

Article

Deep Learning-Based Crack Identification for Steel Pipelines by Extracting Features from 3D Shadow Modeling

Wael A. Altabey ^{1,2}, Mohammad Noori ^{1,3,*}, Tianyu Wang ¹, Ramin Ghiasi ¹, Sin-Chi Kuok ⁴
and Zhishen Wu ^{1,*}

¹ International Institute for Urban Systems Engineering (IIUSE), Southeast University, Nanjing 210096, China; wael.altabey@gmail.com (W.A.A.); ty_wang@seu.edu.cn (T.W.); rghiasi.s@gmail.com (R.G.)

² Department of Mechanical Engineering, Faculty of Engineering, Alexandria University, Alexandria 21544, Egypt

³ Department of Mechanical Engineering, California Polytechnic State University, San Luis Obispo, CA 93405, USA

⁴ State Key Laboratory of Internet of Things for Smart City, Department of Civil and Environmental Engineering, and Guangdong-Hong Kong-Macau Joint Laboratory for Smart City, University of Macau, Macau, China; sckuok@um.edu.mo

* Correspondence: mnoori@outlook.com (M.N.); zsw@seu.edu.cn (Z.W.); Tel.: +1-805-903-2411 (M.N.); +86-25-86266776 (Z.W.)



Citation: Altabey, W.A.; Noori, M.; Wang, T.; Ghiasi, R.; Kuok, S.-C.; Wu, Z. Deep Learning-Based Crack Identification for Steel Pipelines by Extracting Features from 3D Shadow Modeling. *Appl. Sci.* **2021**, *11*, 6063. <https://doi.org/10.3390/app11136063>

Academic Editor: José A.F.O. Correia

Received: 18 May 2021

Accepted: 18 June 2021

Published: 29 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Abstract: Automatic crack identification for pipeline analysis utilizes three-dimensional (3D) image technology to improve the accuracy and reliability of crack identification. A new technique that integrates a deep learning algorithm and 3D shadow modeling (3D-SM) is proposed for the automatic identification of corrosion cracks in pipelines. Since the depth of a corrosion crack is below the surrounding area of the crack, a shadow of the crack is projected when the crack is exposed under light sources. In this study, we analyze the shadow areas of cracks through 3D shadow modeling (3D-SM) and identify the evolving cracks through the shape analysis of the shadows. To denoise the 3D images, the connected domain analysis is implemented so that the shadow groups of the evolving cracks can be retained and the scattered shadow groups that occur due to insignificant defects can be eliminated. Moreover, a novel deep neural network is developed to process the 3D images. The proposed automatic crack identification method successfully processes the 3D images efficiently and accurately diagnoses the corrosion cracks. Experimental results show that the proposed method achieves satisfactory performance with 93.53% accuracy and a 92.04% regression rate.

Keywords: deep learning; automatic crack identification; convolutional neural network (CNN); 3D shadow modeling; structural health monitoring (SHM)

1. Introduction

Steel pipelines play an important role in gas and liquid transportation over long distances. However, due to the harsh environment of the construction locations, pipelines always suffer from structural damage problems caused by corrosion, cracks, etc. Pipeline corrosion has been defined for decades as a major source of pipeline deterioration in transmission lines [1]. In common engineering practices, corrosion crack identification highly relies on manual detection and subjective decision. In recent years, nondestructive evaluation (NDE) approaches have been proposed for corrosion crack identification in steel pipelines. NDE methods detect corrosion cracks of steel pipelines through different mediums, such as X-ray, gamma-ray radiography, ultrasonic, thermography, eddy current [2,3], fiber optic distributed [4,5] and electrical capacitance sensors [6–12]. However, due to the different needs for heat or wave sources and complicated data analyzers of different NDE methods, such applications can encounter various difficulties in terms of operational and monitoring requirements. Moreover, it may not be possible to achieve thorough monitoring of the wide fields [13].

Automatic detection technology of corrosion cracks has attracted increasing applications in actual projects. The early image processing algorithms based on binary image processing (BIP) technology are studied by pioneer researchers to detect the corrosion and cracks of steel pipelines. BIP extracts crack information by determining the optimal threshold, but due to the existence of noise information in steel pipeline images, BIP are still unable to accurately identify cracks [14]. Based on the strong edge features of cracks, some researchers have carried out research on crack identification algorithms based on edge features [15], studied image segmentation methods based on fractal features [16], and considered the comprehensive detection of the crack area and boundary information. However, for automatic identification of pipeline cracks, many algorithms have the problem of low accuracy and low speed. Real-time processing of pipeline images is difficult to be realized automatically.

For mechanical and civil structures, real-time structural health monitoring is difficult when the types of structures are complicated or measured signals are corrupted due to environmental noise. Thus, model-based structural damage identification approaches are not effective. Deep learning can satisfactorily address this problem due to its superior adaptive learning of datasets [17–21]. A large and deep convolutional neural network (CNN) was employed to make a classification of 1.2 million high-resolution images by Krizhevsky et al. [22]. Derivation and application of CNNs were presented by Bouvrie [23]. Real-time vibration-based damage detection and localization methods were proposed using CNNs [24–29]. Wavelets were used to indicate damage occurrence of the structure under seismic load excitation by monitoring the mutations in the wavelet details of responses. [30–32]. A new signal-processing algorithm was formed by combining wavelets, neural networks and Hilbert transform, inspired by the deep learning model [33–37]. A Bayesian network was studied to evaluate the reliability of specific components according to their serviceability and inter-component correlations [38–42], and a hybrid response surface method was used [43].

Due to its flourishing development, deep neural networks have shown powerful capabilities to establish surrogate models with high accuracy and stability for crack detection. Taking the benefits of deep learning, an automatic crack identification method is proposed. The proposed method integrates the 3D-SM and convolutional deep neural network to extract the features of the 3D-SM images and identify the corrosion cracks. The proposed method provides a promising way to detect pipeline corrosion cracks regardless of the disturbing backgrounds and conditions of the 3D images of the pipeline's outer surface.

2. Methodology

The proposed method integrates the 3D-SM with CNN for automatic crack identification in pipelines. Figure 1 shows a flowchart of the proposed method. To ensure the quality of the pipeline's outer surface images, each pixel of the images is covered under the maximum intensity of light beam projection. Once the maximum intensity of all pixels is collected, digitalization of the collected data will be carried out by converting the images to binary maps (shadow maps). In the digitalization of images, we compare the light intensity at each pixel of the pipeline surface with the maximum light intensity and label the pixels accordingly. If the maximum light intensity exceeds the pipeline surface intensity, this pixel will be marked as a shaded area and labeled with the binary number "0". Otherwise, the pixel will be marked as a non-shaded area and labeled with the binary number "1". After the labeling, the shadow maps are fed to a CNN as labeled data. First, we train and calculate the network I/P and O/P weights. Then, we compute the derivative of the error for weights by the backpropagation (BP) algorithm. In the CNN, the convolution layer is used for up-sampling while the subsampling layer is used for down-sampling. The following sections present more in-depth discussions on the proposed work.

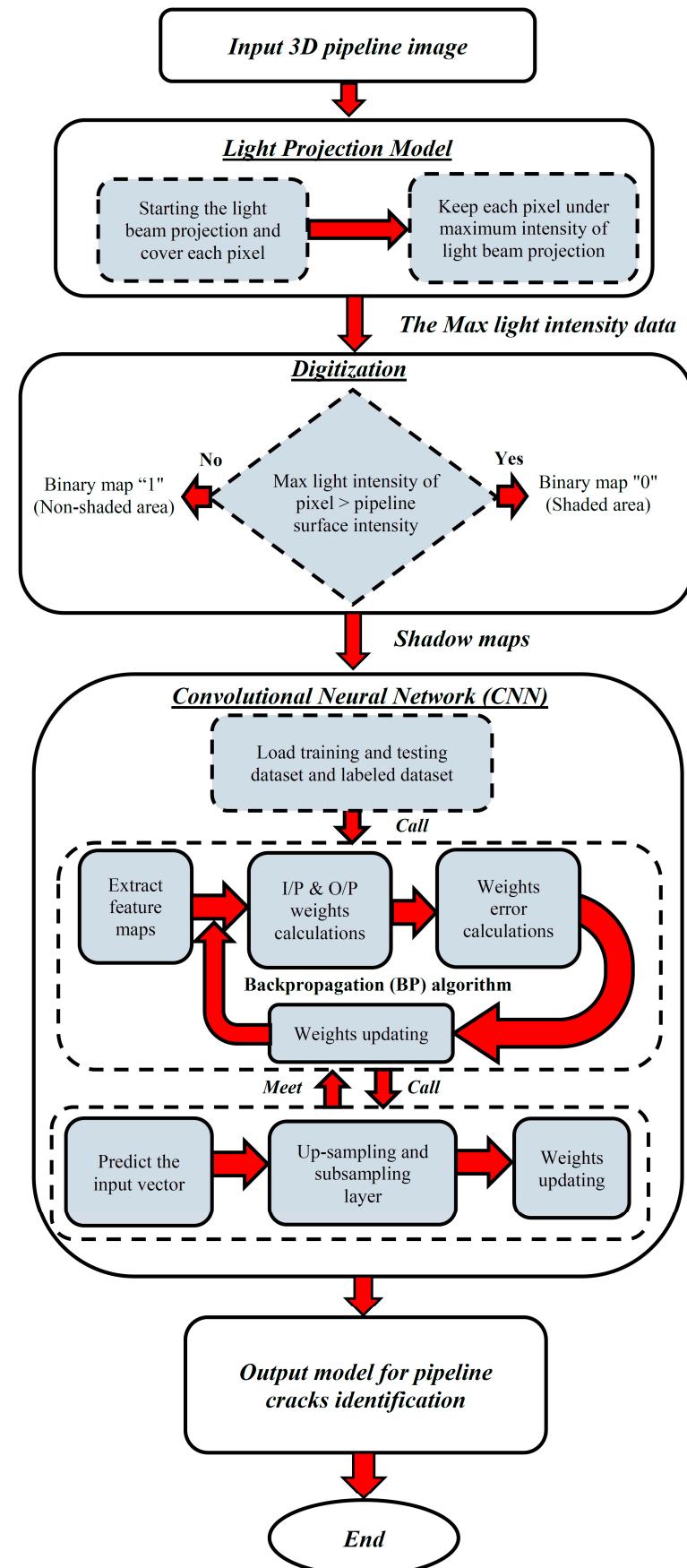


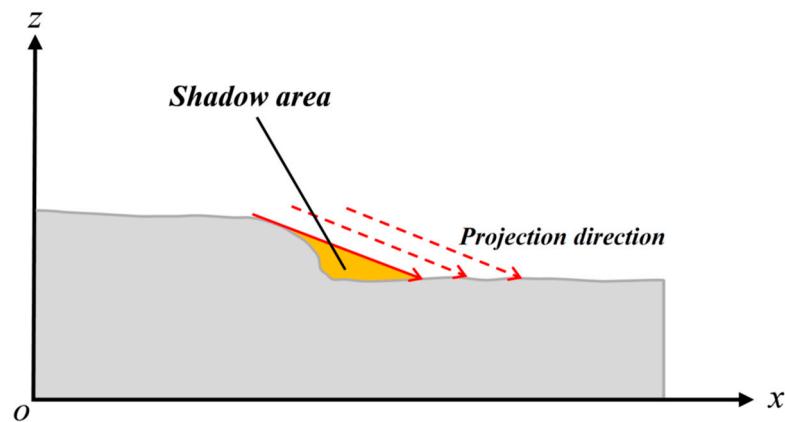
Figure 1. Flowchart of the proposed method.

