



## Distance Insensitive Concrete Crack Detection with Controlled Blurriness Using a Convolutional Neural Network

Su Fen, N.<sup>1</sup>, Shokravi, H.<sup>2</sup>, Bakhary, N.<sup>3\*</sup>, Padil, Kh.H.<sup>2</sup> and Zainal Abidin, A.R.<sup>2</sup>

<sup>1</sup> M.Sc., School of Civil Engineering, University Teknologi Malaysia, Johor, Malaysia.

<sup>2</sup> Assistant Professor, School of Civil Engineering, University Teknologi Malaysia, Johor, Malaysia.

<sup>3</sup> Associate Professor, School of Civil Engineering, University Teknologi Malaysia, Johor, Malaysia.

© University of Tehran 2022

Received: 21 Nov. 2021;

Revised: 16 Feb 2022;

Accepted: 27 Feb. 2022

**ABSTRACT:** Crack detection is one of the critical tasks in health monitoring and inspection of civil engineering structures. The existence of major cracks may have detrimental effects on the integrity and performance of structures that need full consideration. Recent research into crack identification has shown an increasing interest in vision-based automated techniques, employing deep-learning computational methods such as Convolutional Neural Networks (CNNs). However, the wide range of real-world situations (e.g. camera or subject motion, misfocus, mist, and fog) can significantly compromise the accuracy of CNN-based crack identification due to a mismatched dataset in training and testing. Therefore, this study aims to establish an intelligent identification model using deep CNNs to automatically detect concrete cracks from real-world images. Moreover, the efficiency of the algorithm in identifying cracks based on blurred images in the training and validation dataset was investigated. The original dataset is replicated into various blurriness levels and split into eight different crack image sub-datasets. CNN models were trained and crack identification was carried out using different levels of image blurriness. The classification performance of the trained CNN was assessed using the concrete crack image dataset taken around Universiti Teknologi Malaysia. Sensitivity studies were also conducted to investigate the efficiency of the CNN method to identify damage under various image parameters. The results showed that the subset with the combination of sharp and slight blurriness level (blurriness Level 1) reached the highest training accuracy of 98.20%, and the network trained with blurriness Level 1 alone had the best accuracy, precision, and F1 score performance over eight training subsets. Moreover, the robustness of the networks was examined and verified under four different situations, which are; lighting, crack width, colour structures, and camera shooting angle conditions. It was observed that the presence of blurred images in the training dataset can enhance the CNN crack detection performance while high shooting angle and uneven illumination has a negative effect on the accuracy of the proposed CNN.

**Keywords:** Blurriness, Concrete, Convolutional Neural Network, Crack, Distance, Structural Health Monitoring.

\* Corresponding author E-mail: [hisham7348@gmail.com](mailto:hisham7348@gmail.com)

## 1. Introduction

Concrete material has been widely used in civil structures. Over the service life, concrete structures may undergo many types of damage such as cracks, reduction of reinforcement bar diameter due to corrosion, loss of bond in the steel-concrete interface, corrosion of prestressed cables, and spalling. Lighter damage may have an impact on the aesthetic value of the structures, while severe damages could affect the durability and stability of the whole structure. Structural Health Monitoring (SHM) is the process of identifying damages during the service life of a structure (Abdulkareem et al., 2018; Abudallah Habib et al., 2021). Hence, the SHM practice, starting with early detection and followed by appropriate repair strategies, has a very important role in ensuring structural safety. Cracking is one of the most commonly reported defects that may threaten the structural health and integrity of concrete structures (Delatte, 2009).

Vision-based SHM has the potential to provide valuable information on structural monitoring (Ye et al., 2016). Conventionally, visual inspection is carried out by certified inspectors or structural engineers to detect and evaluate structural damage. However, the accuracy of human-based visual inspection is greatly influenced by the skill level and experience of the inspectors (Li et al., 2019). Additionally, this approach is not economical and more time-consuming, especially for large-scale structures. Recent advances in the quality of digital cameras and lenses, as well as their ability to be synchronised with other smart devices and vehicles, have made vision-based methods an attractive choice for SHM applications. Recently the joint application of computer-vision and deep-learning techniques are being widely incorporated in inspection, monitoring, and assessment of infrastructure.

Due to the high-capability in recognising patterns with extreme variability using

convolutional and pooling layers, the Convolutional Neural Network (CNN) has revolutionised the paradigms of crack detection using computer vision and image processing (Wang et al., 2019). CNN is a deep neural network model utilising layers with convolving filters that are inspired by the visual cortex of animals (Ciresan et al., 2011). The utilisation of CNN for crack detection has been found in various civil engineering applications such as road pavements, masonry structures, buildings, roadway, bridges, and steel and concrete structures. Cha et al. (2018) developed a quasi-real-time structural damage detection method based on a faster region-based convolutional neural network. In their study, multiple types of structural damage images of concrete cracks, steel corrosion, bolts corrosion, and steel delamination with  $500 \times 375 \times 3$  pixels resolution under uncontrolled conditions were used to train and validate the networks. The outcome of the study shows that 87.8% of structural damages were correctly detected and localised. Meanwhile, Atha and Jahanshahi (2018) demonstrated the applicability of CNN in detecting corrosion. To obtain the optimal CNN architecture, the authors have compared the performance of the existing pre-trained networks such as ZFNet and VGG16. The authors also investigated the effect of different input image parameters in terms of sizes and colour spaces. The study revealed that the input image size of  $128 \times 128$  pixels resolution and the RGB or YCbCr is the most robust image input parameter. Later, Dorafshan et al. (2018) compared the performance of common edge detectors and CNN for crack detection in concrete structures. The study employed AlexNet CNN architecture in three different modes which are: i) fully train the network from scratch, ii) alter the last fully connected layers to match with the target labels, and iii) fine-tune the layers in AlexNet. It was found that fine-tuned AlexNet provides excellent crack detection accuracy. Recently, Zhang et al. (2020) successfully conducted an investigation on

autonomous bolt loosening detection based on faster R-CNN. Images with tight and loose bolts in  $640 \times 478 \times 3$  pixels resolution were used in the training and validating process. The study successfully demonstrated that the trained networks are able to detect the bolt damage accurately with input images from different conditions in terms of lighting, angles, and vibration. These studies described above demonstrate that the use of images together with CNN is a promising tool for damage detection and the versatility makes CNN potentially outperform most of the existing sensor-based damage detection methods.

In recent years, more detailed studies have been conducted to improve the performance of CNN for crack detection. For example, Cha et al. (2017) proposed a new CNN architecture for crack detection of concrete structures without calculating defect features. The study considered the image degradation effect due to the changes in lighting and shadow conditions. The results showed that considering the image degradation in the training data provides higher robustness and adaptability of the CNN-based method in finding concrete cracks in more realistic situations. Li et al. (2018) applied CNN for detecting cracks in real bridge inspections considering disturbance of noise and clutters. The result of the experimental work showed that the proposed algorithm outperforms the up-to-date methods in crack detection of civil structures. The higher success rate achieved in the experimental verification indicates the advantages of the proposed CNN-based method in crack detection of concrete surfaces.

Since the performance of CNN is highly dependent on the incorporated image quality, a more detailed study has been conducted to improve the performance of CNNs dealing with low-quality images. For example, Zhou and Song (2021) built and trained a novel CNN architecture that can analyse images with various types of disturbances such as low contrast and shallow cracks in roadway inspections,

which led to better classification performance. Deng et al. (2020) carried out a comparative study to investigate the performance of three Faster R-CNNs, region-based fully convolutional networks, and feature pyramid network-based Faster R-CNN algorithms for detection of out of plane cracks incorporating deformable modules. The results show that the addition of deformable modules improved the precision of crack detection in all three algorithms. Chen and Jahanshahi (2020) introduced a rotation-invariant fully convolutional network called active rotating filters to consider the rotation variation property of images for crack detection of concrete and pavements.

