J-INTECH (Journal of Information and Technology)
Accredited Sinta 4 Ministry of Higher Education, Science and Technology Republic of Indonesia SK No. 10/C/C3/DT.05.00/2025
E-ISSN: 2580-720X || P-ISSN: 2303-1425

Post-Disaster Building Damage Segmentation Using Convolutional Neural Networks

Revaldi Rahmatmulya[1], Agung Teguh Wibowo Almais[2], Mokhamad Amin Hariyadi[3*]

[1,2,3] *Universitas Islam Negeri Maulana Malik Ibrahim Malang, Faculty of Science and Technology, Master of Informatics, Jl. Gajayana No.50, Dinoyo, Kec. Lowokwaru, Kota Malang, Jawa Timur 65144, Indonesia*

Keywords

Convolutional Neural Network; Machine Learning; Segmentation

Corresponding Author:
adyt2002@uin-malang.ac.id

Abstract

Natural disasters are events caused by nature such as earthquakes, tornadoes, tsunamis, forest fires, and others. The impacts of natural disasters are significant and varied across various sectors, including the economy, health, and primarily, infrastructure. Effective and efficient actions are needed to assist in the recovery following natural disasters, one of which is aiding in the identification of building damage levels post-disaster. To address this issue, this research proposes a system capable of performing segmentation to determine the level of building damage post-natural disaster using convolutional neural network methods. The data utilized consists of aerial images sourced from xView2: Assess Building Damage, comprising 50 aerial images with 5 classes: no-damage, minor-damage, major-damage, destroyed, and unlabeled. The steps undertaken in this research include data preprocessing using patchify and data augmentation. Subsequently, feature extraction is performed using convolution, followed by the training process using a neural network with the proposed architecture. This study proposes an architecture with 27 hidden layers, with feature extraction utilizing average pooling. The model evaluation process will employ Mean Intersection over Union (MIoU) to assess how closely the segmentation prediction results resemble the original data. The proposed architecture demonstrates the best MIoU result with a value of 0.31 and an accuracy of 0.9577.

1. Introduction

Natural disasters are events of a geological nature, including, but not limited to, earthquakes, tsunamis, floods, forest fires, storms, droughts and heat waves. These events are part of the riskscape with which humans have learnt to coexist. However, the impact of natural disasters has increased significantly in recent years [1].The impact of natural disasters is substantial and diverse, impacting various sectors, including the economic, health, building, and vegetation sectors [2]. Given the wide range of sectors susceptible to the impact of natural disasters, effective and efficient management strategies are paramount.

In this particular instance, one of the sectors that merits particular attention is the physical sector, with a particular focus on the management of damaged buildings in the aftermath of natural disasters. In the event of a building being damaged by a natural disaster, it is imperative to assess the extent of the damage caused by the natural disaster itself. The purpose of this procedure is to ascertain the extent of necessary repairs. The

extent of damage is contingent on the nature of the natural disaster and the geographical location of the building. In some cases, minor damage will only require minor repairs and minimal cost, but damage can also be severe and require significant expense to repair and even reconstruct [3].

The quantification of the damage to buildings in the aftermath of a natural disaster is frequently conducted manually, a method that is inherently not efficacious for the purpose of accurate assessment. There are several factors that cause manual determination to be ineffective. Firstly, it requires a significant amount of energy and time. It is imperative to acknowledge the necessity for a considerable number of individuals to be involved in this process, given the fact that the damage that occurs is not confined to a single building. This, in turn, results in an increase in the time required for the determination process to be completed. Consequently, this may impede the response of the rescue team [4]. The second factor pertains to the reliance on experts in determining the level of damage to buildings in the aftermath of natural disasters. The determination of the level of damage to be consistent is a task best suited to an expert, but the limitations of the expert can act as an obstacle in this process [5].

In order to surmount this limitation, there is a necessity for the development of deep learning models to automatically determine the level of damage to buildings after natural disasters [4]. The author employs a deep learning model with a convolutional neural network (CNN) method, utilising an architecture that has been designed by the author in a computational approach. The power of CNN lies in its ability to automatically learn and extract complex visual features, a capability proven effective across various domains. For example, research journal by Indriani et al. [6] demonstrated that a CNN model could achieve high accuracy in identifying unique and nuanced patterns in handwritten signatures, confirming the method's robustness for complex identification tasks. The utilisation of data, manifesting as images of edifices damaged by natural disasters, empowers the deep learning model with the capability to accurately and expeditiously ascertain the extent of damage post-natural disaster. It is hoped that the impacts that occur due to manual methods in determining the level of damage to buildings after natural disasters will not occur.

A study was conducted by Almas et al. [7] to ascertain the efficacy of the artificial neural network method in determining the level of damage to buildings in the aftermath of natural disasters. In this study, researchers employed text data exclusively, eschewing the use of image data. The experimental findings demonstrate that the E5 data pattern model exhibits an optimal accuracy rate of 97 per cent, accompanied by a Mean Squared Error (MSE) value of 0.06 and a Mean Absolute Percentage Error (MAPE) of 3 per cent. The author's hypothesis, formulated on the basis of previous research, is that this problem can be optimised with image data using the convolutional neural network method.

