Resource constraint crop damage classification using depth channel shuffling

Islam, Md Tanvir; Swapnil, Safkat Shahrier; Billal, Md Masum; Karim, Asif; Shafiabady, Niusha; Hassan, Md Mehedi

Published in:
Engineering Applications of Artificial Intelligence

DOI:
10.1016/j.engappai.2025.110117

Published: 15/03/2025

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Islam, M. T., Swapnil, S. S., Billal, M. M., Karim, A., Shafiabady, N., & Hassan, M. M. (2025). Resource constraint crop damage classification using depth channel shuffling. *Engineering Applications of Artificial Intelligence*, *144*, 1-19. Article 110117. https://doi.org/10.1016/j.engappai.2025.110117

Engineering Applications of Artificial Intelligence 144 (2025) 110117

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Research paper

Resource constraint crop damage classification using depth channel shuffling

Md Tanvir Islam [a], Safkat Shahrier Swapnil [b], Md. Masum Billal [c], Asif Karim [d],*, Niusha Shafiabady [d,e], Md. Mehedi Hassan [f],*

[a] Department of Computer Science and Engineering, Sungkyunkwan University, Suwon, 16419, Republic of Korea
[b] Rajshahi University of Engineering and Technology, Rajshahi, 6204, Bangladesh
[c] University of Science and Technology of China, Hefei, 230026, China
[d] Charles Darwin University, Darwin, 0909, Australia
[e] Australian Catholic University, Sydney, 2060, Australia
[f] Khulna University, Khulna, 9208, Bangladesh

ARTICLE INFO

Dataset link: GitHub

Keywords:
Crop damage classification
Resource constraint agriculture
Precision agriculture
Crop management
Crop monitoring
Deep learning
LightCDC

ABSTRACT

Accurate crop damage classification is crucial for timely interventions, loss reduction, and resource optimization in agriculture. However, datasets and models for binary classification of damaged versus non-damaged crops remain scarce. To address this, we conducted an extensive study on crop damage classification using deep learning, focusing on the challenges posed by imbalanced datasets common in agriculture. We began by preprocessing the *"Consultative Group for International Agricultural Research (CGIAR)"* dataset to enhance data quality and balance class distributions. We created the new *"Crop Damage Classification (CDC)"* dataset tailored for binary classification of *"Damaged"* versus *"Non-damaged"* crops, serving as an effective training medium for deep learning models. Using the CDC dataset, we benchmarked the state-of-the-art models to evaluate their effectiveness in classifying crop damage. Leveraging the depth channel shuffling technique of ShuffleNetV2, we proposed a lightweight model *"Light Crop Damage Classifier (LightCDC)"*, reducing the parameters from 1.40 million to 1.13 million while achieving an accuracy of 89.44%. LightCDC outperformed existing classification and ensemble models in terms of model size, parameter count, inference time, and accuracy. Furthermore, we tested LightCDC under adverse conditions like blur, low light, and fog, validating its robustness for real-world scenarios. Thus, our contributions include a refined dataset and an efficient model tailored for crop damage classification, which is essential for timely interventions and improved crop management in resource-constrained precision agriculture. To ensure reproducibility, we released the code and dataset on **GitHub**.

1. Introduction

Agriculture is a cornerstone of economic stability and food security in many regions across Africa, Southeast Asia, and South America (Reynolds et al., 2015). It is crucial in reducing poverty and providing livelihood opportunities for rural communities. However, climate change has led to a rise in extreme weather events such as droughts, floods, and irregular precipitation patterns, impacting around 30% of crop yields across different geographic areas (Verschuur et al., 2021). Between 2008 and 2018, the agriculture sector in the least-developed countries and low-and middle-income countries suffered significant losses due to droughts, floods, and other natural disasters, totaling over $108 billion. Asia experienced the most severe impact, with economic losses from droughts and floods reaching $49 billion, an amount comparable to the combined gross domestic product of Cambodia, Laos, and East Timor. Additionally, droughts impacted over 34% of crop and livestock production in these regions, resulting in estimated economic damages of $37 billion (FAO, 2021).

Therefore, investing in sustainable agricultural practices and supporting farmers in these regions is crucial to ensure long-term food security and economic growth. Besides, the health of crops in these regions is critically important, as damage from bacteria, viruses, fungi, and natural disasters can severely affect food availability and economic vitality (FAO, 2021; Ahmad et al., 2022; Sakkarvarthi et al., 2022; Kotwal et al., 2023). Reports suggest that pests and diseases lead to a 20% to 40% reduction in crop output globally (Domingues et al., 2022). In regions such as Sub-Saharan and North Africa, economic losses have been particularly stark, with impacts amounting to $30 billion over a decade due to reduced livestock and grain production following natural disasters (Memon et al., 2022).

* Corresponding authors.
E-mail addresses: asif.karim@cdu.edu.au (A. Karim), mehedihassan@ieee.org (M.M. Hassan).

https://doi.org/10.1016/j.engappai.2025.110117
Received 23 June 2024; Received in revised form 7 December 2024; Accepted 17 January 2025
Available online 29 January 2025

Crop damage presents a significant challenge to both food security and economic stability in regions such as Sub-Saharan Africa, South Asia and America (Raimi et al., 2021; Giller, 2020; Reynolds et al., 2015). The inability to identify crop damage quickly and accurately exacerbates the issue, as many rural farming communities lack access to advanced tools for early detection and intervention (Reynolds et al., 2015). Without sufficient resources, farmers often rely on outdated methods, such as manual inspection or rudimentary technologies, which are inefficient and prone to errors (Thierfelder et al., 2016; Baijukya et al., 2020). This delay in identifying crop damage can lead to further deterioration, reducing yields and exacerbating poverty in vulnerable regions. Moreover, the high cost of advanced technologies, such as precision agriculture tools or large-scale imaging systems, is beyond the reach of smallholder farmers (Ncube et al., 2018). As a result, they struggle to implement timely responses to pests, diseases, or climate-induced damage. This impacts food production and hinders economic growth, as agriculture remains a primary source of income and employment in these areas. Without innovations that provide accessible, affordable, and scalable crop monitoring solutions, the cycle of food insecurity and economic vulnerability will likely persist. The need for more efficient, real-time methods that can be deployed in resource-constraint environments (Liu et al., 2024a) is critical, especially in underdeveloped regions where food security is at risk and access to advanced technology is limited. These challenges emphasize the necessity for lightweight, automated models that can accurately and efficiently identify damaged crops from ground-level views, enabling timely interventions to mitigate crop losses (Malek et al., 2022).

In recent years, there has been growing interest in utilizing artificial intelligence (AI) and machine learning techniques to address complex problems across various fields, such as healthcare (Islam et al., 2020, 2019), natural language processing (Ahammad et al., 2024) and image processing (Alam et al., 2025; Huang et al., 2020; Islam et al., 2024b). These successes have also motivated researchers to adapt these methods for accurately identifying various types of crop-related diseases and problems (Tirkey et al., 2023a; Ali et al., 2023; Meena et al., 2023). These technologies have the potential to revolutionize crop health management by providing timely and precise information on the type and extent of damage, enabling farmers to implement targeted. This can help minimize losses, enhance crop resilience, and contribute to global food security (Ben Ayed and Hanana, 2021). Similarly, early identification of damaged crops can help to take proper steps to reduce the loss of crop damage. However, identifying and classifying crop damage accurately and promptly is a major challenge, primarily due to the labor-intensive nature of traditional methods that require expert knowledge and are error-prone (Thierfelder et al., 2016; Baijukya et al., 2020).

Recent research suggest that deep Convolutional Neural Network (CNN) based models can classify and detect pests and diseases that harm crops (Widén et al., 2022; Rani and Singh, 2022; Singh et al., 2022; Meena et al., 2023; Kotwal et al., 2023). These models and algorithms include custom CNN, pest detection algorithm, meta deep learner, and others (Kamilaris and Prenafeta-Boldú, 2018; Alam et al., 2025; Islam et al., 2025). While these models have produced good outcomes, they have encountered several difficulties. A primary obstacle they have encountered is the need for more pictures in the dataset, as deep learning models need a vast quantity of data for training (Waldamichael et al., 2022; Pun et al., 2023). Another problem is the picture quality and the things present in the image. In addition, massive computing resources are needed for the deep learning models to interpret the pictures and extract meaningful features (MacEachern et al., 2023). At the same time, for precision agriculture, it is essential to develop methods with low computational expense (Yaqot et al., 2023). Besides, there is still a lack of diverse datasets that can be directly used to train the classification models for identifying damaged crops. Most of the recent research are related to crop diseases and pest classification rather than identifying damaged crops.

Considering the lack of datasets and lightweight models in the literature that can classify damaged crops; in this paper, we introduce a meticulously pre-processed dataset derived from the original data provided for the Crop Damage Classification Challenge (CGIAR, 2024) by the Consultative Group for International Agricultural Research (CGIAR). We have tailored the CGIAR dataset to train and evaluate the state-of-the-art (SOTA) classification models, enhancing its utility for deep learning applications in agriculture. Additionally, we propose a lightweight model named Light Crop Damage Classifier (LightCDC) that can effectively identify damaged crops. Thus, our contributions to this study are listed as follows:

- CDC Dataset: We perform preprocessing on the imbalanced CGIAR (CGIAR, 2024) dataset to form *"Crop Damage Classification (CDC)"* dataset for training and testing models for binary classification of Damaged versus Non-damaged crops.
- LightCDC: we proposed a custom model named *"Light Crop Damage Classifier (LightCDC)"* leveraging the depth channel shuffling technique of ShuffleNetv2 (Ma et al., 2018), reducing the parameters from 1.40 million to 1.13 million while achieving an accuracy of 89.44%. LightCDC is ideal for real-time crop damage classification in resource-constraint environments, particularly in underdeveloped agricultural regions.
- Benchmarking the SOTA Models: We also benchmark the SOTA classification models on our processed CDC dataset and performed extensive ablation studies on different vital parameters of the proposed LightCDC to make this research a foundation for future research in this domain.
- Ensemble Techniques: Our extensive experiments on the processed CDC dataset also involve ensemble techniques like Stacking and Voting to observe the performance of these techniques for crop damage classification.
- Focus on Ground-Level Crop Damage Classification: Unlike existing studies that use Unmanned Aerial Vehicles (UAVs) or satellite images, our work focuses on classifying damaged versus non-damaged crops from ground-level images. To the best of our knowledge, no other work has addressed this problem with a lightweight solution optimized for a real-world resource-constraint environment.

The structure of this paper is as follows: Section 2 reviews related works, focusing on machine learning and deep learning applications in various crop-related tasks such as disease and pest detection. Section 3 details the methodology, including the workflow, methods, techniques, and our proposed framework. Section 4 benchmarks SOTA classification models for damaged and non-damaged crop classification using our CDC dataset and evaluates our proposed models. Section 5 discusses our study's limitations and suggests future research directions. Finally, Section 6 concludes the paper.

2. Related works

In recent years, Convolutional Neural Networks (CNNs) have emerged as a powerful tool in precision agriculture, enabling researchers to address various challenges related to crop health, disease detection, and pest management. CNNs are particularly well-suited for image-based tasks, making them an ideal choice for analyzing high-dimensional agricultural data such as leaf images, crop field observations, and drone or satellite imagery. In this section, we reviewed some typical applications of deep CNN-based models in various agricultural domains to understand the current progress.

2.1. Common applications of CNNs in agriculture

One of the most common applications of CNNs in agriculture is leaf disease detection (Bhargava et al., 2024; Sarkar et al., 2023). Researchers have developed CNN-based models that can automatically identify diseases in crops by analyzing images of leaves (Karim et al., 2024). For instance, studies have focused on detecting diseases in crops such as wheat, rice, and maize, where CNNs have achieved high accuracy in distinguishing between healthy and diseased leaves (Jiang et al., 2021; Xu et al., 2023). CNN-based models have been employed to automatically identify pests in crop images, enabling early intervention and reducing crop damage (Li et al., 2021). For example, CNNs have been used to detect pests like aphids (Zhang et al., 2023), whiteflies (de Castro Pereira et al., 2022), and locusts (Tabar et al., 2021), helping farmers promptly prevent outbreaks. CNN models have also been applied to detect diseases in fruits and vegetables (Khattak et al., 2021). For instance, tomato disease detection (Shanthi et al., 2024) has been a primary focus of research. CNNs classify diseases such as bacterial spots (Mashamba et al., 2024), leaf mold (Singh et al., 2023), and yellow leaf curl viruses (Liu et al., 2023). In addition to disease and pest detection, CNNs have been used for weed detection and crop classification (Adhinata et al., 2024). Distinguishing between crops and weeds in a field is crucial for ensuring efficient herbicide application and optimizing crop yield. CNN models can identify and classify weeds among various crops, helping farmers manage their fields more effectively and reduce the use of chemicals. CNNs have also been applied to yield prediction tasks (Van Klompenburg et al., 2020). By analyzing images of crops at different growth stages, CNN-based models can predict the potential yield of a field. Beyond plant-based agriculture, CNN models have been used in livestock monitoring (Chelotti et al., 2024) to identify animal health issues and optimize feeding strategies.

In summary, CNN-based models have become a cornerstone of modern agricultural technology, facilitating a wide range of tasks from crop disease detection to pest management and yield prediction.

