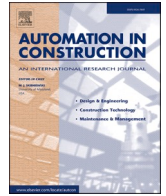




Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autconCombining block-based and pixel-based approaches to improve crack detection and localisation[☆]Mohamed Abdellatif^{a,c,*}, Harriet Peel^a, Anthony G. Cohn^{b,c,d,e,f}, Raul Fuentes^g^a School of Civil Engineering, University of Leeds, Woodhouse lane, LS2 9JT Leeds, UK^b Luzhong Institute of Safety, Environmental Protection Engineering and Materials, Qingdao University of Science & Technology, Zibo 255000, China^c School of Computing, University of Leeds, Woodhouse lane, LS2 9JT Leeds, UK^d School of Mechanical and Electrical Engineering, Qingdao University of Science and Technology, Qingdao 260061, China^e Department of Computer Science and Technology, Tongji University, Shanghai 211985, China^f School of Civil Engineering, Shandong University, Jinan 250061, China^g Institute of Geotechnical Engineering, RWTH Aachen University, Mies-van-der-Rohe-Straße 1, D52074 Aachen, Germany

ARTICLE INFO

Keywords:

Crack discretisation
Noise tolerance
Accurate crack localization
Pavement distress
Crack repair system
CrackIT

ABSTRACT

A variety of civil engineering applications require the identification of cracks in roads and buildings. In such cases, it is frequently helpful for the precise location of cracks to be identified as labelled parts within an image to facilitate precision repair for example. CrackIT is known as a crack detection algorithm that allows a user to choose between a block-based or a pixel-based approach. The block-based approach is noise-tolerant but is not accurate in edge localization while the pixel-based approach gives accurate edge localisation but is not noise-tolerant. We propose a new approach that combines both techniques and retains the advantages of each. The new method is evaluated on three standard crack image datasets. The method was compared with the CrackIT method and three deep learning methods namely, HED, RCF and the FPHB. The new approach outperformed the existing arts and reduced the discretisation errors significantly while still being noise-tolerant.

1. Introduction

Many civil engineering applications require to precisely locate cracks, either to then repair them or to assess the health of the structure. In particular, cracks in pavements for roads, airport runways / taxiways or other situations can reduce the serviceable life of the pavement significantly [1]. For these reasons, pavement crack detection has attracted significant attention lately [2–6]. However, most of the effort has been dedicated to detection of the presence of cracks [7–10] and less to measuring its characteristics. Only few methods exist [11–13] but their cost and complexity reduces the frequency of their use and limits the applicability to less critical infrastructure. This infrastructure is typically under-invested in for maintenance purposes, and therefore shows greater defects such as cracks resulting in poorer performance.

Additionally, the poor localisation of cracks prevents this valuable information for being accurately incorporated into asset management processes, not least those managed or making use of Building

Information Modelling (BIM) techniques [14]. Other more accurate localisation solutions have been presented in [15–18]. These are mostly based on computer vision techniques applied to images. However, these systems can be ‘too successful’ in detecting multiple cracks and therefore result in multiple false positives. Hence, a cost-effective technique that can help to accurately localise cracks and reduce false positives, would be of great use. The remainder of the paper will focus on image-based crack detection and measurement.

The literature of image-based crack detection is quite large, therefore, we will only focus on work done for accurate crack localisation (which has also been termed as quantification or discretisation in the literature) and we have defined here as the identification of crack pixels within an image.

Crack detection from 2D visual image information can be broadly classified into geometric model-based and deep learning approaches. The model-based approaches may exploit one or more basic image processes such as edge detection, segmentation, morphological

[☆] This document is the result of the research project funded by the (ESPRC) award reference EP/NO10523/1.

* Corresponding author at: School of Civil Engineering, University of Leeds, Woodhouse lane, LS2 9JT Leeds, UK.

E-mail addresses: m.abdellatif@leeds.ac.uk (M. Abdellatif), cnhap@leeds.ac.uk (H. Peel), a.g.cohn@leeds.ac.uk (A.G. Cohn), raul.fuentes@rwth-aachen.de (R. Fuentes).

<https://doi.org/10.1016/j.autcon.2020.103492>

Received 28 May 2020; Received in revised form 18 November 2020; Accepted 21 November 2020

Available online 16 December 2020

0926-5805/© 2020 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

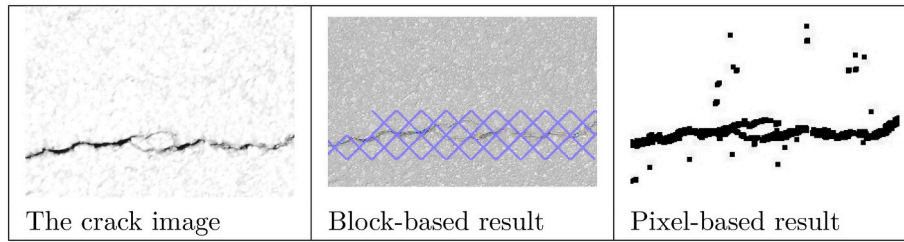
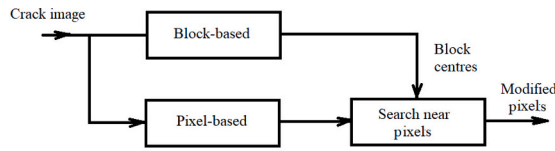
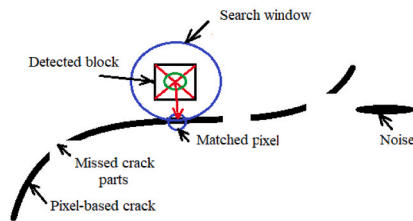


Fig. 1. Zoomed section showing a crack image, block-based and pixel-based (dilated for clarity) crack detection results.



(a) Diagram showing the use of both pixel-based and block-based approaches.



(b) Schematic showing the combined mechanism for searching pixels near to block centre lying on the pixel-based contour output.

Fig. 2. The combined method.

operations and texture analysis to extract crack features from the images [19]. There have been several attempts, in the model-based approach, to improve the reliability of crack detection using image pre-processing and different kinds of spatial smoothing. This can reduce image noise and hence improve the crack detection results. Pre-processing steps may include median filters, opening and closing morphological filters to join crack segments [20–23].

Cracks on concrete surfaces were quantified in [24–26] with the objective of estimating crack width and length rather than the accurate localization for a consequent repair task. Cracks were discretized in [27] using the minimal path selection concept to improve crack detection against the imperfection of incomplete and sparse crack segments.

Deep learning approaches on the other hand, can learn the process from end to end using neural networks trained using hundreds of labelled/annotated images [17,28–37]. Recent developments in the deep learning showed good results through pixelwise segmentation with Unet

Table 1
Summary of test datasets.

Dataset	Number of images	Resolution	Image type
CrackIT [18]	48	2048 × 1536	PNG
Crack Forest (CFD) [43]	118	480 × 320	JPG
CRACK500 [37,38]	200	2560 × 1440	PNG

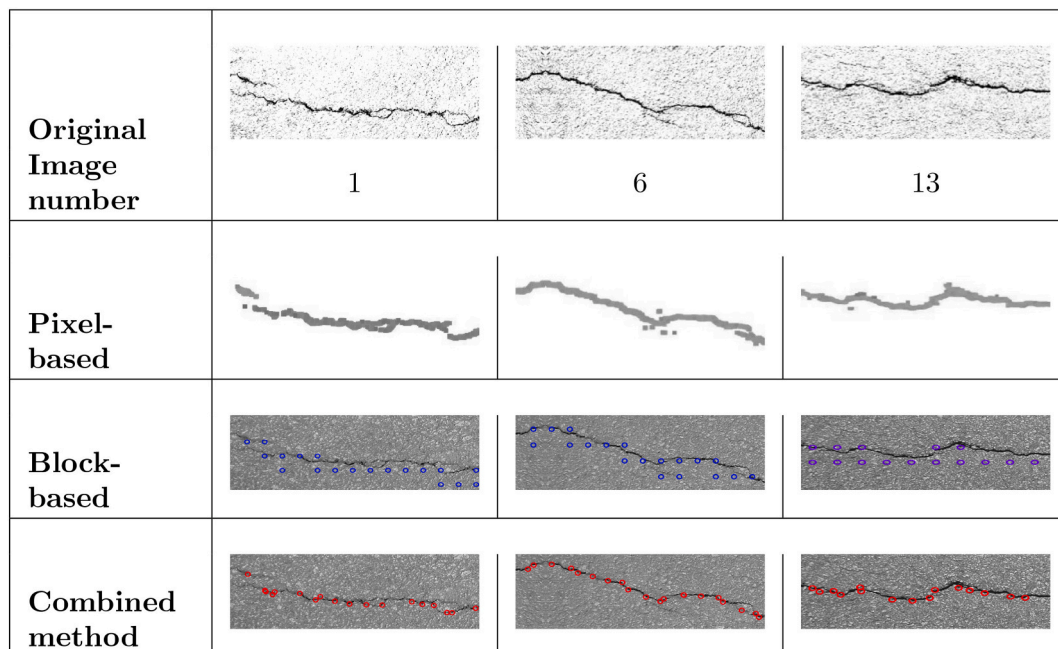


Fig. 3. Sample of crack images from the CrackIT database [18] showing pixel-based cracks (dilated for clarity), block-based cracks (block size 75×75 pixels) and the combined cracks discretisation. Original image size is 2048 × 1536 pixels and these samples are cropped to 1000× 300 pixels.