Subsequently, research on damage segmentation using the convolutional neural network method with Unet architecture has been carried out in the context of disease in rice plants affected by leafblast pests by Annafii et al. [8]. In this study, a model was developed to segment the data, with a total of 300 cases analysed. The model demonstrated an accuracy of 98.60%, with a loss of only 0.0526. This finding indicates that the employment of the U-Net model in the segmentation process, particularly in the context of damage segmentation, yields optimal outcomes.

Research by Kotaridis and Lazaridou. [9] also demonstrated the efficacy of U-Net in the segmentation of geographic elements, such as buildings and vegetation, in map images, achieving an accuracy of 0.97. Di Benedetto et al. [10] developed U-Net with ResNet50 encoder for road crack segmentation, resulting in mIoU of 0.6248 and F1-score of 0.7577. Gangurde. [11] also successfully enhanced the performance of building and road segmentation from UAV images with U-Net and EfficientNet encoder.

W. Li et al. [12] conducted a comparative analysis of U-Net and U-Net-CBAM in the segmentation of waterlogged areas from Sentinel-1A images. Their findings indicated that U-Net-CBAM exhibited a notable capacity to enhance the segmentation outcomes, particularly in scenarios involving smaller areas . As posited by Y. Li et al. [13], the CTMU-UNet model has been demonstrated to produce optimal results when applied to various datasets pertaining to the segmentation of aerial imagery.

Whilst the aforementioned studies establish the U-Net architecture as a powerful tool for segmentation, this research provides a critical analysis of its application in the unique context of post-disaster building damage by positioning itself against key related works. In contrast to the work of Annafii et al. [8], which focused on segmenting relatively uniform objects like pests on rice leaves, this study tackles the far more complex domain of building damage imagery, characterised by high variability in shape, texture, and lighting. Furthermore, this study utilises a standard U-Net architecture that has been built from the ground up, a departure from the approach employed by Di Benedetto et al. [10]. The latter effectively applied a U-Net with a pre-trained ResNet50 encoder (transfer learning) for the detection of linear road cracks. This methodological distinction—building from scratch versus using transfer learning—likely contributes to the lower mIoU score obtained in this study (0.31 vs. 0.62) and underscores the significant advantage of leveraging pre-trained models. Gangurde lends further support to this perspective. [11], whose work represents an evolutionary advancement by achieving state-of-the-art results with a U-Net and a powerful EfficientNet encoder. Whilst the present study corroborates the hypothesis that U-Net is feasible on fundamental level, Gangurde's findings demonstrate that performance can be significantly enhanced through the utilisation of contemporary encoders. Consequently, this research serves as a crucial baseline that highlights the definitive superiority of the transfer learning approach for building segmentation from aerial imagery.

The selection of the Convolutional Neural Network (CNN) method in this study is predicated on the basis of the problem and previous research results, with the CNN method having previously demonstrated success in various image segmentation tasks. The study by Almais et al. [7] demonstrated the efficacy of artificial neural networks in assessing the damage to buildings following a disaster using text data. However, the approach does not incorporate visual information from images. This development presents a valuable opportunity to enhance the analysis process through the utilisation of image data by employing a CNN approach, a methodology that has been demonstrated to be highly effective in the domain of visual data processing. Research by Annafii et al. [8] demonstrated that a CNN with a U-Net architecture is capable of segmenting crop damage with a high degree of accuracy. In addition, a plethora of other studies lend further support to the efficacy of U-Net and its derivatives in segmentation tasks, including, but not limited to, geographic images as demonstrated by Kotaridis and Lazaridou. [9], road cracks as outlined by Di Benedetto et al. [10], and building and road segmentation from UAV images as investigated by Gangurde. [11]. Research by W. Li et al. [12] also proved the improved performance of waterlogged area segmentation by adding a CBAM module to the U-Net architecture, while Y. Li et al. [13] proposed a CTMU-UNet that produces superior performance on various aerial image datasets. Drawing upon the findings of these studies, this research employs a convolutional neural network (CNN) with a U-Net architecture to segment building damage areas from images. The objective is to evaluate the efficacy of this approach in automatically and efficiently mapping damage following natural disasters.

2. Research Method

The research method will address the stages of research. The research process is delineated by the following stages: data collection, data preprocessing, architecture model building, model training, and model evaluation. The data that the author utilised for this research is satellite image data of an area that has experienced a natural disaster. The dataset under scrutiny was procured by the author through a public dataset provider website, xView2: Assess Building Damage. In this research, the author will divide the data into a 70:30 composition. This configuration is indicative of a 70% allocation of data designated for training purposes, with the remaining 30% allocated for testing. The quantity of data to be utilised in the training process is set to be 50 high-resolution images, with a resolution of 1024 x 1024 pixels. To ensure a degree of diversity in the source data, these 50 images were selected to represent a variety of disaster scenarios, including tornadoes, tsunamis, wildfires, floods, and windstorms.