2.2. Methods in agricultural applications

Chen et al. use edge intelligence and deep learning with the tiny version of the popular single-stage object detection method named You Only Look Once (YOLO) on an embedded NVIDIA Jetson TX2 system to identify Tessaratoma papillose pests in fruit orchards in real-time, which reduces pesticide-spraying routes by 19%, decreasing workforce needs by 53% (Chen et al., 2021). Kasinathan et al. work on insect classification and detection in field crops using custom CNN models and pest detection algorithm provides the prospect of employing this algorithm to identify multiple insects and insect photos with varied stages of crop growth (Kasinathan et al., 2021). Researchers also introduced an automatic method for detecting injured leaf areas caused by insect predation for crops like blueberry, corn, potato, and soybean (da Silva Vieira et al., 2022). Ahmad et al. developed the YOLOv5x model for real-time insect pest detection, achieving an impressive average precision of 98.3% on a diverse dataset, with promising applications for detecting pests and reducing pesticide use in agriculture (Ahmad et al., 2022). The study by Sakkarvarthi et al. applies a CNN model for detecting and classifying tomato crop diseases. It outperforms established transfer learning techniques such as ResNet152, VGG19, and InceptionV3 (Sakkarvarthi et al., 2022). Another team of researchers leverages a meta-deep learning technique for the precise identification of diseases in cotton crops using a dataset of 2385 images (Memon et al., 2022). However, due to the high number of parameters, it might fall short of real-life applicability. Widén et al. study on using predator vocalizations to mitigate ungulate crop damage illustrates a novel approach in wildlife management, blending ecological understanding with technology to create a 'landscape of fear' for effective ungulate deterrence (Widén et al., 2022). Concurrently, Rani and Singh's work on using deep belief networks for early detection of crop infections showcases the potential of AI in enhancing agricultural productivity and environmental protection. Their method efficiently identifies diseases, pests, and nutrient deficiencies in paddy crops, thereby enabling targeted interventions and reducing excessive chemical use (Rani and Singh, 2022). Singh et al. explore using Internet of Things sensors and machine learning to reduce post-harvest losses in the perishable food supply chain. Using various image types, their study achieves a 93.33% classification accuracy with deep learning models, highlighting potential applications in quality control and waste reduction (Singh et al., 2022). Meena et al. study on Weed, Pest, and Disease detection uses a transfer learning approach for crop yield improvement (Meena et al., 2023). For training and testing, the accuracy of their feature fusion model of MobileNet, DenseNet201, and Hyperparameter Search 2D layer was 85.14% and 89.19%, respectively. T.B. Pun et al. use YOLO models, specifically YOLOv5 and YOLOv7, to detect and count root-knot nematodes in microscopic images. They introduce a novel application of mosaic augmentation for analyzing microscopic images captured with different objective lens magnifications (Pun et al., 2023). MacEachern et al. applied CNN models, especially YOLOv3, YOLOv4, and their derivatives, for accurate ripeness detection and ripeness estimation of wild blueberries. They have found promising results that could revolutionize fruit management practices (MacEachern et al., 2023). Ali et al. utilize Faster-PestNet, a deep-learning approach using Faster-RCNN with MobileNet as its base network. Faster-PestNet achieves 82.43% accuracy on the challenging IP102 dataset, effectively addressing sample distortions. The authors highlight the need for technologies like Faster-PestNet to enable early pest detection, mitigate crop damage, and reduce pesticide reliance in agriculture (Ali et al., 2023). In the study of Tirkey et al. an accuracy of approximately 95% to 98% is obtained from performance examination of AI-based systems for crop disease identification, detection, and classification utilizing InceptionV3, CNN, and YOLOv5 models (Tirkey et al., 2023b). Akshy et al. used 21-layered CNN and combined features from both far-field satellite and near-field images; the model achieves high efficiency and accuracy by Using VGG16 (Dhande and Malik, 2023). Shiman et al. applied transfer learning in tomato and rice leaf datasets with a convolutional recurrent neural network model to overcome the challenges of insufficient and imbalanced datasets, demonstrating high classification accuracy across various crop disease classes (Mondal et al., 2022). Researchers enhanced plant disease classification using CNN and 121,955 field images captured by mobile phones. Applying the RESNET-MC-1 architecture, which combines image data with crop metadata, they achieved 98% accuracy, significantly outperforming standard classification methods (Picon et al., 2019). Maria et al. proposed a hybrid model like InceptionResNet and ResNet152 in diagnosing tomato leaf diseases by combining VGG blocks with an inception module. The hybrid model retained stability during the training and validation phases and achieved faster convergence rates by integrating multi-scale feature extraction (Sanida et al., 2023). Other researchers investigate AI-based solutions for real-time detection of crop diseases and insect infestations in Soybean plants by using YOLOv5. And treatment of crop diseases, build an Android app, and achieve significant success in improving agricultural efficiency (Tirkey et al., 2023a). Uzair et al. introduce a CNN model with Inception v3, designed to automate and improve the identification of plant diseases through hyperspectral image analysis. And create a mobile application for real-time disease detection for agricultural practitioners (Bhatti et al., 2023). Another team worked on their own dataset named FieldPlant, based on corn, cassava, and tomato disease, and benchmarked the performance of SOTA CNN models, including MobileNet, VGG16, InceptionV3, and InceptionResNetV2. These models showed their ability to classify and detect plant diseases from the images (Moupojou et al., 2023). Moreover, Domingues et al. provide insights in their survey about the application of different machine learning and deep learning models for crop disease and pest detection, emphasizing challenges in data acquisition and highlighting the efficacy of transfer learning and few-shot learning in addressing data limitations (Domingues et al., 2022).

Additionally, researchers have also explored the use of UAV imagery for various applications in agriculture, including crop classification (Jintasuttisak et al., 2022; Tatini et al., 2022; Wibowo et al.,

Table 1
An overview of recent studies that utilize machine learning and deep learning techniques for various agricultural tasks, focusing on crop disease identification, pest management, and yield improvement. The table includes details such as datasets used, data accessibility, methodologies employed, and the primary objectives of each study.

Reference	Dataset (Size)	Public	Methods	Research goal
(Karim et al., 2024)	Grape leaf dataset (27,122)	✓	MobileNetV3Large	Real-time grape leaf disease classification using a lightweight CNN model for deployment on edge devices with explainable AI visualizations.
(Mashamba et al., 2024)	Tomato leaves dataset (8121)	×	VGG16	Detection and classification of tomato leaf diseases, specifically focusing on bacterial spot, early blight, late blight, yellow leaf curl, and Septoria leaf spot.
(Tirkey et al., 2023a)	Soybean crop leaves (3150)	×	YOLOv5, InceptionV3, CNN	Develop deep learning solutions for real-time soybean pest identification and classification to boost pest management efficiency.
(Ali et al., 2023)	IP102 (75,000) and local crops dataset (1950)	✓	Faster-PestNet	Develop a lightweight deep learning framework for efficient and accurate crop pest detection and classification.
(Meena et al., 2023)	Plant Disease (55,449), Weed (15,336), Pest (3150)	✓	DenseNet201, MobileNet, VGG16, InceptionV3	Develop a deep learning model to detect and classify crop threats, improving yield and management.
(Sanida et al., 2023)	PlantVillage (18,160)	✓	Hybrid CNN	Enhance accuracy in tomato leaf disease identification.
(Bhatti et al., 2023)	Images of various tree diseases (80,848)	×	CNN (InceptionV3)	Automate tree disease recognition using a mobile application for practical use in agriculture.
(Moupojou et al., 2023)	FieldPlanDataset (5170)	×	Deep learning	Enhance plant disease detection and classification in field conditions using a new dataset.
(Dhande and Malik, 2023)	Far-field crop datasets in India	×	CNNs, Transfer learning, Ensemble learning	Design an efficient crop damage detection model using an ensemble learning approach with deep CNNs.
(Singh et al., 2022)	Not Specified	×	U-Net, DeepLab, Mask R-CNN	To mitigate postharvest losses and improve the quality of fruits and vegetables using machine learning.
(Rani and Singh, 2022)	Nutrient deficiency (1156), Rice leaf diseases (3355)	✓	Deep Belief Network (DBN)	Develop AI to detect paddy crop issues early, reducing chemical use and environmental impact.
(Memon et al., 2022)	Cotton leaf (2385)	×	VGG16, ResNet50, Meta learner	The research develops a generalized model for identifying cotton leaf diseases.
(Mondal et al., 2022)	Tomato Leaf Dataset (7113), Rice Leaf Dataset (4422)	✓	CRNN, GRU, Transfer Learning	Enhance automatic detection and classification of crop leaf diseases and pest infestations using deep learning.
(Sakkarvarthi et al., 2022)	Tomato Leaf (3000)	✓	CNN	The goal of the research was to detect and classify diseases in tomato crops using a custom CNN model.
(Kasinathan et al., 2021)	Deng (282), IP102 (600)	✓	ANN, SVM, KNN, CNN, Naive Bayes	Classify and detect insects in various crops using machine learning techniques.
(Picon et al., 2019)	Barley, corn, rice, and rape seed (100,000)	×	CNNs	Enhance multi-crop disease classification by integrating crop-specific information into CNN models.

2022; Moazzam et al., 2021; Xiong et al., 2020), disease detection (Zhu et al., 2024; Zhao et al., 2023; Bouguettaya et al., 2021), process monitoring (Keita et al., 2022; Shih et al., 2021; Jia et al., 2021; Junagade et al., 2022; Piani et al., 2021), and yield estimation (Feng et al., 2020; Yang et al., 2019; Yu et al., 2016; Zhou et al., 2021; Zhang et al., 2020). While UAVs equipped with advanced sensory cameras have shown great promise in automating agricultural processes, their widespread adoption faces significant challenges (Istiak et al., 2023; Sousa et al., 2022). A major constraint is the lack of comprehensive flight regulations and legislation governing UAV operations in many countries, particularly Asia and Africa (Sousa et al., 2022). The absence of clear policies for the commercial, scientific, leisure, and military use of UAVs often impedes scientific research and the commercial deployment of UAV-based precision agriculture systems (Cracknell, 2017). This regulatory gap continues to be a barrier to leveraging UAV technology for large-scale agricultural monitoring.

Studying the recent literature works, we found many works that aim to develop models for crop monitoring, yield prediction, pest, and disease identification using different domain adaption techniques (Istiak et al., 2023; Ma et al., 2021; Li et al., 2023; Doha et al., 2021; Liu et al., 2024b), as detailed in Table 1. These tasks involve the detection, identification, and classification of various CNN and traditional machine learning-based models using ground-level and UAV imageries. While numerous crop type classification, pest, and disease detection datasets are publicly available, facilitating significant research and advancements in those areas, there is a notable gap regarding datasets specifically for crop damage classification. The lack of publicly available datasets for this task has limited the development and evaluation of CNN-based models for identifying damaged crops. This gap limits the progress in a critical area of precision agriculture, where early and accurate identification of damaged crops can lead to timely interventions, minimizing losses and improving overall productivity. Early identification allows farmers to address damaged areas more effectively, implementing necessary steps to mitigate the impact of pests, diseases, or environmental stressors. In this paper, we aim to address this gap by introducing a newly processed dataset for crop damage classification, and our proposed LightCDC model is designed to provide a scalable and efficient solution for real-time crop damage assessment. By filling this void, we hope to contribute to more effective crop management strategies, helping farmers reduce losses and increase productivity.

3. Methodology

Our study involves several steps, including data collection, cleaning, preprocessing, and benchmarking SOTA models, culminating in proposing a lightweight model for identifying damaged crops. Fig. 1 presents an overview of our workflow. The diagram outlines the process from data preprocessing, which involves classifying data into non-damaged and damaged categories, to splitting the data into training, validation, and test sets. Various classification models are applied, including SOTA,

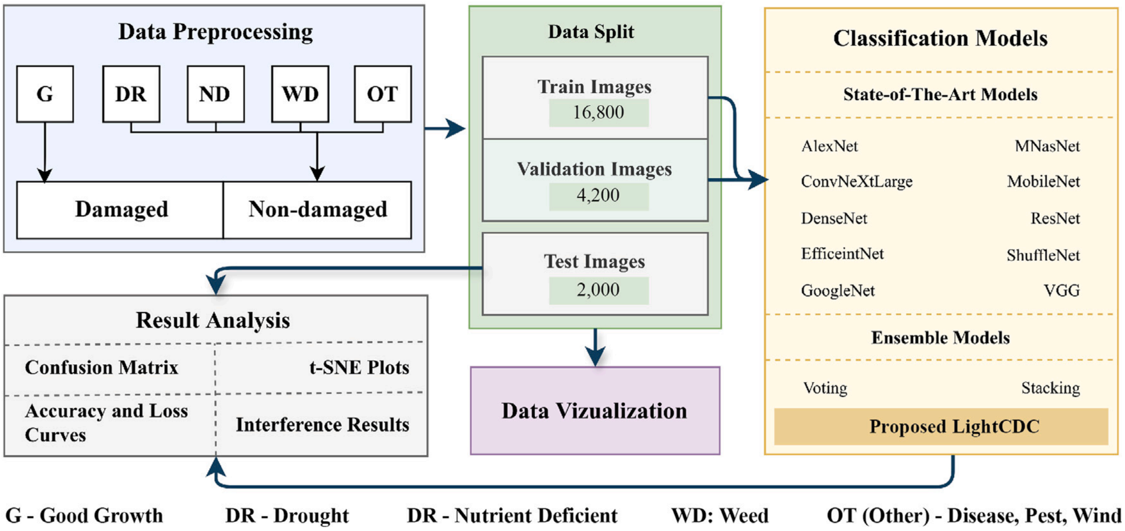


Fig. 1. An overview of the structured workflow, detailing the process from data preprocessing and splitting to model classification and result analysis, including the evaluation of SOTA models and the proposed LightCDC model.

Table 2
Comparison between the original CGIAR dataset and the processed dataset used for crop damage classification. The original dataset includes various classes, such as drought, nutrient deficiency, weeds, and other factors. For our binary classification task, we reclassified the images into two main categories: Damaged (comprising multiple damage types like drought, nutrient deficiency, and weeds) and Non-damaged (healthy crops).

Original CGIAR		Our processed CGIAR		
Classes	# of images	Classes	Damage type	# of images
Drought (DR) Good Growth (G) Nutrient Deficient (ND) Weed (WD)	4516 11,623 272 9238	Damaged	Drought (DR) Weed (WD) Nutrient Deficient (ND) Other (Disease, Pest, Wind)	11,000
Other (Disease, Pest, Wind)	419	Non-damaged	Good Growth (G)	12,000
Total:	**26,068**	**Total:**		**23,000**

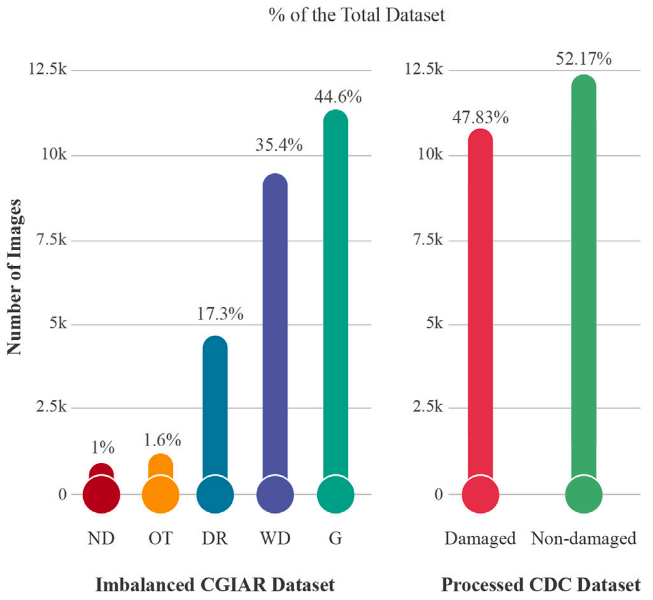


Fig. 2. Comparison of class distribution in between CGIAR and CDC datasets. This bar graph illustrates the distribution of classes within the original CGIAR dataset versus our processed CDC dataset, highlighting improvements in class balance between Damaged and Non-damaged classes.

ensemble, and our proposed LightCDC model. The results are then analyzed and visualized using various metrics.

