Hybrid CNN Approach for Post-Disaster Building Damage Classification Using Satellite Imagery

Sahat Sonang[1,*], Y Yuhandri[2], Muhammad Tajuddin[3]

[1]*Doctoral Student, Information Technology Study Program, Universitas Putra Indonesia YPTK Padang, Indonesia*

[2]*Information Technology Study Program, Universitas Putra Indonesia YPTK Padang, Indonesia*

[3]*Department of Computer Science, Universitas Bumigora Mataram, Indonesia*

(Received: March 25, 2025; Revised: June 18, 2025; Accepted: September 18, 2025; Available online: October 04, 2025)

Abstract

Accurate post-disaster building damage assessment is critical for timely response and effective reconstruction planning. This study proposes a hybrid deep learning architecture that integrates Inception-ResNet-v2 and EfficientNetV2B0, designed to enhance post-disaster damage classification from high-resolution satellite imagery. The model leverages dual-stream feature extraction, followed by concatenated fully connected layers optimized with dropout and batch normalization to improve generalization and reduce overfitting. The objective is to outperform standard Convolutional Neural Network (CNN) models in terms of classification and segmentation performance across multiple damage categories: no damage, minor damage, major damage, destroyed, and unclassified. The model was trained and validated on the publicly available xView dataset, covering over 12,000 annotated images from various natural disasters. Comparative evaluation against ResNet, GoogleNet, DenseNet, and EfficientNet demonstrates that the proposed model achieves the highest accuracy (86%), precision (85%), recall (86%), and F1-score (84%). Furthermore, it outperforms all baseline models in segmentation metrics, achieving an Intersection over Union (IoU) score of 0.7749 and a Dice Similarity Coefficient (DSC) of 0.8726. The model also significantly reduces misclassification rates in critical categories such as "major damage" and "destroyed." A Wilcoxon signed-rank test confirmed that these improvements are statistically significant ($p < 0.05$) across all major performance indicators. The novelty of this study lies in the fusion of two state-of-the-art CNN backbones with tailored architectural modifications, yielding a robust and generalizable model suitable for automated disaster damage assessment. This research contributes a scalable deep learning approach that can be integrated into real-time or semi-automated disaster response systems, offering improved decision-making support in emergency contexts. The results affirm the model's potential as a reliable tool in post-disaster scenarios and set a foundation for future work in multi-modal and real-time AI-based disaster management.

Keywords: Deep Learning, Disaster Damage Classification, CNN, Satellite Imagery, Building Damage Detection

1. Introduction

The advancement of information technology has significantly impacted disaster management, particularly in damage assessment using satellite imagery [1]. In this study, we address the critical challenge of building damage classification from post-disaster satellite data using deep learning techniques [2], [3]. No longer just a supporting tool, information technology has become a critical component across multiple sectors [4]. It plays a vital role in managing data and information, including processing, acquiring, organizing, storing, and manipulating data through diverse methods to produce high-quality information—information that is relevant, accurate, and timely. Such information is essential for personal, business, and governmental purposes. Currently, information technology is widely applied in fields such as education, healthcare, business, geology, and in detecting building damage caused by natural disasters [5], [6].

Building damage resulting from natural disasters poses significant challenges, both in terms of human casualties and property loss. From 2010 to 2019, global natural disasters caused nearly 267,480 fatalities and massive economic losses, particularly when compounded by secondary disasters such as landslides, wildfires, tsunamis, and floods [7]. The vulnerability of humans to fatal threats is exacerbated in densely populated areas, especially in large urban centers

***Corresponding author: Sahat Sonang (sahatsonangstg@gmail.com)**

DOI: https://doi.org/10.47738/jads.v6i4.931

impacted by high-intensity earthquakes [8]. Consequently, accurate information on affected areas, collapsed buildings, and types of damage is crucial for facilitating rescue efforts, estimating economic losses, and supporting reconstruction processes. However, gathering such information through field surveys is challenging, particularly in inaccessible areas, and often requires significant financial resources.

The availability of remote sensing imagery enables the observation of Earth's surface from various perspectives and resolutions. Leveraging diverse sensors, including Light Detection and Ranging (LiDAR), Synthetic Aperture Radar (SAR), and optical imagery, has significantly advanced the mapping of building damage caused by natural disasters. The development of high-resolution spatial imagery enhances the ability to distinguish different land objects through remote sensing. Satellite imagery has increasingly become an essential tool for observing real-world information in recent years [9]. Modern remote sensing systems can capture imagery in various forms and resolutions [10]. Image engineering, an interdisciplinary field that continues to grow, combines theories, technologies, and applications from fields such as mathematics, computer science, and optics. This approach integrates advancements in electronics and imaging technologies to address a wide range of imaging challenges [11], [12].

Applications of high-resolution remote sensing for rapid damage detection are among the most critical needs during emergency responses. Deep learning has proven effective in improving building damage detection performance through automated feature extraction, involving four main stages: pre-processing, deep feature extraction, deep feature fusion, and transfer learning with ResNet-18 [13]. This study focuses on developing a deep learning-based architectural model capable of classifying building damage from high-resolution satellite imagery into two categories: (1) detection based on changes between pre-disaster and post-disaster images, and (2) detection based solely on post-disaster images. Based on existing disaster management reports and literature, typical field surveys cost USD 1,000–3,000 per km² and may take days to weeks to complete, depending on accessibility.

Several prior studies have explored building damage detection using post-disaster satellite imagery. For instance, [14] applied CNN to enhance detection accuracy, achieving over 85%. Another study by [15] used CNN to predict the extent of building damage, with the proposed model achieving a classification accuracy of 98%. Similarly, [16] simulated 5,750 images, achieving 95% accuracy and 97% precision. Research by [17] introduced a multiple-input CNN (MI-CNN) for regional seismic damage assessment, yielding an overall prediction accuracy of 79.7%. Furthermore, [18] evaluated four CNN models—VGG16, Inception V3, ResNet50, and DenseNet121—for detecting earthquake-damaged buildings. The results indicated an improvement in overall accuracy from 64.3% to 88.9%, along with a significant increase in recall for damaged building classes. Additionally, [19] utilized R-CNN and SSD for remote sensing image target recognition, achieving over 80% accuracy in several target categories.

Building upon these previous studies, this research develops a novel model based on the ResNet-V2 and EfficientNet-V2 architectures to classify post-disaster building damage using high-resolution satellite imagery. The proposed model is expected to deliver higher accuracy in damage detection. The model evaluation employs high-resolution post-disaster satellite imagery datasets, aiming to demonstrate significant performance improvements and contribute to advancements in meteorology, climatology, and geophysics. The results are anticipated to provide a more effective solution for image classification in post-disaster damage detection. This research contributes to the development of a novel CNN architecture, named the Proposed Model, designed for post-disaster building damage detection. By optimizing hyperparameters, the Proposed Model enhances classification performance and accuracy. Comprehensive evaluations and comparisons with other classification models, including GoogleNet (Inception V3), ResNet, DenseNet, and EfficientNet, provide valuable insights into the strengths and limitations of each approach. This study is expected to advance satellite image-based classification techniques and algorithms, offering improved accuracy for post-disaster building damage detection, ultimately supporting more effective disaster mitigation efforts in the future. The choice of Inception-ResNet-v2 and EfficientNetV2B0 was based on their proven balance between feature extraction capability, parameter efficiency, and prior empirical success in remote sensing tasks. While ConvNeXt and Swin Transformer are promising, their higher computational demand and requirement for larger datasets were less aligned with our practical deployment goals.