The data set under consideration comprises four distinct classes: no damage (green), minor damage (yellow), major damage (orange), and destroyed (red). The class in question can be observed in the ground truth image that has been obtained. However, for the purposes of this research, the author will introduce an additional

class, namely the unlabelled class (grey). The total number of classes utilised in this study amounts to five. An exemplar of the data utilised is presented in Figure 1.

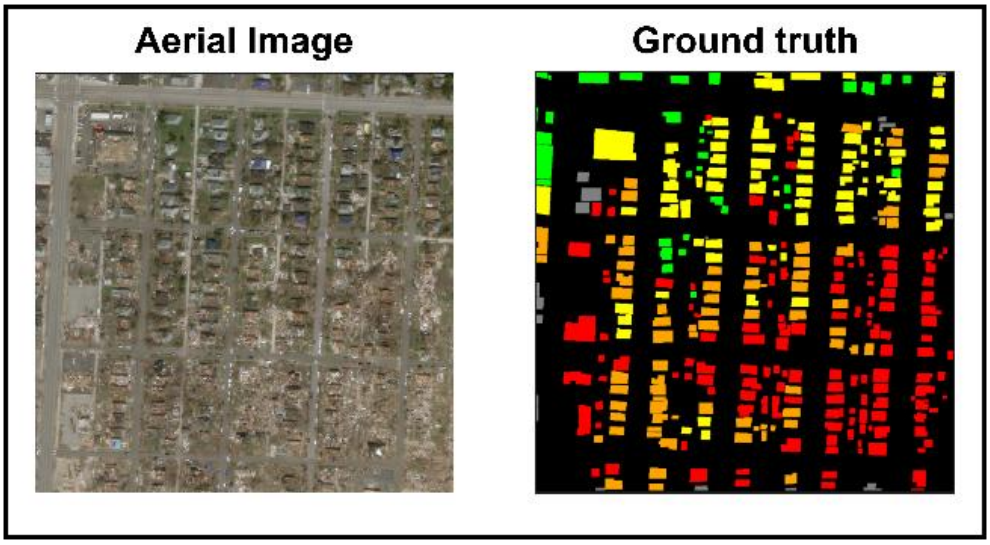

Figure 1. Example of Dataset

Figure 1 illustrates an example of the data used in this study. On the left is the original aerial image, showing a top-down view of a residential area that has sustained damage. On the right is the corresponding ground truth mask. This mask provides pixel-level annotations for each building, where different colors represent the specific damage class: green for "no-damage," yellow for "minor-damage," orange for "major-damage," red for "destroyed," and grey for "unlabeled" structures. This pair of image and mask serves as a single data sample for training and evaluating the segmentation model.

Data preprocessing involves transforming raw data into a clean, organised format that is suitable for analysis, particularly in the context of data mining, machine learning and other data science tasks. This step is essential because real-world data is often incomplete or inconsistent and may contain errors, which can negatively impact the performance and accuracy of analytical models [14].

The subsequent procedure is the data preprocessing stage, which is conducted prior to commencing the training process. At this stage, the image will be subjected to patchification, after which the results of this process will be normalised. The process of image segmentation, whereby an image is divided into smaller components, is known as 'patchify' [15]. The rationale behind this process is that the original image resolution size of 1024 x 1024 is too substantial for direct utilisation in the convolution and training processes. Consequently, the author will divide it into 256 x 256 sizes, each.

Subsequent to the implementation of the patchification process, the augmentation process is then initiated with the objective of further enriching the data and reducing the occurrence of overfitting during the training process [16]. In this particular instance, the augmentation process entails the rotation of each image that has undergone patchification by 90°, 180°, or 270°.

Subsequent to the augmentation process, a normalisation procedure will be executed. The process will be such that the value of each pixel is divided by 255, resulting in a value that falls within the range of 0 to 1. This is done with the intention of reducing the computational burden during the training process [17].

Model architecture refers to the structured design and organisation of a machine learning model. It specifies how its components, such as layers, neurons and connections, are arranged and interact to process data and produce predictions. In deep learning, for instance, the model's architecture determines the sequence and types

of layers (e.g. convolutional, pooling, or fully connected) and how data flows through them. This ultimately influences the model's ability to learn and represent complex patterns [18].

The architecture model utilised in this study employs an architecture comprising four pairs of encoder-decoder blocks and a single bottleneck block. As illustrated in Figure 2, the architectural model is represented by the image.

