Towards identification of long-term building defects using transfer learning

Aravinda Boovaraghavan,
Christy Jackson Joshua and Abdul Quadir Md

School of Computer Science and Engineering,
Vellore Institute of Technology,
Chennai, 600127, India
Email: aravinda.b2019@vitalum.ac.in
Email: christyjackson.j@vit.ac.in
Email: abdulquadir.md@vit.ac.in

Kong Fah Tee*

Department of Civil and Environmental Engineering,
King Fahd University of Petroleum and Minerals,
Dhahran 31261, Saudi Arabia
and
Interdisciplinary Research Center for
Construction and Building Materials,
King Fahd University of Petroleum and Minerals,
Dhahran 31261, Saudi Arabia
Email: tee.fah@kfupm.edu.sa
Email: kftee2010@gmail.com
*Corresponding author

V. Sivakumar

Department of Information and Communication Technology,
Manipal Institute of Science and Technology,
Manipal Academy of Higher Education,
576104, Manipal, India
Email: sivakumar.v@manipal.edu

Abstract: Detecting long-term issues on various types of building wall surfaces, such as cracks, flakes, and roofs, is vital for timely maintenance and repairs before they become too risky and expensive. Currently, building managers manually assess the building conditions to survey and communicate the state of their buildings. However, this manual process is subjective, often leads to inaccuracies, and is time-consuming; thus, it needs to be more efficient. These flaws can severely influence a building's structural stability if they go undiscovered and ignored. In this context, this study proposes an approach named towards identification of long-term building defects using transfer learning (TILT) to identify unnoticed defects such as cracks, flakes, and roofs robustly and accurately in buildings. The proposed model has been tested using images taken from real-world onsite deployments, and the types of construction issues have been determined with 98.33% accuracy predicted by the VGG16 model and 79.13% accuracy predicted by the ResNet50 model. Overall, the VGG16 model gives better results compared to ResNet50.

Keywords: structural health monitoring; damage detection; building defects; classification; transfer learning; VGG16; ResNet50.

Reference to this paper should be made as follows: Boovaraghavan, A., Joshua, C.J., Md, A.Q., Tee, K.F. and Sivakumar, V. (2025) 'Towards identification of long-term building defects using transfer learning', *Int. J. Structural Engineering*, Vol. 15, No. 2, pp.147–170.

Biographical notes: Aravinda Boovaraghavan holds a BTech in Computer Science and Engineering from Vellore Institute of Technology and is currently pursuing his Master of Information Systems Management at Carnegie Mellon University. Previously, he worked as a Technology Consultant at PwC India, specialising in Software Development, Generative AI, and emerging technology. He has experience working across the Government, Education, and Healthcare industry, providing software solutions. He has worked with machine learning and deep learning technology, including neural networks, computer vision, and natural language processing. His research interests include data-driven decision-making, artificial intelligence, machine learning, and the integration of business and technology for impactful solutions.

Christy Jackson Joshua holds a PhD from VIT Chennai and an MS in Information Technology from Griffith University, Australia, specialising in business. He is an Associate Professor in the School of Computer Science and Engineering at VIT Chennai and a member of the cyber physical systems group. He has extensive experience in grid computing, vehicular ad hoc networks, and web application development. He has published in reputed journals, presented at major conferences, and served as session chair and co-conference chair. His research interests include evolutionary computation, ad hoc networks, artificial intelligence, machine learning, and autonomous systems.

Abdul Quadir Md received his BE in Computer Science and Engineering in 2006 from Anna University, Chennai, India, and ME in Computer Science and Engineering in 2011 from College of Engineering, Anna University, Chennai, India, and PhD degree from Vellore Institute of Technology, Chennai, India in 2019. He is currently working as an Associate Professor at Vellore Institute of Technology, Chennai, India. He is the guest editor and also serves as an editorial board member for several international journals and conferences. He is the conference chair for International conference on Innovative Technology for Sustainable Development, co conference chair for International Conference on Big Data and Cloud Computing. His research interests focus on machine learning, cloud computing, and IoT.

Kong Fah Tee is an Associate Professor at KFUPM. He was previously a Professor at INTI International University and a Reader at the University of Greenwich. He has published 4 books and more than 260 papers. He has been included in the World's Top 2% Scientist ranking by Stanford University for five consecutive years from 2020–2024. He has been awarded research grants with a total amount of over 6 million SAR. He has supervised 13 PhD students to successful completion. He was awarded PIFI from the Chinese Academy of Sciences, International Research Collaboration Award from Sydney University, and foreign expert to Nanjing University of Aeronautics and Astronautics.

V. Sivakumar received his PhD in Computer Science and Engineering, specialising in evolutionary algorithms for optimisation techniques for underwater applications. He is an Associate Professor with the Manipal Institute of Technology, Manipal Academy of Higher Education, Karnataka, India, a renowned institute of eminence. Driven by a passion for optimisation algorithms, he is deeply engaged in developing evolutionary algorithms, AI-based models, and platforms for a wide array of applications. With a keen interest in the applications of artificial intelligence, he actively explores its potential in involves cross-domain applications, where he implements AI-based models to tackle complex problems. His dedication to research is evident through numerous publications in esteemed journals and conferences. He has an h-index of 8, indicating the impact of his research work in the academic community. Additionally, highlighting his standing within the IEEE professional organisation.

1 Introduction

Structural damages, faults, and defects in modern buildings are common and are a natural part of the ageing process for any structure. These primarily occur due to various environmental factors, such as water damage, mould growth, pest infestations, earthquakes, and other factors. The environmental effects on a building can be wide-ranging and costly to repair. Such ecological effects can pose a safety hazard to those who use the building, as they may compromise the structural integrity of the building or create other hazards such as fire risks or electrical issues. Figure 1 shows the different types of building defects, such as cracks, flakes, and roof cracks (roofs). In addition, building owners or managers may also have legal liabilities if they are aware of defects in a building and fail to address them. For example, the building owners or managers may be held liable if a defect leads to injury or damage to property. Defects in a building can also affect its value, as they may make the building less attractive to potential buyers or renters. Prior approaches have shown that clients increasingly request efficient methods for detecting and communicating the state of their facilities regularly so that essential maintenance and repairs may be carried out promptly and proactively before they become too risky and expensive (Mohseni et al., 2013; Agdas et al., 2016; Shamshirband et al., 2020). It is essential to identify and address these issues to maintain the building's safety, functionality, and value. Current approaches include a conventional method of hiring building surveyors to undertake a condition evaluation. Assessing the condition of buildings and estimating costs for renewals, repairs, and maintenance, both short and long-term necessitates a manual evaluation and thorough documentation of the physical state of their components (Tee et al., 2003; Wang et al., 2011; Lorenzoni et al., 2016; Oh et al., 2017). However, there are several challenges with building managers having to assess the condition of their buildings manually:

- Time-consuming: assessing the condition of a building can be time-consuming, especially if the building is large or has many different components that need to be inspected. This can be especially challenging for building managers who are responsible for managing multiple buildings or properties.

- Subjective: manual assessments are often subjective, as they are based on the individual observations and opinions of the person doing the assessment. This can lead to inconsistency in how different buildings are evaluated and make it difficult to compare the conditions of different buildings.
- Inaccurate: manual assessments can be prone to errors or inaccuracies, especially if the assessor needs to be fully trained or qualified to do the assessment. This can lead to problems with maintenance and repair work being carried out on the building, as well as issues with budgeting and planning for future repairs.

More recently, several prior approaches focused on the automated detection of building damages caused by natural disasters such as earthquakes. Such applications use deep learning and machine learning approaches to identify structural faults due to earthquakes (Salehi and Burgueño, 2018). Deep learning approaches, particularly convolutional neural networks (CNNs), have shown potential for detecting earthquake-caused structural defects. Deep learning-based fracture damage identification using convolutional neural networks has been developed (Cha et al., 2017) which provides insight into the use of CNNs for crack damage detection. Similarly, autonomous structural visual inspection using region-based deep learning has been proposed (Cha et al., 2018) for the utilisation of region-based deep learning for detecting different forms of damage. These important papers highlight the promise of deep learning approaches in the identification of structural faults caused by seismic occurrences. Recently, a surge of innovative methodologies that utilise machine learning and deep learning techniques has been witnessed in the field of automated detection of building damages, especially those resulting from natural disasters such as earthquakes (Salehi and Burgueño, 2018). While these methods have proven effective in identifying structural faults, it is crucial to recognise that they are not the sole existing solutions addressing this issue.