2. Methodology

2.1. Dataset Used

The xView Dataset is a high-resolution satellite image collection developed by the Defense Innovation Unit (DIU) to improve object recognition in satellite imagery. It contains over 1,400 km² of data with a spatial resolution of approximately 0.3 meters per pixel [20]. Each object in the dataset is manually annotated by experts, providing accurate class labels and bounding boxes. The annotations are in JSON format, while the imagery is provided in GeoTIFF format, which includes geospatial information and is compatible with GIS software [21].

The dataset consists of 12,738 high-resolution satellite images, categorized by disaster events such as tornadoes, wildfires, floods, tsunamis, and volcanic eruptions. We used a combination of random oversampling and image augmentation (rotation, flipping, brightness adjustment) rather than simple duplication, to improve generalization. The largest categories are the Portugal Wildfire with 3,738 images and the Pinery Bushfire with 3,690 images. Other notable groups include the Woolsey Fire (1,756 images), Nepal Flooding (1,238 images), and Tuscaloosa Tornado (686 images). Smaller categories include the Lower Puna Volcano (582 images), Moore Tornado (454 images), Joplin Tornado (298 images), and Sunda Tsunami (296 images). This dataset is a valuable resource for developing and testing algorithms for disaster response. In all experiments, we used an 80:20 split for training and testing. During training, 10% of the training set was reserved internally for validation by the Keras training procedure, which was used for early stopping. The 20% test set was kept strictly separate and used only for final evaluation. Figure 1 shows example images from the xView dataset, including both pre-disaster and post-disaster conditions, in annotated and non-annotated formats. These visuals illustrate the diversity and complexity of the satellite imagery used in this study.

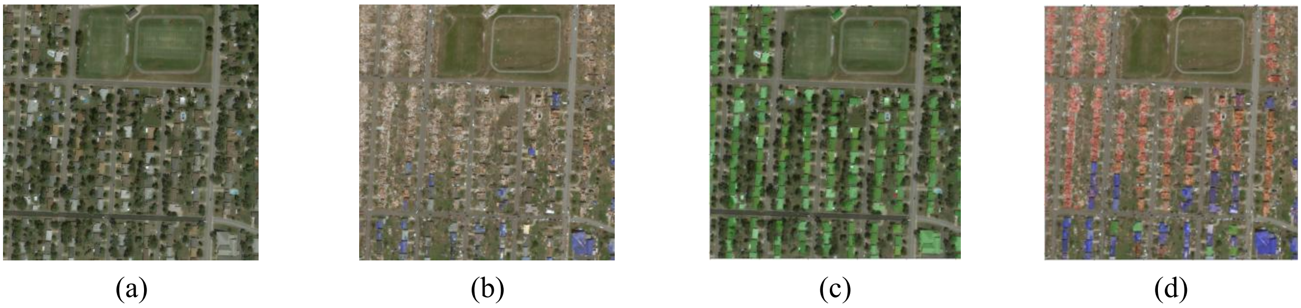


Figure 1. Sample Images from the Annotated xView Dataset: (a) Pre Disaster (not annotated), (b) Post Disaster (not annotated), (c) Pre Disaster (annotated), and (d) Pre Disaster (annotated)

In this study (figure 1), buildings labeled as 'unclassified' in the xView dataset were retained as a separate category to preserve label consistency. To address potential class imbalance, oversampling was applied to underrepresented classes, particularly 'minor damage' and 'destroyed'.

2.2. Proposed Model

The proposed model combines two convolutional neural network architectures that have demonstrated strong performance in image classification tasks: Inception-ResNet-v2 and EfficientNetV2B0 [22], [23]. These architectures are employed as feature extractors without their fully connected classification layers (include_top=False). The model is designed to exploit the complementary strengths of each backbone to improve feature representation and classification accuracy. The complete layer composition of the proposed model is presented in table 1, which outlines the architecture starting from input layers, through feature extraction, concatenation, and ending with dense layers for classification. By combining both pretrained backbones, the model benefits from rich and diverse feature representations. GlobalAveragePooling is applied to each backbone's output, followed by concatenation and fully connected layers equipped with dropout and batch normalization to prevent overfitting and enhance stability.

Table 1. Layer Composition of the Proposed Model

Layer Type	Output Shape	Activation	Details
Input Layer (Inception-ResNet-v2)	(224, 224, 3)	-	Pretrained Inception-ResNet-v2, weights='imagenet', include_top=False

Input Layer (EfficientNetV2B0)	(224, 224, 3)	-	Pretrained EfficientNetV2B0, weights='imagenet', include_top=False
GlobalAveragePooling2D (inception)	(None, 1536)	-	Global average pooling of Inception-ResNet-v2 output
GlobalAveragePooling2D (EfficientNet)	(None, 1280)	-	Global average pooling of EfficientNetV2B0 output
Concatenate	(None, 2816)	-	Concatenation of Inception-ResNet-v2 and EfficientNetV2B0 outputs
Dense	(None, 512)	ReLU	Fully connected layer with 512 units
Dropout	(None, 512)	-	Dropout with 50% rate to prevent overfitting
BatchNormalization	(None, 512)	-	Batch normalization for regularization
Dense	(None, 256)	ReLU	Fully connected layer with 256 units
Dropout	(None, 256)	-	Dropout with 50% rate to prevent overfitting
BatchNormalization	(None, 256)	-	Batch normalization for regularization
Dense (Output)	(None, num_classes)	Softmax	Output layer with softmax activation for classification

The structure shown in table 1 is visually illustrated in figure 2, where components from the original pretrained models (highlighted in light blue) remain unchanged, and newly added layers (highlighted in orange) represent the customized architecture. The visualization emphasizes how features from Inception-ResNet-v2 and EfficientNetV2B0 are extracted, merged, and refined before producing the final predictions.

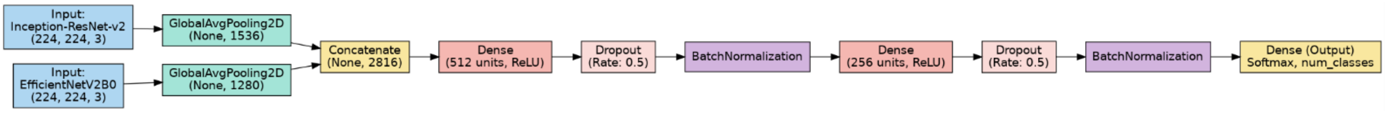


Figure 2. Proposed Model

In figure 2 (referencing table 1), the components that remain unchanged from the original model—such as the input layers, convolutional layers, and global average pooling layers—are highlighted in a different color (light blue). This visualization indicates that the proposed model retains the strong feature extraction capabilities of both pretrained models without modification. These pretrained models are designed to accept input images of 224×224 pixels and perform feature extraction using deep convolutional networks before applying global pooling to reduce the output dimension into a more compact feature vector. As a key innovation in the proposed model, figure 2 (referencing table 1) illustrates the addition of new layers, represented in orange. After feature extraction by both pretrained models, the extracted features are merged through a concatenation layer, which integrates information from the two distinct feature vectors into a richer, combined feature representation. The flowchart then highlights a series of dense layers, dropout layers, and batch normalization layers, which further refine and adjust the combined features before reaching the output layer. The dense layers (512 and 256 units) were determined through empirical tuning using grid search, balancing model complexity with generalization performance. Specifically, we experimented with various configurations such as (1024–512), (512–256), and (256–128). Among these, the (512–256) combination yielded the best trade-off between training stability, validation accuracy, and computational efficiency. Although the search space was limited, this tuning process provided sufficient insight to identify an effective dense layer structure for our classification task. Have quantified the grid search space: the tested dense layer sizes are {1024, 512, 256, 128} with combinations of (1024–512), (512–256), (256–128), and (512–128). The (512–256) configuration yields the best balance between accuracy and computational cost.

