TECHNICAL REPORT STANDARD PAGE

1. Title and Subtitle
   Smart Bridge Monitoring Employing Deep Learning and Unmanned Aerial Vehicles
2. Author(s)
   Hadi Salehi, Ph.D., Assistant Professor
3. Performing Organization Name and Address
   Department of Civil Engineering and Construction Engineering Technology, Louisiana Tech University
   Ruston, LA, 71272
4. Sponsoring Agency Name and Address
   Louisiana Department of Transportation and Development
   P.O. Box 94245
   Baton Rouge, LA 70804-9245
5. Report No.
   FHWA/LA.24/24-2TIRE
6. Report Date
   August 2024
7. Performing Organization Code
   LTRC Project Number: 24-2TIRE
   SIO Number: DOTLT1000497
8. Type of Report and Period Covered
   Final Report
   07/2023 – 06/2024
9. No. of Pages
   42
10. Supplementary Notes
    Conducted in Cooperation with the U.S. Department of Transportation, Federal Highway Administration.
11. Distribution Statement
    Unrestricted. This document is available through the National Technical Information Service, Springfield, VA 21161.
12. Key Words
    Smart bridge monitoring, deep learning, convolutional neural networks, long short-term memory, unmanned aerial vehicles, crack detection, concrete infrastructure
13. Abstract

Concrete is a widely used material in civil infrastructure and is known for its durability, strength, and versatility. Due to aging and degradation, regular monitoring of the physical and functional health of concrete structures is essential to ensure safety and serviceability. Concrete surfaces are prone to damage, including cracks, which are critical indicators of structural integrity. In recent years, computer vision techniques have gained more attention for the structural health monitoring of bridges and have shown accurate performance for structural condition assessment. Nonetheless, existing approaches suffer from several drawbacks, such as being computationally expensive and needing long training time, which present a hurdle to developing a real-time structural monitoring system. Additionally, traditional crack detection methods, relying on manual inspections, are limited by human error, time constraints, and accessibility issues. To tackle these challenges, this study proposed a smart monitoring platform employing deep learning and unmanned aerial vehicles (UAVs). A computer vision-based deep learning approach was developed to analyze images captured by UAVs from laboratory experiments on concrete

beams and field studies of two bridges. By integrating deep learning architectures, such as convolutional neural networks for spatial feature extraction and long short-term memory for temporal analysis, the proposed monitoring system achieved 99.5% accuracy. The results demonstrated the effectiveness of the proposed platform for improving the accuracy and efficiency of bridge health monitoring, with the potential for further enhancement through larger datasets and optimized UAV deployment.

Project Review Committee

Each research project will have an advisory committee appointed by the LTRC Director. The Project Review Committee is responsible for assisting the LTRC Administrator or Manager in the development of acceptable research problem statements, requests for proposals, review of research proposals, oversight of approved research projects, and implementation of findings.

LTRC appreciates the dedication of the following Project Review Committee Members in guiding this research study to fruition.

LTRC Administrator/Manager

Vijaya (VJ) Gopu, Ph.D., P.E.

Associate Director, External Programs

Directorate Implementation Sponsor

Chad Winchester, P.E.

DOTD Chief Engineer

Smart Bridge Monitoring Employing Deep Learning and Unmanned Aerial Vehicles

By
Hadi Salehi, Ph.D.
Assistant Professor

Department of Civil Engineering and Construction Engineering Technology
Louisiana Tech University
Ruston, LA, 71272

LTRC Project No. 24-2TIRE
SIO No. DOTLT1000497

conducted for
Louisiana Department of Transportation and Development
Louisiana Transportation Research Center

August 2024

Abstract

Concrete is a widely used material in civil infrastructures and is known for its durability, strength, and versatility. Due to aging and degradation, regular monitoring of the physical and functional health of concrete structures is essential to ensure safety and serviceability. Concrete surfaces are prone to damage, including cracks, which are critical indicators of structural integrity. In recent years, computer vision techniques have gained more attention for the structural health monitoring of bridges and have shown accurate performance for structural condition assessment. Nonetheless, existing approaches suffer from several drawbacks, such as being computationally expensive and needing long training time, which present a hurdle to developing a real-time structural monitoring system. In addition, traditional crack detection methods, relying on manual inspections, are limited by human error, time constraints, and accessibility issues. To tackle these challenges, this study proposed a smart monitoring platform employing deep learning and unmanned aerial vehicles (UAVs). A computer vision-based deep learning approach was developed to analyze images captured by UAVs from laboratory experiments on concrete beams and field studies of two bridges. By integrating deep learning architectures, such as convolutional neural networks for spatial feature extraction and long short-term memory for temporal analysis, the proposed monitoring system achieved 99.5% accuracy. The results demonstrated the effectiveness of the proposed platform for improving the accuracy and efficiency of bridge health monitoring, with the potential for further enhancement through larger datasets and optimized UAV deployment.

Acknowledgments

We acknowledge the support from the Louisiana Transportation Research Center (LTRC), Louisiana Department of Development and Transportation (DOTD), and the College of Engineering and Science at Louisiana Tech University.

Table of Contents

List of Tables

List of Figures

Introduction

Concrete is a fundamental construction material widely used in critical civil infrastructure, including bridges, buildings, dams, roads, and tunnels, due to its superior durability, strength, longevity, and versatility. Its application in civil infrastructure is widespread and continues to expand over time [1] [2]. Regular and reliable monitoring of the physical and functional health of concrete structures is essential due to aging, degradation, and fatigue load, to ensure safety and serviceability [3]. Concrete surfaces experience various types of damage, including cracks, corrosion, delamination, and spalling throughout their lifespans. Cracks are vital indicators of structural integrity, making them a key parameter in structural health monitoring. Conventional crack detection methods rely on point-by-point human visual inspection, which is limited by the inspector's experience, time constraints, restricted access to certain areas, high costs, labor intensity, and potential safety hazards for the inspector. These challenges necessitate the need to incorporate advanced automated technologies and adopt innovative non-contact approaches to enhance the overall efficiency of crack detection [4] [5].

To address the limitations of conventional crack detection, this research introduced an innovative computer vision-based method integrated with employing unmanned aerial vehicles (UAVs) to improve the accuracy and efficiency of detecting cracks in concrete civil infrastructure. Computer vision, a domain within artificial intelligence, encompasses advanced computational techniques to analyze and process data to identify patterns in images or videos collected by UAVs and detect anomalies for concrete structural assessment [6]. Identifying cracks in images faces challenges such as variation in crack appearance, including different sizes, shapes, and orientations, subtle and fine cracks, image noise, and complex backgrounds. Computer vision mitigates these challenges enabling automated analysis of images to achieve consistent, accurate, and efficient crack detection. Deep learning techniques, such as Convolutional Neural Networks (CNN), can be trained to process large volumes of images quickly to extract spatial features. These features can subsequently be fed into Long Short-Term Memory (LSTM) to analyze sequences in data or temporal dependencies, such as the progression of cracks over time or the correlation between crack formations [7] [8].

Many studies have explored the application of computer vision for crack detection in concrete infrastructure. Despite significant advancements in crack detection using computer vision techniques for concrete structures, several knowledge gaps remain in the current research. First, most of the studies focused on spatial feature extraction by CNN, overlooking different aspects of temporal analysis to understand crack progression over time. Secondly, many of the existing methods are time-consuming and computationally expensive, which often leads to a lack of scalability and real-time processing capabilities,

restricting their practical usage in large-scale monitoring. Additionally, some of the previous research employing single-method techniques fail to fully exploit the potential to integrate various computer vision architectures. Lastly, to train a deep learning model, a considerable number of images are required, and conventional inspection techniques are restricted by access limitations of manual inspection, which makes it a difficult task.

To overcome these challenges, this paper proposed a real-time computer vision-based technique to detect cracks in concrete infrastructure. By leveraging the innovative integration of the CNN algorithm with an LSTM architecture, the method effectively identifies cracks in beams and bridges, offering an automated solution that mitigates the time consumption and computational complexities associated with conventional methods. The novelty of this research stems from combining computer vision, deep learning, CNN, and LSTM architectures, along with the deployment of UAVs, to develop a robust system for automated data collection and sequential data classification. The proposed CNN-LSTM algorithm was trained and evaluated using image data from experimental studies on laboratory-cast beams subjected to shear-dominant loadings, as well as images captured during field study from two bridges in Minden, Louisiana.

The results of this study demonstrate the capabilities, limitations, and optimal conditions for UAV-assisted inspections of concrete surfaces, including beams and bridges to enhance accuracy and facilitate data collection for early crack detection. Furthermore, it validated the effectiveness and applicability of leveraging the CNN-LSTM deep learning framework to accurately detect and classify cracks. Overall, this paper contributes by introducing an advanced efficient structural health monitoring method for concrete infrastructure to reduce the failure rate and ensure the durability and reliability of critical civil infrastructure.