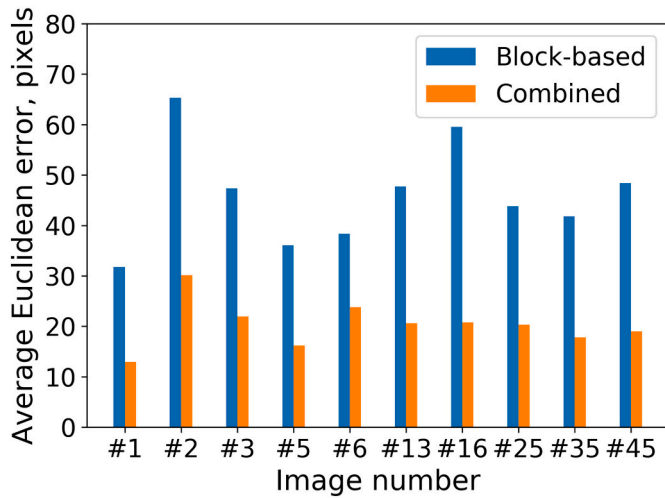


Fig. 4. Average Euclidean errors in pixels for 10 sample crack images for both block-based and combined crack detection methods with a block size of 75×75 pixels.

Table 2

Comparison of errors for block-based, pixel-based and combined crack detection methods at a block size of 75×75 pixels for 24 crack images (image no. 1–24) of the CrackIT dataset [18]. In each image, the combined method has the best precision and the lowest error.

Image no.	Pixel-based recall	Pixel-based precision	Combined method precision	Block-based Euclidean error, pixels	Combined method Euclidean error, pixels
001	0.85	0.85	1	31.77	12.92
002	0.92	0.78	0.92	65.2	30.17
003	0.91	0.57	0.88	47.38	21.93
004	0.96	0.70	0.88	40.68	11.88
005	1	0.73	0.97	36.0	16.2
006	0.90	0.85	0.92	38.3	28.8
007	1	0.68	0.82	43.5	24.6
008	0.86	0.78	1	44	23.4
009	0.85	0.77	0.94	43.7	15.4
010	0.89	0.61	1	41	14.4
011	0.91	0.72	1	37.7	29.7
012	0.92	0.76	0.95	56.4	25.20
013	0.8	0.63	1	47.7	20.6
014	0.88	0.70	1	48.9	24.8
015	0.93	0.50	0.9	52.6	22.4
016	0.90	0.71	1	59.5	20.7
017	0.78	0.51	0.88	39.6	13.70
018	0.76	0.76	1	45.1	23.7
019	0.89	0.80	1	30.3	18.6
020	0.96	0.69	1	42.9	13.4
021	0.89	0.75	1	50.1	25.2
022	0.86	0.68	1	50.6	20.8
023	0.88	0.61	1	48.0	17.8
024	0.93	0.729	1	42.4	21.5

models [38–40]. Several commercial crack identification systems have been tested in the field and reported to have problems with non-crack features present on the road surface, including joints, patches, road markings, resulting in high number of false positives in the reported cracks than those present in the reference data [41]. Since the main issue with crack detection is the appearance of noise, and consequently the increase of false positives, our research strategy aims to explore ways to enhance both the crack detection noise tolerance and accurate localisation.

Oliveira et al. [18] developed the CrackIT algorithm to detect road cracks and characterize them into basic types. It has two modes of operation: (1) pixel-based that segments the crack but is sensitive to

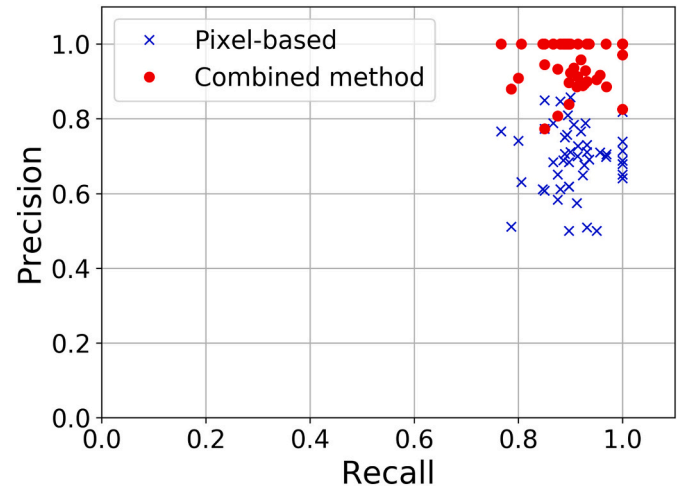


Fig. 5. The Precision recall comparison between the pixel-based and combined methods for all 48 images in the CrackIT dataset.

Table 3

Summary of performance metrics for 48 images of CrackIT dataset at a 75×75 block size.

Metric	Pixel-based	Combined method
Recall	0.9072	0.9072
Precision	0.69	0.952
F1 score	0.7849	0.9292

noise and, (2) block-based to determine whether the block contains a crack or not but whilst it is noise tolerant it does not accurately localise cracks.

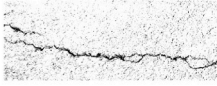
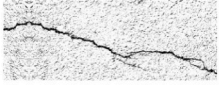
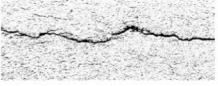



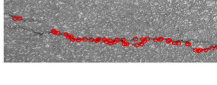
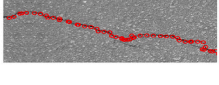
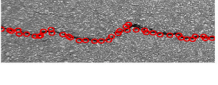
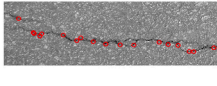
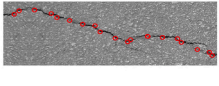
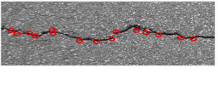
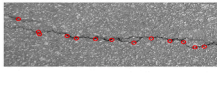
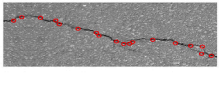
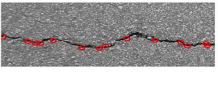
In this paper, the benefits of combining crackIT’s two modes of crack detection (pixel-based and block-based) are explored to develop a robust method that can achieve both accurate localisation and a low level of false positives induced by noise. This is the first most critical step towards the use of crack information for asset management, leaving the image localisation outside the scope of this paper as it requires different techniques.

The paper is structured as follows: the next section describes the CrackIT methods and its modes of operation briefly together with the proposed combined method. The experiment conditions, metrics and comparison with pixel-based method are presented in Section 3 using the CrackIT image dataset. The proposed method is then compared with three deep learning methods running on three crack image datasets and the results are presented in Section 4. The results are discussed in Section 5 and conclusions are finally drawn in Section 6.