3.1. Dataset and preprocessing

We use the CGIAR challenge dataset (CGIAR, 2024), which was mainly introduced to classify crops having different types of damage such as "*Drought*", "*Good Growth*", "*Nutrient Deficiency*", "*Weed*", and "*Other*" (disease, pest, wind), as also detailed in Table 2. Although the dataset contains around 26,000 images for training, the class distribution is not balanced. As shown in Fig. 2, the good class contains around 44.6% of the images in the total number of images in the dataset, while the "*Nutrient Deficiency*" class has only 1% of the total images. This shows how imbalanced the dataset is. Besides, after reviewing the dataset manually, we found many images that were wrongly classified, which made it a massive challenge for the classification models to be generalized. So, we pre-process the dataset by cleaning and reclassifying the images as follows:

- Reclassing for Binary Classification: To create a new dataset for training and evaluating a binary classification model, such as crop damage classification, we first formed Damaged and Non-Damaged classes. In the Damaged class, we included images of various types of crop damage such as "*Drought,*" "*Nutrient Deficiency,*" "*Weed,*" and "*Other,*" as presented in Table 2.
- Data Cleaning: We discovered that many images were incorrectly labeled after creating the two new classes. For example, some images were categorized as damaged despite showing no damage. We manually corrected the labels for those images. Additionally, we identified and removed irrelevant images, such as those containing only soil or fields without crops, to ensure the dataset's quality and improve the model's ability to generalize for crop

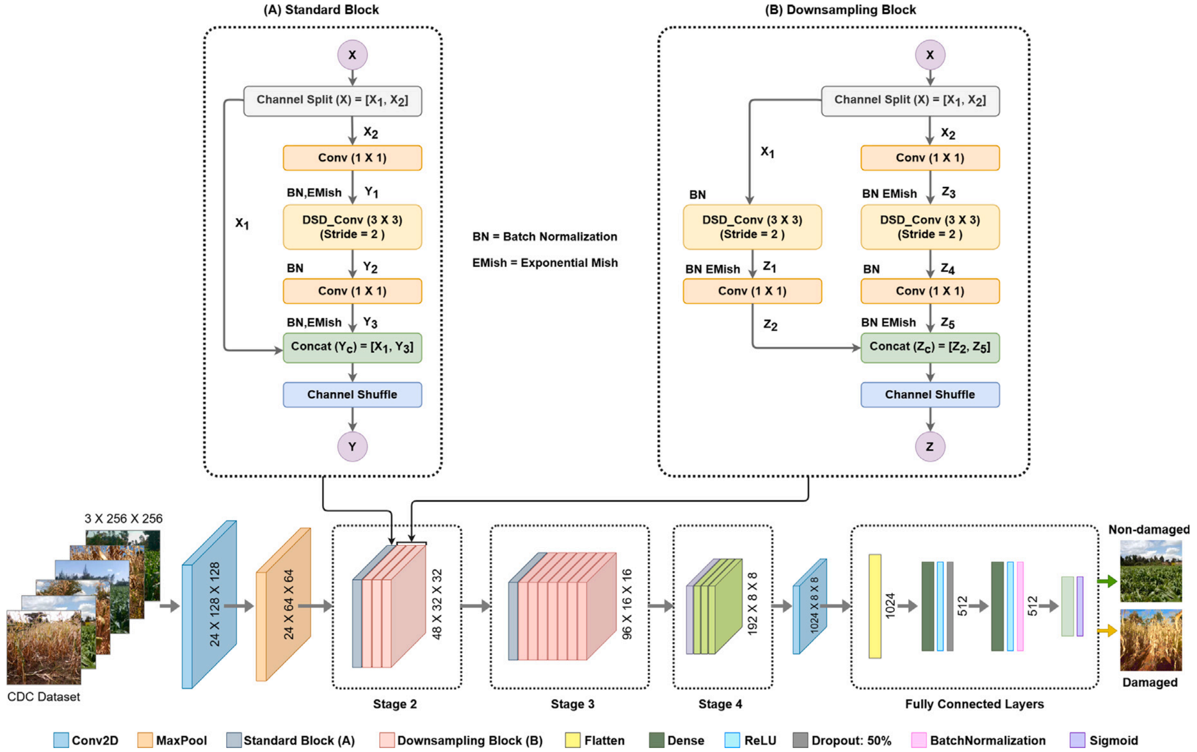


Fig. 3. A detailed illustration of our proposed LightCDC model. It uses the standard and downsampling blocks from the original ShuffleNetV2 model and adds a few custom dense layers, contributing to the model's capacity to predict damaged crops accurately.

damage classification. Thus, after data cleaning, our dataset contains 23,000 images, where 11,000 are damaged and 12,000 are non-damaged.

- Dataset Splitting: Finally, we split the dataset into two different subsets. The train set contains 21,000 images, and the test set contains 2000 images.

Table 2 illustrates which class of the original CGIAR dataset we used as damaged and unmanaged. The class distribution after processing the dataset is displayed in Fig. 2. Thus, leveraging the original CGIAR (CGIAR, 2024) dataset, we introduce a newly processed dataset utilizing the existing dataset that can be used to train, validate, and test SOTA classifiers and any new models for identifying damaged crops. Furthermore, inspired by the data-producing techniques in (Huang et al., 2020; Islam et al., 2024b,a), we generated adverse data to further validate the robustness of the proposed model under different real-life adverse scenarios such as blur, low light, and fog.

3.2. Transfer learning

We use transfer learning to extract features from SOTA classification models. Feature extraction simplifies complex data by highlighting important patterns, enabling more efficient analysis and improving the performance of subsequent learning algorithms, especially with high-dimensional data like images or text. We freeze all layers except the classifier layers of the ImageNet-trained (Deng et al., 2009) SOTA classification models (Krizhevsky et al., 2012; Liu et al., 2022; Huang et al., 2017; Tan and Le, 2019; Szegedy et al., 2015; Tan et al., 2019; Sandler et al., 2018; Howard et al., 2019; He et al., 2016; Ma et al., 2018; Iandola et al., 2016; Karen, 2014), ensuring that only the classifier layer is trainable on our new dataset. This strategy allows us to retain the learned representations from the pre-training phase while adapting the model specifically to our target task. By fine-tuning solely the classifier layer, we optimize the model's performance for our processed CDC dataset without altering the deeper representations, thus effectively leveraging the power of transfer learning.

3.3. Proposed model: LightCDC

We propose the LightCDC, leveraging the ShuffleNetV2 (Ma et al., 2018) architecture as it is suitable for efficient, lightweight, and accurate crop damage classification compared to the other models. We selected ShuffleNetV2 as the baseline for our LightCDC model due to its unique design that optimizes both efficiency and accuracy, particularly in scenarios requiring low computational overhead. ShuffleNetV2 achieves this through its innovative use of channel splitting, depthwise separable convolutions, and channel shuffling, which reduce computational complexity while maintaining the network's ability to extract valuable features from images.

By splitting the channels within the network, ShuffleNetV2 reduces the number of computations required for each convolutional layer, allowing the model to process high-dimensional data like images more efficiently. The depthwise separable convolutions further reduce the number of trainable parameters by decomposing the standard convolution operation into a depthwise convolution and a pointwise convolution, which minimizes the computational cost without sacrificing the richness of feature extraction.The channel shuffling operation ensures that information is exchanged between different feature channels, preventing the model from losing important spatial and contextual relationships between pixels. This combination of operations allows ShuffleNetV2 to maintain a lightweight architecture while effectively capturing complex patterns in the data, making it highly suited for real-time applications where computational resources are limited. By leveraging these technical advantages, ShuffleNetV2 forms a strong foundation for our LightCDC model, enabling it to achieve high accuracy with a low number of parameters.

Fig. 3 depicts the detailed architecture of our LightCDC model, which consists of two primary ShuffleNetV2 blocks: (A) standard block and (B) downsampling block. Each block performs critical operations such as channel splitting, convolutions, depthwise separable convolutions (DSD Conv), and channel shuffling to maintain computational efficiency and effective feature extraction.

The input image, sized 256 × 256, passes through a series of convolutional and max-pooling layers structured into four stages, each applying blocks (A) and (B) with varying strides and kernel sizes as shown in Fig. 3. The output features are then flattened and passed through fully connected (FC) layers, where we modify the final layers to suit our specific task of crop damage classification. The model's final layers include dense layers with ReLU activations, batch normalization, and a sigmoid activation function to output probabilities for damaged and non-damaged crops as illustrated with different color blocks in Fig. 3. This architecture enhances processing efficiency and achieves high accuracy in identifying crop damage, demonstrating the effectiveness of our LightCDC model for smart farming applications.

3.3.1. Architecture breakdown of LightCDC

In this section, we will break down the building blocks of our proposed model from first to last to see how the model processes an input image and extra valuable features that contribute to accurate crop damage classification. First of all, the channel-splitting process can be represented as follows:

$$X = [X_1, X_2], \quad \text{where} \quad X_1, X_2 \in \mathbb{R}^{H \times W \times \frac{C}{2}} \tag{1}$$

Here in Eq. (1), the input tensor X is divided into two equal parts, X_1 and X_2, along the channel dimensions such as height (H), width (W) of the feature maps and the number of channels (C). This operation allows for parallel processing in the subsequent layers. Then, One branch (X_2) undergoes a 1 × 1 convolution, which reduces the number of input channels. This step decreases computational complexity and introduces non-linearity to the model can be presented by Eq. (2).

$$Y_1 = \text{EMish}(\text{BN}(\text{Conv1x1}(X_2))) \tag{2}$$

where, "*EMish*" refers to the Exponential Mish activation function, a variant of the Mish (Misra, 2019) function that enhances non-linearity and smoothness for better gradient flow during training, while "*BN*" denotes Batch Normalization, which helps stabilize and accelerate training by normalizing the output of the preceding convolutional layer.

$$Y_2 = \text{BN}(\text{DSD\_Conv}_{3\times3,\text{stride}=2}(Y_1)) \tag{3}$$

In the equations Eq. (3), Y_2 represents the output of a 3 × 3 depthwise separable convolution applied to the tensor Y_1 from the Eq. (2), followed by batch normalization. This convolution operation with stride 2 helps reduce the spatial dimensions while maintaining the features extracted from Y_1.

Algorithm 1 Pseudocode for reproducing our proposed LightCDC.

```
Require: CDC Dataset ($D$), ShuffleNetV2 ($\mathcal{M}$), Split $D$ into Train ($D_{train}$): 80% and Validation ($D_{val}$): 20%
Parameters: Epochs = 100, Batch size = 64, Image size = 256, Learning rate = 0.001, Early stop steps = 15, Classes ($C$) = 2

Procedure: Initialize LightCDC Model
Freeze all parameters in $\mathcal{M}$ except the final classifier
$f_{\text{in}} \leftarrow$ number of input features to the final FC layer of $\mathcal{M}$
Define new FC layer $\mathcal{F}$ as follows:
    $\mathcal{F} \leftarrow \text{Sequential}(\text{Linear}(f_{\text{in}}, 512), \text{ReLU}(),$
        $\text{Dropout}(0.5), \text{Linear}(512, 512), \text{ReLU}(),$
        $\text{BatchNorm}(512), \text{Linear}(512, C))$
Replace the classifier in $\mathcal{M}$ with $\mathcal{F}$
End Procedure

Procedure: Train LightCDC($D_{train}$, $D_{val}$)
Initialize LightCDC ($\mathcal{M}$)
for $epoch = 1$ to 100 do
    Train model on $D_{train}$
    Evaluate model on $D_{val}$
    Save model if performance improves
    if early stopping criterion is met then
        break
    end if
end for
End Procedure
```

$$Y_3 = \text{EMish}(\text{BN}(\text{Conv}_{1\times1}(Y_2))) \tag{4}$$

$$Y_c = \text{Concat}(X_1, Y_3) \tag{5}$$

Next, Y_3 in Eq. (4) is generated by passing Y_2 through a 1 × 1 convolution, followed by batch normalization and "*EMish*" activation, which introduces non-linearity and refines the feature representation. Finally, the concatenated output Y_c is formed by combining X_1 and Y_3 along the channel dimension, as shown in Eq. (5). This concatenation allows information from both branches, X_1 and Y_3, to be merged, resulting in a richer feature representation for the subsequent layers in the model.

$$Y = \text{ChannelShuffle}(Y_c) \tag{6}$$

Then Eq. (6) introduces the channel shuffle operation, which rearranges the channels to ensure the effective mixing of features from different groups, reducing the overall computational cost. In this context, the concatenated output Y_c from the previous step is processed by a channel shuffle operation to produce the final output Y. This operation rearranges the channels within Y_c, which allows better inter-channel information exchange and ensures that features from both branches (i.e., X_1 and Y_3) are effectively combined.

For the downsampling block from Fig. 3(B), similar operations are applied after channel splitting, but with a stride of 2 in the depthwise convolution to reduce the spatial dimensions by half:

$$Z_1 = \text{EMish}(\text{BN}(\text{DSD\_Conv}_{3\times3,\text{stride}=2}(X_1))) \tag{7}$$

$$Z_2 = \text{BN}(\text{Conv}_{1\times1}(Z_1)) \tag{8}$$

Eqs. (7) and (8) describe the transformations applied to the first branch of the splitter input tensor X_1 in the downsampling block. In Eq. (7), X_1 is first processed by a 3 × 3 depthwise separable convolution with a stride of 2, which helps reduce the spatial dimensions. This is followed by batch normalization to stabilize and normalize the output and then an "*EMish*" activation to introduce non-linearity, resulting in the transformed output Z_1. In Eq. (8), Z_1 is further processed by a 1 × 1 convolution, followed by batch normalization to produce Z_2. This 1 × 1 convolution reduces the number of channels, helping to consolidate information while maintaining the spatial resolution.

$$Z_3 = \text{EMish}(\text{BN}(\text{Conv}_{1\times1}(X_2))) \tag{9}$$

$$Z_4 = \text{BN}(\text{DSD\_Conv}_{3\times3,\text{stride}=2}(Z_3)) \tag{10}$$

$$Z_5 = \text{EMish}(\text{BN}(\text{Conv}_{1\times1}(Z_4))) \tag{11}$$

Similarly, in the second branch of the downsampling block in the Fig. 3(B) the splitter input tensor X_2 goes through different processing steps as presented in the Eqs. (9)–(11). Finally, we have the processed X_1 and X_2 from the both branches as two separate output Z_2 and Z_5 which is concatenated as shown in Eq. (12) as follows:

$$Z_c = \text{Concat}(Z_2, Z_5) \tag{12}$$

$$Z = \text{ChannelShuffle}(Z_c) \tag{13}$$

In the final step in the downsampling block in Fig. 3(B), where the concatenated output Z_c undergoes a channel shuffle operation to produce the final output Z. This channel shuffle operation rearranges the channels within Z_c as we also do in Eq. (6), promoting inter-channel information exchange and enhancing the representational power of the network. This process enhances the model's ability to learn diverse and rich feature representations while maintaining high computational efficiency. Then, Stages 2, 3, and 4 in this architecture as presented in the Fig. 3 progressively reduce the spatial dimensions while increasing

Table 3
Detailed architecture of our proposed LightCDC model, providing a comprehensive overview of the LightCDC architecture, outlining each layer's input and output shapes, kernel sizes, stride, and repeat count.

Layers	Input shape [Channel, Height, Width]	Kernel size	Stride	Repeat	Output shape [Channel, Height, Width]
Input image	[3, 256, 256]	–	–	–	[2]
Convolution 1 (Conv1)	[3, 256, 256]	3 × 3	2	1	[24, 128, 128]
MaxPool	[24, 128, 128]	3 × 3	2	1	[24, 64, 64]
Stage 2	[24, 64, 64]	-	2 1	1 3	[48, 32, 32]
Stage 3	[48, 32, 32]	-	2 1	1 7	[96, 16, 16]
Stage 4	[96, 16, 16]	-	2 1	1 3	[192, 8, 8]
Convolution 5 (Conv5)	[192, 8, 8]	1 × 1	1	1	[1024, 8, 8]
Fully Connected (FC)	[1024]	–	–	–	[512]
	[512]	–	–	–	[512]
	[512]	–	–	–	[2]
Total Parameters				1,131,298	
Trainable params				789,506	
Non-trainable params				341,792	

the channel depth to extract increasingly abstract features. This hierarchical reduction in spatial dimensions and increased channels allow the model to focus on high-level features needed for accurate classification.

The output from Stage 4 [192 × 8 × 8] is passed through a convolutional layer, transforming it to [1024 × 8 × 8]. This reshaped feature map, f_{in}, is then flattened into a 1024-element vector for the fully connected layers as also presented in Algorithm 1. The fully connected layer $\mathcal{F}$ starts with a linear layer mapping f_{in} to 512 dimensions with ReLU activation, followed by 0.5 dropout for regularization. Another linear layer refines it to 512 dimensions, followed by batch normalization and a final layer that outputs C classes (damaged and non-damaged crops) through the prediction probabilities achieved with a sigmoid activation function as follows:

$$\hat{y} = \sigma(W \cdot x + b) \quad \text{where} \quad \sigma(z) = \frac{1}{1 + e^{-z}} \tag{14}$$

In Eq. (14), W represents the weight matrix, x is the input vector, and b is the bias term. The sigmoid function denoted as σ, maps the linear combination of the input features to a probability value between 0 and 1, which allows the model to distinguish between the two classes effectively. The output $\hat{y}$ represents the predicted probability of the input belonging to the "*damaged*" class.

