MAJOR PROJECT PRESENTATION

8th Semester

Project Supervisor:- Prof. Sulata Mitra

GROUP MEMBERS

2021CSB114 Vaishnavi Malviya

2021CSB105 Samriddhi Mishra

2021CSB102 Umang Waghmare

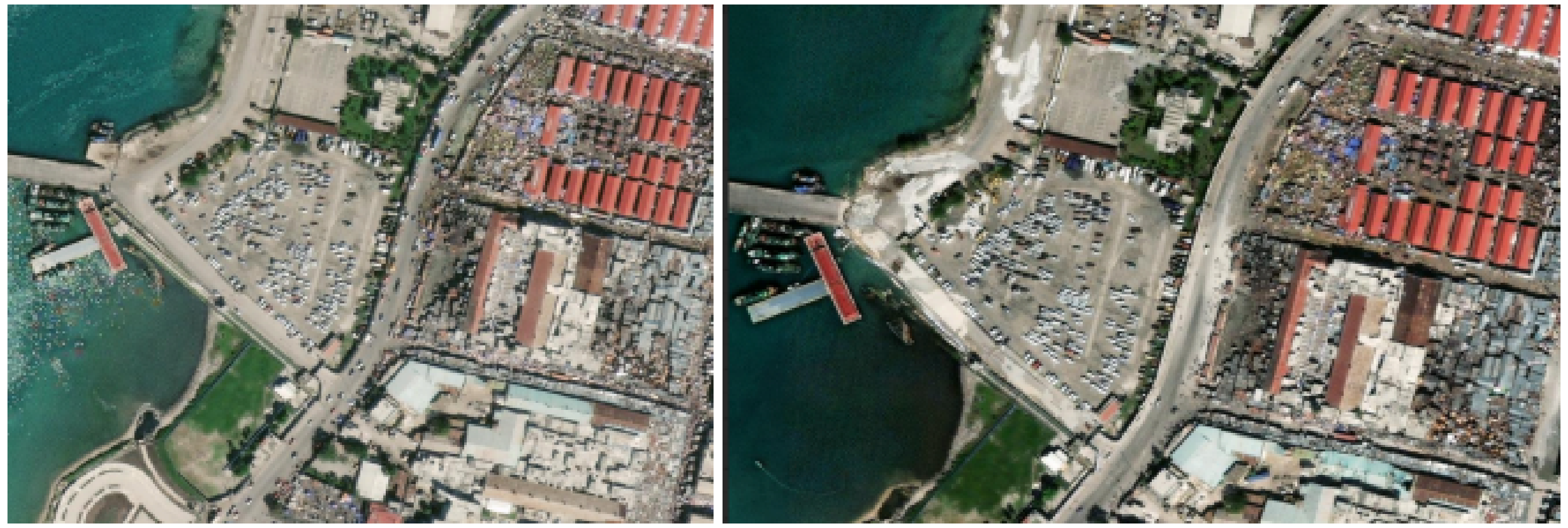
2021CSB092 Nihal Choutapelly

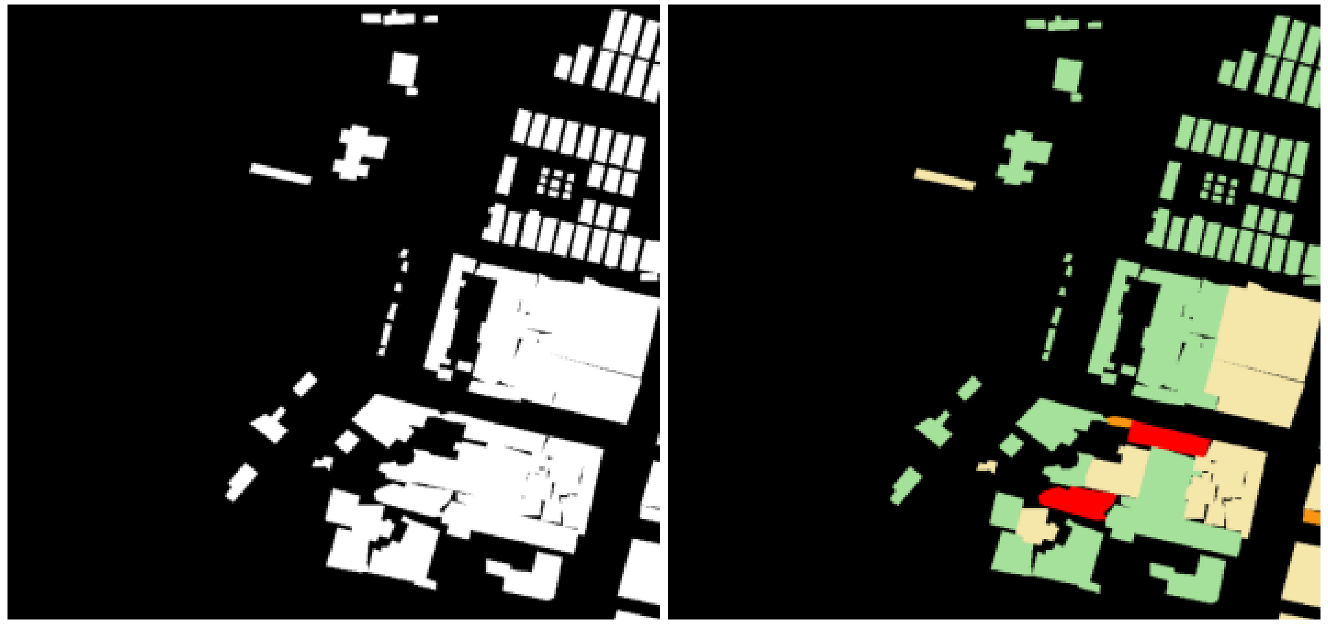
2021CSB086 Sujit Jaiswal

2021CSB055 Ankita Das

2021CSB035 Priya Samanta