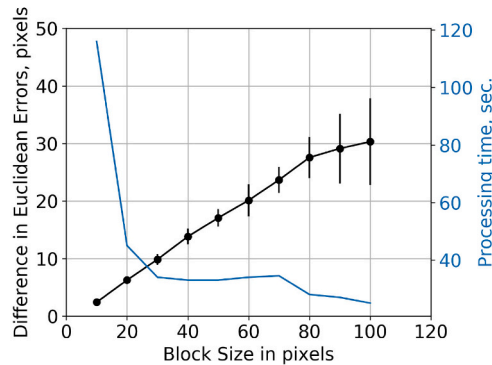
2. The proposed method

2.1. The CrackIT method

The CrackIT algorithm offers two modes: block-based and pixel-based. The block-based approach divides the image into a set of non overlapping windows and makes use of both the average intensity and standard deviation to make a decision if the block contains a crack or normal block. This output delivers only the centre of the block with a label. The benefit of this approach is that the decision is effective and more robust to scattered noise. A zoomed example is shown in Fig. 1. It is clear that the block centres (represented by blue X signs) are far from the crack real contour centre. The pixel-based approach on the other hand tries to segment individual image pixels based on a dual intensity threshold automatically computed for each image to discriminate between crack pixels and background pixels. The output pixels are then

Original image number			
Pixel-based			
Combined (30x30 pixels)			
Combined (75x75 pixels)			
Combined (90x90 pixels)			

(a) Sample of crack images from the CrackIT database [18] showing combined cracks discretisation at block sizes of 30×30, 75×75 and 90×90 pixels,(the image is rotated for display purposes in third column since the crack is vertical in the original image).



(b) Effect of block size on the average differences between combined method coordinates and block-based coordinates as described in Eq. 1. The dots represent the mean difference and the vertical bars show the limits of one standard deviation.

Fig. 6. Sample showing effect of block size and the graph showing error changes versus size.

grouped using a connected component algorithm. Fig. 1 (right) shows the pixel-based output image suffering from noisy pixels which represent false positives.

2.2. The combined approach

In this paper we explore whether the combination of both block-based and pixel-based modes would be beneficial. The overall architecture of the algorithm is shown schematically in Fig. 2a. The concept is illustrated schematically in Fig. 2b. Both the pixel-based and block-based modes of CrackIT are initiated and the centres of the blocks are used as an initial estimate. Then, we take the pixel-based image output and compute the distance between the nearest pixel to block centre using a distance transform. Distance transforms are used popularly in computer vision applications for proximity-based matching [42]. Pixel-

based output images are searched for the nearest crack point lying directly on the crack contour with minimum Euclidean distance to a block centre.

The pseudo code of the new combined method is shown in Algorithm 1.

Algorithm 1. Pseudo-code for combining block-based and pixel-based crack detection modes.

Data: Input image containing cracks and required block size

Result: Crack discretized points' image coordinates

Initialization;

Run CrackIT in Block-based mode;

Run CrackIT in Pixel-based mode;

Compute distance transform for the pixel-based image;

while not last pixel of block centre pixels **do**

 Compute distance from nearest pixels in Pixel-based image;

 Select the pixel with minimum distance;

 Replace the block centre pixel location by the nearest pixel location;

end

Algorithm 1: Pseudo-code for combining block-based and pixel-based crack detection modes.

3. Experiments

3.1. Conditions

The performance of the combined algorithm is tested using three standard datasets as summarized in Table 1. The first dataset is CrackIT, and the public code implementation in MATLAB is used in our experiments [18] with the authors' permission. The CrackIT database provides ground truth only for block-based mode with a coarse block size of 75×75 pixels. Therefore, the images were annotated manually to get the ground truth points at equal intervals of 75 pixels between horizontal or vertical scan lines. The block size is critical for the accuracy of the results and therefore, an investigation into its effect is presented in subsection 3.4. The ground truth data for the other two datasets are provided from sources [38,43].

The new method will be compared with three deep learning methods, namely Holistically-Nested Edge Detection (HED) [44], Richer Convolutional Features (RCF) [45] and the Feature Pyramid and Hierarchical Boosting network (FPHB) [38].

3.2. Assessment metrics

The assessment of crack localisation performance needs a metric that can measure the quality of detection and quality of localization. The error is thus defined as the average image distance between detected pixels and the ground truth pixels on the crack contour as follows Eq [1-8]:

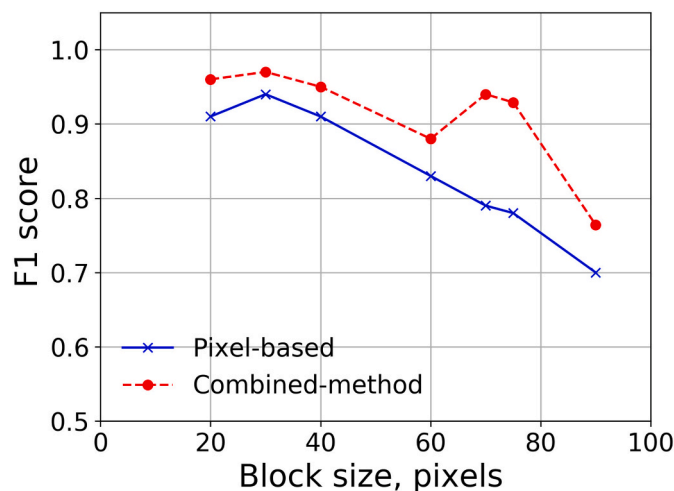


Fig. 7. F1 score comparison when block size is changed for both pixel-based and combined crack detection methods.

$$E = \frac{1}{N} \sum_{i=1}^N \sqrt{(x_i - x_{gt})^2 + (y_i - y_{gt})^2} \quad (1)$$

where N is the number of detected points, (x_i, y_i) are the coordinates of the detected pixels in image and (x_{gt}, y_{gt}) are the coordinates of the ground truth pixel. Recall-precision analysis is used together with the F1 score, mean Intersection over Union (mIoU) and Accuracy metrics, defined as follows:

$$Recall = Re = \frac{TP}{TP + FN} \quad (2)$$

$$Precision = Pr = \frac{TP}{TP + FP} \quad (3)$$

$$F1 = \frac{2 \cdot Pr \cdot Re}{Pr + Re} \quad (4)$$

$$mIoU = \frac{TP}{TP + FP + FN} \quad (5)$$

$$Accuracy = Acc = \frac{TP + TN}{TP + TN + FP + FN} \quad (6)$$

where TP is true positives, FN is false negatives, TN is true negatives and FP is false positives. Two other metrics based on F1 are also adopted in the evaluation: the best F1 on the public database for a fixed threshold (ODS), and the aggregate F1 on the public database for the best threshold in each image (OIS) [38] and defined by

$$ODS = \max \left\{ \frac{2 \times Pr_t \times Re_t}{Pr_t + Re_t} : t = 0.01, 0.02, \dots, 0.99 \right\} \quad (7)$$

$$OIS = \frac{1}{N_{img}} \sum_i^{N_{img}} \max \frac{2 \times Pr_t^i \times Re_t^i}{Pr_t^i + Re_t^i} : t = 0.01, 0.02, \dots, 0.99 \quad (8)$$

The values, i , and N_{img} are the threshold, index and the number of the images. The parameters Pr_t , Re_t , Pr_t^i and Re_t^i are precision and recall based on threshold t and i th image, respectively.

3.3. Comparison with pixel-based method

Fig. 3 shows three zoomed sections of sample cracks (numbers 1, 6 and 13 in the CrackIT database), pixel-based output and block based output for 75×75 pixels block size. The block centres are represented by blue circles and deviate from the crack contour. The improved performance of the combined approach is clearly represented by the red circles in the figure.

The ground truth coordinate pixels were annotated in the detected pixel-based contour at the test block size intervals. The average distance

Name	CrackIT-Crack001	CrackIT-Crack006	CrackIT-Crack031
Original			
Ground truth			
HED			
RCF			
FPHB			
Combined			

Fig. 8. Crack extraction for sample images from the CrackIT dataset. Colour code, white: true positive, red false positive, cyan: false negative. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

was computed for each image according to Eq. 1 and results are shown in Fig. 4. As expected, the error of the combined method is significantly lower than the errors for the block-based approach for every test image.

The results for 24 images of the 48 image dataset are summarized in Table 2. The reduction in Euclidean error ranges from around 21% to 78% with an average of 55.47%. This also shows that the new combined method can discretize the crack contour more accurately and can drive tools to crack repair or filling. The table also shows the recall and precision computed for pixel-based and combined methods. The precision of the combined method is higher than that for the pixel-based for all test images. This table confirms that the combined method enjoys better noise rejection and the resulting points are much closer to the crack real profile, therefore it is better localised.

