Automation in Construction 143 (2022) 104572

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Fast identification of concrete cracks using 1D deep learning and explainable artificial intelligence-based analysis

Ganesh Kolappan Geetha, Sung-Han Sim *

School of Civil, Architectural Engineering and Landscape Architecture, Sungkyunkwan University, Suwon 16419, Republic of Korea

ARTICLE INFO

Keywords:
1D CNN
Deep learning
Fast crack and non-crack classification
Computer vision
Concrete structures
Adaptive threshold image binarization
Image processing
eXplainable Artificial Intelligence (XAI)
Mobile AI

ABSTRACT

The present paper discusses a computationally efficient Deep Learning (DL) model for real-time classification of concrete crack/non-crack and investigates the 'black-box' nature of the proposed DL model using eXplainable Artificial Intelligence (XAI). The state-of-the-art DL models like semantic segmentation require labor-intensive labeling for pixel-level classification. The proposed framework combines image binarization and a Fourier-based 1D DL model for fast detection and classification of concrete crack/non-crack features. Image binarization as a precursor to DL extracts possible Crack Candidate Regions (CCR) and eliminates the plane structural background during DL training and testing. Metadata within the 1D DL model was generated and analyzed using local XAI, wherein t-distributed Stochastic Neighborhood Embedding (t-SNE) was used to visualize the knowledge transfer within the hidden layers. The proposed model enables real-time pixel-level classification of crack/non-crack at the rate of 2 images/s on a mobile platform with limited computational facilities.

1. Introduction

Conventionally, concrete structures are inspected visually to identify the locations of potential defects and to classify the defects based on their severity. Common defects in concrete structures include cracks, spalling, blistering, delamination, pitting, and strain. Among the previously mentioned defects, concrete cracks are directly related to the integrity of the load-carrying components at the system level and are hence critical from a structural health monitoring (SHM) perspective. The conventional visual inspection approach is limited by the inspector's judgment skills, which are subjective and depend on their training and experience [1]. Although physics-based non-destructive inspection techniques like low-frequency ultrasonics in pitch-catch or pulse-echo mode [2–4], thermography [5–8] show promising results in detecting hidden or sub-surface defects at the lab scale, requirements of customized expensive auxiliary devices for measurement and scalability for rapid large area inspection in the context of field measurements for concrete structures are still questionable. With the advent of computer vision and machine learning, there has been a rapid advancement in areas related to automated concrete crack detection [9]. Koch et al. [10] provides a holistic review of current achievements, current practices, and limitations of the visual conditioning of civil structures. One of the critical challenges for a robust computer vision-based SHM of civil structures methodology is detecting and distinguishing cracks from non-crack features in real-time.

Defect detection and classification of cracks in concrete structural components play a vital role in SHM for providing needful maintenance and prolonging the life at the system level [11,12]. Spencer et al. [9] and Koch et al. [10] comprehensively discuss various image processing techniques used along with computer vision-based methods to detect concrete cracks. Although each image-processing scheme has its own advantages, it is not possible to find a universal image-processing scheme that works well under all conditions. However, the accuracy of various image processing tools in the context of field data from real-world applications remains questionable. In the case of field data, there is a large amount of noise; hence, its applicability is limited. The reported image processing schemes are context-specific and require prior knowledge of the defect or feature to be highlighted and suppressed. Although the edge-detection scheme is a possible candidate for automatic crack identification [13] crack edges are often disconnected, making full-scale automation difficult. Small, disconnected crack segments can be misinterpreted as background noise, and require careful handling. Possible solutions for real-world applications include combining image processing scheme with machine learning algorithms [14].

Hsieh and Tsai [15] report an exhaustive review of research works in

* Corresponding author.
E-mail address: ssim@skku.edu (S.-H. Sim).

https://doi.org/10.1016/j.autcon.2022.104572
Received 29 March 2022; Received in revised form 1 September 2022; Accepted 4 September 2022

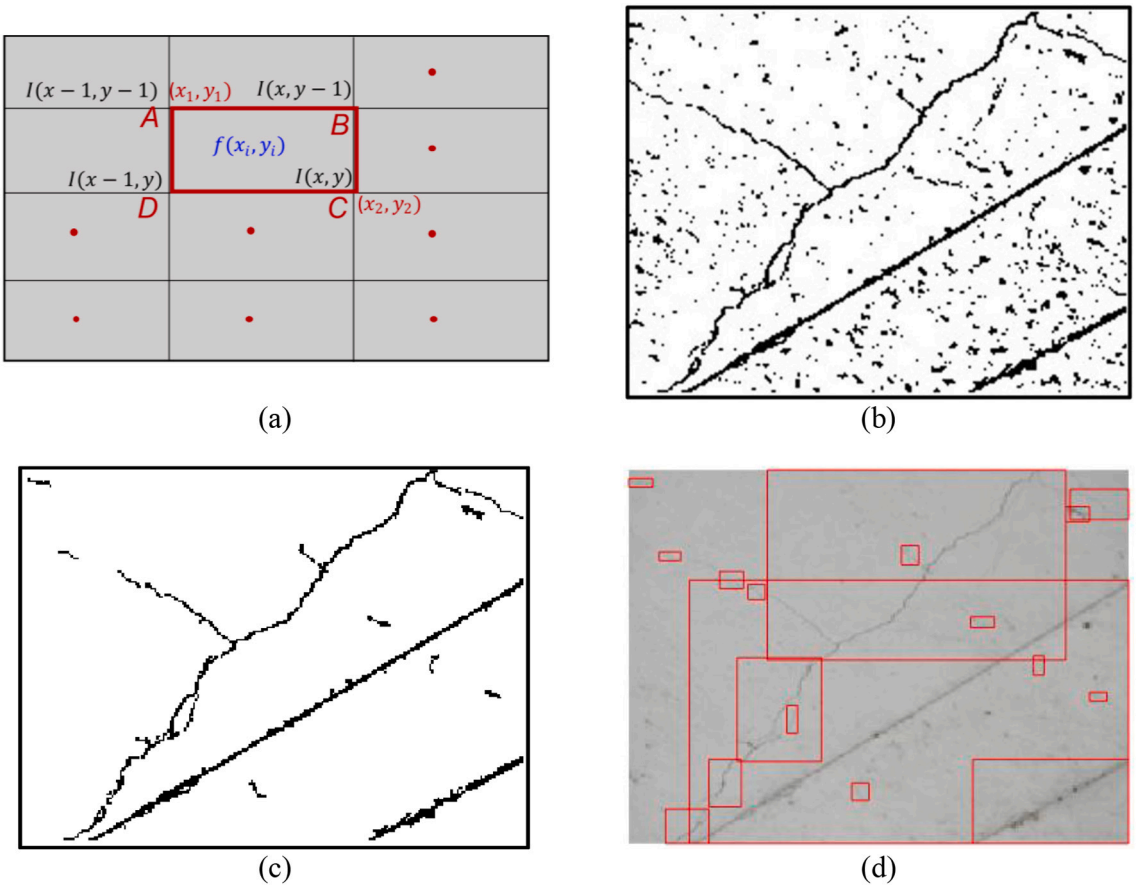

Fig. 1. (a) Schematic for integral image computation over a finite window, and (b) representative results with cracks and non-crack features using adaptive threshold-based integral image binarization. (c) Noise filtered binarized image using area and eccentricity criteria, and (d) corresponding CCR mapped with red color in raw image. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the last decade that uses conventional machine learning other than deep learning (DL) for concrete crack detection. The conventional machine-learning-based methodology requires prior extraction of discriminant features, which is subjective and context-specific. Therefore, shallow learning cannot learn deeper complex information from the images. The accuracy of these methods is highly sensitive to user-defined thresholds and chosen characteristics, which introduce uncertainty under different training and testing conditions. In recent years, extensive research efforts have been devoted to DL-based concrete crack detection and classification [16–23]. State-of-the-art machine learning, particularly supervised learning, extracts unique characteristics related to cracks and non-cracks, depending on which classification is performed [16]. The Discriminant feature extraction for DL requires no manual intervention, and the hidden cascading layers automatically extract complex hidden characteristics.

The state-of-the-art convolutional neural network (CNN) for detection and classification of cracks and non-cracks candidates is performed on a 2D pixel or image space [17], [19–26]. A holistic review of neural network-based concrete crack detection for different applications in pavement engineering are reported in [27]. Guan et al., [28] demonstrated the capability of parallel and oblique photography through a low-cost stereo vision in conjunction with DL to classify pixel-level features. The approaches discussed in previous studies based on the concepts of sliding windows, bounding boxes using faster R-CNN, and pixel-level classification using semantic segmentation are time-consuming because they deal with matrix operations in 2D image space during forward and backward propagation. Hence, they are not suitable to be performed in real time when used with a robotic system or unmanned aerial vehicle (UAV), where only limited computational facility is available in the movable setup. Recently, rapid advancement in the field of optics and semiconductors enable to obtain high-definition images from the target structure. Nevertheless, existing DL schemes cannot be directly scaled for high-definition images due to computational inefficiency. Moreover, the semantic segmentation discussed in [20–23] requires labor-intensive and time-consuming labeling, and is subjective to human expertise. The approach in reference [18] overcomes the labeling limitation by using image processing as a precursor to DL. Taking cue from [18], we employ an adaptive threshold-based imaging binarization scheme as a precursor to the proposed Fourier-based DL in our current work.

Driven by the recent success of deep learning in applications related to ultrasonic-guided waves, we investigated the “black-box” nature of deep learning using eXplainable Artificial Intelligence (XAI). According to [29], “XAI is a field of artificial intelligence (AI) that promotes a set of tools, techniques, algorithms to generate high-quality interpretable, intuitive, human-understandable explanations of AI decision”. Das et al. [29] and references therein provide a holistic view of the current state of XAI in deep learning for different applications. A large number of parameters are involved in the deep CNN, which makes it complex to understand and visualize. Explanations of metadata using suitable visualization enable the improvement of AI algorithms and human understanding [30]. In the current work, we generated a local explanation using cluster analysis to find the features learned by each layer using t-distributed Stochastic Neighborhood Embedding (t-SNE). This enables an understanding of the metadata transfer during DL.

To overcome the limitations of conventional CNNs, R-CNNs, semantic segmentation, and to take advantages of 1D-CNN in the time domain [31] and XAI, the authors propose a scheme to transform and

analyze images from a 2D pixel space to a 1D frequency space. This reduces the computational complexity associated with DL training and testing. In this study, we transform the images in pixel space into a 1D vector using Fourier transformation with a large number of independent bases. This study presents the classification of crack and non-crack structural features in real time using image binarization and 1D-DFT-CNN. First, a framework for crack candidate regions (CCR) identification using image binarization as a precursor to DL is discussed. Subsequently, the details of the proposed 1D-DFT-CNN used for crack and non-crack classification and those related to training and testing are discussed. The details of the implementation of the proposed framework on a mobile platform with limited computational facility are discussed. Subsequently, we discuss the qualitative and quantitative results obtained during XAI knowledge transfer by investigating the metadata of the proposed deep learning architecture.

2. Theory and background

In this section, we discuss the background related to (i) adaptive threshold-based integral image binarization and CCR identification, (ii) transformation of the CCR region from the pixel space to the Fourier domain and 1D vectorization, and (iii) XAI using t-SNE to visualize metadata in the hidden layers during the deep learning process.