POST-DISASTER CRITICAL AREA IDENTIFICATION USING IMAGE PROCESSING

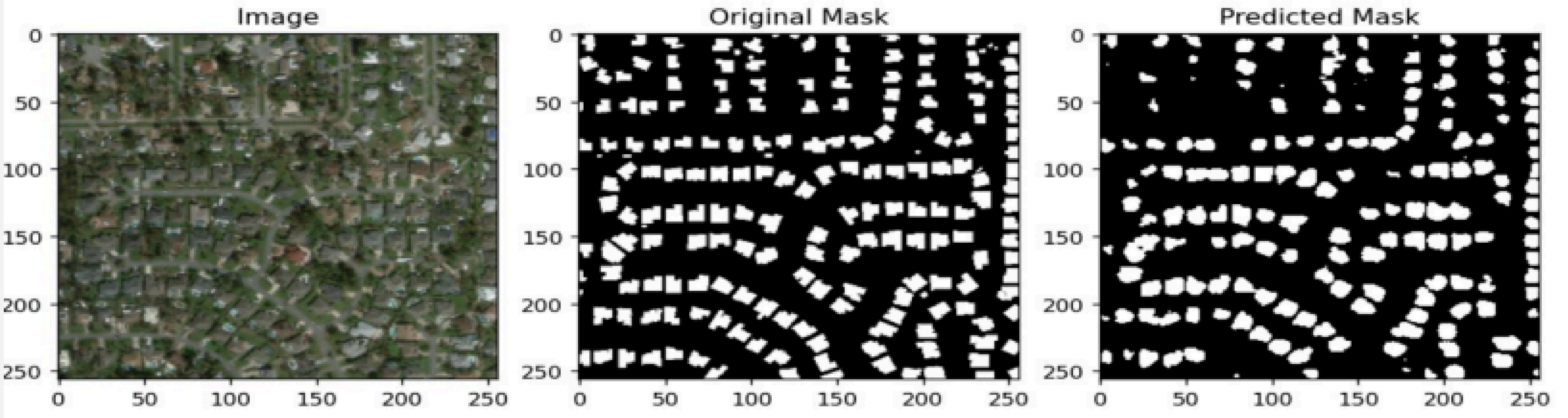




RECAPTULATION

- The ground truth images of post disaster images were predicted and compared with the intial labelled data to assess the efficiency of the model.
- Several models were employed to determine the buildings in an image for an accurate representation.

