Identifying Damage Levels in a Post-Hazard Scenario Using Semantic Segmentation and Machine Learning

Project Final Report

Team Members: Ravi Shastri, Mobina Amrollahi, Sayan Das, Nagasharan Sathish

ABSTRACT

Natural hazards have threatened human civilization for centuries, causing significant loss of life and property. However, the magnitude of these losses is often exacerbated by delays in post-disaster response rather than the hazard itself. Machine learning and artificial intelligence advancements offer promising solutions to bridge this critical response gap. Among these, semantic segmentation, a powerful deep learning technique, is crucial in identifying damage levels caused by hurricanes. This study uses historical damage data from past hurricanes to train and evaluate four segmentation models: Pyramid Scene Parsing Network (PSPNet), DeepLabV3+, Segmenter, and Attention U-Net. Among them, Segmenter achieved the highest mean Intersection over Union (IoU) of (70.36%), pixel accuracy (87.02%), and PSPNet ranked second with the highest recorded values for precision (72.2%), recall (62.2%), and F1 score (64.5%). This comparative analysis highlights the effectiveness of different architectures for disaster assessment and response planning, with PSPNet and Segmenter standing out as the most reliable models for fine-grained damage detection.

1 Introduction

An efficient framework to expedite the recovery path in a post-hazard scenario requires identifying the incurred damage at first (Rosenberg et al. (2022), Xue et al. (2023)). Conventionally, this damage identification is done by manually assessing the images taken post-event occurrence. Although this practice yields good results, the accuracy is highly dependent on the skill of the inspector and the time taken to perform the inspection Chiou (2010). As an alternative, the semantic segmentation technique could be leveraged in such cases (Alisjahbana et al. (2024), Rahnemoonfar et al. (2023)), which not only reduces the inspection time by a significant margin but also increases the accuracy of work. A semantic segmentation performs a pixel-level classification, making identifying damage levels easier and preventing error accumulation, as each pixel is examined separately Guo et al. (2018). In this work, semantic segmentation is used to perform classification at two levels: grayscale classification, also known as bounding box detection, for object boundary delineation; and color-masking classification, for identifying distinct objects within the delineated boundaries and assessing their damage levels. Different algorithms used to perform such evaluations are Pyramid Scene Parsing Network (PSPNet) (Zhao et al. (2017)), DeepLabV3+ (Chen et al. (2018)), Segmentar (Strudel et al. (2021)), and Attention U-Net (Oktay et al. (2018)). The source code used for training, testing, and visualization is publicly available here.

2 Related Work

With increased computational capabilities and advanced architectures, deep learning algorithms are now widely used across various engineering domains Zhang et al. (2021b). Fields such as civil Çakıroğlu and Süzen (2020), mechanical Amini et al. (2023), and aerospace engineering

Wang and Ma (2024) increasingly employ computer vision techniques for classification and damage identification tasks. Huang (2023) makes use of transfer learning with Residual Neural Network (ResNet) 50 and MobileNet to classify RGB images into 21 classes. The target classes include roads, bridges, docks, trees, etc. Ekiz and Acar (2025) uses segmentation networks to identify building boundaries: boundary refinement module leverages morphological operations and line-fitting algorithms to straighten edges and suppress noise. Wang et al. (2022) proposes a Vision Transformer (ViT)-based multiscale feature fusion method named segmentation transformer–multiscale feature pyramid decoder (SETR-MFPD), which combines the expressivity of ViTs with the multiscale fusion power of CNN decoders for remote sensing image segmentation. The authors demonstrate that by transforming 2D ViT features into 3D multiscale maps, enhancing them with a dimension attention module (DAM), and decoding through a multiscale feature pyramid decoder (MFPD), the model effectively captures both semantic and spatial context while reducing information loss.

3 Methods

3.1 Dataset

The data obtained from Rahnemoonfar et al. (2023) consists of three sets of grayscale, color-masked, and ground truth images. Each set includes 3595 images, totaling 10,785 images for training. Similarly, there are 1347 images for validation and 1350 for testing.
Initial extraction of the grayscale and color masks for the ground truth images is shown in Fig. 6b for reference. Similar extraction procedures were applied to all images used for training the models. Grayscale images are used for object boundary training, while color masks label object categories and damage levels. Color mask prediction is performed only on pixels identified in the grayscale inference.
The first three models, PSPNet, DeepLabV3+, and Attention U-Net, were trained for 20 epochs, while Segmenter was trained for 50 epochs. All models used a learning rate of 0.001. To improve computational efficiency and reduce training time, the input images were resized to 512×512 for the first three models, and to a crop size of 640×640 for Segmenter.
After training, testing is performed, and comparisons between the ground truth, actual predictions, and predicted pixel labels for both grayscale and color masks are done.
To maintain consistency across models, the class labels used in color masking are based on the definitions provided in Table 1. While the models correctly identify many bounding boxes, the accuracy of color predictions remains limited, indicating the need for further hyperparameter tuning.

3.2 Metrics

To evaluate model performance, several accuracy metrics are calculated. Among these, the Intersection over Union (IoU) is a key metric for assessing semantic segmentation quality Everingham et al. (2010). IoU measures the overlap between predicted and ground truth pixel regions based on following formula:

$$\text{IoU} = \frac{target \cap prediction}{target \cup prediction} \tag{1}$$

Table 1: Class Labels and Corresponding Colors

Class Name	Color
Background	Black
Water	Blue
Building No Damage	Green
Building Minor Damage	Yellow
Building Major Damage	Orange
Building Total Destruction	Red
Vehicle	Cyan
Road-Clear	White
Road-Blocked	Gray
Tree	Forest Green
Pool	Indigo

3.3 Algorithms

Semantic segmentation is a pixel-level classification technique applicable to various deep learning architectures Long et al. (2015); Csurka et al. (2022). Although the underlying mechanism is consistent across different algorithms, performance varies significantly depending on the model architecture Kamann and Rother (2020). Architecture choice directly affects the depth and quality of feature extraction, thereby influencing the overall performance Ghosh et al. (2022); Basha et al. (2020).