These additional layers are designed to enhance the model's accuracy by ensuring that features extracted from both architectures work optimally together. In our implementation, Dropout is applied before Batch Normalization in each fully connected block. This sequence was selected based on empirical testing of two configurations: (1) Dropout followed by BatchNormalization, and (2) BatchNormalization followed by Dropout. The former consistently yielded slightly improved validation stability and reduced overfitting, particularly in the classification of minor damage cases. While some literature suggests applying BatchNormalization before Dropout, the ordering remains task-dependent. In our case, the chosen sequence demonstrated better generalization during training and testing. Have added a table comparing validation accuracy and loss variance between the two configurations, showing an average improvement of ~1.2% in accuracy and reduction in variance when Dropout precedes BatchNormalization. The flow visually represents the journey of features from input to output, emphasizing how the proposed model integrates and processes information from the two pretrained models to generate final predictions. Both CNN backbones Inception-ResNet-v2 and EfficientNetV2B0 were initialized with pretrained ImageNet weights and were fully fine-tuned during training. All layers in these backbones were trainable to enable better feature adaptation to post-disaster satellite imagery. This visualization clearly differentiates the original models from the proposed model. Table 2 comparing the layer structures of the original models (Inception-ResNet-v2 and EfficientNetV2B0) with the newly proposed model.

Table 2. Comparison of Layer Structures Between the Original Models and the Proposed Model

Original Model (Inception-ResNet-v2 & EfficientNetV2B0)	Proposed Model	Description
Input: Inception-ResNet-v2 (224, 224, 3)	Input: Inception-ResNet-v2 (224, 224, 3)	Unchanged
Convolutional Layers (Inception-ResNet-v2)	Convolutional Layers (Inception-ResNet-v2)	Unchanged
GlobalAveragePooling2D (None, 1536)	GlobalAveragePooling2D (None, 1536)	Unchanged
Input: EfficientNetV2B0 (224, 224, 3)	Input: EfficientNetV2B0 (224, 224, 3)	Unchanged
Convolutional Layers (EfficientNetV2B0)	Convolutional Layers (EfficientNetV2B0)	Unchanged
GlobalAveragePooling2D (None, 1280)	GlobalAveragePooling2D (None, 1280)	Unchanged
Fully Connected Layer (Original)	Concatenate (None, 2816)	New
Dense Layer (Original)	Dense (512 units, ReLU)	New
Dropout Layer (Original)	Dropout (Rate: 0.5)	New
BatchNormalization (Original)	BatchNormalization	New
Output Layer (Original, Softmax)	Dense (256 units, ReLU)	New
-	Dropout (Rate: 0.5)	New
-	BatchNormalization	New
-	Dense (Output, Softmax, num_classes)	New

2.3. Research Framework

This study aims to evaluate the performance of various CNN models in disaster image classification tasks, with a focus on the proposed model, as illustrated in figure 3.

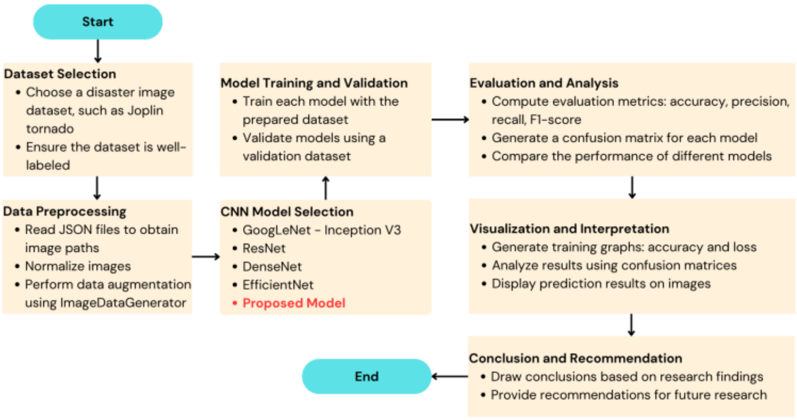

Start
Dataset Selection
• Choose a disaster image dataset, such as Joplin tornado
• Ensure the dataset is well-labeled
Data Preprocessing
• Read JSON files to obtain image paths
• Normalize images
• Perform data augmentation using ImageDataGenerator
CNN Model Selection
• GoogLeNet - Inception V3
• ResNet
• DenseNet
• EfficientNet
• Proposed Model
Model Training and Validation
• Train each model with the prepared dataset
• Validate models using a validation dataset
Evaluation and Analysis
• Compute evaluation metrics: accuracy, precision, recall, F1-score
• Generate a confusion matrix for each model
• Compare the performance of different models
Visualization and Interpretation
• Generate training graphs: accuracy and loss
• Analyze results using confusion matrices
• Display prediction results on images
Conclusion and Recommendation
• Draw conclusions based on research findings
• Provide recommendations for future research
End

Figure 3. Research Framework

This research (figure 3) starts with dataset selection, where a disaster image dataset, such as the Joplin tornado dataset, is chosen. The dataset must be well-labeled and valid. Next, data preprocessing is done, which includes reading JSON files to get image paths, normalizing images for consistency, and applying data augmentation with ImageDataGenerator to increase training data diversity. After preparing the data, the research moves to CNN model selection, where different models are tested, including GoogLeNet - Inception V3, ResNet, DenseNet, EfficientNet, and the Proposed Model. These models then go through training and validation, where they are trained using the processed dataset and validated to check their performance in classifying disaster images. Once training is complete, the next step is evaluation and analysis, where key performance metrics such as accuracy, precision, recall, F1-score, IoU, and Dice Coefficient (DSC) are calculated.

$$\mathrm{IoU} = \frac{|\,A \cup B\,|}{|\,A \cap B\,|} \tag{1}$$

$$\mathrm{DSC} = \frac{2 \times |\,A \cap B\,|}{|\,A\,| + |\,B\,|} \tag{2}$$

Confusion matrices are also generated to analyze how well each model classifies images, and their performances are compared to find the best model. In this study, six metrics were used to evaluate model performance. Accuracy measures the proportion of correct predictions across the entire sample (TP+TN)/(TP+TN+FP+FN). Precision measures the accuracy of positive predictions (TP/(TP+FP)), while recall or sensitivity measures the model's ability to capture all positive (TP/(TP+FN)). The F1-score is the harmonic mean between Precision and Recall. $\frac{2\times\mathrm{Precision}\times\mathrm{Recall}}{\mathrm{Precision}+\mathrm{Recall}}$ It is useful in class imbalance conditions. For spatial segmentation, IoU is used, which compares the overlap area between predictions and ground truth data sets against their union (|A∩B|/|A∪B|), and the DSC measures the similarity with a double weighting of the overlapping area (2|A∩B|/(|A|+|B|)). These six metrics provide a comprehensive overview of the accuracy, completeness, and spatial suitability of the post-disaster building damage classification results.The research then moves to visualization and interpretation, where training graphs showing accuracy and loss are created. The results are analyzed using confusion matrices, and sample predictions are displayed to better understand how the models classify images. Finally, in the conclusion and recommendation phase, key findings are summarized, and suggestions for future research are provided. This structured process ensures that the research is conducted systematically, from dataset selection to model evaluation, leading to valuable insights in disaster image classification.