Table 3 details the architecture of LightCDC, outlining each layer's input and output shapes, kernel sizes, stride, and repeat count. Starting with an input image of size 256 × 256 with 3 channels, the image is passed through an initial convolutional layer (Conv1) with a kernel size of 3 × 3 and stride 2, producing an output of [24, 128, 128]. This is followed by a max-pooling layer with a kernel size of 3 × 3 and stride 2, reducing the output to [24, 64, 64], where 24 is the number of channels and 64 by 64 is the dimension of the features. In Stage 2, the model employs a ShuffleNetV2 block with an input of [24, 64, 64], repeating three times and producing an output of [48, 32, 32]. Here, the input and output shapes are presented in order of the number of channels and the feature vector's dimension (height and width). Stage 3 further processes the data with ShuffleNetV2 blocks (A) and (B), where the standard block (A) was used only once and downsample block (B) was repeated seven times, resulting in an output of [96, 16, 16]. Stage 4 uses a similar block, repeated thrice, with an output of [192, 8, 8]. The final convolutional layer (Conv5) applies a 1 × 1 convolution, maintaining the input dimensions at [192, 8, 8] and the output dimension at [1024, 8, 8]. The features are then flattened and passed through two FC layers of 512 with ReLu activation functions with a dropout of 0.5 in the first layer and a batch normalization layer in the second layer, respectively, before reaching the final output layer, which classifies the input into two categories: damaged or non-damaged using the Sigmoid function presented in Eq. (14). The model comprises a total of 1,131,298 parameters, with 789,506 trainable parameters and 341,792 non-trainable parameters, highlighting its efficiency and suitability for crop damage classification tasks.

Additionally, the Algorithm 1 presents the pseudocode for customizing and training the LightCDC model based on the ShuffleNetV2 architecture. Initially, the CDC dataset is split into training (80%) and validation (20%) sets. The model parameters are defined, including 100 epochs, a batch size of 64, an image size of 256 × 256 pixels, a learning rate of 0.001, and early stopping with a patience of 15 steps.

3.4. Evaluation metrics

We employed widely recognized performance metrics, including accuracy, precision, recall, F1-score, and the confusion matrix, to evaluate the models. Additionally, we used t-SNE (Van der Maaten and Hinton, 2008), a powerful technique for visualizing high-dimensional data in a lower-dimensional space. This method helps to uncover hidden patterns and clusters within complex datasets, by preserving the underlying structure of the data as much as possible in the reduced dimensions.

$$q_{mn} = \frac{\exp(-\|z_m - z_n\|^2)}{\sum_{a \neq b} \exp(-\|z_a - z_b\|^2)} \tag{15}$$

In this equation, z_m and z_n are the low-dimensional representations of the high-dimensional data points x_m and x_n, respectively. The term a refers to the number of nearest neighbors in the high-dimensional space, and q_{mn} denotes the joint probability that points m and n are close neighbors in the lower-dimensional space.

3.5. Explainable AI and GradCAM

Deep learning models are often perceived as *black boxes* because they lack transparency. Explainable AI (XAI) aims to address this by making these models more interpretable. We leverage Gradient Class Activation Mapping (GradCAM) (Selvaraju et al., 2017) to provide insights into the model's predictions. GradCAM identifies the key regions in an image that influence the model's decision by analyzing the gradients in the final convolutional layer, and it produces a heatmap:

$$L^{d}_{\text{GradCAM}} = \text{ReLU}\left(\sum_{j} \beta^{d}_{j} B^{j}\right) \tag{16}$$

In this equation, β^{d}_{j} represents the weights derived from the global average pooling of the gradients of the feature map B^{j} for class d. This method provides a visual explanation of the model's behavior, enhancing the interpretability of its decision-making process.

Table 4
Performance comparison of SOTA Models on the testset of our processed CDC Dataset. Models are evaluated based on various performance metrics and computational demands. The top 3 models are marked by **bold**.

Models	Accuracy	Precision	Recall	F1-score	# of params (M)	Model size (MB)	GFLOPS
AlexNet	85.85	85.89	85.85	85.85	233.10	0.71	0.77
ConvNeXtLarge	**89.15**	**89.17**	**89.15**	**89.15**	197.80	754.50	34.36
DenseNet121	86.30	86.47	86.30	86.28	7.98	30.80	2.83
DenseNet161	86.70	86.76	86.70	86.69	28.70	110.40	7.73
DenseNet169	87.25	87.33	87.25	87.24	14.15	54.70	3.36
DenseNet201	86.95	86.96	86.95	86.95	20.01	77.40	4.29
EfficientNet_B0	84.85	84.93	84.85	84.84	5.30	20.50	0.39
EfficientNetV2Large	85.00	85.01	85.00	85.00	118.50	454.60	56.08
GoogleNet	80.90	81.04	80.90	80.88	6.62	49.70	1.50
MNasNet	84.15	84.15	84.15	84.15	2.20	8.60	0.10
MobileNetV2	85.90	85.91	85.90	85.90	3.50	13.60	0.30
MobileNetV3	84.35	84.36	84.35	84.35	5.50	21.10	0.22
ResNet50	87.75	87.77	87.75	87.75	25.60	97.80	4.09
ResNet101	**88.20**	**88.20**	**88.20**	**88.20**	44.50	170.50	7.80
ResNet152	86.80	86.90	86.80	86.79	60.19	203.50	11.51
ShuffleNetV2	**88.30**	**88.30**	**88.30**	**88.30**	1.40	5.30	0.04
SqueezeNet1_1	87.20	87.24	87.20	87.20	1.20	4.70	0.35
VGG16	81.55	81.56	81.55	81.55	138.40	527.80	15.47
VGG19	82.30	82.30	82.30	82.30	138.40	527.80	15.47

Table 5
Performance evaluation of ensemble techniques, such as stacking and voting, utilizing different combinations of SOTA models to classify damaged and non-damaged crops.

Techniques	Used SOTA models					Accuracy	Precision	Recall	F1-score
	ConvNeXtLarge	ShuffleNetV2	ResNet101	ResNet50	DenseNet169				
Stacking1	✓	✓	✓	×	×	89.05	89.05	89.05	89.05
Stacking2	✓	✓	×	×	×	89.20	89.20	89.20	89.20
Stacking3	✓	×	✓	×	×	88.75	88.75	88.75	88.75
Stacking4	✓	✓	✓	✓	×	**89.65**	**89.66**	**89.65**	**89.65**
Stacking5	✓	✓	✓	✓	✓	89.15	89.18	89.15	89.15
Voting1	✓	✓	✓	×	×	89.35	89.36	89.35	89.35
Voting2	✓	✓	×	×	×	88.40	88.64	88.40	88.38
Voting3	✓	×	✓	×	×	46.00	45.12	46.00	43.46
Voting4	✓	✓	✓	✓	×	89.20	89.30	89.20	89.19
Voting5	✓	✓	✓	✓	✓	89.40	89.40	89.40	89.40

3.6. *Experimental setups*

Following the workflow outlined in Fig. 1, we begin our experiments by training SOTA classification models on the CDC dataset for crop damage classification. We employ transfer learning techniques to train these pre-trained models. After training, we evaluate their performance on a test set from the CDC dataset. Subsequently, we conduct further experiments involving ensemble learning techniques, utilizing the top-performing SOTA models (Krizhevsky et al., 2012; Liu et al., 2022; Huang et al., 2017; Tan and Le, 2019; Szegedy et al., 2015; Tan et al., 2019; Sandler et al., 2018; Howard et al., 2019; He et al., 2016; Ma et al., 2018; Iandola et al., 2016; Karen, 2014) from the initial evaluations. We train the SOTA models for these experiments with a batch size of 32, a learning rate of 0.001, and an image size of 256 × 256 over 50 epochs.

In addition, we conduct an extensive ablation study on our proposed LightCDC model to determine the optimal set of parameters for achieving the highest accuracy. We train the model with varying batch sizes, ranging from 8 to 128, learning rates, ranging from 0.1 to 0.0001, and different optimizers. We also incorporate techniques like early stopping and a learning scheduler during the ablation studies of LightCDC.

4. Results and discussion

Our research workflow involves experimenting with SOTA classification models and ensemble techniques such as stacking and voting. Besides we perform various ablation studies on our proposed LightCDC model. In addition, we do quantitative analysis based on the metrics mentioned in Section 3.4. The results of each experiment are discussed in detail in the following sections.

4.1. *Performance of SOTA models*

The experimental results presented in Table 4 present a performance comparison of various SOTA classification models on the CDC dataset, evaluating their accuracy, precision, recall, F1-score, and computational demands such as the number of parameters, model size, and GFLOPS. Among the models, ConvNeXtLarge, ResNet101, and ShuffleNetV2 demonstrated the highest performance, with ConvNeXtLarge achieving the best overall accuracy, precision, recall, and F1-score at 89.15%. ResNet101 and ShuffleNetV2 followed closely with accuracies of 88.20% and 88.30%, respectively. Despite their high performance, ConvNeXtLarge has a significantly larger model size (754.5 MB) and computational complexity (34.36 GFLOPS) compared to the more efficient ShuffleNetV2, which has a model size of 5.3 MB and only 0.04 GFLOPS. This highlights ShuffleNetV2's suitability for applications requiring a balance between high accuracy and low computational resource usage. Other models like DenseNet169 and DenseNet201 also performed well, with accuracies above 87% but varying model sizes and complexities. Overall, this table underscores the trade-offs between model performance and computational efficiency, guiding the selection of appropriate models for crop damage classification tasks.

Delving deeper into the analysis of model complexity, it becomes evident that the number of parameters and model size vary significantly across the evaluated models. For instance, ConvNeXtLarge, with 197.8 million parameters and a model size of 754.5 megabytes, showcases

Table 6
Comparative analysis of LightCDC with top-performing SOTA models based on various metrics, highlighting the efficiency and effectiveness of the LightCDC model in classifying damaged crops. **MIT (ms)**: Mean Inference Time in milliseconds.

Models	Accuracy	Precision	Recall	F1-score	Params (M)	Size (MB)	GFLOPS	MIT (ms)↓
ConvNeXtLarge	89.15	89.17	89.15	89.15	197.80	754.50	34.36	3772.60
ResNet101	88.20	88.20	88.20	88.20	44.50	170.50	7.80	450.80
ShuffleNetV2	88.30	88.30	88.30	88.30	1.40	5.30	0.04	15.04
SqueezeNet1_1	87.20	87.24	87.20	87.20	1.20	4.70	0.35	14.03
LightCDC	**89.44**	**89.44**	**89.44**	**89.44**	**1.13**	**4.53**	**0.06**	**13.37**

Table 7
Ablation study on Batch Size for the LightCDC model while training the models using and not using transfer learning approaches. This table evaluates the performance of the LightCDC model at different batch sizes, maintaining a constant learning rate of 0.001 and using the Adam optimizer employing the early stopping technique of step 10.

Models	Varying	With transfer learning				Without transfer learning			
	Batch size	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
LightCDC	8	88.40	88.40	88.40	88.40	87.80	88.01	87.80	87.78
	16	88.40	88.40	88.40	88.40	88.00	88.05	88.00	88.00
	32	88.50	88.51	88.50	88.50	88.50	88.79	88.50	88.48
	64	**88.65**	**88.66**	**88.65**	**88.65**	**89.44**	**89.44**	**89.44**	**89.44**
	128	88.30	88.30	88.30	88.30	88.30	88.30	88.30	88.30

a more intricate architecture, enabling it to capture intricate features in the data. Conversely, models like AlexNet and SqueezeNet1_1 exhibit relatively fewer parameters and smaller model sizes, making them computationally efficient options for scenarios where resource constraints are a concern.

Interestingly, ShuffleNetV2 is a compelling choice due to its lightweight design, achieving an accuracy of 88.3% with a mere 1.4 million parameters and a compact model size of 5.3 megabytes. This underscores the importance of considering not only performance metrics but also space efficiency when selecting models for deployment in real-world applications. Moreover, the inclusion of metrics such as GFLOPS (floating-point operations per second) further enhances our understanding of the computational requirements associated with each model, facilitating informed decision-making in the selection process.

While the models evaluated in this study exhibit promising performance, there remains room for further exploration and optimization. Future research endeavors should focus on refining existing architectures, exploring novel model architectures tailored to the specific characteristics of crop damage data, and leveraging techniques such as transfer learning to enhance model generalization and adaptability.

4.2. Performance of ensemble techniques

In Table 5, we present the results of our supplementary experiments, where we explore ensemble techniques such as stacking and majority voting. These techniques aim to harness multiple models' collective intelligence to enhance classification performance potentially. The table showcases the accuracy achieved by each ensemble model and provides insight into the constituent models utilized within each ensemble.

Upon scrutinizing the results, Stacking4 emerges as the standout performer, achieving an impressive accuracy of 89.65%. Remarkably, this accuracy surpasses that of ConvNeXtLarge, a leading single-model performer identified in our preliminary experiments with SOTA models. However, it is important to note that the success of Stacking4 comes at the cost of increased model complexity. As indicated in Table 5, Stacking4 amalgamates four distinct models, resulting in a heavier computational burden.

In contrast, our investigation into majority voting reveals that Voting5 achieves the highest accuracy among the voting models, reaching 89.40%. By aggregating predictions from five individual models, Voting5 demonstrates the potential for ensemble techniques to yield incremental improvements in classification accuracy. Nonetheless, it is crucial to acknowledge the trade-offs associated with employing multiple models within an ensemble. While ensemble methods can offer performance gains, they often entail increased computational overhead, leading to longer inference times.

Overall, our experimentation with ensemble techniques illuminates their efficacy in enhancing classification accuracy, albeit with associated complexities and computational costs. These findings underscore the importance of carefully weighing the benefits and drawbacks of ensemble methods in the context of specific application requirements and resource constraints.

4.3. Performance of LightCDC

A comprehensive performance comparison of our proposed LightCDC model with top-performing SOTA models, including ConvNeXtLarge, ResNet101, SqueezeNet1_1, and ShuffleNetV2, is presented in Table 6. LightCDC achieves the highest accuracy, precision, recall, and F1-score of 89.44%, outperforming all other SOTA models tested against the CDC test dataset for classifying damaged and non-damaged crops. This indicates that LightCDC delivers superior classification performance and excels in computational efficiency. While ConvNeXtLarge achieves an accuracy of 89.15%, making it the second highest in terms of accuracy, it requires significantly more computational resources, with 197.80 million parameters, a model size of 754.50 MB, and 34.36 GFLOPS, resulting in a mean inference time (MIT) of 3772.60 ms.

In contrast, LightCDC is highly efficient with only 1.13 million parameters, a model size of 4.53 MB, and requiring just 0.06 GFLOPS. This efficiency translates to a much lower MIT of 13.37 ms, making LightCDC far more suitable for real-time applications and resource-constrained environments. Additionally, LightCDC outperforms ResNet101, ShuffleNetV2, and SqueezeNet1_1 in terms of computational efficiency while maintaining competitive accuracy and precision. For instance, ResNet101 achieves an accuracy of 88.20% but requires 44.50 million parameters and has an MIT of 450.80 ms. Similarly, ShuffleNetV2 and SqueezeNet1_1, while efficient, do not match LightCDC's balance of accuracy and computational cost.

This comparison highlights LightCDC's advantage in balancing performance and computational demands. While ConvNeXtLarge achieves slightly lower accuracy, the substantial computational resources it requires make LightCDC a more practical and efficient choice for crop damage classification. This efficiency is critical for deploying machine learning models in real-world agricultural scenarios where computational resources are limited, and real-time processing is essential. Therefore, LightCDC offers a compelling solution with its excellent trade-off between high accuracy and low computational cost.

Table 8
Performance comparison of LightCDC model across different optimizers. This table showcases the impact of various optimizers on the LightCDC model's performance, maintaining a constant batch size of 64, learning rate of 0.001, and employing early stopping with a step setup of 10. It presents metrics including accuracy, precision, recall, and F1-score for each optimizer tested.