3. Three-Dimensional (3D) Shadow Modeling

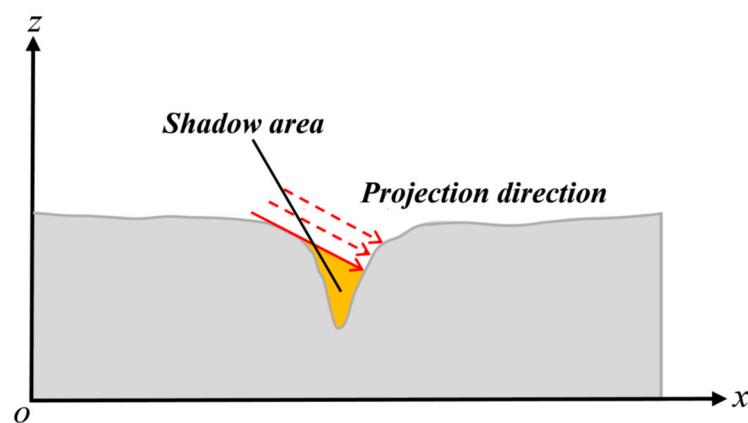
The proposed method implements the feature of crack shadows that are projected due to the height differences between cracks and pipeline surfaces. The identification of cracks is carried out through analysis of the shapes of the shadows. This method is developed under the following assumptions:

1. The projected light source (in this work, the high intensity light torches) is infinite and projected light beams are parallel;
2. The diffusion, reflection and refraction of light are ignorable.

According to these assumptions, both corrosion slopes and corrosion cracks can generate shadow areas under the lighting of the projection light source, as shown in Figure 2. On the other hand, bidirectional light projection provides an effective way to capture shadows caused by corrosion cracks and prevents interference from corrosion slopes. Figure 3 shows the schematic plots of the induced shadow area under bidirectional projection. Since cracks consist of continuous partial grooves, we utilize bidirectional light projection to prepare the pipeline images for crack identification.



(a) Corrosion slope.



(b) Corrosion crack.

Figure 2. Shadow under monodirectional projection.

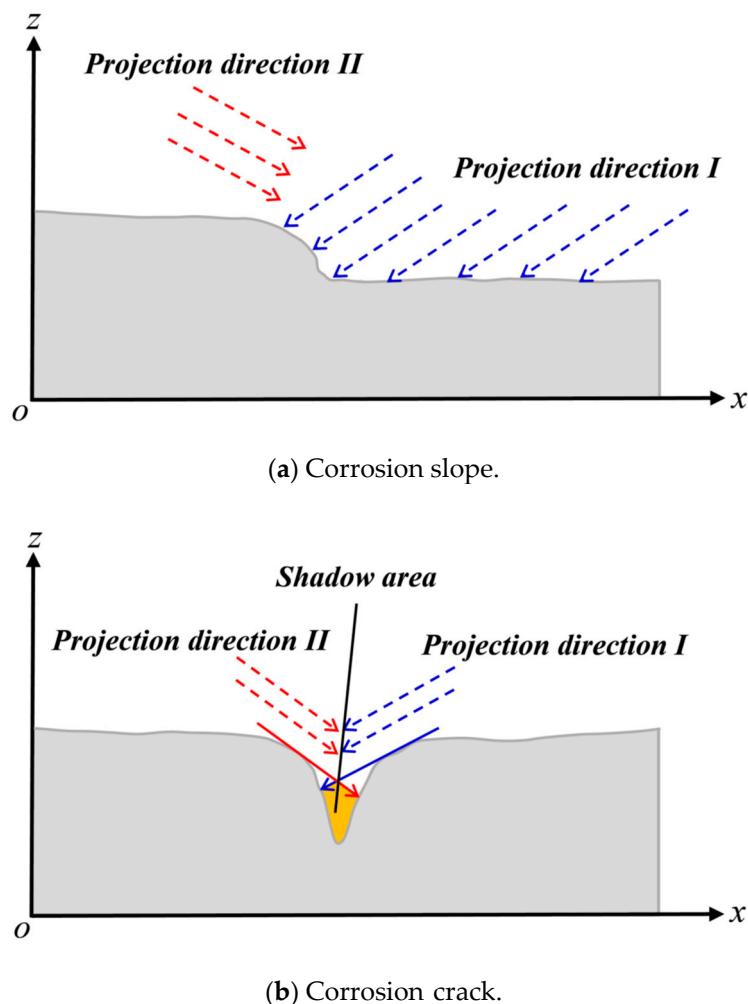


Figure 3. Shadow under bidirectional projection.

Under the bidirectional projection light, the size of the shadow area depends on the projection direction. Therefore, the projection angle (θ) is an important parameter. When the angle is 90° , the shadow areas will not be generated, but when the angle is 0° , infinitely long shadow areas will be generated. By changing the angle of projection direction from 0° to 90° , the 3D-SM can simulate a variety of shadow settings for the cracks on the pipeline's surface. Besides, if the selected projection angle satisfies the condition of generating the shadow areas of a small crack, it can also generate the shadow areas for other cracks. The main advantage of the 3D-SM simulation is that it can identify various cracks and even micro cracks by only adjusting a single variable for control (projection angle). Figure 4 presents the approach used to define the projection angle (θ) using projection of two light beams.

The optimal bidirectional projection direction should be perpendicular to the direction of the plane of the crack. However, actual cracks often have various directions, so the optimal bidirectional projection direction is not always the same. To reduce the calculation time, we use two representative pairs of bidirectional projection directions for the 3D-SM simulation, namely lateral projection (x -direction) and longitudinal projection (y -direction). The geometry settings of these two projections are shown in Figure 5. The lateral projection is used to identify the longitudinal cracks with the crack angle between approximately 45° and 90° and the longitudinal projection is used to identify the transverse cracks with the crack angle between approximately 0° and 45° .

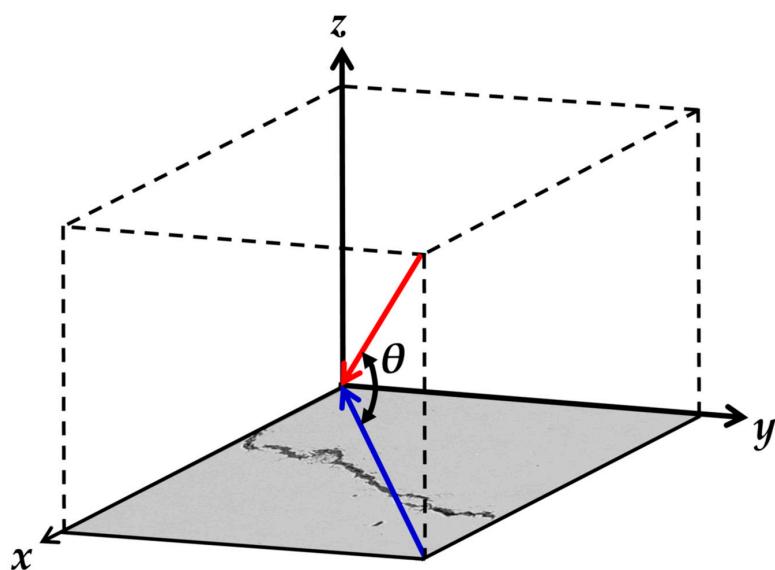
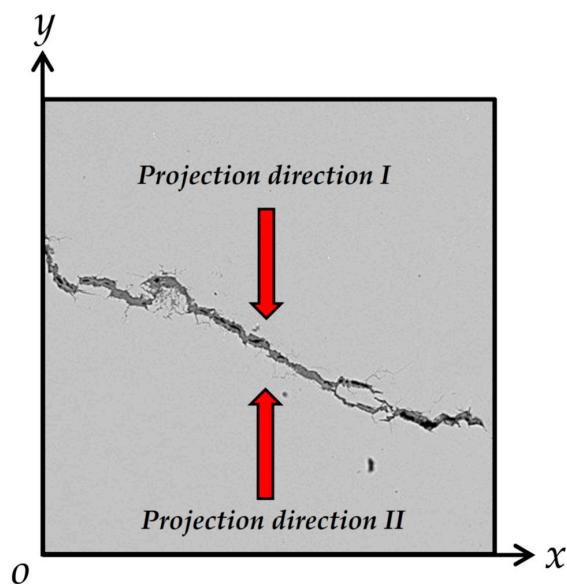


Figure 4. Illustration of projection angle (θ).

In the next step, the original 3D image will be converted into to a binary map to reflect the shadow pattern, where “0” indicates the shaded area, and “1” indicates the non-shaded area. A typical binary map is shown in Figure 6. Let S denote the light projection, P denote the 3D pixel of shadow formed upon a fall the light projection S and denote L denote the unit vector to indicate the direction of light. The sample point along the light projection can be expressed as:

$$S_k = P + \frac{k \cdot \Delta S}{\cos \theta} L, \quad (1)$$

where S_k is the k th sample point along the light projection S . The resolution of 3D pixel of shadow depends on the sampling interval ΔS (mm). A smaller sampling interval provides a higher resolution but consumes more computational resources. In this paper, we choose $\Delta S = 0.5$ so that the resolution of 3D pixel of shadow is sufficient to consider all pixels in the light projection.