Fig. 5 shows the precision-recall comparison between both methods as a scatter plot. Each point in the graph summarizes the precision and

recall values of one test image. It is observed that the precision of the combined method is significantly higher than that for the pixel-based method.

The average precision increased for the whole set from 0.695 (pixel-based) to 0.95 (combined method) at an average recall of 0.9. This corresponds to an F1 score change from 0.78 (pixel-based) to 0.929 (combined method) at a block size of 75×75 pixels which is an increase of 15%. This demonstrates that noise is reduced in the combined method. The results for the 48 images' dataset are summarized in Table 3.

3.4. Effect of block size

The effect of block size is also investigated by studying the test images for a range of non-overlapping blocks of sizes ranging from 10






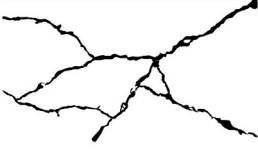
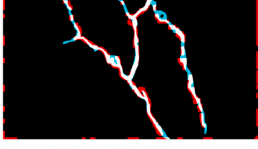
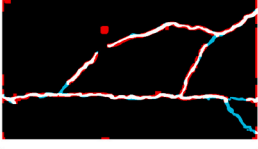
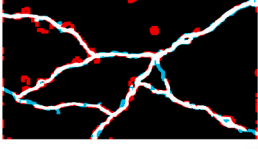
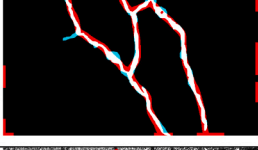
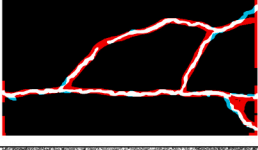
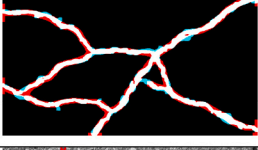
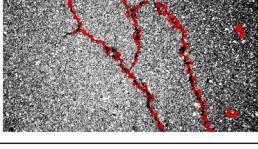
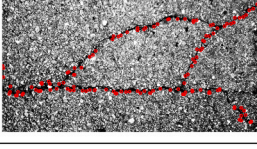
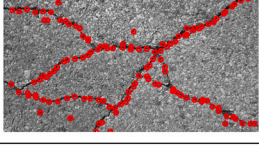
Name	CRACK500-20160225_114535	CRACK500-20160308_160128	CRACK500-20160316_144431
Original			
Ground truth			
RCF			
FPHB			
Combined			

Fig. 9. Crack extraction for sample images from the CRACK500 dataset. Colour code, white: true positive, red false positive, cyan: false negative. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)







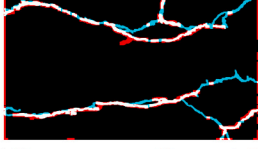
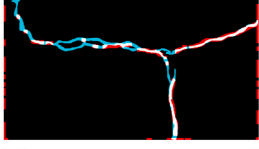
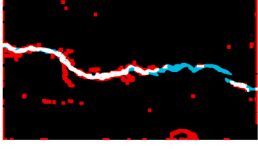
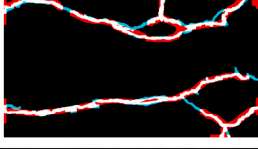
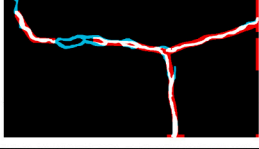
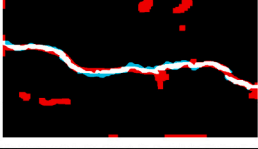
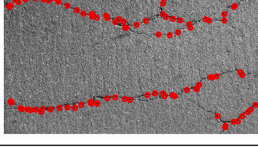
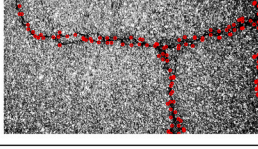
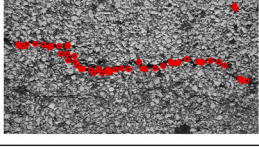
Name	CRACK500-20160308_155737	CRACK500-20160308_155847	CRACK500-20160329_094507
Original			
Ground truth			
RCF			
FPHB			
Combined			

Fig. 10. Crack extraction for sample images from the CRACK500 dataset. Colour code, white: true positive, red false positive, cyan: false negative. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)




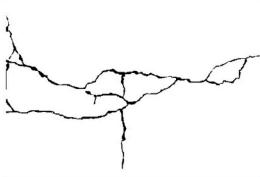

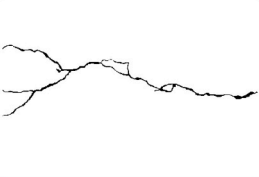
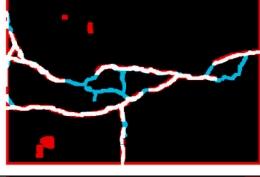
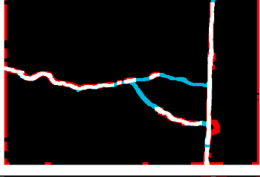
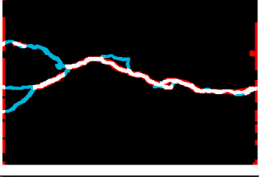
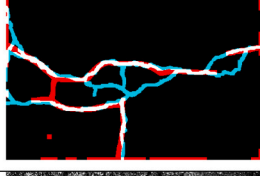
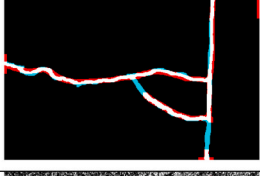
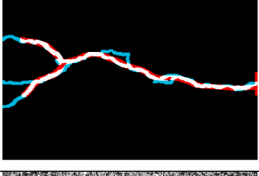
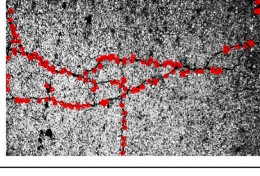
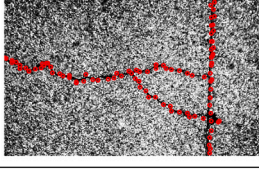
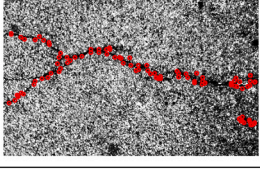
Name	CFD-034	CFD-066	CFD-115
Original			
Ground truth			
RCF			
FPHB			
Combined			

Fig. 11. Crack extraction for sample images from the Crack Forest (CFD) dataset. Colour code, white: true positive, red false positive, cyan: false negative. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

pixels to 100 pixels at an equal intervals of 10 pixels. The effect of size on the quality of combined output is shown in Fig. 6a. It can be observed that the smaller block sizes give better results in terms of spatial resolution. The difference between the Euclidean errors of crack points computed by the combined method and the points computed by block-based method are computed and averaged according to Eq. 1.

Fig. 6 shows these differences and it can be observed that as the block size increases, the error increases, as might be expected. Bigger block sizes are not desirable since it will reduce the crack digitisation resolution, while smaller block sizes are good to improve resolution. Fig. 6 also shows the processing time of the algorithm running on a Thinkpad T460 with Intel i5-6200U CPU (2.30GHz \times 4).

The processing time increases when the block size is decreased and increases significantly for block sizes less than 20×20 pixels at which it increases several fold of its value at larger block size. The selection of block size will be determined mainly from the requirement of spatial resolution for a particular application and can be improved by changing camera optics or camera sensor or digitizing resolution.

The change of F1 score when the block size is changed, is shown in

Fig. 7 and it is observed that F1 for combined method is higher than the pixel based method for all block sizes. The F1 score increases when the block size is decreased in general for both methods. The combined method has a local peak around the size of 70–75 pixels which is the recommended size by CrackIT authors [18].