Models	Parameters		With transfer learning				Without transfer learning			
	Batch size	Optimizer	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
LightCDC	**64**	**Adam**	**88.65**	**88.66**	**88.65**	**88.65**	**89.44**	**89.44**	**89.44**	**89.44**
	64	Adagrad	43.45	42.74	43.45	42.03	41.25	40.61	41.25	40.23
	64	AdamW	53.35	53.39	53.35	53.23	51.75	51.75	51.75	51.15
	64	RMSprop	52.75	52.75	52.75	52.73	52.05	52.18	52.05	51.34
	64	Adadelta	48.55	48.55	48.55	48.54	49.20	49.12	49.20	48.06
	64	Adamax	44.20	43.10	44.20	41.88	52.05	52.18	52.05	51.34
	64	ASGD	54.80	54.80	54.8	54.80	49.20	49.12	49.20	48.06
	64	Rprop	44.95	44.75	44.95	44.42	52.05	52.18	52.05	51.34
	64	SparseAdam	57.40	58.79	57.40	55.64	49.20	49.12	49.20	48.06

Table 9
Influence of different **learning rates (LR)** on the accuracy, precision, recall, and F1-score of the proposed LightCDC model, using a consistent **batch size (BS)** of 64, the Adam optimizer, and employing early stopping with a step setup of 10. The table compares results with and without transfer learning.

Models	Parameters			With transfer learning				Without transfer learning			
	BS	Optimizer	LR	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
LightCDC	64	Adam	0.1	87.80	87.80	87.80	87.80	86.80	86.83	86.80	86.80
	64	Adam	0.01	88.15	88.16	88.15	88.15	88.95	89.07	88.95	88.94
	64	Adam	0.001	**88.65**	**88.65**	**88.65**	**88.65**	**89.44**	**89.44**	**89.44**	**89.44**
	64	Adam	0.0001	87.95	87.95	87.95	87.95	89.30	89.31	89.30	89.30

4.4. Ablation study

This section presents a comprehensive analysis of the LightCDC model's performance under various training conditions through a series of ablation studies. These studies investigate the effects of different batch sizes, optimizers, learning rates, and combinations of transfer learning and data augmentation on the model's accuracy, precision, recall, and F1-score. By systematically varying these parameters and methodologies, we aim to identify the optimal configurations that enhance the efficiency and effectiveness of the LightCDC model for classifying damaged and non-damaged crops. Tables 7 to 10 provide detailed insights into these experiments, highlighting the model's performance across different scenarios.

4.4.1. Impact of batch size

Table 7 presents an ablation study on the impact of varying batch sizes on the performance of the LightCDC model, both with and without transfer learning. The model was trained with batch sizes of 8, 16, 32, 64, and 128, maintaining a constant learning rate of 0.001 and using the Adam optimizer with an early stopping technique of step 10.

With transfer learning, the highest accuracy, precision, recall, and F1-score were achieved with a batch size of 64, all at 88.65%. Smaller batch sizes, such as 8 and 16, resulted in lower performance, with all metrics at 88.40%. Increasing the batch size to 128 caused a slight drop in performance to 88.30% across all metrics.

Without transfer learning, the highest performance was observed with a batch size of 64, achieving 89.44% in accuracy, precision, recall, and F1-score. Smaller batch sizes, such as 8 and 16, showed lower performance, with accuracy at 87.80% and 88.00%, respectively. Larger batch sizes, such as 128, resulted in a minor decrease in performance, with an accuracy of 88.30%. This suggests that a batch size of 64 is optimal for training the LightCDC model, providing the best balance between performance and computational efficiency.

Some of the key observations from the table are as follows:

- For both training scenarios (with and without transfer learning), batch size 64 consistently provided the best performance, indicating it is the optimal batch size for the LightCDC model.
- Smaller batch sizes (e.g., 8 and 16) tend to underperform compared to larger batch sizes, likely due to less stable gradient estimates.
- Extremely large batch sizes such as 128 also show a decrease in performance.

4.4.2. Impact of optimizers

As with Table 7, which uses the same set of parameters, Table 8 compares the performance of the LightCDC model using different optimizers, both with and without transfer learning.

With transfer learning, the Adam optimizer achieved the highest performance, with an accuracy, precision, recall, and F1-score of 88.65%. Other optimizers such as Adagrad, AdamW, and RMSprop showed significantly lower performance, with accuracy ranging from 43.45% to 57.40% and F1-scores ranging from 42.03% to 55.64%.

Without transfer learning, the Adam optimizer again achieved the highest performance, with all metrics at 89.44%. Among other optimizers, RMSprop, Adamax, and Rprop demonstrated relatively better performance, with accuracy and precision reaching up to 52.05%, but still significantly lagged behind Adam. Some of the key observations from the table are as follows:

- The Adam optimizer consistently delivered the best performance for LightCDC, indicating its robustness and efficiency.
- Other optimizers showed lower accuracy and F1-scores, making Adam the most suitable choice.

4.4.3. Impact of learning rates

Table 9 examines the effect of varying learning rates on the performance of the LightCDC model, both with and without transfer learning. The experiments were conducted using a consistent batch size of 64, the Adam optimizer, and early stopping with a step setup of 10. The table presents metrics including accuracy, precision, recall, and F1-score for each learning rate tested.

With transfer learning, the highest performance was achieved with a learning rate of 0.001, showing an accuracy, precision, recall, and F1-score of 88.65%. Lower learning rates (0.0001) and higher learning rates (0.1) resulted in reduced performance, with accuracy and F1-scores dropping to 87.80% and 87.95%, respectively.

Without transfer learning, the optimal learning rate was also 0.001, achieving the highest accuracy, precision, recall, and F1-score of 89.44%. Lower learning rates (0.0001) and higher learning rates (0.1) showed decreased performance, with accuracy and F1-scores around 86.80% and 89.30%, respectively. Some of the key observations from the table are as follows:

Table 10
Evaluation results of our proposed LightCDC while training it with different approaches, including using transfer learning, without transfer learning, and with or without data augmentations. We train these combinations with a set of the best parameters that we get from our previous ablation studies in Tables 7, 8, and 9.

Variables	Proposed LightCDC			
	Accuracy	Precision	Recall	F1-score
Without transfer learning	**89.44**	**89.44**	**89.44**	**89.44**
Without transfer learning + Data augmentation	87.80	87.87	87.80	87.79
With transfer learning	88.65	88.66	88.65	88.65
With transfer learning + Data augmentation	87.40	87.43	87.40	87.40

- A learning rate of 0.001 consistently provided the best performance for LightCDC, with or without transfer learning.
- Both higher (0.1) and lower (0.0001) learning rates resulted in poorer performance.

4.4.4. Impact of transfer learning and data augmentation

Table 10 evaluates the performance of the LightCDC model under different combinations of transfer learning and data augmentation. The experiments used the best parameters identified in previous ablation studies as presented in Tables 7–9.

Key observations from the table indicate that the highest performance was achieved without transfer learning, with all metrics (accuracy, precision, recall, and F1-score) at 89.44%. Adding data augmentation without transfer learning resulted in a performance drop, with accuracy and F1-score decreasing to 87.80% and 87.79%, respectively. When using transfer learning, the model achieved high performance, with accuracy, precision, recall, and F1-score of 88.65%. However, combining transfer learning with data augmentation further reduced performance, with accuracy and F1-score at 87.40%.

4.4.5. Adverse real-life scenerios

The validation results of the robustness of our model under adverse real-life scenarios such as blur, low-light, and fog, applied at different intensity levels: light, moderate, and dense are presented in Table 11. Observing the table thoroughly, we noticed that the model consistently performed well for the light and moderate intense levels of different types of hazes, where it achieved 89.44% accuracy, precision, recall, and f1-score, which was also the accuracy of the model when tested on the original, unaltered test dataset. In addition, we found a noticeable but minimal decrease in performance as the severity of adverse conditions increases. For example, in the presence of dense blur, low-light, and fog, the accuracy drops slightly to 89.38%, 89.33%, and 89.31%, respectively. Overall, averaging all the values, we found the accuracy degree by only 0.03% even though we tested against a diverse unseen adverse dataset.

This slight degradation demonstrates that while our model is robust and can maintain high performance under mild and moderate adverse conditions, extreme situations like dense fog or low-light can challenge the model. However, even under the most adverse conditions, the performance metrics remain above 89%, showcasing the model's ability to generalize and reliably classify crops in challenging real-world scenarios. This further validates the robustness of LightCDC, making it suitable for deployment in diverse agricultural environments where such conditions may occur.

4.5. Visual results

For further analysis of our experimental results, we present loss curves for monitoring the training and validation losses over epochs, t-SNE for visualizing the high-dimensional feature representations learned by the models, confusion matrix for assessing the model's performance in classifying different classes and identifying any patterns of misclassification, GradCAM for understanding the regions of input images that contribute most to the model's predictions. Finally, we show some single-input image inference results using our proposed LightCDC model.

Table 11
Performance of the LightCDC under different adverse conditions such as Blur, Low-light, and Fog with varying intensity levels (Light, Moderate, and Dense).

Effects	Intensity	Accuracy	Precision	Recall	F1-score
Blur	Light	89.44	89.44	89.44	89.44
	Moderate	89.44	89.44	89.44	89.44
	Dense	89.38	89.41	89.35	89.40
Low-light	Light	89.44	89.44	89.44	89.44
	Moderate	89.44	89.40	89.43	89.40
	Dense	89.33	89.39	89.32	89.34
Fog	Light	89.44	89.44	89.44	89.44
	Moderate	89.44	89.44	89.44	89.44
	Dense	89.31	89.39	89.31	89.39
Average		89.41	89.40	89.40	89.42

The training and validation loss and accuracy curves for ConvNeXt-Large, ShuffleNetV2, ResNet101, and the proposed LightCDC model across 100 epochs, with a batch size of 64 and an early stopping patience of 15 steps are displayed in Fig. 4. ConvNeXtLarge shows a steady decrease in both training and validation losses, with high and consistent accuracy, indicating effective learning and good generalization. ShuffleNetV2 exhibits similar trends, with rapid improvement in accuracy and stabilization. ResNet101's loss curves show more fluctuations, particularly in validation loss, suggesting some instability, though accuracy remains consistent. The LightCDC model demonstrates a smooth decrease in training loss and minor fluctuations in validation loss that cause higher training accuracy over validation accuracy.

the t-SNE plots of the feature embeddings for ConvNeXtLarge, ShuffleNetV2, ResNet101, and the proposed LightCDC model are presented in Fig. 5. Each plot visualizes the distribution of the high-dimensional features in a 2D space, where green and red dots represent the non-damaged and damaged crop samples, respectively. ConvNeXtLarge and LightCDC display well-separated clusters, indicating strong feature discrimination between damaged and non-damaged crops. ShuffleNetV2 shows very distinct linear separations, suggesting effective clustering but with a simpler structure. ResNet101, while showing clusters, has more overlap between the classes compared to ConvNeXtLarge and LightCDC. These visualizations demonstrate that the LightCDC model effectively differentiates between damaged and non-damaged crops, comparable to ConvNeXtLarge and better than ResNet101, making it a viable option for crop damage classification.

The confusion matrices are presented in Fig. 6, comparing the classification performance of ConvNeXtLarge, ShuffleNetV2, ResNet101, and LightCDC on the test set of the CDC dataset. Each matrix illustrates the distribution of actual versus predicted classes for damaged and non-damaged crops.

ConvNeXtLarge (A) shows 880 true positives and 903 true negatives, with 120 false positives and 97 false negatives, indicating good overall performance but slightly lower precision due to a higher number of false positives. ShuffleNetV2 (B) has 885 true positives and

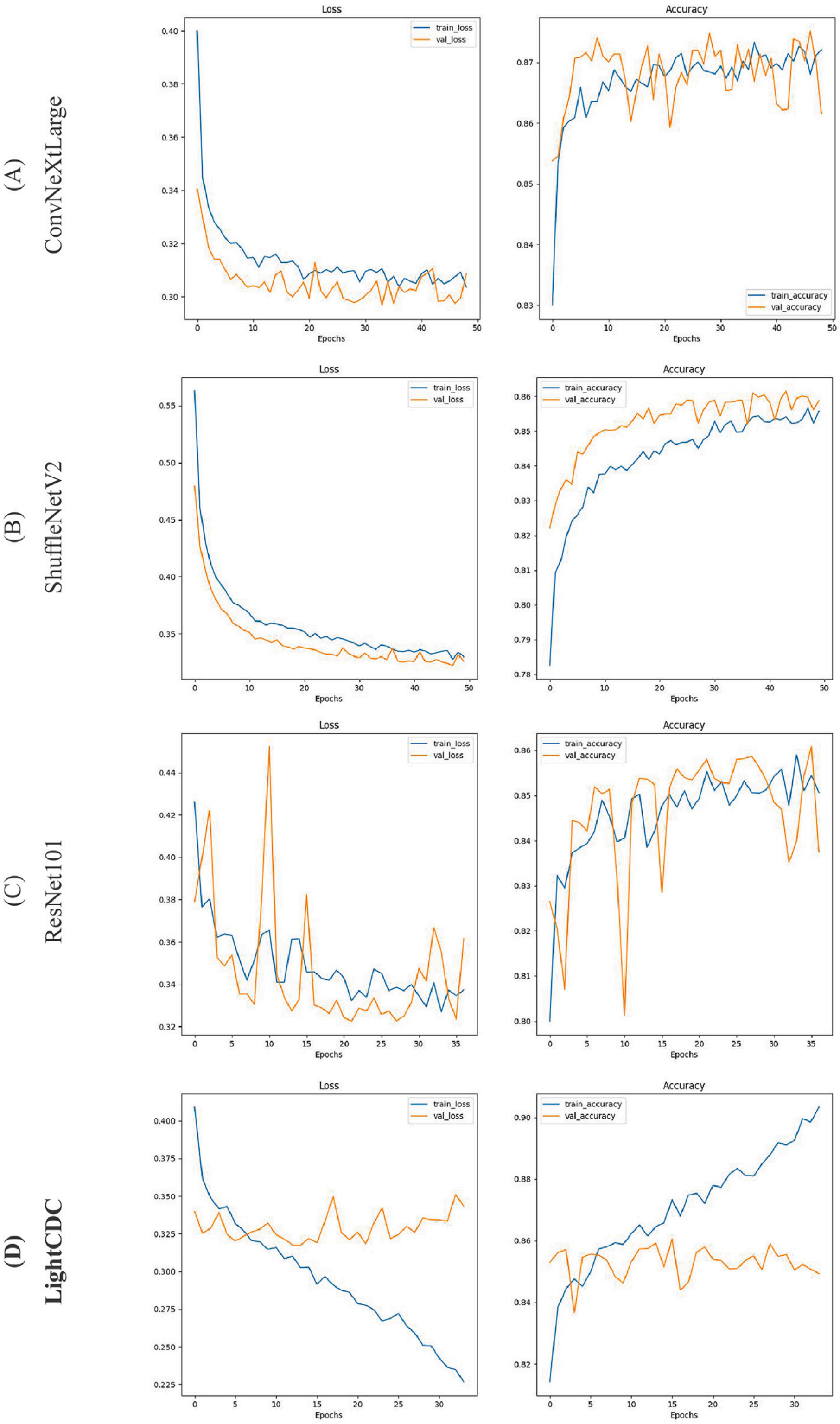
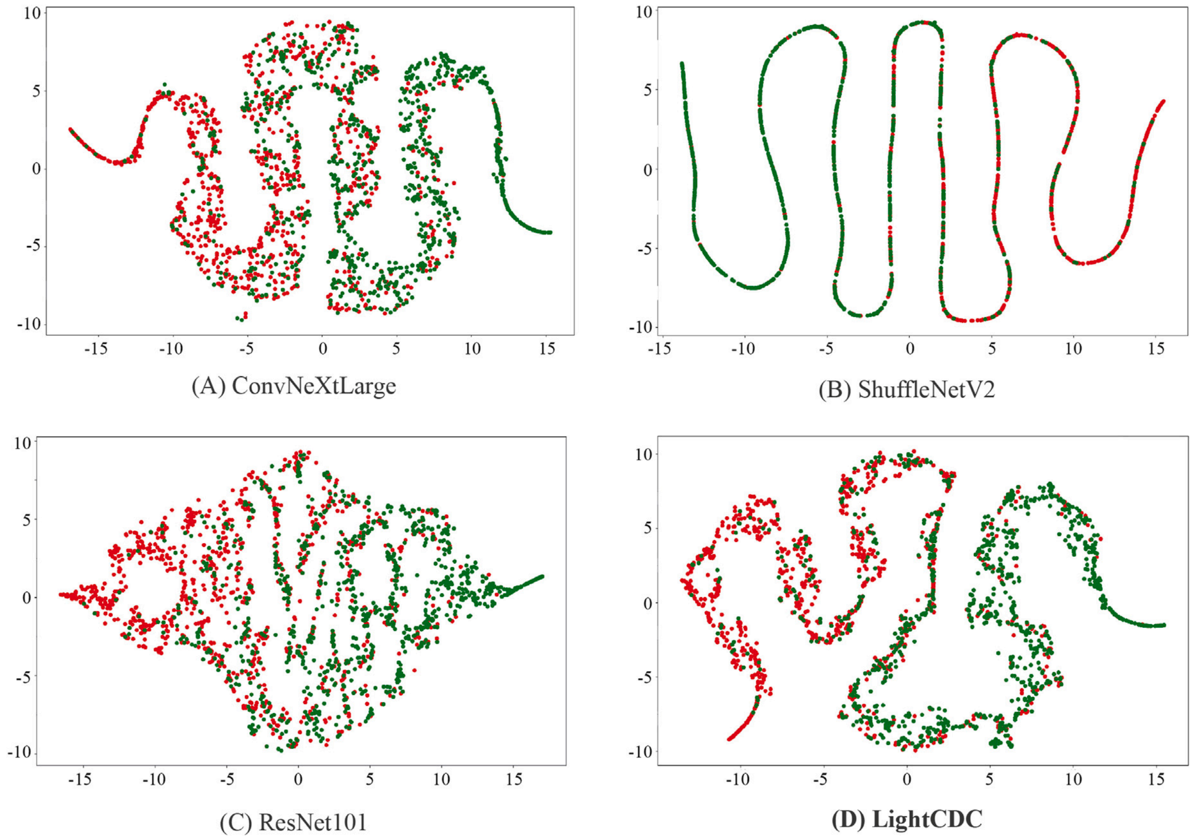