In this study, four widely adopted architectures, PSPNet, DeepLabV3+, Segmenter, and Attention U-Net, are employed for the post-hazard semantic segmentation inference task. These models will be discussed in detail in the following subsections.

3.3.1 PSPNet

PSPNet is a semantic segmentation technique that leverages contextual information at multiple scales using a Pyramid Pooling Module (PPM). This module enables the model to aggregate global and local context, enhancing scene understanding by learning representations at multiple levels of granularity Zhao et al. (2017).

Architecture. The architecture of PSPNet comprises three main components: (i) a Feature Extraction Backbone, (ii) a Pyramid Pooling Module (PPM), and (iii) a Final Classification Layer. The feature extractor typically uses a deep convolutional neural network such as ResNet-50 or ResNet-101 He et al. (2016), which captures high-level semantic features from the input image. The PPM is the core innovation of PSPNet; it captures multi-scale contextual information using four pooling scales: (i) 1×1 global average pooling, (ii) 2×2 coarse-level pooling, (iii) 3×3 medium-level pooling, and (iv) 6×6 fine-level pooling. These pooled features are then upsampled and concatenated to form a rich feature representation. Finally, a 1×1 convolutional layer is used to generate a pixel-wise classification map. An overview of the PSPNet architecture Zhao et al. (2017) is demonstrated in Figure 1.

Implementation Details. Like other convolutional neural networks, the process begins with feature extraction from the input image using the ResNet backbone. Afterward, the Pyramid Pooling Module performs pooling at multiple scales, which are upsampled and concatenated to

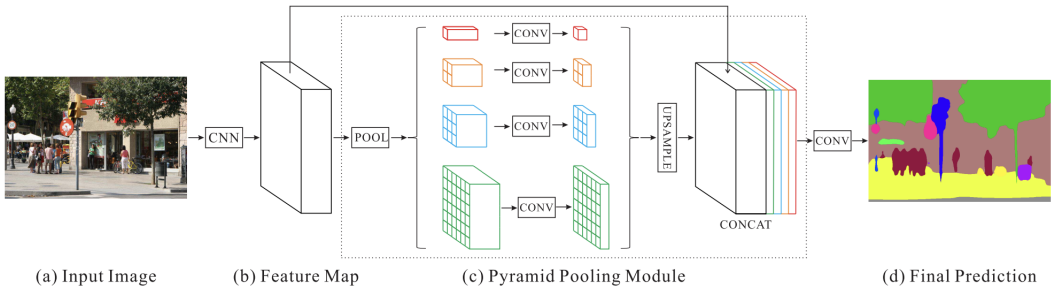


Figure 1: Overview of the PSPNet Architecture

match the original resolution. These aggregated features are then passed through the classification layer to produce the final segmentation map.

3.3.2 DeepLabV3+

DeepLabV3+ is an advanced semantic segmentation model developed by Google Research that extends DeepLabV3 by adding a decoder module to improve spatial detail and object boundary detection. It incorporates Atrous Spatial Pyramid Pooling (ASPP) for multi-scale feature extraction and depthwise separable convolutions for computational efficiency. Supporting popular backbones such as ResNet and Xception, DeepLabV3+ achieves state-of-the-art accuracy and is widely applied in domains including autonomous driving, medical imaging, satellite analysis, and agriculture Chen et al. (2018).

Architecture. DeepLabV3+ features an encoder-decoder structure designed for precise segmentation and boundary refinement. The encoder uses a backbone network like ResNet or Xception to extract deep semantic features. These are then passed through the ASPP module, which applies parallel atrous (dilated) convolutions with varying dilation rates to capture contextual information at multiple scales. To enhance spatial detail, low-level features from earlier layers of the encoder are fused with the upsampled ASPP output in the decoder. This fusion is followed by additional convolution and upsampling operations to produce a pixel-wise segmentation map. An overview of the DeepLabV3+ architecture is illustrated in Figure 2.

Implementation Details. The inference process begins by feeding the input image through the backbone (e.g., ResNet or Xception), which generates hierarchical feature maps. Atrous convolutions are used in the deeper layers to preserve spatial resolution while maintaining a wide receptive field. The resulting high-level features are processed by the ASPP module, which captures multi-scale context using parallel dilated convolutions and global average pooling. The ASPP output is then upsampled and combined with low-level features from early encoder layers after they are compressed via a 1×1 convolution. The combined features are refined through a sequence of 3×3 convolutions and then upsampled to match the input image resolution, resulting in a dense, per-pixel segmentation output. This architecture effectively balances global context and fine detail, making DeepLabV3+ well-suited for complex segmentation tasks.