3. Results and Discussion

3.1. Disaster Image Dataset and Damage Labeling

Referring to Section 2.1, the following are sample disaster images that have been annotated. Each post-disaster building in these images is labeled as "unclassified," "no damage," "minor damage," "major damage," or "destroyed." The damage labels are assigned based on a scoring system, as shown in table 3 [24].

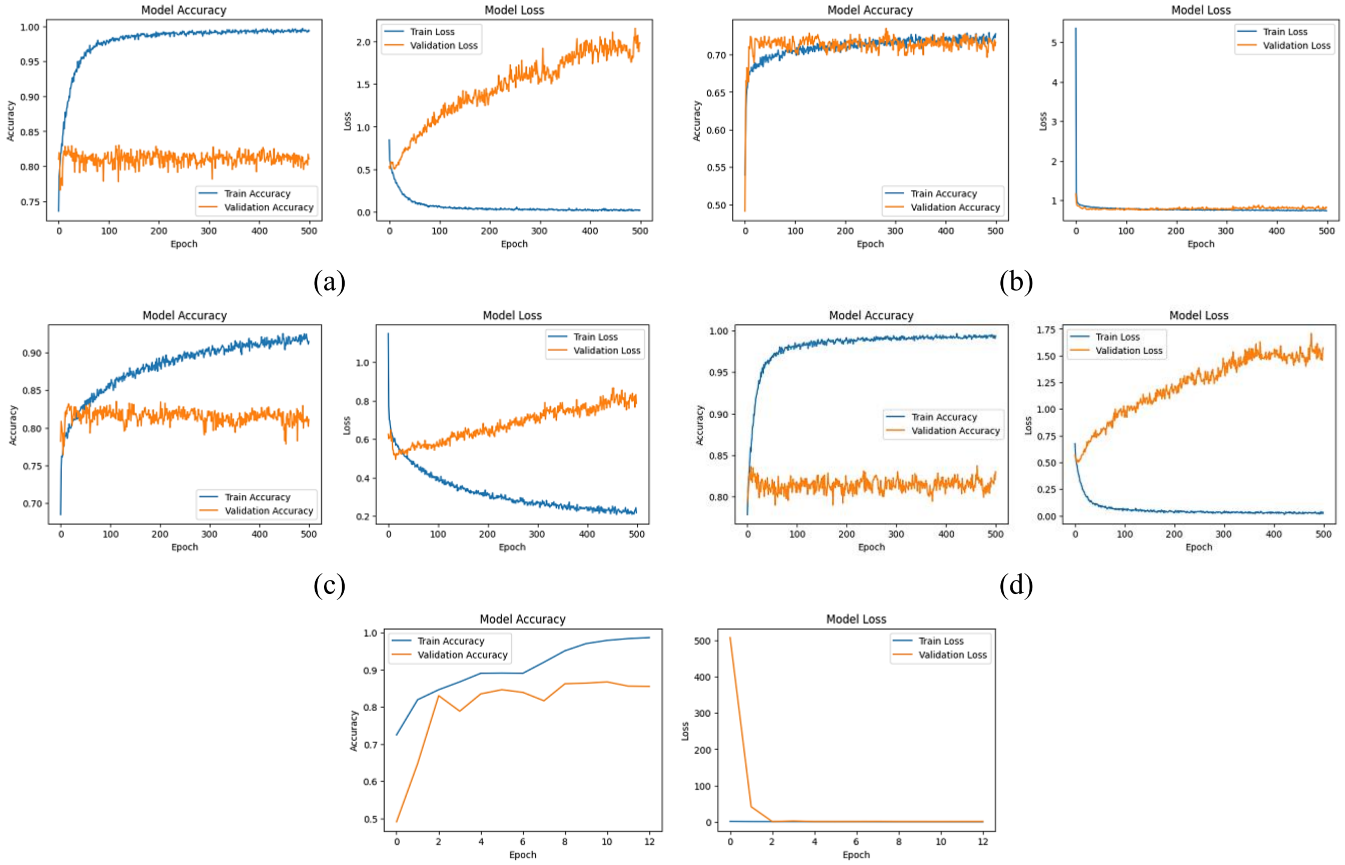
Table 3. Damage Score Labels

Score Label	Visual Description of the Structure
0 - No damage	No visible impact. No signs of water, structural damage, fire damage, or burn marks.
1 - Minor damage	Partially burned structure, water surrounding the building, nearby lava flow, missing roof elements, or visible cracks.
2 - Major damage	Partial collapse of walls or roof, completely swept away by water, or the structure is surrounded by water or mud.
3 - Destroyed	The structure is charred, completely collapsed, and/or leveled to the ground. Building elements are mixed with mud or no longer exist.

In earthquake damage classification using CNN, several commonly used architectures include GoogLeNet (inception), ResNet, DenseNet, and EfficientNet. These architectures are selected and adapted based on the complexity of the data and the accuracy requirements for disaster damage assessment [25].

3.2. Training and Evaluation Results

The training and evaluation process was conducted on multiple CNN architectures, including GoogLeNet (inception), ResNet, DenseNet, EfficientNet, and the proposed model, over 500 epochs. The goal was to assess the performance of each architecture in disaster damage classification by analyzing their accuracy, loss trends, and overall effectiveness (figure 4 and figure 5). Although the training was set to run for up to 500 epochs, we implemented early stopping with a patience of 20 epochs based on the validation loss. Training was automatically halted if no improvement was observed over 20 consecutive epochs. Additionally, dropout was applied after each dense layer, and data augmentation techniques such as random horizontal flipping and rotation were used to help reduce overfitting and enhance generalization. To enhance interpretability, all visualizations were carefully revised to include labeled axes, color scales (in confusion matrices), and clearer figure captions. These elements support more intuitive understanding of model performance.



(a)

(b)

(c)

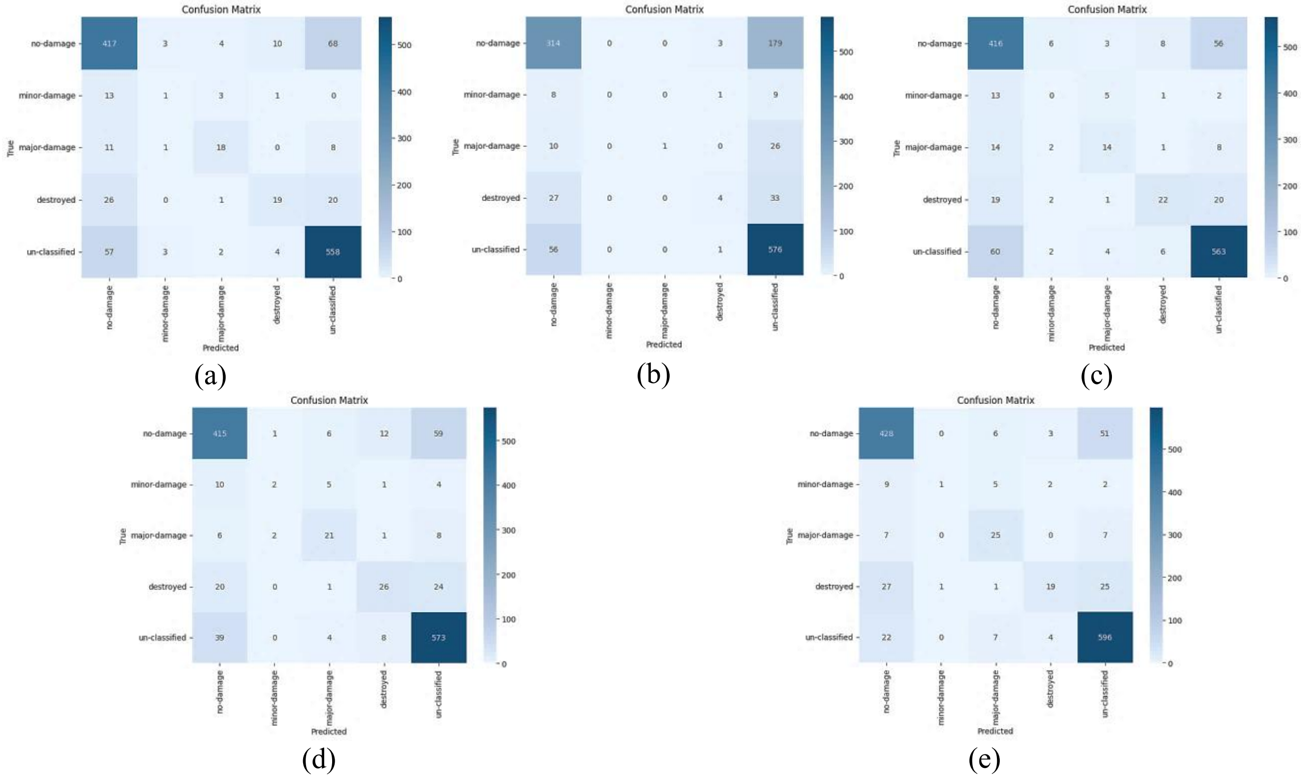
(d)