Fig. 4. Training and Validation Loss and Accuracy Curves for Different Models. The graphs display the training and validation loss and accuracy across epochs for ConvNeXtLarge, ShuffleNetV2, ResNet101, and the proposed LightCDC model.

881 true negatives, with 115 false positives and 119 false negatives, displaying a balanced performance but a notable number of false negatives impacting recall. ResNet101 (C) demonstrates 877 true positives and 887 true negatives, with 123 false positives and 113 false negatives, showing comparable performance to ConvNeXtLarge but slightly higher false positives and negatives, affecting both precision and recall. LightCDC (D) performs best among the models, with 879 true positives, 909 true negatives, and only 121 false positives and 91 false negatives. This model collectively exhibits the highest accuracy and precision, attributed to the lowest number of false positives and negatives.

In Fig. 7, we provide GradCAM visualizations for test images used in crop damage classification. Here, we applied GradCAM to visualize



(A) ConvNeXtLarge

(B) ShuffleNetV2

(C) ResNet101

(D) LightCDC

Fig. 5. Visualization of t-SNE plots for ConvNeXtLarge, ShuffleNetV2, ResNet101, and LightCDC, showcasing how each model clusters “***Damaged (red)***” and “***Non-damaged (green)***”s classes, thus illustrating the effectiveness of each model in feature separation by dimensionality reduction.

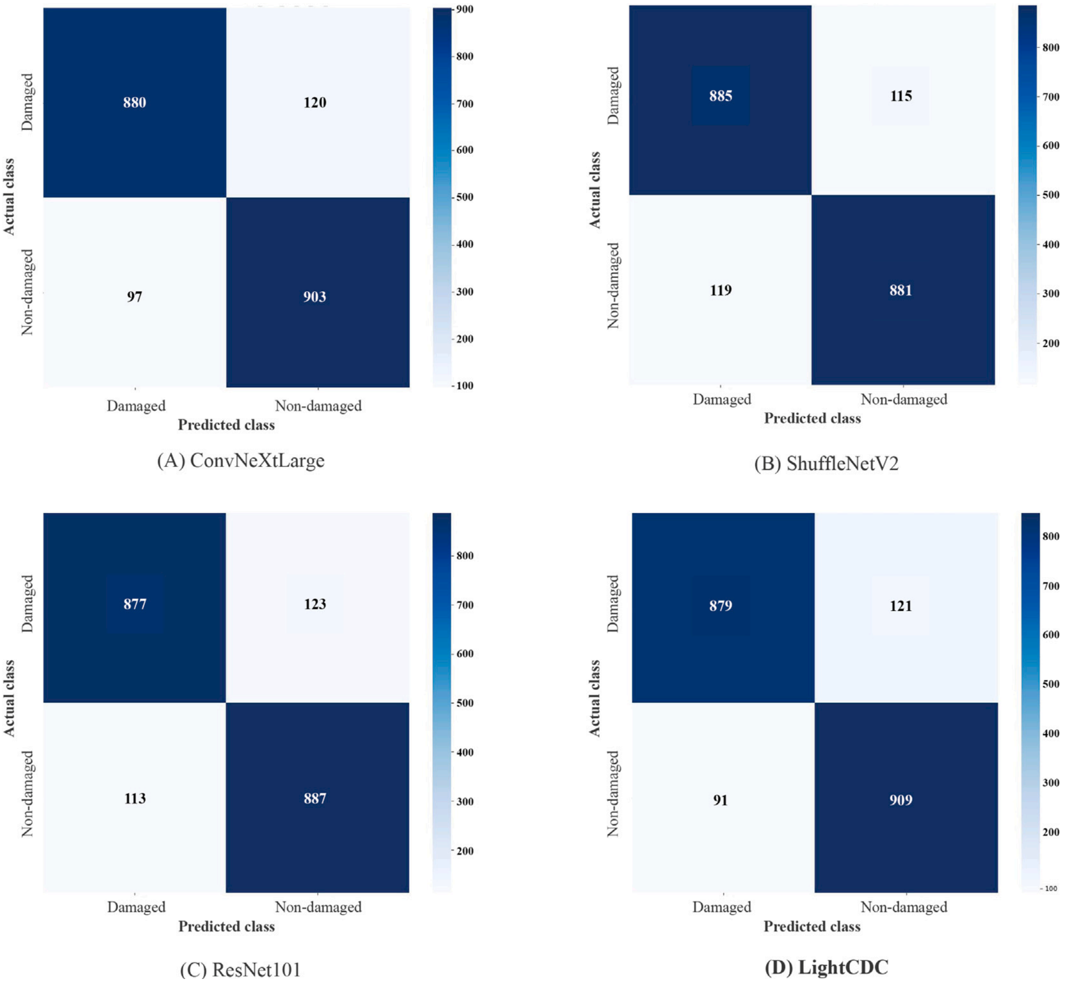
the areas of attention for four different models: ConvNeXtLarge, ResNet101, ShuffleNetV2, and our proposed LightCDC model. The heatmaps in the Fig. 7 illustrate the regions where each model focuses when classifying crops as Damaged or Non-damaged. The highlighted areas (colored regions) indicate the features and image regions that contributed most to the model’s final classification. For instance, the heatmaps generated by LightCDC clearly show its focus on the damaged parts of crops in the “Damaged” class, such as dry or discolored regions of the leaves and stems. Similarly, in the “Non-damaged” class, the attention is drawn to the lush, healthy regions of the crop, which are distinctly different from the damaged images. Furthermore, Fig. 7 also enables a comparative analysis between different models. ConvNeXtLarge and ResNet101, while producing clear heatmaps, tend to distribute attention across larger areas, which could introduce ambiguity in highly detailed tasks such as detecting specific types of crop damage. In contrast, ShuffleNetV2 and LightCDC exhibit more focused attention, particularly the LightCDC model, which effectively isolates the regions of crop damage, showing superior interpretability and robustness for this task. This makes LightCDC more suitable for practical applications where precise damage localization is critical.

Fig. 8 presents the results of single image inferences using LightCDC on test images from the processed CDC dataset. The figure showcases the model’s correct and incorrect predictions, with the predicted class, prediction probability, and whether the prediction was correct (green) or wrong (red) clearly indicated. Each image is classified as either Damaged or Non-damaged with varying levels of prediction probability. The model successfully predicts most cases as presented in the examples in Fig. 8, demonstrating high confidence in identifying damaged and healthy crops. However, a few cases, such as the images in the second column, show incorrect predictions, indicating the challenge the model faces in distinguishing subtle differences in crop health, especially under visually ambiguous conditions such as mixed crop health. These results highlight the overall robustness of the LightCDC model while also demonstrating areas where improvements in accuracy could be made, particularly in complex real-world scenarios.

In Fig. 9, the input images are from the “Non-damaged” class, while in Fig. 10, the input images belong to the “Damaged” class of our test dataset. We applied various adverse effects for both figures that the model might face under real-world scenarios, including blur, low light, and fog, at three intensity levels: light, moderate, and dense. These figures present the prediction outcomes and the model’s prediction confidence scores. We tested the model with two types of inputs: (A) relatively easy-to-predict images that are perceptually distinguishable and (B) relatively hard-to-predict images that are challenging to distinguish perceptually.

In Fig. 9 (Non-damaged class), for the relatively easy-to-predict images in part (A), the model performs consistently well with high confidence, even under moderate and dense adverse conditions. In contrast, in part (B), where the images are perceptually challenging, the model shows a slight decline in performance, particularly under dense low-light and foggy conditions. However, for light and moderate, it still can predict the classes correctly with a reasonable confidence score.

Observing the prediction outcomes for the damaged class input image shown in Fig. 10, we found LightCDC to perform accurately and with high confidence for the easy-to-predict images in part (A). Even though part (B) contains a more perceptually challenging input image, the model accurately predicted all the damaged crop images with high confidence scores for all the adverse effects like blur, low light, and fog. Which again validates the robustness of our proposed LightCDC in real-world scenarios. Overall, testing LightCDC against these unseen adverse data further validates its robustness and real-life applicability.



(A) ConvNeXtLarge

(B) ShuffleNetV2

(C) ResNet101

(D) LightCDC

Fig. 6. Confusion matrices for ConvNeXtLarge, ShuffleNetV2, ResNet101, and LightCDC, illustrating their classification performance on the test set of the CDC dataset by comparing actual versus predicted classes for “Damaged” and “Non-damaged classes”.

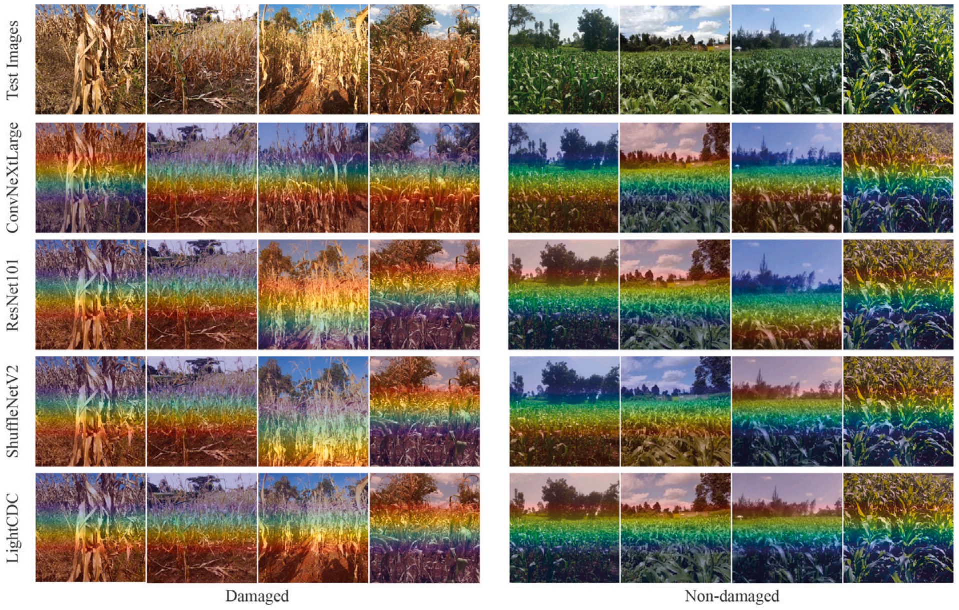


Fig. 7. Random test images of “Damaged” and “Non-Damaged” crops alongside GradCAM heatmaps generated by different models (ConvNeXtLarge, ResNet101, ShuffleNetV2, LightCDC), illustrating the models’ focus areas that influenced their classification decisions.

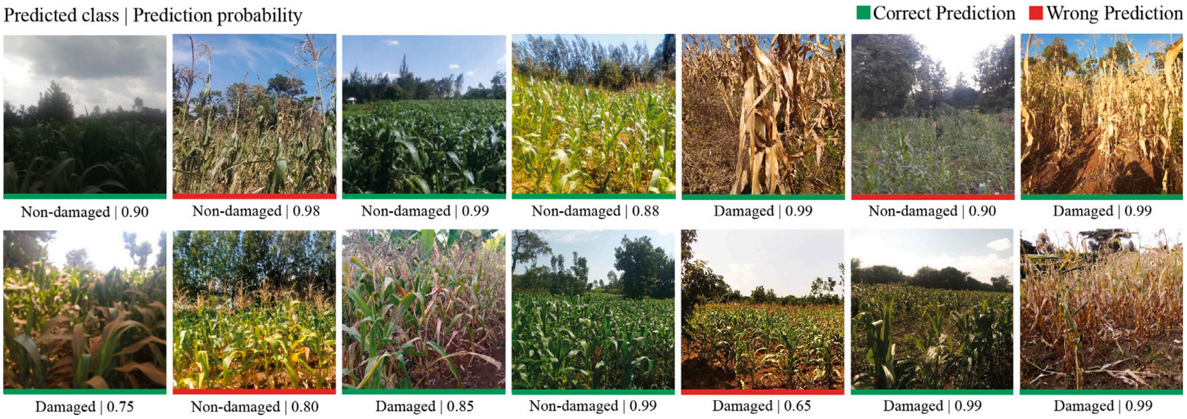


Fig. 8. Results of some single input image inference using our proposed LightCDC model. These images are from the test dataset of CDC dataset.