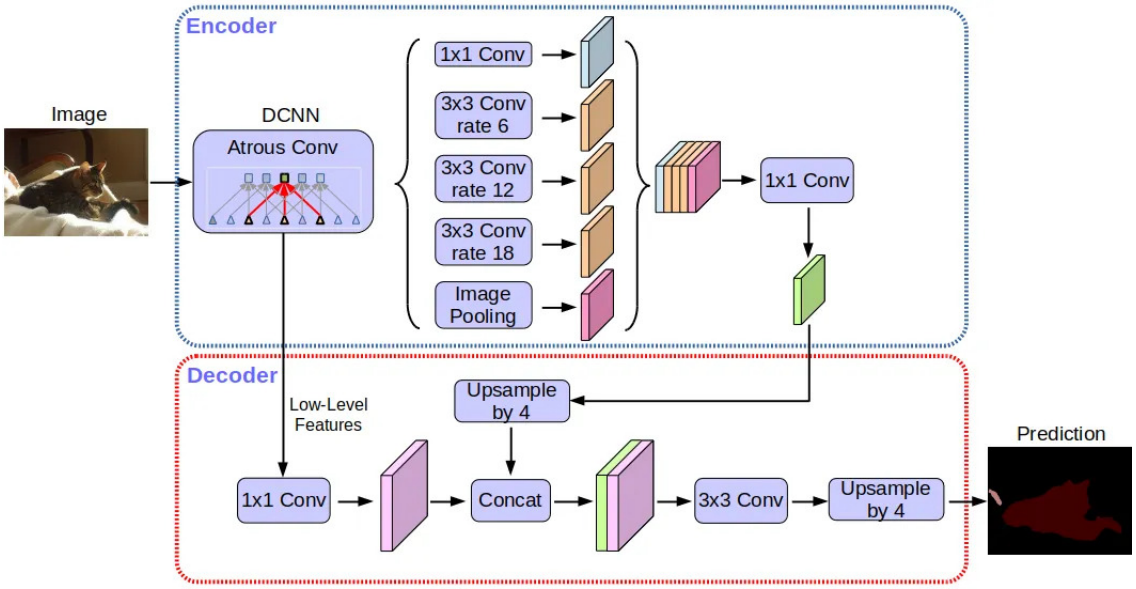


Figure 2: Overview of the DeepLabV3+ Architecture

3.3.3 Segmenter

The paper Segmenter: Transformer for Semantic Segmentation Strudel et al. (2021) proposes a transformer-based architecture for semantic segmentation tasks. It employs output embeddings associated with image patches and generates class labels using either a point-wise linear decoder or a mask transformer decoder. Instead of convolutions, this architecture naturally captures global image context by design, resulting in competitive performance on standard image segmentation benchmarks.

Architecture. An overview of the Segmenter architecture is shown in Figure 3. The input image is divided into a sequence of patches. These patches are encoded into embeddings and decoded into segmentation maps using either a point-wise linear decoder or a mask transformer decoder. The architecture inherently captures global context by design, without relying on convolutions. Detailed information about the encoder and decoder can be found in Strudel et al. (2021).

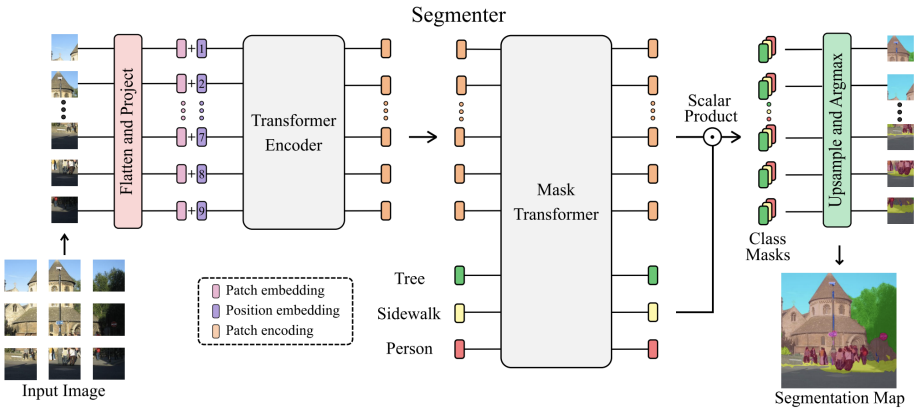


Figure 3: Overview of the Segmenter approach.

Implementation Details. Training was conducted on an NVIDIA GeForce RTX 4090 GPU to handle the computational demands of transformer-based segmentation. The model utilizes a ViT backbone for expressive feature extraction and a Mask Transformer decoder for segmentation.

3.3.4 Attention U-Net

Architecture. The Attention U-Net introduced by Oktay et al. (2018) is an enhancement of the traditional U-Net architecture, designed to improve segmentation performance by integrating attention mechanisms. Unlike the standard U-Net, which processes all feature information uniformly, the Attention U-Net employs attention gates to selectively focus on relevant features while suppressing irrelevant regions in the input images. This targeted attention mechanism allows the model to emphasize salient features of varying shapes and sizes, enhancing its ability to distinguish important structures even in complex and cluttered scenes. The attention mechanism operates by computing spatial attention coefficients that moderate features from the encoder path before merging them with decoder features through skip connections. This approach effectively guides the model to concentrate on critical areas of interest, improving segmentation accuracy without significantly increasing computational complexity Oktay et al. (2018). Schlemper et al. (2019) further demonstrated the effectiveness of attention mechanisms in enhancing model sensitivity and precision across various segmentation tasks without significantly increasing computational load.

An overview of the Attention U-Net architecture is shown in Figure 4.

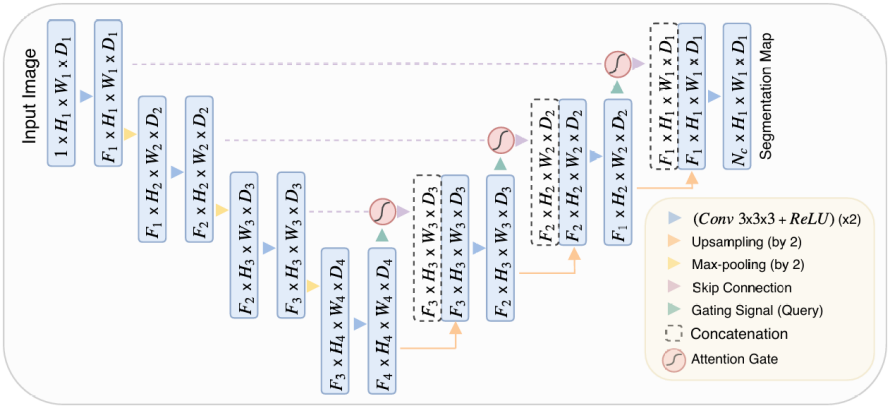


Figure 4: Overview of the Attention U-Net Architecture.