Literature Review

In recent years, the use of unmanned aerial vehicles (UAVs) equipped with vision sensors for bridge inspection has garnered significant attention in different countries due to its safety and reliability. Rakha et al. [9] conducted a comprehensive review of the use of Unmanned Aerial Systems (UAS) for building performance analysis and energy audits, proposing a standardized framework for UAS operation. Kim et al. [10] proposed an automated crack detection technique for concrete surfaces in on-site environments using a CNN trained with diverse image classes, including field images and real-time UAV video frames, thereby advancing the potential for replacing traditional visual inspections. Freimuth et al. [11] developed an integrated and automated UAV inspection workflow, utilizing a 3D planning environment to generate collision-free flight paths based on BIM data, validating the approach through a case study. Ding et al. [12] developed a UAV-based approach for the accurate detection and quantification of concrete cracks without reference markers, introducing an improved calibration method and an independent boundary refinement transformer (IBR-Former) for enhanced crack segmentation. Kim et al. [13] introduced a UAV-based crack identification system that integrates hybrid image processing with ultrasonic displacement sensing to accurately measure crack width and length in concrete structures.

Many studies have explored the application of computer vision for crack detection in concrete infrastructure. Solhmirzaei et al. [4] proposed an approach employing CNN architecture on images collected in an experiment to detect major and minor cracks in Ultra-high-performance Concrete (UHPC) members. Kim et al. [14] introduced a method to identify cracks on aging concrete bridges using region with CNN-based transfer learning (R-CNN) and measured the crack size with a square-shaped marker. Li et al. [15] presented an image-based crack detection method utilizing CNN by modifying AlexNet for concrete surfaces. Golding et al. [16] proposed a CNN architecture on 40,000 RGB pre-trained images with VGG16 for image processing of concrete surfaces. Cha et al. introduced methods using CNN and R-CNN to detect anomalies such as concrete cracks and steel delamination and corrosion [17] [18]. Dorafshan et al. [7] conducted a comparative analysis of edge detection algorithms and deep convolutional neural networks (DCNNs) for crack detection in concrete structures. Deng et al. [19] provided a comprehensive review of computer vision-based crack analysis methodologies, highlighting both qualitative and quantitative approaches for civil infrastructure. Dinh et al. [20] developed a computer vision-based method for automatic concrete crack detection using histogram thresholding to distinguish cracks from background noise.

Objective

Despite significant advancements in crack detection using computer vision techniques for concrete infrastructure, several knowledge gaps remain in the current research. First, most of the studies focused on spatial feature extraction by CNN, overlooking different aspects of temporal analysis to understand crack progression over time. Second, many of the existing methods are time-consuming and computationally expensive, which often leads to a lack of scalability and real-time processing capabilities, thereby restricting their practical usage in large-scale monitoring. Additionally, some of the previous research employing single-method techniques fail to fully exploit the potential of integrating various computer vision architectures. Lastly, to train a deep learning model a considerable number of images are required, and conventional inspection techniques are restricted by access limitations of manual inspection. which makes it a difficult task.

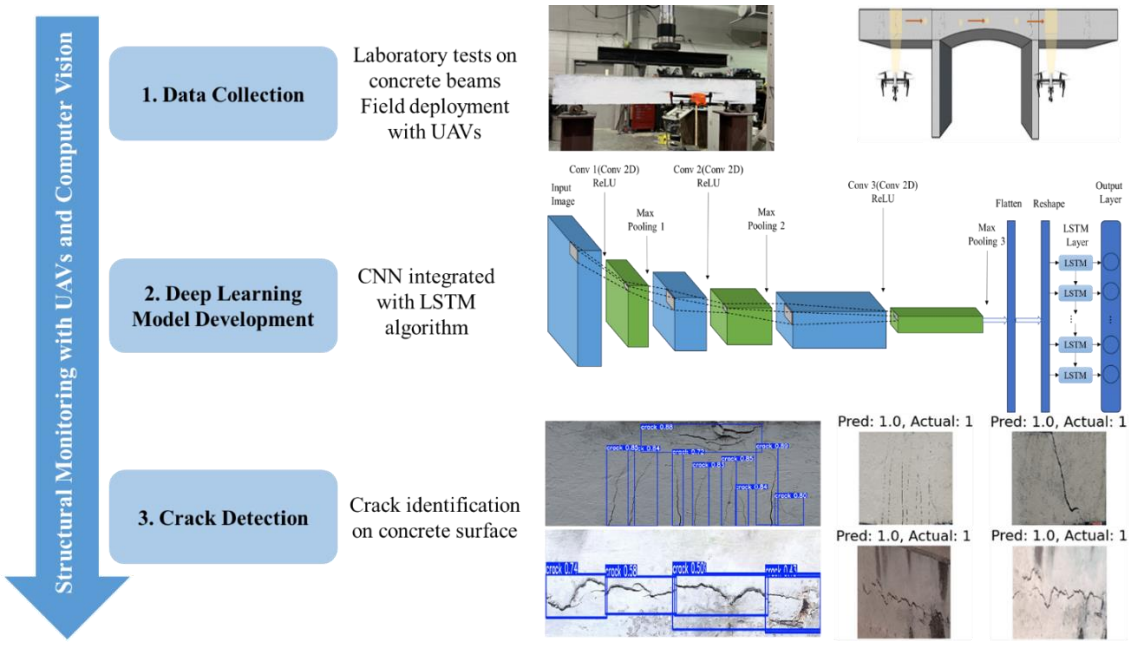
To overcome these challenges, this paper proposed a real-time computer vision-based technique to detect cracks in concrete infrastructure. By leveraging the innovative integration of the CNN algorithm with an LSTM architecture, the method effectively identifies cracks in beams and bridges, offering an automated solution that mitigates the time consumption and computational complexities associated with conventional methods. The novelty of this research stems from combining computer vision, deep learning, CNN, and LSTM architectures, along with the deployment of UAVs, to develop a robust system for automated data collection and sequential data classification. The proposed CNN-LSTM algorithm was trained and evaluated using image data from experimental studies on laboratory-cast beams subjected to shear-dominant loadings, as well as images captured during field study from two bridges in Minden, Louisiana.

Methodology

Figure 1 provides an overview of the proposed structural monitoring crack detection method using a CNN-LSTM model with UAV technology. The study comprises three distinct phases. The initial phase focuses on comprehensive image collection to serve as the dataset for the deep learning model. This phase is subdivided into two parts: (1) experimental studies and laboratory testing on reinforced concrete beams, and (2) field studies on two concrete bridges. In the laboratory, a UAV equipped with high-resolution cameras and sensors captured images of various crack conditions during shear-dominant loading tests, ranging from intact to fully cracked. In the field, the UAV gathered crack images from two bridges under real-world conditions.

The second phase focused on developing a robust and reliable deep learning model to process and analyze the collected data. This model leverages convolutional neural networks (CNNs) integrated with long short-term memory (LSTM) units to enhance its capabilities. A comprehensive explanation of CNN and LSTM, along with the implementation of the proposed model architecture, is provided in Section 2.2. The final phase concentrated on detecting and classifying cracks in concrete surfaces. Using the well-trained deep learning model on the collected images detected the presence and exact locations of cracks. This phase began with data preprocessing and model training, including hyperparameter tuning to optimize performance and achieve optimal results. The model classifies the dataset into two groups (cracked and uncracked), performs object detection, and highlights crack areas with rectangular bounding boxes.