2.1. Adaptive threshold-based integral image binarization and CCR identification

Image binarization an important pre-processing step, especially for pixelized data, uses a 'non-parametric' and an 'unsupervised optimal threshold' for feature discrimination in the binary pixel space. Image binarization first transforms an image's RGB components to grayscale, ranging from 0 (black) to 255 (white) and later to a binary pixel space using an optimal threshold. The pixels above the optimal threshold are assigned one (white) in the binary scale, while those below are zero (dark). The unsupervised threshold determination for optimal feature discrimination is classified based on local or global approach. Techniques like Otsu's method [32] and its improvisation [33–35] maximize the discriminant feature class variance to estimate the global threshold. These techniques fail for concrete crack detection because (i) the gray-level histogram is often uni-modal or close to unimodal distribution, and (ii) the lighting conditions across the entire image are non-uniform. Adaptive thresholding over a finite window of pixels overcomes the previous limitations [36–38]. Kim et al. [39] summarizes different local threshold-based concrete crack detection schemes. In the current work, we employ an adaptive threshold-based integral image binarization, one of the robust local thresholding-based approaches [40,41].

Below, we concisely summarize the mathematical scheme for adaptive threshold-based integral image binarization. Following conventional notation [40], the integral image $I(x,y)$ (also known as the summed-area table) computed at each pixel location is given by

$$I(x,y) = f(x,y) + I(x-1,y) + I(x,y-1) - I(x-1,y-1), \quad (1)$$

where $f(x,y)$ is the intensity of each pixel. Fig. 1(a) schematically shows the computed integral image of each pixel. The cumulative sum of the function over rectangle ABCD with upper left corner $A(x_1,y_1)$ and lower-left corner $D(x_2,y_2)$ is computed as:

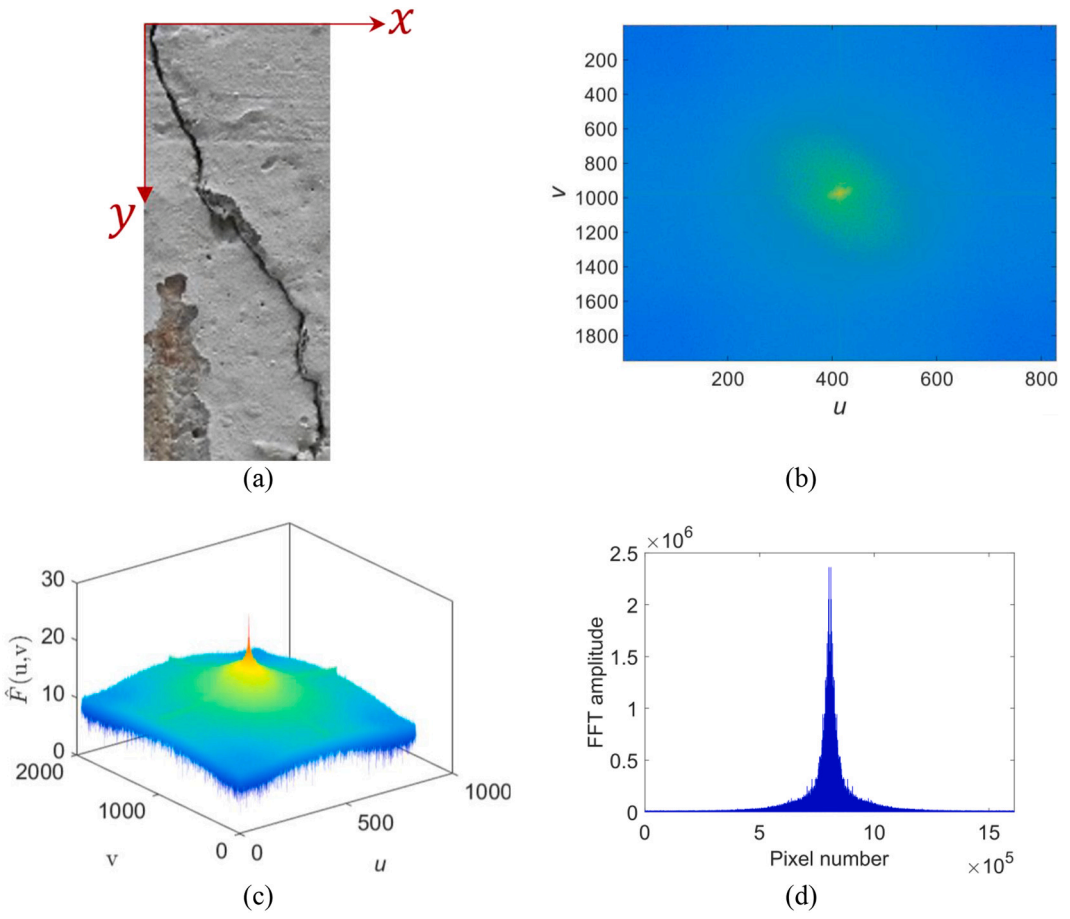

Fig. 2. (a) A representative CCR obtained after integral image binarization. Corresponding 2D and 3D frequency-shifted Discrete Fourier Transform (DFT) are shown in (b) and (c), respectively. (d) Vectorized CCR in frequency domain.

$$f_s(x,y) = \sum_{x=x_1}^{x_2}\sum_{y=y_1}^{y_2} f(x,y) = I(x_2,y_2) - I(x_2,y_1-1) - I(x_1-1,y_2) + I(x_1-1,y_1-1). \quad (2)$$

For a window of size 's × s', image binarization threshold $T(x,y)$ is:

$$T(x,y) = \begin{cases} 1 & T(x,y) > \frac{f_s(x,y)}{s\times s}\times\left(1-\frac{t}{100}\right) \\ 0 & otherwise \end{cases}, \quad (3)$$

where $T(x,y)$ is the binarized value and t is the sensitivity factor. Fig. 1 (b) shows the representative binarization results with cracks, non-cracks, and noise. Local adaptive thresholding computes different threshold values for each image pixel (x,y). The derived integral image $I(x,y)$ and subsequently, estimated image binarization threshold $T(x,y)$ are computed at each pixel location (see (Eqs. (1)–(3)). The analytical expressions described in Eqs. (1)–(3), in an approximate sense, is a moving average of the 's' surrounding pixels while sliding through the image. The local adaptive threshold, a function of sliding window size and sensitivity factor prescribed by the inspector, controls the statistical features of the finite window. The method preserves hard contrast lines and ignores soft gradient changes; hence, adaptive thresholding accounts for spatial variation in illumination. Hence, the technique accounts for spatial variation in illumination.

The binarized images require additional filtering to remove the noisy surface texture and shot noise. Shot noise originates from the discrete nature of the electric charge and occurs during photon counting in optical devices and is associated with the particle nature of light [42]. Post adaptive threshold-based integral image binarization (Fig. 1(b)), we perform a connected component analysis [43] to identify a possible continuous stretch of the pixelated residue. We approximate it to an ellipse with finite major and minor axes as the length and width of the pixelated residue. We filter pixelated residue corresponding to noisy background features using the crack's geometric properties or aspect ratio [18], including eccentricity and area-based threshold. One of the characteristics of concrete cracks is a thinner shape than other background textural patterns. A large aspect ratio of concrete crack is a potential discriminant signature to separate from the noisy background features. Moreover, from fracture mechanics, cracks are often modeled as elliptical cracks with a region of high-stress intensity, and the crack signature is directly correlated to the degree of eccentricity. The ellipse's eccentricity is the ratio of the distance from center to foci and center to the vertices. Next, we compute the eccentricity and area of the elliptical region using the major and minor axes. Finally, we apply eccentricity and area-based threshold to filter out other noisy background and texture-related features as seen on concrete surfaces. We have used an excentricity and area threshold of 0.85 and 250 pixels, respectively. These threshold numbers are derived based on heuristics. The filtered pixelated residue is shown in Fig. 1(c).

Further, we obtain the horizontal and vertical bounds of each continuous stretch of pixelated residue and construct rectangular bounding boxes with these bounds. These rectangular bounding boxes are called Crack Candidate Regions (CCRs). The CCRs from the pixelated image are mapped back to the original RGB raw image (Fig. 1(d)). The mapped rectangular CCRs from the original RGB raw image are used subsequently training and testing of DL.

2.2. Transforming CCRs to 1D vector space using Fourier basis

Conventionally, image processing for concrete crack detection is performed in the spatial domain, that is, algorithms are applied directly to the raw image. Abdek-Qader [13] demonstrated the capability of concrete crack detection in the frequency domain. Literatures [44–46] reports analysis of images in the transformed multi-scale spatial-frequency domain. Reference [47] reports the efficacy of analyzing images in the frequency domain which motivates the authors of this paper to implement DL in the frequency domain. Frequency domain-based analysis gives a distinct signature arising from abrupt changes in spatial pixel intensity at the crack location, which is a further advantage of the proposed method. The distinct crack signature enabled DL to effectively distinguish the edges and shallow background features. For a CCR region (Fig. 2(a)) of size $M \times N$, the 2D DFT of the CCR is given as:

$$\widehat{F}(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} \quad (4)$$

where $f(x,y)$ is the image in the pixel space and $e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$ is the basis function corresponding to each point $\widehat{F}(u,v)$. The Fourier transform produces complex values that include information related to the magnitude and phase. $\widehat{F}(u,v)$ has low-frequency components at the corners and a high-frequency component at the center, which is difficult to interpret. Hence, we perform frequency shifting, which transforms all low-frequency components with higher magnitudes to the center of the axis system. Fig. 2(b) and (c) shows the 2D and 3D frequency-shifted Discrete Fourier Transform (DFT) of the representative CCR shown in Fig. 2(a). Transforming the image from the 2D pixel space to 1D Fourier space reduces the dimensional complexity without loss of information. The corresponding vectorized CCR in the frequency domain was provided as an input to the 1D CNN architecture (Fig. 2(d)). The vectorized image in the frequency domain accelerates the CNN computation. Unlike 2D CNN, where the kernel slides in two dimensions of the data, the characteristic of 1D CNN architecture is that the kernel slides along one dimension; hence, reducing the number of parameters and decreasing the training time as compared to 2D CNN. The advantages of 1D CNN architecture over 2D CNN architecture are as follows: (i) The 1D CNN replaces the matrix computation involved in gradient descent-based optimization during the forward and backward propagation with simple vector array operations. (ii) 1D CNNs with relatively shallow architectures (i.e., a smaller number of hidden layers and neurons) learn hidden features than the conventional 2D CNN, requiring a deeper architecture to learn the same. (iii) Training 2D CNNs with deeper architecture requires either cloud computing or high-performance GPUs. On the contrary, low computational requirements enable the implementation of 1D CNNs with standard computer or mobile, or hand-held devices.