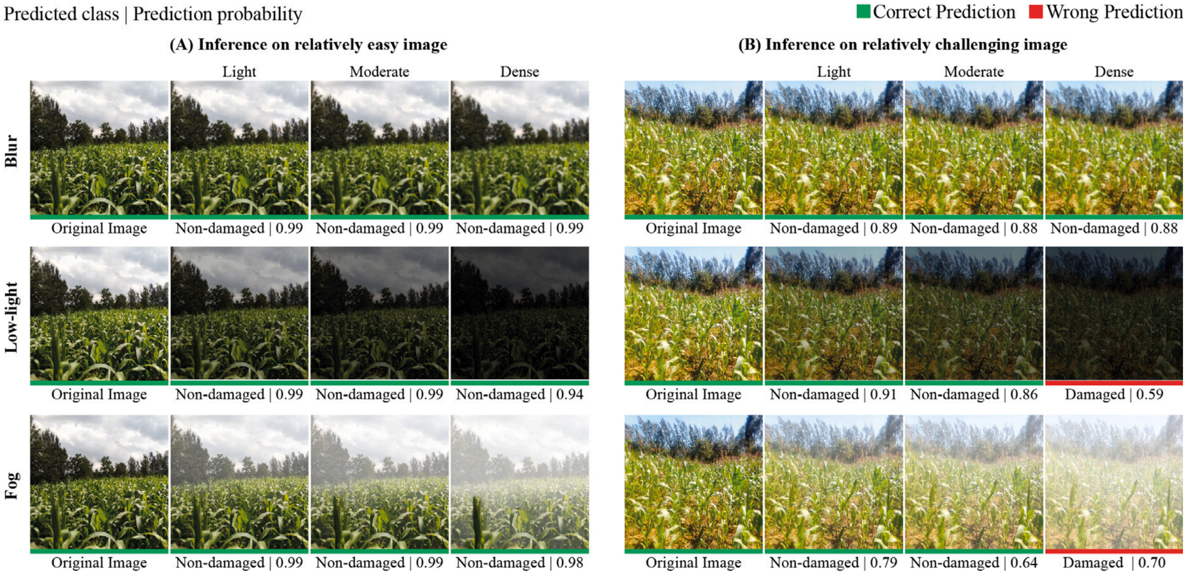


Fig. 9. Inference results on two example input images of class "***Non-damage***" from the test subset, where different effects such as blur, low-light, and fog at three intensity levels (light, moderate, and dense) were applied to test the model's robustness under real-life adverse conditions. **(A)** shows an example of a relatively easy-to-predict image, while **(B)** presents a more challenging image.

5. Limitations

Despite the promising results demonstrated by the LightCDC model for crop damage classification, several limitations should be acknowledged. Firstly, while the CDC dataset used in this study is comprehensive, it may not encompass the full diversity of crop damage scenarios encountered in different geographical regions, climates, and agricultural practices. For instance, the dataset may lack representations of specific crops, damage types, or agricultural diseases prevalent in other regions, limiting the model's generalizability to unseen conditions. Future work could expand the dataset to include a broader range of crops and environmental factors to improve model robustness.

Additionally, variations in weather and lighting conditions during image capture, such as low-light scenarios, shadows, fog, or rain, were not explicitly addressed in this study. While we performed some ablation studies in adverse conditions, they set a foundation for future dataset extension with more diverse real-life adverse scenes. These environmental factors, common in real-world agricultural settings, could impact the model's performance and reduce classification accuracy when deployed in diverse field conditions. For instance, the model may struggle to distinguish between damaged and non-damaged crops under heavy foggy weather conditions, where the visibility of key features could be compromised. Training the model with a more diverse dataset captured under these adverse conditions and incorporating more advanced data augmentation techniques that simulate such scenarios in the training process could be a potential area for improvement.

Moreover, while advantageous for deployment in resource-constrained environments, the model's lightweight architecture could limit its capacity to handle more complex features or very subtle damage types that heavier, more computationally intensive models might capture more effectively. Future improvements could focus on balancing model efficiency with performance, potentially by exploring hybrid models or combining multiple lightweight architectures to improve accuracy without significantly increasing computational costs.

Lastly, while explainability techniques such as GradCAM provide insights into the model's decision-making process, further work is needed to enhance interpretability for end-users, particularly farmers and agricultural workers, who may need more intuitive tools for understanding and acting on model outputs.

6. Conclusion

Crop damage classification is crucial for improving agricultural productivity and mitigating losses due to pests and diseases. In this study, we also introduced a newly processed dataset named CDC that can be used for binary classification to identify damaged crops. Leveraging

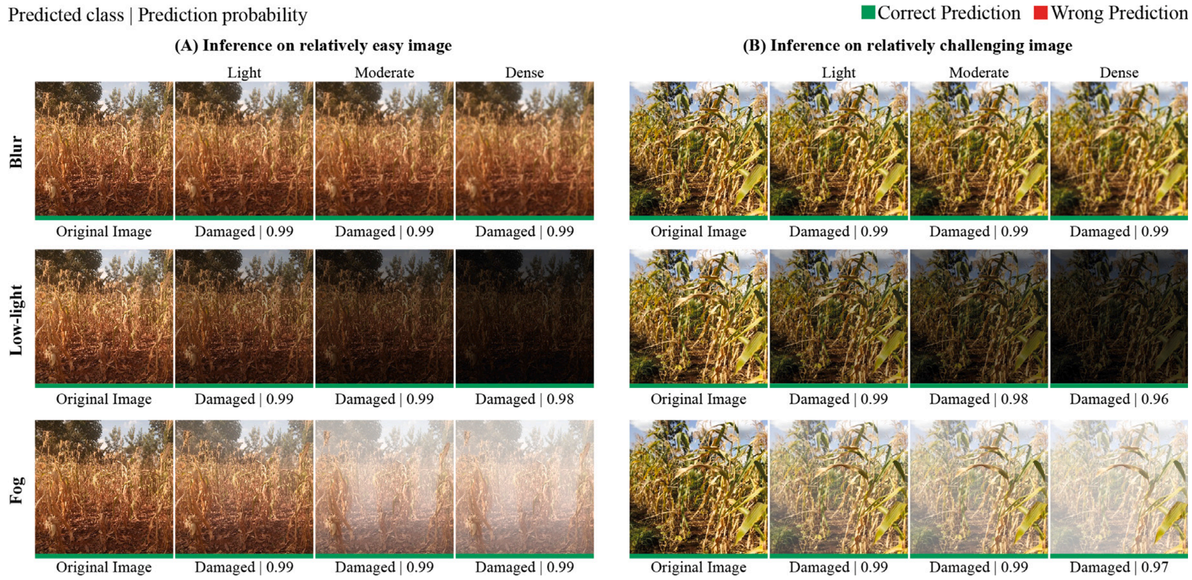


Fig. 10. Inference results on two example input images of class "***Damage***" from the test subset, where different effects such as blur, low-light, and fog at three intensity levels (light, moderate, and dense) were applied to test the model's robustness under real-life adverse conditions. **(A)** shows an example of a relatively easy-to-predict image, while **(B)** presents a more challenging image.

the dataset, we also developed the LightCDC model for crop damage classification. We leverage the ShuffleNetV2 architecture in our model to balance high performance with low computational requirements, making it suitable for real-time applications in resource-constrained environments. Our extensive experiments demonstrated that LightCDC achieves competitive accuracy, precision, recall, and F1-score compared to top-performing SOTA models while significantly reducing model size and inference time. Visual analyses, including GradCAM visualizations and t-SNE plots, confirmed the model's robust feature extraction and decision-making capabilities. Additionally, the confusion matrices illustrated LightCDC's strong classification performance with a low misclassification rate.

Despite these promising results, we identified several limitations, such as the need for a more diverse dataset and the impact of varying weather and lighting conditions on model performance. Future research should focus on expanding the dataset to include various crops, damage types, and environmental conditions from different regions. This will improve the model's generalizability across different agricultural landscapes and conditions. Additionally, optimizing LightCDC for deployment in resource-constrained environments remains a critical area for future work. This could involve exploring the integration of adaptive learning techniques, enabling the model to improve and adapt to new data continuously. Such adaptive capabilities could help address challenges posed by changing environmental conditions or new types of crop damage.

Moreover, future research could explore the cross-modality feature alignment techniques (Li et al., 2024) to integrate additional sensor data alongside visual crop data. This could further enhance the robustness and accuracy of crop damage classification models in varied agricultural environments. Additionally, developing multi-modal models that combine ground-level data with satellite or UAV imagery provides a more holistic view of crop health (Radoglou-Grammatikis et al., 2020), which could improve classification accuracy in larger fields. Another direction for extending this research is to employ YOLO models by labeling the dataset for detecting damaged areas in crops and subsequently classifying them. Although these approaches may increase the number of parameters, it has the potential to improve accuracy in crop damage detection and classification, which can be explored in future work.

In summary, our study and the proposed LightCDC model represent a significant step forward in applying advanced machine learning techniques to precision agriculture. By offering a scalable, efficient solution for real-time crop damage assessment, our approach lays the groundwork for future innovations in agricultural technology that can contribute to global food security and sustainable farming practices.

CRediT authorship contribution statement

Md Tanvir Islam: Conceptualization, Methodology, Data curation, Software, Validation, Investigation, Visualization, Formal analysis, Writing – original draft. **Safkat Shahrier Swapnil:** Conceptualization, Data curation, Visualization, Writing – original draft. **Md. Masum Billal:** Data curation, Visualization, Writing – original draft. **Asif Karim:** Formal analysis, Supervision, Writing – review & editing. **Niusha Shafiabady:** Investigation, Writing – review & editing. **Md. Mehedi Hassan:** Resources, Project administration, Writing – review & editing.

Funding

No external funding is available for this research.

Declaration of competing interest

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

Data and code availability

The processed CDC dataset, trained models, codes, and accompanying documentation are publicly available on GitHub to support future research and further extensions of this study.

References

Adhinata, F.D., Sumiharto, R., et al., 2024. A comprehensive survey on weed and crop classification using machine learning and deep learning. Artificial Intelligence in Agriculture.

Ahammad, M., Sani, A., Rahman, K., Islam, M.T., Masud, M.M.R., Hassan, M.M., Rony, M.A.T., Alam, S.M.N., Mukta, M.S.H., 2024. Roberta-gcn: A novel approach for combating fake news in bangla using advanced language processing and graph convolutional networks. IEEE Access.

Ahmad, I., Yang, Y., Yue, Y., Ye, C., Hassan, M., Cheng, X., Wu, Y., Zhang, Y., 2022. Deep learning based detector YOLOv5 for identifying insect pests. Appl. Sci. 12 (19), 10167.

Alam, I., Samiullah, M., Asaduzzaman, S., Kabir, U., Aahad, A., Woo, S.S., 2025. MIRACLE: Malware image recognition and classification by layered extraction. Data Min. Knowl. Discov. 39 (1), 10.

Ali, F., Qayyum, H., Iqbal, M.J., 2023. Faster-PestNet: A lightweight deep learning framework for crop pest detection and classification. IEEE Access.

Baijukya, F.P., Sabula, L., Mruma, S., Mzee, F., Mtoka, E., Masigo, J., Ndunguru, A., Swai, E., 2020. Maize production manual for smallholder farmers in Tanzania.

Ben Ayed, R., Hanana, M., 2021. Artificial intelligence to improve the food and agriculture sector. J. Food Quality 2021 (1), 5584754.

Bhargava, A., Shukla, A., Goswami, O., Alsharif, M.H., Uthansakul, P., Uthansakul, M., 2024. Plant leaf disease detection, classification and diagnosis using computer vision and artificial intelligence: A review. IEEE Access.

Bhatti, U.A., Bazai, S.U., Hussain, S., Fakhar, S., Ku, C.S., Marjan, S., Yee, P.L., Jing, L., 2023. Deep learning-based trees disease recognition and classification using hyperspectral data. Comput. Mater. Contin. 77 (1).

Bouguettaya, A., Zarzour, H., Kechida, A., Taberkit, A.M., 2021. Recent advances on UAV and deep learning for early crop diseases identification: A short review. In: 2021 International Conference on Information Technology. ICIT, IEEE, pp. 334–339.

CGIAR, 2024. CGIAR Crop Damage Classification Challenge: Can you identify types of damage in cellphone images of crops?. URL https://zindi.africa/competitions/cgiar-crop-damage-classification-challenge.

Chelotti, J.O., Martinez-Rau, L.S., Ferrero, M., Vignolo, L.D., Galli, J.R., Planisich, A.M., Rufiner, H.L., Giovanini, L.L., 2024. Livestock feeding behaviour: A review on automated systems for ruminant monitoring. Biosyst. Eng. 246, 150–177.

Chen, C.-J., Huang, Y.-Y., Li, Y.-S., Chen, Y.-C., Chang, C.-Y., Huang, Y.-M., 2021. Identification of fruit tree pests with deep learning on embedded drone to achieve accurate pesticide spraying. IEEE Access 9, 21986–21997. http://dx.doi.org/10.1109/ACCESS.2021.3056082.

Cracknell, A.P., 2017. UAVs: regulations and law enforcement. Int. J. Remote Sens. 38 (8–10), 3054–3067.

de Castro Pereira, R., Hirose, E., de Carvalho, O.L.F., da Costa, R.M., Borges, D.L., 2022. Detection and classification of whiteflies and development stages on soybean leaves images using an improved deep learning strategy. Comput. Electron. Agric. 199, 107132.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. Ieee, pp. 248–255.

Dhande, A., Malik, R., 2023. Design of a highly efficient crop damage detection ensemble learning model using deep convolutional networks. J. Ambient Intell. Humaniz. Comput. 14 (8), 10811–10821.

Doha, R., Al Hasan, M., Anwar, S., Rajendran, V., 2021. Deep learning based crop row detection with online domain adaptation. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. pp. 2773–2781.

Domingues, T., Brandão, T., Ferreira, J.C., 2022. Machine learning for detection and prediction of crop diseases and pests: A comprehensive survey. Agriculture 12 (9), 1350.

FAO, 2021. The impact of disasters and crises on agriculture and food security: 2021. Rome, http://dx.doi.org/10.4060/cb3673en.

Feng, A., Zhou, J., Vories, E.D., Sudduth, K.A., Zhang, M., 2020. Yield estimation in cotton using UAV-based multi-sensor imagery. Biosyst. Eng. 193, 101–114.

Giller, K.E., 2020. The food security conundrum of sub-Saharan Africa. Glob. Food Secur. 26, 100431.

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

Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., et al., 2019. Searching for mobilenetv3. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1314–1324.

Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q., 2017. Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4700–4708.

Huang, B., Zhi, L., Yang, C., Sun, F., Song, Y., 2020. Single satellite optical imagery dehazing using SAR image prior based on conditional generative adversarial networks. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1806–1813.

Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K., 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size.

Islam, M.T., Alam, I., Woo, S.S., Anwar, S., Lee, I., Muhammad, K., 2024a. LoLI-Street: Benchmarking low-light image enhancement and beyond. arXiv preprint arXiv:2410.09831.

Islam, M.T., Lee, I.H., Alzahrani, A.I., Muhammad, K., 2025. MEXFIC: A meta ensemble eXplainable approach for AI-synthesized fake image classification. Alex. Eng. J. 116, 351–363.

Islam, M.T., Rahim, N., Anwar, S., Muhammad, S., Bakshi, S., Muhammad, K., 2024b. HazeSpace2M: A dataset for haze aware single image dehazing. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 9155–9164. http://dx.doi.org/10.1145/3664647.3681382.

Islam, M.T., Raihan, M., Akash, S.R.I., Farzana, F., Aktar, N., 2019. Diabetes mellitus prediction using ensemble machine learning techniques. In: International Conference on Computational Intelligence, Security and Internet of Things. Springer, pp. 453–467.

Islam, M.T., Raihan, M., Farzana, F., Aktar, N., Ghosh, P., Kabiraj, S., 2020. Typical and non-typical diabetes disease prediction using random forest algorithm. In: 2020 11th International conference on computing, communication and networking technologies (ICCCNT). IEEE, pp. 1–6.

Istiak, M.A., Syeed, M.M., Hossain, M.S., Uddin, M.F., Hasan, M., Khan, R.H., Azad, N.S., 2023. Adoption of unmanned aerial vehicle (UAV) imagery in agricultural management: A systematic literature review. Ecol. Inform. 102305.