Figure 1. Proposed smart bridge monitoring approach employing deep learning and UAV

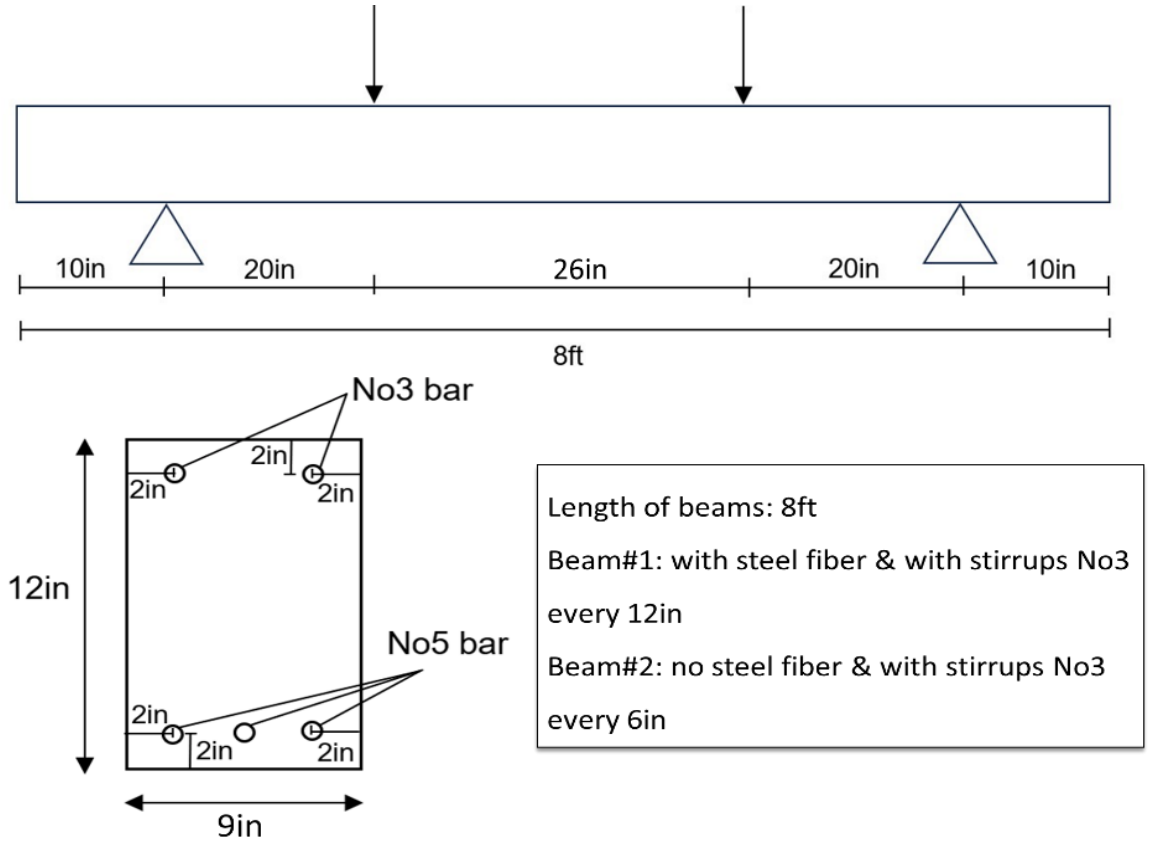


2.1. Data Collection

2.1.1. Experimental Study (Laboratory Testing)

The dataset for this study was derived from experiments performed in a laboratory setting by the authors. This involved the fabrication and testing of two reinforced concrete beams subjected to shear-dominant loading conditions. The experimental variables in casting two beams included variations in the composition of the concrete mix (utilizing steel fibers in one beam), sectional properties (such as the quantity and layout of stirrups), and the loading test setup. Each beam was fabricated with a high-strength concrete mix with differential spacing of stirrups along the longitudinal reinforcement bars. In one of the beams, steel fibers were incorporated into the mix along with longitudinal and transverse reinforcements. Both beams were in a rectangular cross-section, with a dimension of 8 feet in length, 9 inches in width, and 12 inches in depth. Illustrations of sectional dimensions and reinforced configurations of casted beams are depicted in Figure 2.

Figure 2. Beams layout and cross section for experimental tests



Response parameters, such as failure loads, failure patterns, and crack propagation were measured during tests and employed during the assessment of structural response for two beams under shear-dominant stress. The test setup was designed to evaluate the shear and flexural behavior of the beams with a four-point loading setup. This setup applied a dual-point loading on the beam's upper surface, executed via a displacement-controlled actuator (MTS machine with a capacity of 120 KN). Incremental loading was applied to the beams, and crack propagation image data were collected at different load levels utilizing a UAV and a stand-alone camera, both strategically placed two meters distance from the beam surface for optimal data capture. Data acquisition with a high-resolution camera of the UAV is discussed further in Section 2.1.3. Figure 3 displays the beam casting process, the experimental setup, and the data acquisition using the UAV. Furthermore, the sample of data collected via UAV during the experimental test is shown in Figure 4 as intact and cracked.

Figure 3. Data collection for experimental studies

(a) Reinforced concrete mold for casting beams

(b) Casting beams in Laboratory

(c) Test setup

(d) Data acquisition utilizing UAV

Figure 4. Illustration of intact and cracked concrete beams from experiments

(a) Intact (non-cracked)

(b) With crack

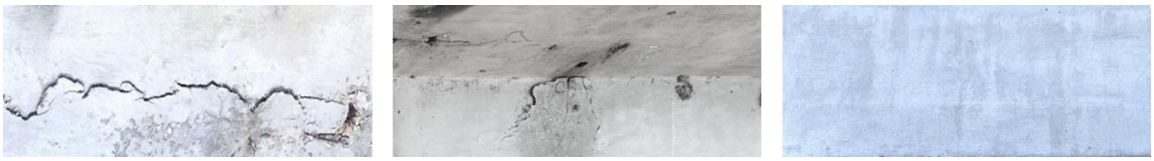
2.1.2. Field Study

The field study was conducted on two reinforced concrete bridges located in Minden, Louisiana. Data acquisition was carried out employing a UAV, operated manually by a certified UAV pilot from the Louisiana Department of Transportation and Development (DOTD). This ensured full compliance with safety regulations and operational guidelines. Additionally, necessary permits and registrations were obtained in advance to ensure public safety and adherence to legal requirements.

During the flight missions, the UAV was navigated around the lateral sides and undersides of the bridges. The pilot utilized remote control along with the UAV's obstacle avoidance technology to navigate around bridges efficiently. High-resolution images were captured at multiple points using the UAV’s onboard camera. These images were stored directly on the UAV’s internal storage card. The data collected during the field test is shown in Figure 5 as cracked and non-cracked. To ensure data accuracy and minimize data discrepancies, regular calibration of the UAV's sensors and cameras was conducted.

The operational duration of the UAV flight was constrained by its battery life, limiting each flight to approximately 20 minutes. This restriction resulted in a reduced coverage area per flight, necessitating multiple missions for data collection on the bridges. Different weather conditions such as extreme temperatures, wind, and rain also frequently disrupted flight schedules and prolonged the field study. Additionally, poor lighting conditions, areas inaccessible to UAVs, GPS signal obstructions, and environmental interferences such as traffic vibrations and local wildlife caused significant challenges for data collection.

Figure 5. Data collected from bridges using UAV



(a) With crack (b) With crack (c) Intact (non-cracked)

2.1.3. Image acquisition with a high-resolution camera on the UAVs

Datasets for both experimental and field studies were collected by UAV. The priority in utilizing UAVs was ensuring safety and reliable flight operation by following the guidelines. For this research, an EVO II Pro V3 UAV, equipped with a Sony 20-megapixel 1-inch CMOS image sensor camera, was employed (Figure 6). These components were connected via a gimbal manufactured by Autel pilot UAV. The gimbal allowed the camera to tilt from -130° to +45° and pan from -100° to +100°. The camera had a resolution of 5472 × 3076 pixels with a 16:9 aspect ratio. The UAV was manually operated during the flights, maintaining a distance of approximately 2 meters from the beams while scanning their surface and taking pictures. Nearly 5,000 images were captured for the beams, and approximately 120 images were taken for bridges, in both cracked and non-cracked conditions.

Figure 6. Autel Evo II pro V3 UAV used for data/image collection

2.2. Deep Learning Model Development

2.2.1. Convolutional Neural Network (CNN)

Convolutional Neural Network, known as CNN, is one of the most commonly used supervised deep learning methods, inspired by the biological mechanism of the animal visual cortex and mostly employed for image recognition, classification, and segmentation [21]. CNN is comprised of an input and output layer, along with several hidden layers, including convolutional layers, pooling layers, activation functions, and fully connected layers [22].

The convolutional layer, the fundamental component of CNN, performs convolutional operations to extract features of input data with a set of filters or convolutional kernels and generates feature maps. Each kernel is a small matrix that convolves with input data and slides over the input images with a fixed stride until all receptive fields are covered. The convolutional layer uses the same set of weights across the input and captures the local dependencies which reduces the number of model parameters and enhances the efficiency of the training network.

Following each convolutional layer, a pooling layer is employed to reduce the spatial dimensions of the feature maps, thereby decreasing the number of calculations and avoiding overfitting. The pooling layer's primary function is to select essential features, ensuring feature invariance and consequently lowering the number of parameters and the computational complexity. This down-sampling process contributes to better generalization and faster convergence of the model. Average pooling and max pooling are two commonly used pooling techniques. In max pooling, the highest value from a set of

neurons in the previous convolutional layer is selected for the new layer, while average pooling transmits the mean value of a set from the preceding layer to the next.

The final layer of the CNN model is fully connected, serving as a classifier by performing a series of nonlinear transformations on the feature map derived from the convolution and pooling operations to compute class scores and generate an output. Fully connected layers link each neuron in one layer to every neuron in the subsequent layer, similar to a traditional neural network. At the end, a flattened matrix is passed through the fully connected layer to classify the images.