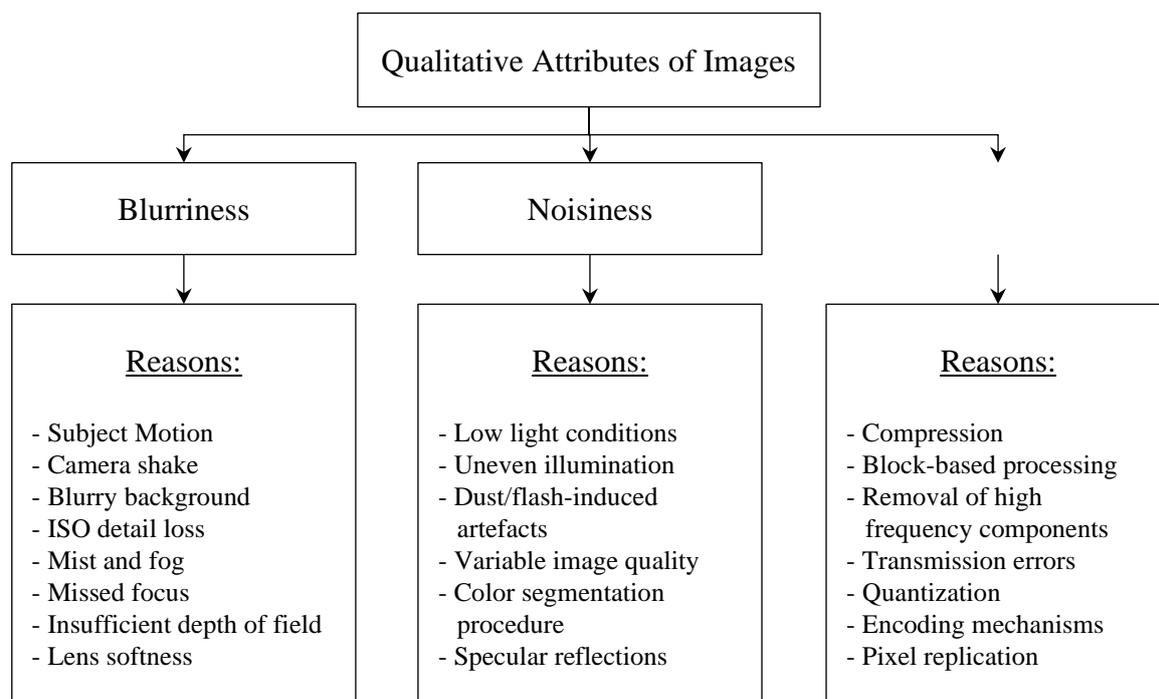
Currently, the extraction of information from blurred images is a popular research topic in the image processing field. Unmanned aerial vehicles which are capable of using a visual mounted camera are widely deployed to replace traditional human-based visual inspection of civil structures (Dorafshan and Maguire, 2018). One of the main challenges of unmanned aerial vehicles imagery is the blurriness caused by environmental and operational factors such as wind, mist, and fog (Sieberth et al., 2015). On the other hand, the application of smartphone cameras has been rapidly introduced to the visual monitoring of structures (Ratnam et al., 2019). Smartphones show great application prospects due to their unique combination of properties such as embedded hardware devices, intelligent antenna systems for transmitting and receiving data and, excellent optical properties. However, the pictures captured by mobile phones are prone to blurriness, probably due to operational factors such as camera movement and missed focus (Fankhauser et al., 2020).

Blurriness is one of the main qualitative attributes of images in computer vision systems. The qualitative attributes of images can be divided into three general categories of blurriness, noisiness, and blockiness (Shen et al., 2018). The

qualitative attributes of images and the reasons for their degradation are shown in Figure 1. Blur in images is one of the most common degradation phenomena that is mainly caused by several reasons such as camera shake, blurry background due to dust and debris, mist and fog, and missed focus. Blurriness is defined as the lack of spatial detail features resulting in a reduction in the sharpness of edges. Several solutions are introduced to deal with the blurriness of images. Dash et al. (2009) proposed restoration of images, while Han et al. (2018) and Sieberth et al. (2013) proposed removing the blurred images from the database. Though isolation of blurred images can improve the accuracy of the visual analysis and interpretation of data and reduce the rate of errors and false alarms, it may result in a partial loss of valuable information about the health state of a structure. Restoration of blurred images is another solution that is expensive due to the high computation burden and large execution time.

Several research studies have been carried out to enhance the quality of detecting cracks on concrete surfaces using

computer-vision and deep-learning techniques. Jang et al. (2019) and Fan et al. (2018) studied methods to improve crack detectability while minimising false alarms. Cha, Choi, and Büyüköztürk (2017), Kim and Cho (2018) and Protopapadakis et al. (2019) investigated the accuracy, noise immunity, and versatility of the acquired images on CNN-based crack identification. However, the number of studies to discuss the quality of image attributes and enhancing the quality of acquired images is limited for damage detection of civil engineering structures, and detailed study of the effect of images degraded with blur is also quite limited. Therefore, this study investigates the applicability of a CNN in autonomously detecting concrete cracks at various levels of images blur. Images of concrete cracks pulled from open-source databases of *Mendeley Data* and *Data in Brief* are used in this study. Selected images were compiled into a single training database for training the CNN. Afterward, the original dataset is split into eight different crack image sub-datasets with the combinations of various blurriness levels for this purpose.



**Fig. 1.** The qualitative attributes of images and the reasons for the degradation of images

CNN models are trained and crack identifications are completed using different levels of image blurriness. A sensitivity study is also conducted to investigate the efficiency of the CNN in identifying cracks based on a different level of image blur in the training and validation datasets.

## 2. Research Methodology

This section describes the CNN configuration and the dataset used in this study. The approach is the combination of data collection, pre-processing, configuring the CNN and building the model.

### 2.1. Database Preparation

Existing open-source image databases of concrete cracks available in *Mendeley Data* and *Data in Brief* are compiled into a single training database to train the CNN in this research. Figure 2 shows the sample of images from the database. To ensure consistency and robustness of the training process, certain necessary image characteristics are specified, which are: i) the image must be sharp and can be seen without soft edges, ii) the distance between the camera lens and the concrete surface must be between 0.2 m to 1.0 m, and iii) the lighting condition of the image should be between 50 lux and 25,000 lux, which represents the dark to intensive lighting condition, respectively. Each of the images collected is then cropped into sub-images with a resolution of  $227 \times 227$  pixels. The

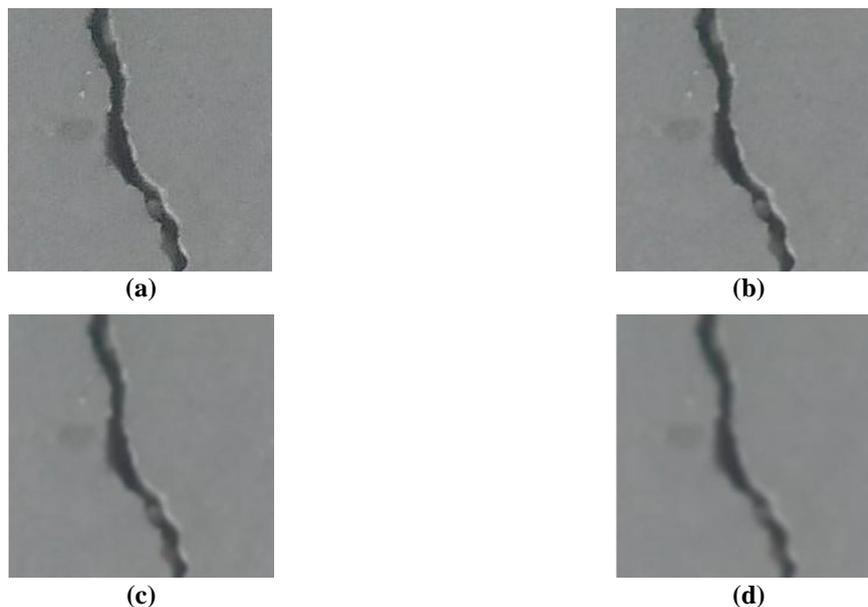
pixel of each image represents an element in the input matrix of the CNN. To prevent the occurrence of false annotations in the generation process of the database, cropped images with cracks on the edges are excluded. In total, 53,905 cropped small images were chosen which comprise 20,935 cracked and 32,970 non-cracked images. These images were stored in JPG format. As a supervised learning model, the trained CNN requires labelled information for all data. Hence, all of the  $227 \times 227$  pixel images were labelled as "crack" or "positive" and "non-crack" or "negative" for classification purposes. The images were then randomly divided into training and validation sets with a ratio of 7:3.

In order to reduce the possibility of a network overfitting, data augmentation is performed with the use of the MATLAB software. An "*imageDataAugmenter*" is created to flip the training images horizontally and vertically. Subsequently, during the training process, an "*augmentedImageDatastore*" is responsible for transforming the training batches with the specified data augmentation options in the "*imageDataAugmenter*". It augments the training images randomly for each epoch without storing any images in memory. Thus, the transformed training datasets will be slightly different at every epoch. Once the network parameters are updated, the augmented images will be discarded. It should be noted that the actual number of training images remains unchanged at each epoch.



Fig. 2. Sample of training images: a) Fine images; and b) Shadowed images

To identify the effect of the blurriness level and the performance of the CNN, various levels of blur are evaluated in this study. Each of the sub-images is blurred using a built-in function in MATLAB; 2-D Gaussian smoothing kernel with various standard deviations. The higher the value of standard deviation equals a higher level of blurriness. In this study, three levels of blurriness are examined where the standard deviation ranges from 1 to 3. Illustrations of different blurriness level on the images are depicted in Figure 3. The original dataset is replicated and split into eight different sub-datasets to train and validate the proposed CNN model in this study. These databases are composed of images with and without a blurriness level. Table 1 tabulates the image database combination used to train each network in this research. These databases are used to train different CNN models to detect damage which also consists of different levels of blurred images.



**Fig. 3.** Blurriness Level of: a) Sharp image; b) Level 1 blurred image; c) Level 2 blurred image; and d) Level 3 blurred image

## 2.2. Convolutional Neural Network

Figure 4 shows the CNN architecture, which consists of multiple convolution blocks followed by the fully connected and classified layers. The input layer reads the image of  $227 \times 227 \times 3$  pixels resolution, and transfers it to the convolution blocks. Each convolution block is comprised of a convolution layer, an activation unit, and a pooling layer. The convolution operations with filters are performed to extract image features. Additionally, batch normalisation and dropout layers are introduced with the aforementioned layers in accordance with the purposes of use. After extracting the features of images in the learning layers, the output layer is responsible for classifying the images into their respective categories. Table 2 presents the detailed parameters of each layer and operation in the proposed CNN.