2.3. XAI using t-SNE

t-distributed Stochastic Neighbor Embedding (t-SNE) is a local XAI scheme that extracts metadata within hidden layers in the high-dimensional space and transforms to a low-dimensional (two-or three-dimensional) space. The t-SNE enables effective mapping of the hidden learned features at several different scales to low-dimensional space. This is particularly important when transforming high-dimensional data at several different scales into a low-dimensional space. Following the standard notation discussed in [48], below, we concisely summarize t-SNE as follows: The t-SNE converts high-dimensional Euclidean distances between data points into conditional probabilities that represent similarities. The pairwise similarity ($p_{j|i}$) of datapoints x_i and x_j in the feature space f_i is given as

$$P_{j|i} = \frac{e^{-\left\|x_j-x_i\right\|^2/2\sigma_i^2}}{\sum_{k\neq i} e^{-\left\|x_i-x_k\right\|^2/2\sigma_i^2}}, \quad (5)$$

where σ_i is the variance of the Gaussian centered on data point x_i. The low-dimensional counterparts for high-dimensional data x_i and x_j are y_i and y_j, respectively. The pairwise similarity ($q_{j|i}$) of datapoints x_i and x_j in the low-dimensional feature space is given as

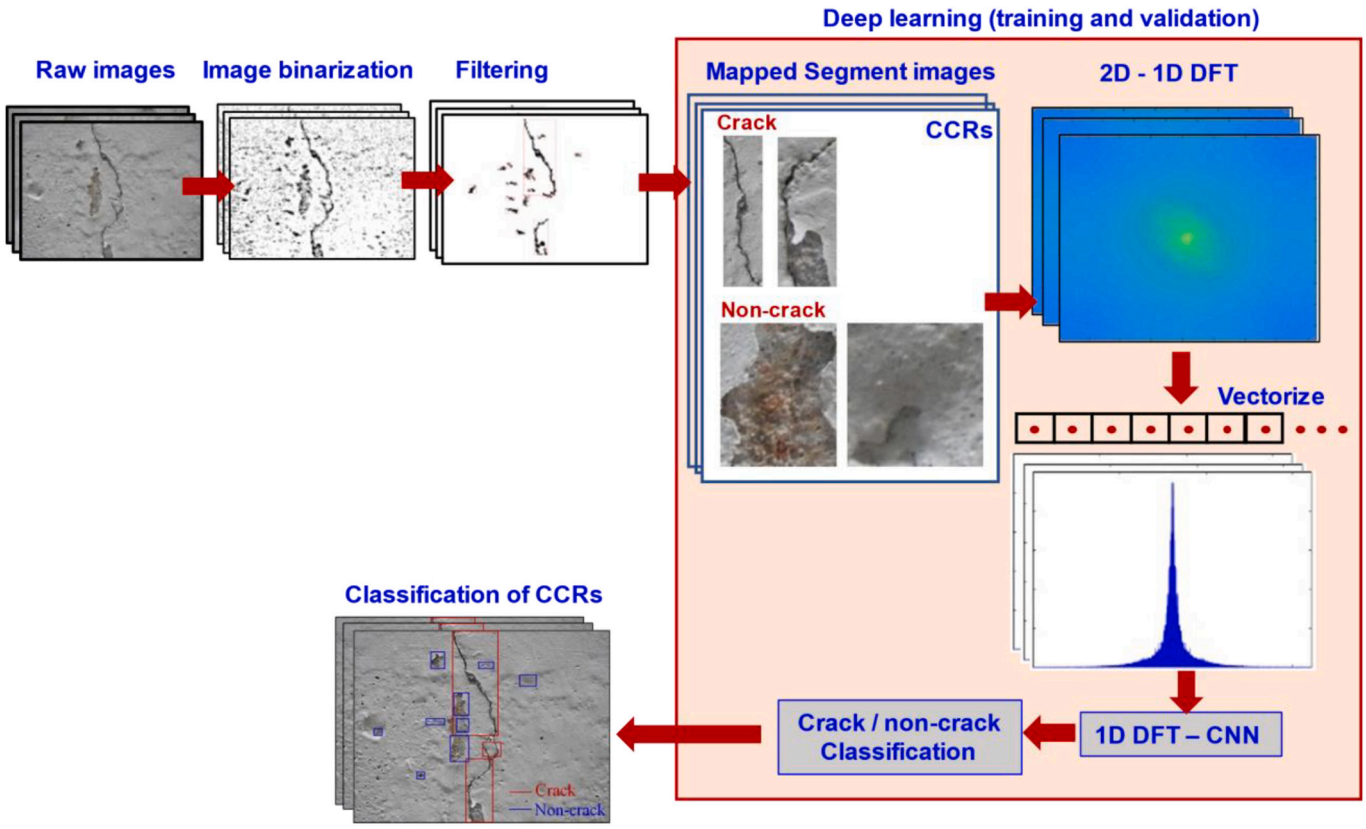

Fig. 3. Overall approach for detection and classification of crack and non-crack features by combining adaptive image binarization with DL.

$$q_{j|i} = \frac{e^{-\|y_j - y_i\|^2}}{\sum_{k \neq i} e^{-\|y_i - y_k\|^2}}; q_{i|i} = 0. \tag{6}$$

The conditional probabilities $p_{j|i}$ and $q_{j|i}$ will be equal if y_i and y_j correctly model the similarity between high-dimensional points x_i and x_j. t-SNE minimizes the mismatch between $p_{j|i}$ and $q_{j|i}$. The symmetric joint probabilities q_{ij} is obtained by expanding Eq. (6) to a first-order approximation using the Taylor series:

$$q_{ij} = \frac{\left(1 + \|y_j - y_i\|^2\right)^{-1}}{\left(\sum_{k \neq l} 1 + \|y_j - y_i\|^2\right)^{-1}}. \tag{7}$$

The previous equation has the advantage that $(1 + \|y_j - y_i\|^2)^{-1}$ approaches an inverse square law for large pairwise distances, $\|y_j - y_i\|$. This ensures that the joint probabilities are almost invariant to changes in map scale. A natural measure of faithfulness with $p_{j|i}$ and $q_{j|i}$, is Kullback-Leibler divergence. t-SNE minimizes the sum of the Kullback-Leibler divergence over all data points by using gradient descent. Assuming symmetricity, the cost function is given as:

$$C = \sum_i KL(P_i \| Q_i) = \sum_i \sum_j P_{ij} log \frac{P_{ij}}{q_{ij}}, \tag{8}$$

where P_i represents the conditional probability distribution over all data points x_i, and Q_i represents the conditional probability distribution over mapping point y_i. Minimization of the cost function in Eq. (8) is performed using the gradient descent method as follows:

$$\frac{\partial C}{\partial y_i} = 4 \sum_j (P_{ij} - q_{ij}) \left(1 + \|y_i - y_j\|^2\right)^{-1} (y_i - y_j). \tag{9}$$

Mapping features in the low-dimensional space using t-SNE enables visualization of the metadata feature space during knowledge transfer across the layers of the deep neural network. t-SNE is analogous to eigenmap analysis to find the relevant clusters, which essentially gives the eigengap between successive clusters.

3. Crack and non-crack classification using 1D DFT-CNN and XAI meta-analysis

The proposed framework combines the advantages of adaptive threshold-based image binarization and a 1D CNN model for computationally fast and efficient pixel-level classification of cracks and non-crack features. The CCRs obtained before DL significantly reduce the plane background features, which occupy a large image area during DL training and testing. Also, the present model replaces the 2D forward and backward propagation-based matrix computation with vector operations during DL. In addition, we investigated the hidden knowledge transfer between the convolutional layers and visualized the efficacy of the proposed 1D deep-learning model using XAI metadata analysis. The remainder of this section explains the proposed method, as shown in Fig. 3.

The first step in the proposed approach is to generate possible CCRs using adaptive-threshold-based integral image binarization. Adaptive thresholding considers spatial variation in illumination. The parameters influencing the binarization results are (i) the window size and (ii) the sensitivity factor. In the present work, window size '*s*' was chosen as $\frac{1}{8}^{th}$ the original size of the image and the sensitivity factor as $t = 45\%$ (see Eq. 3). In the current study, the size of the images was 4608 × 3456 pixels. A large portion of noisy CCRs can be removed by filtering using the eccentricity and number of pixels criteria. The CCRs obtained in the pixel space are transformed into Fourier space and later vectorized. Transforming the image from the pixel space to Fourier space and vectorizing the independent 2D Fourier components to 1D reduces the dimensional complexity. The Fourier transform produces complex values that include information related to the magnitude and the phase

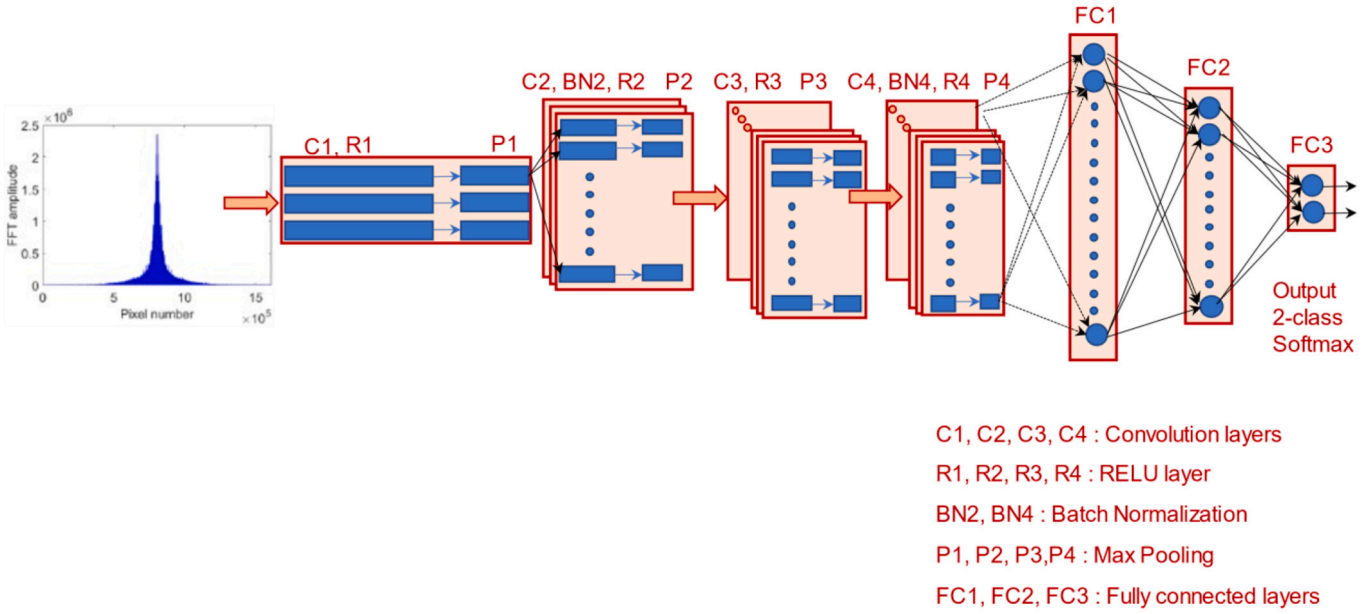

Fig. 4. Dimensionally reduced 2D to 1D DFT-CNN architecture.

Table 1
Parameters involved in 1D-DFT-CNN.

Layer	Kernel Shape	No: of Kernels	Stride	Activations	Learnable	
					Weights	Bias
C1: Convolution 1	1 × 102	3	1	1 × 65,535 × 3	1 × 102 × 1 × 3	1 × 1 × 3
R1: ReLU 1	1 × 102	3	1	1 × 65,535 × 3	–	–
P1: Maxpool 1	1 × 102	3	2	1 × 32,717 × 3	–	–
C2: Convolution 2	1 × 24 × 3	10	1	1 × 32,694 × 10	1 × 24 × 3 × 10	1 × 1 × 10
BN2: Batch Normalization 1	1 × 24 × 3	10	1	1 × 32,694 × 10	1 × 1 × 10	1 × 1 × 10
R2: ReLU 2	1 × 24 × 3	10	1	1 × 32,694 × 10	–	–
P2: Maxpool 2	1 × 24 × 3	10	2	1 × 16,347 × 10	–	–
C3: Convolution 3	1 × 11 × 10	10	1	1 × 16,337 × 10	1 × 11 × 10 × 10	1 × 1 × 10
R3: ReLU 3	1 × 11 × 10	10	1	1 × 16,337 × 10	–	–
P3: Maxpool 3	1 × 11 × 10	10	2	1 × 8168 × 10	–	–
C4: Convolution 4	1 × 9 × 10	10	1	1 × 8160 × 10	1 × 9 × 10 × 10	1 × 1 × 10
BN4: Batch Normalization 2	1 × 9 × 10	10	1	1 × 8160 × 10	1 × 1 × 10	1 × 1 × 10
R4: ReLU 4	1 × 9 × 10	10	1	1 × 8160 × 10	–	–
P4: Maxpool 4	1 × 9 × 10	10	2	1 × 4080 × 10	–	–
FC1: Fully Connected Layer 1	30	30	–	1 × 1 × 30	30 × 40,800	30 × 1
DP1: Dropout 1 (25%)	–	–	–	1 × 1 × 30	–	–
FC2: Fully Connected Layer 2	10	10	–	1 × 1 × 10	10 × 30	10 × 1
DP2: Dropout 2 (20%)	–	–	–	1 × 1 × 10	–	–
FC3: Fully Connected Layer 3	2	2	–	1 × 1 × 2	2 × 10	2 × 1
SM: Softmax	2	2	–	–	–	–

of CCRs. The 2D DFT vectorized to the 1D DFT is fed to the 1D CNN. Finally, we perform binary classification of vectorized CCRs as crack or non-crack in the frequency domain using 1D CNN. Each image segment (CCRs) classified as crack or non-crack is mapped to the original raw image. The present approach of image binarization with DL reduces the computational burden and eliminates the plane background during processing. The proposed 1D vector array-based computation significantly reduces the computational time compared to the state-of-the-art pretrained 2D matrix-based CNNs while maintaining the performance.

