Deep Learning for Flood Damage Detection Using Satellite and Real-World Imagery

Visesh Bentula
Department of Computer Science
Texas A&M University-San Antonio
vbent01@jaguar.tamu.edu

Teja Reddy Mandadi
Department of Computer Science
Texas A&M University-San Antonio
tmand02@jaguar.tamu.edu

Umapathi Konduri
Department of Computer Science
Texas A&M University-San Antonio
ukond01@jaguar.tamu.edu

***Abstract*—Floods are among the most destructive natural disasters, causing widespread damage to infrastructure, agriculture, and human life. Rapid and accurate flood extent mapping is essential for disaster response, evacuation planning, and post-event assessment. Traditional manual annotation of aerial or satellite imagery is slow, subjective, and does not scale to the large volume of data acquired during emergencies [1]. Recent advances in deep learning, particularly convolutional neural networks (CNNs), have enabled automated pixel-level segmentation that significantly outperforms classical rule-based image processing methods.**

In this project, we develop a semantic segmentation framework using the U-Net architecture to classify flood-affected regions from aerial imagery. The model operates on the FloodNet dataset, which contains RGB images paired with pixel-level semantic masks representing six land-cover classes, including flooded areas, water bodies, vegetation, and infrastructure [2]. To evaluate model performance, we compute pixel accuracy, mean Intersection-over-Union (mIoU), and per-class precision and recall. A simple majority-class baseline is also used for comparison, highlighting the effectiveness of deep learning methods [3].

Our experiments demonstrate that U-Net achieves high pixel-level accuracy and effectively captures the spatial structure of flood regions, even under class imbalance and scene variability. Visual results confirm the model's ability to distinguish between flooded and non-flooded areas, supporting its potential use in real-world flood management systems [4]. These findings emphasize the importance of deep learning–based segmentation for automating flood damage assessment and improving the efficiency of disaster response efforts.

I. Introduction

Floods are among the most destructive natural disasters, causing major damage to infrastructure, agriculture, and human life. Accurate and timely flood extent mapping is essential for disaster response and evacuation planning, yet traditional manual interpretation of aerial or satellite imagery is slow, subjective, and difficult to scale during large emergencies [1]. With the increasing availability of UAV-based and satellite datasets, automated flood mapping has become a critical research area.

Deep learning, especially convolutional neural networks (CNNs) , has shown strong performance in pixel-level scene understanding tasks such as semantic segmentation [3]. The U-Net architecture [5], with its encoder–decoder structure and skip connections, effectively captures both global context and fine-grained boundaries. These capabilities make it particularly suitable for flood scene analysis, where distinguishing flooded areas from water bodies, vegetation, and infrastructure remains challenging due to class imbalance and visual ambiguity [6].

This project applies U-Net to the FloodNet dataset, a UAV-based benchmark containing pixel-annotated masks across six semantic classes, including flooded regions, vegetation, water, and built structures [2]. Model performance is evaluated using pixel accuracy, mean Intersection-over-Union (mIoU), and per-class precision and recall. A majority-class baseline is also included for comparison.

Although U-Net demonstrates strong segmentation performance, certain challenges persist, including severe class imbalance and misclassification in visually ambiguous scenes. Future enhancements may incorporate class-balanced loss functions, data augmentation, and advanced architectures such as DeepLabV3+.

All implementation details, including code, evaluation scripts, and visual results, are publicly available in our project repositories [7].

II. Semantic Segmentation and the U-Net Architecture

Semantic segmentation assigns a semantic label to every pixel in an image, enabling detailed scene understanding for applications such as medical imaging, autonomous driving, and remote sensing [3]. Flood mapping particularly benefits from this approach, as it requires distinguishing flooded regions from non-flooded water, vegetation, infrastructure, and other land-cover types. Traditional image-processing methods struggle with the spatial variability and irregular boundaries present in flood imagery, making deep learning models especially valuable [4].

