case study of the Pahtajokk bridge. *Remote Sens* 13(14):2665
11. Mirzazade A, Nodeh MP, Popescu C, Blanksvard T, Taljsten B (2021) Utilization of computer vision technique for automated crack detection based on UAV-taken images. In: International conference of the European association on quality control of bridges and structures
12. Riveiro B, Jauregui D, Arias P, Armesto J, Jiang R (2012) An innovative method for remote measurement of minimum vertical underclearance in routine bridge inspection. *Autom Constr* 25:34–40
13. Sacks R, Kedar A, Borrmann A, Ma L, Brilakis I, Huthwohl P, Daum S, Kattel U, Yosef R, Liebich T (2018) See Bridge as next generation bridge inspection: overview, information delivery manual and model view definition. *Autom Constr* 90:134–145
14. Cha G, Park S, Oh T (2019) A terrestrial LiDAR-based detection of shape deformation for maintenance of bridge structures. *J Constr Eng Manag* 145(12):04019075
15. Isailovic D, Stojanovic V, Trapp M, Richter R, Hajdin R, Dollner J (2020) Bridge damage: detection, IFC-based semantic enrichment and visualization. *Autom Constr* 112:103088

16. Borin P, Cavazzini F (2019) Condition assessment of RC bridges Integrating machine learning, photogrammetry and BIM. *Int Arch Photogramm Remote Sens Spatial Inf Sci*. <https://doi.org/10.5194/isprs-archives-XLII-2-W15-201-2019>
17. Mirzazade A, Popescu C, Blanksvard T, Taljsten B (2021) Application of close range photogrammetry in structural health monitoring by processing generated point cloud datasets. In: IABSE congress 2021: structural engineering for future societal needs, Ghent, September 22–24
18. Azimi M, Eslamlou AD, Pekcan G (2020) Data-driven structural health monitoring and damage detection through deep learning: State-of-the-art review. *Sensors* 20(10):2778
19. Sony S, Dunphy K, Sadhu A, Capretz M (2021) A systematic review of convolutional neural network-based structural condition assessment techniques. *Eng Struct* 226:111347
20. Jang K, Kim N, An Y-K (2019) Deep learning-based autonomous concrete crack evaluation through hybrid image scanning. *Struct Health Monit* 18(5–6):1722–1737
21. Dung CV, Anh LD (2019) Autonomous concrete crack detection using deep fully convolutional neural network. *Autom Constr* 99:52–58
22. Li S, Zhao X (2018) Convolutional neural networks-based crack detection for real concrete surface. In: *Sensors and smart structures technologies for civil, mechanical, and aerospace systems 2018*
23. Silva WR, Lucena DS (2018) Concrete cracks detection based on deep learning image classification. In: *Multidisciplinary digital publishing institute proceedings*, vol 2, p 489
24. Cha Y-J, Choi W, Buyukozturk O (2017) Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civ Infrastruct Eng* 32(5):361–378
25. Yokoyama S, Matsumoto T (2017) Development of an automatic detector of cracks in concrete using machine learning. *Procedia Eng* 171:1250–1255
26. Maeda H, Sekimoto Y, Seto T (2016) Lightweight road manager: smartphone-based automatic determination of road damage status by deep neural network. In: *Proceedings of the 5th ACM SIGSPATIAL international workshop on mobile geographic information systems*, pp 37–45
27. Zhang L, Yang F, Zhang YD, Zhu YJ (2016) Road crack detection using deep convolutional neural network. In: *IEEE international conference on image processing (ICIP)*
28. Zhang A, Wang KC, Fei Y, Liu Y, Tao S, Chen C, Li JQ, Li B (2018) Deep learning-based fully automated pavement crack detection on 3D asphalt surfaces with an improved CrackNet. *J Comput Civ Eng* 32(5):04018041
29. Wang K, Zhang A, Li JQ, Fei Y, Chen C, Li B (2017) Deep learning for asphalt pavement cracking recognition using convolutional neural network. In: *Airfield highway pavements 2017: Airfield pavement technology and safety*, pp 166–177
30. Zhang A, Wang KC, Li B, Yang E, Dai X, Peng Y, Fei Y, Liu Y, Li JQ, Chen C (2017) Automated pixel-level pavement crack detection on 3D asphalt surfaces using a deep-learning network. *Computer-Aided Civ Infrastruct Eng* 32(10):805–819
31. Fan Z, Wu Y, Lu J, Li W (2018) Automatic pavement crack detection based on structured prediction with the convolutional neural network. *arXiv preprint arXiv:1802.02208*
