WARSAW UNIVERSITY OF TECHNOLOGY Index 351733
FACULTY OF CIVIL ENGINEERING
COMMITTEE FOR CIVIL AND WATER ENGINEERING
POLISH ACADEMY OF SCIENCES ISSN 1230-2945

DOI: 10.24425/ace.2020.135225

ARCHIVES OF CIVIL ENGINEERING

Vol. LXVI ISSUE 4 2020

ASESMENT OF STATE-OF-THE-ART METHODS FOR BRIDGE INSPECTION: CASE STUDY

B. WÓJCIK [1], M. ŻARSKI [2]

Despite the progress in digitization of civil engineering, the process of bridge inspection is still outdated. In most cases, its documentation consists of notes, sketches and photos. This results in significant data loss during structure maintenance and can even lead to critical failures. As a solution to this problem, many researchers see the use of modern technologies that are gaining popularity in civil engineering. Namely Building Information Modelling (BIM), 3D reconstruction and Artificial Intelligence (AI). However, despite their work, no particular solution was implemented. In this article, we evaluated the applicability of state-of-the-art methods based on a case study. We have considered each step starting from data acquisition and ending on BIM model enrichment. Additionally, the comparison of deep learning crack semantic segmentation algorithm with human inspector was performed. Authors believe that this kind of work is crucial for further advancements in the field of bridge maintenance.

Keywords: building information modelling, 3D reconstruction, photogrammetry, artificial intelligence, bridge inspection

[1] MSc, Eng. Gliwice, Silesian University of Technology, Faculty of Civil Engineering, Akademicka 5, 44-100 Gliwice, Poland, e-mail: bartosz.wojcik@polsl.pl
MSc, Eng. Seoul, Chung-Ang University, School of Architecture and Building Science, Heukseok-ro 84, Dongjak-gu, 06974 Seoul, Republic of Korea, e-mail: bwojcik@cau.ac.kr
[2] MSc, Eng. Gliwice, Silesian University of Technology, Faculty of Civil Engineering, Akademicka 5, 44-100 Gliwice, Poland, e-mail: mateusz.zarski@polsl.pl
MSc, Eng. Seoul, Chung-Ang University, School of Architecture and Building Science, Heukseok-ro 84, Dongjak-gu, 06974 Seoul, Republic of Korea, e-mail: matzar92@cau.ac.kr

1 INTRODUCTION

On 14 August 2018 Morandi Bridge collapsed, 43 people died. Investigation showed that the main reason for bridge failure was corrosion of the steel cables contained in concrete stays. Defect was already diagnosed and repairs were scheduled, but they did not come in time [1]. This tragic event clearly shows the importance of bridge maintenance and early defect detection.

This need for better bridge management systems is in stark contrast to the fact that to this day inspection documentation consists of sketches, drawings and digital photographs. This means that inspection as a whole is liable to significant loss of information. This has been demonstrated by Phares et al., as their study shows that bridge inspectors can omit serious structural defects in their notes. In case of photographic documentation further lacks were found, as more than 60% of inspectors did not take photos of the cracks on bottom surface of bridge superstructure and as much as 96% omitted spalling of the wings [2]. Additionally, this is not only crucial for structure safety, as according to Kong and Frangopoulos preventing minor defects from developing into major ones can reduce costs up to 65% [3]. This underlines the importance of maintenance tools in the aspect of lean construction and sustainability in infrastructure engineering.

On the other hand, in recent years Building Information Modelling became more popular in the bridge industry. Literature review performed by Costin et al. shows that work on implementing this technology applies to all aspects of transport infrastructure, including maintenance [4]. The taxonomy and comparison of different dimensionality Bridge Management System models was presented by Bień [5], in his work he foresaw wider application of 3D models for this task. Additionally, one of the reasons to develop BIM comes from significant data loss with traditional paper based methods during transitions between project phases, design – construction – maintenance, which in return yield additional costs [6].

Costin et al. pointed out the 3D reconstruction as one of the most important technologies used in infrastructure construction [4]. The aspect of using 3D reconstruction in general civil engineering applications has been broadly described in the literature review performed by Wang and Kim. It shows that as much as 48% of researchers use 3D reconstruction techniques to reconstruct the geometric or semantic model of a structure. It is also worth to stress out that semantic enrichment of BIM models has been identified as one of the directions of future work [7]. Additionally, there are various examples of defect detection and quantification that uses 3D reconstruction for concrete and steel elements [8]–[12].

Another important factor in context of visual inspection would be Artificial Intelligence, as the use of AI algorithms can also be widely recognized in the field of civil engineering. It was used before for both design [13] and infrastructure maintenance [14], in the form of Artificial Neural Networks and with spatially aware Convolutional Neural Networks for more recent maintenance applications. Good example would be defect detection with laser scanning [15] and image recognition [16]. Additionally, there are some examples of automatic crack detection. Li et al. proposed modified active contour model and greedy search-based support vector machine for recognition and evaluation of bridge cracks [17]. Cha, Choi and Büyüköztürk approach include the use of Convolutional Neural Networks for image recognition and classification [18]. That is not the only one example of CNNs use, Mei, Gül and Azim as well as Dung and Anh or Yang et al. utilised Fully Convolutional Networks in order to quickly semantically segment large-scale images [19]–[21].

The technologies mentioned above were integrated in a project called Semantic Enrichment Engine for Bridges, SeeBridge for short, that focused on bridge inspections [22]. The result of which have been described in several articles. Sacks et al. described Information Delivery Manual and Model View Definition for bridge inspection process with complacence to IFC4 Add 2 schema [23]. As an extension of this work, Hüthwohl et al. proposed a way of integrating information about defects of reinforced concrete bridges into BIM models [24]. Which lead to development of AI algorithms for detecting healthy concrete surfaces [25] and classification of unhealthy ones with 83.5% accuracy [26]. However, SeeBridge is not an isolated case, other researchers have also worked on the subject. For example Shim et al. proposed integration of Digital Twining into maintenance process [27] and Isailović et al. described multiview-classification method for detection of spalling defects, with more than 70% accuracy [28].