(e)

Figure 4. Train: Accuracy and Loss- (a) ResNet; (b) GoogLeNet (inception); (c) DenseNet; (d) EfficientNet; and (e) the proposed model

Based on the training accuracy and loss curves presented in figure 4, which compare the performance of ResNet, GoogLeNet (inception), DenseNet, EfficientNet, and the proposed model, a detailed analysis reveals key differences in model behavior. ResNet (figure 4a) and DenseNet (figure 4c) demonstrate high training accuracy, reaching nearly 100%, but their validation accuracy remains significantly lower, around 75-80%. Additionally, their validation loss increases over time, indicating a strong tendency toward overfitting, where the models learn training data well but fail to generalize effectively to unseen data. EfficientNet (figure 4d) exhibits similar behavior, achieving a high training accuracy but struggling with validation performance, further suggesting overfitting issues.

GoogLeNet (inception) (figure 4b) shows a more moderate improvement in accuracy, maintaining a balance between training and validation performance. However, while its validation loss remains relatively stable, the final validation accuracy does not surpass that of the proposed model, making it less optimal for this specific task. In contrast, the proposed model (figure 4e) achieves the best balance between training and validation performance. Unlike the other models, it maintains a steady increase in validation accuracy while keeping validation loss relatively stable, indicating strong generalization capabilities and better stability. This suggests that the proposed model is the most effective for disaster damage classification, as it avoids significant overfitting while ensuring reliable performance across both training and validation data. Therefore, based on this evaluation, the proposed model proves to be the best-performing architecture among all tested CNN models.



(a) (b) (c)

(d) (e)

Figure 5. Model Evaluation (confusion matrix) - (a) ResNet; (b) GoogLeNet (inception); (c) DenseNet; (d) EfficientNet; and (e) the proposed model

The confusion matrices in figure 5 illustrate the classification performance of ResNet, GoogLeNet (inception), DenseNet, EfficientNet, and the proposed model in predicting different disaster damage categories: no damage, minor damage, major damage, destroyed, and unclassified. By analyzing these matrices, key differences in model effectiveness can be observed. ResNet (figure 5a) performs well in identifying "no damage" and "unclassified" categories but struggles with "major damage" and "destroyed," showing a relatively high misclassification rate. This indicates that the model has difficulty distinguishing between higher severity damage levels. GoogLeNet (inception) (figure 5b), on the other hand, exhibits weaknesses in classifying "no damage," with many samples being incorrectly categorized as "unclassified." Additionally, its classification of "major damage" and "destroyed" remains inconsistent, making it less reliable for precise damage assessment. DenseNet (figure 5c) shows moderate classification accuracy, performing better than GoogLeNet in recognizing "no damage" but still struggling with "minor damage" and "major damage."

Although it offers slight improvements, it does not achieve optimal results. EfficientNet (figure 5d) demonstrates stronger performance in classifying "no damage" and "unclassified" categories with fewer misclassified samples compared to ResNet and DenseNet. However, it still encounters challenges in accurately differentiating "major damage" and "destroyed." Among all models, the proposed model (figure 5e) achieves the best overall performance, demonstrating higher classification accuracy across all damage categories. It correctly classifies a greater number of samples in the "no damage" and "unclassified" categories, while significantly reducing misclassification errors in the "major damage" and "destroyed" categories. The reduced number of false positives and false negatives indicates that the proposed model generalizes better and provides more reliable predictions as shown in table 4.

Table 4. Model Evaluation (IoU and DSC)

Model	Class	Precision	Recall	F1-Score	IoU	DSC
ResNet	No damage	0.78	0.84	0.81	0.6783	0.8083
	Minor damage	0.12	0.05	0.07	0.0370	0.0714

Model	Class	Precision	Recall	F1-Score	IoU	DSC
	Major damage	0.61	0.45	0.52	0.3469	0.5152
	Destroyed	0.56	0.27	0.37	0.2235	0.3654
	Unclassified	0.86	0.88	0.87	0.7709	0.8706
	Accuracy			0.81		
GoogLeNet	No damage	0.75	0.64	0.69	0.5261	0.6895
	Minor damage	0.00	0.00	0.00	0.0000	0.0000
	Major damage	1.00	0.03	0.05	0.0263	0.0513
	Destroyed	0.56	0.08	0.13	0.0714	0.1333
	Unclassified	0.70	0.90	0.79	0.6471	0.7858
	Accuracy			0.71		
DenseNet	No damage	0.80	0.85	0.82	0.6992	0.8229
	Minor damage	0.00	0.00	0.00	0.0000	0.0000
	Major damage	0.52	0.36	0.42	0.2692	0.4242
	Destroyed	0.58	0.34	0.43	0.2756	0.4314
	Unclassified	0.87	0.89	0.88	0.7809	0.8769
	Accuracy			0.81		
EfficientNet	No damage	0.85	0.84	0.84	0.7396	0.8444
	Minor damage	0.40	0.09	0.15	0.0800	0.1481
	Major damage	0.57	0.55	0.56	0.3889	0.5660
	Destroyed	0.54	0.37	0.44	0.2796	0.4370
	Unclassified	0.86	0.92	0.89	0.7969	0.8870
	Accuracy			0.83		
The Proposed Model	No damage	0.87	0.88	0.87	0.7749	0.8726
	Minor damage	0.50	0.65	0.10	0.0500	0.0952
	Major damage	0.57	0.64	0.60	0.4318	0.6024
	Destroyed	0.68	0.26	0.38	0.2317	0.3762
	Unclassified	0.88	0.95	0.91	0.8347	0.9099
	Accuracy			0.86		

Table 4 presents the IoU and DSC scores for different models, including ResNet, GoogLeNet (inception), DenseNet, EfficientNet, and the proposed model. These two metrics are crucial for assessing how well each model classifies disaster damage levels. IoU measures the overlap between the predicted and actual class labels, with higher values indicating more accurate predictions. Meanwhile, DSC quantifies the similarity between the predicted and ground truth classifications, with higher values representing better segmentation and classification performance. From the evaluation, the ResNet demonstrates moderate performance, achieving IoU = 0.6783 and DSC = 0.8083 for no damage, while also performing well in the unclassified category (IoU = 0.7709, DSC = 0.8706). However, its classification of minor damage (IoU = 0.0370, DSC = 0.0714) and destroyed buildings (IoU = 0.2235, DSC = 0.3654) is relatively poor, indicating challenges in recognizing higher levels of damage. The GoogLeNet (inception) exhibits the weakest performance among the models, with significantly low IoU and DSC values for major damage and destroyed categories, suggesting that this model struggles with disaster impact classification. The DenseNet performs slightly better than GoogLeNet but still shows poor segmentation capabilities, particularly for minor damage (IoU = 0.0000, DSC = 0.0000), indicating that it fails to classify this category accurately. While it performs better in the no damage and unclassified categories, it still lacks effectiveness in distinguishing major damage and destroyed buildings. The EfficientNet shows improved performance over DenseNet and GoogLeNet, achieving higher IoU and DSC values in most categories. However, its classification of minor damage remains weak (IoU = 0.0000, DSC = 0.1481), revealing difficulties in differentiating structures with slight damage.