4. Comparison with deep learning methods

The combined method is compared with HED [44], RCF [45] and FPHB [38] tested on the CrackIT, CRACK500 and the CFD datasets and the complete set of results can be found in <https://doi.org/10.5518/937>. The deep learning methods considered here produce thick lines, and to overcome this, thinning provided by Zhang Suen's method [46] was used for Non Maximum Suppression (NMS). Then, the thinned cracks were dilated to be of similar thickness as ground truth cracks, similar to the approach in [47]. Sample results of the combined method compared with the deep learning methods are presented in Fig. 8 from the CrackIT dataset. The HED images show false positives and miss several crack segments (false negatives). The borders for all deep learning methods




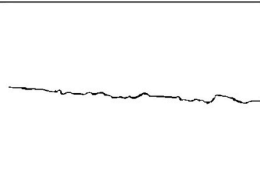
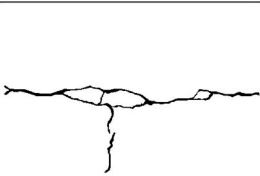
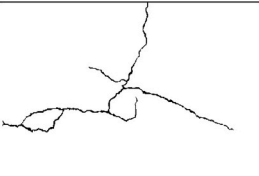
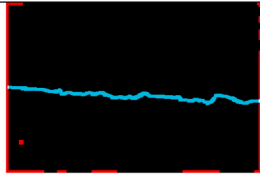
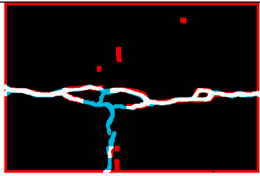

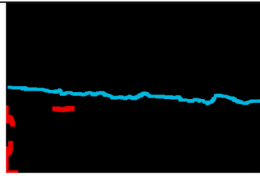
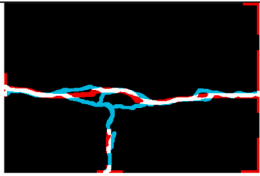
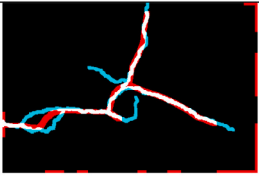
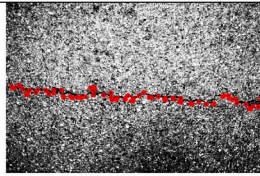
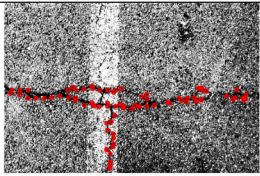
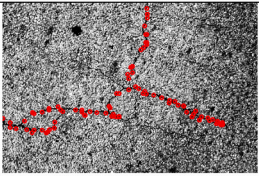
Name	CFD-039	CFD-084	CFD-102
Original			
Ground truth			
RCF			
FPHB			
Combined			

Fig. 12. Crack extraction for sample images from the Crack Forest (CFD) dataset. Colour code, white: true positive, red false positive, cyan: false negative. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

show false positives. The RCF results are slightly better than the HED in terms of fewer false positives but still miss several crack segments. The deep learning methods smooth out the spatial details and this becomes clear in bends. The combined method results show few false positive pixels and well localised cracks.

We will limit comparison in the two other datasets to the RCF and FPHB to display more cases, but they will all be summarized statistically later. The sample results for the CRACK500 dataset (only 200 images from the test section) are presented in Fig. 9 and Fig. 10. In Fig. 9, both FPHB and the combined method can detect the cracks well while RCF has more false positives and misses several crack segments. More differences are shown in cases of Fig. 10, where the left and middle images show that the FPHB and RCF miss several crack segments that are recovered in the combined method. The right image also shows that FPHB and RCF have more false positives compared with the combined method. A sample of the results running on CFD is shown in Fig. 11 and Fig. 12. The CFD dataset required preprocessing by histogram equalisation when using the combined method, because the histograms of images were quite narrow and thus hard to detect cracks. The RCF has several false positives and misses several crack segments. Several crack segments are also missing in the FPHB results especially in the Fig. 12

cases. The combined method can detect cracks in these images with less missing crack segments and with good localisation. Both FPHB and the combined method are tolerant to the appearance of road lane markings as shown in the middle image of Fig. 12.

The image borders contain several false positives in the deep learning method (although the borders were excluded from the metrics calculation process). The deep learning results suffer poor localisation in sudden bends and junctions or curves as shown in Fig. 13. The images show that in several cases the FPHB results deviate from the crack and the crack spatial details are smoothed which results in missing the location provided by the ground truth as is clear in the junction of the middle image and the triangle of the right image. The combined method performs better than the FBHB and other deep learning methods and shows better crack localisation and better recovery of crack segments. The hybrid nature of the combined method is quite useful to improve the crack localisation since it searches for precise location on crack contour.

A precision-recall analysis is made and shown in Fig. 14. The combined method outperforms the FPHB method for the three tested datasets. The graph for FPHB is drawn by varying the threshold value. The only parameter that can be varied in the combined method is the block size; thus the combined method is represented as a single point in the

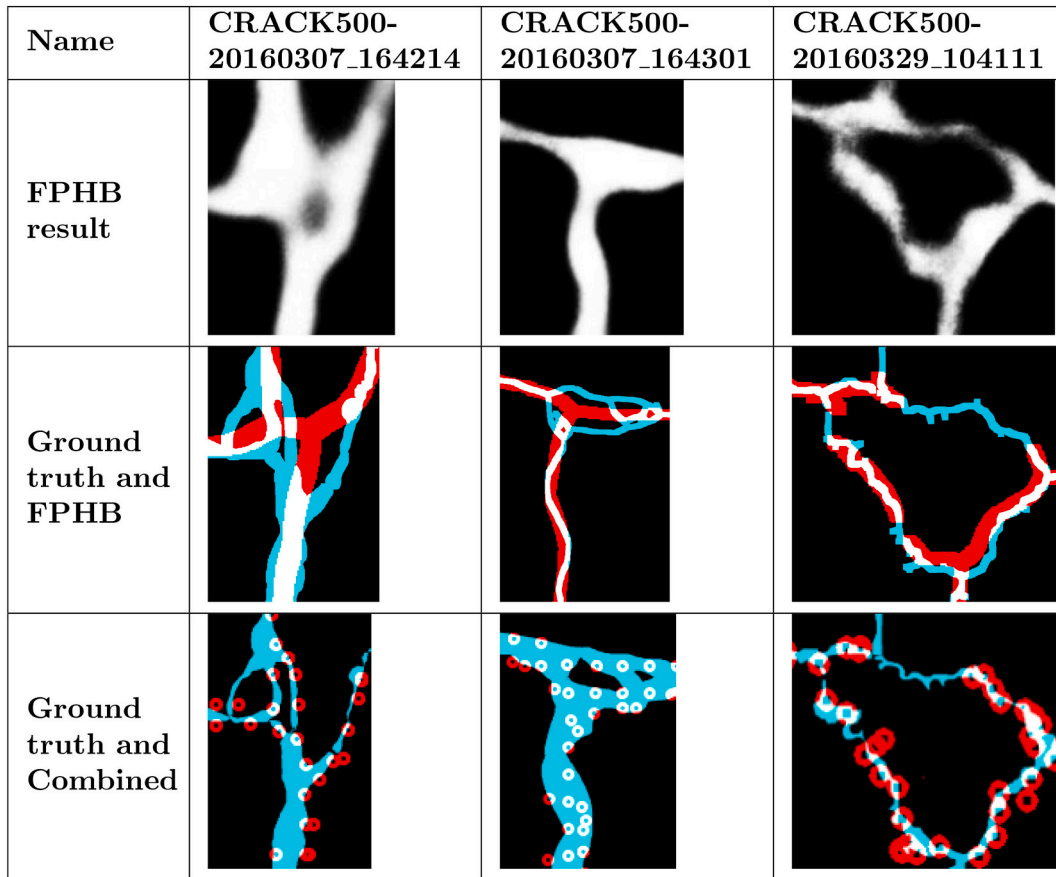


Fig. 13. Crack extraction for sample images from CRACK500 dataset showing the FPHB results together with ground truth. Red colour: FPHB result, White colour: FPHB + Ground truth, Cyan colour: Ground truth only. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

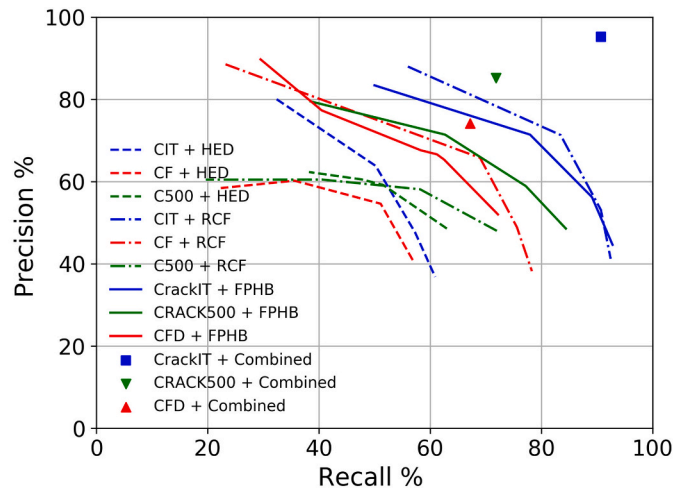


Fig. 14. The precision-recall comparison between the combined method and the FPHB deep learning method for three datasets. For the combined method, the average of the Pr-Re is shown as a single point representing the average value across each dataset.