Indeed, many structural health monitoring and damage detection approaches have been developed such as vibration-based methods (Tee, 2018; Wu et al., 2021), statistical approaches (Zhang et al., 2018; Tee et al., 2013), final element model updating (Haidarpour and Tee, 2020; Koh et al., 2006), etc. A multitude of novel approaches have been developed that not only consider defects due to structural changes but also offer real-time detection capabilities. Examples of these include the application of autonomous unmanned aerial vehicles (UAVs) for structural health monitoring using deep learning and an ultrasonic beacon system with geo-tagging. Further, methods such as real-time multiple damage mapping using autonomous UAVs, deep faster region-based neural networks for global positioning system (GPS)-denied structures, and deep learning-based obstacle-avoiding autonomous UAVs with fiducial marker-based localisation also address structural health monitoring (Waqas and Cha, 2023).

Moreover, there has been considerable progress in the development of real-time methodologies. These include an efficient attention-based deep encoder and decoder for automatic crack segmentation, an attention-based generative adversarial network with internal damage segmentation using thermography, and SDDNet: real-time crack segmentation (Kang and Cha, 2022; Choi and Cha, 2020). Notably, these methods were conceived and developed without reliance on pre-existing publicly available networks. As such, they are optimised, lightweight networks, specifically designed for structural damage segmentation in real-time scenarios. These networks can handle large image inputs (such as 1000x500) and process at least 30 frames per second on complex background scenes, making them highly applicable to real-world challenges.

Figure 1 Building defects, (a) cracks (b) flakes (c) roofs (see online version for colours)

(a) (b) (c)

To overcome these challenges, a transfer learning-based model is presented that uses deep learning to robustly and accurately detect and identify unnoticed defects in buildings, such as cracks, flakes, and roofs, as shown in Figure 1. The proposed approach uses a CNN-based pre-trained model-VGG16 that is fine-tuned to assess the type of natural defects in the building over time. The goal is to lessen the subjectivity involved in classifying faults on a scale and at a rate that has previously exceeded typical project cost restrictions. Overall, the following contribution is made:

- The design and implementation of the proposed approach are presented to robustly and accurately identify unnoticed defects in buildings, such as cracks, flakes and roofs.

Additionally, the results of the proposed approach, including testing on real-world images with an accuracy of 98.33% and numerous optimisations for deep learning-based training and serving, are discussed in the results section to highlight its efficacy. Despite these advancements, there is a pressing need for solutions that address defects resulting from both structural changes and natural ageing processes in buildings. Existing systems often fail to balance accuracy, scalability, and practical usability across diverse defect categories. The proposed research aims to fill this gap by introducing a robust, transfer-learning-based model using fine-tuned VGG16 for accurately identifying building defects like cracks, flakes, and roof damage.

This section provides background information by highlighting the need for efficient defect detection in building structures, the drawbacks of conventional manual inspection techniques, and the recent development of deep learning approaches for structural health monitoring. Following this, the second section discusses relevant research, emphasises the novelty of the proposed methodology, and provides insights into previous studies. In the third section, the proposed methodology is explained, along with the deep learning models and data augmentation techniques. The fourth section summarises the findings, demonstrating the accuracy and performance of the model. Finally, the last section concludes the paper with a summary of the findings and a discussion on future work, addressing potential advancements and applications of the research.

2 Related work

Several efforts have been made to evaluate structural issues and showcase the construction defects as discussed. As a multi-label classifier, Dung (2019) suggested a CNN model for identifying pavement cracks. It is suggested that a fully convolutional network (FCN) be applied to a vision-based approach for concrete crack detection and density estimation. To determine the efficiency architecture for the FCN encoder, the performance of multiple pre-trained deep CNN architectures for image classification on a public concrete crack dataset is compared. Future research will focus on its use in structural health monitoring for concrete structures. Qu et al. (2020) proposed crack detection based on a set of neural networks and recognition algorithms for detecting cracks. To properly recognise concrete pavement crack images from a range of images with similar properties, the classification model has been trained. For images with basic cracks and a single background, the net and percolation algorithms perform better. Images with shadows and unclear cracks have worse image effects, and image noise has a significant impact on accuracy. Image preprocessing techniques can be used to minimise background noise and enhance the accuracy and effectiveness of crack identification to overcome the mentioned challenges.

Minhas and Zelek (2020) proposed a transfer learning approach based on a network for investigating anomaly detection and model performance using the area under the receiver operating characteristic curve (AUROC) metric, which is a more robust and accurate measure of separation capability than classification accuracy alone. The model uses network-based transfer learning with CNNs to detect defects, and it achieves an excellent AUROC value, indicating the CNNs' strong separation power across all datasets. Researchers have also worked with different algorithms to inspect the conditions of different assets. Zhang et al. (2019) worked on Bayesian methods for operational modal analysis and structural health monitoring design of a 250-m steel-concrete tall building in Shanghai. Davoudi et al. (2018) employed neural networks, support vector machines (SVM), and regression to estimate the internal load magnitudes of beams and slabs, based on surface crack images. Özgenel and Sorguç (2018) emphasise the significance of the convolutional layers used performance is affected by the size of the filter, the number of layers used (apart from convolutional layers), the type of layers utilised, and the number of images used for training, which the reported accuracy of the crack detection model. The proposed pre-trained network approach is adjusted for crack classification tasks using a limited number of training samples if the data variance is given or the test case includes images that resemble the training samples. The optimal number of layers was determined through a study, and networks with 16 to 22 convolutional layers outperformed ResNet networks with more than 50 layers and AlexNet networks with eight convolutional layers. Kessai et al. (2022) employed finite element analysis combined with an artificial neural network (ANN) for the estimation of circular arc crack depths and locations in rotary drilling pipes subjected to free vibration. Perez et al. (2019) used convolutional neural networks (CNN) for the automatic localisation of building flaws, such as mould, wear, and stain, from images. The proposed model is based on a pre-trained CNN classifier of type VGG-16 (later compared with ResNet-50 and Inception models) with class activation mapping (CAM) for object localisation. The suggested approach can correctly identify and categorise multi-class defects, whereas some previous work has only classified cracks, which is a binary classification problem.

A method for image processing was suggested by Hoang for routinely assessing wall conditions. In an integrated model, cracks and spall damage can be identified concurrently. To extract valuable information from digital images, steerable filters, and projection integrals are used as image processing techniques. To extract valuable information from digital images, steerable filters, and projection integrals are used as image processing methods. The classification boundaries of the newly proposed model, which divides the conditions of a wall into five labels – longitudinal crack, transverse crack, diagonal crack, spall damage, and intact wall – are generalised using support vector machines and least squares support vector machines (Hoang, 2018). Jiang et al. (2021) proposed a precise method for U-Net architecture-based building and infrastructure facility flaw detection. The suggested solution functions with various fault shapes and styles. A few defined protocols identify various structural distresses using image functions and fuzzy sets (Pragalath et al., 2018). Different imaging techniques are used to discover and quantify structural flaws as well as give a thorough assessment of civil constructions using fuzzy logic and image processing algorithms (Valero et al., 2019). A brand-new method for automatically identifying and classifying problems was presented. The developed method was used in conjunction with supervised machine learning algorithms to digitise ashlar masonry walling. Segmentation and histogram shape-based thresholding were used in a method for automatic pothole detection in images of asphalt pavement presented by Koch and Brilakis (2011). The detection performance was calculated using accuracy, precision, and recall. The 86% accuracy, 82% precision, and 86% recall obtained as a result show that the majority of potholes in asphalt pavement images can be accurately identified. Using an adaptive histogram-based thresholding algorithm, a new method for automatically recognising spalled spots on the surface of reinforced concrete columns and analysing their properties was proposed (German et al., 2012). By considering large cracks as the continuation of the spalled zone, this approach to detecting spalled regions may exaggerate spalled regions.