Implementation Details. The foundation of this model is the U-Net architecture, characterized by its symmetric encoder-decoder structure with skip connections. The encoder path reduces spatial dimensions while increasing feature channels via convolutional blocks and max pooling, capturing contextual information. The decoder restores spatial resolution using upsampling and convolutional operations Ronneberger et al. (2015). The key innovation in Oktay et al. (2018) is the use of attention gates within the skip connections, which act as feature selectors to highlight salient features and suppress irrelevant ones before merging encoder and decoder features.

For this experiment, a total of 3595 images were used. Images were resized to 256×256, and the model was trained for 20 epochs using a learning rate of 0.0001 on a CPU.

4 Results

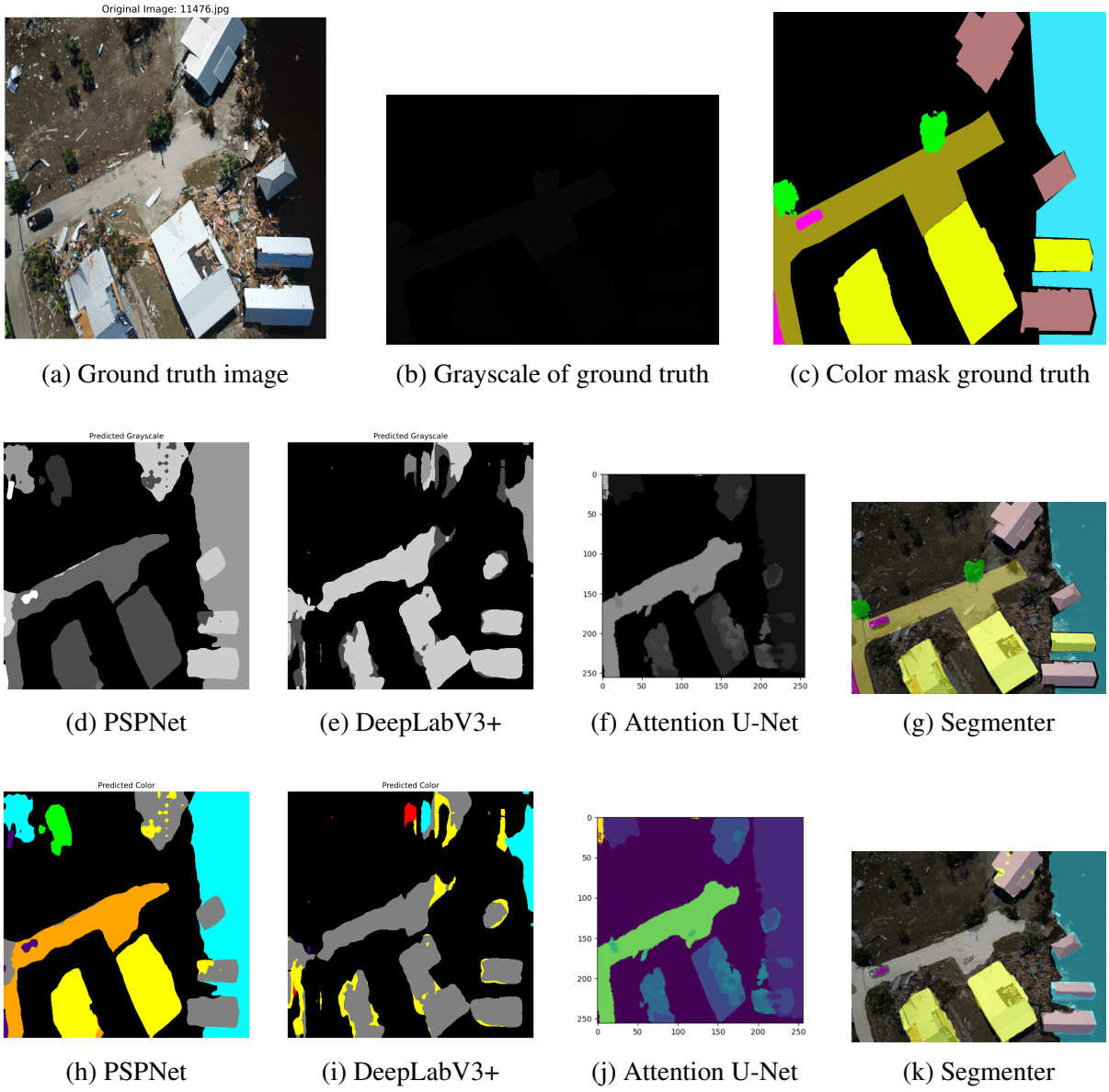
The class-wise IoU scores and overall performance metrics for the four segmentation models are presented in Tables 2 and 3, respectively. Notably, the Segmenter model does not include precision, recall, or F1 score values due to the substantial memory demands of these computations. Despite utilizing an NVIDIA GeForce RTX 4090 GPU, the calculation of these metrics exceeded available resources, making them infeasible within the current computational setup.
As shown in Table 2, Segmenter achieved the highest class-wise IoU in **8** out of **11** categories. PSPNet led in 3 categories and achieved the highest precision (72.2%), recall (62.2%), and F1 score (64.5%) (Table 3). In contrast, several models failed to segment certain classes entirely, DeepLabV3+ achieved a class-wise IoU of **0** for Classes 2, 6, 8, 9, 10, and 11, while PSPNet also had **0 IoU** for Classes 2 and 8. Attention U-Net performed better across most classes but still showed **0 IoU** in Classes 9 and 11.
In terms of mean IoU and pixel-level accuracy, Segmenter outperformed all other models with scores of 70.36% and 87.02%, respectively. For reference, the highest precision, recall, and F1 score among the remaining models were 72.2%, 62.2%, and 64.5%, respectively, all achieved by PSPNet. DeepLabV3+ had the weakest overall performance across both IoU and classification metrics, while Attention U-Net showed moderate results, excelling in specific classes but lacking consistency.
A visual comparison of the model outputs is provided in Figure 5, illustrating segmentation quality across grayscale and color mask predictions. As can be seen from the images, almost all four architectures can capture the bounding boxes, i.e., they can differentiate the background and the objects, such as houses, trees, vehicles, etc. But, not all the models can predict the color masking efficiently. PSPNet and Attention U-Net do predict the classes to some extent, while DeepLabV3+ fails to do the color mask predictions for the identified bounding boxes. As seen from Figs. 5c and 5k, Segmenter performs better than any other model in performing the color mask prediction.
Additional results across all models are provided in the appendix for reference and comparison.