Table 4
Summary of performance metrics for three tested datasets.

Dataset	HED	RCF	FPHB	Combined
CrackIT	Re = 0.502	Re = 0.8345	Re = 0.7796	Re = 0.907 Pr
	Pr = 0.638	Pr = 0.7131	Pr = 0.714 F1	= 0.952 F1 =
	F1 = 0.539	F1 = 0.7635	= 0.7408	0.9274 mIoU =
	mIoU =	mIoU = 0.623	mIoU =	0.915 Acc =
	0.3861 Acc	Acc = 0.9898	0.5952 Acc =	0.995 ODS =
	= 0.9846	ODS = 0.769	0.9893 ODS	0.9292 OIS =
ODS = 0.562	OIS = 0.7635	= 0.7455 OIS	0.9274	
0.5424		= 0.7418		
CRACK500	Re = 0.515	Re = 0.582	Re = 0.7719	Re = 0.718 Pr
	Pr = 0.592	Pr = 0.581 F1	Pr = 0.589 F1	= 0.851 F1 =
	F1 = 0.517	= 0.556	= 0.657	0.756 mIoU =
	mIoU =	mIoU =	mIoU = 0.497	0.656 Acc =
	0.3708 Acc	0.3999 Acc =	Acc = 0.969	0.976 ODS =
	= 0.9654	0.9645 ODS	ODS =	0.7794 OIS =
ODS =	= 0.5823 OIS	0.6683 OIS =	0.7564	
0.5509 OIS	= 0.6131	0.661		
0.5538				
CFD	Re = 0.5687	Re = 0.6897	Re = 0.625	Re = 0.672 Pr
	Pr = 0.4089	Pr = 0.6577	Pr = 0.6534	= 0.741 F1 =
	F1 = 0.432	F1 = 0.6522	F1 = 0.6256	0.688 mIoU =
	mIoU =	mIoU =	mIoU =	0.596 Acc =
	0.2976 Acc	0.5081 Acc =	0.4817 Acc =	0.969 ODS =
	= 0.95 ODS	0.9756 ODS	0.9756 ODS	0.704 OIS =
= 0.5286	= 0.6733 OIS	= 0.6391 OIS	0.688	
OIS = 0.50	= 0.6542	= 0.6393		

graph which represents the average F1 across each of the datasets. The figure shows that RCF outperforms FPHB for two datasets but FPHB outperforms RCF for the CRACK500 dataset only. This is attributed to the fact that FPHB is trained on the CRACK500 dataset.

The statistical data is also summarized in Table 4 for each dataset. There is a significant increase in all tested metrics of the F1 score, the mIoU and the accuracy for the combined method. It can be observed that the improvement by the combined method is higher for the CrackIT images and is less for the CFD. This implies that the combined method works better for high quality images.

The appearance of stains or foreign objects has been investigated from a subset of images provided in the CRACK500 and CFD datasets. Sample cases for images with asphalt stains are shown in Fig. 15. The left image show asphalt stains which did not affect the result for either FPHB or combined methods, but it affects the results of both HED and RCF. The right image shows a large stain which appeared in the HED, RCF and FPHB results but it did not affect the result of the combined method which, in this particular case correctly, showed no cracks in the image. The effects on HED and RCF is attributed to the fact that both methods are searching for edges. The effect on FPHB depends on the shape of the stain.

The effect of lighting or illumination changes is shown in Fig. 16. The three images contain shadows and non uniform illumination which did not affect the results of either FPHB or the combined method. Shadows do not affect the results because the block-based detection removes isolated blocks. The combined method also tolerates the appearance of road lanes, since CrackIT has a separate function to identify road lane lines [18].

5. Discussion

CrackIT belongs to the category of model-based crack detection

methods and it is shown here that it can be exploited to obtain good crack detection and digitization. The proposed combined method inherits useful traits from both the pixel-based and block-based modes of operation in terms of noise resistance and accurate localisation. The combined method aims to get a digitized/discretized crack representation, which is different from the conventional analogue representation. The combined method is more suitable, especially when the output is required in a discretised form for further action such as driving a robot to seal or spray on the cracks, as shown by a crack sealing robot [48]. The main limitation of the combined method is when cracks cannot be detected especially at the pixel based level, a problem that is less prominent in the deep learning methods in general. This may happen in particular for some low quality images, such as those in the CFD dataset that required image enhancement. Therefore, the method performs better when the images are of good quality with distributed histograms as expected. It is also preferable in terms of performance to detect cracks with a uniform spatial scale and to set the camera lens parallel to the road surface. Regarding sensitivity to foreign object appearance, further investigations are still needed to fully analyse the performance of the algorithm.

In a statistical sense, the combined method outperformed FPHB and two other tested deep learning methods. The deep learning results suffered from the appearance of false positives and missing crack segments. In addition, the extracted cracks give relatively poor localisation for crack bends and junctions and thus it would be challenging to use for further processes for crack repair such as robotic crack sealing.

The issue for block size selection was explored and found to be critical for good performance. The size selection should be based on the required spatial resolution in each application, and can be modified through control of the imaging optics.

6. Conclusions

A novel approach that combines the two modes of CrackIT algorithm was proposed for reducing noise in crack detection and improving crack localisation. The block-based mode generates noise-tolerant crack regions and in which then, the nearest point is found from the pixel-based mode result for providing accurate digital crack coordinates (localisation). The block mode reduces noise through aggregation of pixels. The new combined approach enjoys both benefits of the noise robustness inherited from the block-based approach with accuracy inherited from the search for nearest points from the pixel-based approach on the crack contour. The method was tested on three datasets namely the CrackIT, the CRACK500 and the Crack Forest dataset (CFD). The results were compared with the pixel-based mode as well as three deep learning methods, HED, RCF and FPHB. The new method outperformed the existing methods for all the datasets and delivered better results especially for accurate localisation of cracks. The new method has a strong potential for civil engineering applications requiring accurate digital discretization of cracks. Complete set of results are available at <https://doi.org/10.5518/937>

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.



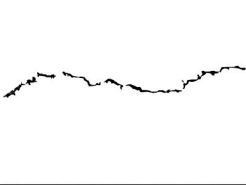

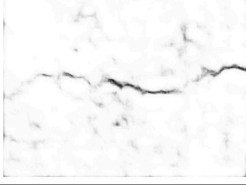

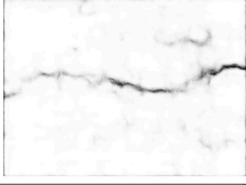
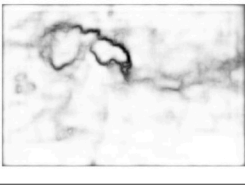


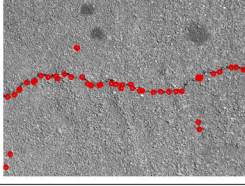

	CRACK500-IMG-2910	CFD-321
Original		
Ground truth		
HED		
RCF		
FPHB		
Combined		

Fig. 15. Sample results for pavement stains.