(a) Lateral projection.

Figure 5. Cont.

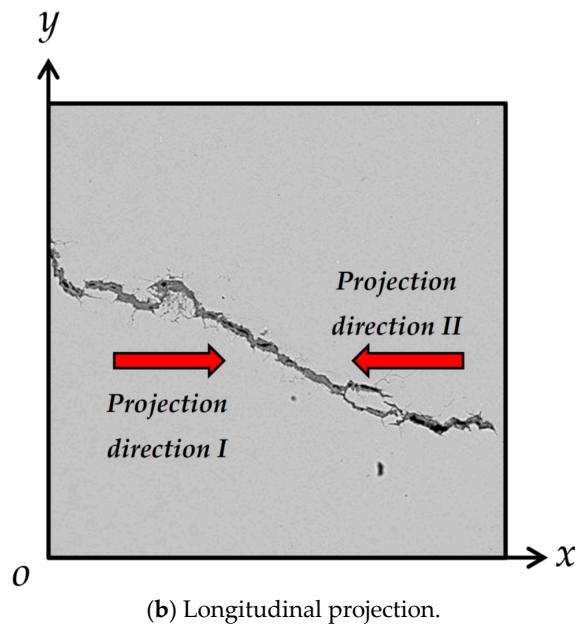


Figure 5. Lateral and longitudinal projections.

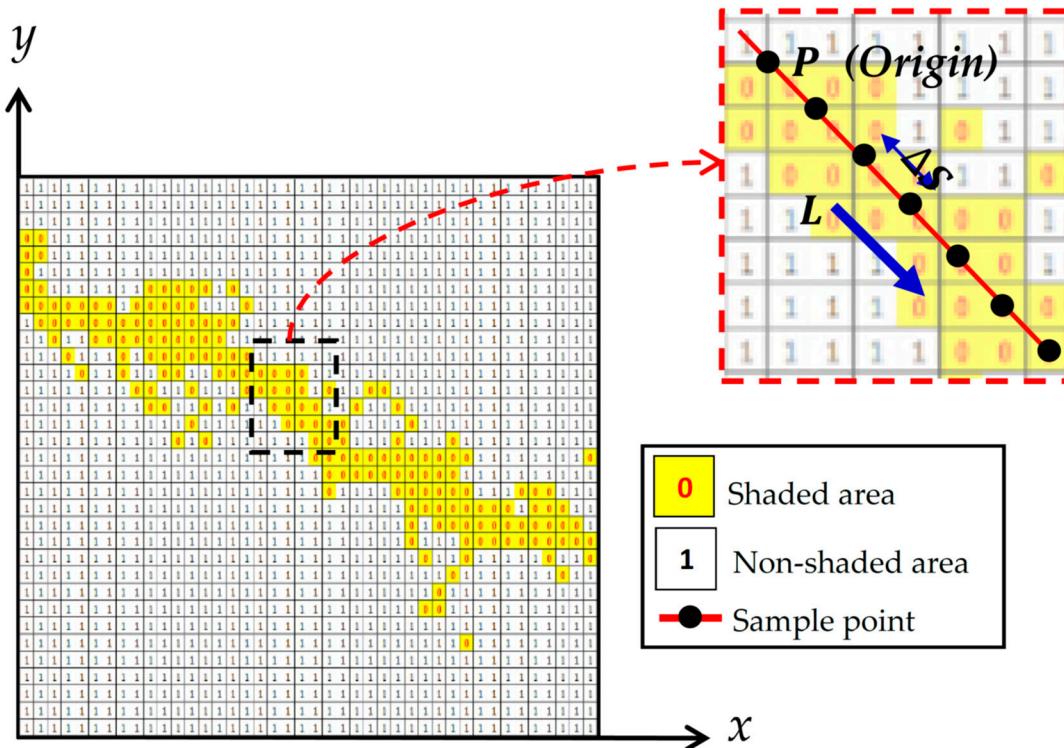


Figure 6. Illustration of tracing a single light projection.

Figure 7 shows the shadow map of a 3D crack image under the lateral and longitudinal projection.

To obtain the final integrated shadow map, the shadow maps under lateral and longitudinal projection are arranged by:

$$I_c(i, j) = \begin{cases} 1, & \text{if } B_c(i, j) = 0 \text{ or } \overline{B_c}(i, j) = 0 \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where $I_c(i, j)$ is the integrated shadow map; $B(i, j)_c$ is the binary value of the point (i, j) in the horizontal shadow map; $\bar{B}_c(i, j)$ is the binary value of the point (i, j) in the vertical shadow map; and (i, j) is the binary value of the integrated shadow map.

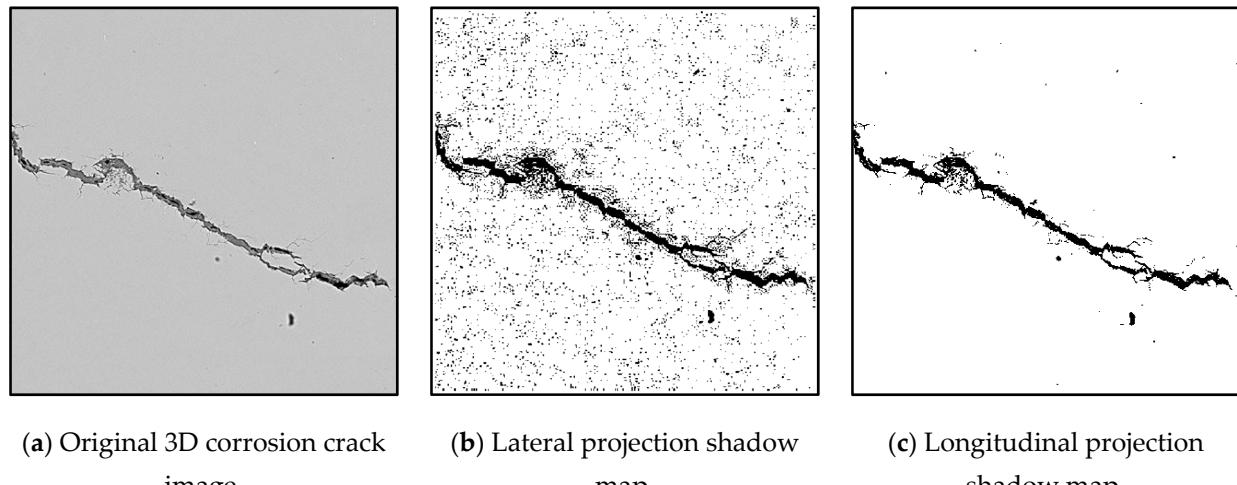


Figure 7. Shadow maps under lateral and longitudinal projections.

The integrated shadow map is noise corrupted. Therefore, the next step of the 3D-SM is noise elimination using the connected domain analysis [44]. The connected domain analysis makes use of the continuity and geometry properties of corrosion cracks. In general, a concerned crack has a continuous linear shape, and its length is much greater than its width. Hence, interconnected shadow points in the integrated shadow map are classified into groups for crack inspection and scattered shadow points are considered as noise to be eliminated. After the noise elimination, crack extraction from shadow groups based on that crack characteristics is carried out. For the corrosion cracks on the pipeline's surface, with which we are presently concerned, they are in a slender linear shape, in which their height is much less than that of the pipeline's surface. Moreover, their width is relatively uniform in a small range. According to these crack characteristics, the linear pattern analysis technique is used to filter shadow groups. If a shadow group does not fulfill the any of the following three criteria, it will be considered as noise to be eliminated.

First, to ensure that the shadow group has a slender linear pattern, a shape aspect ratio is defined:

$$P_i = \sum_{j=1}^{N_i-1} \frac{|W_{i,j+1} - W_{i,j}|}{W_{i,j}(N_i - 1)} \leq R_P \quad (3)$$

where $W_{i,j}$ is the width of the j th section of shadow group i ; N_i is the total number of profiles of shadow group i ; and R_P is the threshold value of the average profile width difference rate, $R_P = 1.0$.

Finally, we must check the angle between the current direction and the next direction in each section of the shadow group. Each direction is described a vector from the center of previous section to the center of current section of the shadow group. We use these vectors to calculate the average trend difference value of the shadow group by Equations (4)–(6).

$$V_{i,j} = P_{t_{i,j}} - P_{t_{i,j-1}} \quad (4)$$

$$\tilde{V}_{i,j} = P_{t_{i,j+1}} - P_{t_{i,j}} \quad (5)$$

$$T_i = \sum_{j=2}^{N_i-1} \frac{\arccos\left(\frac{V_{i,j}\tilde{V}_{i,j}}{\|V_{i,j}\|\times\|\tilde{V}_{i,j}\|}\right)}{N_i - 2} \quad (6)$$

$$T_i \leq R_t \quad (7)$$

where $P_{t_{ij}}$ is the center point of the section of shadow group i ; $V_{i,j}$ is the current direction of the third section of shadow group i ; $\tilde{V}_{i,j}$ is the next direction of the j th section of shadow group i ; N_i is the total number of profiles of shadow group i ; and R_t is the average direction difference threshold, $R_t = \pi/12$.

To ensure that an extracted crack has slender linear patterns, we use the thresholds given in Equations (3), (4) and (7) to control each quantified index. When a shadow group does not fulfill the above criteria, it is considered as a noise group to be eliminated. In this study, the threshold values are assigned to be noise. Figure 8 demonstrates an example for noise shadow groups cleaning. Several shadow groups were considered as noise groups because they did not fulfill the criteria. It is obvious that the resultant shadow map shown in Figure 8c filtered unnecessary scatter shadow groups and depicted a clearer crack pattern.