In this paper we are going to evaluate the applicability of these state-of-the-art methods proposed for bridge inspection.

2 EVALUATED SOLUTION

The ability to capture object geometry and appearance using 3D reconstruction in combination with BIM technology and AI presents the possibility for redefining bridge inspection. This could address the previous stated problem of data loss, that affects current approach.

Nowadays, bridge inspections are mainly performed on site by inspectors themselves. However, the inspections can be virtual, i.e. performed on computer screen rather than in the field. Additionally, inspectors could be backed by Machine Learning algorithms in defect detection, or even completely replaced.

In scenario like that, inspectors would not have to be present on site. Data can be acquired and processed by specialized employees or companies, then provided for inspection in the form of enriched BIM model. After that these models would be inspected in a virtual environment. The results of this virtually preformed inspection would be stored back in the BIM model and available for further processing (Fig. 1).

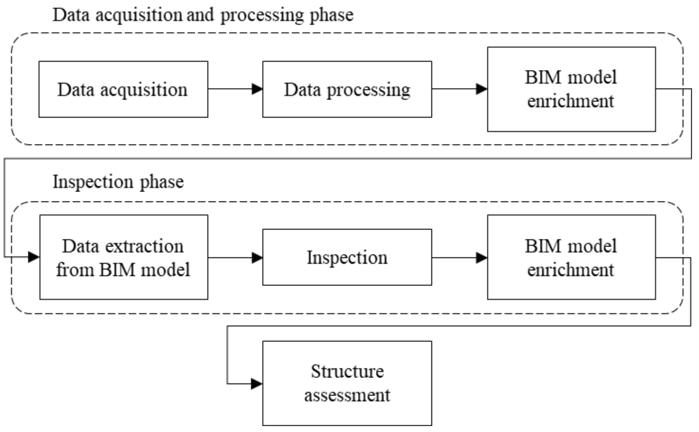

Fig. 1 Evaluated solution diagram

Inspection process like that would not only improve data exchange, but also relieve inspectors from tedious and dangerous parts of their work, while increasing their productivity by allowing them to focus only on ensuring structure safety. Moreover, storing data in open format like IFC would also be beneficial in terms of data preserving, sharing and management. IFC models can be kept on central as well as local servers, and can contain results from multiple inspections, thus allowing for recording defect propagation. Also, if needed these models could be easily shared to consult with experts.

3 METHODOLGY

Although textured models were already used by Hüthwohl and Brilakis for inspection purposes [25], these models were created directly from meshes captured by 3D reconstruction. This approach is highly impractical, because it would mean that for every inspection new model should be created. Instead it would be better to enrich existing as-built BIM models to as-is state, since they contain more information than visible structure geometry.

As for automatic defect detection, it should be stressed out that there are considerable drawbacks that prevent wide practical application of AI that have to be researched. These drawbacks include problems like lack of trained models and datasets published along with research papers – problems addressed already in [29] or significant demand of computing power for both training and inferencing with the use of AI algorithms [30].

To assess the applicability of these methods, an inspection of sample bridge element was performed in the form of a case study, with two major purposes. First one is illustrative - by presenting an extensive description of each step we want to provide information to anyone interested in deploying this kind of solution. Second one is evaluative - by assessing the possibility of each of the step's usage in practical application. For the purpose of assessing various approaches we also check current state-of-the art methods from the most current fields of active research, like Machine Learning.

Fig. 2 Inspected bridge pier

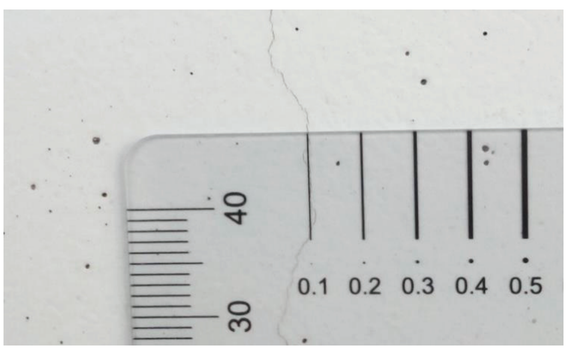

Fig. 3 Width measurement on crack featured on test case pillar

For the test case one of the supports of a multi-span prestressed box girder bridge was selected. The pier itself consists of foundation slab and two columns with pedestals for bearings (Fig. 2). The main reason behind case selection was the fact that the surfaces of these columns were extensively cracked. The presence of so many cracks would require an assessment of the technical condition of this column, which in turn would involve an inventory of the defects. This would not be possible in normal inspection.

Additionally, the cracks present on the surface of this test case (as depicted in Fig. 3) were mostly of width below 0.2 mm. While observable with naked eye, they would prove to be tedious in manual detection. Furthermore, such thin defects would serve as additional challenges for both photogrammetry and AI methods employed in the article.

We hope that performing this case study will reveal problems that could be researched in future and provide a base of knowledge in the latest inspection methods for current researchers to further develop them. We also deeply believe that our work will inspire more focus on the evolution of practically used inspection methods in order to facilitate the work of the infrastructure inspectors.

4 CASE STUDY

4.1 ASSEMBLED BIM MODEL

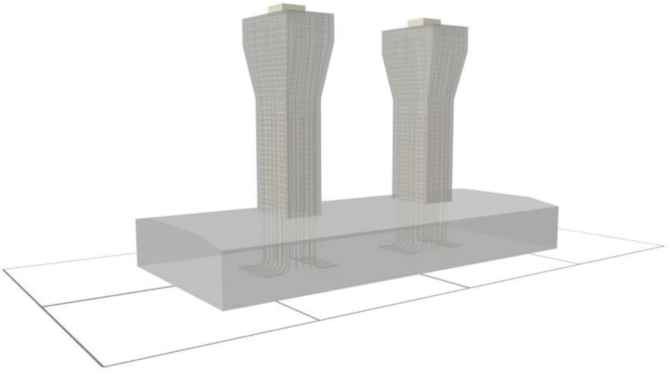

Fig. 4 Created BIM model of the support with visible column reinforcement

The as-built BIM model of the selected pier was assembled according to IFC schema (Fig. 4). The whole substructure was modelled as *IfcElementAssembly*. Each element like slab (*IfcSlab*) or column (*IfcColumn*) was aggregated into this assembly using *IfcRelAgregates* and connected with each other with help of *IfcRelConnectsElements* relation. Additionally, each concrete element aggregates *IfcElementAssembly*, which contains all of the reinforcement bars in the form of *IfcReinforcingBar* objects. This ensured that correct spatial dependencies were in place.