Unlike conventional CNN models that operate exclusively on 2D image space, the authors used a 1D DFT-CNN as an alternative (Fig. 4), a modified version of 2D CNNs that has been developed and tested recently. An overview of 1D CNNs is provided in [31] and the references therein. The optimal configuration of 1D CNN architecture used by the authors of this work consists of four convolutional layers and three Fully Connected (FC) layers. The DL architecture is an improvisation of previously adopted architecture for 1D time signal application [31]. The metrics used to evaluate the learning algorithm's performance using the chosen architecture provided asymptotic stability of loss function and higher accuracy for the given dataset. DL architecture reported in [31] consists of three convolutional layers and two fully connected layers. An additional convolutional and fully connected layers are required for a smooth transition of large kernel shape from the first hidden layer to the last hidden layer. Vectorized decomposition of two-dimensional CCRs to 1D CCRs in the frequency domain required a larger kernel shape than [31] in the first hidden layer for computational efficiency. Smooth transition through additional layers ensures optimal learning of weight functions and bias.

Table 1 presents the detailed parameters used in the proposed architecture. Each convolutional layer has a Rectified Linear Unit (ReLU) and max pooling layer. A detailed discussion on the use of a convolutional network, ReLU, max pooling, and FC is discussed in [17]. Some auxiliary layers, such as batch normalization and dropout layers, were used to avoid overfitting [49,50]. Batch normalization was applied in

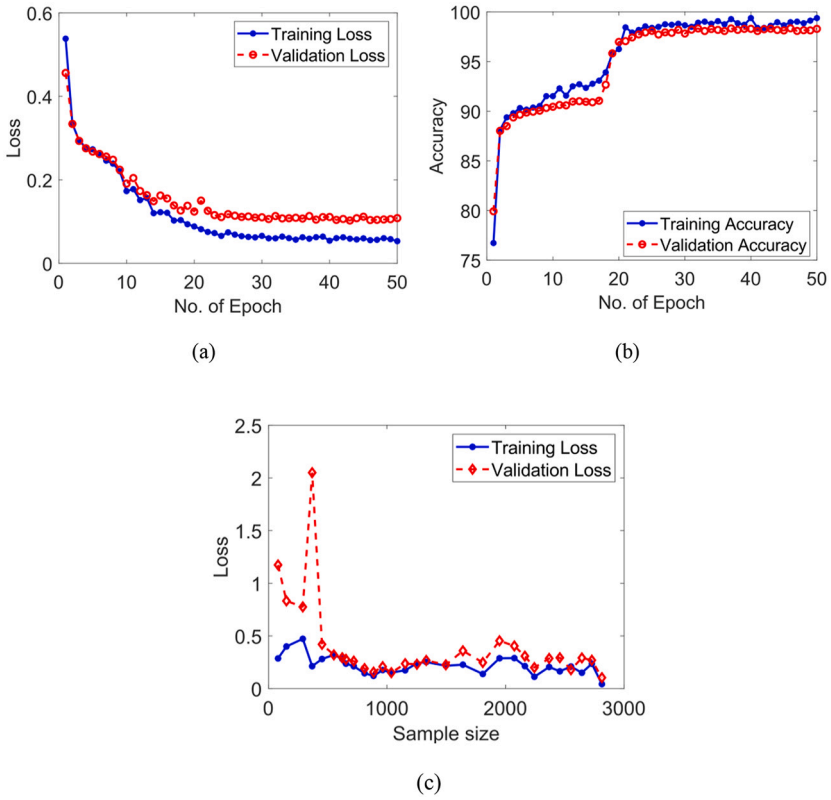

Fig. 5. Optimal parameters for training and validation of proposed DL architecture. Optimal number of epochs for training and validation dataset based on asymptotic (a) loss and (b) accuracy. (c) Loss convergence curve to identify minimal sample size for DL.

the second and fourth layers, whereas a dropout layer followed 1st and 2nd FC layers. The two-dimensional CCRs of unequal sizes are pre-conditioned to a uniform size before vector decomposition. Any segment size greater than or less than 256 × 256 is resized or padded with zeros. We use a "sweet spot" batch size of 128 samples to achieve an optimal performance during the training and testing [51].

Metadata analysis of DL models provides meaningful insights on the input features, patterns learned by DNN, and output correlation, thereby promoting the transparency of the model. Different global and local models are available in the literature to explain the XAI perspective. We investigated the efficacy of the proposed shallow 1D DFT-CNN architecture using a local XAI. Local XAI distinctly clusters the multidimensional crack and non-crack features learned in each layer in a 2D DL basis space. Here, we visualize multidimensional hidden knowledge or deep-learned features between layers in a 2D space using t-distributed stochastic neighborhood embedding (t-SNE) (Section 2.3). The segmented visualization of deep-learned features provides the qualitative efficacy of the proposed shallow 1D architecture.

4. Results and discussion

4.1. Crack and non-crack classification using the proposed architecture

The designed 1D DFT-CNN was trained on 1492 crack and 1321 non-crack images. The sizes of the cropped images obtained after adaptive integral threshold-based integral image binarization are different; hence, zero paddings are performed considering the maximum size of the image present in the database before feeding into the training network. Sample concrete crack and non-crack images used for training and testing are shown in Fig. A.1 and A.2, respectively. A 70/30 ratio of the database was used for training and testing. CCRs that include both crack and non-crack features in a single segmented image are removed from the training database. In addition, CCRs with crack-like features

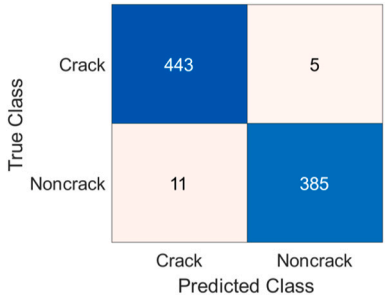

Fig. 6. Confusion matrix for crack and non-crack classification.

owing to paint peeling were not included in the training. The optimal number of epochs for training the 1D-DFT-CNN was determined as 50 epochs based on the asymptotic behavior of the binary entropy loss (Fig. 5(a)) and accuracy (Fig. 5(b)) on the training and validation datasets. The minimal dataset for the 1D-DFT-CNN training was selected based on the asymptotic behavior of the training and validation loss (see Fig. 5(c)). The binary entropy loss is given by

$$L_{BCE}(y,\hat{y}) = y log\hat{y} + (1-y) log(1-\hat{y}), \tag{10}$$

where y and $\hat{y}$ are the output and the predicted output, respectively.

The performance of the binary classification for crack and non-crack CCR obtained using the proposed methodology is shown in the form of a confusion chart (see Fig. 6). An F_1 score of 0.9823 was obtained for the testing dataset. F_1 score is given as

$$F_1 \text{score} = \frac{2 \text{ x Precision x Recall}}{\text{Precision} + \text{Recall}}, \tag{11}$$

where Precision $= \frac{TP}{TP+FP}$ and Recall $= \frac{TP}{TP+FN}$. TP, FP, FN, and TN denote

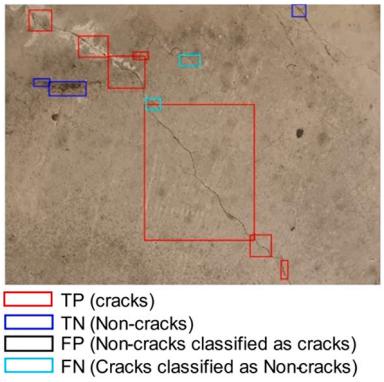

Fig. 7. Classification results on a random test image.

the numbers of True Positives, False Positives, False Negatives, and True Negatives, respectively. The compact 1D DFT-CNN with fewer neurons gives an F_1 score of 0.9823 and an accuracy of 98.10%, a similar performance compared with existing published works discussed previously. We could infer that 1D Fourier-based CNN with shallow architecture and fewer neurons can effectively outperform existing 2D CNNs. The developed scheme was tested using random images that were not included in the training process. Sample results for classification on different test images are shown in Fig. 7 - Fig. 9. Fig. 7 and Fig. 8 maps the classification results in terms of TP, FP, FN, and TN. While Fig. 9 maps the prediction results on random images in terms of crack and non-crack Although the classification performance of the 1D DFT-CNN was better, certain crack-like features were classified as non-cracks (Cyan color in Fig. 7). Certain local features were misclassified during the testing process, which required further investigation.

The chosen database of concrete images accounts for the effects of optical variability in terms of lighting conditions, the stand-off distance between the camera and target structure, focal lengths, field-of-view, and lens. In addition, we considered concrete surface texture variability by populating databased with images from different kinds of target structures. We estimate the quality of optical images in the database using a no-reference-based indicator, *Natural Image Quality Evaluation* (NIQE) [52]. NIQE is a natural scene statistics-based model that predicts the image quality using deviations in the image statistics due to artifacts created through optical variability. Fig. 10(a) shows a