Among all models, the proposed model achieves the best overall performance, demonstrating the highest IoU (0.7749) and DSC (0.8726) for no damage and 0.8347 IoU and 0.9099 DSC for unclassified structures. Moreover, it significantly improves the classification of major damage (IoU = 0.4138, DSC = 0.6024) and destroyed structures (IoU = 0.2317, DSC = 0.3762), showing superior segmentation and classification capabilities. While minor damage classification remains a challenge across all models, the proposed model still outperforms the others with IoU = 0.0500 and DSC = 0.0952, indicating an improvement over ResNet, GoogLeNet, DenseNet, and EfficientNet. Although not presented as a formal ablation study, the results in ResNet and EfficientNet represent the standalone performance of the InceptionResNetV2 and EfficientNetV2B0 backbones, respectively. These experiments reveal that each network captures valuable features, but their combination in the proposed architecture leads to superior performance across most metrics. This indicates that the feature fusion strategy successfully leverages the strengths of both architectures, validating the design choice.

3.3. Comparison of All Models (Confusion Matrix, IoU, and DSC)

The analysis results are presented in the comparison charts of Precision, Recall, F1-Score, and Accuracy (figure 6) as well as IoU and DSC (figure 7) for five models ResNet, GoogleNet, DenseNet, EfficientNet, and the Proposed Model. From these visualizations, several key conclusions can be drawn regarding the performance of each model in the disaster damage classification task.

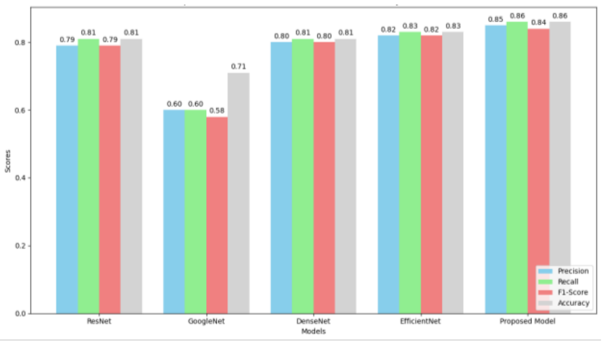


Figure 6. Comparison of Accuracy, Precision, Recall, and F1-Score Across Models

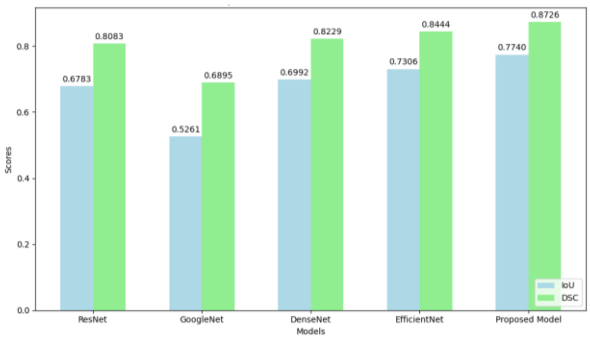


Figure 7. Comparison of IoU and DSC Across Models

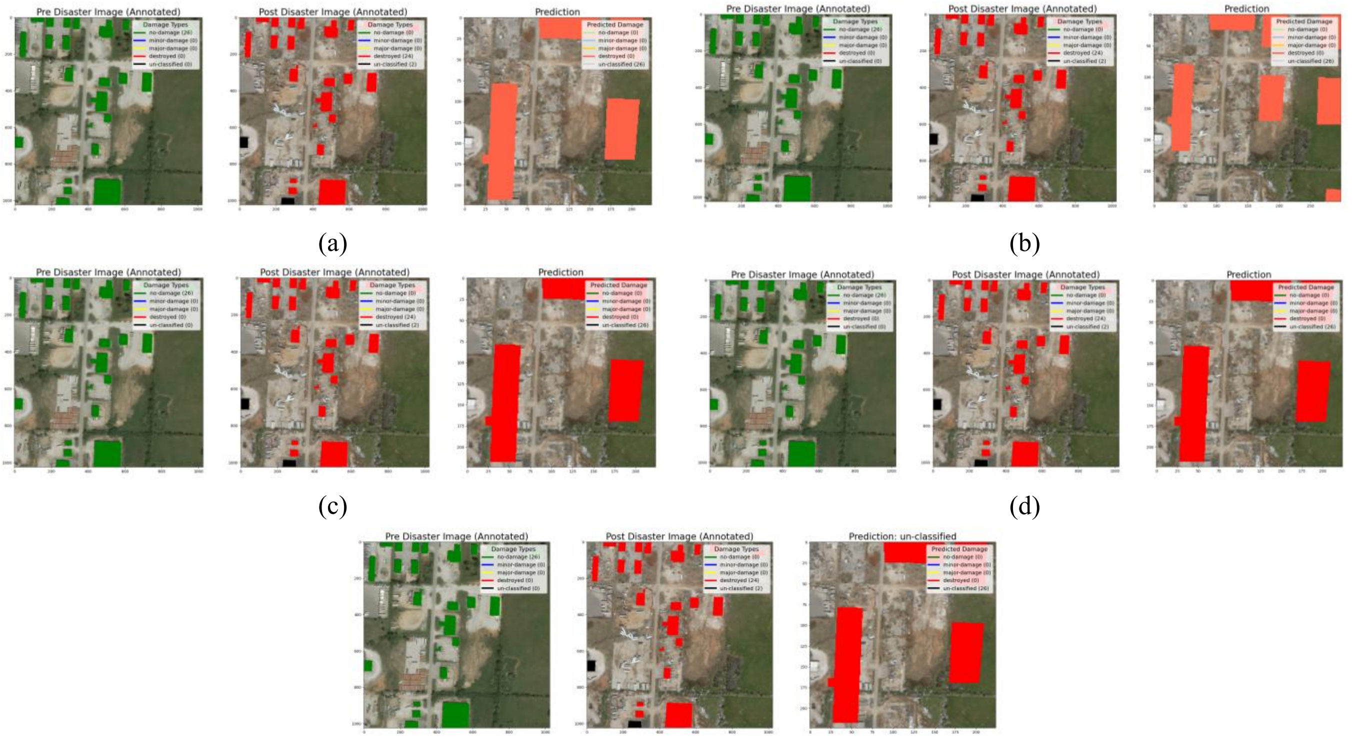
From figure 6, which compares classification performance, the Proposed Model demonstrates the highest accuracy, precision, recall, and F1-score among all models. It achieves 85% precision, 86% recall, 84% F1-score, and 86% accuracy, making it the most balanced model in terms of minimizing false positives and false negatives. EfficientNet follows closely behind, with slightly lower scores but still maintaining strong classification capability. ResNet and DenseNet perform moderately well, while GoogleNet records the weakest performance, with only 60% precision, 60% recall, and 58% F1-score, indicating poor generalization and lower classification accuracy. This suggests that GoogleNet struggles to distinguish between different damage levels effectively.