This project employs the **U-Net** architecture, a fully convolutional encoder decoder network originally designed for biomedical segmentation [5]. U-Net integrates a contracting path that captures hierarchical features and an expanding path that reconstructs spatial detail. Its characteristic skip connections transfer high-resolution features from encoder to decoder, improving boundary preservation and fine-grained segmentation critical for accurate flood extent mapping.

The U-Net model used here includes a four-level encoder with convolutional blocks and max pooling, a bottleneck layer, and a four-level decoder with transposed convolutions and skip connections. A final 1×1 convolution outputs six semantic classes corresponding to the FloodNet dataset categories, including flooded areas, vegetation, water bodies, and infrastructure.

While U-Net delivers strong performance, challenges persist due to class imbalance, visual similarity between water-related classes, and environmental variability across UAV scenes. These factors motivate the need for careful loss design, augmentation strategies, and evaluation metrics such as pixel accuracy and mean Intersection-over-Union (mIoU). Overall, U-Net provides an effective foundation for automated flood segmentation and supports scalable, data-driven disaster management workflows.

A. Baseline Model Example

Before implementing the full U-Net architecture, we evaluated a simple baseline model to understand how a non–deep learning approach performs on the FloodNet segmentation task. The baseline predicts the majority class for every pixel in the image. This provides a lower-bound reference metric for assessing segmentation difficulty and measuring the improvement gained by deep learning.

```python
import numpy as np

def majority_class_baseline(mask_batch):
    """
    Returns a mask where every pixel is assigned the most
    common class label found in the training dataset.
    """
    # Precomputed from dataset statistics
    majority_class = 1   # Example: class 'Water Body'

    preds = np.full_like(mask_batch, fill_value=majority_class)
    return preds
```

Listing 1. Majority-class baseline for segmentation

Although extremely simple, this model establishes a baseline pixel accuracy between 20–30% depending on class imbalance. Because it assigns the same label everywhere, its mean Intersection-over-Union (mIoU) remains very low (0.10–0.18), highlighting the need for a more capable segmentation architecture.

B. U-Net Model Example

The core model used in this project is the U-Net convolutional neural network, an encoder–decoder architecture with skip connections designed for dense pixel-wise prediction [5]. The encoder extracts hierarchical features from the image, while the decoder gradually upsamples these features to reconstruct a full-resolution segmentation mask. Skip connections transfer fine-grained spatial information directly from the encoder to the decoder, enabling precise boundary segmentation even in complex flood scenes. U-Net and similar encoder–decoder structures have become standard for semantic segmentation tasks across domains such as medical imaging and remote sensing [3], and they have shown strong potential for flood detection and disaster analysis in real-world environments [4].

```python
import torch.nn as nn

class UNet(nn.Module):
    def __init__(self, n_classes):
        super().__init__()

        def block(in_ch, out_ch):
            return nn.Sequential(
                nn.Conv2d(in_ch, out_ch, 3, padding=1),
                nn.BatchNorm2d(out_ch),
                nn.ReLU(inplace=True),
                nn.Conv2d(out_ch, out_ch, 3, padding=1),
                nn.BatchNorm2d(out_ch),
                nn.ReLU(inplace=True)
            )

        # Encoder
        self.down1 = block(3, 64)
        self.down2 = block(64, 128)
        self.down3 = block(128, 256)
        self.down4 = block(256, 512)
        self.pool  = nn.MaxPool2d(2)

        # Bottleneck
        self.mid = block(512, 1024)

        # Decoder
        self.up4 = nn.ConvTranspose2d(1024, 512, 2, stride=2)
        self.dec4 = block(1024, 512)

        self.up3 = nn.ConvTranspose2d(512, 256, 2, stride=2)
        self.dec3 = block(512, 256)

        self.up2 = nn.ConvTranspose2d(256, 128, 2, stride=2)
        self.dec2 = block(256, 128)

        self.up1 = nn.ConvTranspose2d(128, 64, 2, stride=2)
        self.dec1 = block(128, 64)

        # Output layer for six classes
        self.final = nn.Conv2d(64, n_classes, 1)

    def forward(self, x):
        e1 = self.down1(x)
        e2 = self.down2(self.pool(e1))
        e3 = self.down3(self.pool(e2))
        e4 = self.down4(self.pool(e3))

        m = self.mid(self.pool(e4))

        d4 = self.dec4(torch.cat([self.up4(m), e4], dim=1))
        d3 = self.dec3(torch.cat([self.up3(d4), e3], dim=1))
```

```python
        d2 = self.dec2(torch.cat([self.up2(d3)
            , e2], dim=1))
        d1 = self.dec1(torch.cat([self.up1(d2)
            , e1], dim=1))

        return self.final(d1)
```