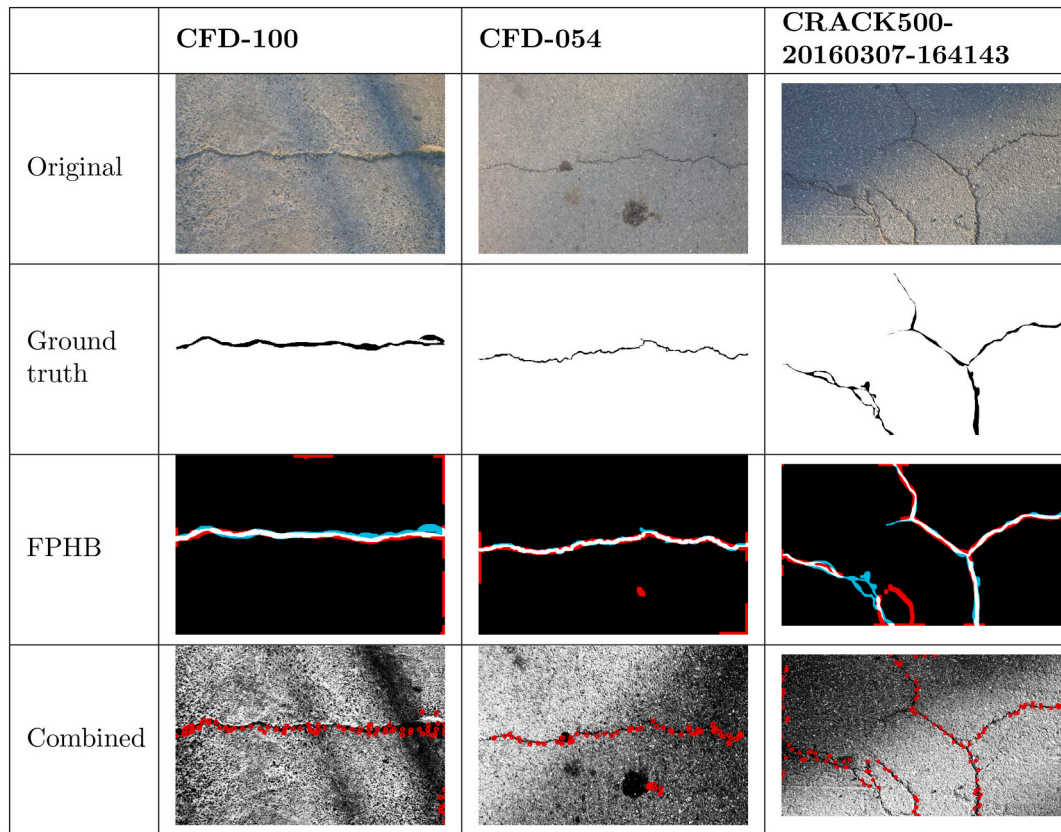


Fig. 16. Sample results for pavement with varying illumination.

Acknowledgements

This work was supported by the UK Engineering and Physical Sciences Research Council (EPSRC), award reference: EP/NO10523/1. We thank Dr. Bilal Kaddouh for fruitful discussion about the crack detection requirements for 3D asphalt printer. The third author is partially supported by an Alan Turing Institute fellowship which is gratefully acknowledged.

References

- [1] W. Wang, M. Wang, H. Li, H. Zhao, K. Wang, C. He, J. Wang, S. Zheng, J. Chen, Pavement crack image acquisition methods and crack extraction algorithms: a review, *J. Traffic Transportation Eng. (English Edition)* 6 (6) (2019) 535–556, <https://doi.org/10.1016/j.jtte.2019.10.001>.
- [2] G. Li, X. Zhao, K. Du, F. Ru, Y. Zhang, Recognition and evaluation of bridge cracks with modified active contour model and greedy search-based support vector machine, *Autom. Constr.* 78 (2017) 51–61, <https://doi.org/10.1016/j.autcon.2017.01.019>.
- [3] E. Menendez, J. Victores, R. Montero, S. Marta nez, C. Balaguer, Tunnel structural inspection and assessment using an autonomous robotic system, *Autom. Constr.* 87 (2018) 117–126, <https://doi.org/10.1016/j.autcon.2017.12.001>.
- [4] H. Nhat-Duc, Q.-L. Nguyen, V.-D. Tran, Automatic recognition of asphalt pavement cracks using metaheuristic optimized edge detection algorithms and convolution neural network, *Autom. Constr.* 94 (2018) 203–213, <https://doi.org/10.1016/j.autcon.2018.07.008>.
- [5] D. Zhang, Q. Zou, H. Lin, X. Xu, L. He, R. Gui, Q. Li, Automatic pavement defect detection using 3D laser profiling technology, *Autom. Constr.* 96 (2018) 350–365, <https://doi.org/10.1016/j.autcon.2018.09.019>.
- [6] G. Morgenthal, N. Hallermann, J. Kersten, J. Taraben, P. Debus, M. Helmrich, V. Rodehorst, Framework for automated UAS-based structural condition assessment of bridges, *Autom. Constr.* 97 (2019) 77–95, <https://doi.org/10.1016/j.autcon.2018.10.006>.
- [7] F. Kucuksubasi, A.G. Sorguc, Transfer learning-based crack detection by autonomous UAVs, in: *ISARC 2018 - 35th International Symposium on Automation and Robotics in Construction and International*, 2018, pp. 593–600, <https://doi.org/10.22260/ISARC2018/0081>.
- [8] A. Ji, X. Xue, Y. Wang, X. Luo, W. Xue, An integrated approach to automatic pixel-level crack detection and quantification of asphalt pavement, *Autom. Constr.* 114 (20176). doi : <https://doi.org/10.1016/j.autcon.2020.103176>.
- [9] S. Zhou, W. Song, Deep learning-based roadway crack classification using laser-scanned range images: A comparative study on hyperparameter selection, *Autom. Constr.* 114 (20171). doi : <https://doi.org/10.1016/j.autcon.2020.103171>.
- [10] M. Abdellatif, H. Peel, A. Cohn, R. Fuentes, Pavement crack detection from hyperspectral images using a novel asphalt crack index, *Remote Sens.* 12 (18) (2020) 3084, <https://doi.org/10.3390/rs12183084>.
- [11] M. Monti, Large-area laser scanner with holographic detector optics for real-time recognition of cracks in road surfaces, *Opt. Eng.* 34 (7) (1995) 2017–2023, <https://doi.org/10.1117/12.204793>.
- [12] J. Laurent, M. Talbot, M. Doucet, Road surface inspection using laser scanners adapted for the high precision 3D measurements of large flat surfaces, in: *International Conference on Recent Advances in 3D Digital Imaging and Modeling*, 1997, pp. 303–310, <https://doi.org/10.1109/IM.1997.603880>.
- [13] A. Miraliakbari, M. Hahn, H.-G. Maas, Development of a multi-sensor system for road condition mapping, *Int. Archiv. Photogrammetry, Remote Sens. Spatial Inform. Sci. - ISPRS Archiv.* 40 (1) (2014) 265–272, <https://doi.org/10.5194/isprsarchives-XL-1-265-2014>.
- [14] K.M. Doycheva, C. Koch, Computer vision and deep learning for real-time pavement distress detection, in: I. Mutis, T. Hartmann (Eds.), *Advances in Informatics and Computing in Civil and Construction Engineering*, Springer, 2019, pp. 601–607, https://doi.org/10.1007/978-3-030-00220-6_72.
- [15] Y. Tsai, V. Kaul, A. Yezzi, Automating the crack map detection process for machine operated crack sealer, *Autom. Constr.* 31 (2013) 10–18, <https://doi.org/10.1016/j.autcon.2012.11.033>.
- [16] R. Davoudi, G. Miller, J. Kutz, Data-driven vision-based inspection for reinforced concrete beams and slabs: quantitative damage and load estimation, *Autom. Constr.* 96 (2018) 292–309, <https://doi.org/10.1016/j.autcon.2018.09.024>.
- [17] S. Jin, S. Lee, J.-W. Hong, A Vision-Based Approach for Autonomous Crack Width Measurement with Flexible Kernel, *Automation in Construction* 110, 2019, <https://doi.org/10.1016/j.autcon.2019.103019>.
- [18] H. Oliveira, P.L. Correia, CrackIT—an image processing toolbox for crack detection and characterization, *IEEE Int. Conf. Image Process, ICIP 2014 (2014)* 798–802, <https://doi.org/10.1109/ICIP.2014.7025160>.
- [19] D. Qi, Y. Liu, X. Wu, Z. Zhang, An algorithm to detect the crack in the tunnel based on the image processing, in: *10th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Kitakyushu, Japan*, 27–29 Aug. 2014, pp. 860–863, <https://doi.org/10.1109/IHH-MSP.2014.217>.