**Table 1.** Image databases combination

Network	Datasets
A	Sharp
B	Blurriness level 1
C	Blurriness level 2
D	Blurriness level 3
E	Sharp + Blurriness level 1
F	Sharp + Blurriness level 2
G	Sharp + Blurriness level 3
H	Sharp + Blurriness level 1 + 2 + 3

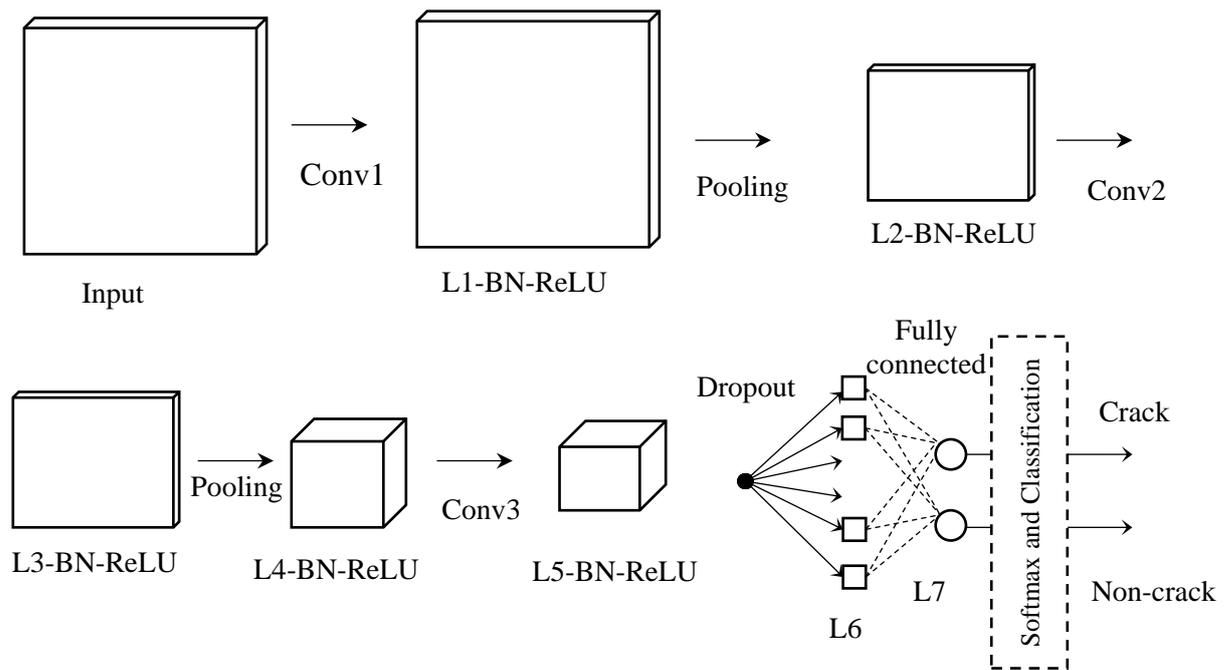


Fig. 4. Overview of CNN architecture

Table 2. Detailed parameters of each layer and operation in CNN

Layer	W/H	Channel	Operation	W/H	Channel	Number	Stride	Padding
Input	227	3	Conv1	3	3	8	1	1
L1	227	8	Pooling	3	3	-	2	0
L2	113	8	Conv2	3	8	16	1	1
L3	113	16	Pooling	3	3	-	2	0
L4	56	16	Conv3	3	16	32	1	1
L5	56	32	Dropout	-	-	-	-	-
L6	56	32	Fully Connected	1	-	2	-	-
L7	1	2	Softmax and Classification	-	-	-	-	-
L9	1	2	-	-	-	-	-	-

### 2.3. Sliding Window Technique

As depicted in Figure 5, during the sliding process, some of the cropped images may only have cracks on the edge spaces of images. These cracks can cause misclassification in the separation process. The sliding window technique is employed to overcome this problem in identifying the intact or cracked regions. The CNN architecture processes each receptive field and extracts features corresponding to weight kernels. The class of each region is recognized based on these weight dependent features. The features of the CNN architecture are automatically revised in every layer by the updated receptive fields' weights. The number of receptive fields of the designed architecture was 221, with  $227 \times 227$  pixels. In this experiment's developed network, the three convolutional

layers train the network. Features obtained from the first and second layers are only speculated as concrete surface cavities in the training data set. The visualized features are recognized and exploited by the trained network. The features from the last convolutional layer can be regarded as crack features. The segmentation results from the third layer are consistent, having low sensitivity to noise on the model's performance.

The sliding window searches over an image by shifting 227 pixels horizontally or vertically for a new region without overlapping. Since these kinds of images are disregarded in the generation of the database, the proposed CNN might not be able to detect such features as a crack or intact accurately. Therefore, based on Cha et al. (2017), an image is allowed to be

scanned twice with the designated sliding window approach, as illustrated in Figure 6. Directly, it can reduce the presence of such crack features and minimise misclassification.

#### 2.4. Performance Indicator and Network Testing

The performance of the trained CNN is assessed using pictures of concrete cracks taken from Universiti Teknologi Malaysia. These pictures are referred to hereinafter as the test set. The test set is taken in such a way that they contain one or more of: i) different lighting conditions, ii) different crack widths, iii) different colour structures, iv) different shooting angles, and v) different image distances. The results are

then compared to the actual observation in the test set. True Positives (TP) refer to the sub-image that is correctly labelled as a "crack" image by the network whilst False Positives (FP) indicate that the sub-image is falsely labelled as a "crack" image by the network. Furthermore, a True Negative (TN) test result denotes that the network correctly classified the sub-image as a "non-crack" image, and a False Negative (FN) happens when the network falsely classifies the sub-image as a "non-crack" image. Subsequently, the actual crack or Positive observation (P) is the sum of TP and FN in the testing image. In contrast, actual "non-crack" or Negative observation (N) is the sum of TN and FP in the test set.



Fig. 5. Presence of cracks on the edges of testing cropped image spaces

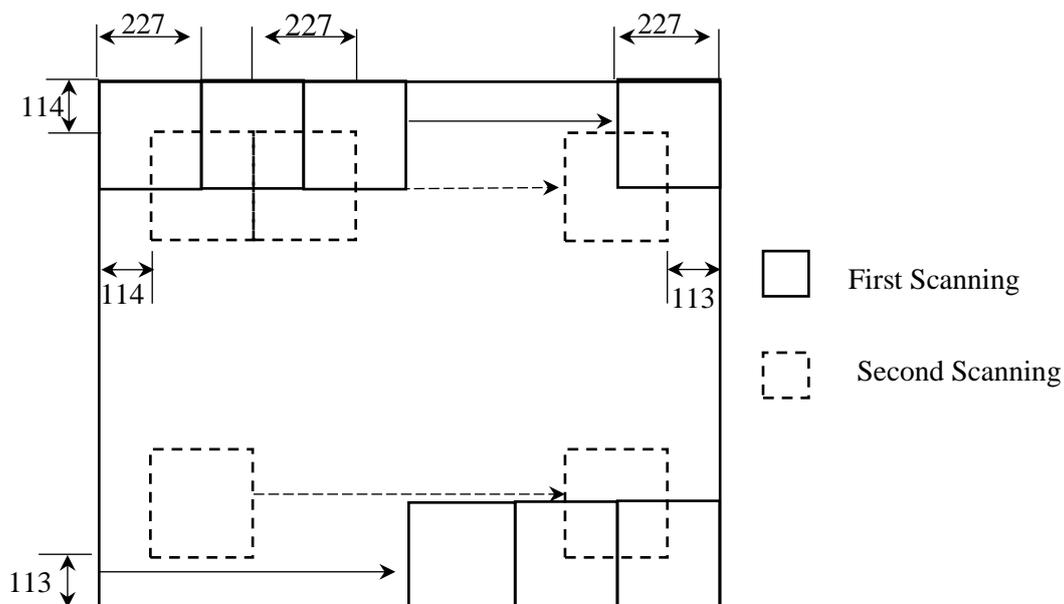


Fig. 6. Sliding window technique

In order to evaluate the performance of each network, performance indicators including accuracy, precision, recall, and F1 score will be deployed in this study. It should be noted that the accuracy does not always reflect the true performance of the CNN. The equations for the performance indicators are defined as:

$$Accuracy = \frac{\sum TP + \sum TN}{\sum P + \sum N} \quad (1)$$

$$Precision = \frac{\sum TP}{\sum TP + \sum FP} \quad (2)$$

$$Recall = \frac{\sum TP}{\sum TP + \sum FN} \quad (3)$$

$$F1 = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (4)$$

where  $TP$ ,  $TN$ ,  $P$  and  $N$ : are true positive, true negative, positive observation and negative observation as mentioned earlier.

By taking into account the precision, recall, and F1, a better understanding of the effects of FN and FP regions can be provided on the performance of the CNN.