32. Atha DJ, Jahanshahi MR (2018) Evaluation of deep learning approaches based on convolutional neural networks for corrosion detection. *Struct Health Monit* 17(5):1110–1128
33. Wu W, Qurishee MA, Owino J, Fomunung I, Onyango M, Atolagbe B (2018) Coupling deep learning and UAV for infrastructure condition assessment automation. In: *2018 IEEE international smart cities conference (ISC2)*, IEEE, pp 1–7
34. Cha Y-J, Choi W, Suh G, Mahmoudkhani S, Buyukozturk O (2018) Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. *Computer-Aided Civ Infrastruct Eng* 33(9):731–747
35. Wang N, Zhao Q, Li S, Zhao X, Zhao P (2018) Damage classification for masonry historic structures using convolutional neural networks based on still images. *Computer-Aided Civ Infrastruct Eng* 33(12):1073–1089
36. Rytter A (1993) *Vibrational based inspection of civil engineering structures*. Dept. of Building Technology and Structural Engineering, Aalborg University
37. Remondino F, Spera MG, Nocerino E, Menna F, Nex F (2014) State of the art in high density image matching. *Photogram Rec* 29(146):144–166
38. Broome L (2016) Comparison between terrestrial close range photogrammetry and terrestrial laser scanning
39. Sargent I, Harding J, Freeman M (2007) "Data quality in 3D: gauging quality measures from users' requirements", *international archives of photogrammetry, remote sensing and spatial. Inf Sci* 36(2/C43):8
40. Koutsoudis A, Vidmar B, Ioannakis G, Arnaoutoglou F, Pavlidis G, Chamzas C (2014) Multi-image 3D reconstruction data evaluation. *J Cult Herit* 15(1):73–79
41. Cheng S-W, Lau M-K (2017) Denoising a point cloud for surface reconstruction. *arXiv preprint arXiv:1704.04038*
42. Chen S, Laefer DF, Mangina E, Zolanvari SI, Byrne J (2019) UAV bridge inspection through evaluated 3D reconstructions. *J Bridge Eng* 24(4):05019001
43. Valenca J, Puente I, Julio E, Gonzalez-Jorge H, Arias-Sanchez P (2017) Assessment of cracks on concrete bridges using image processing supported by laser scanning survey. *Constr Build Mater* 146:668–678
44. Jahanshahi MR, Masri SF (2012) Adaptive vision-based crack detection using 3D scene reconstruction for condition assessment of structures. *Autom Constr* 22:567–576
45. Lu Q, Lee S (2017) Image-based technologies for constructing as-is building information models for existing buildings. *J Comput Civ Eng* 31(4):04017005
46. Kwak E, Datchev I, Habib A, El-Badry M, Hughes C (2013) Precise photogrammetric reconstruction using model-based image fitting for 3D beam deformation monitoring. *J Surv Eng* 139(3):143–155
47. Khaloo A, Lattanzi D (2017) Hierarchical dense structure-from-motion reconstructions for infrastructure condition assessment. *J Comput Civ Eng* 31(1):04016047
48. Khaloo A, Lattanzi D, Cunningham K, Dell'Andrea R, Riley M (2018) Unmanned aerial vehicle inspection of the Placer River Trail Bridge through image-based 3D modelling. *Struct Infrastruct Eng* 14(1):124–136
49. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*
50. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*
51. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*
52. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*
53. Kim B, Cho S (2018) Automated vision-based detection of cracks on concrete surfaces using a deep learning technique. *Sensors* 18(10):3452
54. Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K (2016) SqueezeNet: AlexNet-level accuracy with 50x fewer

- parameters and < 0.5 MB model size. arXiv preprint [arXiv:1602.07360](https://arxiv.org/abs/1602.07360)
55. Besl PJ, McKay ND (1992) Method for registration of 3-D shapes. In: Sensor fusion IV: control paradigms and data structures
 56. Ozgenel CF, Sorguc AG (2018) Performance comparison of pre-trained convolutional neural networks on crack detection in buildings. In: Isarc. proceedings of the international symposium on automation and robotics in construction
 57. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention
 58. Badrinarayanan V, Kendall A, Cipolla R (2017) Segnet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans Pattern Anal Mach Intell* 39(12):2481–2495
 59. Schenk T (2005) Introduction to photogrammetry. The Ohio State University, Columbus, vol 106

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.