- [20] A. Ouyang, C. Luo, C. Zhou, Surface distresses detection of pavement based on digital image processing, in: *Computer and Computing Technologies in Agriculture IV*, Springer, 2011, pp. 368–375, https://doi.org/10.1007/978-3-642-18369-0_42.
- [21] E. Aldea, S. Le Hegarat-Masclé, Robust crack detection for unmanned aerial vehicles inspection in an a-contrario decision framework, *J. Electronic Imaging* 24 (6). doi : <https://doi.org/10.1117/1.JEI.24.6.061119>.
- [22] A. Cubero-Fernandez, F. Rodriguez-Lozano, R. Villatoro, J. Olivares, J. Palomares, Efficient pavement crack detection and classification, *EURASIP J. Image and Video Processing* 39 (1) (2017), <https://doi.org/10.1186/s13640-017-0187-0>.
- [23] D. Ai, G. Jiang, L. Siew Kei, C. Li, Automatic pixel-level pavement crack detection using information of multi-scale neighborhoods, *IEEE Access* 6 (2018) 24452–24463, <https://doi.org/10.1109/ACCESS.2018.2829347>.
- [24] Z. Chen, J. Zhu, T. Hutchinson, Image-based framework for concrete surface crack monitoring and quantification, *Adv. Civil Eng.* doi : <https://doi.org/10.1155/2010/215295>.
- [25] Y.-S. Yang, C.-L. Wu, T. Hsu, H.-C. Yang, H.-J. Lu, C.-C. Chang, Image analysis method for crack distribution and width estimation for reinforced concrete structures, *Autom. Constr.* 91 (2018) 120–132, <https://doi.org/10.1016/j.autcon.2018.03.012>.
- [26] W. Zhang, Z. Zhang, D. Qi, Y. Liu, Automatic crack detection and classification method for subway tunnel safety monitoring, *Sensors* 14 (10) (2014) 19307–19328, <https://doi.org/10.3390/s141019307>.
- [27] R. Amhaz, S. Chambon, J. Idier, V. Baltazart, Automatic crack detection on two-dimensional pavement images: an algorithm based on minimal path selection, *IEEE Trans. Intell. Transp. Syst.* 17 (10) (2016) 2718–2729, <https://doi.org/10.1109/TITS.2015.2477675>.
- [28] H. Chen, H. Lin, M. Yao, Improving the efficiency of encoder-decoder architecture for pixel-level crack detection, *IEEE Access* 7 (2019) 186657–186670, <https://doi.org/10.1109/ACCESS.2019.2961375>.
- [29] Q. Mei, M. Gl, M. Azim, Densely connected deep neural network considering connectivity of pixels for automatic crack detection, in: *Automation in Construction* 110, 2019, <https://doi.org/10.1016/j.autcon.2019.103018>.
- [30] L. Pauly, H. Peel, L. Shan, D. Hogg, R. Fuentes, Deeper networks for pavement crack detection, in: *34th International Symposium on Automation and Robotics in Construction*, 2017, pp. 479–485, <https://doi.org/10.22260/ISARC2017/0066>.
- [31] Q. Zou, Z. Zhang, Q. Li, X. Qi, Q. Wang, S. Wang, Deepcrack: learning hierarchical convolutional features for crack detection, *IEEE Trans. Image Process.* 28 (3) (2019) 1498–1512, <https://doi.org/10.1109/TIP.2018.2878966>.
- [32] M.M.M. Islam, J.-M. Kim, Vision-based autonomous crack detection of concrete structures using a fully convolutional encoderdecoder network, *Sensors* 19 (19) (2019) 4251, <https://doi.org/10.3390/s19194251>.
- [33] Z. Liu, Y. Cao, Y. Wang, W. Wang, Computer vision-based concrete crack detection using U-net fully convolutional networks, *Autom. Constr.* 104 (2019) 129–139, <https://doi.org/10.1016/j.autcon.2019.04.005>.
- [34] R. Di Pace, W. Song, G. Jia, H. Zhu, D. Jia, L. Gao, Automated pavement crack damage detection using deep multiscale convolutional features, *J. Adv. Transp.* (2020), <https://doi.org/10.1155/2020/6412562>.
- [35] C. Feng, H. Zhang, H. Wang, S. Wang, Y. Li., Automatic pixel-level crack detection on dam surface using deep convolutional network., *Sensors (Basel)* 20 (7). doi : <https://doi.org/10.3390/s20072069>.
- [36] X. Yin, Y. Chen, A. Bouferguene, H. Zaman, M. Al-Hussein, L. Kurach, A deep learning-based framework for an automated defect detection system for sewer pipes, *Autom. Constr.* 109 (2019), <https://doi.org/10.1016/j.autcon.2019.102967>.
- [37] L. Zhang, F. Yang, Y.D. Zhang, Y.J. Zhu, Road crack detection using deep convolutional neural network, in: *2016 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2016, pp. 3708–3712, <https://doi.org/10.1109/ICIP.2016.7533052>.
- [38] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, H. Ling, Feature pyramid and hierarchical boosting network for pavement crack detection, *IEEE Trans. Intell. Transp. Syst.* 21 (2019) 1525–1535, <https://doi.org/10.1109/TITS.2019.2910595>.
- [39] Z. Fan, C. Li, Y. Chen, J. Wei, G. Loprencipe, X. Chen, P. Di Mascio, Automatic crack detection on road pavements using encoder-decoder architecture, *Materials (Basel)* 13 (13) (2020 Jul) 2960, <https://doi.org/10.3390/ma13132960>.
- [40] H. Majidifard, Y. Adu-Gyamfi, W.G. Buttler, Deep machine learning approach to develop a new asphalt pavement condition index, *Constr. Build. Mater.* 247 (2020) 118513, <https://doi.org/10.1016/j.conbuildmat.2020.118513>.
- [41] G. Furness, S. Barnes, A. Wright, Crack detection on local roads, phase 2, traffic management division, department for transport , Tech. Rep. (accessed 4th December 2019). URL <https://trl.co.uk/sites/default/files/PPR084.pdf>.
- [42] R. Fabbri, L. Costa, J. Torelli, O. Bruno, 2D Euclidean distance transform algorithms: a comparative survey, *ACM Comput. Surv.* 40 (1). doi : <https://doi.org/10.1145/1322432.1322434>.
- [43] Y. Shi, L. Cui, Z. Qi, F. Meng, Z. Chen, Automatic road crack detection using random structured forests, *IEEE Trans. Intell. Transp. Syst.* 17 (12) (2016) 3434–3445, <https://doi.org/10.1109/TITS.2016.2552248>.
- [44] S. Xie, Z. Tu, Holistically-nested edge detection, *Int. J. Comput. Vis.* 125 (1–3) (2017) 3–18, <https://doi.org/10.1007/s11263-017-1004-z>.
- [45] Y. Liu, M.-M. Cheng, X. Hu, J.-W. Bian, L. Zhang, X. Bai, J. Tang, Richer convolutional features for edge detection, *IEEE Trans. Pattern Anal. Mach. Intell.* 41 (8) (2019) 1939–1946, <https://doi.org/10.1109/tpami.2018.2878849>.
- [46] M. Sudarma, N.P. Sutramiani, The thinning Zhang-Suen application method in the image of Balinese scripts on the papyrus, *Int. J. Comput. Appl.* 91 (1) (2014) 9–13, <https://doi.org/10.5120/15844-4726>.
- [47] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: *Proceedings of the 8th International Conference on Computer Vision Vol. 2*, 2001, pp. 416–423. https://www.bibsonomy.org/bibtex/20e6d2db0c6ea14b03241bbfb727bad01/alex_ruff/ (Accessed 24th Sept. 2020.).
- [48] V. Doychinov, M. Abdellatif, B. Kaddouh, B. Malik, G. Jackson-Mills, R. Fuentes, A. Cohn, R. Richardson, N. Chudpooti, I. Robertson, N. Somjit, Infrastructure robotics research at the University of Leeds, 2019. <http://eprints.whiterose.ac.uk/152078/>. Accessed 24th Sept. 2020.