After each convolution operation to introduce non-linearity into the model and improve the network's representation ability, non-linear activation functions are employed. Common activation functions include ReLU (Rectified Linear Unit), Sigmoid, and Tanh, as presented in equations 1, 2, and 3. ReLU is the most widely used as it helps to address the vanishing gradient problem and improves convergence speed during network training.

$$ReLU(x) = max(0, x) \quad [1]$$

$$\sigma(x) = \frac{1}{1 + exp(-x)} \quad [2]$$

$$tanh(x) = \frac{exp(x) - exp(-x)}{exp(x) = exp(-x)} \quad [3]$$

During the model's training process, the loss function is calculated by measuring the error between the model's output and the target output. This error is then backpropagated through the network to update the weights of the filters and neurons. The chain rule is employed to derive the gradients of the loss with respect to each weight. Weight optimization is performed using algorithms such as gradient descent or its variants (e.g., SGD, Adam) [21-24].

2.2.2. Long Short-Term Memory (LSTM)

The Long Short-Term Memory (LSTM) network is an enhanced version of standard recurrent neural networks (RNNs), specifically designed for capturing long-term dependencies and overcoming the gradient vanishing issue in RNNs. It achieves this by levering a gating mechanism and a memory unit. As a result, LSTM networks are more effective in processing time series data and long sequences, such as in time series forecasting, natural language processing, and speech recognition tasks [25]. Figure 2 depicts the structure of the LSTM unit employed in this study.

LSTM networks are comprised of two main unit structures (hidden unit h_t, and memory unit C_t), along with three gate structures: an input gate(i_t), a forget gate(f_t), and an output gate (o_t). The gating mechanism in LSTM networks enables the selection of input information and updates the state of the memory unit, thus facilitating the recording of long-term historical information and its transmission.

In an LSTM network, the cell state progresses along the entire sequence directly, with linear interactions, while the hidden state carries the output information to the next time step or other layers of the stacked LSTM structure. The forget gate's primary role is to decide which information from the previous cell state should be discarded, thus facilitating selective forgetting. Equation 4 shows the forget gate functions, where x_t is the current input, h_{t-1} is the previous hidden state, W_f is the weight of the forget gate, b_f is the bias, and σ is the sigmoid function.

$$f_t = \sigma\big(W_f . [h_{t-1}, x_t] + b_f\big) \quad [4]$$

The input gate regulates that the extent of the new value flows is incorporated into the cell state, determining what part of the new input is significant. In Equations 5 and 6, i_t is the input gate activation, C_t is the candidate cell state, and ReLU is the activation function.

$$i_t = \sigma(W_i . [h_{t-1} + b_i]) \quad [5]$$

$$C_t = ReLU(Wc . [h_{t-1} + b_c]) \quad [6]$$

In Updating Memory Unit State, the new cell state results from the combination of the old cell state and the new candidate cell state. Equation 7 illustrates the corresponding linear function.

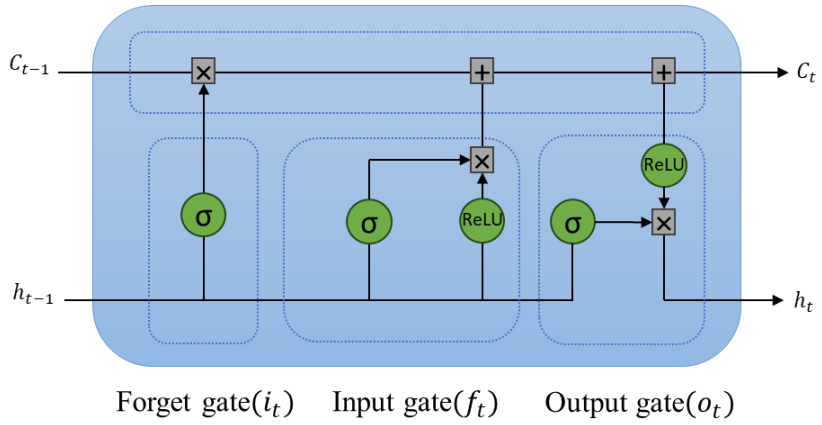
$$C_t = f_t . C_{t-1} + i_t . C_t \quad [7]$$

The output gate determines which part of the cell state to output, filtering the information and sending it as the LSTM cell's output. Equations 8 and 9 illustrate the corresponding functions which, o_t is the output gate activation and h_t is the new hidden state. Figure 7 depicts a schematic of the LSTM model used in this study [26]–[28].

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad [8]$$

$$h_t = o_t . ReLU(C_t) \quad [9]$$

Figure 7. The LSTM model architecture



2.2.3. Architecture of the Proposed CNN-LSTM Model

In this study, a combination of CNN and LSTM architecture was designed to leverage the strengths of both spatial feature extraction and temporal sequence modeling for the purpose of crack classification and object detection. The architecture of the CNN model consists of three convolutional layers, to extract high-dimensional spatial features of the input image dataset which are denoted as CNN features. The first convolutional layer employs 32 filters with a fixed 3×3 kernel size, the second layer utilizes 64 filters, and the third convolutional layer uses 128 filters. Each convolutional layer is followed by the ReLU activation function to introduce non-linearity and a down-sampling layer known as the max pooling layer with a kernel size of 2×2 to reduce the spatial dimension, highlight critical features, and enhance the generalization and convergence of the model.

The extracted spatial features of input images by the CNN architecture are then flattened and reshaped to serve as input for the LSTM model. The first layer of LSTM comprises 128 units, followed by the second LSTM layer with 64 units, both using ReLU as an activation function. Temporal dependencies and sequential patterns are captured by these layers. The sigmoid activation function is employed in the final dense layer to give a single value as an output for the binary classification task, predicting the cracked or intact output images to indicate the structural conditions of beams and bridges. The architecture of the CNN-LSTM model is depicted in Figures 8 and 9, and the information regarding each layer within the model is presented in Table 1.