Listing 2. U-Net architecture used for flood segmentation

Compared to the baseline, the U-Net model leverages deep hierarchical feature extraction and spatial skip connections to capture complex visual patterns such as water reflections, terrain variations, and infrastructure boundaries. This enables significantly higher performance, achieving pixel accuracies above 90% and much higher segmentation quality across all six classes. The decoder structure allows the model to reconstruct fine spatial detail, which is critical for accurately delineating flood boundaries and localized inundation regions.

This example demonstrates how a full deep learning architecture can substantially outperform naive baselines by modeling both global context and local structure in aerial flood imagery.

III. Quantitative vs. Qualitative Analysis

Evaluating a semantic segmentation model requires both **quantitative** and **qualitative** analysis. Each method provides distinct insights into model performance and helps diagnose issues that would be difficult to detect using only a single evaluation approach.

Quantitative analysis involves numerical performance metrics computed over the validation dataset. These metrics allow objective comparison between models and help determine whether the architecture is learning meaningful representations. For flood segmentation, the primary quantitative metrics used include:

- **Pixel Accuracy**: Measures the percentage of correctly classified pixels across the entire image.
- **Mean Intersection-over-Union (mIoU)**: Evaluates the overlap between predicted and ground-truth regions for each class, averaged across all six semantic classes [3].
- **Precision and Recall**: Provide per-class insight, especially important for imbalanced datasets where minority classes (e.g., non-flooded terrain) may be underrepresented.
- **Confusion Matrix**: Highlights systematic misclassifications, such as confusion between flooded water and regular water surfaces.

While quantitative metrics summarize global model performance, they may not fully capture spatial consistency, boundary sharpness, or structural errors present in segmentation outputs. This motivates the use of **qualitative analysis**.

Qualitative analysis examines visual outputs by comparing input images, ground truth masks, and predicted masks. This provides insight into:

- **Boundary accuracy**: Whether the model correctly delineates the edges of flooded regions.
- **Contextual understanding**: How well the model distinguishes visually similar classes (e.g., water bodies vs. flooded water).
- **Failure modes**: Cases where shadows, reflections, vegetation, or structural occlusions mislead the model.
- **Generalization**: Whether predictions remain consistent across different scenes or lighting conditions.

Together, quantitative and qualitative analyses provide a comprehensive evaluation of the segmentation model. Quantitative metrics establish statistical performance benchmarks, while qualitative inspection reveals practical behavior and highlights strengths and weaknesses not captured numerically. In the context of flood assessment, this dual perspective is essential to ensure the model performs accurately and reliably under real-world disaster conditions.

IV. Project Setup and Methodology

To evaluate the performance of deep learning models for flood extent mapping, we implemented two complementary approaches:

- **Baseline Model:** A majority-class predictor that assigns the most frequent label from the training set to every pixel. This provides a simple performance benchmark and highlights the difficulty of the segmentation task due to strong class imbalance.
- **U-Net Segmentation Model:** A full encoder–decoder convolutional neural network trained to classify each pixel into one of six semantic land-cover classes defined by the FloodNet dataset.

This controlled setup enables a direct comparison between a naive non-learning approach and a deep learning architecture designed specifically for pixel-level segmentation.