PREVIOUS RESULT



DATASET

- The xView2 dataset, developed by the Defense Innovation Unit and Carnegie Mellon University, is a large-scale geospatial dataset designed for assessing building damage from natural disasters using satellite imagery. It contains pre- and post-disaster high-resolution images across multiple global disaster events (e.g., earthquakes, hurricanes, floods).
- Pre-event images (taken before the disaster),
- Post-event images (taken after the disaster),
- Labeled geoJSON files providing building footprints with corresponding damage classification levels: no damage, minor damage, major damage, or destroyed.

DATA BALANCING

- **Class Imbalance Issue:** Many images in the xView2 dataset contain only no damage labels, leading to a heavily imbalanced dataset across damage classes.
- **Model Bias Risk**: The dominance of the no damage class can cause the model to favor the majority class, reducing its ability to accurately detect damaged structures.
- **Balancing Strategy:** The dataset was balanced through data augmentation (on underrepresented damage classes) and cleaning (removal of irrelevant or low-quality non-damage samples).

DATA AUGMENTATION - I

Applied **data augmentation techniques** such as:

- Rotation
- Horizontal/Vertical flipping
- Zoom in/Zoom out
- A mix of all of them

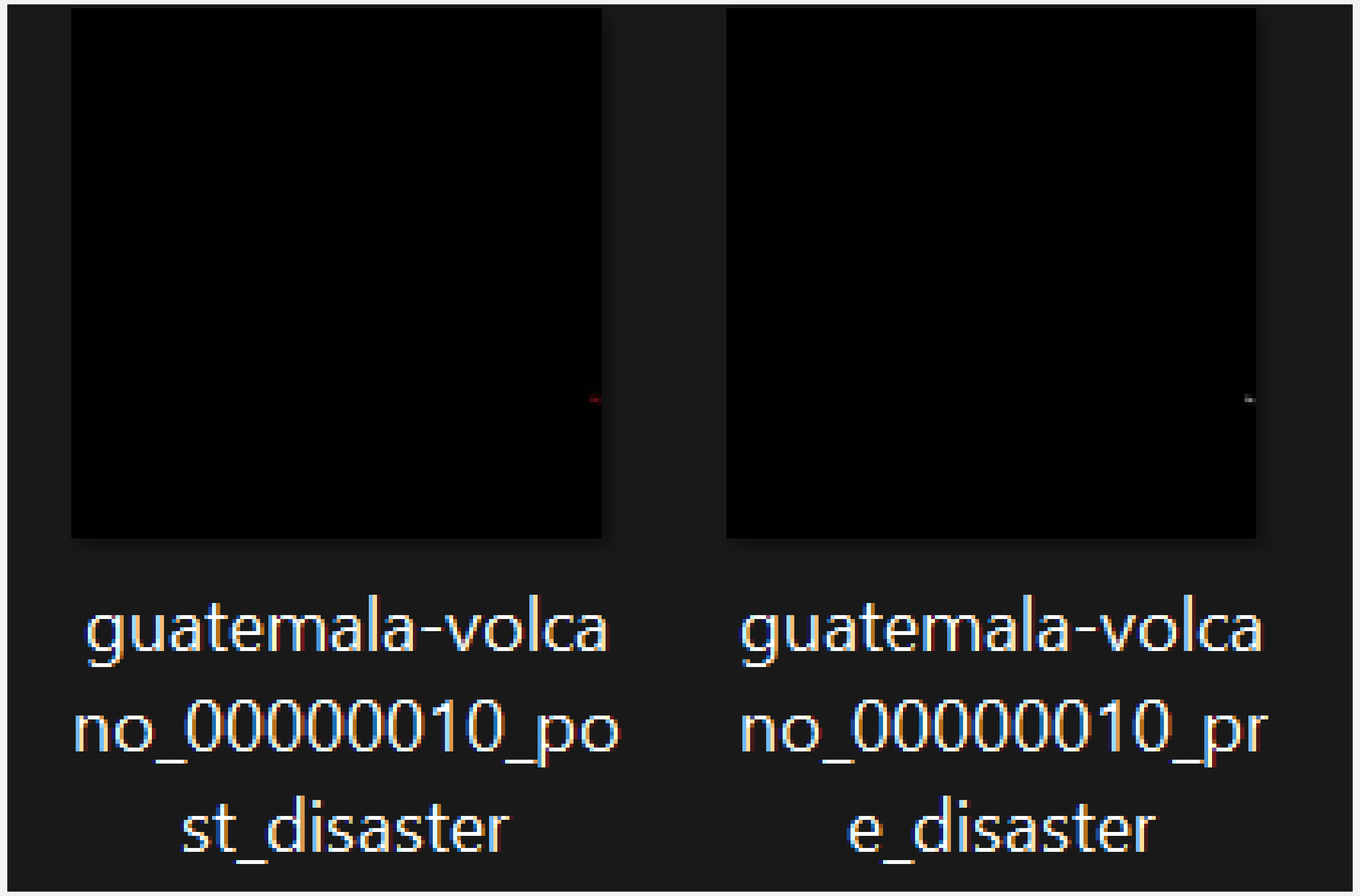
Augmentation was applied probabilistically (50% randomly chosen), allowing the generation of multiple variations per image without overfitting.