Figure 8. Schematic of the computer vision-based approach

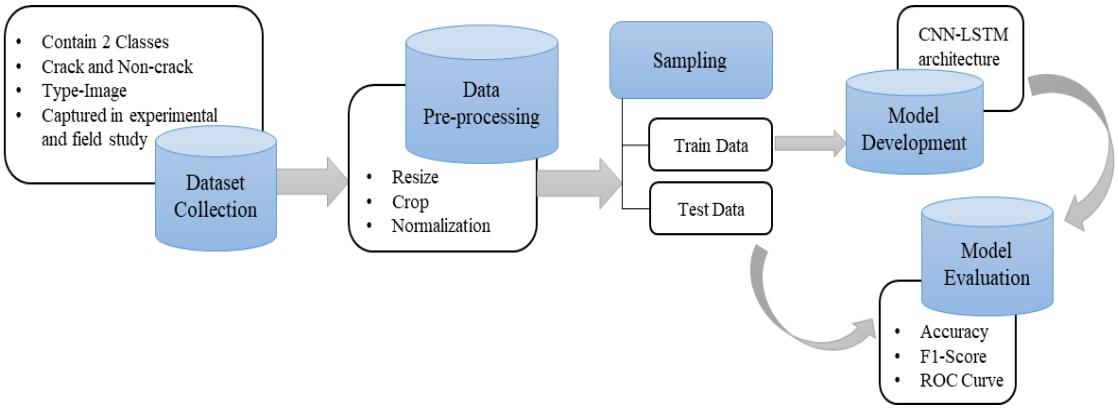


Figure 9. Schematic of the CNN-LSTM architecture

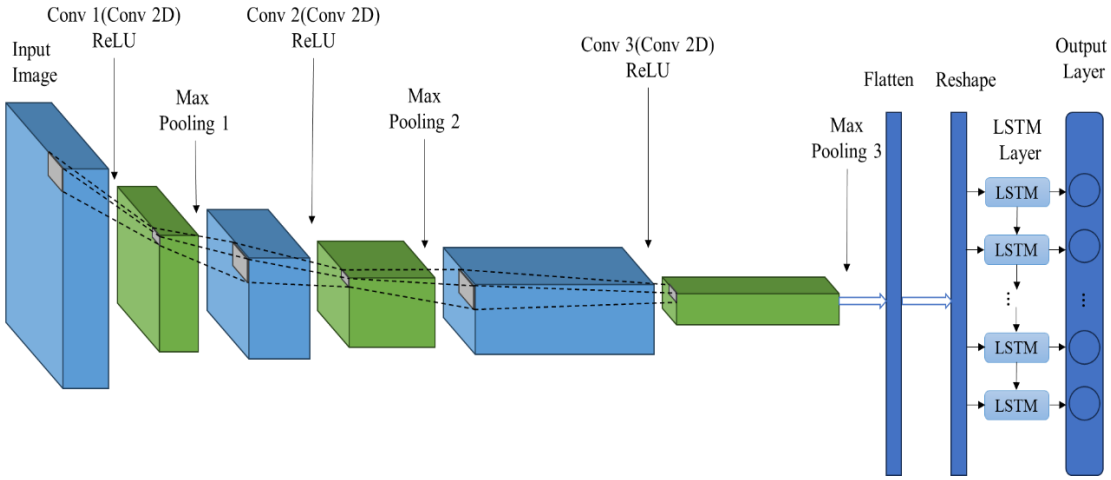


Table 1. Details of the layers of the deep learning architecture

Layer	Operator	Output Shape (Height × Width × Depth)	Kernel (Height × Width)	Number of Parameters
Layer 1	Conv 1 (Conv2D) / ReLU	98 × 98 × 32	3 × 3	896
Layer 2	Max Pooling 1	49 × 49 × 32	2 × 2	0
Layer 3	Conv 2 (Conv2D) / ReLU	47 × 47 × 64	3 × 3	18496
Layer 4	Max Pooling 2	23 × 23 × 64	2 × 2	0
Layer 5	Conv 3 (Conv2D) / ReLU	21 × 21 × 128	3 × 3	73856
Layer 6	Max Pooling 3	10 × 10 × 128	2 × 2	0
Layer 7	Flatten	12800	-	0
Layer 8	Reshape	1 × 12800	-	0
Layer 9	LSTM 1 / ReLU	1 × 128	-	6619648
Layer 10	LSTM 2 / ReLU	64	-	49408
Layer 11	Dense / Sigmoid	1	-	65

2.3 Crack Identification with Deep Learning Model

2.3.1. Dataset Preprocessing

To prepare the raw images of the collected dataset for final processing within the proposed CNN-LSTM architecture, a series of preprocessing steps were conducted. These steps comprised cropping, resizing, shuffling, splitting, and normalization. Initially, images were cropped and resized to focus solely on the surfaces of the beams and bridges, and extraneous parts of the surrounding environment were eliminated. This step ensured that only relevant portions of the images were retained. Subsequently, the processed images were employed to form our dataset for the proposed architecture. During this stage, each image was uniformly resized to 227×227 pixels to maintain consistency across the dataset. Following resizing, the pixel values were normalized to a

range of 0 to 1. This standardization was essential to enhance training performance by providing uniform input for the neural network.

2.3.2. Model Training and Hyperparameters

In this research, the model was trained on images collected by UAV from laboratory experiments of two beams subjected to shear-dominant loading, as well as a dataset of images captured from bridges, as described in Section 3. The dataset included 3,000 images, equally 1,500 images for each class of cracked and intact instances. These images then were split into training and test sets. To improve the model's robustness, data pre-processing and augmentation techniques such as cropping, resizing, shuffling, splitting, and normalization were employed during the training stage. After training the dataset using the proposed CNN-LSTM model architecture, the system’s performance was evaluated utilizing several metrics such as accuracy, ROC curve, F-1 score, and visualization of classification and object detection results. These evaluations were conducted to determine the model's effectiveness.

This study utilized the TensorFlow deep learning framework and Python programming language for the model compilation. Simulations were executed on a desktop system featuring an Intel Core i7-13700 CPU running at 2.1 GHz, an Intel UHD Graphics 770 GPU, and 32 GB of RAM. Furthermore, the software environment included Windows 11, TensorFlow 2.16.1, and Python 3.11.5, all operated within a Jupyter notebook.

In deep learning, the selection of hyperparameters is essential for optimal model performance, thus necessitating the careful adjustment of the model parameters. In this research, the model's hyperparameters were optimized utilizing the control variable method, by focusing on the optimizer, learning rate, and test size. Optimizers are designed to change the weight of neural networks to minimize the loss function. Common optimizers comprise Adam, Stochastic Gradient Descent (SDG), RMSprop, and Adagrad, each with a unique algorithm to update weights and provide efficient convergence. The learning rate is a crucial hyperparameter that dictates the step size of weight updates during the training process. Table 2 illustrates the results of tuning hyperparameters.

The training process of the model involved 20 epochs, with an average duration of 30 seconds per epoch. In this classification task, accuracy measures the frequency of correct class label predictions and was calculated as the ratio of the number of correct predictions to the total number of predictions that happened. Loss, on the other hand, measures the degree of alignment between the model’s prediction and the actual data, highlighting the discrepancy between the actual and predicted values. The primary objective of the training model is to decrease the loss, thus enhancing the model’s overall performance and accuracy.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \quad [9]$$

Table 2. Optimizing hyperparameters of CNN-LSTM model and performance of the model

Optimizer	Learning Rate	Test Size	Accuracy	Loss
Adam	0.00001	0.15	0.9822	0.0604
Adam	0.00001	0.20	0.9950	0.0250
Adam	0.00001	0.25	0.9879	0.0319
Adam	0.00001	0.30	0.9911	0.0676
Adam	0.00001	0.35	0.9828	0.0979
Adam	0.000001	0.15	0.4799	0.6897
Adam	0.000001	0.20	0.4783	0.6897
Adam	0.000001	0.25	0.4866	0.6912
Adam	0.000001	0.30	0.4877	0.6881
Adam	0.000001	0.35	0.4847	0.6908
SDG	0.001	0.15	0.4799	0.6933
SDG	0.001	0.20	0.4783	0.6935
SDG	0.001	0.25	0.4866	0.6932
SDG	0.001	0.30	0.4877	0.6930
SDG	0.001	0.35	0.4847	0.6934
SDG	0.0001	0.15	0.4799	0.6931
SDG	0.0001	0.20	0.4783	0.6934
SDG	0.0001	0.25	0.4866	0.6931

Optimizer	Learning Rate	Test Size	Accuracy	Loss
SDG	0.0001	0.30	0.4877	0.6932
SDG	0.0001	0.35	0..4847	0.6933
RMSprop	0.0001	0.15	0.9977	0.0193
RMSprop	0.0001	0.20	0.9966	0.0131
RMSprop	0.0001	0.25	0.9893	0.0250
RMSprop	0.0001	0.30	0.9944	0.0210
RMSprop	0.0001	0.35	0.9514	0.1299
RMSprop	0.00001	0.15	0.4799	0.6931
RMSprop	0.00001	0.20	0.6650	0.6863
RMSprop	0.00001	0.25	0.4866	0.6922
RMSprop	0.00001	0.30	0.4877	0.6929
RMSprop	0.00001	0.35	0.4847	0.6937
Adagrad	0.01	0.15	0.5088	0.6905
Adagrad	0.01	0.20	0.4783	0.6930
Adagrad	0.01	0.25	0.4866	0.6920
Adagrad	0.01	0.30	0.6555	0.6916
Adagrad	0.01	0.35	0.4847	0.6929
Adagrad	0.001	0.15	0.4799	0.6932
Adagrad	0.001	0.20	0.4783	0.6934
Adagrad	0.001	0.25	0.4866	0.6933
Adagrad	0.001	0.30	0.4877	0.6931

Optimizer	Learning Rate	Test Size	Accuracy	Loss
Adagrad	0.001	0.35	0.4847	0.6934

Discussion of Results

The overall performance of the proposed CNN-LSTM model for structural health monitoring of concrete bridges and beams was assessed using UAV-captured images from experiments conducted on two beams and the surface of two bridges in the field study. Following the pre-processing, the dataset was split randomly into training and test sets. To evaluate the impact of different dataset sizes on model performance, five different scenarios were considered: 85% training set and 15% test set (Case 1), 80% training set and 20% test set (Case 2), 75% training set and 25% test set (Case 3), 70% training set and 30% test set (Case 4), and 65% training set and 35% test set (Case 5). The performance of the model was measured and presented for each case by using accuracy, ROC curve, F1 score, and visualizations of classification and object detection results. The performance of the model for each case is illustrated in Table 3. The results demonstrated that the highest performance was observed in Case 2.

Table 3. Performance of the proposed CNN-LSTM model across five cases with varying test set size

Data Subset	Test Size	Accuracy	AUC	F-1 Score
Case 1	15%	98.22	1.00	0.98
Case 2	20%	99.50	1.00	0.99
Case 3	25%	98.79	1.00	0.99
Case 4	30%	99.11	1.00	0.99
Case 5	35%	98.28	0.99	0.98

The results show that the proposed model demonstrated excellent accuracy and reliability in detecting cracks. One of the essential metrics to evaluate within the model is the confusion matrix, which represents true positive, false positive, true negative, and false negative visually. In this study, positive and negative refer to cracked and uncracked classes, respectively. Figure 10 represents the confusion matrix for all five scenarios with different test sizes. The results indicate that Case 2 performed best (refer to Figure 10(b)), with an overall classification accuracy of 99.50%. Specifically, the accuracy for cracked beams (positive samples) was 99.3%, while for intact beams (negative samples), it was 99.7%.