Recent advancements have further enhanced defect detection methodologies. Chen et al. (2024) introduced an integrated BIM-based framework (D3M) for defect mapping in built environments that combines defect modelling with efficient management strategies. Zhang et al. (2023) proposed using deep learning techniques alongside percussion sound analysis to identify voids behind tunnel linings effectively. Feng et al. (2023) improved defect detection in shield tunnel linings by utilising an enhanced SOLOv2 method tailored specifically for water leakage identification. Additionally, Wang et al.'s (2024) work highlights multi-objective optimisation in recycled aggregate concrete using explainable machine learning approaches to enhance sustainability in construction materials. A comprehensive review by Shah and Panchal (2022) highlights buckling restrained braces as an effective solution for earthquake-resistant design in industrial structures in India, emphasising their role in mitigating seismic risks through advanced structural engineering approaches. Additionally, Kumar and Agarwal (2022) proposed an optimal residual subspace model tailored for structural damage diagnosis that operates independently of environmental or operational variations, ensuring robust damage detection under diverse conditions.

The goal of Nasrollahi et al.'s (2019) research is to create a technique for identifying concrete surface flaws using a deep neural network (DNN) built on LiDAR scans. In addition, PointNet is modified in this study to identify surface faults using point cloud datasets obtained from scanning bridge surfaces. The trained model was more effective at identifying the deeper flaws. The outcomes were unaffected by increasing the block's density over that of the underlying dataset. A deep learning-based smartphone application was created by Perez and Tah (2021) for the real-time detection of key building flaws, such as cracks, as well as three additional categories of flaws brought on by dampness: mould, stains, and paint deterioration, which include peeling, blistering, flaking, and crazing. Using larger datasets and input images with greater resolutions will allow for further improvement of the reported results. These prior approaches enable quick fixes, minimise catastrophic events, and give a clear picture of the factors to consider. However, the problem with such approaches is that they require a lot of fieldwork and physical effort. Approaches that use deep learning techniques to identify environmental effects on buildings require large real-world datasets which are hard to obtain. In comparison, the proposed transfer learning-based approach requires minimal training data and can be generalisable to different building domain settings with minimal user intervention. The proposed research on using transfer learning to build defect identification aligns with cutting-edge techniques in related domains regarding the role machine learning plays in data processing and urban infrastructures. The integration of vision transformers with CNNs for efficient traffic flow forecasting in urban areas (Ramana et al., 2023), the application of deep reinforcement learning to improve smart building operations (Alhamed et al., 2022), and the use of natural language processing (NLP) for concise legal document summarisation (Md et al., 2023) are all examples of how machine learning and deep learning has the potential to revolutionise various fields. These methodologies not only streamline complex data analysis but also pave the way for more efficient, safe, and sustainable urban environments.

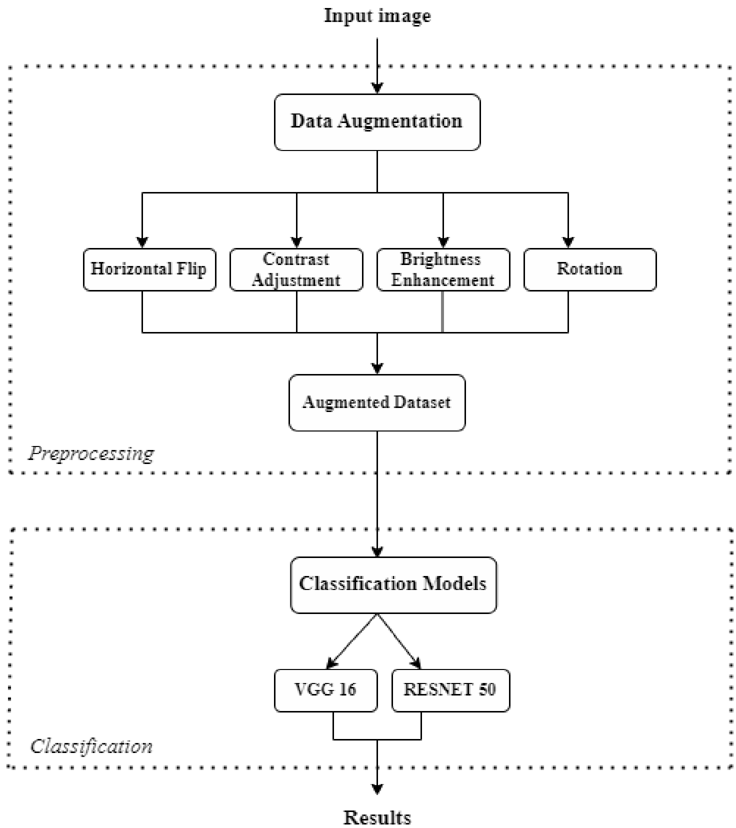
3 Proposed methodology

This section contains an overview of the proposed work, the overall flow chart, data augmentation methods and the architecture of two different deep learning models based on CNN are discussed.

3.1 Overall approach

Figure 2 shows the overall process flow of our approach to identifying unnoticed defects in buildings, such as cracks, flakes, and roofs. The input images in the building defect dataset are used which consists of different types of images that showcase the building faults. To enable high accuracy in the detection of building faults, the images are enhanced by several methods before being fed into the classification model. The images are augmented by horizontal flipping, adjusting contrast, improving brightness, and rotating the image for the classification model. The classification model receives the augmented images, and the models predict the classes. Two classification model approaches are presented, one that uses a VGG16-CNN and the other a ResNet50-CNN-based architecture. This model architecture is fine-tuned and trained on the building defect dataset to accurately detect building faults. The final results are obtained from the output of these models, which are then thoroughly evaluated using a set of performance measures. This consists of accuracy, precision, recall, F1-score and mean intersection over union (mIoU) that depict each model's predictive capabilities. By leveraging these metrics, one can effectively measure the model's capability to not only recognise the presence of defects but also to accurately identify and classify the damage observed in the building images.

Figure 2 Proposed work



The algorithm for classification is discussed as follows:

Algorithm 1

Input: Building defect image I

Output: Class C of the image I

Create an instance of pre-trained VGG16 architecture (or) ResNet50 architecture
Split the building defects training dataset D_m into D_{mT} training and D_{mV} validation set. D_{mt} exists separately already as a test set.
Allot 80% of images of D_m to D_{mT} for training the classification model
Allot 20% of images of D_m to D_{mV} for validating the classification model
for $\forall I_i \in D_{mT}$ **do**
    Perform supervised training of classification using Adam optimiser
end for
Validate images in D_{mV} using the trained model
Choose the test input defect image I belonging to D_{mt}
Resize the I to 224 × 224 × 3 pixels
Use the trained classification model to classify the defect image

3.2 Data augmentation

The building defects dataset consists of 458 images (4,032 × 3,024 pixels) (Zhang et al., 2019) that showcase different types of building defects. However, even though the input images are high resolution, the image in the real world may not be in focus, may have incorrect image orientation (such as tilted or rotated), or may be too bright or too dark based on the lighting conditions. In addition, the goal is to prevent overfitting in the proposed model approach based on the provided dataset. Given that the dataset consists of a limited number of images (458 images in total across different classes), any image trained on this dataset would likely have the problem of overfitting. This problem occurs because if the model is evaluated on a new dataset that does not contain the exact patterns in the original data, it may perform poorly as it needs to be trained to recognise the patterns in the new dataset. The data augmentation approaches can help address this problem by generating new training examples similar to, but not exactly the same as, the original training examples. This approach allows the model to learn more robust features that are less reliant on specific details of the training data and can generalise better with new data. The proposed approach is generalisable to different situations, and different data augmentation techniques are performed as mentioned below. The augmented images are shown in Figure 3.