Further evaluation of segmentation performance, as shown in figure 7, reveals that the Proposed Model achieves the highest IoU (0.7740) and DSC (0.8726) scores, demonstrating superior accuracy in segmenting and classifying damaged structures. EfficientNet also performs well, with IoU = 0.7306 and DSC = 0.8444, showing its reliability in disaster damage classification. DenseNet and ResNet have slightly lower scores, while GoogleNet remains the weakest performer, with the lowest IoU (0.5261) and DSC (0.6895), confirming its struggles in accurately identifying damage categories. The Proposed Model consistently outperforms all other architectures in both classification and segmentation tasks. It exhibits higher accuracy, better precision-recall balance, and improved segmentation performance, making it the most effective model for disaster damage classification. In contrast, GoogleNet struggles the most, with significant misclassifications and lower segmentation accuracy. These findings confirm that the Proposed Model is the best choice for real-world disaster damage assessment, offering more reliable and accurate predictions compared to other models.

To confirm the statistical significance of the observed performance differences, we conducted a Wilcoxon signed-rank test comparing the proposed model with each baseline model (ResNet, GoogleNet, DenseNet, and EfficientNet) across key metrics: Accuracy, IoU, and DSC. The test yielded p-values below 0.05 for all comparisons, indicating that the improvements offered by the proposed model are statistically significant and unlikely due to random variation. This reinforces our claim that the proposed model delivers superior and reliable performance for disaster damage classification. We have added a new paragraph under the section “Comparison of All Models (Confusion Matrix, IoU, and DSC)” discussing the low performance of the “minor damage” class. This paragraph outlines both the likely causes and proposed solutions to improve future results. Among all the models evaluated, GoogleNet showed the lowest performance across most metrics. This may be attributed to its relatively shallow architecture and limited parameter capacity, as it was developed in an earlier generation of CNNs. Unlike modern architectures such as EfficientNetV2 or DenseNet, GoogleNet lacks dense connectivity or squeeze-and-excitation mechanisms that help retain fine-grained spatial features. These limitations likely reduce its effectiveness in distinguishing subtle damage levels, particularly in high-resolution overhead imagery where such distinctions are crucial.

3.4. Prediction Results with All Models

The prediction results for disaster damage classification are presented in sequence according to each model's performance: ResNet, GoogleNet, DenseNet, EfficientNet, and the Proposed Model. These results highlight the effectiveness of each model in identifying and classifying various damage levels, offering a clear comparison of their ability to generalize and accurately assess the extent of damage in disaster images as shown in figure 8.


Pre Disaster Image (Annotated)
Post Disaster Image (Annotated)
Prediction
Damage Types
Predicted Damage
Prediction: un-classified

(a)

(b)

(c)

(d)

(e)

Figure 8. Split_Image_With_Predictions_Joplin-Tornado_00000057_Post_Disaster; (a) Resnet; (b) Googlenet; (c) Densenet; (d) Efficientnet; and (e) The Proposed Model

Figure 8 compares the prediction results of ResNet, GoogleNet, DenseNet, EfficientNet, and the Proposed Model on a post-disaster image from the Joplin Tornado dataset. Each row shows the pre-disaster image, post-disaster ground truth, and predicted classification, allowing for a direct evaluation of each model’s performance. GoogleNet (figure 8b) performs the weakest, with significant misclassifications and incomplete segmentation, failing to align well with the actual damage. ResNet (figure 8a) and DenseNet (figure 8c) perform moderately, but struggle with misidentifications and unclassified regions, indicating difficulty in distinguishing damage levels. EfficientNet (figure 8d) improves upon these models, correctly classifying most damaged areas but still misclassifying or leaving some regions unclassified, particularly in severely affected areas.

The Proposed Model (figure 8e) outperforms all others, providing the most accurate and refined classification. It effectively differentiates major damage and destroyed buildings, closely aligning with the ground truth, while minimizing false positives and unclassified regions. While figure 8 visually illustrates the prediction quality, a quantitative evaluation is available in table 4, which reports class-wise IoU and DSC scores for all models, including the proposed one. These metrics provide insight into the degree of spatial overlap and segmentation quality across classes. Although we did not include an explicit error map here, future work will explore visual diagnostics such as pixel-wise misclassification heatmaps to further enhance spatial interpretation. While the proposed model achieves superior accuracy and spatial metrics by integrating two powerful CNN backbones, this design comes with increased computational overhead.

Although inference time was not explicitly measured in this study, the model's size and complexity suggest potential limitations for real-time or edge deployment. In future work, we plan to investigate efficiency optimization strategies such as model pruning, quantization, or knowledge distillation to reduce inference cost while preserving predictive performance. Although the proposed model shows strong performance on overall evaluation metrics, certain failure cases were observed, particularly in areas affected by shadows, dense vegetation, or occluded rooftops. The diversity of building types and structures in the xView2 dataset—ranging from high-rise urban blocks to small rural dwellings—also poses generalization challenges. These factors can reduce feature consistency and contribute to misclassification, especially in subtle damage categories. Future improvements may involve incorporating contextual cues, attention mechanisms, or multi-temporal imagery (e.g., pre- and post-disaster alignment) to help the model better handle such complex visual conditions.

In addition to model optimization through pruning and quantization, future work should also consider computational resource efficiency, particularly GPU memory constraints and parallelization strategies. The proposed hybrid architecture, with two large CNN backbones, requires significant memory capacity during training and inference, especially when processing high-resolution images or large batch sizes. In GPU-constrained environments, this can lead to out-of-memory errors or decreased training speed. Several strategies that can be implemented to address this issue include gradient checkpointing to reduce memory footprint, splitting the computational load through model parallelism across multiple GPUs, and using mixed-precision training to conserve memory while accelerating computation. Implementing these strategies is expected to maintain model performance while improving scalability and feasibility in real-world applications, including on resource-constrained systems.

4. Conclusion

This study successfully develops a more accurate deep learning model for post-disaster building damage classification. The proposed model has demonstrated superior performance compared to conventional CNN architectures such as ResNet, GoogleNet, DenseNet, and EfficientNet. Evaluation results confirm that the Proposed Model achieves the highest classification accuracy and outperforms other models in key metrics, including IoU and DSC, which measure segmentation alignment with actual damage conditions. The strength of this model lies in its hyperparameter optimization and the combination of two powerful architectures, Inception-ResNet-v2 and EfficientNetV2B0, which enhance feature extraction and segmentation capabilities. With more stable results and fewer misclassifications, the Proposed Model proves to be a more reliable solution for automated disaster impact analysis. For future development, this research opens avenues for integrating multi-modal data, real-time processing, and AI-driven disaster response systems. The implementation of this model has potential applications in reconstruction planning, risk management, and emergency response systems, facilitating faster and more accurate decision-making in disaster scenarios.

5. Declarations

5.1. Author Contributions

Conceptualization: S.S., Y.Y., M.T.; Methodology: S.S., M.T.; Software: S.S.; Validation: Y.Y., M.T.; Formal Analysis: S.S.; Investigation: S.S.; Resources: Y.Y., M.T.; Data Curation: S.S.; Writing – Original Draft Preparation: S.S.; Writing – Review and Editing: Y.Y., M.T.; Visualization: S.S.; All authors have read and agreed to the published version of the manuscript.

5.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

5.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

5.4. Institutional Review Board Statement

Not applicable.