Table 2: Class-wise IoU scores for different models. The highest value highlighted in **bold**.

Class	1	2	3	4	5	6	7	8	9	10	11
PSPNet	0.78	0.00	**0.78**	0.42	0.27	0.49	**0.77**	0.00	**0.52**	0.14	0.41
DeepLabV3+	0.61	0.00	0.35	0.09	0.39	0.00	0.34	0.00	0.00	0.00	0.00
Attention U-Net	0.69	0.69	0.38	0.19	0.18	0.20	0.20	0.50	0.00	0.70	0.00
Segmenter	**0.82**	**0.81**	0.60	**0.47**	**0.47**	**0.56**	0.41	**0.72**	0.35	**0.81**	**0.51**

Table 3: Performance metrics of segmentation models. The highest values highlighted in **bold**.

Model	**Mean IoU (%)**	**Pixel Accuracy (%)**	**Mean Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 Score (%)**
PSPNet	41.6	82.6	–	**72.2**	**62.2**	**64.5**
DeepLabV3+	15.5	74.0	–	62.9	38.0	40.8
Attention U-Net	36.8	76.9	–	46.8	45.5	45.3
Segmenter	**70.36**	**87.02**	**70.36**	–	–	–



(a) Ground truth image (b) Grayscale of ground truth (c) Color mask ground truth

(d) PSPNet (e) DeepLabV3+ (f) Attention U-Net (g) Segmenter

(h) PSPNet (i) DeepLabV3+ (j) Attention U-Net (k) Segmenter

Figure 5: Performance comparison of different architectures for both grayscale and color mask labels

5 Conclusions

The superior performance of Segmenter can be attributed to its Vision Transformer (ViT) backbone and Mask Transformer decoder, which allow for long-range contextual understanding and robust multi-scale feature fusion Zhang et al. (2021a). Moreover, while Segmenter's precision, recall, and F1 score could not be computed, its consistently high class-wise IoUs and top overall accuracy metrics strongly suggest it generalizes well across classes. For reference, the highest precision, recall, and F1 score values among the remaining models were 72.2%, 62.2%, and 64.5%, respectively, all achieved by PSPNet.

Certain classes exhibited a complete failure of detection, as indicated by an IoU score of zero. Two plausible explanations for this phenomenon are outlined below. First, the models may have failed to recognize specific classes even in the grayscale domain, thereby preventing the generation of corresponding color masks during post-processing. Second, a skewed distribution in the training dataset Van Hulse et al. (2007) may have resulted in inadequate representation of some classes, impairing the models' ability to learn distinguishing features for those categories.

Also, Segmenter stands out by leveraging global attention mechanisms, which allow it to model long-range dependencies across the entire image. This gives it a distinct advantage when dealing with class imbalance, as it can incorporate global context Pereira and Hussain (2024) to correctly classify even those regions associated with underrepresented classes.

Attention U-Net offers a more targeted approach by integrating attention gates into the traditional U-Net architecture Oktay et al. (2018). These gates enable the model to selectively emphasize relevant spatial regions while suppressing irrelevant features.

PSPNet, though pioneering in its use of pyramid pooling to capture multi-scale context, lacks mechanisms to adaptively prioritize minority-class features. Its pooling strategy aggregates spatial information at fixed levels, treating all features uniformly without any attention or weighting to favor rare class regions. As a result, PSPNet can become biased toward dominant classes Wang (2024) and often struggles to adequately capture fine-grained or infrequent structures in imbalanced datasets.

DeepLabV3+, while effective on balanced datasets, is the most constrained under class imbalance among these models. It relies on dilated convolutions and ASPP to extract context, but like PSPNet, its mechanisms are inherently local.

In summary, when dealing with class imbalance, these models typically perform in the following order:

$$Segmenter > Attention\ U\text{-}Net > PSPNet > DeepLabV3+ \tag{2}$$

These findings suggest that future work should consider techniques for addressing class imbalance, such as data augmentation Van Dyk and Meng (2001) or class-weighted loss functions Xu et al. (2020), and explore model architectures that enhance feature discrimination for minority classes.

6 Acknowledgement

We would like to thank Huy Quang[1] for making the implementation of the Segmenter model publicly available. Their work significantly contributed to the development and evaluation of our semantic segmentation experiments.