3.2.1 Horizontal flip

A data augmentation technique is performed to flip the rows and columns of such a matrix horizontally. As a result, the image will be horizontally flipped along the y-axis, as shown in equation (1). A horizontal flip is performed to rotate the images in different patterns to obtain different examples of inverted images.

$$f'(X, Y) = (-X, Y) \tag{1}$$

3.2.2 Linear contrast

Contrast enhancement techniques modify the relative brightness and darkness of the objects in the image to make them more noticeable. This improves the color and the lighting differences between its various components are adjusted. Linear contrast is adjusted by scaling each pixel, as shown in equation (2).

$$127 + \alpha * (p - 127) \tag{2}$$

where p is the pixel value and α is sampled uniformly from the interval [x, y], x and y represent the numerical values.

3.2.3 Scaling

The value of each pixel is modified by a constant to vary the brightness of an image. The image becomes brighter by adding a positive constant to each of the image's pixel values, as shown in equation (3). The picture can also be made darker by subtracting a positive constant from each pixel value.

$$A=\begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_n \end{bmatrix} \qquad A = k * A \qquad Z=\begin{bmatrix} ka_1 & ka_2 & ka_3 \\ ka_4 & ka_5 & ka_6 \\ ka_7 & ka_8 & ka_n \end{bmatrix} \tag{3}$$

where A is the original matrix of the image, k is the constant, which is multiplied to enhance the brightness of the image, and Z is the final output of the enhanced image.

3.2.4 Rotation

In the frame, an object's position is changed by rotating a source image randomly, either clockwise or anticlockwise. The images are rotated in the range [–15, 15]. The centre of the image is represented in equation (4).

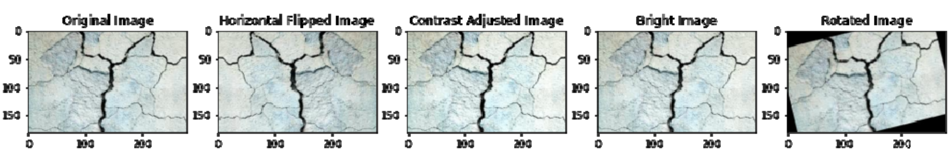
$$D_x = \frac{width}{2}; \quad D_y = \frac{Height}{2} \tag{4}$$

The transformation matrix is required to produce the new point as given in equation (5).

$$S=\begin{bmatrix} P & Q & (1-P)D_x - QD_y \\ -Q & P & QD_x + (1-P)D_y \end{bmatrix} \tag{5}$$

where $P = scale * \cos\theta$, $Q = scale * \sin\theta$, and θ represents the rotation angle.

Figure 3 Images after augmentation (see online version for colours)



3.3 Classification model

Several deep learning-based model architectures are explored and eventually, our model is built based on VGG16 CNN and ResNet50-based model approach architectures primarily due to their promise in image recognition and object recognition tasks.

The proposed approach uses a pre-trained VGG16 CNN-based model which trained on 1.2 million images for identifying 1,000 variant categories (Kumar and Agarwal, 2022) and 1.2 million pictures to categorise images into 1,000 separate groups. The VGG16 model approach extracts the visual features, and the CNN model is used for final-model prediction to different classes. Since the features generated from VGG16 are specific to images, this approach is extended to the setting since the input source is also images from various building defects. Consequently, in the combined approach, the image-building dataset is analysed utilising pre-trained layers from VGG16 and extracting visual characteristics for prediction. To predict three classes of interest: cracks, flakes, and roofs, a fully connected neural network layer and the output layer are built. Eventually, backpropagation is used to train the layers. The proposed model approaches are discussed in detail as to how the classification approach can be used to detect different building defects accurately. In this approach, ResNet50 is used. ResNet50 is trained on the ImageNet dataset, which is a large-scale image classification dataset with over 1 million annotated images and 1,000 classes. The pseudocode of the approach is given as follows:

Pseudocode 1

1. **Input:** A set of images labelled P = $\{p_1, p_2, \ldots, p_n\}$ with dimensions (224 × 224 × 3) and their corresponding class labels Q = $\{q_1, q_2, \ldots, q_3\}$.
2. Initialise the VGG16 and ResNet50 pre-trained weights as the model's initial parameters. The VGG16 and ResNet50 model is a deep CNN architecture built on the ImageNet dataset.
3. Extract feature vectors from the input images:
   - VGG16
     a. Pass the input images through the convolutional and pooling layers of the VGG16 model to obtain a set of feature vectors C = $\{c_1, c_2, \ldots, c_n\}$ with dimension (1 × 1 × 4,096).
     b. This can be represented mathematically as: c_i = VGG16(x_i) for i =1, 2, …, n
   - ResNet50
     a. Pass the input images through the convolutional, pooling and residual layers of the ResNet50 model to obtain a set of feature vectors F = $\{f_1, f_2, \ldots, f_n\}$ with dimension (1 × 1 × 2,048).
     b. This can be represented mathematically as: f_i = ResNet50(x_i) for i = 1, 2, …, n
4. Apply a fully connected layer to the output feature vector to obtain the final class scores.
5. Apply a SoftMax activation function to the class scores to obtain the final class probabilities.
6. **Output:** The class label with the highest probability.

3.4 Visual geometry group (VGG16)

VGG16 is a convolutional neural network trained using data from a portion of the ImageNet dataset, which consists of approximately 14 million pictures divided into 22,000 categories. This model was introduced by Simonyan and Zisserman (2014) and the pre-trained network could categorise photos into 1,000 different classes. As a result, the network has collected detailed visual features for a range of images. The system accepts RGB images with a size of 224 by 224. The model has 3 × 3 convolutional layers with stride 1 and the same padding followed by 2 × 2 max pool layers with stride 2. This pattern is followed by 3 fully connected layers at the end, followed by a culminating softmax layer. The architecture diagram of VGG16 is shown in Figure 4. For better accuracy, the VGG 16 has been modified using a transfer learning approach. The use of an ML model that has already been trained to carry out a related job to accelerate development is known as transfer learning. Even though the model is trained on a smaller dataset, it still helps in achieving improved performance. The trained model already has information about characteristics that could be significant for the present classification problem.

- to do this, all the convolutional blocks must be frozen, the completely trained fully connected layers must be discarded

- new fully connected layers must be initialised with weights randomly that will be modified through backpropagation during training.

The transfer learning is achieved using the feature extraction approach. In this technique, a new dataset is generated from the input images using the architecture of the pre-trained model. So, the convolutional and pooling layers are imported, leaving out the 'top portion' which is the fully connected layer. The images will be sent through the convolutional layers of VGG16, which will provide a feature stack containing the identified image features. After this, flatten the three-dimensional feature stack into a NumPy array. The 'top model' layer, i.e., the new fully connected layer is integrated with the pre-trained layers.

3.5 Residual network (ResNet50)

ResNet refers to a specific type of CNN by the term of residual network. It was first introduced by He et al. (2016). A 50-layer convolutional neural network is called ResNet50 and it has 48 convolutional layers, 1 MaxPool layer, and 1 average pool layer. ResNet architecture is shown in Figure 5. ResNet architecture follows two basic design concepts.

1 there are the same amount of filters for each layer irrespective of the size of the output feature map

2 to maintain the temporal complexity of each layer even if the size of the feature map is half, it has twice as many filters.

The building block of the 50-layer ResNet has a bottleneck design. The number of parameters and matrix multiplications is reduced by a 'bottleneck' residual block that uses 11 convolutions. This significantly speeds up the training process for each layer. Instead of using simply two layers, it utilises a stack of three. Setting the top layer to false allows for the addition of custom input and output layers to an issue. The weights argument instructs the model to train on the ImageNet dataset using its weights. The model does not have to learn its weights again because of the loop on its layers, which also conserves time and space.

3.6 Convolutional neural networks

The fully connected layer in this project utilised convolutional neural networks (CNN), a type of neural network capable of extracting significant information from both time series and image data. For image-related tasks like object categorisation, pattern identification, and picture recognition, it is quite beneficial. Through matrix multiplication and other concepts from linear algebra, CNN offers a more scalable method for performing picture classification and object identification tasks. In this paper, VGG16 and ResNet50 are used to obtain the visual features, and the custom-built CNN model is used to predict the classes. The first layer in a four-layered CNN is known as the convolutional layer, where most of the processing takes place. This layer computes the output of nodes that are linked to specific regions in the input matrix. A collection of weights and values related to a certain area of the input are determined as dot products. The output volume of the convolution layer is given in equation (6).

$$Nout = \frac{N - S + 2Q}{F} + 1 \tag{6}$$

where N is the input size of the image, S is the spatial size, F represents stride and Q is the amount of padding.