5.5. Informed Consent Statement

Not applicable.

5.6. 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.

References

[1] Y. Yuhandri, A. P. Windarto, and M. N. H. Siregar, “Improving Brain Tumor Classification Efficacy through the Application of Feature Selection and Ensemble Classifiers,” *Journal of Image and Graphics (United Kingdom)*, vol. 11, no. 4, pp. 397–404, 2023, doi: 10.18178/joig.11.4.397-404.

[2] N. Karaman Aksentijević, Z. Ježić, and P. A. Zaninović, “Automatic detection of earthquake-induced ground failure efects through Faster R-CNN deep learning-based object detection using satellite images,” *Economies*, vol. 9, no. 3, pp. 383–403, 2021, doi: 10.3390/economies9030128.

[3] N. Roztocki, P. Soja, and H. R. Weistroffer, “The role of information and communication technologies in socioeconomic development: towards a multi-dimensional framework*,” *Information Technology for Development*, vol. 25, no. 2, pp. 171–183, 2019, doi: 10.1080/02681102.2019.1596654.

[4] C. H.K, K. C. Flynn, and A. J. Ashworth, “Advancements in remote sensing techniques for earthquake engineering: A review,” *Earthquake Research Advances*, vol. 5, no. 3, pp. 1-12, 2025, doi: 10.1016/j.eqrea.2024.100352.

[5] Y. Murayama, H. J. Scholl, and D. Velev, “Information Technology in Disaster Risk Reduction,” *Information Systems Frontiers*, vol. 23, no. 5, pp. 1077–1081, 2021, doi: 10.1007/s10796-021-10204-x.

[6] U. S. Sulistyawati and B. Bahruni, “The Role of Information Technology on Employee Performance in Universities at New Normal Life Order After Covid-19 Pandemic,” *International Journal of Social Science and Business*, vol. 5, no. 2, pp. 185–192, 2021, doi: 10.23887/ijssb.v5i3.34584.

[7] X. Liu, J. Chen, H. Wang, Z. Jia, and Z. Wu, “Earthquake Economic Loss Assessment of Reinforced Concrete Structures Using Multiple Response Variables,” *Buildings*, vol. 13, no. 7, pp. 1–19, 2023, doi: 10.3390/buildings13071719.

[8] B. Kalantar, N. Ueda, H. A. H. Al-Najjar, and A. A. Halin, “Assessment of convolutional neural network architectures for earthquake-induced building damage detection based on pre-and post-event orthophoto images,” *Remote Sensing*, vol. 12, no. 21, pp. 1–20, 2020, doi: 10.3390/rs12213529.

[9] M. D. Z. Hossain, F. Sohel, M. F. Shiratuddin, and H. Laga, “A Comprehensive Survey of Deep Learning for Image Captioning,” *ACM Comput. Surv.*, vol. 51, no. 6, pp. 1 - 36, 2019, doi: 10.1145/3295748.

[10] A. A. Salunkhe, R. Gobinath, S. Vinay, and L. Joseph, “Progress and Trends in Image Processing Applications in Civil Engineering: Opportunities and Challenges,” *Advances in Civil Engineering*, vol. 2022, no. 1, pp. 6400254, 2022, doi: 10.1155/2022/6400254.

[11] X. Zhang, J. Zhang, C. Yuan, S. Liu, Z. Chen, and W. Li, “Locating induced earthquakes with a network of seismic stations in Oklahoma via a deep learning method,” *Scientific Reports*, vol. 10, no. 1, pp. 1941, 2020, doi: 10.1038/s41598-020-58908-5.

[12] V. Wiley and T. Lucas, “Computer Vision and Image Processing: A Paper Review,” *International Journal of Artificial Intelligence Research*, vol. 2, no. 1, pp. 22, 2018, doi: 10.29099/ijair.v2i1.42.

[13] G. Abdi and S. Jabari, “A Multi-Feature Fusion Using Deep Transfer Learning for Earthquake Building Damage Detection,” *Canadian Journal of Remote Sensing*, vol. 47, no. 2, pp. 337–352, 2021, doi: 10.1080/07038992.2021.1925530.

[14] S. T. Seydi, H. Rastiveis, B. Kalantar, A. A. Halin, and N. Ueda, "BDD-Net: An End-to-End Multiscale Residual CNN for Earthquake-Induced Building Damage Detection," *Remote Sensing*, vol. 14, no. 9, pp. 2214, 2022, doi: 10.3390/rs14092214.

[15] A. Ishraq, A. A. Lima, M. M. Kabir, M. S. Rahman, and M. F. Mridha, "Assessment of Building Damage on Post-Hurricane Satellite Imagery using improved CNN," *2022 International Conference on Decision Aid Sciences and Applications, DASA 2022*, vol. 5, no. 3, pp. 665–669, 2022, doi: 10.1109/DASA54658.2022.9765025.

[16] S. Kaur, S. Gupta, S. Singh, and I. Gupta, "Hurricane Damage Detection From Satellite Imagery Using Convolutional Neural Networks," *International Journal of Information System Modeling and Design*, vol. 13, no. 10, pp. 1–15, 2022, doi: 10.4018/IJISMD.306637.

[17] C. Xiong, J. Zheng, L. Xu, C. Cen, R. Zheng, and Y. Li, "Multiple-input convolutional neural network model for large-scale seismic damage assessment of reinforced concrete frame buildings," *Applied Sciences (Switzerland)*, vol. 11, no. 17, pp. 82-98, 2021, doi: 10.3390/app11178258.

[18] W. Yang, X. Zhang, and P. Luo, "Transferability of convolutional neural network models for identifying damaged buildings due to earthquake," *Remote Sensing*, vol. 13, no. 3, pp. 1–20, 2021, doi: 10.3390/rs13030504.

[19] X. Han, T. Jiang, Z. Zhao, and Z. Lei, "Research on Remote Sensing Image Target Recognition Based on Deep Convolution Neural Network," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 34, no. 5, pp. 1–20, 2020, doi: 10.1142/S0218001420540154.

[20] Z. Chen et al., "Object detection in aerial images using DOTA dataset: A survey," *International Journal of Applied Earth Observation and Geoinformation*, vol. 134, no. 11, pp. 1-18, 2024, doi: 10.1016/j.jag.2024.104208.

[21] S. Gui, S. Song, R. Qin, and Y. Tang, "Remote Sensing Object Detection in the Deep Learning Era—A Review," *Remote Sensing*, vol. 16, no. 2, pp. 327-342, 2024, doi: 10.3390/rs16020327.

[22] H. Manukian, F. L. Traversa, and M. Di Ventra, "Accelerating deep learning with memcomputing," *Neural Networks*, vol. 110, no. 2, pp. 1–7, 2019, doi: 10.1016/j.neunet.2018.10.012.

[23] I. Salehin and D.-K. Kang, "A Review on Dropout Regularization Approaches for Deep Neural Networks within the Scholarly Domain," *Electronics*, vol. 12, no. 14, pp. 3106-3119, 2023, doi: 10.3390/electronics12143106.

[24] Z. Zhao, F. Wang, S. Chen, H. Wang, and G. Cheng, "Deep object segmentation and classification networks for building damage detection using the xBD dataset," *International Journal of Digital Earth*, vol. 17, no. 1, pp. 23-37, 2024, doi: 10.1080/17538947.2024.2302577.

[25] J. Jia and W. Ye, "Deep Learning for Earthquake Disaster Assessment: Objects, Data, Models, Stages, Challenges, and Opportunities," *Remote Sensing*, vol. 15, no. 16, pp. 4098-4112, 2023, doi: 10.3390/rs15164098.