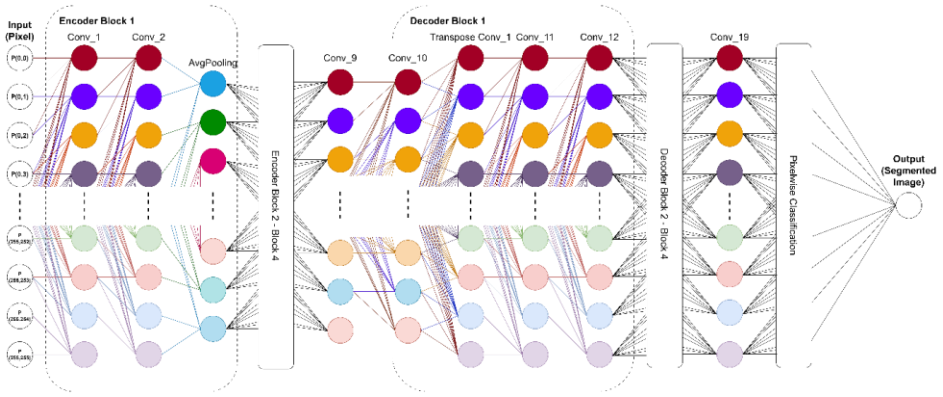

Figure 2. Proposed Model

As demonstrated in Figure 2, the proposed model follows a U-Net-like structure, consisting of a contracting path (encoder) on the left and an expansive path (decoder) on the right. The encoder accepts the input pixels and passes them through a series of convolutional and pooling layers (AvgPooling) in order to capture contextual features while progressively reducing spatial dimensions. The decoder path then takes these features and uses transpose convolutions to upsample them, gradually restoring the spatial resolution. A fundamental component of this architecture is the skip connections, which concatenate feature maps from the encoder path directly to the corresponding layers in the decoder path. This process enables the model to recuperate fine-grained spatial information that is forfeited during the encoding phase, a factor that is pivotal for the generation of precise segmentation maps. The final layer performs a pixel-wise classification to generate the segmented output image.

Within each block, the encoder will execute a convolution process with ReLU activation, followed by batch normalisation and culminating in pooling. Moreover, the decoder component will execute transpose convolution, followed by skip connection, utilising the feature map from the encoder. Following the execution of the transpose convolution operation, the subsequent convolution process will be conducted in the conventional manner, accompanied by the implementation of batch normalisation. As indicated in the section concluding the decoder, a pixel-based process will be conducted to categorise each pixel according to the predetermined class. This process also endeavours to calculate the loss function.

In order to mitigate the issue of overfitting, this study relied on two primary techniques. The initial approach is data augmentation, as outlined in the preprocessing section, which involves the artificial enhancement of the diversity of the training data. The second is the implementation of Batch Normalization layers following each convolution process within the encoder and decoder blocks. Batch Normalization has been demonstrated to assist in the regularisation of models and the stabilisation of the training process . However, it should be noted that other common regularization techniques, such as Dropout layers, were not implemented in the proposed architecture. The decision to exclude them was made to first establish a baseline performance of the U-Net structure with minimal architectural additions.

Training a convolutional neural network (CNN) involves teaching the network to recognise patterns in data, typically images, by iteratively adjusting its internal parameters (weights and biases) to minimise the difference between its predictions and the actual labels. The process includes a forward pass, in which data is processed through the network layers to produce an output; the calculation of a loss function, which measures

prediction error; backpropagation, which computes the gradients of the loss with respect to the parameters; and optimisation steps (e.g. gradient descent), which update the parameters and improve accuracy over many iterations or epochs [19], [20].

Within the context of the training process, it is imperative to initialise the parameters that will subsequently be utilised. These parameters are commonly referred to as hyperparameters. The hyperparameters that have been identified as relevant in this context include learning rate, epoch, batch size, and error tolerance. Following the initialisation process, the value of the hyperparameters is fixed and remains constant throughout the training process [21]. The following hyperparameters, which will be utilised in the present study, can be observed in Table 1.

Table 1. Hyperparameter

No	***Hyperparameter***	Nilai
1.	*Learning rate*	0.001
2.	*Epoch*	50 *epoch*
3.	*Batch size*	2 *Batch size*

Table 1 specifies the key hyperparameters used for the training phase of the model. A learning rate of 0.001 was chosen to control the step size of parameter updates during optimization. The model was trained for a total of 50 epochs, meaning it iterated through the entire training dataset 50 times. A batch size of 2 was used, indicating that the model processed two images at a time before updating its weights. These specific values were selected to balance training time and model performance.

In this study, model evaluation will be conducted using the mean intersection over the union score (MIoU), a metric that is frequently abbreviated. The resulting value will range from 0 to 1, with 1 representing a more accurate prediction[22]. The calculation formula is expressed in equation 1 :

$$MIoU = \sum_{1}^{n} \frac{TP}{(TP + FP + FN)} * \frac{1}{n} \quad (1)$$

Description:
TP = true positive value
FP = false positive value
FN = false negative value

3. Result and Discussions

The ensuing discourse will be divided into the following sections: data preparation, data preprocessing, training model, predicted image and evaluation model. Following the collection of data from the Xview2 Dataset, two distinct types of data were employed during the model training process: namely, the original aerial images showing post-disaster scenarios and their corresponding ground truth masks. The Xview2 dataset has been specifically designed for building damage assessment challenges, making it a highly suitable source for this research. The training data set under consideration is comprised of 50 high-resolution images, with a spatial resolution of 1024 x 1024 pixels. The high resolution of the system is of critical importance in the capture of the fine-grained details necessary to distinguish between different levels of structural damage. Each aerial photograph is paired with a meticulously annotated ground truth image, which provides the pixel-level labels that are essential for training a supervised semantic segmentation model.