After the convolutional layer, the output volume is typically passed through a rectified-linear unit (ReLu) activation function. The ReLu layer decides if an input node will produce an output based on the input data. This output indicates whether the convolution layer has discovered a visual characteristic. The ReLu layer does not alter the dimensions of the output volume. The function is computed as given in equation (7).

$$f(k) = \max(0, k) \tag{7}$$

The activation function represented is merely a threshold at zero.

The pooling layer, which aggregates statistics from surrounding outputs, takes the place of the network output at specific locations, reducing the spatial size of the representation and thus reducing the computational and weight requirements. The pooling process is applied to each slice of the representation individually. The size of the volume output from the pooling layer is given in equation (8).

$$Mout = \frac{M - S}{F} + 1 \tag{8}$$

where M is the activation map size, S is the spatial size of the pooling kernel, and F represents stride.

The final output volume, known as 'convolved features', is sent to a layer of fully connected nodes. These nodes are activated using the ReLu function and are positioned before the classification output in a CNN. They are used to flatten the results for classification purposes. The class probabilities are then calculated and outputted in a 3D array with specified dimensions.

$$1 * 1 * L \tag{9}$$

where L represents the number of classes.

The final layer, softmax, assigns probabilities in the form of decimals to each class in a multi-class problem. These probabilities must sum to 1.0, which accelerates the convergence of training. Softmax is integrated as a neural network layer immediately before the output layer.

$$\sigma(\vec{z})_i = \frac{e^{Zi}}{\sum_{j=1}^{K} e^{Zj}} \tag{10}$$

where σ is softmax, $\vec{z}$ is the input vector, e^{Zi} is the standard exponential function for the input vector, K is the number of classes in the multi-class classifier and e^{Zj} is the standard exponential function for the output vector.

Our proposed model is optimal for its real-time applicability and cost-efficiency in structural defect detection. It utilises the sophisticated architectures of VGG16 and ResNet50, optimised for high accuracy, essential in real-time applications. The model leverages transfer learning for quick image processing and classification, capable of delivering results in seconds, aligning with the practical demands of routine building inspections. Enhanced with image processing techniques like vertical flipping, contrast adjustments, and brightness improvements, the model ensures adaptability and reliability across various scenarios. The model's adoption of transfer learning not only enhances its real-time functionality by reducing training time but also increases cost efficiency by minimising the need for extensive data and computational resources. This, combined with the automation of the defect detection process, significantly reduces labour costs and the potential for human error in manual inspections. Additionally, its capability for early detection of defects such as cracks and flakes prevents minor issues from escalating into costly repairs, enhancing overall cost-effectiveness. In summary, the proposed model is both practical and cost-efficient for real-time building maintenance and defect detection.

Figure 4 VGG16 architecture (see online version for colours)

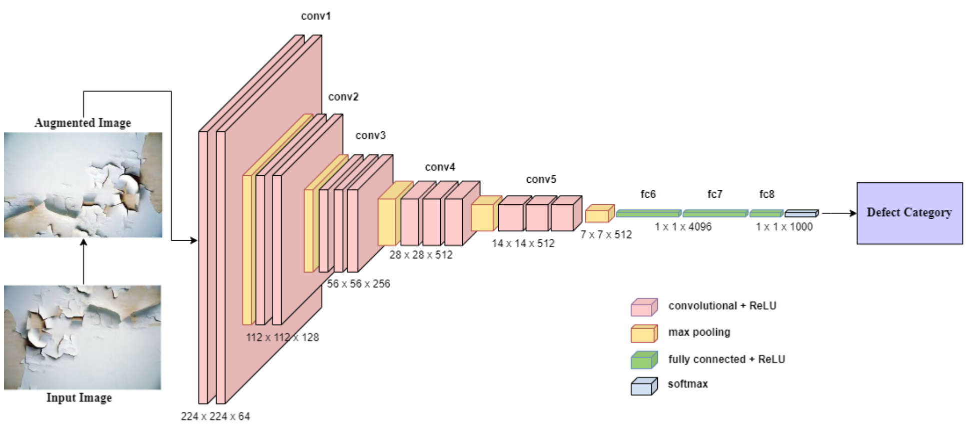
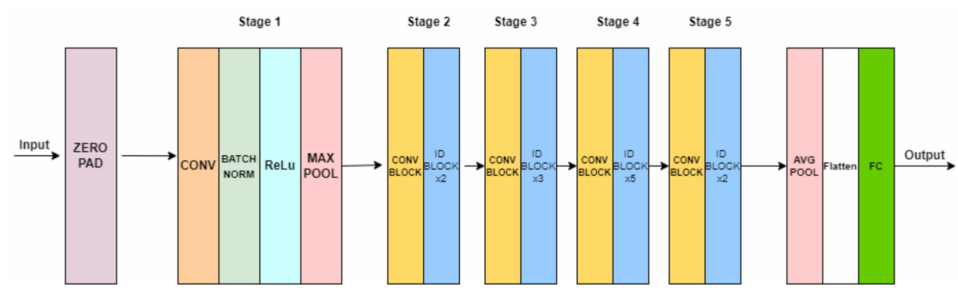


Figure 5 ResNet50 architecture (see online version for colours)



4 Results and evaluation

4.1 Dataset

The datasets contain images of various categories from cracks, flakes, and roofs. The image data is divided into three parts in a separate folder for image classification. There are a total of 436 images as seen in Table 1 with a horizontal and vertical resolution of 96 dpi. Data augmentation is applied in terms of random rotation, flipping, and contrast adjustment. It helps to improve the quantity and quality of the dataset. After augmentation, the total data used for training and testing purposes is 1,744 images as shown in Table 2.

The dataset is categorised into three groups in Table 1: cracks, flakes, and roofs. The cracks category has 118 images, the flakes category has 227 images, and the roofs category has 91 images. The dataset has 436 images in total before augmentation. Table 2 displays data statistics after augmentation and is divided into two sets: training and testing. There are 472 photos in the cracks category, with 440 in the training set and 32 in the test set. The flakes category includes 908 images, with 857 in the training set, and 51 in the test set. The category roof includes 364 images, with 327 in the training set, and 37 in the test set. After augmentation, the total number of images in the dataset has increased to 1,744, with 1,624 images in the training set and 120 in the test set.

The images in the dataset categorised as cracks, flakes, and roofs, had an initial size of 278 × 278 for 'cracks', 306 × 306 for 'flakes' and 612 × 612 for 'roof', as shown in Tables 1 and 2. Considering these variations, standardising the image dimensions is essential in ensuring consistent feature extraction and pattern identification by the CNN models, VGG16 and ResNet50. As a result, all images were scaled to the standard input size of 224 × 224 pixels, a resolution that is consistent with the architecture of both models employed. This standardisation step is a necessary preprocessing approach to facilitating rapid model training and reliable performance evaluation, ensuring consistency across different image categories while leveraging the inherent strengths of VGG16 and ResNet50 models.