The other metric to assess the model’s performance is the ROC Curve, which is employed to evaluate the model's ability to differentiate between positive and negative classes at various threshold settings. A higher area under the curve (AUC) reflects superior model performance and the sensitivity and specificity of the model is indicated by the sharp ascent of the ROC curve. Results for the ROC curve are depicted in Figure 11, highlighting the deep learning model's robustness and efficiency in identifying and distinguishing between cracked and intact surfaces, with AUC=1 for almost all cases.

Figure 10. Confusion matrix for five cases with different test sizes

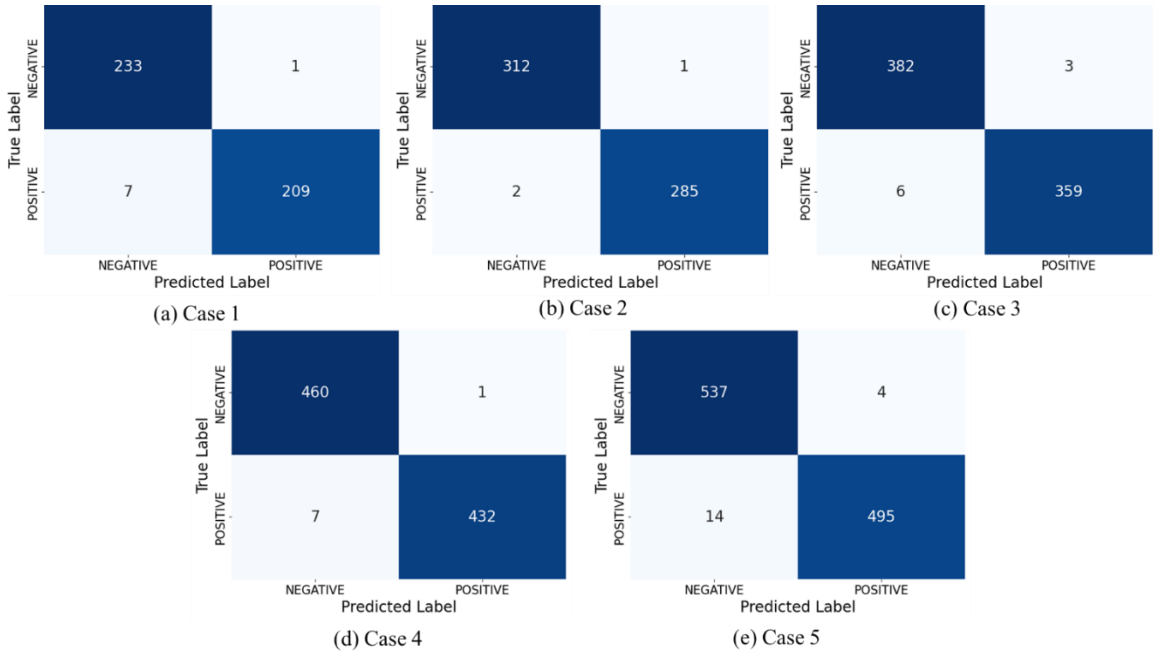
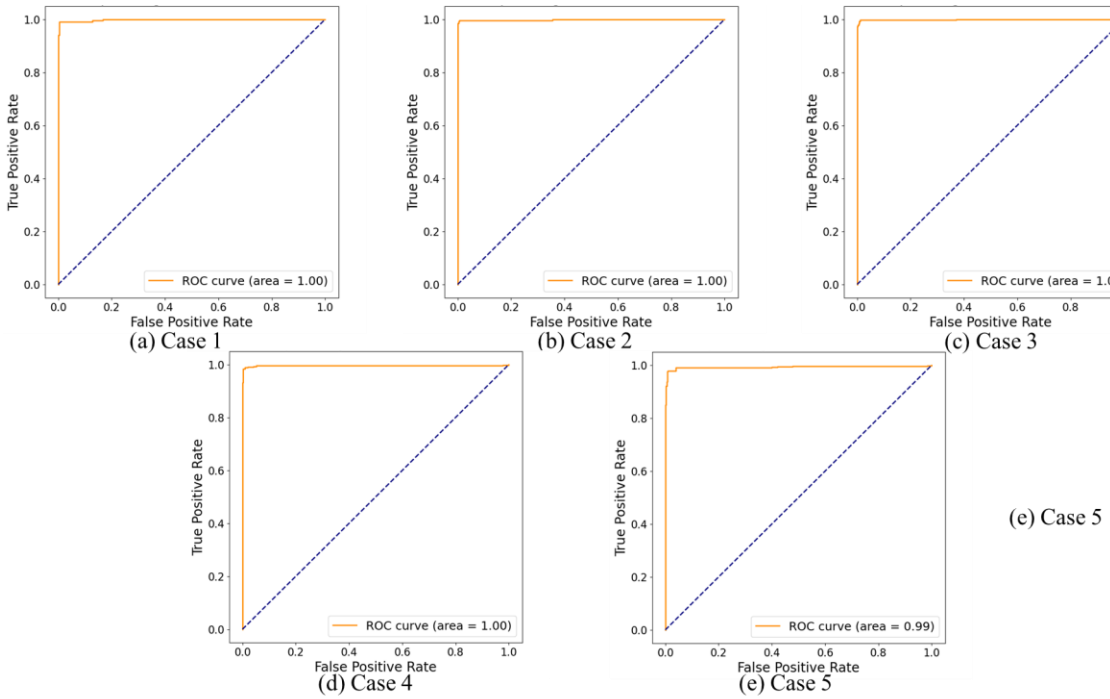


Figure 11. ROC curve for each case with different test size



The visualization of the predicted and actual instances of both beams and bridges are shown in Figures 12 and 13. The results show the effectiveness of the model in correctly identifying instances. Note that the amounts 1 and 0 refer to cracked (positive) and uncracked (negative), respectively. This visual representation helps to interpret the model's performance easily and further confirms the model’s capability to accurately distinguish between the two classes.

Figure 12. Crack classification for dataset collected from experimental beams: 0 refers to intact (negative), and 1 refers to cracked (positive) surfaces

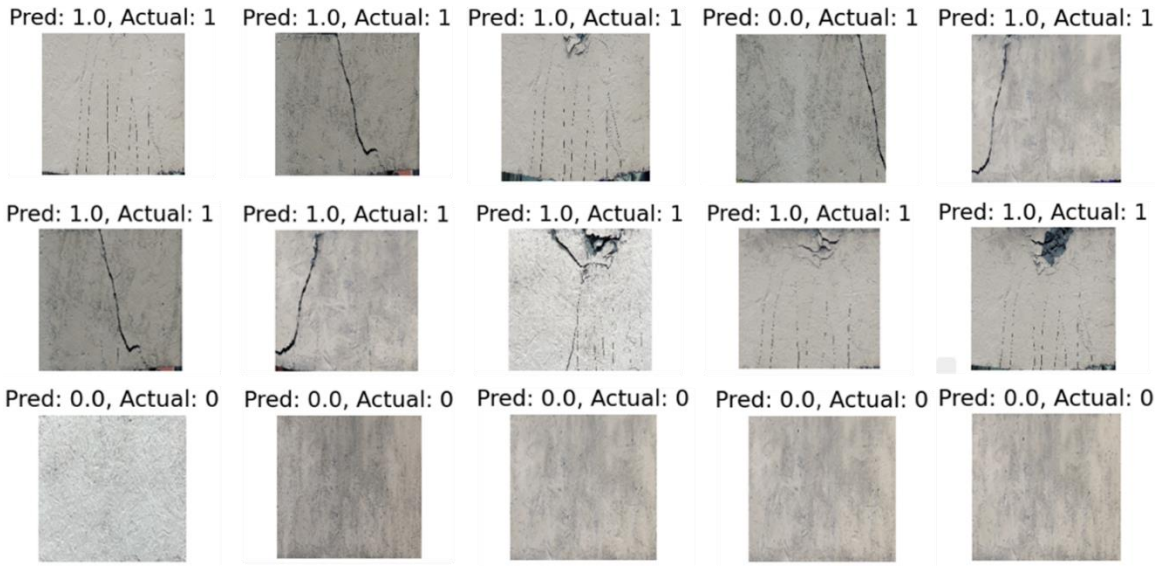
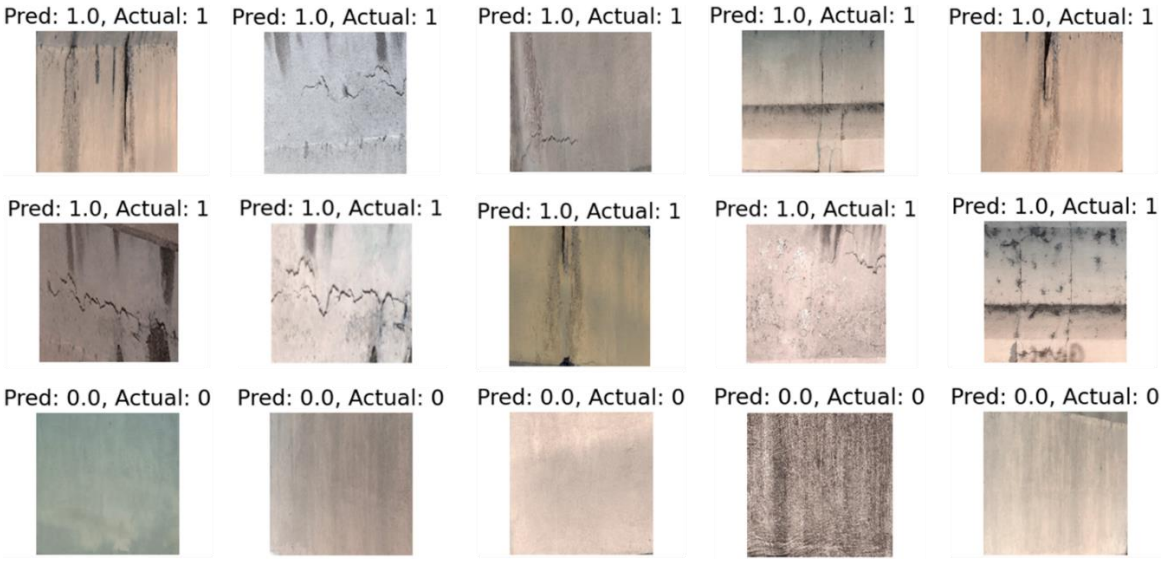