4.2 DATA ACQUISITION

Photos of column surfaces were taken with 12.3 MPix DSLR (Sony Alpha 500) from approximately 20-30 cm in order to capture thin defects on the pillar. They were taken from ground level and the camera axis was kept perpendicular to captured surfaces. The overlap of the horizontal rows and vertical columns was 80% and 60% respectively. Camera was equipped with Ø55 18-55 mm f/3.5-5.6 lens, with parameters fixed throughout data acquisition process:

- focal length: 35 mm,
- shutter speed: 1/40,
- f-stop: f/18,
- light sensitivity: ISO 200.

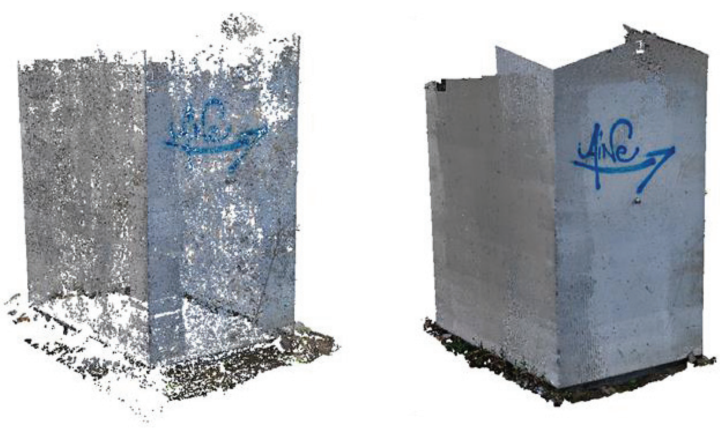

Fig. 5 Created sparse (left) and dense point cloud (right)

Whole data set contained 969 images, that covered over 11 square meters of column surface. It was captured by one person in 1 hour and 40 minutes. On this dataset sparse and dense reconstruction was performed, resulting in nearly 585 thousand points in form of sparse and over 61 million points in form of dense cloud (Fig. 5).

4.3 DATA PROCESSING AND EMBEDDING INTO IFC FILE

Although raw dense point clouds could be used for inspection purposes, Krijnen and Beetz showed that the current IFC schema needs to be extended to store point clouds with additional data like point colour [31]. However, alternatively to raw point clouds, information about appearance can be embedded into the BIM model in the form of textured triangular meshes [24]. Texturing in computer graphics is a technique that uses images (textures) or mathematical functions (procedural textures) to represent the appearance of a 3D object surface. This approach is well known in computer graphics and the photogrammetry pipeline was already used by game developers to reliably and fast create immersive game assets [32]. In this case texture would be created for simplified mesh reconstructed in surface reconstruction step. Nonetheless, having geometry from the BIM model it is possible to texture column representation itself. Although, to use a photogrammetric model to texture as-built BIM geometry we have to align the point cloud to the BIM element coordination system.
The process of alignment is presented in Fig. 6 and was done with help of open source Point Cloud Library [33] that is written in C++.

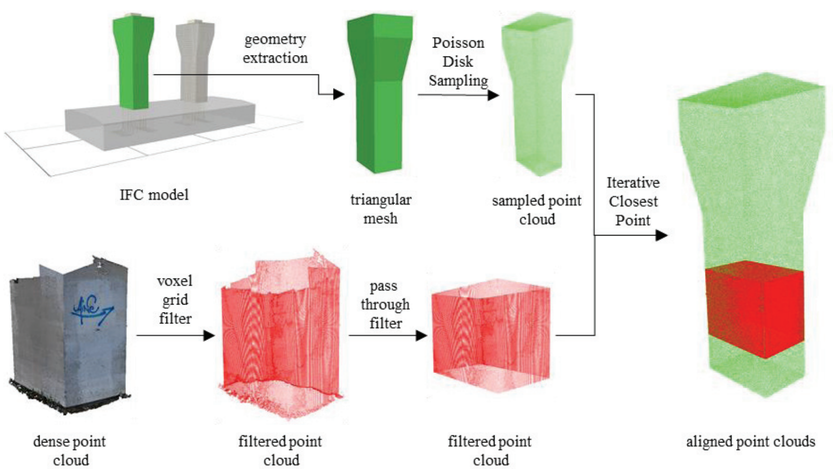

Fig. 6 Point cloud alignment

The alignment itself is done with well-known Iterative Closest Point algorithm. ICP can minimize difference between two point clouds and is often used for point cloud registration. Nevertheless, it is sensitive to initial position of input cloud relative to target cloud, the best solution is to manually define initial transformation and refine it with ICP. In this case, the final transformation can be calculated according to Eq. (4.1).

$$\begin{bmatrix}\text{final}\\\text{transformation}\\\text{matrix}\end{bmatrix}=\begin{bmatrix}\text{ICP}\\\text{transformation}\\\text{matrix}\end{bmatrix}\begin{bmatrix}\text{initial}\\\text{transformation}\\\text{matrix}\end{bmatrix}\tag{4.1}$$

ICP estimated transformation with fitness score of 4.81×10^{-5}, which translates to Root-Mean-Square Error of 6.93 mm. Also, it has to be stressed out, that without a survey benchmark the present solution found by ICP would be ambiguous.

Additionally, before running ICP some pre-processing had to be done. This process can be broken down into two separate tasks that produce inputs for transformation matrix estimation.

First of all, element of interest geometry has to be extracted from the IFC file and converted into point cloud. This could be done with Poisson Disk Sampling that can produce random evenly distributed points on mesh surfaces. These points are spaced no closer than required distance and at the same time are packed as tight as possible. The resulting point cloud reassembles real data better.