Table 1 Dataset statistics

Image category	*Count*	*Image size*
Cracks	118	278 × 278
Flakes	227	306 × 306
Roof	91	612 × 612
Total	436	-

Table 2 Data statistics after augmentation

Image category	*Count*		*Image size*
	Training set	*Test set*	
Cracks	440	32	278 × 278
Flakes	857	51	306 × 306
Roof	327	37	612 × 612
Total	1,624	120	-

4.2 Evaluation metrics

Evaluation metrics like accuracy, recall, precision, F1-score and mean intersection over union (mIoU) are used for the evaluation of the proposed model approach. The accuracy of a model is determined by dividing the number of accurate predictions by the total number of predictions made. The accuracy of images is determined by dividing the number of pixels properly detected over all the pixels. The accuracy of the image is given in equation (11).

$$Accuracy = \frac{Pixels\ detected\ correctly}{Total\ \ pixels} \tag{11}$$

Precision reflects the proportion of positively identified instances among the total number of instances that were correctly predicted by the model. The ability of a model to represent just accurate pixels is known as precision. Precision is calculated in equation (12).

$$Precision = \frac{True\ positive}{True\ positive + False\ positive} \tag{12}$$

Recall, also known as sensitivity or true positive rate, is a measure of the model's ability to identify all positive cases in the data. The recall is defined as the ratio of the number of positive cases that the model correctly identified to the total number of actual positive cases. It is a metric that quantifies the model's ability to correctly identify all relevant pixels of a specific class. The recall is represented in equation (13).

$$Recall = \frac{True\ positive}{True\ positive + False\ negative} \tag{13}$$

F1-score is a combination of recall and precision, calculated as the harmonic mean of the two. It is given in equation (14).

$$F1\text{-}score = 2\left(\frac{Precision * Recall}{Precision + Recall}\right) \tag{14}$$

Intersection over union (IoU) metric is used to measure object detection accuracy on a dataset. The IoU is determined for each class c as the ratio of the area of overlap (true positives) between the predicted and actual annotations to the area of union (the total of true positives, false positives and false negatives) (Kang and Cha, 2022).

For each defect class *c*, the IoU is calculated as follows [equation (15)]:

$$IoU_c = \frac{True\ positive_c}{True\ positive_c + False\ positive_c + False\ negative_c} \tag{15}$$

The mean intersection (mIoU) is the average IoU across all classes C in the dataset, providing a single performance summary statistic [equation (16)].

$$mIoU = \frac{1}{|C|}\sum_{c \in C} IoU_c \tag{16}$$

In equation (15), c denotes an individual class for which the IoU is being calculated. In equation (16), $|C|$ represents the total number of classes in the dataset.

The IoU metric provides an objective measure of the overlap between predicted and actual annotations for each class, and the mIoU averages the IoU across classes, providing a balanced perspective that accounts for dataset class imbalances.

4.3 Implementation

The networks are built using Keras from Tensorflow and implemented in Python. The full topic is run on Kaggle's IPython notebooks with an NVIDIA Tesla P100 GPU. ResNet50 took 11 minutes to train over 1,624 images, whereas VGG16 took about 4 minutes.

Table 3 Hyperparameters of VGG16 and ResNet50

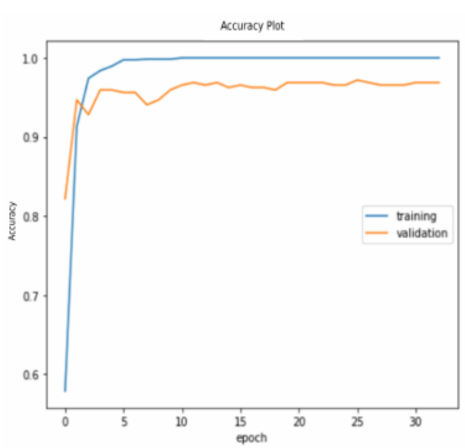
Model	*Batch size*	*Learning rate*	*Epochs*
VGG16	64	0.001	100
ResNet50	32	0.001	100

The Adam optimiser is utilised for both VGG16 and ResNet50. The Adam optimiser procedure is used to update network weights iterative based on training data. When dealing with complex problems involving a lot of data, the strategy is extremely effective and uses little memory. Adam optimiser's learning rate is set to 0.001 for both models.

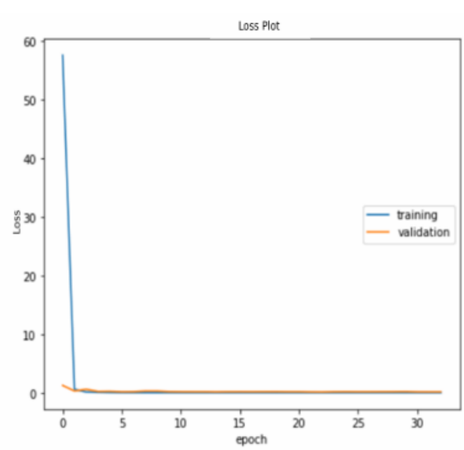
4.4 Results

Figure 6(a) represents the accuracy curve of the training and validation set. The small gap between training and validation accuracy indicates that the model is slightly overfitting. Figure 6(b) includes the loss curve with respect to the epochs of the training and validation set. The loss plot shows that the given VGG16 model is a good fit as the training and validation curves decrease at a point of stability and as there is only a small gap between the training and validation learning curves, these gaps are referred to as 'generation gap'.

Figure 6 Training and validation statistics for VGG16, (a) accuracy plot (b) loss plot (see online version for colours)

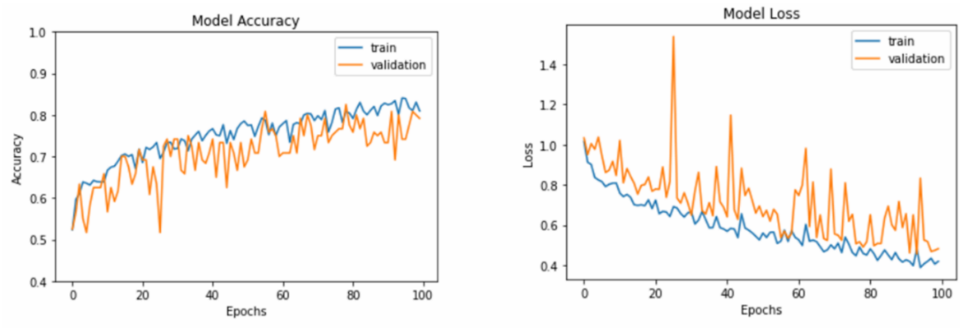


(a)



(b)

Figure 7 Training and validation statistics for ResNet50, (a) accuracy plot (b) loss plot (see online version for colours)



(a) (b)

In Figure 7(a), the accuracy plot shows that there is uncertainty in the model and since there is a large gap between the training and validation curves, the model seems to be overfitting and uncertain. Figure 7(b) represents the model loss plot, this shows that the train data is unrepresentative. An unrepresentative training set means that the training dataset does not provide sufficient information to learn the problem, relative to the validation dataset used to evaluate it.

Table 4 Results of evaluation metrics from the testing set

Model	*Accuracy*	*Precision*	*Recall*	*F1-score*
VGG16	0.9833	0.9833	0.9833	0.9833
ResNet50	0.7916	0.8157	0.7749	0.7947

Table 5 Results of evaluation metrics from the validation set

Model	*Accuracy*	*Precision*	*Recall*	*F1-score*
VGG16	0.9660	0.9660	0.9660	0.9660
ResNet50	0.7812	0.8201	77.54	0.7971

Table 6 Results of evaluation metrics for intersection over union (IoU) and mean intersection over union (mIoU)

Image category	*IoU (VGG16)*	*IoU (ResNet50)*
Cracks	0.9688	0.7356
Flakes	0.9423	0.7491
Roofs	0.9487	0.7403
mIoU	0.9533	0.7417

From these plots, it can be inferred that VGG16 performs well compared to ResNet50. This further can be verified using the results from Tables 4 and 5 evaluation metrics of the testing and validation dataset. From Table 4, results can be seen that VGG16 has an accuracy rate of 98.33% and ResNet50 has 79.16% for the testing. VGG16 model outperforms the ResNet50 model. Evaluation metrics for VGG16 show the F1 score is 0.98 which is close to 1. This shows how good the model is and the F1 score, and accuracy produce the same number, this is because when the dataset is balanced. The precision and recall have the same value of 0.98, due to the micro-averaging. ResNet50 has an F1 score of 0.79 which is slightly closer to 1 and precision score of 0.81 and a recall of 0.77. Table 5 gives information about the validation data accuracy of 96.60% for VGG16 and 78.12% for ResNet50. Table 6 gives information about mIoU metrics, the VGG16 is consistently high across all defect categories, resulting in a mIoU of 0.9533. In comparison, ResNet50's mIoU is 0.7417, while significant, implies space for improvement, particularly in terms of obtaining consistent segmentation across various fault categories. The gap in performance highlights VGG16's robustness and potential suitability for deployment in automated building defect detection systems.