Jia, X., Cao, Y., O'Connor, D., Zhu, J., Tsang, D.C., Zou, B., Hou, D., 2021. Mapping soil pollution by using drone image recognition and machine learning at an arsenic-contaminated agricultural field. Environ. Pollut. 270, 116281.

Jiang, Z., Dong, Z., Jiang, W., Yang, Y., 2021. Recognition of rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning. Comput. Electron. Agric. 186, 106184.

Jintasuttisak, T., Edirisinghe, E., Elbattay, A., 2022. Deep neural network based date palm tree detection in drone imagery. Comput. Electron. Agric. 192, 106560.

Junagade, S., Choudhury, S.B., Sarangi, S., Pappula, S., 2022. Estimation of plucking points with overhead imaging in tea-a case study. In: 2022 IEEE Region 10 Symposium. TENSYMP, IEEE, pp. 1–6.

Kamilaris, A., Prenafeta-Boldú, F.X., 2018. A review of the use of convolutional neural networks in agriculture. J. Agric. Sci. 156 (3), 312–322.

Karen, S., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Karim, M.J., Goni, M.O.F., Nahiduzzaman, M., Ahsan, M., Haider, J., Kowalski, M., 2024. Enhancing agriculture through real-time grape leaf disease classification via an edge device with a lightweight CNN architecture and Grad-CAM. Sci. Rep. 14 (1), 16022.

Kasinathan, T., Singaraju, D., Uyyala, S.R., 2021. Insect classification and detection in field crops using modern machine learning techniques. Inf. Process. Agric. 8 (3), 446–457.

Keita, E., Kimura, T., Nobuhiko, I., Hiraguri, T., 2022. Semantic segmentation based field detection using drones. In: 2022 IEEE International Conference on Consumer Electronics-Taiwan. IEEE, pp. 213–214.

Khattak, A., Asghar, M.U., Batool, U., Asghar, M.Z., Ullah, H., Al-Rakhami, M., Gumaei, A., 2021. Automatic detection of citrus fruit and leaves diseases using deep neural network model. IEEE Access 9, 112942–112954.

Kotwal, J., Kashyap, D., Pathan, D., 2023. Agricultural plant diseases identification: From traditional approach to deep learning. Mater. Today: Proc. (ISSN: 2214-7853) 80, 344–356. http://dx.doi.org/10.1016/j.matpr.2023.02.370, 3rd International Congress on Mechanical and Systems Engineering (CAMSE 2022).

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. 25.

Li, X., Yu, S., Lei, Y., Li, N., Yang, B., 2024. Dynamic vision-based machinery fault diagnosis with cross-modality feature alignment. IEEE/CAA J. Autom. Sin. 11 (10), 2068–2081.

Li, X., Zhang, W., Li, X., Hao, H., 2023. Partial domain adaptation in remaining useful life prediction with incomplete target data. IEEE/ASME Trans. Mechatronics.

Li, W., Zheng, T., Yang, Z., Li, M., Sun, C., Yang, X., 2021. Classification and detection of insects from field images using deep learning for smart pest management: A systematic review. Ecol. Inform. 66, 101460.

Liu, H., Chang, Z., Zhao, S., Gong, P., Zhang, M., Lozano-Durán, R., Yan, H., Zhou, X., Li, F., 2023. Functional identification of a novel C7 protein of tomato yellow leaf curl virus. Virology 585, 117–126.

Liu, H.-I., Galindo, M., Xie, H., Wong, L.-K., Shuai, H.-H., Li, Y.-H., Cheng, W.-H., 2024a. Lightweight deep learning for resource-constrained environments: A survey. ACM Comput. Surv..

Liu, Z., Li, J., Ashraf, M., Syam, M., Asif, M., Awwad, E.M., Al-Razgan, M., Bhatti, U.A., 2024b. Remote sensing-enhanced transfer learning approach for agricultural damage and change detection: A deep learning perspective. Big Data Res. 36, 100449.

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., Xie, S., 2022. A convnet for the 2020s. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11976–11986.

Ma, Y., Zhang, Z., Yang, H.L., Yang, Z., 2021. An adaptive adversarial domain adaptation approach for corn yield prediction. Comput. Electron. Agric. 187, 106314.

Ma, N., Zhang, X., Zheng, H.-T., Sun, J., 2018. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In: Proceedings of the European Conference on Computer Vision. ECCV, pp. 116–131.

Van der Maaten, L., Hinton, G., 2008. Visualizing data using t-SNE. J. Mach. Learn. Res. 9 (11).

MacEachern, C.B., Esau, T.J., Schumann, A.W., Hennessy, P.J., Zaman, Q.U., 2023. Detection of fruit maturity stage and yield estimation in wild blueberry using deep learning convolutional neural networks. Smart Agric. Technol. 3, 100099.

Malek, M., Dhiraj, B., Upadhyaya, D., Patel, D., 2022. A review of precision agriculture methodologies, challenges, and applications. In: Emerging Technologies for Computing, Communication and Smart Cities: Proceedings of ETCCS 2021. Springer, pp. 329–346.

Mashamba, M.M., Telukdarie, A., Munien, I., Onkonkwo, U., Vermeulen, A., 2024. Detection of bacterial spot disease on tomato leaves using a Convolutional Neural Network (CNN). Procedia Comput. Sci. 237, 602–609.

Meena, S.D., Susank, M., Guttula, T., Chandana, S.H., Sheela, J., 2023. Crop yield improvement with weeds, pest and disease detection. Procedia Comput. Sci. 218, 2369–2382.

Memon, M.S., Kumar, P., Iqbal, R., 2022. Meta deep learn leaf disease identification model for cotton crop. Computers 11 (7), 102.

Misra, D., 2019. Mish: A self regularized non-monotonic activation function. arXiv preprint arXiv:1908.08681.

Moazzam, S.I., Khan, U.S., Qureshi, W.S., Tiwana, M.I., Rashid, N., Alasmary, W.S., Iqbal, J., Hamza, A., 2021. A patch-image based classification approach for detection of weeds in sugar beet crop. IEEE Access 9, 121698–121715.

Mondal, D., Roy, K., Pal, D., Kole, D.K., 2022. Deep learning-based approach to detect and classify signs of crop leaf diseases and pest damage. SN Comput. Sci. 3 (6), 433.

Moupojou, E., Tagne, A., Retraint, F., Tadonkemwa, A., Wilfried, D., Tapamo, H., Nkenlifack, M., 2023. FieldPlant: A dataset of field plant images for plant disease detection and classification with deep learning. IEEE Access 11, 35398–35410.

Ncube, B., Mupangwa, W., French, A., 2018. Precision agriculture and food security in Africa. Syst. Anal. Approach Complex Glob. Chall. 159–178.

Piani, M., Bortolotti, G., Manfrini, L., 2021. Apple orchard flower clusters density mapping by unmanned aerial vehicle RGB acquisitions. In: 2021 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor). IEEE, pp. 92–96.

Picon, A., Seitz, M., Alvarez-Gila, A., Mohnke, P., Ortiz-Barredo, A., Echazarra, J., 2019. Crop conditional Convolutional Neural Networks for massive multi-crop plant disease classification over cell phone acquired images taken on real field conditions. Comput. Electron. Agric. 167, 105093.

Pun, T.B., Neupane, A., Koech, R., Walsh, K., 2023. Detection and counting of root-knot nematodes using YOLO models with mosaic augmentation. Biosens. Bioelectron.: X 15, 100407.

Radoglou-Grammatikis, P., Sarigiannidis, P., Lagkas, T., Moscholios, I., 2020. A compilation of UAV applications for precision agriculture. Comput. Netw. 172, 107148.

Raimi, L., Panait, M., Sule, R., 2021. Leveraging precision agriculture for sustainable food security in sub-Saharan Africa: a theoretical discourse. Shifting Patterns Agric. Trade: Prot. Outbreak Food Secur. 491–509.

Rani, A.P.A.S., Singh, N.S., 2022. Protecting the environment from pollution through early detection of infections on crops using the deep belief network in paddy. Total. Environ. Res. Themes 3, 100020.

Reynolds, T.W., Waddington, S.R., Anderson, C.L., Chew, A., True, Z., Cullen, A., 2015. Environmental impacts and constraints associated with the production of major food crops in Sub-Saharan Africa and South Asia. Food Secur. 7, 795–822.

Sakkarvarthi, G., Sathianesan, G.W., Murugan, V.S., Reddy, A.J., Jayagopal, P., Elsisi, M., 2022. Detection and classification of tomato crop disease using convolutional neural network. Electronics 11 (21), 3618.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.-C., 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4510–4520.

Sanida, M.V., Sanida, T., Sideris, A., Dasygenis, M., 2023. An Efficient hybrid cnn classification model for tomato crop disease. Technologies 11 (1), 10.

Sarkar, C., Gupta, D., Gupta, U., Hazarika, B.B., 2023. Leaf disease detection using machine learning and deep learning: Review and challenges. Appl. Soft Comput. 145, 110534.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D., 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 618–626.

Shanthi, D., Vinutha, K., Ashwini, N., Vashistha, S., 2024. Tomato leaf disease detection using CNN. Procedia Comput. Sci. 235, 2975–2984.

Shih, M.-J., Chiou, Y.-S., Chen, Y.-C., 2021. Detection and interpretation of transplanted positions using drone's eye-view images for rice paddies. In: 2021 IEEE 3rd Eurasia Conference on IOT, Communication and Engineering. ECICE, IEEE, pp. 1–5.

da Silva Vieira, G., Rocha, B.M., Fonseca, A.U., de Sousa, N.M., Ferreira, J.C., Cabacinha, C.D., Soares, F., 2022. Automatic detection of insect predation through the segmentation of damaged leaves. Smart Agric. Technol. 2, 100056.

Singh, P.P., Kumar, D., Srivastava, A., Basumatary, M., Prasad, S., 2023. A CNN model based approach for disease detection in mango plant leaves. In: International Conference on Soft Computing for Problem-Solving. Springer, pp. 389–399.

Singh, A., Vaidya, G., Jagota, V., Darko, D.A., Agarwal, R.K., Debnath, S., Potrich, E., 2022. Recent advancement in postharvest loss mitigation and quality management of fruits and vegetables using machine learning frameworks. J. Food Quality 2022, 1–9.

Sousa, J.J., Toscano, P., Matese, A., Di Gennaro, S.F., Berton, A., Gatti, M., Poni, S., Pádua, L., Hruška, J., Morais, R., et al., 2022. UAV-based hyperspectral monitoring using push-broom and snapshot sensors: A multisite assessment for precision viticulture applications. Sensors 22 (17), 6574.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–9.

Tabar, M., Gluck, J., Goyal, A., Jiang, F., Morr, D., Kehs, A., Lee, D., Hughes, D.P., Yadav, A., 2021. A PLAN for tackling the locust crisis in east Africa: harnessing spatiotemporal deep models for locust movement forecasting. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. pp. 3595–3604.

Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q.V., 2019. Mnasnet: Platform-aware neural architecture search for mobile. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2820–2828.

Tan, M., Le, Q., 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning. PMLR, pp. 6105–6114.

Tatini, N.B., Lu, G.-Y., Tan, T.-H., Alkhaleefah, M., Koo, V.C., Chan, Y.K., Chang, Y.-L., 2022. Yolov4 based rice fields classification from high-resolution images taken by drones. In: IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 5043–5046.

Thierfelder, C., Matemba-Mutasa, R., Bunderson, W.T., Mutenje, M., Nyagumbo, I., Mupangwa, W., 2016. Evaluating manual conservation agriculture systems in southern Africa. Agric. Ecosyst. Environ. 222, 112–124.

Tirkey, D., Singh, K.K., Tripathi, S., 2023a. Performance analysis of AI-based solutions for crop disease identification, detection, and classification. Smart Agric. Technol. (ISSN: 2772-3755) 5, 100238. http://dx.doi.org/10.1016/j.atech.2023.100238.

Tirkey, D., Singh, K.K., Tripathi, S., 2023b. Performance analysis of AI-based solutions for crop disease identification, detection, and classification. Smart Agric. Technol. 5, 100238.

Van Klompenburg, T., Kassahun, A., Catal, C., 2020. Crop yield prediction using machine learning: A systematic literature review. Comput. Electron. Agric. 177, 105709.

Verschuur, J., Li, S., Wolski, P., Otto, F.E., 2021. Climate change as a driver of food insecurity in the 2007 Lesotho-South Africa drought. Sci. Rep. 11 (1), 3852.

Waldamichael, F.G., Debelee, T.G., Schwenker, F., Ayano, Y.M., Kebede, S.R., 2022. Machine learning in cereal crops disease detection: a review. Algorithms 15 (3), 75.

Wibowo, H., Sitanggang, I.S., Mushthofa, M., Adrianto, H.A., 2022. Large-scale oil palm trees detection from high-resolution remote sensing images using deep learning. Big Data Cogn. Comput. 6 (3), 89.

Widén, A., Clinchy, M., Felton, A.M., Hofmeester, T.R., Kuijper, D.P., Singh, N.J., Widemo, F., Zanette, L.Y., Cromsigt, J.P., 2022. Playbacks of predator vocalizations reduce crop damage by ungulates. Agric. Ecosyst. Environ. 328, 107853.

Xiong, J., Liu, Z., Chen, S., Liu, B., Zheng, Z., Zhong, Z., Yang, Z., Peng, H., 2020. Visual detection of green mangoes by an unmanned aerial vehicle in orchards based on a deep learning method. Biosyst. Eng. 194, 261–272.

Xu, L., Cao, B., Zhao, F., Ning, S., Xu, P., Zhang, W., Hou, X., 2023. Wheat leaf disease identification based on deep learning algorithms. Physiol. Mol. Plant Path. 123, 101940.

Yang, Q., Shi, L., Han, J., Zha, Y., Zhu, P., 2019. Deep convolutional neural networks for rice grain yield estimation at the ripening stage using UAV-based remotely sensed images. Field Crops Res. 235, 142–153.

Yaqot, M., Menezes, B.C., Al-Ansari, T., 2023. Roadmap to precision agriculture under circular economy constraints. J. Inf. Knowl. Manag. 22 (05), 2250092.

Yu, N., Li, L., Schmitz, N., Tian, L.F., Greenberg, J.A., Diers, B.W., 2016. Development of methods to improve soybean yield estimation and predict plant maturity with an unmanned aerial vehicle based platform. Remote Sens. Environ. 187, 91–101.

Zhang, T., Li, K., Chen, X., Zhong, C., Luo, B., Grijalva, I., McCornack, B., Flippo, D., Sharda, A., Wang, G., 2023. Aphid cluster recognition and detection in the wild using deep learning models. Sci. Rep. 13 (1), 13410.

Zhang, M., Zhou, J., Sudduth, K.A., Kitchen, N.R., 2020. Estimation of maize yield and effects of variable-rate nitrogen application using UAV-based RGB imagery. Biosyst. Eng. 189, 24–35.

Zhao, G., Zhang, Y., Lan, Y., Deng, J., Zhang, Q., Zhang, Z., Li, Z., Liu, L., Huang, X., Ma, J., 2023. Application progress of UAV-LARS in identification of crop diseases and pests. Agronomy 13 (9), 2232.

Zhou, J., Zhou, J., Ye, H., Ali, M.L., Chen, P., Nguyen, H.T., 2021. Yield estimation of soybean breeding lines under drought stress using unmanned aerial vehicle-based imagery and convolutional neural network. Biosyst. Eng. 204, 90–103.

Zhu, H., Lin, C., Liu, G., Wang, D., Qin, S., Li, A., Xu, J.-L., He, Y., 2024. Intelligent agriculture: Deep learning in UAV-based remote sensing imagery for crop diseases and pests detection. Front. Plant Sci. 15, 1435016.