Secondly, due to a fact that created from acquired data dense point cloud consists of too many points to be processed efficiently and it has to be downsampled. We have decided to use two filters for this task. First voxel grid filter reduced point count and then pass through filter removed unwanted regions.

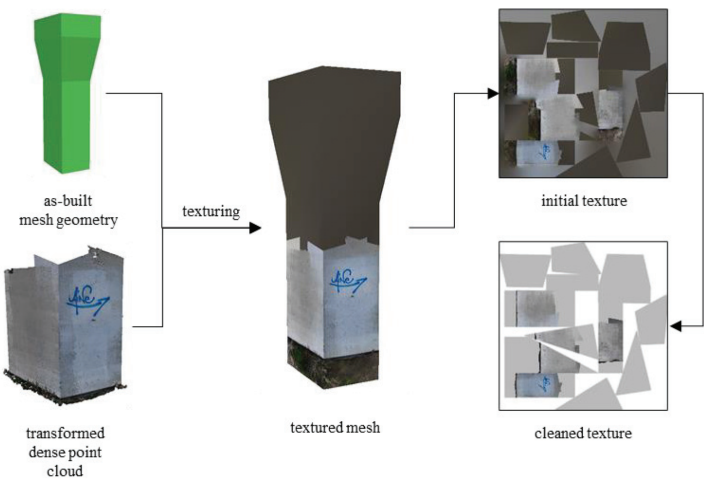

Fig. 7 Texturing column mesh extracted from BIM model

As mentioned before, knowing transformation of point cloud and having geometry from BIM model it is possible to texture column representation itself (Fig. 7). This enabled us to create texture with resolution of 10 pixels per mm. Texture was then manually cleaned but this was done for visualization reasons only and has no impact on further processing. Texture acquired this way can be embedded into the BIM model. In IFC schema textures can be modelled using one of *IfcSurfaceTexture* subclasses:

- *IfcImageTexture,*
- *IfcPixelTexture,*
- *IfcBlopTexture*.

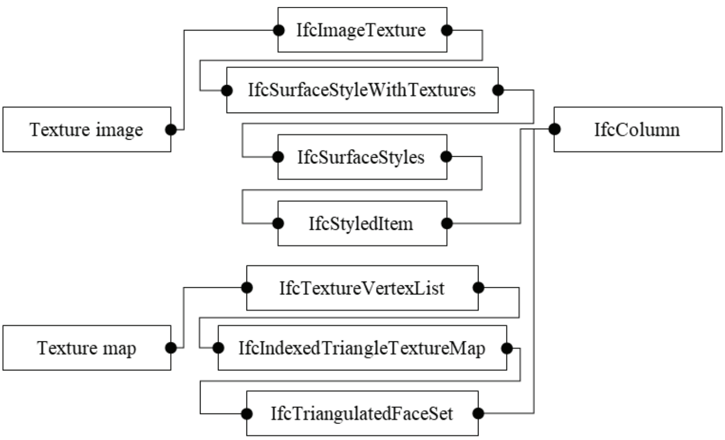

Fig. 8 Texture image and UV map mapped to IFC classes

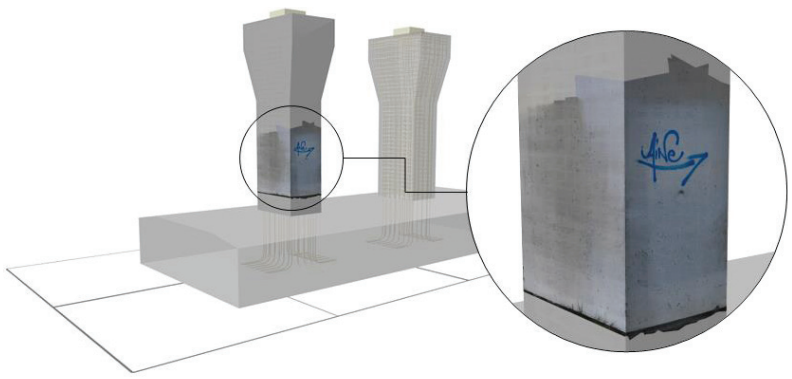

Fig. 9 Texture data visualization

For this particular case, *IfcImageTexture* was chosen as using an external image has no impact on IFC file size. Also, it can be easily replaced by a new one once it is linked. Additionally, to apply 2D image texture onto 3D objects some transformations have to be defined. These transformations take the form of texture maps that define the way textures wrap around objects. Process of creating maps is called texture mapping or UV mapping, and takes its name form a coordination system that is commonly used for textures. The way of embedding, both texture and texture map, into IFC file is shown in Fig. 8. Whereas, visualization of textured model is depicted in Fig. 9.

4.4 MANUAL SURFACE INSPECTION

Having information about surface appearance already embedded into the BIM model, a custom application for visual inspection was developed in C# programming language. Application would load the IFC model, then present the texture of selected faces for the user as an image. On this image defects are detected and defined manually (Fig. 10). This was achieved with help of xBIM toolkit [34] and Emgu CV [35] libraries. The xBIM provides tools for creating, reading, editing and viewing BIM models defined according to the IFC schema. Whereas Emgu CV is C# wrapper for popular computer vision library OpenCV, and was used for image presentation and manual defect geometry definition.