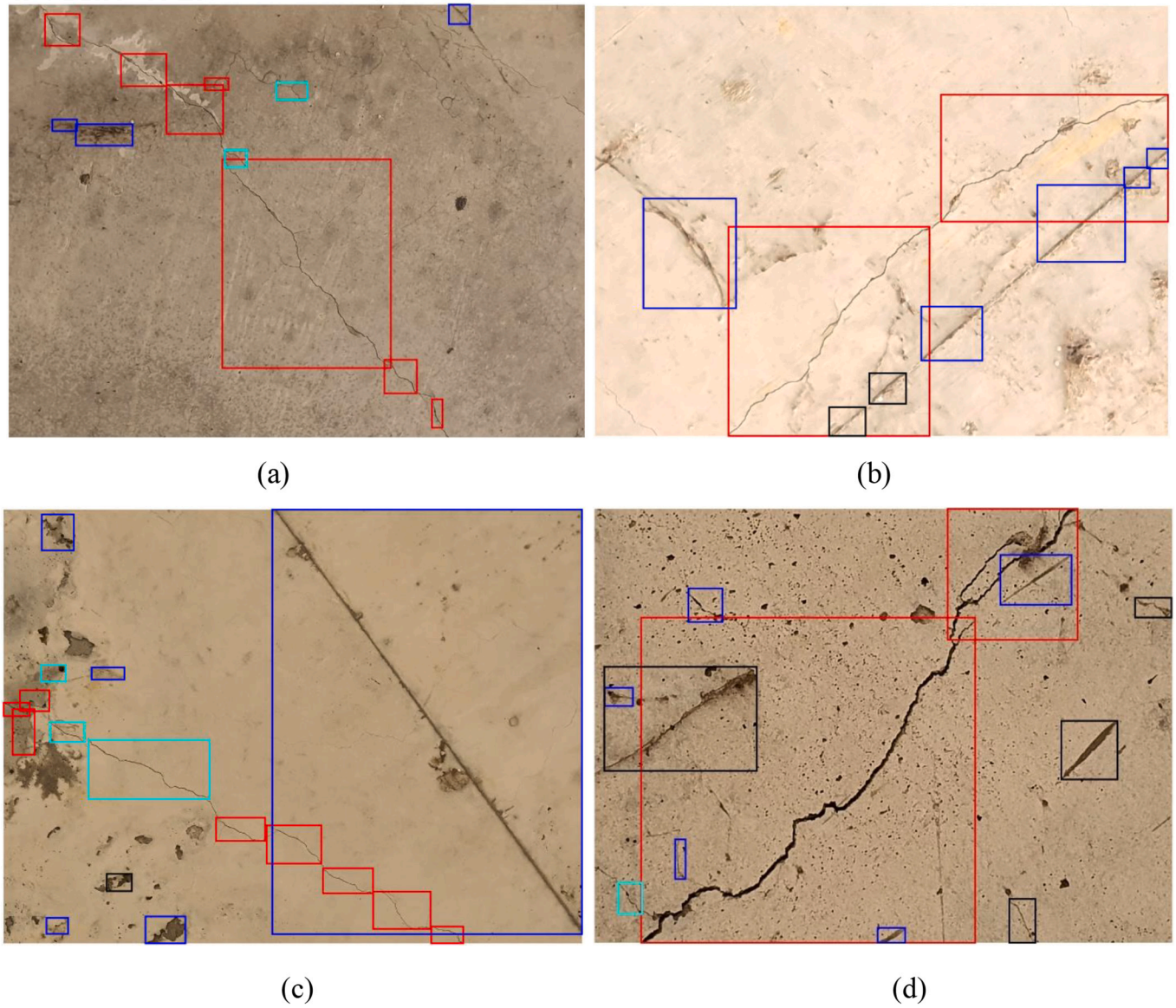

Fig. 8. Mapping TP, TN, FP, FN in test data set.

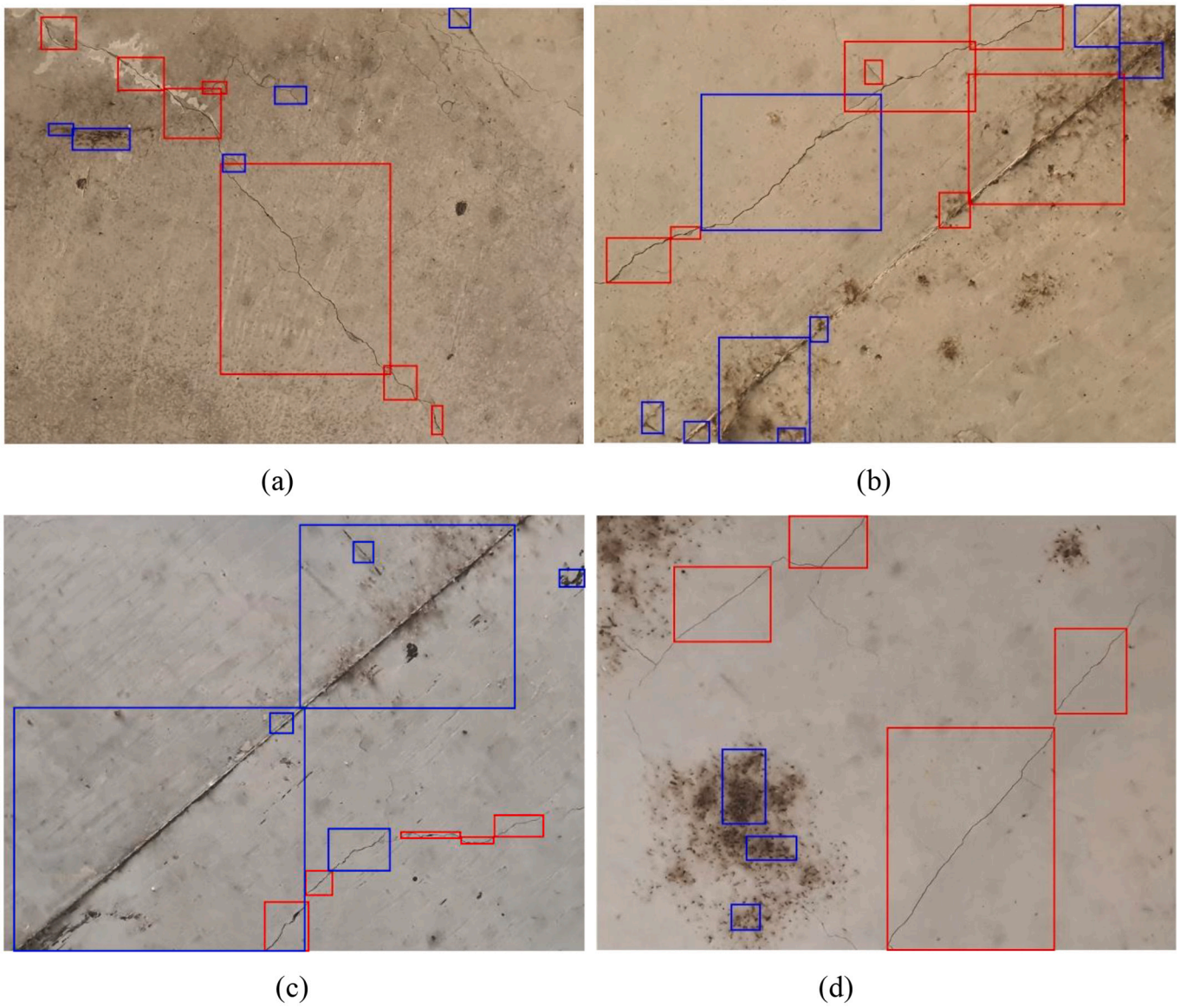

Fig. 9. Predicting crack and non-crack CCRs on random test images.

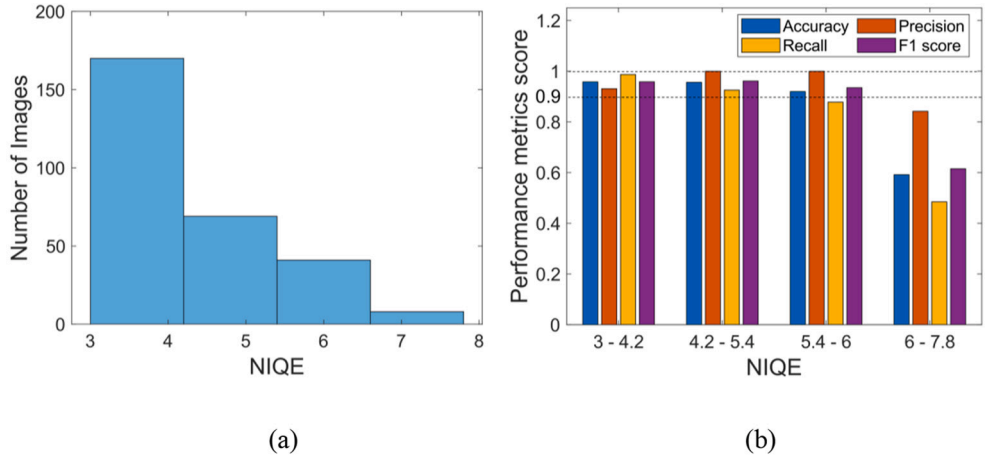

Fig. 10. (a) Histogram of optical images for different bands of NIQE scores used to train proposed DL model. (b) Corresponding performance metrics score of 1D-DFT-CNN for different bands of NIQE.

Table 2
Implementation time comparison for standard 2D-CNN architectures and proposed methodology.

Method	Training time	Testing time
Standard 2D-CNN	2 h 45 min 16 s	38–59 seconds per image [53]
1D-CNN-LSTM	1 h 12 min 4 s	5–7 seconds per image [53]
Sematic segmentation	18 h	350 seconds per image [54]
Image binarization with 1D-DFT-CNN	11 min, 55 s (inclusive of preprocessing and DL)	Desktop system: Approx 0.02 seconds per image (60 images per second) (excluding image preprocessing, only DL testing) Approx 0.1–0.2 seconds per image (5–10 images per second) (including image preprocessing and DL testing) Mobile platform: Approx 0.5 seconds per image (2 images per second) (excluding image preprocessing, only DL testing) Approx 2–2.5 seconds per image (including image preprocessing and DL testing)

histogram of optical images for different bands of NIQE scores used to train the proposed DL model. The range of NIQE scores in the database varies from 3 to 7.8, where the lowest NIQE score indicates an undistorted image with the best perceptual quality. The proposed model is trained and tested with this broad database, thereby accounting for environmental uncertainty and noise-induced effects. Ideally, one would expect the model to be stable within the range of uncertainty features considered in the training database and the model's prediction accuracy to decrease for any unmodelled effects or parameters, a defacto known fact and does not require a relooking. In contrast, we observe that the performance metrics of the 1D-DFT-CNN model are approximately constant for $3 \leq \text{NIQE} \leq 6$ and drastically reduced for $\text{NIQE} > 6$ (Fig. 10 (b)). A closer investigation into this discrepancy reveals a skewed weightage in the number of images used for different bands of NIQE, especially for $\text{NIQE} > 6$. In summary, the performance of the proposed system is stable for a major range ($\text{NIQE} \leq 6$) of image quality/noise ratio considered in the database. However, a detailed Probability Of Detection (POD) analysis with balanced and unbalanced datasets for different bands of NIQE is necessary considering the variabilities and environmental uncertainties, which is beyond the scope of this paper

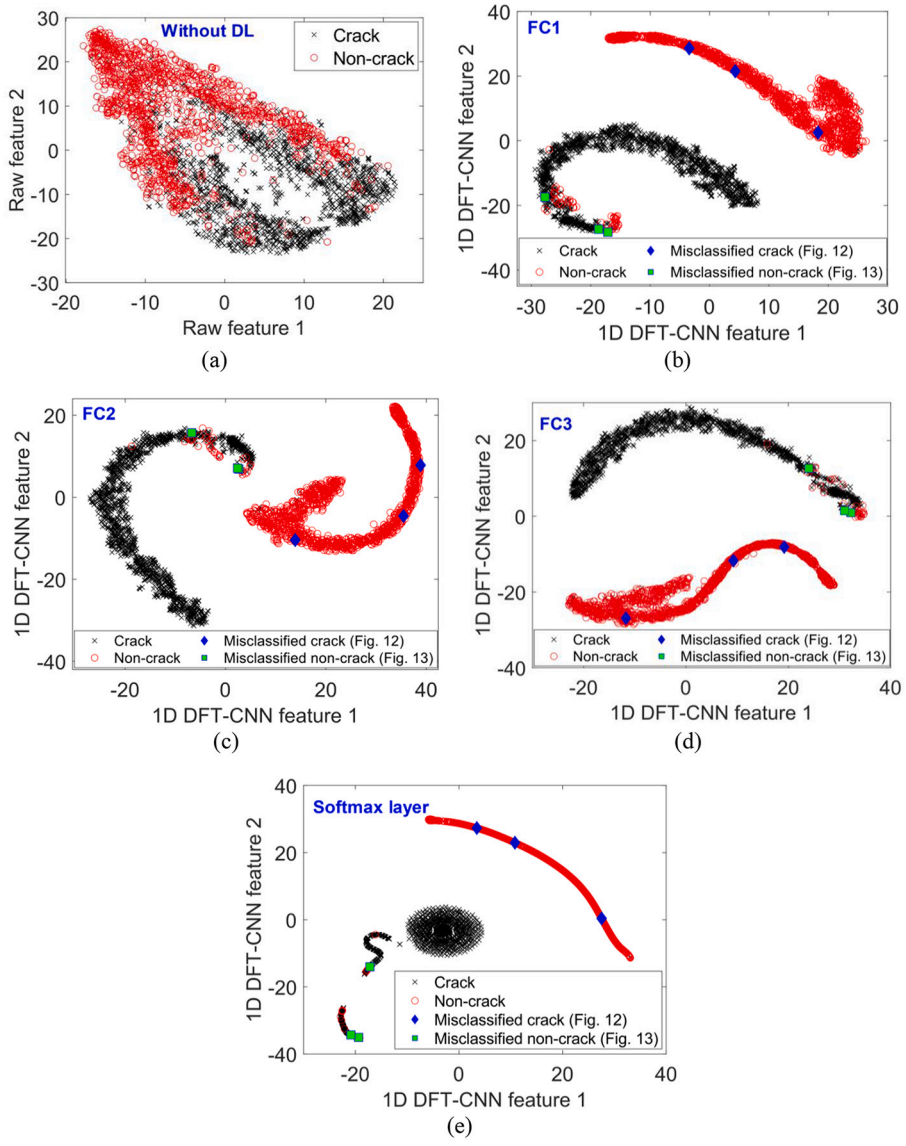

Fig. 11. Comparison of learned crack and non-crack discriminant features obtained from (a) a raw image (see Fig. 7) and different layers of proposed 1D-DFT-CNN architecture (b) FC1, (c) FC2, (d) FC3, and (e) softmax layer. (The notation 'FC' denotes a Fully Connected layer).