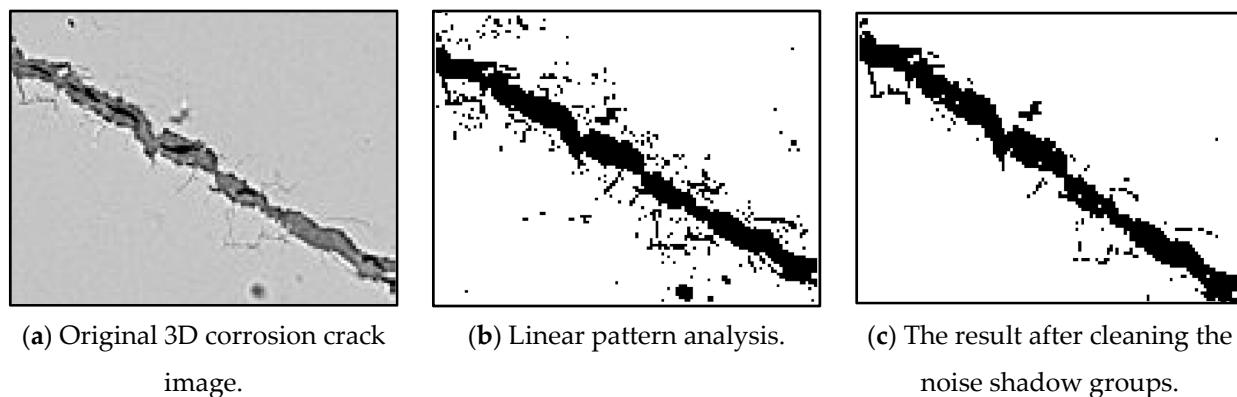


Figure 8. Demonstration of noise shadow group cleaning.

4. Convolutional Neural Network (CNN)

In this section, the detailed design of a CNN for corrosion crack identification of pipelines using the 3D-SM images is introduced. A CNN consists of alternating convolution with sub-sampling operations and a general multi-layer fully connected network as the last layer. Compare with other types of neural network models, the sub-sampling layers and convolutional layers improve the configural and spatial invariance of the neural network as well as increasing its computational efficiency.

In a CNN, the convolution layer is used to extract the representative local features of the previous layer, and the sub-sampling layer is used to reduce the complexity of the network by combining similar features. Therefore, one of the most useful features of deep learning is its ability to use the outputs of intermediate layers as another group of data. These data can be considered as the learned features through network adaptive learning. The learned features can be subsequently used for similarity comparison. The parameters of the CNN can be learned effectively using the training data.

4.1. Convolution Layers and Sub-Sampling Layers

A convolution converts the pixels of the input image in its receptive field into a single value. A convolution layer of a CNN applies a convolution operation to the input and transfers the result to the next layer. The convolutional operation combines the multiple input feature maps with convolutional kernels and outputs the new feature maps before activation [19]. In the proposed method, this process is defined as:

$$x_j^l = f \left(\sum_{i \in M_j} x_i^{l-1} * k_{ij}^l + b_j^l \right) \quad (8)$$

where $*$ denotes the (2D) convolution operation; x_i^{l-1} is the i th output map in layer $l-1$; k_{ij}^l is the kernel weight; b_j^l is the bias; M_j is a set of input feature maps; $f(\cdot)$ is a nonlinear function that is applied component-wise.

To train the CNN model using the BP algorithm, we need to obtain the gradients of each layer. In the CNN model, a down-sampling layer $l+1$ and convolution layer l follow each other. For sensitivity evaluation, the interested nodes in the current layer l are connected to the unit nodes of the next layer $l+1$, and both layer's connections are associated with weights defined at layer $l+1$. For example, a down-sampling layer map has weights β and we use the previous result scaled by β to calculate β^l . Each map j repeats the same computation in the convolutional layer as well as in the sub-sampling layer. The sub-sampling layer is defined as:

$$\delta_j^l = \beta_j^{l+1} \left(f' \left(u_j^l \right) \circ up \left(\delta_j^{l+1} \right) \right) \quad (9)$$

where \circ denotes the pointwise product of vectors.; u_j^l is the layer's gradient; $f'(\cdot)$ is a typical convolutional transformation; $up(\cdot)$ is an operation of up-sampling which tiles each pixel in the input horizontally and vertically n times in the output if the subsampling layer subsamples by a factor of n . Moreover, the gradients regarding bias b_j and kernel weight k_{ij}^l are described as:

$$\frac{\partial E}{\partial b_j} = \sum_{u,v} (\delta_j^l)_{uv} \quad (10)$$

$$\frac{\partial E}{\partial k_{ij}^l} = \sum_{u,v} (\delta_j^l)_{uv} (\mathbf{p}_i^{l-1})_{uv} \quad (11)$$

where $(\mathbf{p}_i^{l-1})_{uv}$ is the patch in x_i^{l-1} multiplied by k_{ij}^l via elementwise operation to calculate the element at (u, v) in the output map x_j^l in the convolutional layer.

For a sub-sampling layer, the down-sampled version of the input maps has the form:

$$x_j^l = f \left(\beta_j^l \text{down}(x_j^{l-1}) + b_j^l \right) \quad (12)$$

where $\text{down}(\cdot)$ is a sub-sampling operation.

Similar as the convolution layers, the weights and bias parameters of the sub-sampling layers are updated using gradients. The gradients of the sub-sampling layers with respect to the bias b_j and kernel weight k_{ij}^l can be described as:

$$\frac{\partial E}{\partial b_j} = \sum_{u,v} (\delta_j^l)_{uv}, \quad (13)$$

$$\frac{\partial E}{\partial \beta_j} = \sum_{u,v} (\delta_j^l \circ d_j^l)_{uv} \quad (14)$$

4.2. Learning Combinations of Feature Maps

Let α_{ij} be the weight of input map i and output map j . The output map j can be given as:

$$x_j^l = f \left(\sum_{i=1}^{N_{in}} \alpha_{ij} (x_i^{l-1} * k_i^l) + b_j^l \right) \quad (15)$$

The output map j is subject to the constraints:

$$\sum_i \alpha_{ij} = 1 \text{ and } 0 \leq \alpha_{ij} \leq 1 \quad (16)$$

The softmax operation derivative can be written as:

$$\frac{\partial \alpha_k}{\partial c_i} = \delta_{ki} \alpha_{ij} - \alpha_{ij} \alpha_k \quad (17)$$

The error derivative of α_i and c_i is expressed by:

$$\frac{\partial E}{\partial \alpha_i} = \frac{\partial E}{\partial u^l} \frac{\partial u^l}{\partial \alpha_i} = \sum_{u,v} \left(\delta^l \circ \left(x_i^{l-1} * k_i^l \right) \right)_{uv} \quad (18)$$

By applying the chain rule, we can calculate the network error gradient with respect to the implicit weights c_i :

$$\frac{\partial E}{\partial c_i} = \sum_k \frac{\partial E}{\partial \alpha_k} \frac{\partial \alpha_k}{\partial c_i} = \alpha_i \left(\frac{\partial E}{\partial \alpha_i} - \sum_k \frac{\partial E}{\partial \alpha_k} \alpha_k \right) \quad (19)$$

4.3. Enforcing Sparse Combinations

Assume that the sparseness constraints of the weight distribution α_i is a combination of a regularization penalty term $\Omega(\alpha_i)$ and the final error function. The regularization penalty $\Omega(\alpha_i)$ satisfies:

$$\frac{\partial \Omega}{\partial \alpha_i} = \lambda \text{sign}(\alpha_i) \quad (20)$$

$$\frac{\partial \Omega}{\partial c_i} = \sum_k \frac{\partial \Omega}{\partial \alpha_k} \frac{\partial \alpha_k}{\partial c_i} = \lambda \left(|\alpha_i| - \alpha_i \sum_k |\alpha_k| \right) \quad (21)$$

Figure 9 shows an example to demonstrate the processes of convolution. Herein, the input data are composed of a $7 \times 7 \times 3$ dataset, which means the network consists of 7×7 (width \times height) pixels with three color channels (blue, red and green).

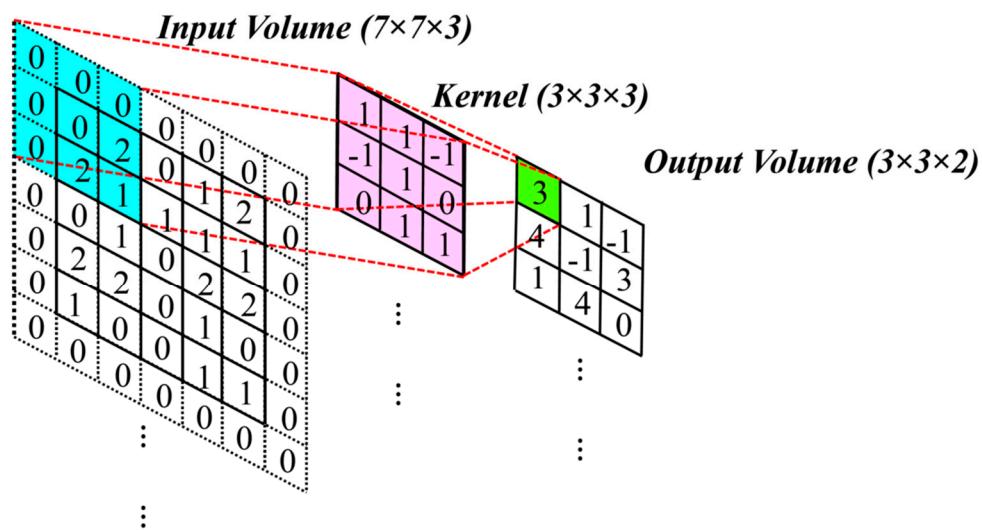


Figure 9. Convolution operation.

Figure 10 demonstrates the process of max pooling, which is defined as taking the maximum value of the specific data window area. There is another pooling method in the CNN model, known as average pooling, which involves taking the average value instead of the maximum value. In this paper, we applied the max pooling.