[1] https://github.com/quanghuy0497/Segmenter

References

Alisjahbana, I., Li, J., Zhang, Y., et al. (2024). "Deepdamagenet: A two-step deep-learning model for multi-disaster building damage segmentation and classification using satellite imagery." *arXiv preprint arXiv:2405.04800*.

Amini, M., Sharifani, K., and Rahmani, A. (2023). "Machine learning model towards evaluating data gathering methods in manufacturing and mechanical engineering." *International Journal of Applied Science and Engineering Research*, 15(2023), 349–362.

Basha, S. S., Dubey, S. R., Pulabaigari, V., and Mukherjee, S. (2020). "Impact of fully connected layers on performance of convolutional neural networks for image classification." *Neurocomputing*, 378, 112–119.

Çakıroğlu, M. A. and Süzen, A. A. (2020). "Assessment and application of deep learning algorithms in civil engineering." *El-Cezeri*, 7(2), 906–922.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). "Encoder-decoder with atrous separable convolution for semantic image segmentation." *Proceedings of the European conference on computer vision (ECCV)*, 801–818.

Chiou, Y.-C. (2010). "Intelligent segmentation method for real-time defect inspection system." *Computers in Industry*, 61(7), 646–658.

Csurka, G., Volpi, R., Chidlovskii, B., et al. (2022). "Semantic image segmentation: Two decades of research." *Foundations and Trends® in Computer Graphics and Vision*, 14(1-2), 1–162.

Ekiz, S. and Acar, U. (2025). "Improving building extraction from high-resolution aerial images: Error correction and performance enhancement using deep learning on the inria dataset." *Science Progress*, 108(1), 00368504251318202.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). "The pascal visual object classes (voc) challenge." *International Journal of Computer Vision*, 88(2), 303–338.

Ghosh, A., Jana, N. D., Mallik, S., and Zhao, Z. (2022). "Designing optimal convolutional neural network architecture using differential evolution algorithm." *Patterns*, 3(9).

Guo, Y., Liu, Y., Georgiou, T., and Lew, M. S. (2018). "A review of semantic segmentation using deep neural networks." *International journal of multimedia information retrieval*, 7, 87–93.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Huang, X. (2023). "High resolution remote sensing image classification based on deep transfer learning and multi feature network." *IEEE Access*, 11, 110075–110085.

Kamann, C. and Rother, C. (2020). "Benchmarking the robustness of semantic segmentation models." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8828–8838.

Long, J., Shelhamer, E., and Darrell, T. (2015). "Fully convolutional networks for semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3431–3440.

Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N. Y., Kainz, B., et al. (2018). "Attention u-net: Learning where to look for the pancreas." *arXiv preprint arXiv:1804.03999*.

Pereira, G. A. and Hussain, M. (2024). "A review of transformer-based models for computer vision tasks: Capturing global context and spatial relationships." *arXiv preprint arXiv:2408.15178*.

Rahnemoonfar, M., Chowdhury, T., and Murphy, R. (2023). "Rescuenet: a high resolution uav semantic segmentation dataset for natural disaster damage assessment." *Scientific data*, 10(1), 913.

Ronneberger, O., Fischer, P., and Brox, T. (2015). "U-net: Convolutional networks for biomedical image segmentation." *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, eds., Cham, Springer International Publishing, 234–241.

Rosenberg, H., Errett, N. A., and Eisenman, D. P. (2022). "Working with disaster-affected communities to envision healthier futures: A trauma-informed approach to post-disaster recovery planning." *International journal of environmental research and public health*, 19(3), 1723.

Schlemper, J., Oktay, O., Schaap, M., Heinrich, M., Kainz, B., Glocker, B., and Rueckert, D. (2019). "Attention gated networks: Learning to leverage salient regions in medical images." *Medical Image Analysis*, 53, 197–207.

Strudel, R., Garcia, R., Laptev, I., and Schmid, C. (2021). "Segmenter: Transformer for semantic segmentation." *Proceedings of the IEEE/CVF international conference on computer vision*, 7262–7272.

Van Dyk, D. A. and Meng, X.-L. (2001). "The art of data augmentation." *Journal of Computational and Graphical Statistics*, 10(1), 1–50.

Van Hulse, J., Khoshgoftaar, T. M., and Napolitano, A. (2007). "Skewed class distributions and mislabeled examples." *Seventh IEEE International Conference on Data Mining Workshops (ICDMW 2007)*, IEEE, 477–482.

Wang, W. and Ma, J. (2024). "A review: Applications of machine learning and deep learning in aerospace engineering and aero-engine engineering." *Advances in Engineering Innovation*, 6, 54–72.

Wang, W., Tang, C., Wang, X., and Zheng, B. (2022). "A vit-based multiscale feature fusion approach for remote sensing image segmentation." *IEEE Geoscience and Remote Sensing Letters*, 19, 1–5.

Wang, Y. (2024). "Overview of image segmentation methods based on deep learning." *Third International Conference on Electronic Information Engineering and Data Processing (EIEDP 2024)*, Vol. 13184, SPIE, 720–728.

Xu, Z., Dan, C., Khim, J., and Ravikumar, P. (2020). "Class-weighted classification: Trade-offs and robust approaches." *International conference on machine learning*, PMLR, 10544–10554.

Xue, J., Park, S., Mondal, W. U., Reia, S. M., Yao, T., and Ukkusuri, S. V. (2023). "Supporting post-disaster recovery with agent-based modeling in multilayer socio-physical networks." *arXiv preprint arXiv:2307.11464*.

Zhang, P., Dai, X., Yang, J., Xiao, B., Yuan, L., Zhang, L., and Gao, J. (2021a). "Multi-scale vision longformer: A new vision transformer for high-resolution image encoding." *Proceedings of the IEEE/CVF international conference on computer vision*, 2998–3008.