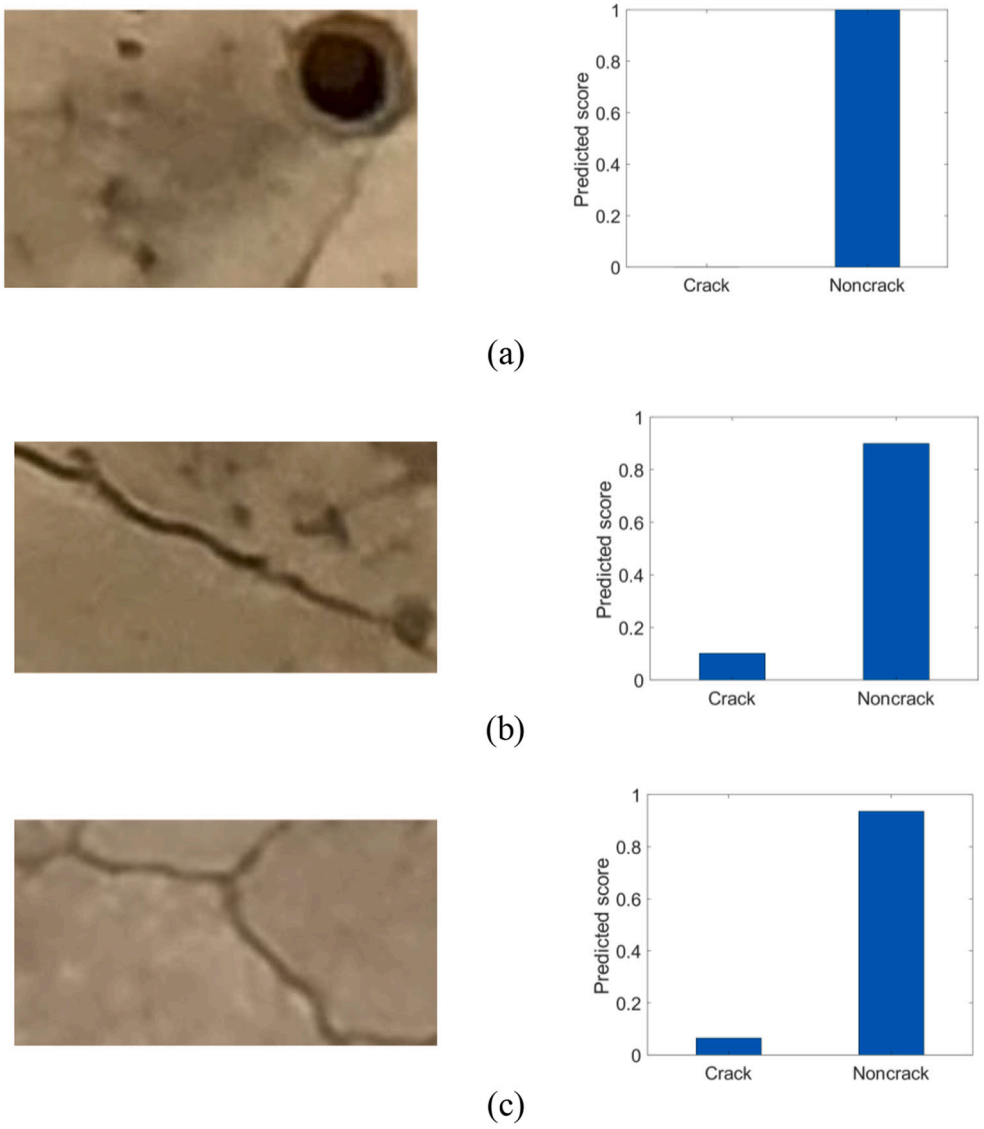

Fig. 12. Case study for crack CCRs classified as non-crack. Representative images of cracks and corresponding predicted scores from proposed DL network.

and will be considered as a separate work.

We demonstrated the capability of the proposed 1D-DFT-CNN DL network for a real-time implementation on a mobile handset test platform with limited computational capability. The software and computational details are as follows: The iPhone SE 2nd generation (2020) has an Apple A13 Bionic (7 nm+) chipset and a Hexa-core (2 × 2.65 GHz Lighting + 4 × 1.8 GHz Thunder) CPU. The system uses an 'ios' 14.4 operating system with Apple GPU (4-core graphics) and 64 GB storage capacity. Various hidden processing steps in the mobile testing platform include (i) loading a raw image with a size as large as 4608 × 3456 pixels, (ii) adaptive threshold-based integral image binarization, (iii) image filtering to remove noise using eccentricity and area filtering, (iv) CCR identification in the binarized image, and mapping the same to the raw image, and (v) loading the previously trained 1D-DFT-CNN network and predicting the crack and non-crack in the CCR using the trained network. Table 2 shows the computational time comparison between the existing 2D CNN, semantic segmentation, and 1D-DFT-CNN based DL architectures. The testing time for the standard 2D-CNN, semantic segmentation, and 1D-DFT-CNN was approximately 38–59 s/image [53], 350 s/image [54], and 0.02 s/image (60 images/s), respectively, when implemented on a desktop computer. The testing time for 1D-DFT-CNN on the mobile testing platform is approximately 0.5 s/image (2 images/s), which excludes all hidden processing steps (i) –(iv). However, in Table 2 we have indicated the time for "Image binarization with 1D-DFT-CNN" without (step (v)) and with image preprocessing (steps (i)-(v)). The fast implementation of 1D-DFT-CNN makes it ideal for the real-time detection and classification of concrete cracks from non-cracks with reasonably good accuracy.

4.2. XAI understanding of the proposed architecture

Recently, the vast success demonstrated by artificial intelligence, primarily driven by DL, has enabled the reduction of the gap between machine-level performance and human-level performance in identifying cracks from non-cracks. However, deep learning is still considered a "black box" because of its weak interpretability and the unknown reasoning behind the classification or predicted results. The prerequisite for understanding metadata transfer during unsupervised deep learning implementation [55] for critical applications, that is, crack detection of critical load-bearing structures in our case, is a good understanding of metadata transfer during the supervised DL process. The complex discriminant features learnt in DL for classifying crack from non-crack are still a mystery. Here we investigate the significant discriminant features learned within metadata and how they are transferred across

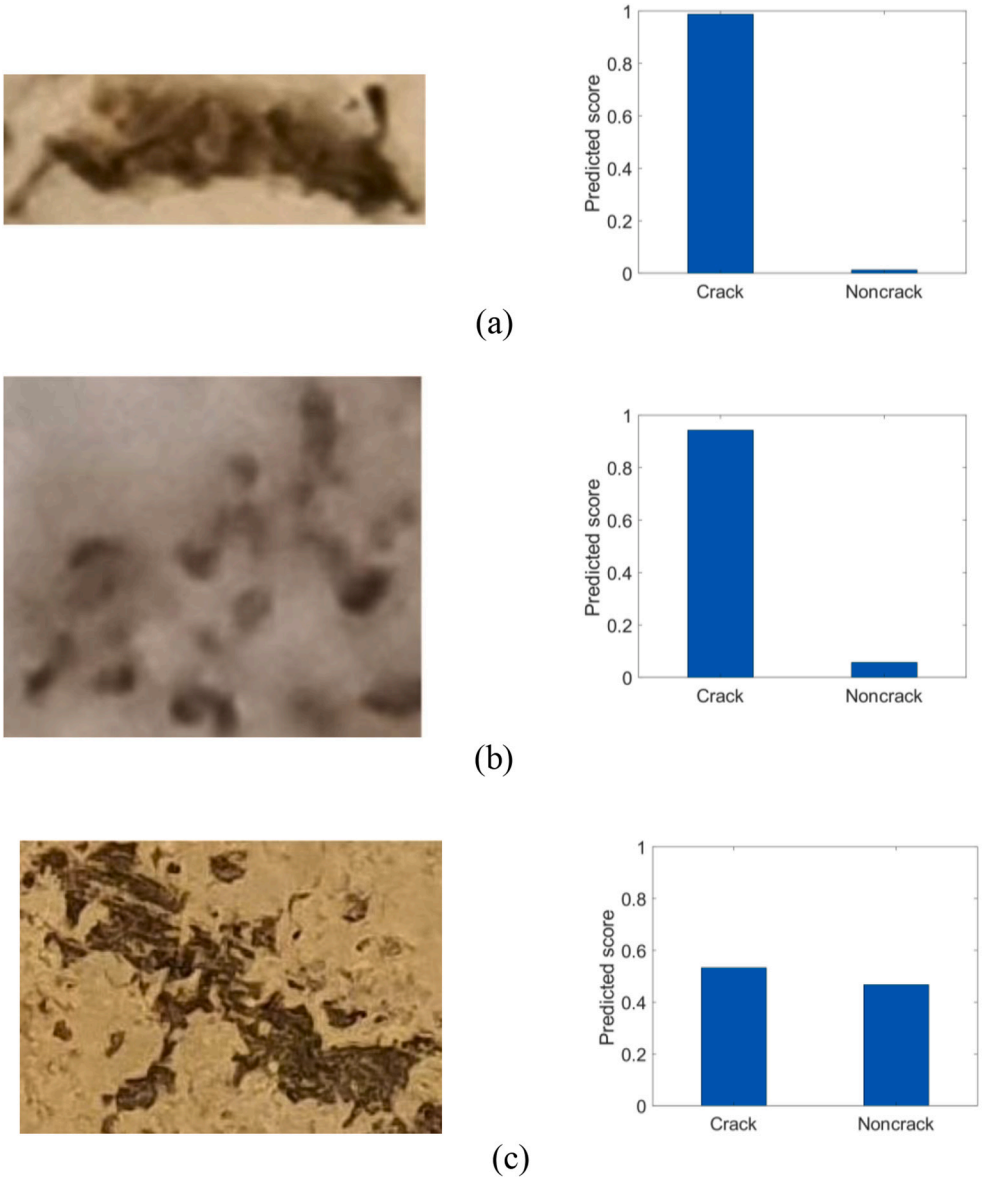

Fig. 13. Case study for non-crack CCRs classified as crack. Representative images of non-crack (a), (c), (e) and corresponding predicted scores from proposed DL network.

these hidden layers. Also, the XAI enables (i) verification of unintentional leakage of discriminant features and (ii) data shift arising due to the difference in training and testing data [56].