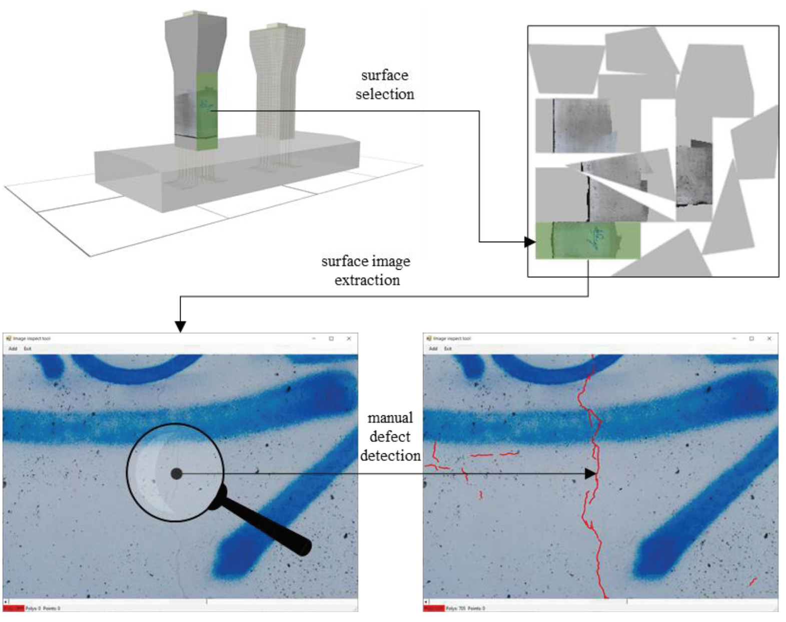

Fig. 10 Manual defects detection and annotation

4.5 COMPARISON WITH AI SURFACE INSPECTION

To assess the possibility to replace human inspectors with AI based algorithms for defect detection, the results of manual inspection were compared with state-of-the-art AI algorithm, DeepCrack [29]. This deep learning algorithm was developed for semantic segmentation of concrete surfaces with visible crack defects that obtained an Accuracy metric of above 97% noted in the research paper.

For the purpose of comparison, texture of heavily cracked concrete surfaces earlier inspected manually were fed to the DeepCrack framework for automatic AI annotation. At this point however, the first major limitation of AI approach was noted – it was unable to process such large images without exceeding 25 GB of VRAM memory – an amount that could be found more commonly in specialized workstations for video processing and rendering. For this reason, before annotation, each texture had to be divided to up to 600 of image crops and then stitched back together after defect recognition. It was also found out that the computations were extremely resource heavy and the framework required a powerful GPU unit by default. Those however are scarce in governmental infrastructure management departments. Furthermore, in order for the algorithm to be able to detect any cracks on the featured surfaces, the threshold value of certainty of defect detection had to be set to as low as 6% (15/255), while with only two classes, the default certainty threshold would be 50%. The results of DeepCrack annotation, as compared to manual crack detection, are presented in the Table 1 below in the number of annotated pixels.

Table 1 Confusion matrix for DeepCrack framework

	Predicted Background	**Predicted Crack**
Actual Background	3357223533 (TN)	60662463 (FP)
Actual Crack	23843766 (FN)	1750338 (TP)
Sum of pixels	3443480100 (TOT)	

Considering the confusion matrix, DeepCrack metrics were calculated using Eq. (4.2) – (4.5) and listed in the Table 2.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TOT}} \tag{4.2}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{4.3}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{4.4}$$

$$\text{Balanced Accuracy} = \frac{\text{Precision} + \text{Recall}}{2} \tag{4.5}$$

Table 2 Metrics of DeepCrack framework measured on featured test case

Accuracy	Precision	Recall	Balanced accuracy
0.975	0.028	0.068	0.048

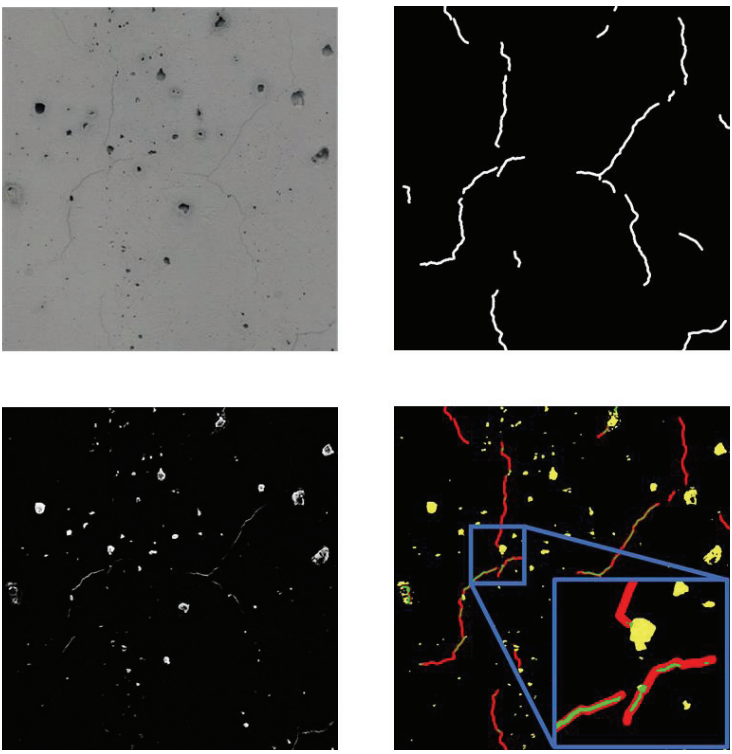

Fig. 11 Original image (top left), cracks detected in manual inspection (top right), DeepCrack cracks indications (bottom left) and comparison between human and AI (bottom right)

While DeepCrack accuracy is seemingly excellent, because of high class imbalance (much higher negative class count), this metric alone can be misleading. It does not consider class count or impact of practical classification results in given task. It should be stressed out that false negative classification can be especially dangerous in the task of infrastructure maintenance, where it has a major impact on infrastructure management cost or safety. Moreover, as seen in Precision and Recall metrics, DeepCrack framework does not handle both false negative and false positive classification well yielding Balanced Accuracy of just below 5%. For better illustration of the algorithm's performance, comparison of AI to manual annotation is depicted in the Fig. 11. In the comparison part of the image red colour stands for false negative, yellow for false positive and green for true positive DeepCrack predictions.

In the Fig. 11 it is clearly visible that DeepCrack framework classifies nearly all of the surface irregularities as cracks, without consideration of their shape. It is also apparent that it does not recognize most of the thin cracks featured in the texture and the algorithm's output consists mostly of false positive classification like concrete pores.

In conclusion, despite CNNs ability to generalize knowledge across vast datasets, these methods still have to overcome substantial drawbacks that render them mostly useless in practical applications, where the exact conditions of the dataset used for its training are not met.