### 3. Prediction on Training Models

Under the same CNN architecture, eight different networks were used in the training process. These image datasets are made up of images with and without a blurriness level. This is to investigate the influence of the inclusion of blurred images on the performance of CNN. The corresponding

datasets and detailed proportions of the training and validation sets for every network are listed in Table 3.

The validation accuracy is the indicator of the model response to new data and it is defined as the rate of correct classification of a newly seen image to the validation set at the iteration. The recorded training results including validation accuracy and training time for every network from the corresponding datasets are summarised in Table 4.

Stochastic gradient descent momentum was used to optimize CNN training with an initial learning rate of 0.01. The learning rate selection is a trade-off problem for accuracy and computation speed. A lower learning rate results in a longer training time, while a higher learning rate yields a shorter learning time. However, the network might reach a suboptimal result or diverge eventually. On the other hand, an epoch is a complete training cycle on the entire training dataset. The maximum number of epochs is the number of times the network has been updated for all of the images in the training set. In this study, the maximum epoch number considered is 10. The early stopping method was applied in the training process. Once there is little or no improvement on the loss of the validation set in ten consecutive times, the training progress will be stopped automatically. If the validation criterion is not met by the end of the 10th epoch, the training will be stopped at the maximum epoch number.

**Table 3.** Proportions of cracked and non-cracked images in training and validation sets

Network	No. of images					
	Training	Crack Validation	Total	Training	Non-crack Validation	Total
A	14655	6280	20935	23079	9891	32970
B	14655	6280	20935	23079	9891	32970
C	14655	6280	20935	23079	9891	32970
D	14655	6280	20935	23079	9891	32970
E	23909	12561	41870	46158	19782	65940
F	23909	12561	41870	46158	19782	65940
G	23909	12561	41870	46158	19782	65940
H	58618	25122	83740	92316	39564	131880

As shown in Table 4, the training time consumed for network A to network D is similar. As the number of images in the datasets increases, the training time increases. Network H was observed to have the longest training time since it was trained with the largest dataset. As for the validation accuracy, all networks achieved an accuracy above 95%. Network B and Network E yielded the highest accuracy as compared to the respective network made up of the same amounts of datasets. It should be noted that datasets from both networks consist of Level 1 blurriness images. On the other hand, Network H attained the lowest validation accuracy among all other networks. The lower validation accuracy can be due to the wide diversity of encompassed subclasses and a large number of training samples in Network H. Hence it can be noted that Network A performs as the optimum model based on the accuracy and training time results as shown in Table 4. Therefore, Network A is adopted for further performance investigation of the network under different conditions.

As shown in Table 4, increasing the number of training samples increased the training time. Generally, a larger training dataset could contribute more real-world scenarios, leading to a better generalization performance of the CNN algorithm. Overall, the total training time and computational overhead for performing one-time image training are not crucial if they help achieve enhanced accuracy. Hence, In real-world applications, increasing the number of training samples and training time is advisable if this increase could enhance the CNN's

classification performance and detection accuracy in practical applications. In such a way, a trade-off between the training time and the accuracy should be set.

### 3.1. Performance of Network A under Different Conditions

Network A is used to detect concrete cracks on images in the test set. 23 raw images taken at a distance of 0.5 m are used to build the test set. It is noted that the images in the test set were not used in the training of Network A. Therefore, the images in the test set are entirely arbitrary to the trained Network A. Table 5 presents the summarised results of the testing images. As can be seen from Table 5, the highest accuracy reported is 100% in testing images No. 9 and 12. The test results are shown in Figures 7a and 7b, respectively. This may be due mainly to the absence of noisy features in these testing images. In contrast, the lowest accuracy reached is 40% in testing image No. 15 as shown in Figure 7c. It can be noticed that the existence of FP regions is mainly concentrated in the area of orange colour and spall concrete surfaces. Therefore, it leads to a lower number of TNs and eventually decreases the accuracy achieved. Overall, the network accurately detected these testing images with 83% accuracy, with less than approximately 14.82% degradation of accuracy obtained during the validation process, as shown in Table 4. Additionally, four different conditions in terms of lighting condition, crack width, colour structure, and angle are assessed during the testing process. The detailed detection results are displayed and discussed in the following subsections.

**Table 4.** Results for eight networks with different datasets

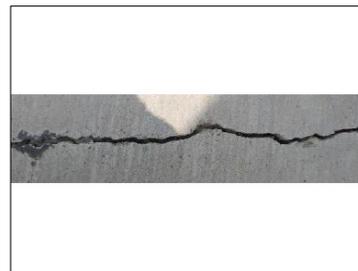
Network	Accuracy (%)	Training time (hr)
A	97.64	6.32
B	97.66	6.62
C	96.09	6.63
D	96.75	6.62
E	98.20	15.15
F	96.47	11.37
G	97.63	19.52
H	95.85	25.08

**Table 5.** Summarised result of scanned testing images

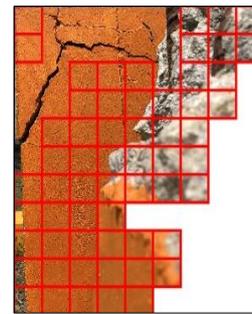
No.	P	N	TP	FP	TN	FN	Accuracy	Precision	Recall	F1	Remark
1	4	8	3	1	6	1	0.83	0.75	0.75	0.75	Figure 8a
2	33	188	33	27	161	0	0.88	0.55	1.00	0.71	-
3	32	189	31	24	165	1	0.89	0.56	0.97	0.71	-
4	24	129	14	9	120	10	0.88	0.61	0.58	0.59	-
5	29	192	28	53	139	1	0.76	0.35	0.97	0.51	-
6	23	198	10	4	194	13	0.92	0.71	0.43	0.54	-
7	26	195	3	6	189	23	0.87	0.33	0.12	0.18	-
8	21	200	21	63	137	0	0.71	0.25	1.00	0.40	Figure 9a
9	4	8	4	0	8	0	1.00	1.00	1.00	1.00	-
10	4	8	4	2	6	0	0.83	0.67	1.00	0.80	-
11	23	198	21	21	177	2	0.90	0.50	0.91	0.65	Figure 9b
12	4	8	4	0	8	0	1.00	1.00	1.00	1.00	Figure 8b
13	29	192	28	72	120	1	0.67	0.28	0.97	0.43	Figure 9a
14	28	193	28	24	169	0	0.89	0.54	1.00	0.70	Figure 8c
15	16	83	17	59	23	0	0.40	0.22	1.00	0.36	Figure 10a
16	18	177	17	14	163	1	0.92	0.55	0.94	0.69	-
17	34	187	32	71	116	2	0.67	0.31	0.94	0.47	Figure 11b
18	49	172	47	18	154	2	0.91	0.72	0.96	0.82	Figure 9c
19	16	205	11	14	191	5	0.91	0.44	0.69	0.54	Figure 8d
20	29	192	28	23	169	1	0.89	0.55	0.97	0.70	Figure 11c
21	38	183	38	9	174	0	0.96	0.81	1.00	0.90	-
22	9	9	7	3	6	2	0.72	0.70	0.78	0.74	-
23	17	82	16	27	55	1	0.72	0.37	0.94	0.53	Figure 10b
Average							0.83	0.56	0.87	0.64	



(a)



(b)



False Positive

False Negative

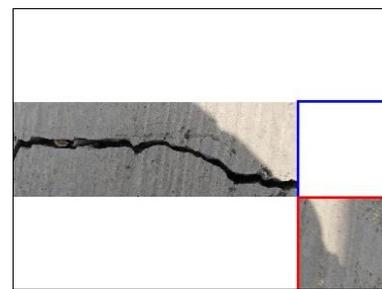
(c)

**Fig. 7.** Images scanned using the trained CNN: a) No.9; b) No.12; and c) No.15

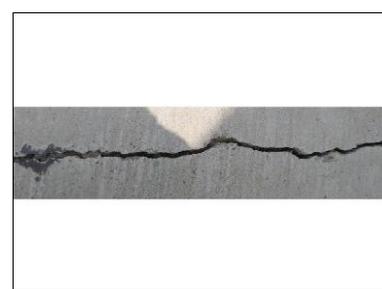
### 3.1.1. Crack Detection under Different Lighting

Figure 8 presents the detection result for the images with a shadowed area. As evidenced from Figure 8a, 1 out of 4 crack regions with the shadowed area is not successfully detected. In Figures 8b and 8c, the whole crack patterns are successfully identified by the proposed CNN. However, there are several FP regions distributed across the fine scratches and dirt stains of the image. As for Figure 8d, most of the crack regions are detected. Nevertheless, a few FN regions exist in the intensive lighting region, while FP regions are mainly distributed in the dark lighting region of the image. The test results for these images are tabulated in Table 6. It can be noted that the accuracy and recall for all testing images are sufficiently high. In contrast, precision and F1 are relatively low due to the high FP presence in Figures 8c and 8d. Figure 8d

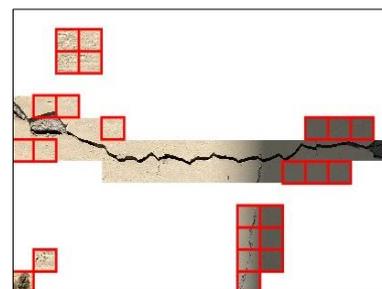
achieved the lowest precision followed by Figures 8c and 8a. This is owing to the existence of non-crack surfaces that are misclassified as crack surfaces. On the other hand, the presence of FN regions in Figures 8a and 8d reduced the recall to 0.75 and 0.65, respectively. Based on the results presented in Figure 8 and Table 6, it can be concluded that the proposed CNN is slightly sensitive to lighting conditions. Uneven illumination is an artefact classified as an image noisiness qualitative attribute of images. Figure 8d shows undetected crack regions in bright areas due to uneven illumination. The shadows or glow are created due to improper illumination mask the existing features. Several approaches are applicable for the correction of uneven illumination, including using illumination invariant features and illumination compensation approaches (Yang and Byun, 2007).