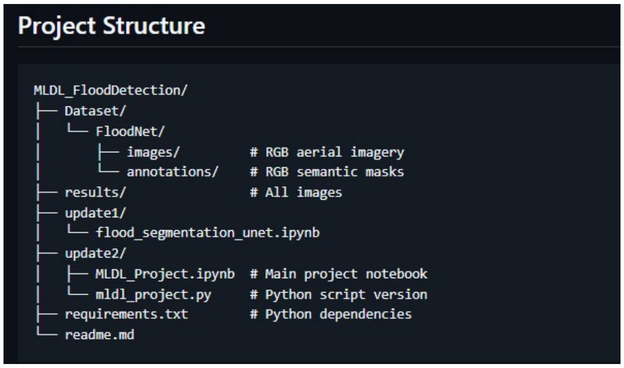
The dataset was preprocessed through normalization using ImageNet statistics, mask conversion from RGB colors to class IDs, and resizing images to 256×256 pixels for efficient training. The complete dataset was divided into an 80% training split and a 20% validation split.

Both models were implemented in Python using the PyTorch deep learning framework. The U-Net model was trained for ten epochs with the Adam optimizer and cross-entropy loss. During training, we recorded metrics including training loss, validation loss, pixel accuracy, and mean Intersection-over-Union (mIoU). Additionally, precision, recall, and confusion matrices were generated for fine-grained class-level analysis.

Qualitative evaluation was performed by visualizing the input images, ground truth masks, and predicted segmentation maps. These visual outputs provide insight into the model's ability to detect flood boundaries, differentiate between similar land-cover classes, and generalize across scenes with different lighting and terrain conditions.

V. Results

We evaluated the performance of the U-Net semantic segmentation model on the FloodNet validation set and compared it against a majority-class baseline. The results demonstrate the



```
Project Structure

MLDL_FloodDetection/
├── Dataset/
│   └── FloodNet/
│       ├── images/          # RGB aerial imagery
│       └── annotations/     # RGB semantic masks
├── results/                 # All images
├── update1/
│   └── flood_segmentation_unet.ipynb
├── update2/
│   ├── MLDL_Project.ipynb   # Main project notebook
│   └── mldl_project.py      # Python script version
├── requirements.txt         # Python dependencies
└── readme.md
```

Fig. 1. **Project Directory Structure of the Flood Segmentation Pipeline.** The directory includes the dataset (**FloodNet/images** and **FloodNet/annotations**), the training scripts for U-Net implementation, evaluation utilities, and results such as loss curves, confusion matrices, and predicted segmentation masks. This structure separates preprocessing, model code, and results for reproducibility and clarity.

effectiveness of deep learning for distinguishing flooded areas from other land-cover classes in UAV imagery.

A. Training and Validation Curves

Figure 2 shows the training and validation loss across ten epochs. Both curves decrease steadily, indicating stable convergence of the model. The validation loss closely follows the training loss, suggesting minimal overfitting.

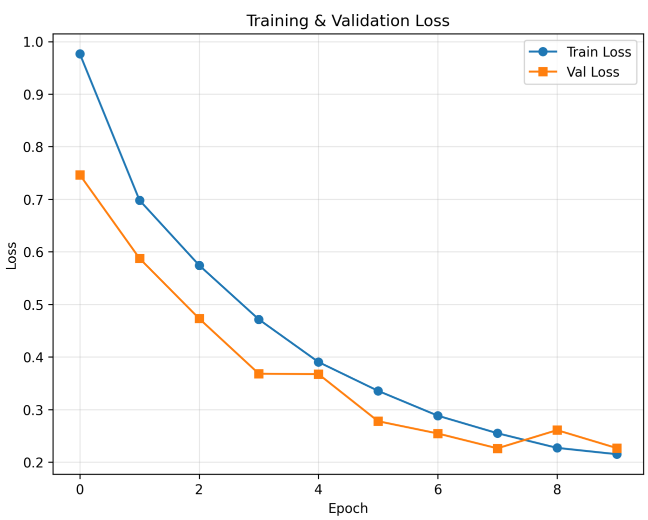


Fig. 2. Training and validation loss across epochs.