4.6 DEFECT MODELLING

Hüthwohl et al. already described defect modelling according to IFC schema [24]. Each defect is modelled as *IfcSurfaceFature* objects with corresponding to its type *IfcPropertySet*. And defects of the same type are then aggregated into *IfcElementAssembly* that is a part of any subclass of *IfcBuildingElement*. The same way as previously described reinforcement.

However, they proposed defect representation in simplified form of textured mesh created from defect bounding box. It has to be noted that this kind of representation would be misleading in real life scenarios as defects could overlap and therefore one representation could contain multiple defects at once. Alternatively the texture of this mesh could be masked to include only pixels classified as defect or even more complex geometry could be created to represent only the area of the defect itself, but that would solve the problem only partially.

Yet another solution would be to use explicit geometry instead of textured mesh. For example, cracks as linear defect could be modelled using *IfcPolyline* class. This would not only solve the problem with overlapping defect representations but also provide detailed information about defect shape. Moreover, this precise description could be used for further structure assessment. This approach also highlights the advantages of manual inspection, as output is in the form of polylines. Whereas there is no AI algorithm capable of extraction crack polyline representation yet.

4.7 RESULTS

Presented case study shows that the evaluated solution is partially viable. We were able to acquire data with consumer grade DSLR that enabled us to create and embed surface texture with a density of 10 pixels per mm. That was more than enough to perform visual inspection of said surface. However, state-of-the-art AI performed poorly in crack segmentation and cracks had to be detected manually. Apart from that we were able to embed all detected cracks into the BIM model (Fig. 12).
The most encouraging fact is how accurate and fast the whole process was. Data, that was collected in less than two hours by inexperienced personnel, produced high quality results that can be inspected in mere minutes.

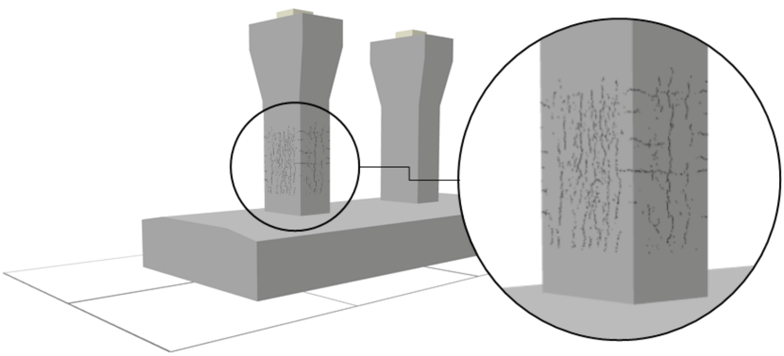

Fig. 12 Result of visual inspection, model with all visible cracks detected on scanned surface

5 DISCUSSION

It has to be noted that manual data acquisition is labour intensive and sometimes close to impossible. This could change with usage of UAVs as it should become faster and more reliable. In addition, there is a lot of potential in automation of this process and some may even consider the possibility of making it fully autonomous. Phung et al. described an algorithm for planning and optimisation of UAV path for surface inspection [36]. González-Sieira et al. presented a motion planning algorithm for UAV autonomous navigation and proved that when employing it, quality dense point clouds could be acquired [37].
Still, the ultimate goal of fully automatic infrastructure bridge inspection is ahead. As seen in the results of comparison between current state-of-the-art AI approach and human abilities, it was found out that precise, automatic crack detection is not yet obtainable. It may require training detectors on

datasets with finer defects and using AI algorithms utilizing more complex microarchitectures like in the work of Augustauskas [38] or using methods employing other imaging spectra in conjunction with AI [39]. Nonetheless it should be considered with the actual hardware possibilities of governmental units performing inspections. Also, the sole task of dataset building, often considered as tedious and non-scientific is crucial to developing accurate AI algorithms that would be useful in practical applications. Although it had already been done in different fields (eg. self-driving cars), in Civil Engineering it lags behind other industries. Lastly, datasets currently available seem to be built with the aim of obtaining high accuracies in research papers only.

Further steps of automatic inspection would include measuring of the defects dimensions like its length and width. Work on methods of obtaining such features is already ongoing, an example of which is the paper of Wang et al. who presented a new means of pavement crack width measurement based on Laplace's Equation [40]. It also makes an attempt on extracting defects geometry, basing on skeletonization and point clustering algorithms. This kind of framework paired with working detection algorithms could provide all information needed to embed this type of defects into the BIM model. Having this in mind, it should be pointed out that further automation of the inspection process would only be possible after perfection of detection step mentioned earlier. It is also worth noting that while crack detection is extensively researched, other defects are underrepresented in the research papers and in order to develop complex, automated systems, holistic approach to defects should be employed.

6 CONCLUSION AND FUTURE WORK

In this paper we evaluated the applicability of the state-of-the-art methods for bridge inspections, because despite research that are already done, they are not yet implemented in real life scenarios. In order to have meaningful insight into that matter, a broad case study was performed and analysed according to the described solution. The following conclusions, regarding each used method, can be drawn from this study.

Currently used IFC schema is capable of describing complex bridge geometry with spatial and non-spatial dependencies, model defect information and geometry. Additionally, to some extent it is possible to enrich this kind of models with data captured by 3D reconstruction, in the form of textured meshes. This is the only way of including appearance information. However, it has to be noted that no currently available IFC viewer supports textured models. For purposes of this study custom WPF viewer application had to be developed on top of xBIM viewer. Moreover, this was not the only custom application developed for the purpose of IFC file interaction. Such lack of out-of-the-box software underlines the need for IFC-based tools in the process.

The photogrammetric pipeline used was more than suitable for the task at hand. We have not encountered any problems regarding sparse/dense reconstruction or texturing. The model was reconstructed properly despite the fact, that the plain concrete surface at first glance seems to be completely featureless. Moreover, achieved texture resolution enabled us to perform visual inspection and detect cracks less than 0.2 mm thick. That would not be possible during normal inspection with currently used tools. However, labour intensity of data acquisition has to be noted, making it impractical without the help of UAVs.

Lastly, application of Artificial Intelligence should be considered as the biggest disappointment of this case study. State-of-the-art deep learning model for crack detection and image semantic segmentation DeepCrack, was not able to achieve even acceptable results, let alone perform on a human-like level.