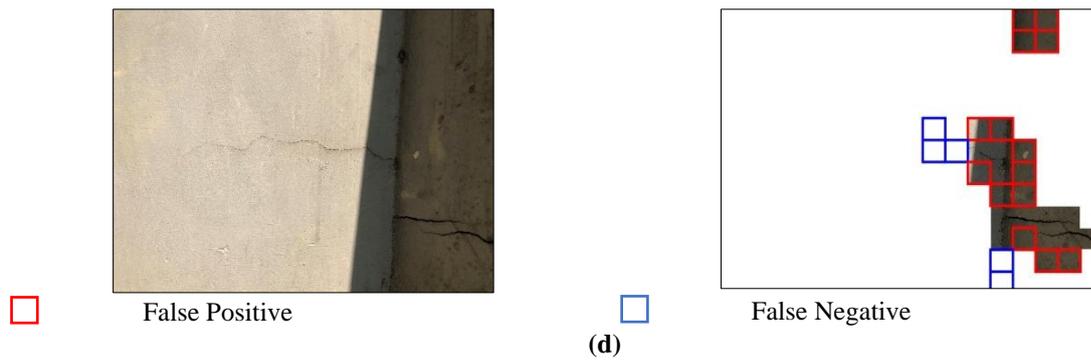
(a)



(b)



(c)



**Fig. 8.** Shadowed images scanned using the trained CNN: a)  $908 \times 681$  pixels; b)  $908 \times 681$  pixels; c)  $3859 \times 2951$  pixels; and d)  $3859 \times 2951$  pixels

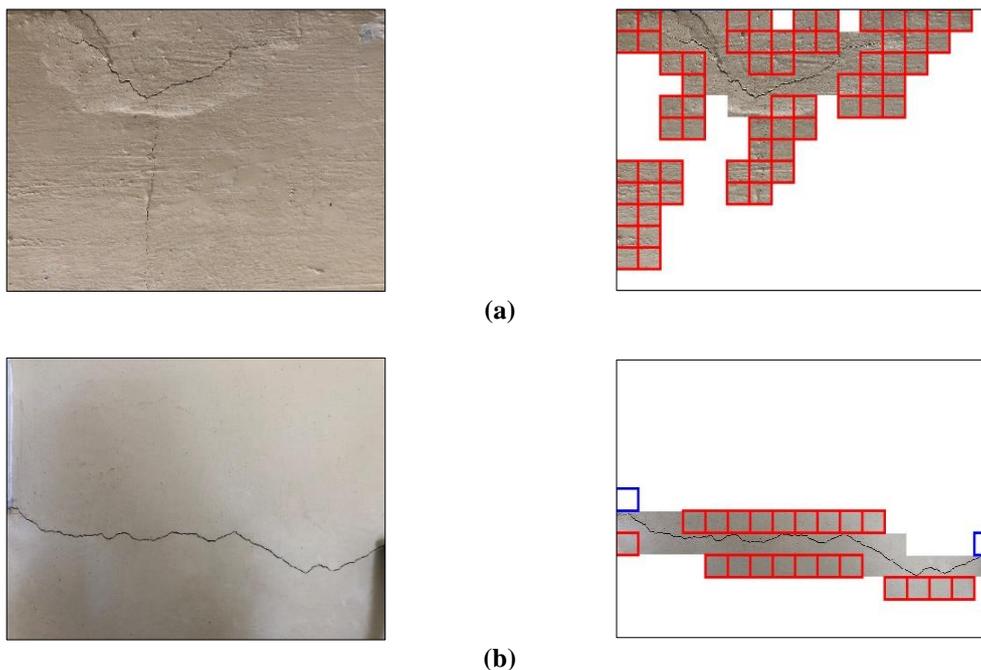
**Table 6.** Detailed analysis of test images in Figure 8

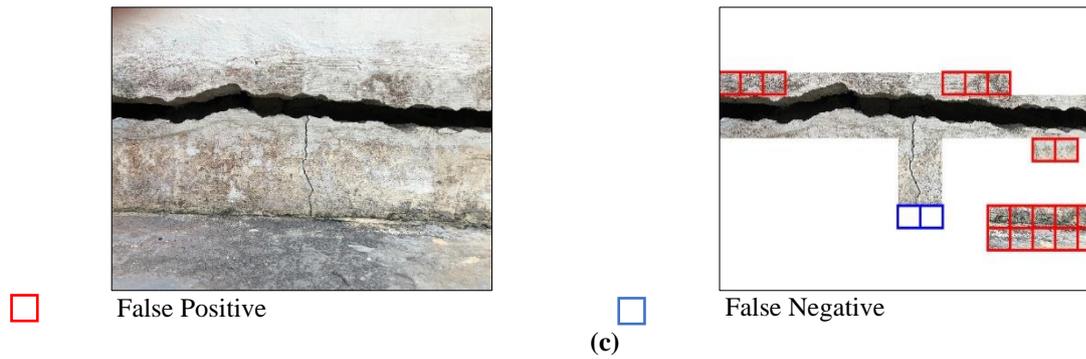
Test image	TP	FP	TN	FN	Accuracy	Precision	Recall	F1
Figure 8a	3	1	7	1	0.83	0.75	0.75	0.75
Figure 8b	4	0	8	0	1.00	1.00	1.00	1.00
Figure 8c	28	24	169	0	0.89	0.54	1.00	0.70
Figure 8d	11	14	191	5	0.91	0.44	0.69	0.54
Average					0.91	0.68	0.86	0.75

### 3.1.2. Crack Detection under Different Crack Width

Images containing various crack widths are also tested as depicted in Figure 9. The number of TP, FP, TN, and FN regions are summarised and tabulated in Table 7. It can be seen that the crack detection results for testing images are very effective regardless of crack width. The detection accuracy and recall are relatively high, which indicates

that the majority of the cracks that theoretically exist are successfully spotted. Despite the successful detection, several FP regions occurred around the crack areas. Due to the high number of such regions, it will significantly reduce the precision and F1. As evidenced in Table 7, Figure 9a is reported to have the lowest precision as well as F1.





**Fig. 9.** Images scanned using the trained CNN: a) Thin crack; b) Medium crack; and c) Thick crack

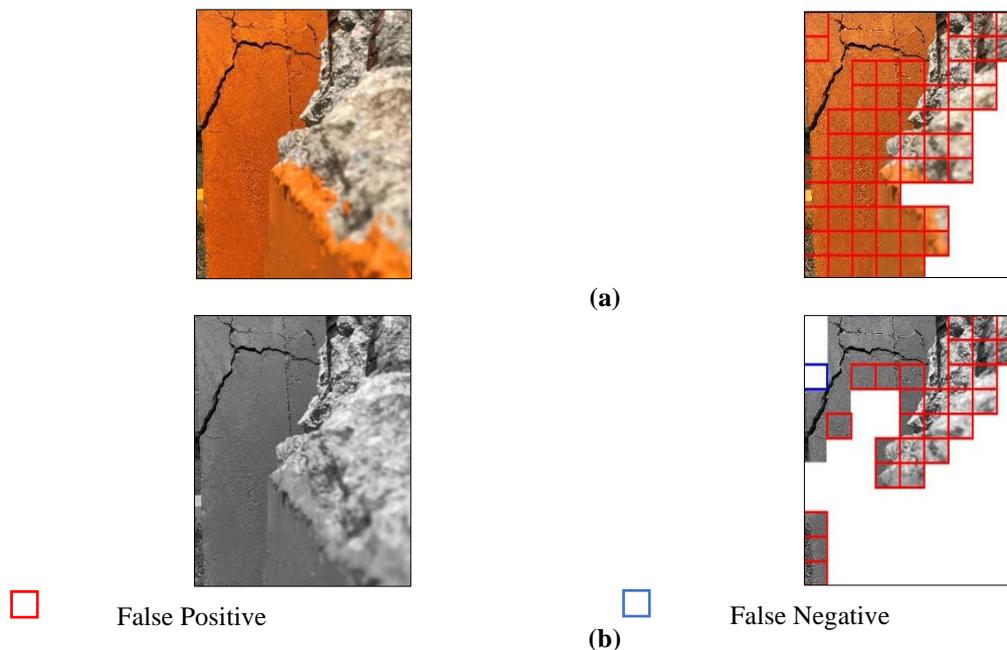
**Table 7.** Detailed analysis of test images in Figure 9

Test image	TP	FP	TN	FN	Accuracy	Precision	Recall	F1
Figure 9a	21	63	137	0	0.71	0.25	1.00	0.40
Figure 9b	21	21	177	2	0.90	0.50	0.91	0.65
Figure 9c	47	18	154	2	0.91	0.72	0.96	0.82
Average					0.84	0.49	0.96	0.62

### 3.1.3. Crack Detection under Different Colour Structures

Figure 10 shows the crack detection results with different colour structures with the proposed CNN model. It should be noted that they are made up of the same image yet different colours, in which Figure 10a is the RGB image while Figure 10b is a grayscale image. A close examination of Figures 10a and 10b reveals that the presence of orange colour resulted in a

significant increase in the number of FP regions. Consequently, the accuracy, precision, and F1 for Figure 10a is much lower than that of Figure 10b as shown in Table 8. The occurrence of a large number of FP regions are due to the colour of the testing image which is radically different from the training images used in the database. This explicitly reflects the low generalisability of Network A.