The collected data will undergo a series of preliminary processing stages prior to its utilisation in the model training process. As delineated in subchapter 3.2, the preliminary processing stages encompass the processes of patching, augmentation, and normalisation. The total number of high-resolution images utilised is 50, with a resolution of 1024 x 1024. Subsequent to this preprocessing stage, the aggregate number of images will

amount to 3,200. As mentioned in the data collection stage (Section 2.1), this expanded dataset was then split using a 70:30 ratio, resulting in 2,240 patches for the training set and 960 patches for the testing set. The following example illustrates the outcomes of the data preprocessing procedure, as depicted in Figure 3.

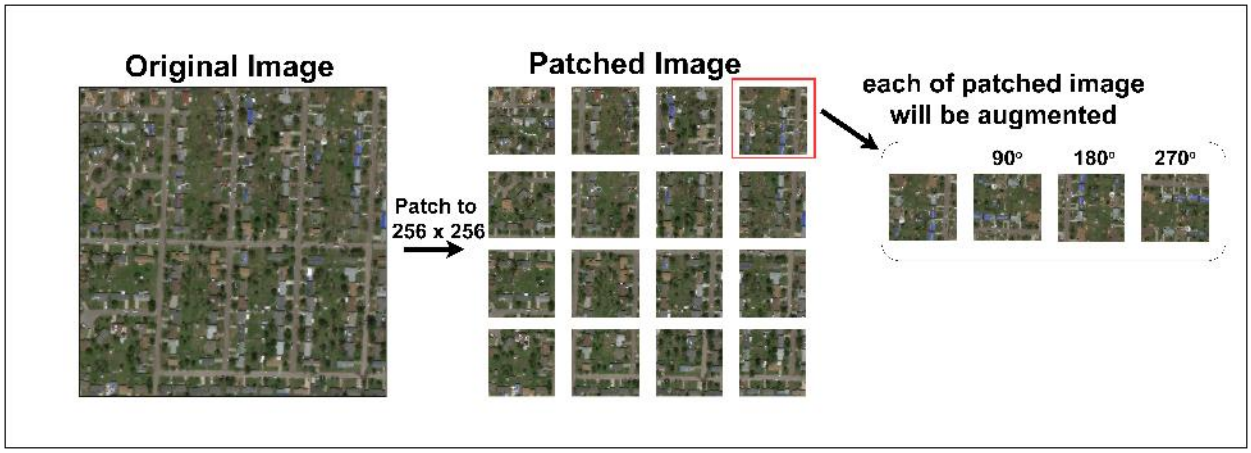

Figure 3. Preprocessing Data

Figure 3 presents a visual summary of the data preprocessing workflow. The 'Original Image' on the left is a high-resolution 1024x1024 satellite image. The image is then subjected to a process known as 'patching', whereby it is divided into multiple smaller 256x256 images, as illustrated in the 'Patched Image' grid. Finally, in order to increase data diversity and mitigate overfitting, each of these patched images is augmented through rotation at 90°, 180°, and 270°, thereby creating additional training samples.

Subsequent to the completion of the data preparation stage, the data will then progress to the model training stage. Hyperparameters for the training process are delineated in section 3.4.

The convolutional neural network model utilises four encoder and decoder blocks, thus comprising a total of 27 hidden layers within this architecture. The model utilises the mean value of the pool during the aggregation process. The model's accuracy was determined to be 0.9577, as indicated in Figure 4, following the training process that incorporated 2240 train data and 960 testing data.

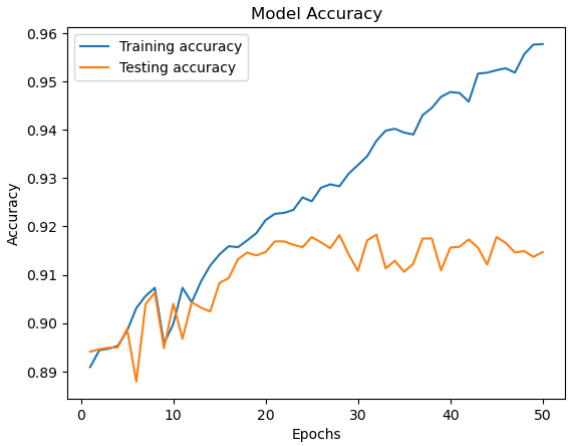

Figure 4. Model Accuracy

As demonstrated in Figure 4, the training and testing accuracy curves are presented. The training accuracy (blue line) displays a consistent upward trend, attaining a high value of approximately 0.95, while the testing accuracy (orange line) exhibits variability and stabilises at a lower value of approximately 0.91-0.92. The

presence of a clear gap between these two curves serves as a reliable indicator of overfitting. Subsequently, the loss value obtained was 0.1133, as illustrated in Figure 5