Figure 13. Crack classification for dataset collected from bridges: 0 refers to intact (negative) and 1 refers to cracked (positive) surfaces



Figures 14 and 15 illustrate the results for the object detection applied to concrete surfaces; to identify and locate cracks. a rectangular bounding box is drawn around the detected cracks. Overall, it is expected that enlarging the dataset and integrating a broader range of data will significantly enhance the performance of the proposed CNN-LSTM model in identifying cracks in beams and bridges.

Figure 14. Cracks object detection for concrete surfaces of experimental beams

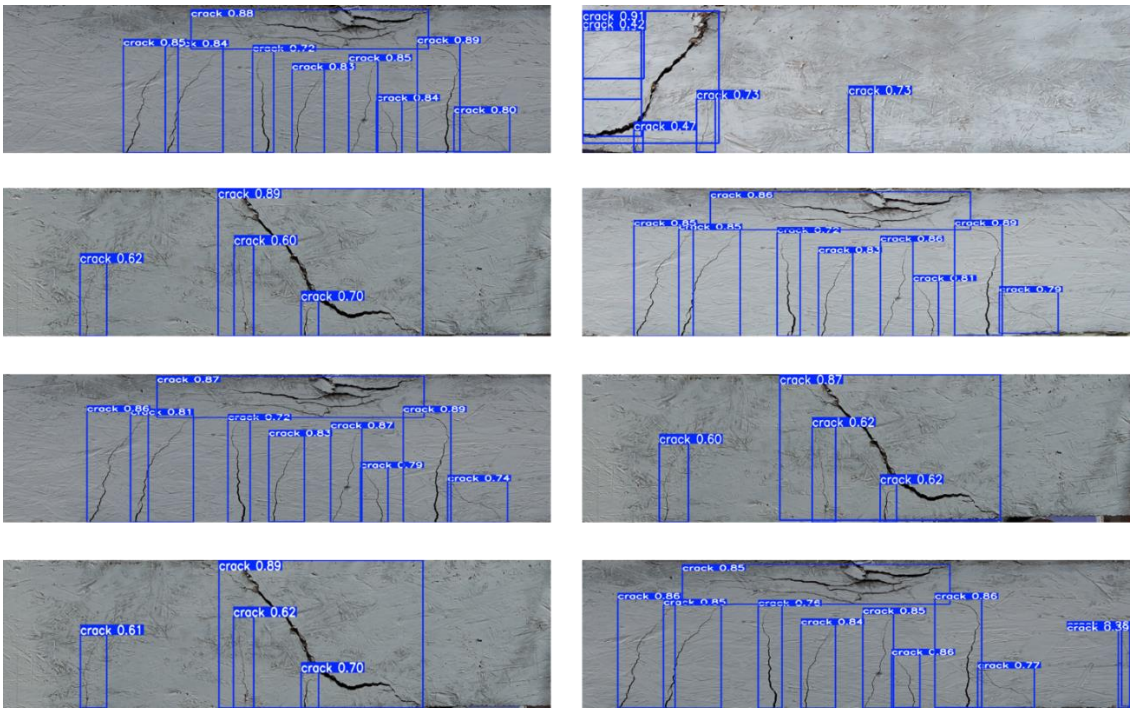
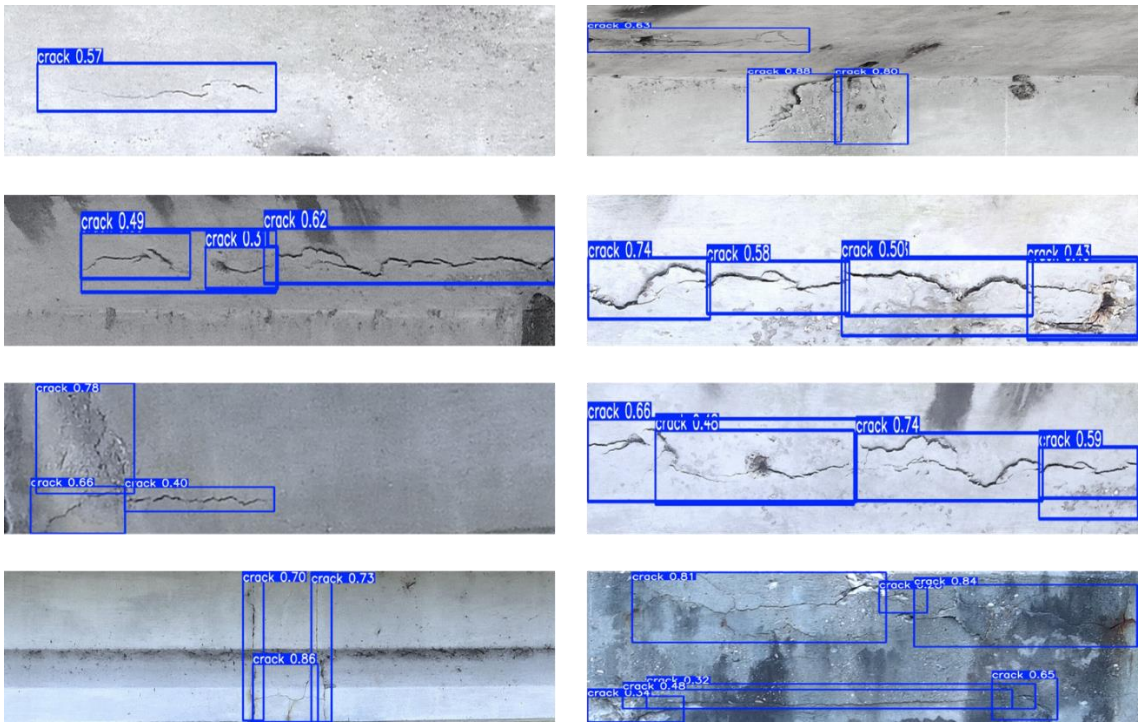


Figure 15. Cracks object detection for concrete surfaces of the bridges



Moreover, as shown in Table 2, researchers evaluated the performance of our model using various optimizers, learning rates, and test set sizes to determine the optimal combination to achieve the highest accuracy and the lowest loss. The table presents the results of these assessments. Notably, the Adam optimizer, with a learning rate of 0.00001, consistently achieved high accuracy (above 98%) and low loss across different test sizes, indicating its robustness and reliability. Similarly, the RMSprop optimizer, with a learning rate of 0.0001, demonstrated outstanding performance, achieving perfect accuracy across all test sizes. In contrast, the SGD and Adagrad optimizers showed lower and more variable performance, particularly at lower learning rates, suggesting they are less suitable for this task. These findings highlight the effectiveness of the Adam and RMSprop optimizers for training the model, with the Adam optimizer (learning rate of 0.00001) being the most consistently reliable choice.

Conclusions

In this research, a novel method for structural condition assessment of concrete beams and bridges was presented by integrating a CNN-LSTM computer vision model with the utilization of UAV for data collection. The proposed method employs CNN to extract high-dimensional spatial features from the input and LSTM to extract time-series features of the dataset. A total of 3,000 images of cracked and non-cracked concrete surfaces were captured in both laboratory and field tests and were randomly split to train and test the proposed CNN-LSTM architecture. Various test sizes were evaluated, with the best performance achieved using 80% of the data for training and 20% for testing. The model demonstrated an accuracy of 99.5% on the testing dataset.