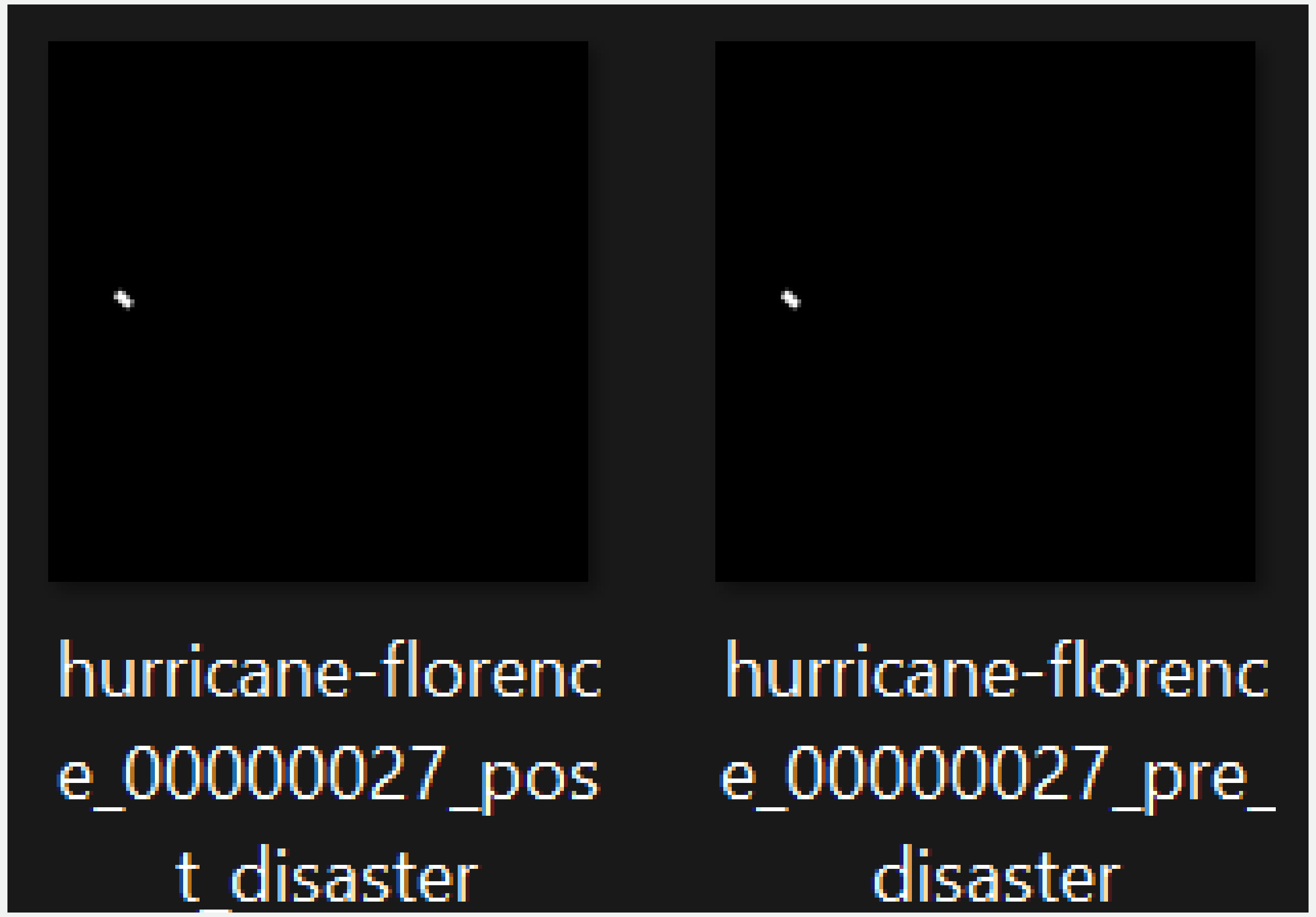
REMOVING NON-INFORMATIVE SAMPLES

Objective: To remove non-informative samples and further reduce class imbalance.