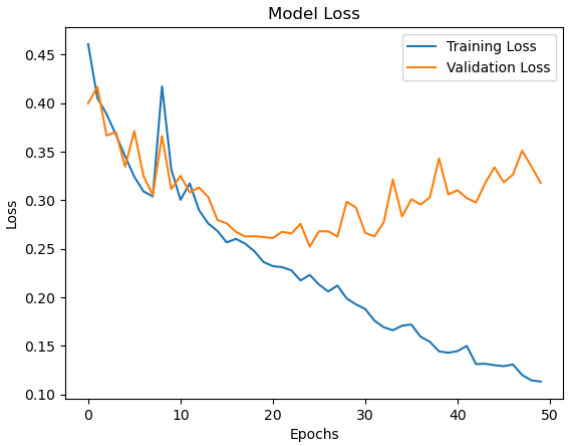

Figure 5. Model Loss

As illustrated in Figure 5, the training and validation loss curves are presented. The training loss (blue line) displays a consistent decrease, while the validation loss (orange line) exhibits a more erratic pattern, increasing after an initial decline. This further corroborates the hypothesis that the model is memorising the training data rather than demonstrating effective generalisation.

Following the conclusion of the training process for the model, the subsequent model that has undergone training weights will be utilised in the prediction process. This will result in the display of the segmentation result image. The data utilised in this process constitutes 30% of the total data. This 30% figure has been determined during the data preparation process, which allocates 70:30 for the training and testing data, respectively. The model has predicted a total of 960 data points. The ensuing illustration exemplifies the outcomes yielded by the predict model, as depicted in Figure 6.

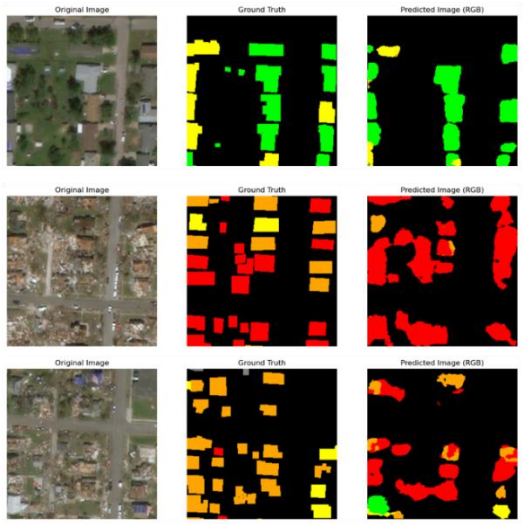

Figure 6. Predicted Image

As illustrated in Figure 6, the model produced three distinct segmentation results on three test images. Each row in the presentation illustrates the original input image, the corresponding ground truth mask, and the model's predicted mask. While the model accurately identifies the general location and class of some buildings

(e.g., the green "no-damage" buildings in the top row), it demonstrates significant confusion in more complex scenes. For instance, in the second and third rows, the model incorrectly classifies many "major-damage" (orange) and "destroyed" (red) buildings, often mixing the two classes or misclassifying them as other damage levels. This highlights the challenges discussed in the evaluation section.

Following the successful prediction by the model, the results of the prediction will be utilised for the purpose of model evaluation. This will be achieved by calculating the mean intersection over union (mIoU) and displaying the confusion matrix. The mIoU result from predicting all testing data is 0.3018. It is evident that the scale of this phenomenon is relatively diminutive. This phenomenon can be attributed to the paucity of datasets, thereby constraining the knowledge acquired by the model during the training process. As demonstrated in Table 2, both the training results and the mIoU results are available for perusal.

Table 2. Result of training and evaluation

Model	**Accuracy**	***Loss***	**mIoU**
Proposed model	0.9577	0.1133	0.3118

As illustrated in Table 2, the ultimate performance metrics of the proposed model are outlined following the completion of 50 epochs of training. The model demonstrated a high final training accuracy of 0.9577, accompanied by a low loss of 0.1133. However, the more critical mIoU score, which measures segmentation quality on the test set, was only 0.3118, indicating a significant discrepancy between training performance and real-world applicability. The subsequent step involves the examination of the confusion matrix, as illustrated in Figure 7.

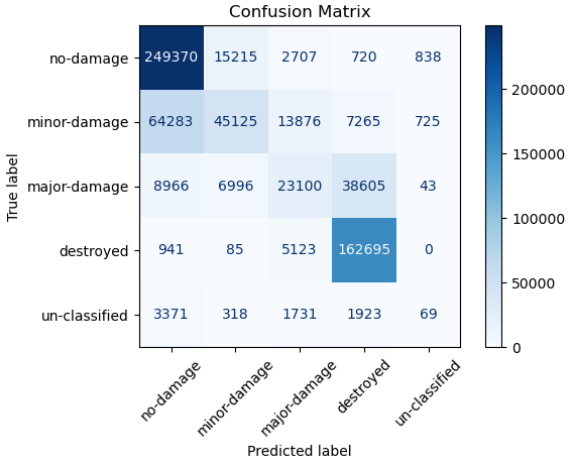

Figure 7. Confussion Matrix

Figure 7 provides a detailed quantitative breakdown of the model's classification performance for each class. The diagonal values (illustrated in dark blue) represent the number of pixels that have been correctly classified. While the model performs well for the "no-damage" (249,370 correct) and "destroyed" (162,695 correct) classes, the off-diagonal values reveal significant confusion. For instance, a significant proportion of pixels classified as "minor-damage" (64,283) were erroneously categorised as "no-damage", while a substantial number of pixels designated as "major-damage" (38,605) were inaccurately classified as "destroyed". This matrix provides quantitative confirmation of the model's inability to differentiate between visually similar levels of damage.