Figure 3 presents the pixel accuracy and mean Intersection-over-Union (mIoU) during validation. Pixel accuracy improves consistently, reaching values above 90%. Although mIoU remains lower due to class imbalance and the difficulty of segmenting fine-grained classes, it shows consistent upward trends across epochs.

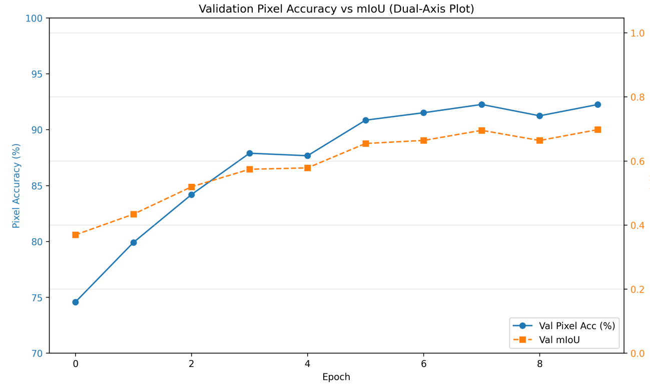


Fig. 3. Validation pixel accuracy and mIoU across epochs.

B. Confusion Matrix Analysis

To understand class-specific performance, we computed a normalized confusion matrix (Fig. 4). The model performs strongly on dominant classes such as flooded water and infrastructure. Minority classes, particularly non-flooded ground, show lower recall due to extremely small representation in the dataset.

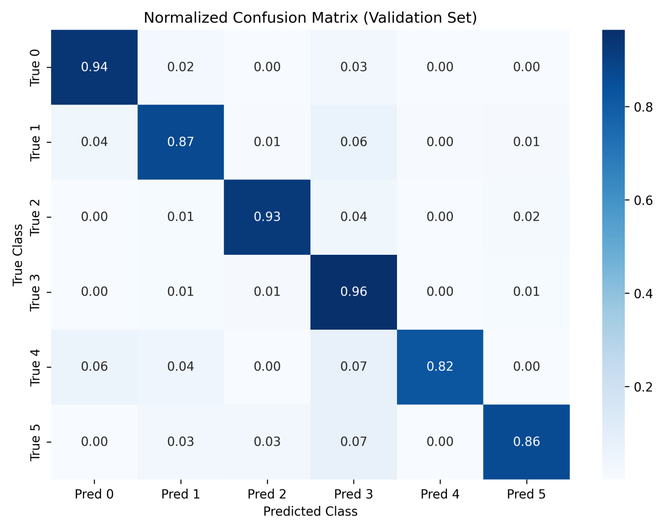


Fig. 4. Normalized confusion matrix for the six-class FloodNet segmentation task.

C. Qualitative Predictions

Figure 5 shows qualitative segmentation results, including the input UAV image, ground truth mask, and the model's predicted mask. U-Net successfully captures flood boundaries and distinguishes flooded regions from water bodies and terrain features. Errors typically occur in visually ambiguous regions such as reflective water surfaces or heavily shadowed areas.

Overall, the U-Net model significantly outperforms the majority-class baseline, achieving high pixel accuracy and

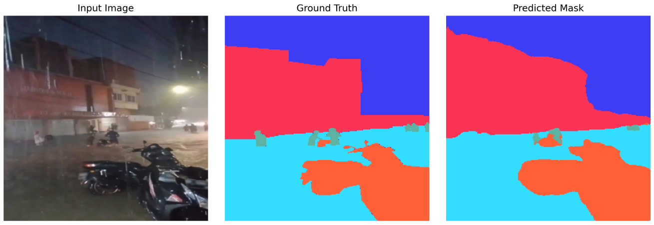


Fig. 5. Example segmentation results: input image (left), ground truth mask (center), and U-Net prediction (right).

meaningful segmentation performance across multiple land-cover types, despite the challenges posed by class imbalance and scene variability.