Fig. 11 shows the comparative visualization of raw features extracted from the original image and deep-learned features using metadata from different layers of the 1D-DFT-CNN. DL learning transforms the raw images from the pixel space to a new space, where the class discriminant is amplified to separate the crack features from the non-crack features, as shown in Fig. 11(b)-(e). The distinguishable crack and non-crack features learned by the proposed deep-learning framework for fully connected layers FC1, FC2, FC3, and softmax layers are shown in Fig. 11 (b), (c), (d), and (e), respectively. XAI-based metadata visualization is conducted using t-SNE (Section 2.3), which maps the complex multidimensional data during deep learning to a two-dimensional space. The multi-scaled discriminant features learned from different neurons or perceptron's in each layer cluster together when mapped to a low-dimensional space using t-SNE. Local cluster segregation with a prescribed mean and standard deviation enables us to visualize the classification process in the metadata space. The discriminant scales are relative to the maximum and minimum values of the amplitude of the deep-learned features in each layer. The t-SNE computation is highly complex due to high dimensional discriminant in multi-scaled space. For fast computation, we assume the maximum size of the high-dimensional discriminant equal to the maximum number of neurons corresponding to each layer. While analyzing the raw image (Fig. 7), we perform a dimensional reduction of the principal components and map to a 2D space (see Fig. 11(a)). We infer that DL clusters the discriminant features separately across different hidden layers compared to principal features learned from the conventional principal component analysis.

Further, we investigate the reason for the misclassification of certain local CCRs shown in Fig. 8. Although DL shows promising results in predicting crack signatures from background features, overwhelming evidence [57,58] proves that DL is non-robust or unstable in certain specific cases. The robustness of the neural network algorithms is at the heart of the numerical analysis involving weights, bias, feed-forward propagation of input information through hidden layers to the output layer, and back-propagation-based error minimization of the cost function [59]. The nonlinear neural network is an approximation of a

continuous function. On deeper investigation of Smale's 18th mathematical problem for the 21st century on the limits of AI [60], we could derive the following explanation. Currently, no algorithm can train a neural network to an accuracy of K digits bounded via condition numbers. Only under certain case-specific conditions can one compute it to the desired accuracy of K digits. These numerical errors significantly influence the output classification results leading to misclassification in certain instances. With this background, we can point to examples shown in Fig. 12, which is a classic case of misclassification of crack as non-crack, even when the shape and characteristics match the crack features. Fig. 12 gives the predicted score of CCRs that are classified as False Negatives (crack-like CCRs predicted as non-crack). While Fig. 13 gives the predicted score of False Positives or non-crack-like CCRs predicted as crack. To investigate further, we mapped back the misclassified false negatives and false positives, as shown in Fig. 12 and Fig. 13, to the t-SNE metadata space (see Fig. 11(b)-(e)). Fig. 11(b)-(e) shows that crack signatures of images shown in Fig. 12(a)-(c) are clustered with non-crack in the metadata space and hence misclassified as non-crack. Similarly, non-crack signatures of images shown in Fig. 13(a)-(c) are clustered with cracks in the metadata space, hence misclassified as crack. Using Fig. 11(b)-(e), one can generalize that learned crack features in the first hidden layer are clustered with non-crack features and carried forward. Another possible reason for the cracks in Fig. 12(b) and (c) to be misclassified even after they have obvious shape and color characteristics is as follows: NIQE score of the segmented image shown in Fig. 12(b) and (c) are 18.8 and 6.6, which falls to the skewed bands of NIQE, used in the proposed model (see Fig. 10(a)). The prediction accuracy in the skewed regime drastically reduces to approximately 60% (see Fig. 10(b)). We also reevaluate the corresponding output class labels by finding the prediction scores predicted by the softmax layer. The uncertain classification can also be attributed to a multinomial probability distribution that extracts the higher probability for classification, which leads to false alarms (Fig. 12 and Fig. 13). The approach is another DL-based interpretability method used to predict the image category, regardless of its architecture, thereby overcoming the limitations of Class Activation Maps based XAI [61].

In Fig. 11(e), we extract the multi-scale discriminant features from the softmax layer for the trained data with known ground truth and map to a 2D dimensional space for visualization. t-SNE-based XAI visualization provides only insights into the data's class structure, and any derived information is subjective and incoherent to all datasets. Although the discriminant features extracted using t-SNE for the current dataset clusters in the form of a circle and adjacent linear patterns and the metadata obtained from the misclassified non-crack are on the linear pattern, it cannot be generalized due to following reasons. (i) t-SNE-based metadata visualization depends on the choice of the optimization parameters as the cost function is non-convex, and (ii) the patterns are subjective to the local nature of data making it sensitive to the intrinsic dimensionality of data. Due to the prevailing assumptions, by definition of t-SNE, it is impossible to fully represent the structure of intrinsically high-dimensional data in two or three-dimensional space.

5. Conclusions

In this paper, adaptive threshold-based integral image binarization was used as a pre-processor to a 1D DL model with a Fourier basis for real-time detection and classification of cracks and non-crack features on concrete surfaces. The proposed framework first identifies possible CCRs in 2D pixel space using adaptive threshold-based integral image binarization, which works effectively regardless of the size or the lighting conditions of the test image. Subsequently, the CCRs are transformed and vectorized in the Fourier basis space. The pre-processed image was fed into a 1D DFT-CNN for training and testing. A Fourier-based 1D-CNN contains spatially distinct frequency features for cracks and non-cracks, effectively enabling a shallow neural network to segregate distinct features in the early layers. Pre-processing eliminates non-CCR, which occupies a significant segment of the concrete region, thereby reducing the data fed into the 1D neural network. The robustness and adaptability of the proposed scheme are demonstrated on a database obtained from different structures under various lighting conditions. The designed framework for distinguishing between cracks and non-cracks showed promising results, with an F_1 score of 0.9823.

Using the computational efficiency of a 1D CNN for 1D DFT-CNN, the forward and backward propagation matrix operations in conventional 2D CNNs are replaced with simple 1D vector array operations. The 1D DFT-CNN with shallow architectures learns hidden features with fewer neurons than the conventional 2D CNN, requiring a deeper architecture to learn the same. The proposed framework (i) removes the conventional time-consuming sliding window technique to scan large images and (ii) eliminates the necessity of labor-intensive pixel-level labeling, as implemented in semantic segmentation. The advantages of the proposed scheme enable scalability irrespective of the image size. This paper is relevant in the automated computer vision-based SHM of concrete structures to detect and classify crack candidates from the remaining structural features using a mobile platform with limited computational facility. The capability of the technique is demonstrated for the real-time classification of cracks and non-cracks on a mobile platform at the rate of approximately 2 images/s. The performance results from the computationally faster shallow 1D-DFT-CNN are plausibly comparable with conventional 2D CNNs and semantic segmentation. Further, the hidden knowledge transfer between layers of the 1D CNN using local XAI are investigated, where t-SNE is used to visualize multidimensional deep learned features in 2D space. DL clusters discriminant features separately across different hidden layers compared to features learned from the principal component analysis.

Acknowledgement

This work has supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2020R1A2C2014797).

Appendix 1. Sample images of crack and non-crack used for training and testing

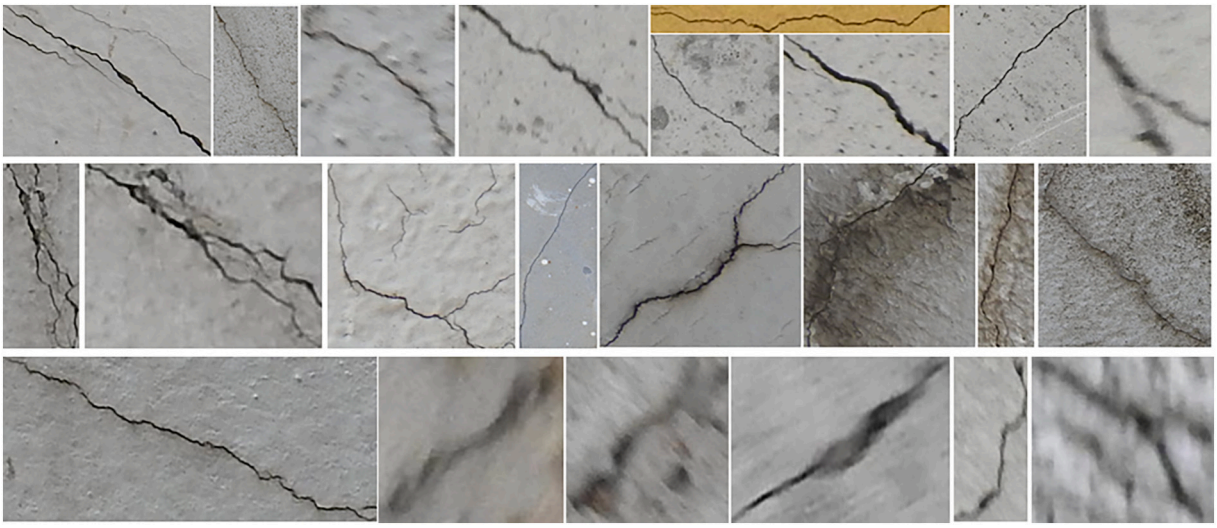

Fig. A.1. Representative images of crack from the database.

Fig. A.2. Representative images of non-crack from the database.

References

[1] L.E. Campbell, R.J. Connor, J.M. Whitehead, G.A. Washer, Human factors affecting visual inspection of fatigue cracking in steel bridges, Struct. Infrastruct. Eng. 17 (11) (2020) 1447–1458, https://doi.org/10.1080/15732479.2020.1813783.

[2] C. Dumoulin, A. Deraemaeker, Real-time fast ultrasonic monitoring of concrete cracking using embedded piezoelectric transducers, Smart Mater. Struct. 26 (10) (2017), 104006, https://doi.org/10.1088/1361-665X/aa765e.

[3] G. Kolappan Geetha, D. Roy Mahapatra, S. Gopalakrishnan, S. Hanagud, Laser Doppler imaging of delamination in a composite T-joint with remotely located ultrasonic actuators, Compos. Struct. 147 (2016) 197–210, https://doi.org/10.1016/j.compstruct.2016.03.039.

[4] G. Kolappan Geetha, V. Rathod, N. Chakraborty, D. Roy Mahapatra, S. Gopalakrishnan, Rapid localization and ultrasonic imaging of multiple damages in structural panel with piezoelectric sensor-actuator network, in: http://dpi-proceedings.com/index.php/shm2011/article/view/22254, 2011.

[5] G. Kolappan Geetha, R.K. Munian, D. Roy Mahapatra, C.-W. In, D.A. Raulerson, Ultrasonic horn contact-induced transient anharmonic resonance effect on vibro-thermography, J. Sound Vib. (2022) 116786, https://doi.org/10.1016/j.jsv.2022.116786.

[6] G. Kolappan Geetha, D. Roy Mahapatra, C.-W. In, D. Raulerson, Transient vibro-thermography and nonlinear resonant modes, J. Vib. Acoust. (Apr. 2020) 1–21, https://doi.org/10.1115/1.4046860.

[7] K. Jang, H. Jung, Y.-K. An, Automated bridge crack evaluation through deep super resolution network-based hybrid image matching, Autom. Constr. 137 (2022), 104229, https://doi.org/10.1016/j.autcon.2022.104229.

[8] G. Kolappan Geetha, D. Roy Mahapatra, Modeling and simulation of vibro-thermography including nonlinear contact dynamics of ultrasonic actuator, Ultrasonics 93 (2019) 81–92, https://doi.org/10.1016/j.ultras.2018.11.001.