Figure 8 Confusion matrix for VGG16 (see online version for colours)

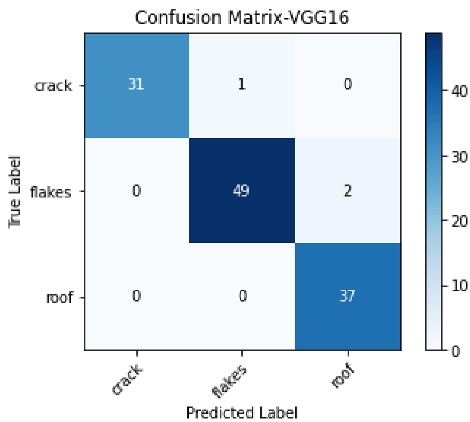
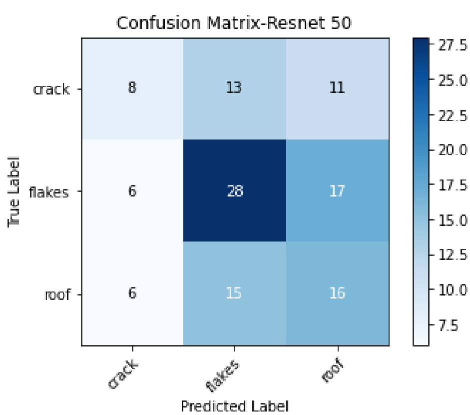


Figure 9 Confusion matrix for ResNet50 (see online version for colours)



The confusion matrix is used to evaluate the performance algorithm. It shows whether the data are classified accurately or not. In Figure 8, it can be inferred that the VGG16 algorithm has been classified accurately as they have larger values down the diagonal for cracks, flakes, and roofs and only a very few images were predicted wrongly. Whereas in

Figure 9, it can be seen that the ResNet50 performed poorly by detecting some of the flaws inaccurately as the lower amount of values is seen down the diagonal.

While the proposed model is quite effective at identifying certain flaws like flakes, cracks, and roofs, it has a few limitations that will be addressed in future research. It focuses on evident defects, sometimes overlooking subtle flaws that might get worse with time. Future work will focus on increasing the model's sensitivity to identify various flaws, including subtle ones. Furthermore, the model can only analyse a single defect type at a time, which limits its capacity to identify multiple defect types simultaneously. Future research aims to enhance the model's versatility by expanding its capability to identify a broader range of construction problems, such as fracture, structural movement, spalling, and corrosion.

5 Conclusions and future work

This paper presents a transfer learning-based approach for robust and accurate detection of unnoticed building defects, such as cracks, flakes, and roofs. The proposed methodology utilises fine-tuned VGG16 and ResNet50 models and has been rigorously tested using real-world images obtained from onsite deployments. The VGG16 model achieved a superior performance, with an accuracy of 98.33% on the testing data and 96.60% on the validation data. In contrast, the ResNet50 model attained 79.16% accuracy on the testing data and 78.12% on the validation data. These results indicate that the VGG16 model more effectively identifies building defects in this context. Additionally, the models were evaluated using key metrics such as F1-score, precision, recall and mIoU, where VGG16 consistently outperformed ResNet50, confirming its robustness and applicability for real-world defect detection. A detailed analysis of training performance demonstrated that the VGG16 model converged more rapidly and effectively over 100 epochs, showcasing its efficiency in learning from the augmented dataset of 1,744 images.

The results also highlight the efficacy of the transfer learning-based approach in achieving high accuracy while maintaining computational efficiency, making it a viable option for practical applications in construction site inspections. Moreover, the segmentation outputs produced by the VGG16 model demonstrated its capability to handle diverse real-world scenarios, even in complex background settings. Despite these achievements, certain constraints were identified in the current research. Firstly, the model examines defects from a single category at a time, limiting its applicability to scenarios involving multiple types of flaws simultaneously. Secondly, the dataset comprises only images with clearly visible defects, which may restrict the model's performance in cases of subtle or hidden damage. These limitations present opportunities for future work.

In subsequent research, the focus will be on addressing these constraints by developing models capable of identifying multiple defect categories concurrently. Further advancements will include exploring defect severity estimation to enhance the practical utility of automated inspections. Efforts will also be directed towards creating real-time software applications for visual inspections, leveraging drones and other autonomous devices for broader scalability. Expanding the scope of this research to detect additional construction issues such as fractures, structural movements, spalling and corrosion will also be prioritised. This comprehensive evaluation and the promising results of this study underscore the potential of transfer learning-based models in transforming building defect detection and inspection processes. By addressing the outlined limitations, future research can further enhance these models' accuracy, scalability, and usability in real-world applications.

Data availability

Data generated or analysed during this study are available from the corresponding author upon reasonable request.

References

Agdas, D., Rice, J.A., Martinez, J.R. and Lasa, I.R. (2016) 'Comparison of visual inspection and structural health monitoring as bridge condition assessment methods', *Journal of Performance of Constructed Facilities*, Vol. 30, No. 3, p.04015049.

Alhamed, K.M., Iwendi, C., Dutta, A.K., Almutairi, B., Alsaghier, H. and Almotairi, S. (2022) 'Building construction based on video surveillance and deep reinforcement learning using smart grid power system', *Computers and Electrical Engineering*, Vol. 103, No. 1, p.108273.

Cha, Y.J., Choi, W. and Büyüköztürk, O. (2017) 'Deep learning-based crack damage detection using convolutional neural networks', *Computer-Aided Civil and Infrastructure Engineering*, Vol. 32, No. 5, pp.361–378.

Cha, Y.J., Choi, W., Suh, G., Mahmoudkhani, S. and Büyüköztürk, O. (2018) 'Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types', *Computer-Aided Civil and Infrastructure Engineering*, Vol. 33, No. 9, pp.731–747.

Chen, J., Lu, W. and Liu, D. (2024) 'Built environment defect mapping, modeling, and management (D3M): a BIM-based integrated framework', *Journal of Intelligent Construction*, Vol. 2, No. 1, pp.1–15.

Choi, W. and Cha, Y.J. (2020) 'SDDNet: real-time crack segmentation', *IEEE Transactions on Industrial Electronics*, Vol. 67, No. 9, pp.8016–8025.

Davoudi, R., Miller, G.R. and Kutz, J.N. (2018) 'Structural load estimation using machine vision and surface crack patterns for shear-critical RC beams and slabs', *Journal of Computing in Civil Engineering*, Vol. 32, No. 4, p.04018024.

Dung, C.V. (2019) 'Autonomous concrete crack detection using deep fully convolutional neural network', *Automation in Construction*, Vol. 99, No. 1, pp.52–58.

Feng, Y., Zhang, X., Feng, S., Chen, H., Zhao, Y. and Chen, Y. (2023) 'Improved SOLOv2 detection method for shield tunnel lining water leakages', *Journal of Intelligent Construction*, Vol. 1, No. 1, p.9180004.

German, S., Brilakis, I. and DesRoches, R. (2012) 'Rapid entropy-based detection and properties measurement of concrete spalling with machine vision for post-earthquake safety assessments', *Advanced Engineering Informatics*, Vol. 26, No. 4, pp.846–858.

Haidarpour, A. and Tee, K.F. (2020) 'Finite element model updating for structural health monitoring', *Structural Durability and Health Monitoring*, Vol. 14, No. 1, pp.1–17.

He, K., Zhang, X., Ren, S. and Sun, J. (2016) 'Deep residual learning for image recognition', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.770–778.

Hoang, N.D. (2018) 'Image processing-based recognition of wall defects using machine learning approaches and steerable filters', *Computational Intelligence and Neuroscience*, Vol. 2018, pp.1–12.

Jiang, Y., Han, S. and Bai, Y. (2021) 'Building and infrastructure defect detection and visualization using drone and deep learning technologies', *Journal of Performance of Constructed Facilities*, Vol. 35, No. 6, Article ID 04021092.