This observation has also been made in the prediction results. It is evident that there is a similarity in the classification systems, with the presence of categories such as 'no damage', 'minor damage', and 'major damage'

alongside the concept of 'destroyed'. This phenomenon can be attributed to the limited richness of the data employed during the training process.

An in-depth analysis of the confusion matrix in Figure 7 and the prediction results in Figure 6 reveals that the model's primary challenge lies in the high inter-class similarity and potential label ambiguity within the dataset. The frequent confusion between "minor-damage" and "no-damage," and similarly between "major-damage" and "destroyed," can be attributed to several factors.

Firstly, from an aerial perspective, the visual similarity between classes is significant. Minor damage, such as small holes or cracked roofing tiles, can be visually almost indistinguishable from the normal texture of an intact roof, especially when affected by shadows or image resolution limitations. This is quantitatively supported by the confusion matrix, which shows 64,283 instances of "minor-damage" pixels being misclassified as "no-damage." Similarly, the visual line between a building with a collapsed roof ("major-damage") and a pile of rubble ("destroyed") is often blurred. The model struggles to differentiate these states, as evidenced by the 38,605 "major-damage" pixels that were incorrectly predicted as "destroyed."

Secondly, there is an inherent label ambiguity in defining the boundaries of damage levels. The threshold separating "minor" from "major" damage, or "major" damage from "destroyed," is often subjective and can vary even among human annotators. This ambiguity in the ground-truth labels makes it exceedingly difficult for the model to learn a consistent and precise decision boundary, as the features it learns for one class heavily overlap with another. This issue is apparent in the middle and bottom rows of Figure 6, where the model's predictions show a mixture of classes where the ground truth is more distinctly defined. This indicates that the model is struggling not just with feature extraction, but with the fundamental definition of the classes themselves within the provided dataset.

4. Conclusions and Future Works

The present study investigated the application of a custom-built Convolutional Neural Network (CNN) with a U-Net architecture for the semantic segmentation of post-disaster building damage. The findings indicate that while the proposed model achieved a high training accuracy of 0.9577, this metric was misleading. The performance metric of choice for this task, Mean Intersection over Union (mIoU), was found to be a mere 0.3118, thus indicating that the model was not effective in accurately segmenting the various damage classes.

The principal cause of this unsatisfactory performance was identified as severe overfitting, a phenomenon that can be ascribed to a number of intertwined factors that have been identified in this research. Firstly, the proposed 27-layer architecture, built from scratch, proved to be overly complex for the limited diversity of the training data, which was derived from only 50 unique source images. Secondly, the model demonstrated a notable challenge in differentiating between "minor-damage" and "no-damage" and "major-damage" and "destroyed," particularly in the context of high inter-class visual similarity and inherent label ambiguity present within the dataset. The absence of explicit regularisation techniques, such as Dropout, served to compound the overfitting issue.

Notwithstanding the suboptimal mIoU score, this research makes a significant contribution by establishing a critical performance baseline. This finding highlights the substantial limitations of a CNN approach that is entirely developed from the beginning for this intricate, real-world segmentation task. The investigation further substantiates the necessity of employing more sophisticated techniques.

To address the identified shortcomings, future work should focus on several key improvements. First, adopting a transfer learning approach is recommended, moving away from the from-scratch methodology. By leveraging pre-trained encoders, such as ResNet50 or, more effectively, EfficientNet—both of which have shown successful results in similar studies—models can utilize robust, pre-learned features, which are essential for handling complex visual tasks. Additionally, implementing stronger regularization techniques is crucial. Specifically, incorporating Dropout layers into the model's architecture will help combat overfitting more effectively. Finally, efforts should be made to enhance data diversity. Although challenging, increasing the

variety of unique source images in the training set is essential for improving the model's generalization capabilities across different geographical locations, disaster types, and building architectures. By pursuing these strategies, future research can build upon the baseline established here to develop a genuinely effective and reliable automated system for post-disaster building damage assessment.

5. References

[1] J. Rosselló, S. Becken, and M. Santana-Gallego, "The effects of natural disasters on international tourism: A global analysis," *Tour Manag*, vol. 79, p. 104080, Aug. 2020, doi: 10.1016/j.tourman.2020.104080.

[2] J. Padli, M. Shah Habibullah, and A. H. Baharom, "Economic impact of natural disasters' fatalities," *Int J Soc Econ*, vol. 37, no. 6, pp. 429–441, May 2010, doi: 10.1108/03068291011042319.

[3] A. Pramono, I. W. S. Mahendra, I. B. A. Wijaya, I. A. Agustina, and R. T. Herman, "Diagnosis and Repair of the Cracking House Next to the River," *IOP Conf Ser Earth Environ Sci*, vol. 998, no. 1, p. 012002, Feb. 2022, doi: 10.1088/1755-1315/998/1/012002.