**Fig. 10.** Images scanned using the trained and validated CNN: a) Normal color; and b) Grayscale

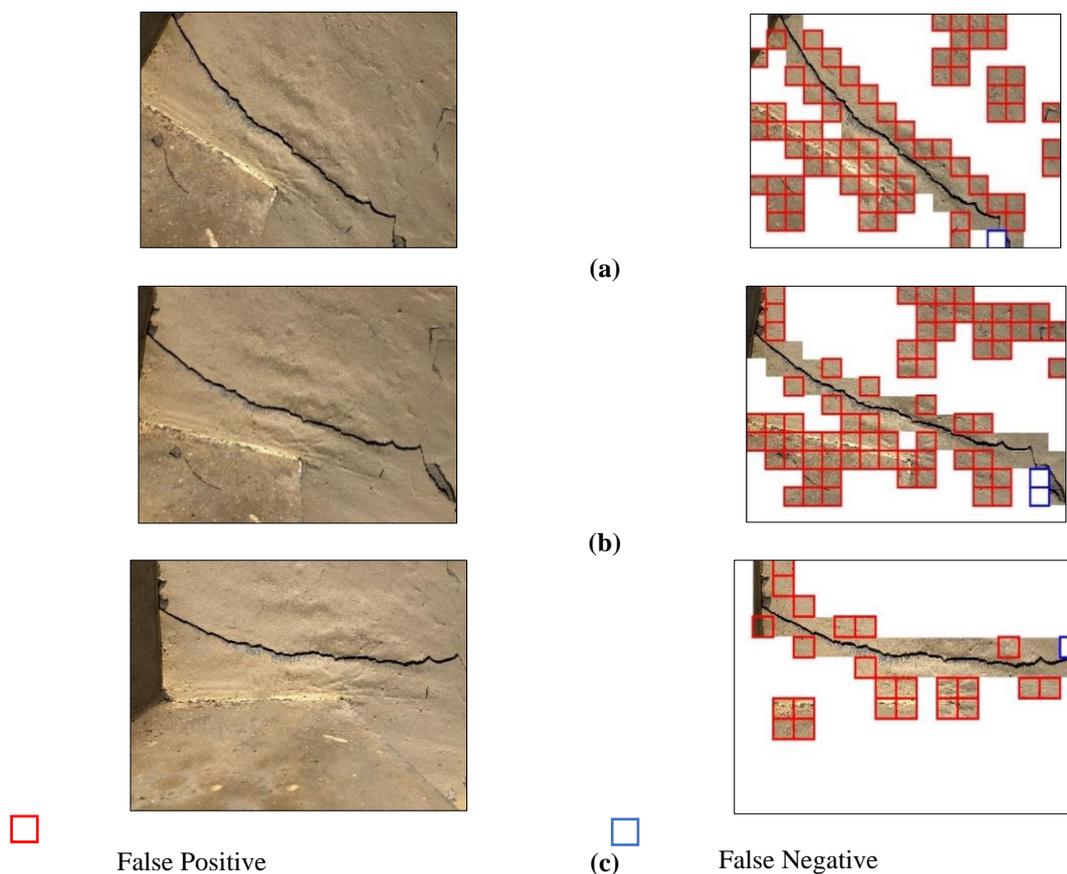
**Table 8.** Detailed analysis of test images in Figure 10

Test image	TP	FP	TN	FN	Accuracy	Precision	Recall	F1
Figure 10a	16	60	23	0	0.39	0.21	1.00	0.35
Figure 10b	16	27	55	1	0.72	0.37	0.94	0.53
Average					0.56	0.29	0.97	0.44

### 3.1.4. Crack Detection under Different Shooting Angles

Several testing images were photographed at different angles as illustrated in Figure 11. They can be divided into high angle ( $45^\circ - 90^\circ$ ), medium angle ( $0^\circ - 45^\circ$ ) and low angle ( $0^\circ$ ). Based on the results presented in Figure 11, it can be noted that the majority of the cracks in these images are successfully detected. The existence of FN regions is very minimal. However, the number of FP regions increases by increasing the shooting angle. Table 9 shows the detailed test results for Figure 11. The images acquired at the low angle achieved the highest accuracy, precision, recall, and F1. In contrast, the medium and high angle images showed

lower values for all 4 performance indicators. Both of these images showed similar values for the four performance indicators. The high number of FP regions in these testing images greatly reduced the value of precision and F1. Notwithstanding that, it is clearly shown that the accuracy and recall based on the proposed CNN are in an acceptable range and true cracks can be identified and detected. Based on the results shown in Figure 11 and Table 9, it can be concluded that the shooting angle has a degree of influence on the precision of the proposed CNN. At a higher shooting angle, the proposed CNN is expected to give a higher number of FP regions. However, the proposed CNN still satisfied the requirement of identifying and locating the cracks in the testing images.



**Fig. 11.** Images scanned using the trained and validated CNN: a) High angle; b) Medium angle; and c) Low angle

**Table 9.** Detailed analysis of test images in Figure 11

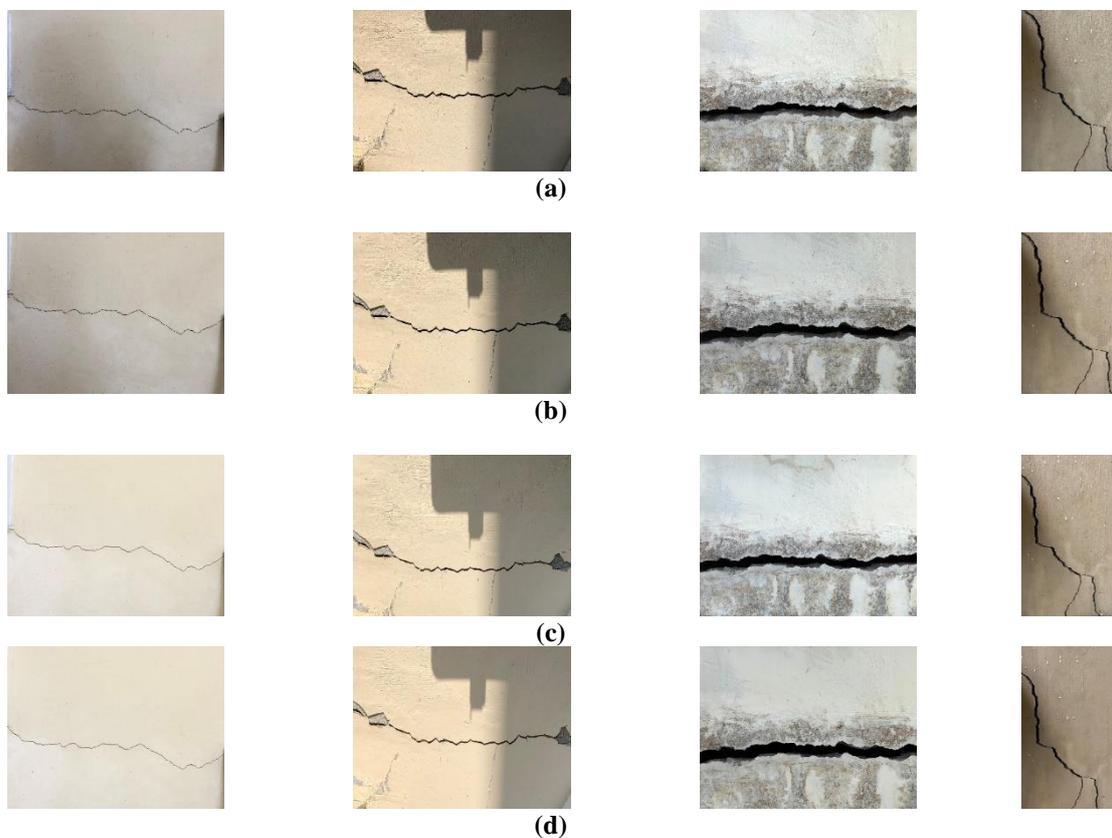
Test image	TP	FP	TN	FN	Accuracy	Precision	Recall	F1
Figure 11a	28	72	120	1	0.67	0.28	0.97	0.43
Figure 11b	32	71	116	2	0.67	0.31	0.94	0.47
Figure 11c	28	23	169	1	0.89	0.55	0.97	0.70
Average					0.74	0.38	0.96	0.53

### 3.2. Performance of the CNN under Various Image Distances

In real applications, the performance of the CNN on detecting cracks is influenced by the distance between the camera and object. In order to reduce such effects, eight networks with combinations of sharp and blurred images are trained using different datasets as shown in Table 2. Subsequently, these networks are then used to classify the images in the test sets as depicted in Figure 12. The test sets consist of four different classes of images taken in the distance of 0.5 m, 1.0 m, 1.5 m, and 2.0 m, respectively.