Kang, D.H. and Cha, Y.J. (2022) 'Efficient attention-based deep encoder and decoder for automatic crack segmentation', *Structural Health Monitoring*, Vol. 21, No. 5, pp.2190–2205.

Kessai, I., Benammar, S., Doghmane, M.Z. and Tee, K.F. (2022) 'Estimation of circular arc crack depths and locations in rotary drilling pipes subjected to free vibrations', *Vibration*, Vol. 5, No. 1, pp.165–182.

Koch, C. and Brilakis, I. (2011) 'Pothole detection in asphalt pavement images', *Advanced Engineering Informatics*, Vol. 25, No. 3, pp.507–515.

Koh, C.G., Tee, K.F. and Quek, S.T. (2006) 'Condensed model identification and recovery for structural damage assessment', *Journal of Structural Engineering, ASCE*, Vol. 132, No. 12, pp.2018–2026.

Kumar, K. and Agarwal, S. (2022) 'Optimal residual subspace model for structural damage diagnosis: an approach independent of operational and environmental variations', *International Journal of Structural Engineering*, Vol. 12, No. 1, pp.44–61.

Lorenzoni, F., Casarin, F., Caldon, M., Islami, K. and Modena, C. (2016) 'Uncertainty quantification in structural health monitoring: applications on cultural heritage buildings', *Mechanical Systems and Signal Processing*, Vol. 66, No. 1, pp.268–281.

Md, A.Q., Anand, R.V., Mohan, S., Joshua, C.J., Girish, S.S., Devarajan, A. and Iwendi, C. (2023) 'Data-driven analysis of privacy policies using LexRank and KL summarizer for environmental sustainability', *Sustainability*, Vol. 15, No. 7, p.5941.

Minhas, M. and Zelek, J. (2020) 'Defect detection using deep learning from minimal annotations', *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications – Volume 4: VISAPP*, pp.506–513, ISBN 978-989-758-402-2, DOI: 10.5220/0009168005060513.

Mohseni, H., Setunge, S., Zhang, G.M. and Wakefield, R. (2013) 'Condition monitoring and condition aggregation for optimised decision making in management of buildings', *Applied Mechanics and Materials*, Vol. 438, No. 1, pp.1719–1725, Trans Tech Publications Ltd.

Nasrollahi, M., Bolourian, N., Hammad A. (2019) 'Concrete surface defect detection using deep neural network based on lidar scanning', *Proc. of the CSCE Annual Conference*, Laval, Greater Montreal, QC, Canada, pp.12–15.

Oh, B.K., Kim, K.J., Kim, Y., Park, H.S. and Adeli, H. (2017) 'Evolutionary learning based sustainable strain sensing model for structural health monitoring of high-rise buildings', *Applied Soft Computing*, Vol. 58, No. C, pp.576–585.

Özgenel, Ç.F. and Sorguç, A.G. (2018) Performance comparison of pretrained convolutional neural networks on crack detection in buildings', *ISARC: Proceedings of the International Symposium on Automation and Robotics in Construction*, IAARC Publications, Vol. 35, pp.1–8.

Perez, H. and Tah, J.H. (2021) 'Deep learning smartphone application for real-time detection of defects in buildings', *Structural Control and Health Monitoring*, Vol. 28, No. 7, p.e2751.

Perez, H., Tah, J.H. and Mosavi, A. (2019) 'Deep learning for detecting building defects using convolutional neural networks', *Sensors*, Vol. 19, No. 16, p.3556.

Pragalath, H., Seshathiri, S., Rathod, H., Esakki, B., and Gupta, R. (2018) 'Deterioration assessment of infrastructure using fuzzy logic and image processing algorithm', *Journal of Performance of Constructed Facilities*, Vol. 32, No. 2, pp.1–13, DOI: 10.1061/(ASCE)CF. 1943-5509.0001151.

Qu, Z., Mei, J., Liu, L. and Zhou, D.Y. (2020) 'Crack detection of concrete pavement with cross-entropy loss function and improved VGG16 network model', *IEEE Access*, Vol. 8, pp.54564–54573, DOI: 10.1109/ACCESS.2020.2981561.

Ramana, K., Srivastava, G., Kumar, M.R., Gadekallu, T.R., Lin, J.C.W., Alazab, M. and Iwendi, C. (2023) 'A vision transformer approach for traffic congestion prediction in urban areas', *IEEE Transactions on Intelligent Transportation Systems*, Vol. 24, No. 4, pp.3922–3934.

Salehi, H. and Burgueño, R. (2018) 'Emerging artificial intelligence methods in structural engineering', *Engineering structures*, Vol. 171, No. 1, pp.170–189.

Shah, B.S. and Panchal, V.R. (2022) 'Review of buckling restrained braces for earthquake resistant design of industrial structures in India', *International Journal of Structural Engineering*, Vol. 12, No. 1, pp.77–102.

Shamshirband, S., Mosavi, A. and Rabczuk, T. (2020) 'Particle swarm optimization model to predict scour depth around a bridge pier', *Frontiers of Structural and Civil Engineering*, Vol. 14, No. 4, pp.855–866.

Simonyan, K. and Zisserman, A. (2014) *Very Deep Convolutional Networks for Large-Scale Image Recognition*, arXiv preprint arXiv:1409.1556.

Tee, K.F. (2018) 'Time series analysis for vibration-based structural health monitoring: a review', *Structural Durability and Health Monitoring*, Vol. 12, No. 3, pp.129–147.

Tee, K.F., Cai, Y. and Chen, H.P. (2013) 'Structural damage detection using quantile regression', *Journal of Civil Structural Health Monitoring*, Vol. 3, No. 1, pp.19–31.

Tee, K.F., Koh, C.G. and Quek, S.T. (2003) 'Substructural identification with incomplete measurement for damage assessment', *Proc. of the 1st International Conference on Structural Health Monitoring and Intelligent Infrastructure*, 13–15 November, Tokyo, Japan, Vol. 1, pp.379–386.

Valero, E., Forster, A., Bosché, F., Hyslop, E., Wilson, L. and Turmel, A. (2019) 'Automated defect detection and classification in ashlar masonry walls using machine learning', *Automation in Construction*, Vol. 106, No. 1, p.102846.

Wang, S., Xia, P., Gong, F., Zeng, Q., Chen, K. and Zhao, Y. (2024) 'Multi objective optimization of recycled aggregate concrete based on explainable machine learning', *Journal of Cleaner Production*, Vol. 445, No. 1, p.141045.

Wang, T., Zhang, L. and Tee, K.F. (2011) 'Extraction of real modes and physical matrices from modal testing', *Earthquake Engineering and Engineering Vibration*, Vol. 10, No. 2, pp.219–227.

Waqas, A. and Cha, Y.J. (2023) 'Fiducial marker-based localization of autonomous UAV for structural health monitoring', *Society for Experimental Mechanics Annual Conference and Exposition*, Vol. 8, pp.141–147.

Wu, Z., Huang, B., Tee, K.F. and Zhang, W. (2021) 'A novel stochastic approach for static damage identification of beam structures using homotopy analysis algorithm', *Sensors*, Vol. 21, No. 7, p.2366.

Zhang, F.L., Yang, Y.P., Xiong, H.B., Yang, J.H. and Yu, Z. (2019) 'Structural health monitoring of a 250-m super-tall building and operational modal analysis using the fast Bayesian FFT method', *Structural Control and Health Monitoring*, Vol. 26, No. 8, p.e2383.

Zhang, X., Lin, X., Zhang, W., Feng, Y., Lan, W., Da, Y. and Hu, K. (2023) 'Intelligent recognition of voids behind tunnel linings using deep learning and percussion sound', *Journal of Intelligent Construction*, Vol. 1, No. 4, pp.1–18.

Zhang, Y., Kim, C-W. Tee, K.F., Garg, A. and Garg, A. (2018) 'Long-term health monitoring for deteriorated bridge structures based on copula theory', *Smart Structures and Systems*, Vol. 21, No. 2, pp.171–185.