In the end we were able to perform intended inspection, however noting current limitations, further work can be planned. In order to reap all of the benefits from existing technology and incorporate it to the infrastructure inspections, there is still work to be done. First of all, while still developing, BIM technology has to make up with vision-based 3D reconstruction techniques. It is especially apparent in current IFC viewer software that requires additional tinkering to be able to display textured models or even in IFC schema to store and view point clouds. At the same time, unlike commercial products, the IFC file standard is managed by a non-profit foundation, so its update may be delayed

As for the use of photogrammetry in infrastructure inspections, we believe that this technique deserves a broader introduction to the field. It showed great potential in the test case and confirmed our conviction that it would allow for affordable 3D point cloud acquisition, especially when paired with UAVs. It could also significantly increase the efficiency of bridge inspector's work.

The last part of test case – use of AI algorithms directs our attention to the development of reliable AI models to be used for infrastructure inspection. It should be stressed out, that despite the impressive results obtained by Machine Learning in the research papers, these methods are still in their infancy and are not yet adaptable in practical applications. It therefore remains as an active research field for the future work.

ACKNOWLEDGMENTS

Work presented in this study was supported by Polish Ministry of Investment and Economic Development as a part of a Silesian University of Technology as a Centre of Modern Education based on research and innovation project (POWR.03.05.00 00.z098/17 00) and Silesian University of Technology as part of financing young scientists research (BKM 537/Rb-7/202o).

REFERENCES

[1] F. Gatti, "Crollo Ponte Genova, tiranti 'ridotti del venti per cento': Ministero e Autostrade sapevano [Collapse of Genoa bridge, tendons 'reduced by twenty percent': the Ministry and the Highways knew]," *L'Espresso*, 2018. [Online]. Available: https://espresso.repubblica.it/inchieste/2018/08/19/news/genova-tiranti-su-ponte-morandi-ridotti-del-20-per-cento-ministero-e-autostrade-sapevano-1.326085.

[2] B. M. Phares, G. A. Washer, D. D. Rolander, B. A. Graybeal, and M. Moore, "Routine Highway Bridge Inspection Condition Documentation Accuracy and Reliability," *J. Bridg. Eng.*, vol. 9, no. 4, pp. 403–413, Jul. 2004.

[3] J. S. Kong and D. M. Frangopol, "Life-Cycle Reliability-Based Maintenance Cost Optimization of Deteriorating Structures with Emphasis on Bridges," *J. Struct. Eng.*, vol. 129, no. 6, pp. 818–828, Jun. 2003.

[4] A. Costin, A. Adibfar, H. Hu, and S. S. Chen, "Building Information Modeling (BIM) for transportation infrastructure – Literature review, applications, challenges, and recommendations," *Autom. Constr.*, vol. 94, no. June, pp. 257–281, 2018.

[5] J. Bień, "Modelling of structure geometry in Bridge Management Systems," *Arch. Civ. Mech. Eng.*, vol. 11, no. 3, pp. 519–532, Jan. 2011.

[6] C. Eastman, R. Sacks, G. Lee, and P. Teicholz, *BIM Handbook*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2018.

[7] Q. Wang and M. K. Kim, "Applications of 3D point cloud data in the construction industry: A fifteen-year review from 2004 to 2018," *Adv. Eng. Informatics*, vol. 39, no. September 2018, pp. 306–319, 2019.

[8] G. Teza, A. Galgaro, and F. Moro, "Contactless recognition of concrete surface damage from laser scanning and curvature computation," *NDT E Int.*, vol. 42, no. 4, pp. 240–249, 2009.

[9] A. Janowski, K. Nagrodzka-Godycka, J. Szulwic, and P. Ziółkowski, "Modes of Failure Analysis in Reinforced Concrete Beam Using Laser Scanning and Synchro-Photogrammetry," *Second Int. Conf. Adv. Civil, Struct. Environ. Eng. 2014*, pp. 16–20, 2014.

[10] M.-K. Kim, H. Sohn, and C.-C. Chang, "Localization and Quantification of Concrete Spalling Defects Using Terrestrial Laser Scanning," *J. Comput. Civ. Eng.*, vol. 29, no. 6, p. 04014086, Nov. 2015.

[11] B. Guldur Erkal and J. F. Hajjar, "Laser-based surface damage detection and quantification using predicted surface properties," *Autom. Constr.*, vol. 83, no. September 2016, pp. 285–302, 2017.

[12] M. K. Kim, Q. Wang, and H. Li, "Non-contact sensing based geometric quality assessment of buildings and civil structures: A review," *Autom. Constr.*, vol. 100, no. December 2018, pp. 163–179, 2019.

[13] M. Ochmański, G. Modoni, and J. Bzówka, "Prediction of the diameter of jet grouting columns with artificial neural networks," *Soils Found.*, vol. 55, no. 2, pp. 425–436, Apr. 2015.

[14] X. Wu, J. Ghaboussi, and J. H. Garrett, "Use of neural networks in detection of structural damage," *Comput. Struct.*, vol. 42, no. 4, pp. 649–659, Feb. 1992.

[15] M. Rabah, A. Elhattab, and A. Fayad, "Automatic concrete cracks detection and mapping of terrestrial laser scan data," *NRIAG J. Astron. Geophys.*, vol. 2, no. 2, pp. 250–255, Dec. 2013.

[16] S. Li, X. Zhao, and G. Zhou, "Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network," *Comput. Civ. Infrastruct. Eng.*, vol. 34, no. 7, pp. 616–634, 2019.

[17] G. Li, X. Zhao, K. Du, F. Ru, and Y. Zhang, "Recognition and evaluation of bridge cracks with modified active contour model and greedy search-based support vector machine," *Autom. Constr.*, vol. 78, pp. 51–61, Jun. 2017.

[18] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks," *Comput. Civ. Infrastruct. Eng.*, vol. 32, no. 5, pp. 361–378, 2017.

[19] Q. Mei, M. Gül, and M. R. Azim, "Densely connected deep neural network considering connectivity of pixels for automatic crack detection," *Autom. Constr.*, vol. 110, no. June 2019, p. 103018, Feb. 2020.