The implemented framework effectively utilized the well-trained model and showed highly effective performance in accurately detecting cracks, despite being trained on a limited dataset. Confusion matrix, ROC curve, f-1 score, and visual representation of object detection and classification were implemented to validate the results. Although the proposed deep learning model achieved good performance, and the rate of false positives and false negatives from the confusion matrix was satisfactory, its performance could be further enhanced by incorporating more training data to improve generalization to unseen cases.

Future research could aim to develop models utilizing larger and more diverse datasets or integrating other deep learning methods to achieve better performance. Moreover, while UAVs offer significant benefits for data collection, they also present limitations, such as difficulties in accessing hard-to-reach areas and dependence on weather conditions. Future studies could explore optimizing the use of UAVs to overcome these limitations and achieve the best results in data collection.

Recommendations

Although the proposed method to detect cracks for concrete surfaces by integrating UAV for data collection and the CNN-LSTM model to train and test the dataset performed well, some limitations should be noted. One notable challenge was the data collection constraint. Despite the benefits of using UAVs in visual inspection of civil infrastructure, UAV operation could be significantly challenged by poor lighting, different weather conditions such as rain, fog, strong winds, and extreme hot weather, and limited battery life of the UAV, which restricts the frequency and timing of data collections in mission flight. The other significant limitation of this study is the dependence on a limited dataset, since deep learning models generally require extensive and diverse datasets to be trained effectively. Thus, the restricted quantity of the dataset may have impacted the model's performance.

To overcome the challenges noted above, future research should consider developing advanced UAVs equipped with weather-resistant features, enhanced lighting systems, and improved battery technology to ensure the higher quality of collected data. To tackle the limitation of a small dataset, future work should focus on expanding the dataset by collecting a broader range of data that gathers a variety of types of cracks and damage scenarios under different environmental conditions. Furthermore, applying data augmentation methods and generating synthetic data can contribute to creating a balanced and broad dataset, thereby enhancing the model's training and generalization capabilities. Furthermore, future work can explore integrating other hybrid deep learning models and transfer learning to leverage pre-trained models on similar tasks and enhance the model's accuracy and robustness.

Acronyms, Abbreviations, and Symbols

Term	Description
CNN	Convolutional Neural Networks
in.	inch(es)
LSTM	Long Short-Term Memory
m	meter(s)
UAVs	Unmanned Aerial Vehicles
UHPC	Ultra-High-Performance Concrete

References

[1] Z. Li, X. Zhou, H. Ma, and D. Hou, *Advanced concrete technology*. John Wiley \& Sons, 2022.

[2] B. Han, L. Zhang, and J. Ou, *Smart and multifunctional concrete toward sustainable infrastructures*. Springer, 2017.

[3] T. Harms, S. Sedigh, and F. Bastianini, "Structural Health Monitoring of Bridges Using Wireless Sensor Networks," *IEEE Instrum. Meas. Mag.*, vol. 13, no. 6, pp. 14–18, 2010, doi: 10.1109/MIM.2010.5669608.

[4] R. Solhmirzaei, H. Salehi, and V. Kodur, "A computer vision-based approach for crack detection in ultra high performance concrete beams," *Comput. Concr.*, vol. 33, no. 4, p. 341, 2024.

[5] H. J. Jung, J. H. Lee, S. S. Yoon, I. H. Kim, and S. S. Jin, "Condition assessment of bridges based on unmanned aerial vehicles with hybrid imaging devices," in *Proceedings of the 2017 World Congress on Advances in Structural Engineering and Mechanics (ASEM17), Ilsan, Korea*, 2017.

[6] B. F. Spencer, V. Hoskere, and Y. Narazaki, "Advances in Computer Vision-Based Civil Infrastructure Inspection and Monitoring," *Engineering*, vol. 5, no. 2, pp. 199–222, 2019, doi: https://doi.org/10.1016/j.eng.2018.11.030.

[7] S. Dorafshan, R. J. Thomas, and M. Maguire, "Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete," *Constr. Build. Mater.*, vol. 186, pp. 1031–1045, 2018.

[8] P. Malhotra, L. Vig, G. Shroff, P. Agarwal, and others, "Long short term memory networks for anomaly detection in time series.," in *Esann*, 2015, p. 89.

[9] T. Rakha and A. Gorodetsky, "Review of Unmanned Aerial System (UAS) applications in the built environment: Towards automated building inspection procedures using drones," *Autom. Constr.*, vol. 93, pp. 252–264, 2018, doi: https://doi.org/10.1016/j.autcon.2018.05.002.

[10] B. Kim and S. Cho, "Automated Vision-Based Detection of Cracks on Concrete Surfaces Using a Deep Learning Technique," *Sensors*, vol. 18, no. 10, 2018, doi: 10.3390/s18103452.

[11] H. Freimuth and M. König, "Planning and executing construction inspections with unmanned aerial vehicles," *Autom. Constr.*, vol. 96, pp. 540–553, 2018, doi: https://doi.org/10.1016/j.autcon.2018.10.016.

[12] W. Ding, H. Yang, K. Yu, and J. Shu, “Crack detection and quantification for concrete structures using UAV and transformer,” *Autom. Constr.*, vol. 152, p. 104929, 2023, doi: https://doi.org/10.1016/j.autcon.2023.104929.

[13] H. Kim, J. Lee, E. Ahn, S. Cho, M. Shin, and S.-H. Sim, “Concrete Crack Identification Using a UAV Incorporating Hybrid Image Processing,” *Sensors*, vol. 17, no. 9, 2017, doi: 10.3390/s17092052.

[14] I.-H. Kim, H. Jeon, S.-C. Baek, W.-H. Hong, and H.-J. Jung, “Application of Crack Identification Techniques for an Aging Concrete Bridge Inspection Using an Unmanned Aerial Vehicle,” *Sensors*, vol. 18, no. 6, 2018, doi: 10.3390/s18061881.

[15] S. Li and X. Zhao, “Image-based concrete crack detection using convolutional neural network and exhaustive search technique,” *Adv. Civ. Eng.*, vol. 2019, no. 1, p. 6520620, 2019.

[16] V. P. Golding, Z. Gharineiat, H. S. Munawar, and F. Ullah, “Crack detection in concrete structures using deep learning,” *Sustainability*, vol. 14, no. 13, p. 8117, 2022.

[17] Y. Cha, W. Choi, and O. Buyukozturk, “Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks,” *Comput. Civ. Infrastruct. Eng.*, vol. 32, pp. 361–378, Mar. 2017, doi: 10.1111/mice.12263.

[18] Y.-J. Cha, W. Choi, G. Suh, S. Mahmoudkhani, and O. Büyüköztürk, “Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types,” *Comput. Civ. Infrastruct. Eng.*, vol. 33, no. 9, pp. 731–747, 2018.

[19] J. Deng, A. Singh, Y. Zhou, Y. Lu, and V. C.-S. Lee, “Review on computer vision-based crack detection and quantification methodologies for civil structures,” *Constr. Build. Mater.*, vol. 356, p. 129238, 2022, doi: https://doi.org/10.1016/j.conbuildmat.2022.129238.

[20] T. H. Dinh, Q. P. Ha, and H. M. La, “Computer vision-based method for concrete crack detection,” in *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2016, pp. 1–6. doi: 10.1109/ICARCV.2016.7838682.

[21] A. Saxena, “An introduction to convolutional neural networks,” *Int. J. Res. Appl. Sci. Eng. Technol*, vol. 10, no. 12, pp. 943–947, 2022.

[22] S. Sakib, N. Ahmed, A. J. Kabir, and H. Ahmed, “An overview of convolutional neural network: Its architecture and applications,” 2019.

[23] N. Aloysius and G. Madathilkulangara, *A review on deep convolutional neural networks*. 2017. doi: 10.1109/ICCSP.2017.8286426.

[24] S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” in *2017 International Conference on Engineering and Technology (ICET)*, 2017, pp. 1–6. doi: 10.1109/ICEngTechnol.2017.8308186.

[25] R. Jozefowicz, W. Zaremba, and I. Sutskever, “An empirical exploration of recurrent network architectures,” in *International conference on machine learning*, 2015, pp. 2342–2350.

[26] Y. Yu, X. Si, C. Hu, and J. Zhang, “A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures,” *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, Jul. 2019, doi: 10.1162/neco_a_01199.

[27] A. Sherstinsky, “Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network,” *Phys. D Nonlinear Phenom.*, vol. 404, p. 132306, 2020, doi: https://doi.org/10.1016/j.physd.2019.132306.

[28] R. C. Staudemeyer and E. R. Morris, “Understanding LSTM--a tutorial into long short-term memory recurrent neural networks,” *arXiv Prepr. arXiv1909.09586*, 2019.