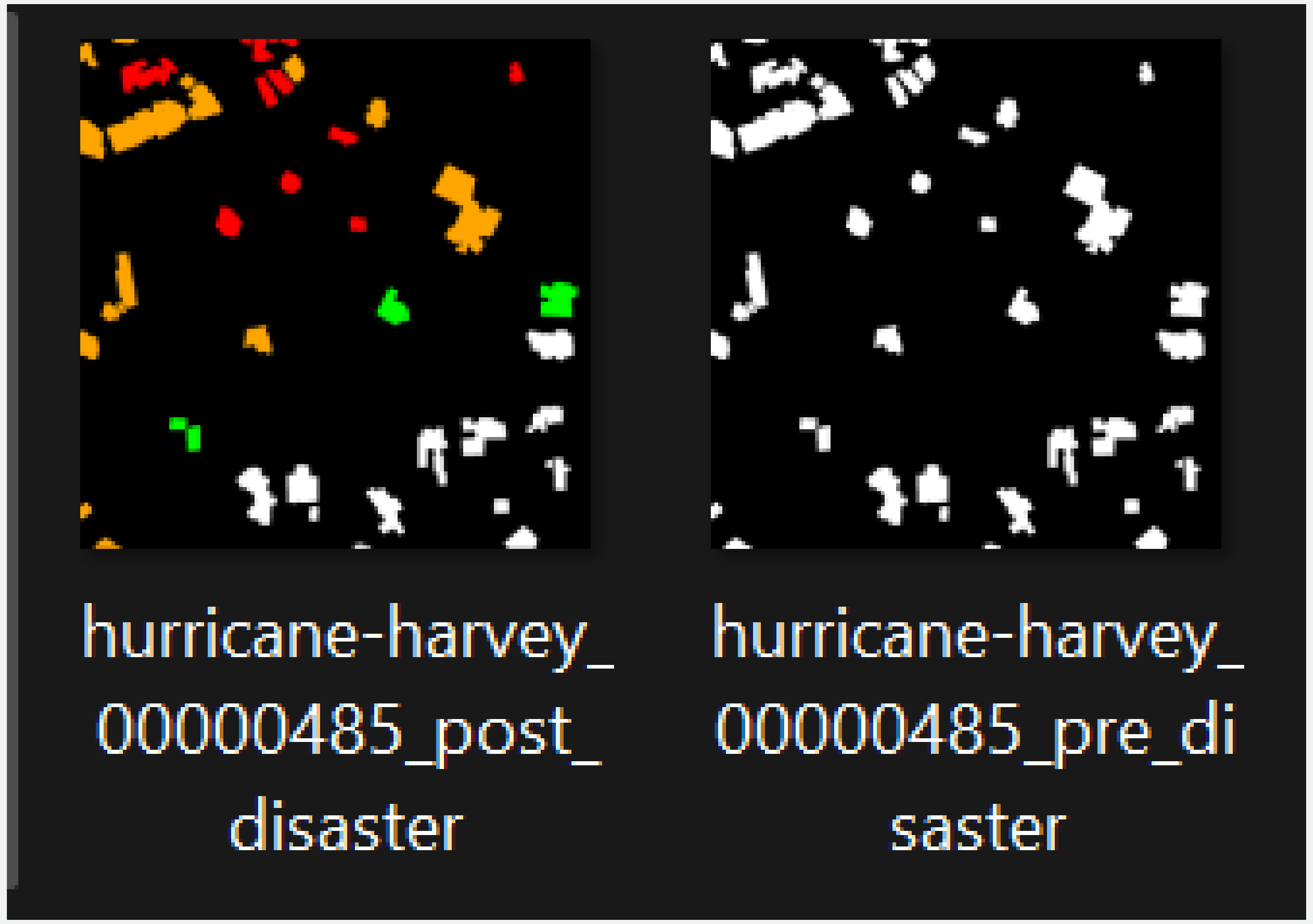
- After augmentation, a cleaning step was performed to filter out irrelevant or empty data.
- Images with less than 0.25% white pixel coverage (indicating absence of meaningful structure or damage) were removed.
- This ensured the model trained only on images with visible and annotated building structures, improving learning efficiency and reducing noise.