[20] C. V. Dung and L. D. Anh, "Autonomous concrete crack detection using deep fully convolutional neural network," *Autom. Constr.*, vol. 99, pp. 52–58, Mar. 2019.

[21] X. Yang, H. Li, Y. Yu, X. Luo, T. Huang, and X. Yang, "Automatic Pixel-Level Crack Detection and Measurement Using Fully Convolutional Network," *Comput. Civ. Infrastruct. Eng.*, vol. 33, no. 12, pp. 1090–

1109, Dec. 2018.
[22] "SeeBridge." [Online]. Available: https://seebridge.net.technion.ac.il/.
[23] R. Sacks *et al.*, "SeeBridge as next generation bridge inspection: Overview, Information Delivery Manual and Model View Definition," *Autom. Constr.*, vol. 90, no. February, pp. 134–145, 2018.
[24] P. Hüthwohl, I. Brilakis, A. Borrmann, and R. Sacks, "Integrating RC Bridge Defect Information into BIM Models," *J. Comput. Civ. Eng.*, vol. 32, no. 3, 2018.
[25] P. Hüthwohl and I. Brilakis, "Detecting healthy concrete surfaces," *Adv. Eng. Informatics*, vol. 37, no. May, pp. 150–162, 2018.
[26] P. Hüthwohl, R. Lu, and I. Brilakis, "Multi-classifier for reinforced concrete bridge defects," *Autom. Constr.*, vol. 105, no. December 2018, p. 102824, 2019.
[27] C. S. Shim, N. S. Dang, S. Lon, and C. H. Jeon, "Development of a bridge maintenance system for prestressed concrete bridges using 3D digital twin model," *Struct. Infrastruct. Eng.*, vol. 15, no. 10, pp. 1319–1332, 2019.
[28] D. Isailović, V. Stojanovic, M. Trapp, R. Richter, R. Hajdin, and J. Döllner, "Bridge damage: Detection, IFC-based semantic enrichment and visualization," *Autom. Constr.*, vol. 112, no. May 2019, p. 103088, Apr. 2020.
[29] Y. Liu, J. Yao, X. Lu, R. Xie, and L. Li, "DeepCrack: A deep hierarchical feature learning architecture for crack segmentation," *Neurocomputing*, vol. 338, pp. 139–153, 2019.
[30] Y. E. Wang, G.-Y. Wei, and D. Brooks, "Benchmarking TPU, GPU, and CPU Platforms for Deep Learning," 2019.
[31] T. Krijnen and J. Beetz, "An IFC schema extension and binary serialization format to efficiently integrate point cloud data into building models," *Adv. Eng. Informatics*, vol. 33, no. 2017, pp. 473–490, 2017.
[32] N. Statham, "Use of Photogrammetry in Video Games: A Historical Overview," *Games Cult.*, pp. 1–19, 2018.
[33] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 1–4.
[34] S. Lockley, C. Benghi, and M. Černý, "Xbim.Essentials: a library for interoperable building information applications," *J. Open Source Softw.*, vol. 2, no. 20, p. 473, 2017.
[35] "Emgu CV," 2020. [Online]. Available: http://www.emgu.com.
[36] M. D. Phung, C. H. Quach, T. H. Dinh, and Q. Ha, "Enhanced discrete particle swarm optimization path planning for UAV vision-based surface inspection," *Autom. Constr.*, vol. 81, no. April, pp. 25–33, 2017.
[37] A. González-Sieira, D. Cores, M. Mucientes, and A. Bugarín, "Autonomous navigation for UAVs managing motion and sensing uncertainty," *Rob. Auton. Syst.*, vol. 126, p. 103455, 2020.
[38] R. Augustauskas and A. Lipnickas, "Improved Pixel-Level Pavement-Defect Segmentation Using a Deep Autoencoder," *Sensors*, vol. 20, no. 9, p. 2557, Apr. 2020.
[39] J. Yang, W. Wang, G. Lin, Q. Li, Y. Sun, and Y. Sun, "Infrared Thermal Imaging-Based Crack Detection Using Deep Learning," *IEEE Access*, vol. 7, pp. 182060–182077, 2019.
[40] W. Wang, A. Zhang, K. C. P. Wang, A. F. Braham, and S. Qiu, "Pavement Crack Width Measurement Based on Laplace's Equation for Continuity and Unambiguity," *Comput. Civ. Infrastruct. Eng.*, vol. 33, no. 2, pp. 110–123, 2018.

LIST OF FIGURES AND TABLES:

OCENA NAJNOWOCZEŚNIEJSZYCH METOD STOSOWANYCH DO INSPEKCJI MOSTÓW: STUDIUM PRZYPADKU

Słowa kluczowe: modelowanie informacji o budowli, rekonstrukcja 3D, fotogrametria, sztuczna inteligencja, inspekcja obiektów mostowych

Pomimo postępu w cyfryzacji budownictwa, proces inspekcji mostów jest nadal przestarzały. W większości przypadków jego dokumentacja składa się z notatek, szkiców i zdjęć. Powoduje to znaczną utratę danych podczas fazy utrzymania konstrukcji, a nawet może prowadzić do awarii. Wielu badaczy jako rozwiązanie tego problemu upatruje w wykorzystaniu nowoczesnych technologii, które zyskują na popularności w inżynierii lądowej. Technologii takich jak modelowanie informacji o budynku (BIM), rekonstrukcja 3D i sztuczna inteligencja (AI). Jednak pomimo wykonanej do tej pory pracy nie zaimplementowano żadnego konkretnego rozwiązania.
W tym artykule oceniliśmy przydatność tych najnowocześniejszych metod na podstawie studium przypadku. Rozważaliśmy każdy krok począwszy od pozyskania danych, a skończywszy na wzbogaceniu modelu BIM. Ponadto przeprowadzono porównanie algorytmu segmentacji semantycznej pęknięć w uczeniu głębokim z ludzkim inspektorem. Uważamy, że tego rodzaju prace są kluczowe dla dalszych postępów w utrzymaniu mostów.

Received: 05.05.2020, Revised: 29.09.2020