[4] E. H. Zaryabi, B. Kalantar, L. Moradi, A. A. Halin, and N. Ueda, "MSBDA-Net: Multi-scale Siamese Building Damage Assessment Network," in *2022 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, IEEE, Dec. 2022, pp. 1–6. doi: 10.1109/CSDE56538.2022.10089353.

[5] P. Pahlavani, F. Samadzadegan, and M. R. Delavar, "A GIS-Based Approach for Urban Multi-criteria Quasi Optimized Route Guidance by Considering Unspecified Site Satisfaction," 2006, pp. 287–303. doi: 10.1007/11863939_19.

[6] D. S. Deswita Indriani, E. Juni Arta Sinaga, G. Oktavia, H. Syahputra, F. Ramadhani, and I. Komputer, "Identifikasi Tanda Tangan Dengan Menggunakan Metode Convolution Neural Network (CNN)".

[7] A. T. W. Almais *et al.*, "SASSD: A Smart Assessment System For Sector Damage Post-Natural Disaster Using Artificial Neural Networks," in *2023 2nd International Conference on Computer System, Information Technology, and Electrical Engineering (COSITE)*, IEEE, Aug. 2023, pp. 96–101. doi: 10.1109/COSITE60233.2023.10249540.

[8] Moch. N. Annafii, O. V. Putra, T. Harmini, and N. Trisnaningrum, "Segmentasi Semantik pada Citra Hama Leafblast Menggunakan Unet dan Optimasi Hyperband," *Prosiding Sains Nasional dan Teknologi*, vol. 12, no. 1, pp. 453–459, Nov. 2022, doi: 10.36499/psnst.v12i1.7230.

[9] I. Kotaridis and M. Lazaridou, "SEMANTIC SEGMENTATION USING A UNET ARCHITECTURE ON SENTINEL-2 DATA," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLIII-B3-2022, pp. 119–126, May 2022, doi: 10.5194/isprs-archives-XLIII-B3-2022-119-2022.

[10] A. Di Benedetto, M. Fiani, and L. M. Gujski, "U-Net-Based CNN Architecture for Road Crack Segmentation," *Infrastructures (Basel)*, vol. 8, no. 5, p. 90, May 2023, doi: 10.3390/infrastructures8050090.

[11] S. Gangurde, "Building and Road Segmentation Using EffUNet and Transfer Learning Approach," Jul. 2023.

[12] W. Li, J. Wu, H. Chen, Y. Wang, Y. Jia, and G. Gui, "UNet Combined With Attention Mechanism Method for Extracting Flood Submerged Range," *IEEE J Sel Top Appl Earth Obs Remote Sens*, vol. 15, pp. 6588–6597, 2022, doi: 10.1109/JSTARS.2022.3194375.

[13] Y. Li *et al.*, "CTMU-Net: An Improved U-Net for Semantic Segmentation of Remote-Sensing Images Based on the Combined Attention Mechanism," *IEEE J Sel Top Appl Earth Obs Remote Sens*, vol. 16, pp. 10148–10161, 2023, doi: 10.1109/JSTARS.2023.3326960.

[14] D. Varma, A. Nehansh, and P. Swathy, "Data Preprocessing Toolkit : An Approach to Automate Data Preprocessing," *INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, vol. 07, no. 03, Mar. 2023, doi: 10.55041/IJSREM18270.

[15] Z. Chen *et al.*, "DPT: Deformable Patch-based Transformer for Visual Recognition," Jul. 2021, doi: 10.1145/3474085.3475467.

[16] S. Yang, W. Xiao, M. Zhang, S. Guo, J. Zhao, and F. Shen, "Image Data Augmentation for Deep Learning: A Survey," Apr. 2022.

[17] W. Gao, "Investigation of multiple convolutional neural network models on emotion detection," *Applied and Computational Engineering*, vol. 22, no. 1, pp. 35–41, Oct. 2023, doi: 10.54254/2755-2721/22/20231164.

[18] Y. Tzach *et al.*, "The mechanism underlying successful deep learning," May 2023.

[19] R. Scodellaro, A. Kulkarni, F. Alves, and M. Schröter, "Training Convolutional Neural Networks with the Forward-Forward algorithm," Dec. 2023.

[20] M. S. Tuna and A. Kristianto, "Klasifikasi Cuaca Berbasis Citra dengan Model CNN LeNet-5 yang Dimodifikasi," *J-Intech : Journal of Information and Technology*, no. 204, pp. 401–410, 2022.

[21] H. Jin, "Hyperparameter Importance for Machine Learning Algorithms," Jan. 2022, [Online]. Available: http://arxiv.org/abs/2201.05132

[22] M. Z. Khan, M. K. Gajendran, Y. Lee, and M. A. Khan, "Deep Neural Architectures for Medical Image Semantic Segmentation: Review," *IEEE Access*, vol. 9, pp. 83002–83024, 2021, doi: 10.1109/ACCESS.2021.3086530.