Figure 13 shows the performance indicators of all networks on the test sets. Among these networks, Network B appears to be the network with the best performance. The accuracy achieved by Network B increases gradually as the distance of the images in the test set increases from 0.5 m to 1.5 m. As the distance of the images increases from 1.5 m to 2.0 m, the accuracy of Network B shows a slight drop of 3.92%. The precision of

Network B consistently outperformed most of the trained networks regardless of the image distances. This implies that the likelihood of misclassifying a non-crack as a crack is very minimal. The recall of Network B is slightly unsatisfactory as compared to other trained networks. This implies that Network B tends to miss more cracks as compared to the other trained networks. Since the F1 of Network B remains the highest among the trained networks, it can be said that the high precision outweighs the low recall of Network B and the low recall problem can be compensated by incorporating a larger volume of training data (Ryu, 2018). This suggests the highest robustness of Network B for crack detection across various distances of images from 0.5 m to 2.0 m. This detection performance is achieved due to the higher rate of TP diagnosis coupled with a lower number of FP in the detection of cracks incorporating replicated images of Level 1 blurriness for training.



**Fig. 12.** Test sets taken in the distance of: a) 0.5 m; b) 1.0 m; c) 1.5 m; and d) 2.0 m

The accuracy achieved for Network A and Network H exhibited a similar trend as shown in Figure 13. Their accuracy rivals that of Network B. In fact, both Network A and Network H achieved slightly higher accuracy when the image distance was 0.5 m. However, Network A and Network H achieved different precision, recall, and F1 scores. By taking precision and recall into consideration, the F1 of Network A is significantly higher than that of Network H. Hence, it can be concluded that the true performance of Network A is better than Network H. As evidenced from Figure 13, the accuracy, precision, and F1 of Network C and D significantly increase as the distances of the images increases from 0.5

m to 2.0 m. This indicates that the performance of Networks C and D improves as the distances of the images increases. This is attributed to the fact that both of the networks were trained by using blurred images only. This finding also supports the idea that higher image distance can be captured by using a more blurred image training set. The consistently high recall achieved by Network D suggests that this network can accurately capture true cracks regardless of image distances. This implies that training with a blurred image can improve the ability of the network to capture cracks with various image distances.

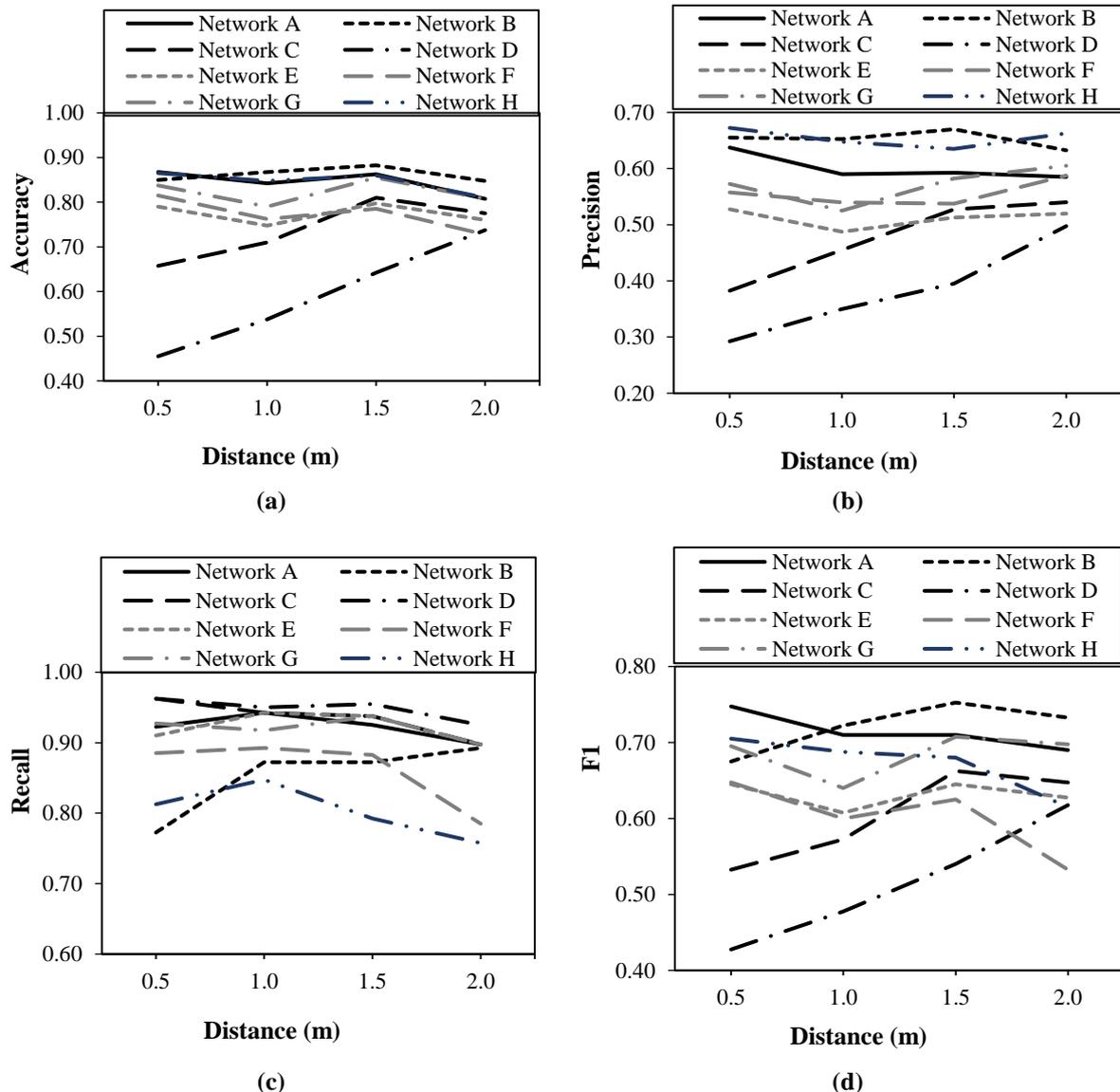


Fig. 13. Performance of all networks on the test sets: a) Accuracy; b) Precision; c) Recall; and d) F1

To further verify the performance of blurred images in the training database, Networks E, F, and G are compared. Among these three networks, it is noticeable that Network G attained the highest accuracy and F1 across various distances of images from 0.5 m to 2.0 m. It is also noted that the performance of Network F is better than Network E, and Network G is better than Network F. This means that the performance of the network improves when a higher blurriness level is used in the training database across various distances of images. Therefore, it can be said that Network G has a better true performance than the other two networks. It is noted that the performance of these networks is poorer than Networks A and H. This finding suggests that training the CNN with additional images with a single blurriness level cannot improve the performance of the network. Such a combination of training databases may result in a decrease in the performance of the network.

Different performance trends can be observed when comparing a network composed of merely blurred images with a network made up of both sharp images and blurred images. Network B consistently outperformed the accuracy and F1 of Network E for all distances of images. In contrast, Network G has a better performance on accuracy and F1 than that of Network D across various distances of images. Network C yielded a better performance as compared to Network F when the distances of images fall in the range of 1.5 m to 2.0 m. Although these networks achieved satisfactory accuracy during the training and validation process, the performance of the networks might suffer when used on images with significant deviations from the context represented in the training and validation set. Under these circumstances, the model could be improved by retraining with a wider range of databases.

#### 4. Conclusions

This paper proposed a CNN-based framework for autonomous crack detection in concrete surfaces. The study aimed to increase the performance of the CNN algorithm that was least influenced by the mismatch of the dataset in training and testing caused by image blurriness artefacts. Hence, images of concrete cracks pulled from open-source databases *Mendeley Data* and *Data in Brief* were used in this study. Selected images were compiled into a single training database for training the CNN. A total of 53,905 images with  $227 \times 227$  pixels were chosen based on the requirements set. Subsequently, each image was blurred with three levels of blurriness. Unlike previous research, data augmentation in this study was carried out automatically with the use of a built-in function in MATLAB. The training images were allowed to flip horizontally and vertically in a probabilistic approach. The proposed CNNs were successfully developed to detect concrete cracks autonomously from images, with validation accuracy attained by all networks exceeding 95%. The performance of the trained and validated CNN with the use of sharp images was evaluated using test sets with different pixels and under various lighting, crack widths, colour structures, and shooting angle conditions. A sliding window technique was deployed to perform the scanning of images larger than  $227 \times 227$  pixels. The accuracy obtained by the trained and validated CNN was lower than the validation accuracy recorded due to the presence of FPs in the region of dark lighting, dirt stains, fine scratches, and visible light spectrum. Additionally, the intercorrelations between blurriness and image distance are summarised, including:

- As the distance of the images increases, the performance of the network trained with the use of blurred images significantly increases,
- The highest performing range of blurriness level is 1, followed by 2 and 3,

- CNNs trained with a sharp image and single blurriness level do not improve the performance of the network across varying image distances,
- Networks with the combination of sharp images and all blurriness level images exhibited similar accuracy trends compared to the network with only sharp images and
- The network with the sole use of blurriness level 1 images as the dataset outperforms all other networks across varying distances of the images.