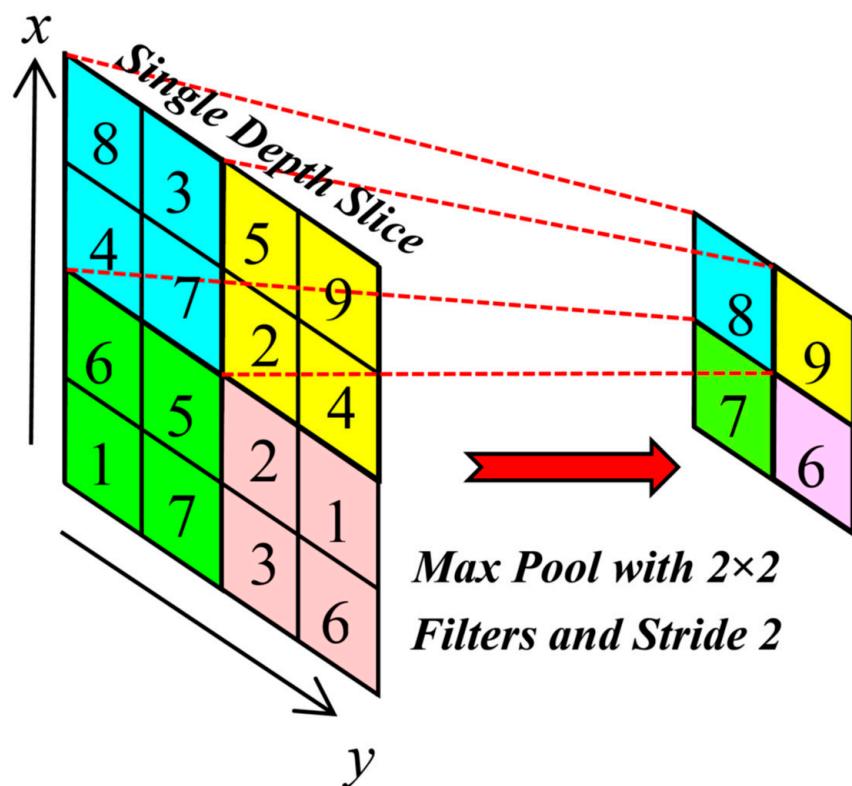


Figure 10. Max pooling operation.

5. Results and Discussions

The proposed 3D image based automatic corrosion crack identification method utilizes a 3D light and shadow model to convert the complex 3D crack images into binary shadow maps. Then, the proposed method carries out denoising of the image and extracts cracks from the shadow map. The difference between the traditional 2D grayscale images and 3D crack images is that many pipelines that suffer damages, such as cracking and corrosion, have distinctive characteristics in the third dimension, which all are irretrievable in 2D images. By feeding the dataset of 3D shadow maps to a trained CNN, the shaded areas can be automatically marked in red.

5.1. 3D Crack Images Database

A total of 900 raw pipeline crack images with sizes of 256×256 pixels were used. Among the 900 raw images, 700 randomly chosen raw images were used as the training dataset and the remaining 200 raw images were used as the testing dataset. As shown in Section 3, the projection angle (horizontal and vertical projection angle) is the only parameter that controls the proposed method. We determine that the optimal horizontal and vertical projection angles are 38° and 76° , respectively, by repeating tests. Figure 11 shows some representative raw training images and the associated resultant shadow map obtained by the proposed 3D-SM. It is found that the some of the raw images had disturbing backgrounds and conditions. Therefore, cleaning the interference noise is challenging and essential for the identification of cracks.

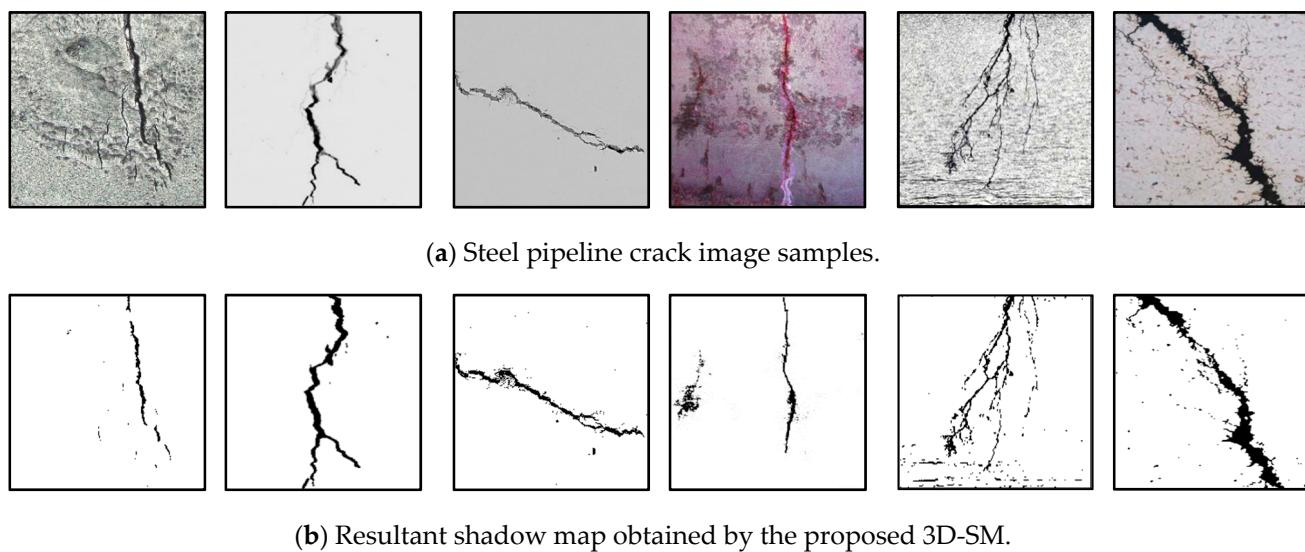


Figure 11. Steel pipeline crack images and 3D-SM results for training and testing of the CNN model.

5.2. Pipeline Corrosion Cracks Identification Using CNN

In the above sections, 3D-SM were studied to identify cracks in the pipeline that were suffering from corrosion. However, due to the 3D-SM having a lot of noise information effects and the model's inherent complexity during the analysis process, and the fact that it often ignores some cracks during the denoising processes, a more efficient way to extract the features of the 3D-SM distribution is employed for the identification of cracks. The schematic diagram of CNN configurations of the setup, training and testing model can be shown in Figure 12. The CNN setup, training and testing model functions are presented in Table 1.

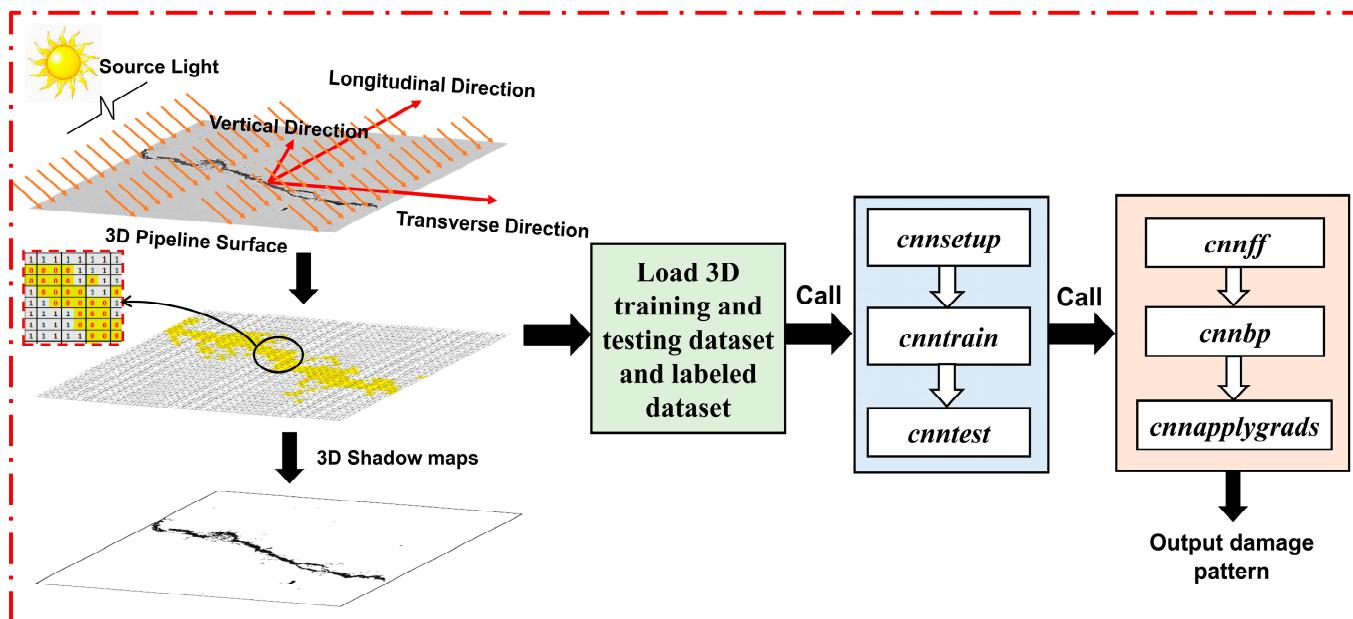
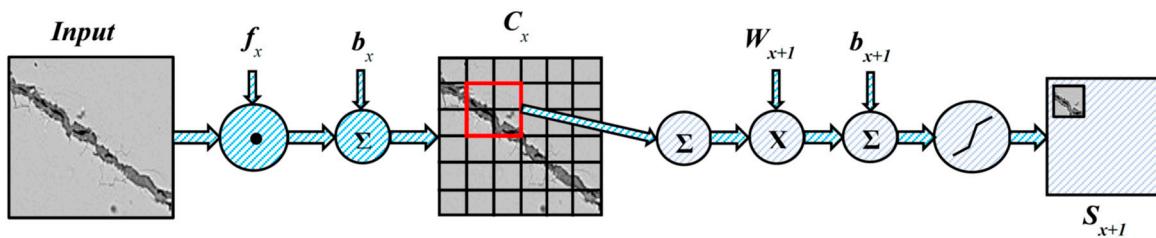


Figure 12. CNN configurations of the setup, training and testing model.

Table 1. The CNN setup, training and testing model's functions.

<i>Cnnsetup</i>	The Cnnsetup is used to set up the feature maps. The kernel window is composed of elements with the size of kernelsize \times kernelsize. Each element is an independent weight.
<i>cnntrain</i>	The cnntrain is used to reorder the sample, and randomly train and calculate the weights of network input and output. Then, the derivative of the error is calculated with respect to weights by the back propagation (BP) algorithm.
<i>Cnnff</i>	The Cnnff uses the neural network to predict the input vector. The samples are first reordered and then randomly trained.
<i>cnnbp</i>	The cnnbp layer is used for up-sampling and subsampling layer is used for down-sampling.
<i>Cnnaplygrads</i>	The Cnnaplygrads is used for updating of the weight depending on training, testing, labeled training, and labeled testing datasets.

In this work, the CNN consists of two convolutional layers, two sub-sampling layers, and a full connection layer. Detailed settings of each layer including the kernel size, number of tuned parameters and number of connections are included in this section. The architecture of the convolutional layer and sub-sampling layer is depicted in Figure 13. Herein, the x th layer with convolutional operation is denoted as C_x and the sub-sampling operation is denoted as S_x .