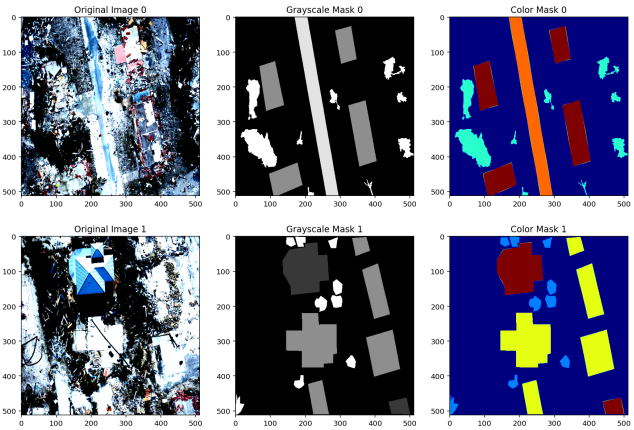
Zhang, W., Li, H., Li, Y., Liu, H., Chen, Y., and Ding, X. (2021b). "Application of deep learning algorithms in geotechnical engineering: a short critical review." *Artificial Intelligence Review*, 1–41.

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). "Pyramid scene parsing network." *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2881–2890.

7 Appendix

Detailed visual comparisons of ground truth and predicted outputs for each model, including PSPNet (Figure 6d), DeepLabV3+ (Figure 7), Attention U-Net (Figure 9), and Segmenter (Figure 8), are provided in this appendix.

(a) Ground truth image



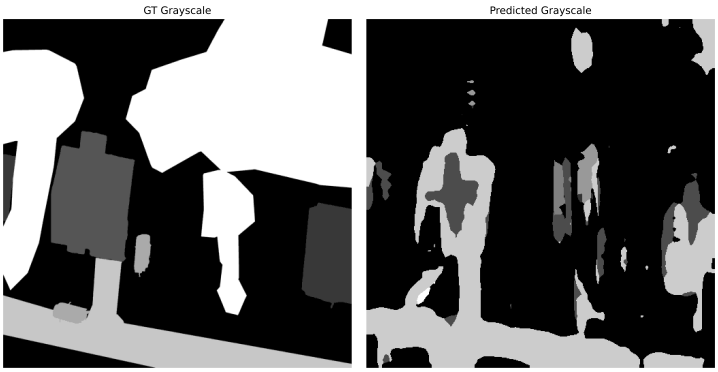
(b) Grayscale and color mask mapping

(c) Grayscale comparison: original vs. predicted

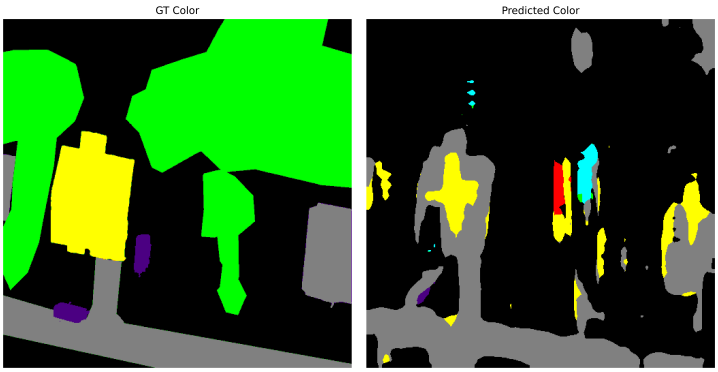
(d) Color mask comparison: original vs. predicted

Figure 6: PSPNet model predictions compared against ground truth and extracted masks.

(a) Ground truth image



(b) Grayscale comparison: original vs. predicted

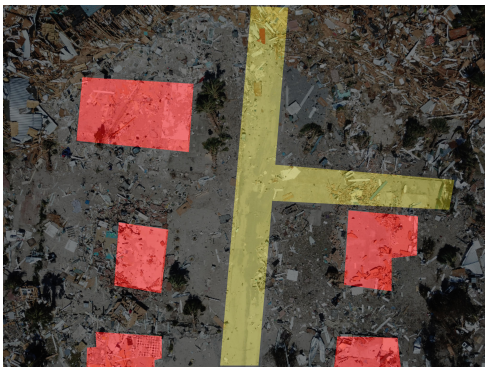


(c) Color mask comparison: original vs. predicted

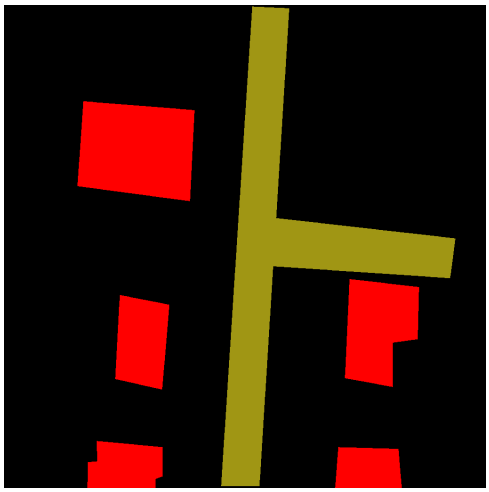
Figure 7: DeepLabV3+ model predictions compared to ground truth across grayscale and color mask outputs.