It was observed that high shooting angle and uneven illumination has a negative effect on the accuracy of the proposed CNN. Therefore, this study concludes that the presence of blurred images in training data can solve the image distance issue associated with CNN. It could revolutionise automated inspection systems.

## 5. Acknowledgement

The authors would like to thank the Ministry of Higher Education, Malaysia, and Universiti Teknologi Malaysia (UTM) for their financial support through the HICoE Grant (4J224) and Transdisciplinary Research Grant (07G46).

## 6. References

- Abdulkareem, M., Bakhary, N., Vafaei, M., Noor, N.M. and Padil, K.H. (2018). "Non-probabilistic wavelet method to consider uncertainties in structural damage detection", *Journal of Sound and Vibration*, 433(27 October), 77-98, <https://doi.org/hdl.handle.net/11250/238284>.
- Abudallah Habib, I., Wan Mohtar, W.H.M., Muftah Shahot, K., El-shafie, A. and Abd Manan, T.S. (2021). "Bridge failure prevention: An overview of self-protected pier as flow altering countermeasures for scour protection", *Civil Engineering Infrastructures Journal*, 54(1), 1-22, <https://doi.org/10.22059/CEIJ.2020.292296.1627>.
- Atha, D.J. and Jahanshahi, M.R. (2018). "Evaluation of deep learning approaches based on convolutional neural networks for corrosion detection", *Structural Health Monitoring*, 17(5), 1110-11128, <https://doi.org/10.1177/1475921717737051>.
- Cha, Y.J., Choi, W. and Büyüköztürk, O. (2017). "Deep learning-based crack damage detection using convolutional neural networks", *Computer-Aided Civil and Infrastructure Engineering*, 32(5), 361-78, <https://doi.org/10.1111/mice.12263>.
- Cha, Y.J., Choi, W., Suh, G., Mahmoudkhani, S. and Büyüköztürk, O. (2018). "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types", *Computer-Aided Civil and Infrastructure Engineering*, 33(9), 731-747, <https://doi.org/10.1111/mice.12334>.
- Chen, F.C. and Jahanshahi, M.R. (2020). "ARF-crack: Rotation invariant deep fully convolutional network for pixel-level crack detection", *Machine Vision and Applications*, 31(6), 1-12, <https://doi.org/10.1007/s00138-020-01098-x>.
- Ciresan, D.C., Meier, U., Masci, J., Gambardella, L.M. and Schmidhuber, J. (2011). "Flexible, high performance convolutional neural networks for image classification", *Twenty Second International Joint Conference on Artificial Intelligence*, Catalonia, Spain, <https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-210>.
- Dash, R., Sa, P. K., and Majhi, B. (2009). "RBFN based motion blur parameter estimation", *International Conference on Advanced Computer Control*, 327-331, IEEE, Singapore, <https://doi.org/10.1109/ICACC.2009.98>.
- Delatte, N. (2009). *Failure, distress and repair of concrete structures*, Elsevier, <https://doi.org/10.1533/9781845697037>.
- Deng, L., Chu, H.H., Shi, P., Wang, W. and Kong, X. (2020). "Region-based CNN method with deformable modules for visually classifying concrete cracks", *Applied Sciences*, 10(7), 2528, <https://doi.org/10.3390/app10072528>.
- Dorafshan, S. and Maguire, M. (2018). "Bridge inspection: Human performance, unmanned aerial systems and automation", *Journal of Civil Structural Health Monitoring*, 8(July), 443-476.
- Dorafshan, S., Thomas, R.J. and Maguire, M. (2018). "Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete", *Construction and Building Materials*, 186(20 October), 1031-1045, <https://doi.org/10.1007/s13349-018-0285-4>.
- Fan, Y., Zhao, Q., Ni, S., Rui, T., Ma, S. and Pang, N. (2018). "Crack detection based on the mesoscale geometric features for visual concrete bridge inspection", *Journal of Electronic Imaging*, 27(5), 053011, <https://doi.org/10.1117/1.JEI.27.5.053011>.
- Fankhauser, N., Kalberer, N., Müller, F., Leles, C.R., Schimmel, M. and Srinivasan, M. (2020). "Comparison of smartphone-camera and

- conventional flatbed scanner images for analytical evaluation of chewing function", *Journal of oral rehabilitation*, 47(12), 1496-502, <https://doi.org/10.1111/joor.13094>.
- Han, X., Thomasson, J.A., Bagnall, G.C., Pugh, N.A., Horne, D.W., Rooney, W.L., Jung, J., Chang, A., Molambo, L., Popescu, S.C., Gates, I.T. and Cope, D.A. (2018). "Measurement and calibration of plant-height from fixed-wing UAV images", *Sensors*, 18(12), 4092, <https://doi.org/10.3390/s18124092>.
- Jang, K., Kim, N., and An, Y.-K. (2019). "Deep learning-based autonomous concrete crack evaluation through hybrid image scanning", *Structural Health Monitoring*, 18(5-6), 1722-1737, <https://doi.org/10.3390/s18124092>.
- Kim, B. and Cho, S. (2018). "Automated vision-based detection of cracks on concrete surfaces using a deep learning technique", *Sensors*, 18(10), 3452, <https://doi.org/10.3390/s18103452>.
- Li, S., Zhao, X. and Zhou, G. (2019). "Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network", *Computer-Aided Civil and Infrastructure Engineering*, 34(7), 616-34, <https://doi.org/10.1111/mice.12433>.
- Li, Y., Zhao, W., Zhang, X. and Zhou, O. (2018). "A two-stage crack detection method for concrete bridges using Convolutional Neural Networks", *IEICE Transactions on Information and Systems*, 101(12), 3249-3252, <https://doi.org/10.1587/transinf.2018EDL8150>.
- Protopapadakis, E., Voulodimos, A., Doulamis, A., Doulamis, N. and Stathaki, T. (2019). "Automatic crack detection for tunnel inspection using deep learning and heuristic image post-processing", *Applied intelligence*, 49(2), 2793-2806, <https://doi.org/10.1007/s10489-018-01396-y>.
- Ratnam, M.M., Ooi, B.Y. and Yen, K.S. (2019). "Novel moiré-based crack monitoring system with smartphone interface and cloud processing", *Structural Control and Health Monitoring*, 26(10), e2420, <https://doi.org/10.1002/stc.2420>.
- Ryu, P.-M. (2018). "Predicting the unemployment rate using social media analysis", *Journal of Information Processing Systems*, 14(4), 904-915, <https://doi.org/10.3745/JIPS.04.0079>.
- Shen, L., Fang, R., Yao, Y., Geng, X. and Wu, D. (2018). "No-reference stereoscopic image quality assessment based on image distortion and stereo perceptual information", *IEEE Transactions on Emerging Topics in Computational Intelligence*, 3(1), 59-72, <https://doi.org/10.1109/TETCI.2018.2804885>.
- Sieberth, T., Wackrow, R., and Chandler, J. (2013). "Automatic isolation of blurred images from UAV image sequences", *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Rostock, Germany.
- Sieberth, T., Wackrow, R. and Chandler, J. (2015). "UAV image blur-its influence and ways to correct it", *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Toronto, Canada, <https://doi.org/10.5194/isprsarchives-XL-1-W4-33-2015>.
- Wang, Y., Fang, Z. and Hong, H. (2019). "Comparison of convolutional neural networks for landslide susceptibility mapping in Yanshan County, China", *Science of The Total Environment*, 666(20 May), 975-93, <https://doi.org/10.1016/j.scitotenv.2019.02.263>.
- Yang, J. and Byun, H. (2007). "Illumination compensation algorithm using eigenspaces transformation for facial images", In: Gagalowicz, A., Philips, W. (eds.), *Computer Vision/Computer Graphics Collaboration Techniques (MIRAGE 2007)*, Lecture Notes in Computer Science, Vol. 4418, Springer, Berlin, Heidelberg, [https://doi.org/10.1007/978-3-540-71457-6\\_18](https://doi.org/10.1007/978-3-540-71457-6_18).
- Ye, X.-W., Dong, C.-Z. and Liu, T. (2016). "A review of machine vision-based structural health monitoring: methodologies and applications", *Journal of Sensors*, 2016, Article ID 7013039, <https://doi.org/10.1155/2016/7103039>.
- Zhang, Y., Sun, X., Loh, K.J., Su, W., Xue, Z. and Zhao, X. (2020). "Autonomous bolt loosening detection using deep learning", *Structural Health Monitoring*, 19(1), 105-22, <https://doi.org/10.1177/1475921719837509>.
- Zhou, S. and Song, W. (2021). "Deep learning-based roadway crack classification with heterogeneous image data fusion", *Structural Health Monitoring*, 20(3), 1274-1293, <https://doi.org/10.1177/1475921720948434>.



This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license.