**Figure 13.** Sub-sampling and convolutional layer connection.

As shown in Table 2, C_1 is the first convolutional layer composed of six feature maps. By doing the convolutional operation, the feature of the original signal can be enhanced and the effects of the interference noise can be reduced. Each neuron of the feature map is connected with a 12×12 pixel neighborhood of input images. The feature map size is 168×168 pixels. C_1 has 580 tuned parameters (there are four filters in total and each filter has 12×12 unit parameter weights and a bias parameter, totaling $(12 \times 12 + 1) \times 4 = 580$ tuned parameters). One kernel is used between the input and C_1 and it turns out that there are $580 \times (168 \times 168) = 16,369,920$ connections in total. S_2 is a sub-sampling layer for compressing the image while retaining useful information. Two 72×72 feature maps are used. Each unit of the feature map is connected to a 6×6 neighborhood of C_1 . S_2 has $(1 + 1) \times 2 = 4$ tuned parameters and $(6 \times 6 + 1) \times 2 \times (72 \times 72) = 383,616$ connections. The next two layers, C_3 and S_4 , have similar architectures. The kernel sizes of C_3 and C_4 are 12×12 and 6×6 , respectively, and the output layer is composed of Euclidean radial basis function units. Images that are fed into the network as the input and the output can be obtained by $O_p = F_n(\dots(F_2(F_1(X_p W^{(1)})W^{(2)}))\dots W^{(n)})$, where X_p is the input matrix; $W^{(i)}$ is a matrix of shape $l_x \times l_{x-1}$; and $F_i(\cdot)$ is the activation functions at layer i . The error between the output O_p and desired output Y_p can be calculated by the BP algorithm.

Figure 14 shows some of the crack detection results obtained from the proposed method. In particular, the subplots on the upper row show the input raw images, the subplots on the middle row show the obtained shadow maps obtained from the 3D-SM, and the bottom row shows the automatic crack identification results obtained from the CNN. It is found that the shadow maps successfully depicted the representative features of the corrosion cracks and filtered the insignificant shadow groups. Moreover, the trained

CNN provided satisfactory performance in detecting the locations and evolution of the corrosion cracks.

Table 2. Detailed settings of the CNN for crack identification.

Layer	Kernel Size	Number of Parameter	Number of Connection
C1	12×12	580	16,369,920
S2	6×12	4	383,616
C3	12×12	/	/
S4	6×6	/	/

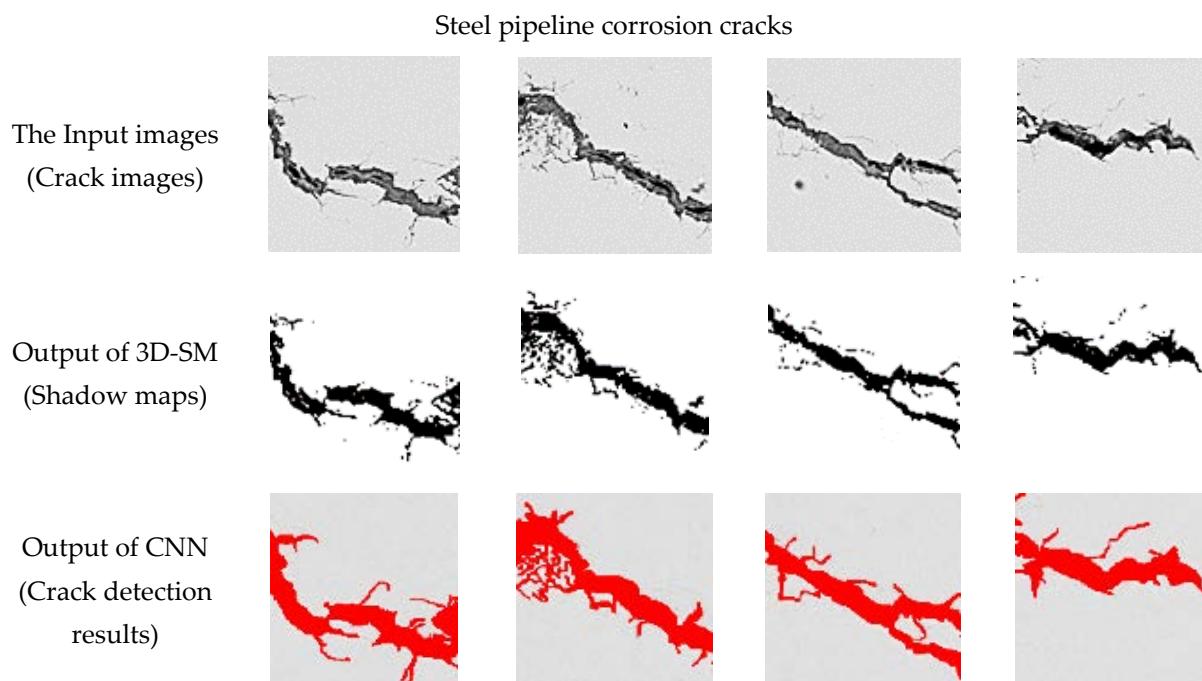


Figure 14. Identification results for pipeline corrosion cracks.

5.3. Evaluation of Accuracy and Reliability of the Present Algorithm

To quantify the performance of the proposed crack identification method, the accuracy rate, regression rate, and F-score were calculated throughout the training process. The accuracy rate is defined as the percentage of correctly classified pixels in all detected pixels in 3D shadow maps; the regression rate is defined as the percentage of correctly classified cracked pixels out of all actual cracked pixels; and F-score is defined as a measure of the training accuracy [45,46].

To verify the effectiveness and accuracy of the resultant CNN in pipeline corrosion crack identification, 200 testing images were employed and 81 selected crack samples are utilized for computing the accuracy and regression rate. Figure 15 and Table 3 show the crack identification results for the 81 crack samples. The testing samples are divided into 10 groups of 3D shadow maps, denoted as G_1, G_2, \dots, G_{10} . Four indexes that include the true-positive rate (TPR), true-negative rate (TNR), false-positive rate (FPR), and false-negative rate (FNR) are computed. The definitions of these four indexes are:

True-positive rate (TPR): The element originally belongs to the target class but it is also correctly identified and marked as the target class;

True-Negative rate (TNR): The element does not belong to the target class but it is also correctly identified and marked as a non-target class;

False-positive rate (FPR): The element does not belong to the target class but it is incorrectly identified and marked as the target class;

False-negative rate (FNR): The element originally belonged to the target class but was incorrectly identified and marked as a non-target class.

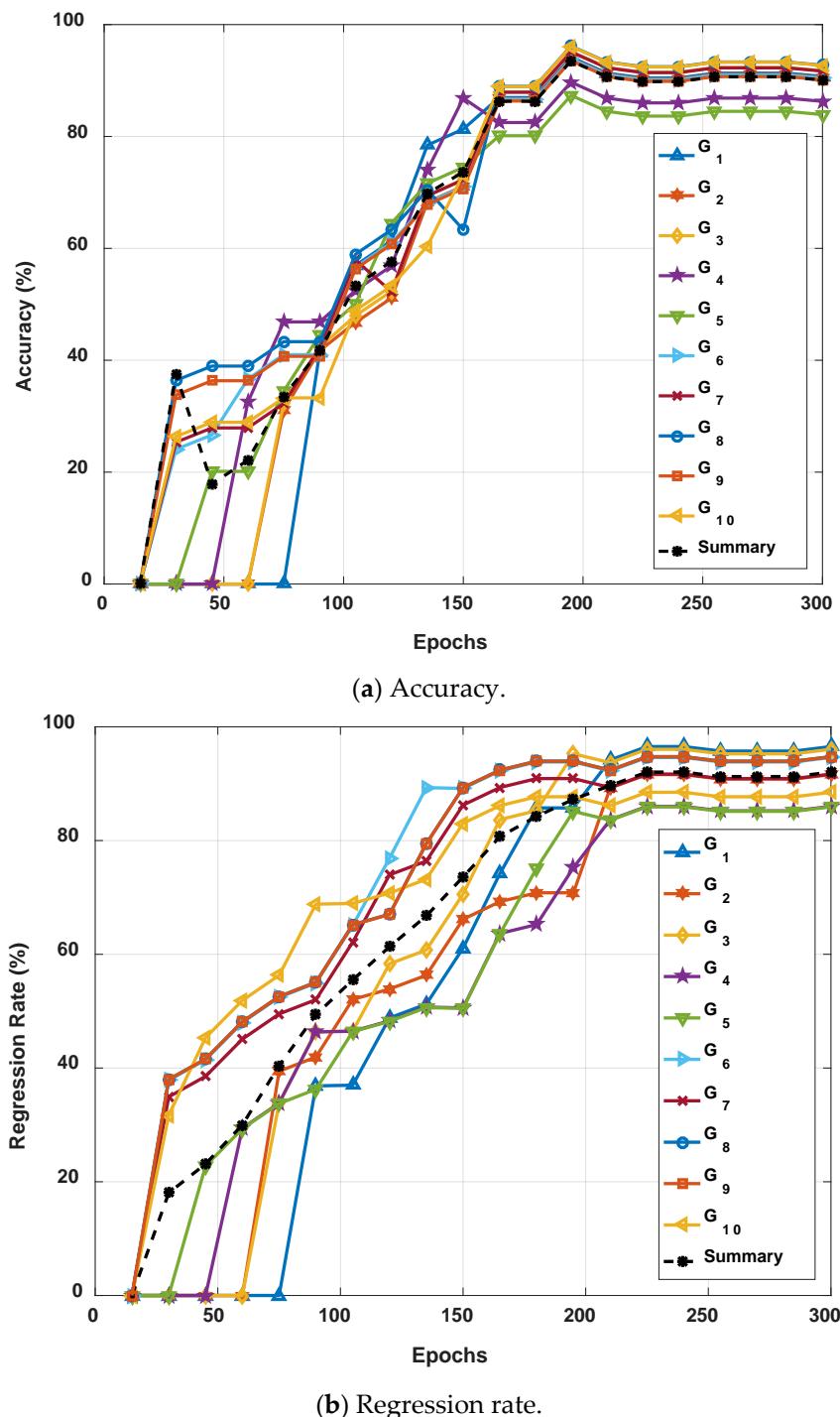


Figure 15. Cont.