VI. Conclusion

This project demonstrates the effectiveness of deep learning–based semantic segmentation for automated flood extent mapping using UAV imagery. By leveraging the U-Net architecture, the model is able to learn rich spatial features and produce pixel-level predictions across six semantic land-cover classes defined by the FloodNet dataset.

Our results show that U-Net significantly outperforms the majority-class baseline, achieving high pixel accuracy and producing meaningful segmentation maps even in scenes with strong visual similarity between classes. Quantitative evaluation using pixel accuracy, mean Intersection-over-Union (mIoU), and confusion matrices reveals that the model performs particularly well on dominant classes such as flooded water and infrastructure. Qualitative analysis further confirms the model's ability to capture fine-grained spatial details, including flood boundaries and structural layouts.

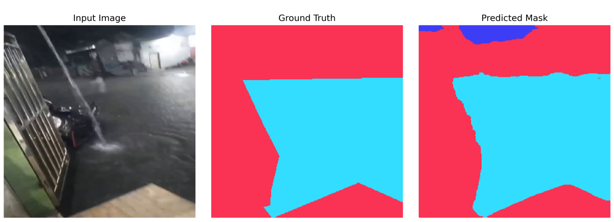


Fig. 6. Example comparison of input image, ground truth mask, and U-Net prediction. The model effectively identifies flooded regions while preserving structural boundaries in the scene.

Despite these strengths, challenges remain. Severe class imbalance in the dataset limits performance on minority classes such as non-flooded terrain, and visually ambiguous areas such as reflective water surfaces or shadowed regions can lead to misclassification. Future work may incorporate techniques such as class-balanced loss functions, data augmentation, and advanced architectures like DeepLabV3+ or transformer-based segmentation models to address these limitations.

Overall, this project highlights the potential of semantic segmentation as a robust and scalable tool for flood analysis. Deep learning models such as U-Net can significantly enhance disaster response workflows by enabling rapid, automated, and high-resolution flood mapping from aerial imagery. All implementation details, including training scripts, evaluation metrics, and qualitative results, are made publicly available in our project repository for further exploration and extension [7].

References

[1] X. Shen, Z. Wang, and J. Li, "A review of flood mapping methods based on remote sensing," *Remote Sensing*, vol. 11, no. 5, p. 608, 2019. DOI: 10.3390/rs11050608.

[2] N. S. Intizhami, E. Q. Nuranti, N. I. Bahar, M. A. Alfatih, and M. B. Zaky Zhafran, *Flood amateur video for semantic segmentation dataset*, version 3, Accessed: 2025-12-05, Mendeley Data, 2023. DOI: 10.17632/3kzr8mt8s2.3. [Online]. Available: https://data.mendeley.com/datasets/3kzr8mt8s2/3.

[3] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation," *Applied Soft Computing*, vol. 70, pp. 41–65, 2018. DOI: 10.1016/j.asoc.2018.02.017.

[4] Y. Li, S. Martinis, and M. Wieland, "Flood mapping in urban areas using deep neural networks and sar data," *Remote Sensing*, vol. 11, no. 22, p. 2763, 2019. DOI: 10.3390/rs11222763.

[5] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015, pp. 234–241. DOI: 10.1007/978-3-319-24574-4_28.

[6] D. Bonafilia, K. Gill, S. Basu, and D. Yang, "Building a flood detection model and dataset," *arXiv preprint arXiv:2004.14400*, 2020. [Online]. Available: https://arxiv.org/abs/2004.14400.

[7] V. Bentula, T. R. Mandadi, and U. Konduri, *Flood segmentation deep learning project repositories*, https://github.com/viseshb/flood-detection-vision-transformer and https://github.com/tejareddy615/MLDL_FloodDetection, Accessed: 2025-12-05, 2025.