(a) Original image

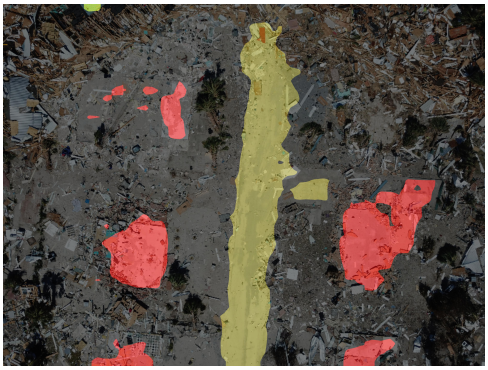
(b) Ground truth bounding box (grayscale)



(c) Predicted bounding box by Segmenter

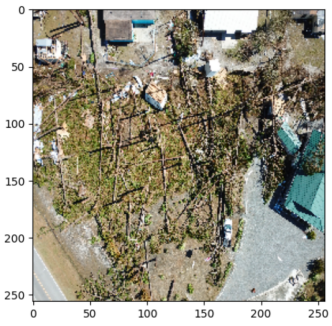


(d) Ground truth color mask

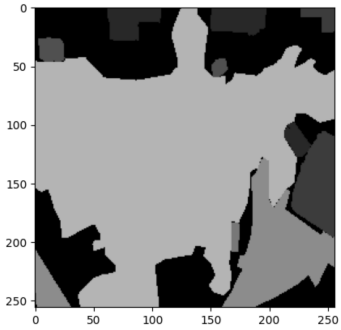


(e) Predicted color mask by Segmenter

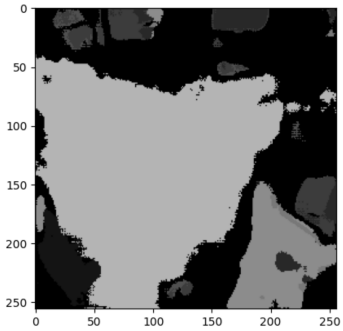
Figure 8: Segmenter model predictions compared to ground truth for grayscale bounding boxes and color masks.



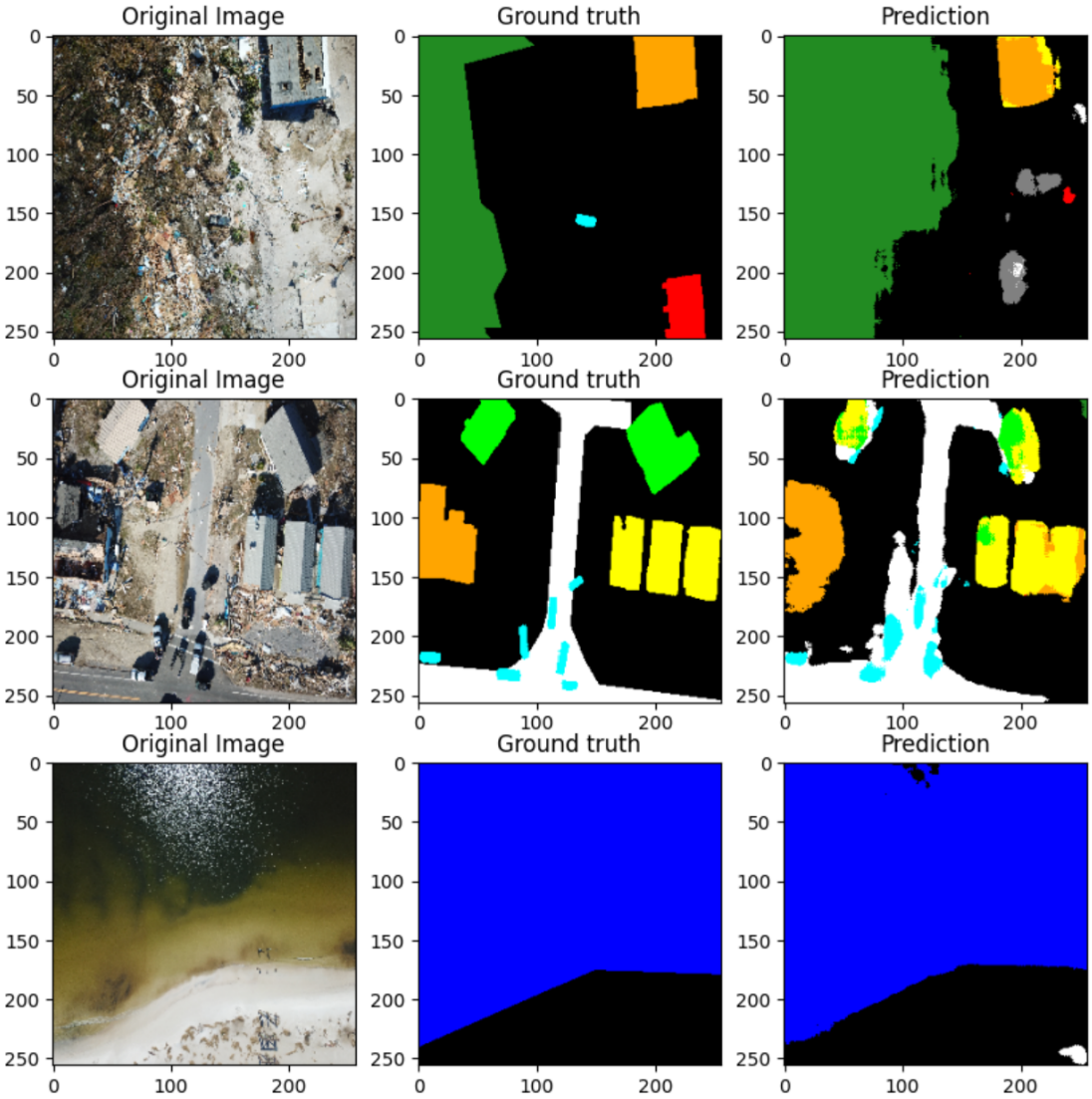
(a) Ground truth image



(b) Grayscale of ground truth



(c) Grayscale prediction



(d) Color masks comparison: original vs. predicted

Figure 9: Attention U-Net model predictions compared against ground truth across grayscale and color mask outputs.