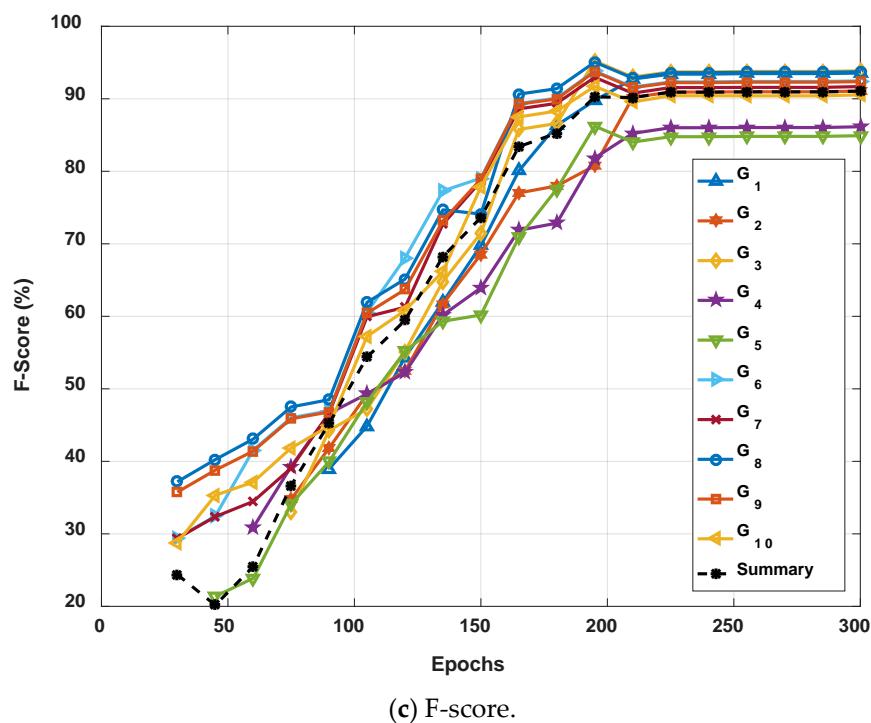


Figure 15. The training proceed comparison based on the crack detection test data for the 10 groups of 3D shadow maps.

Table 3. Comprehensive evaluation results for automatic crack detection algorithm.

Group No.	Image Number Range	Accuracy (%)	Regression Rate (%)	F-Score (%)
1	0 ~ 7	94.17	96.53	85.4
2	8 ~ 15	93.97	91.61	85.27
3	16 ~ 23	95.19	96.05	90.28
4	24 ~ 31	89.71	86.02	90.18
5	32 ~ 39	87.35	85.93	91.92
6	40 ~ 47	93.86	94.61	94.47
7	48 ~ 55	95.12	91.69	92.96
8	56 ~ 63	96.18	94.76	93.85
9	64 ~ 71	93.56	94.73	94.39
10	72 ~ 80	96.14	88.49	93.06
Summary	0 ~ 80	93.53	92.04	91.18

Based on the definitions of TPR, TNR, FPR, and FPR, the accuracy rate (P), regression rate (R) and F-score (F) can be determined:

$$P = \frac{N_{TP}}{N_{TP} + N_{FP}} \quad (22)$$

$$R = \frac{N_{TP}}{N_{TP} + N_{FN}} \quad (23)$$

$$F = \frac{2N_{TP}}{2N_{TP} + N_{FN} + N_{FP}} \quad (24)$$

where N_{TP} is N_{FP} and N_{FN} are the number of true-positive, false-positive, and false-negative elements, respectively.

Figure 15 shows the resultant accuracy, regression rate and F-score versus the number of epochs (an epoch refers to one cycle through the full training dataset). The results of the ten groups of 3D shadow maps (i.e., G_1, G_2, \dots, G_{10}) are indicated with solid

lines with different markers, while the overall performances of all 80 selected shadow maps is indicated with dotted lines. From the plots, it can be concluded that G_3 has the highest order of accuracy and regression rate—this means it has the highest detection accuracy for cracks—whereas G_5 has the lowest order of accuracy and regression rate, and the other groups' accuracy and regression rates are in between these values. In general, the differences in performance curves between the ten groups of 3D shadow maps are due to the differences in noise in each group. Nevertheless, the overall performance of the proposed method is satisfactory. The accuracy, regression rate and F-score of the overall performance are 93.53%, 92.04% and 91.18%, respectively. The result confirmed that the proposed method can automatically identify corrosion cracks on pipelines with satisfactory performance regardless of the image backgrounds and conditions.

In engineering applications, most studies focus on using artificial neural networks to process images taken from structures such as pipelines, bridges, and tunnels. However, even when we use deep learning models such as CNN, this method usually does not have sufficient generalization to reflect the fact of structural health status. By combining the 3D-SM with CNN, the 3D shadow maps' information can be extracted and then fed into CNN for training and testing. This combination will be a useful tool to solve actual crack identification problems of complex structures in mechanical and civil engineering.

6. Conclusions

In this paper, we proposed an automatic crack identification method to detect corrosion cracks of pipelines using 3D images. The proposed method integrates 3D shadow modeling (3D-SM) and deep learning theory for reliable and efficient analysis. Since the depth of a corrosion cracks is below its surrounding areas, a shadow of the corrosion crack can be projected under light sources. Consequently, 3D shadow images of the pipeline's outer surface provide information on the corrosion cracks. Three-dimensional shadow maps were built to reflect the actual status of structures. Then, the proposed method analyzed the shape of the shadow to identify the pipeline's cracks. To avoid losing the necessary information on evolving cracks during the denoising process, an efficient means of extracting the features of 3D shadow maps is used. A CNN was developed to process the 3D shadow maps obtained from the 3D-SM. By feeding the feature maps to the CNN, the evolving pipeline cracks were identified regardless the disturbing backgrounds and conditions of the original images. The results of this pipeline corrosion crack identification process showed that the automatic crack identification method achieved satisfactory performance with high stability, accuracy and regression rate.

Author Contributions: W.A.A., the first author, was responsible for the establishing and applying the new methodology introduced in this work, which combined 3D Shadow Modeling with a Deep Learning algorithm for Crack Identification, and carried out the majority of the research project including the writing of the manuscript. Both analytical derivations and statistical methods were closely tested and confirmed by M.N., who suggested the procedures that were utilized. S.-C.K., R.G. and T.W. worked closely with W.A.A. in the reviewing and editing phase. As a leading scholar and pioneer in SHM and fiber optic sensors, Z.W. offered useful guidance and recommendations that contributed greatly to this research project. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available corrosion cracking images were analyzed in this study. These images can be found here: <https://www.shutterstock.com/search/>, accessed on 18 June 2021.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

$(i:j)$	The binary value of the integrated shadow map
$B_c(i,j)$	The binary value of the point (i,j) in the horizontal shadow map
$\bar{B}_c(i,j)$	The binary value of the point (i,j) in the vertical shadow map
$I_c(i,j)$	The integrated shadow map
K_i	The aspect ratio of the shadow group
L_i	The total length of the shadow group
\bar{W}_i	The average width of the shadow group
R_k	The aspect ratio threshold, $R_k = 10$
P_i	The average profile width difference rate of shadow group i
N_i	The total number of profiles of shadow group i
$W_{i,j}$	The width of the j th section of shadow group i
R_P	The threshold value of the average profile width difference rate, $R_P = 1.0$
$V_{i,j}$	The current direction of the third section of shadow group i
$\tilde{V}_{i,j}$	The next direction of the j th section of shadow group i
$P_{t,i,j}$	The center point of the section of shadow group i
T_i	The center point of shadow group i
R_t	The average direction difference threshold, $R_t = \pi/12$
M_j	A set of input feature maps.
b	The bias for each output feature map.
n	The subsampling layer number.
$up(\cdot)$	An operation of up-sampling which tiles each pixel in the input horizontally and vertically n times in the output if the subsampling layer subsamples by a factor of n .
b_j	The bias.
k_{ij}^l	The kernel weight.
$(p_i^{l-1})_{uv}$	The patch in x_i^{l-1} multiplied by k_{ij}^l via the element-wise procedure to calculate the element at (u, v) in the output map x_j^l in the convolutional layer.
x_j^l	The input maps in the convolutional layer.
$down(\cdot)$	A sub-sampling procedure.
β and b	The bias parameters.
P	The accuracy rate
R	The regression rate
F	F-score
3D-SM	Three-dimensional Shadow Modeling
CNN	Convolutional Neural Network
SHM	Structural Health Monitoring
I/P	Input
O/P	Output
N_{TP}	The number of true-positive elements
N_{FP}	The number of false-positive elements
N_{FN}	The number of false-negative elements

References

1. Altabey, W.A. FE and ANN model of ECS to simulate the pipelines suffer from internal corrosion. *Struct. Monit. Maint.* **2016**, *3*, 297–314. [[CrossRef](#)]
2. He, Y. *Corrosion Monitoring, Reference Module in Materials Science and Materials Engineering*; Science Direct: Amsterdam, The Netherlands, 2015.
3. Cole, I.S.; Marney, D. The science of pipe corrosion: A review of the literature on the corrosion of ferrous metals in soils. *Corros. Sci.* **2012**, *56*, 5–16. [[CrossRef](#)]
4. Alwis, L.; Sun, T.; Grattan, K.T.V. Optical fibre-based sensor technology for humidity and moisture measurement: Review of recent progress. *Measurement* **2013**, *46*, 4052–4074. [[CrossRef](#)]
5. Dong, S.; Liao, Y.; Tian, Q. Intensity-based optical fiber sensor for monitoring corrosion of aluminum alloys. *Appl. Opt.* **2005**, *44*, 5773–5777. [[CrossRef](#)]
6. Altabey, W.A.; Noori, M. Detection of Fatigue Crack in Basalt FRP Laminate Composite Pipe using Electrical Potential Change Method. *J. Phys. Conf. Ser.* **2017**, *842*, 012079. [[CrossRef](#)]
7. Altabey, W.A. Delamination evaluation on basalt FRP composite pipe by electrical potential change. *Adv. Aircr. Spacecr. Sci.* **2017**, *4*, 515–528. [[CrossRef](#)]