[9] B.F. Spencer Jr., V. Hoskere, Y. Narazaki, Advances in computer vision-based civil infrastructure inspection and monitoring, Engineering 5 (2) (2019) 199–222, https://doi.org/10.1016/j.eng.2018.11.030.

[10] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, P. Fieguth, A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure, Adv. Eng. Inform. 29 (2) (2015) 196–210, https://doi.org/10.1016/j.aei.2015.01.008.

[11] J.M.W. Brownjohn, Structural health monitoring of civil infrastructure, Philos. Trans. R. Soc. A Math. Phys. Eng. Sci. 365 (1851) (2007) 589–622. ISBN: 978-1-84569-392-3.

[12] C.J. Larosche, Types and causes of cracking in concrete structures, in: In Failure, Distress And Repair of Concrete Structures, Elsevier, 2009, pp. 57–83, https://doi.org/10.1533/9781845697037.1.57.

[13] I. Abdel-Qader, O. Abudayyeh, M.E. Kelly, Analysis of edge-detection techniques for crack identification in bridges, J. Comput. Civ. Eng. 17 (4) (2003) 255–263, https://doi.org/10.1061/(ASCE)0887-3801(2003)17:4(255).

[14] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324. https://ieeexplore.ieee.org/document/726791.

[15] Y.-A. Hsieh, Y.J. Tsai, Machine learning for crack detection: review and model performance comparison, J. Comput. Civ. Eng. 34 (5) (2020) 4020038, https://doi.org/10.1061/(ASCE)CP.1943-5487.0000918.

[16] D. Lattanzi, G.R. Miller, Robust automated concrete damage detection algorithms for field applications, J. Comput. Civ. Eng. 28 (2) (2014) 253–262, https://doi.org/10.1061/(ASCE)CP.1943-5487.0000257.

[17] Y.-J. Cha, W. Choi, O. Büyüköztürk, Deep learning-based crack damage detection using convolutional neural networks, Comput. Civ. Infrastruct. Eng. 32 (5) (2017) 361–378, https://doi.org/10.1111/mice.12263.

[18] H. Kim, E. Ahn, M. Shin, S.-H. Sim, Crack and noncrack classification from concrete surface images using machine learning, Struct. Health Monit. 18 (3) (2019) 725–738, https://doi.org/10.1177/1475921718768747.

[19] C.V. Dung, L.D. Anh, Autonomous concrete crack detection using deep fully convolutional neural network, Autom. Constr. 99 (2019) 52–58, https://doi.org/10.1016/j.autcon.2018.11.028.

[20] C. Zhang, C. Chang, M. Jamshidi, Simultaneous pixel-level concrete defect detection and grouping using a fully convolutional model, Struct. Health Monit. 20 (4) (2021) 2199–2215, https://doi.org/10.1177/1475921720985437.

[21] Y.-J. Cha, W. Choi, G. Suh, S. Mahmoudkhani, O. Büyüköztürk, Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types, Comput. Civ. Infrastruct. Eng. 33 (9) (2018) 731–747, https://doi.org/10.1111/mice.12334.

[22] M. Abdellatif, H. Peel, A.G. Cohn, R. Fuentes, Combining block-based and pixel-based approaches to improve crack detection and localisation, Autom. Constr. 122 (2021), 103492, https://doi.org/10.1016/j.autcon.2020.103492.

[23] U.A. Nnolim, Fully adaptive segmentation of cracks on concrete surfaces, Comput. Electr. Eng. 83 (2020), 106561, https://doi.org/10.1016/j.compeleceng.2020.106561.

[24] D. Kang, S.S. Benipal, D.L. Gopal, Y.-J. Cha, Hybrid pixel-level concrete crack segmentation and quantification across complex backgrounds using deep learning, Autom. Constr. 118 (2020), 103291, https://doi.org/10.1016/j.autcon.2020.103291.

[25] A. Zhang, et al., Deep learning–based fully automated pavement crack detection on 3D asphalt surfaces with an improved CrackNet, J. Comput. Civ. Eng. 32 (5) (2018) 4018041, https://doi.org/10.1061/(ASCE)CP.1943-5487.0000775.

[26] U.H. Billah, H.M. La, A. Tavakkoli, Deep learning-based feature silencing for accurate concrete crack detection, Sensors 20 (16) (2020) 4403, https://doi.org/10.3390/s20164403.

[27] X. Yang, et al., Research and applications of artificial neural network in pavement engineering: a state-of-the-art review, J. Traffic Transp. Eng. 8 (6) (2021) 1000–1021, https://doi.org/10.1016/j.jtte.2021.03.005 (English Ed.).

[28] J. Guan, X. Yang, L. Ding, X. Cheng, V.C.S. Lee, C. Jin, Automated pixel-level pavement distress detection based on stereo vision and deep learning, Autom. Constr. 129 (2021), 103788, https://doi.org/10.1016/j.autcon.2021.103788.

[29] A. Das, P. Rad, Opportunities and Challenges in Explainable Artificial Intelligence (XAI): A Survey, *arXiv Prepr. arXiv2006.11371*, 2020, https://doi.org/10.48550/arXiv.2006.11371.

[30] S.L. Brunton, J.N. Kutz, Data-Driven Science and Engineering, Cambridge University Press, 2019, https://doi.org/10.1017/9781108380690.

[31] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, D.J. Inman, 1D convolutional neural networks and applications: a survey, Mech. Syst. Signal Process. 151 (2021), 107398, https://doi.org/10.1016/j.ymssp.2020.107398.

[32] N. Otsu, A threshold selection method from gray-level histograms, IEEE Trans. Syst. Man. Cybern. 9 (1) (1979) 62–66, https://doi.org/10.1109/TSMC.1979.4310076.

[33] J.-L. Fan, B. Lei, A modified valley-emphasis method for automatic thresholding, Pattern Recogn. Lett. 33 (6) (2012) 703–708, https://doi.org/10.1016/j.patrec.2011.12.009.

[34] H.-F. Ng, D. Jargalsaikhan, H.-C. Tsai, C.-Y. Lin, An improved method for image thresholding based on the valley-emphasis method, in: In *IEEE Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 2013, pp. 1–4, https://doi.org/10.1109/APSIPA.2013.6694261.

[35] M.T.N. Truong, S. Kim, Automatic image thresholding using Otsu's method and entropy weighting scheme for surface defect detection, Soft. Comput. 22 (13) (2018) 4197–4203, https://doi.org/10.1007/s00500-017-2709-1.

[36] J. Sauvola, M. Pietikäinen, Adaptive document image binarization, Pattern Recogn. 33 (2) (2000) 225–236, https://doi.org/10.1016/S0031-3203(99)00055-2.

[37] W. Niblack, An introduction to digital image processing, Prentice Hall, 1985. ISBN: 978–0134806747.

[38] C. Wolf, J.-M. Jolion, Extraction and recognition of artificial text in multimedia documents, Form. Pattern Anal. Appl. 6 (4) (2004) 309–326, https://doi.org/10.1007/s10044-003-0197-7.

[39] H. Kim, E. Ahn, S. Cho, M. Shin, S.-H. Sim, Comparative analysis of image binarization methods for crack identification in concrete structures, Cem. Concr. Res. 99 (2017) 53–61, https://doi.org/10.1016/j.cemconres.2017.04.018.

[40] D. Bradley, G. Roth, Adaptive thresholding using the integral image, J. Graph. tools 12 (2) (2007) 13–21, https://doi.org/10.1080/2151237X.2007.10129236.

[41] P.D. Wellner, Adaptive thresholding for the digital desk, in: Xerox, EPC1993-110, 1993, pp. 1–19. https://www.semanticscholar.org/paper/Adaptive-Thresholding-for-the-DigitalDesk-Wellner/ea59dc10e8cf6f2d13088415d72b48f417f817bd.

[42] T. Julliand, V. Nozick, H. Talbot, Image noise and digital image forensics, Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) 9569 (2016) 3–17, https://doi.org/10.1007/978-3-319-31960-5_1.

[43] T. Yamaguchi, S. Hashimoto, Automated crack detection for concrete surface image using percolation model and edge information, in: In *IECON 2006-32nd Annual Conference on IEEE Industrial Electronics*, 2006, pp. 3355–3360, https://doi.org/10.1109/IECON.2006.348070.

[44] S.G. Chang, B. Yu, M. Vetterli, Adaptive wavelet thresholding for image denoising and compression, IEEE Trans. Image Process. 9 (9) (2000) 1532–1546, https://doi.org/10.1109/83.862633.

[45] M. Zhang, B.K. Gunturk, Multiresolution bilateral filtering for image denoising, IEEE Trans. Image Process. 17 (12) (2008) 2324–2333, https://doi.org/10.1109/TIP.2008.2006658.

[46] J. Portilla, V. Strela, M.J. Wainwright, E.P. Simoncelli, Image denoising using scale mixtures of Gaussians in the wavelet domain, IEEE Trans. Image Process. 12 (11) (2003) 1338–1351, https://doi.org/10.1109/TIP.2003.818640.

[47] R.C. Gonzalez, R.E. Woods, others, Digital Image Processing 455, Pearson Publ, 2002. ISBN-10: 9780133356724.

[48] L. der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008). http://jmlr.org/papers/v9/vandermaaten08a.html.

[49] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958. https://dl.acm.org/doi/10.5555/2627435.2670313.

[50] J. Bjorck, C. Gomes, B. Selman, K.Q. Weinberger, Understanding Batch Normalization, arXiv Prepr. arXiv1806.02375, 2018, https://doi.org/10.48550/arXiv.1806.02375.

[51] E. Hoffer, I. Hubara, D. Soudry, Train longer, generalize better: closing the generalization gap in large batch training of neural networks, Adv. Neural Inf. Proces. Syst. 30 (2017), https://doi.org/10.48550/arXiv.1705.08741.

[52] A. Mittal, R. Soundararajan, A.C. Bovik, Making a 'completely blind' image quality analyzer, IEEE Signal Process. Lett. 20 (3) (2012) 209–212, https://doi.org/10.1109/LSP.2012.2227726.

[53] Q. Zhang, K. Barri, S.K. Babanajad, A.H. Alavi, Real-time detection of cracks on concrete bridge decks using deep learning in the frequency domain, Engineering (2020), https://doi.org/10.1016/j.eng.2020.07.026.

[54] S. Zhou, W. Song, Crack segmentation through deep convolutional neural networks and heterogeneous image fusion, Autom. Constr. 125 (2021), 103605, https://doi.org/10.1016/j.autcon.2021.103605.

[55] Y. Bengio, Deep learning of representations for unsupervised and transfer learning, in: In *Proceedings of ICML workshop on unsupervised and transfer learning*, 2012, pp. 17–36, in: https://proceedings.mlr.press/v27/bengio12a.html.

[56] Sotiris Kotsiantis, Linardatos Pantelis, Vasilis Papastefanopoulos, Explainable AI: A review of machine learning interpretability methods, Entropy 23 (1) (2020), https://doi.org/10.3390/e23010018 no. 18.

[57] C.Q. Choi, 7 revealing ways AIs fail: neural networks can be disastrously brittle, forgetful, and surprisingly bad at math, IEEE Spectr. 58 (10) (2021) 42–47, https://doi.org/10.1109/MSPEC.2021.9563958.

[58] D. Heaven, others, Why deep-learning AIs are so easy to fool, Nature 574 (7777) (2019) 163–166. https://www.nature.com/articles/d41586-019-03013-5.

[59] N.J. Higham, Accuracy and stability of numerical algorithms, SIAM (2002), https://doi.org/10.1137/1.9780898718027.

[60] S. Smale, Mathematical problems for the next century, Math. Intell. 20 (2) (1998) 7–15, https://doi.org/10.1007/BF03025291.

[61] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2921–2929, https://doi.org/10.48550/arXiv.1512.04150.