8. Altabey, W.A.; Noori, M.; Alarjani, A.; Zhao, Y. Tensile creep monitoring of basalt fiber-reinforced polymer plates via electrical potential change and artificial neural network. *Sci. Iran. Int. J. Sci. Technol. Trans. Mech. Eng. B* **2020**, *27*, 1995–2008. [[CrossRef](#)]
9. Altabey, W.A. EPC method for delamination assessment of basalt FRP pipe: Electrodes number effect. *Struct. Monit. Maint.* **2017**, *4*, 69–84. [[CrossRef](#)]
10. Altabey, W.A.; Mohammad, N. Monitoring the water absorption in GFRE pipes via an electrical capacitance sensors. *Adv. Aircr. Spacecr. Sci.* **2018**, *5*, 411–434. [[CrossRef](#)]
11. Altabey, W.A.; Noori, M.; Alarjani, A.; Zhao, Y. Nano-Delamination Monitoring of BFRP Nano-Pipes of Electrical Potential Change with ANNs. *Adv. Nano Res.* **2020**, *9*, 1–13. [[CrossRef](#)]
12. Zhao, Y.; Noori, M.; Altabey, W.A.; Wu, Z. Fatigue damage identification for composite pipeline systems using electrical capacitance sensors. *Smart Mater. Struct.* **2018**, *27*, 085023. [[CrossRef](#)]
13. Kumar, R.; Ismail, M.; Zhao, W.; Noori, M.; Yadav, A.R.; Chen, S.; Singh, V.; Altabey, W.A.; Silik, A.I.H.; Kumar, G.; et al. Damage detection of wind turbine system based on signal processing approach: A critical review. *Clean Technol. Environ. Policy* **2021**, *23*, 561–580. [[CrossRef](#)]
14. Wang, H.; Zhu, N.; Wang, Q. Segmentation of pavement cracks using differential box-counting sppmach. *J. Harbin Inst. Technol.* **2007**, *39*, 142–144.
15. Martin, D.R.; Fowlkes, C.C.; Malik, J. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Trans. Pattern Anal. Mach. Intell.* **2004**, *26*, 530–549. [[CrossRef](#)]
16. Stahl, J.S.; Wang, S. Edge grouping combining boundary and region information. *IEEE Trans. Image Process.* **2007**, *16*, 2590–2606. [[CrossRef](#)]
17. Ghiasi, R.; Ghasemi, M.R.; Noori, M.; Altabey, W. A non-parametric approach toward structural health monitoring for processing big data collected from the sensor network. Structural Health Monitoring 2019: Enabling Intelligent Life-Cycle Health Management for Industry Internet of Things (IIOT). In Proceedings of the 12th International Workshop on Structural Health Monitoring (IWSHM 2019), Stanford, CA, USA, 10–12 September 2019. [[CrossRef](#)]
18. Ghiasi, R.; Noori, M.; Altabey, W.A.; Silik, A.; Wang, T.; Wu, Z. Uncertainty Handling in Structural Damage Detection via Non-Probabilistic Meta-Models and Interval Mathematics, a Data-Analytics Approach. *Appl. Sci.* **2021**, *11*, 770. [[CrossRef](#)]
19. Zhao, Y.; Noori, M.; Altabey, W.A.; Ghiasi, R.; Wu, Z.; Noori, M. Deep Learning-Based Damage, Load and Support Identification for a Composite Pipeline by Extracting Modal Macro Strains from Dynamic Excitations. *Appl. Sci.* **2018**, *8*, 2564. [[CrossRef](#)]
20. Wang, T.; Altabey, W.A.; Noori, M.; Ghiasi, R. A Deep Learning Based Approach for Response Prediction of Beam-Like Structures. *Struct. Durab. Health Monit.* **2020**, *14*, 315–388. [[CrossRef](#)]
21. Kost, A.; Altabey, W.A.; Noori, M.; Awad, T. Applying Neural Networks for Tire Pressure Monitoring Systems. *Struct. Durab. Health Monit.* **2019**, *13*, 247–266. [[CrossRef](#)]
22. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [[CrossRef](#)]
23. Bouvrie, J. *Notes on Convolutional Neural Networks*; Center for Biological Learning, Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology: Cambridge, MA, USA, 2006.
24. Altabey, W.A. Applying deep learning and wavelet transform for predicting the vibration behavior in variable thickness skew composite plates with intermediate elastic support. *J. Vibroeng.* **2021**. [[CrossRef](#)]
25. Altabey, W.A. Free vibration of basalt fiber reinforced polymer (FRP) laminated variable thickness plates with intermediate elastic support using finite strip transition matrix (FSTM) method. *J. Vibroeng.* **2017**, *19*, 2873–2885. [[CrossRef](#)]
26. Altabey, W.A. Prediction of natural frequency of basalt fiber reinforced polymer (FRP) laminated variable thickness plates with intermediate elastic support using artificial neural networks (ANNs) method. *J. Vibroeng.* **2017**, *19*, 3668–3678. [[CrossRef](#)]
27. Altabey, W.A. High performance estimations of natural frequency of basalt FRP laminated plates with intermediate elastic support using response surfaces method. *J. Vibroeng.* **2018**, *20*, 1099–1107. [[CrossRef](#)]
28. Altabey, W.A. Vibration analysis of laminated composite variable thickness plate using finite strip transition matrix technique. In *MATLAB Verifications MATLAB—Particular for Engineer*; Bennett, K., Ed.; IntechOpen: Monterey, CA, USA, 2014; Volume 21, pp. 583–620, ISBN 980-953-307-1128-8. [[CrossRef](#)]
29. Abdeljaber, O.; Avci, O.; Kiranyaz, S.; Gabbouj, M.; Inman, D.J. Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. *J. Sound Vib.* **2017**, *388*, 154–170. [[CrossRef](#)]
30. Zhao, Y.; Noori, M.; Altabey, W.A. Damage detection for a beam under transient excitation via three different algorithms. *Struct. Eng. Mech.* **2017**, *63*, 803–817. [[CrossRef](#)]
31. Zhao, Y.; Noori, M.; Noori, M.; Awad, T. A Comparison of Three Different Methods for the Identification of Hysterically Degrading Structures Using BWBN Model. *Front. Built Environ.* **2019**, *4*, 80. [[CrossRef](#)]
32. Noori, M.; Haifegn, W.; Altabey, W.A.; Silik, A.I.H. A Modified Wavelet Energy Rate Based Damage Identification Method for Steel Bridges. *Sci. Iran. Int. J. Sci. Technol. Trans. Mech. Eng. B* **2018**, *25*, 3210–3230. [[CrossRef](#)]
33. Zhao, Y.; Noori, M.; Beheshti-Aval, S.B. Mode shape-based damage identification for a reinforced concrete beam using wavelet coefficient differences and multiresolution analysis. *Struct. Control. Health Monit.* **2017**, *25*, e2041. [[CrossRef](#)]
34. Silik, A.; Noori, M.; Altabey, W.A.; Ghiasi, R. Selecting optimum levels of wavelet multi-resolution analysis for time-varying signals in structural health monitoring. *Struct. Control. Health Monit.* **2021**, in press. [[CrossRef](#)]

35. Silik, A.; Noori, M.; Altabey, W.A.; Ghiasi, R.; Wu, Z. Comparative Analysis of Wavelet Transform for Time-Frequency Analysis and Transient Localization in Structural Health Monitoring. *Struct. Durab. Health Monit.* **2021**, *15*, 1–22. [CrossRef]
36. Silik, A.; Noori, M.; Altabey, W.; Dang, J.; Ghiasi, R.; Wu, Z. Optimum wavelet selection for nonparametric analysis toward structural health monitoring for processing big data from sensor network: A comparative study. *Struct. Health Monit.* **2021**, in press. [CrossRef]
37. Silik, A.; Noori, M.; Altabey, W.A. Analytic Wavelet Selection for Time—Frequency Analysis of Big Data Form Civil Structure Monitoring. In *Civil Structural Health Monitoring, Proceedings of CSHM-8 Workshop 2021*; Rainieri, C., Fabbrocino, G., Caterino, N., Eds.; Springer: Berlin/Heidelberg, Germany, 2021; Chapter 29. [CrossRef]
38. Liang, Y.; Wu, D.; Liu, G.; Li, Y.; Gao, C.; Ma, Z.J.; Wu, W. Big data-enabled multiscale serviceability analysis for aging bridges. *Digit. Commun. Netw.* **2016**, *2*, 97–107. [CrossRef]
39. Wang, T.; Noori, M.; Altabey, W.A. Identification of cracks in an Euler—Bernoulli beam using Bayesian inference and closed-form solution of vibration modes. *Proc. Inst. Mech. Eng. Part L J. Mater. Des. Appl.* **2021**, *235*, 421–438. [CrossRef]
40. Wang, T.; Noori, M.; Altabey, W.A.; Li, Z. Parameter Identification and Dynamic Response Analysis of a Modified Prandtl-Ishlinskii Asymmetric Hysteresis Model via Least-Mean Square algorithm and Particle Swarm Optimization. *Proc. IMechE Part L J. Mater. Des. Appl.* **2021**. [CrossRef]
41. Li, Z.; Noori, M.; Zhao, Y.; Wan, C.; Feng, D.; Altabey, W.A. A multi-objective optimization algorithm for Bouc-Wen-Baber-Noori model to identify re-inforced concrete columns failing in different modes. *Proc. IMechE Part L J. Mater. Des. Appl.* **2021**. [CrossRef]
42. Zhao, Y.; Noori, M.; Altabey, W.A.; Reaching Law Based Sliding Mode Control for a Frame Structure under Seismic Load. *Earthq. Eng. Eng. Vib.* **2021**. In press. Available online: <https://www.springer.com/journal/11803> (accessed on 18 June 2021).
43. Zhao, Y.; Noori, M.; Altabey, W.A.; Lu, N. Reliability Evaluation of a Laminate Composite Plate Under Distributed Pressure Using a Hybrid Response Surface method. *Int. J. Reliab. Qual. Saf. Eng.* **2017**, *24*, 1750013. [CrossRef]
44. Grama, A.; Gupta, A.; Karypis, G.; Kumar, V.; Grama, A. *Introduction to Parallel Computing*, 2nd ed.; Addison Wesley: Boston, MA, USA, 2003; pp. 209–210.
45. Davis, J.; Goadrich, M. The relationship between precision-recall and ROC curves. In *Proceeding of the 23rd International Conference on Machine Learning, Pittsburgh, PA, USA, 25–29 June 2006*; ACM: New York, NY, USA, 2006; pp. 233–240.
46. Fawcet, T. An introduction to ROC analysis. *Pattern Recognit. Lett.* **2006**, *27*, 861–874. [CrossRef]