VISUAL REPRESENTATION





BALANCED SAMPLE



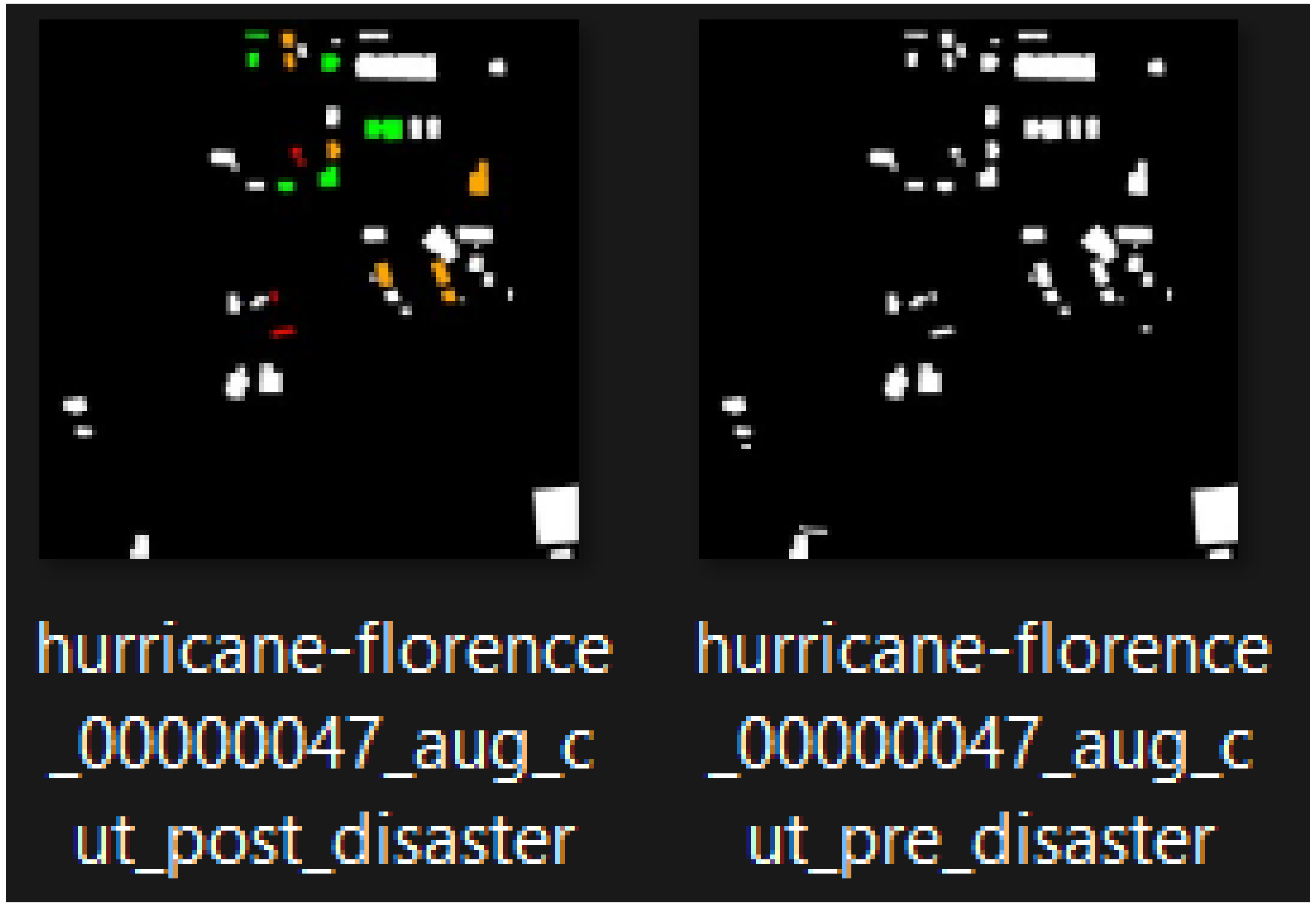
LIMITATIONS

- A key drawback of the augmentation process was that black (empty) regions of the images were sometimes amplified, especially during zoom and flip operations.
- This introduced non-informative artifacts, reducing the model's ability to focus on actual building structures.
- To mitigate this, we applied CutMix, which merges informative regions from multiple images to preserve relevant features and improve generalization.

SELECTIVE CLEANING

- Selective cleaning was applied to **remove non-informative samples** from the dataset.
- Images where the label masks contained only white (i.e., no damage class) were discarded.
- We retained only those images where the labels included damage indicators: **red** (destroyed), **orange** (major), or **green** (minor), either **individually or in combination.**
- This ensured the model was trained on samples with a**ctual damage variation**, improving its ability to estimate and classify damage more accurately.

DESIRED DATA SAMPLES



To sustain images that have buildings and damage is done with different levels

CUTMIX

- CutMix augmentation was used to merge patches from both random and difficult samples, enhancing class diversity.
- Combined pre- and post-disaster images and their label masks to retain spatial damage context.
- Helped reduce the influence of black/empty regions and improved learning on hard-to-classify damage areas.

CUTMIX

- CutMix blends the features and labels of two images, encouraging the model to predict based on mixed information. This regularizes the model by preventing over-reliance on a single region, improving generalization.
- In damage classification, this reduces local overfitting to undamaged or homogenous regions.

VISUAL WORKTHROUGH

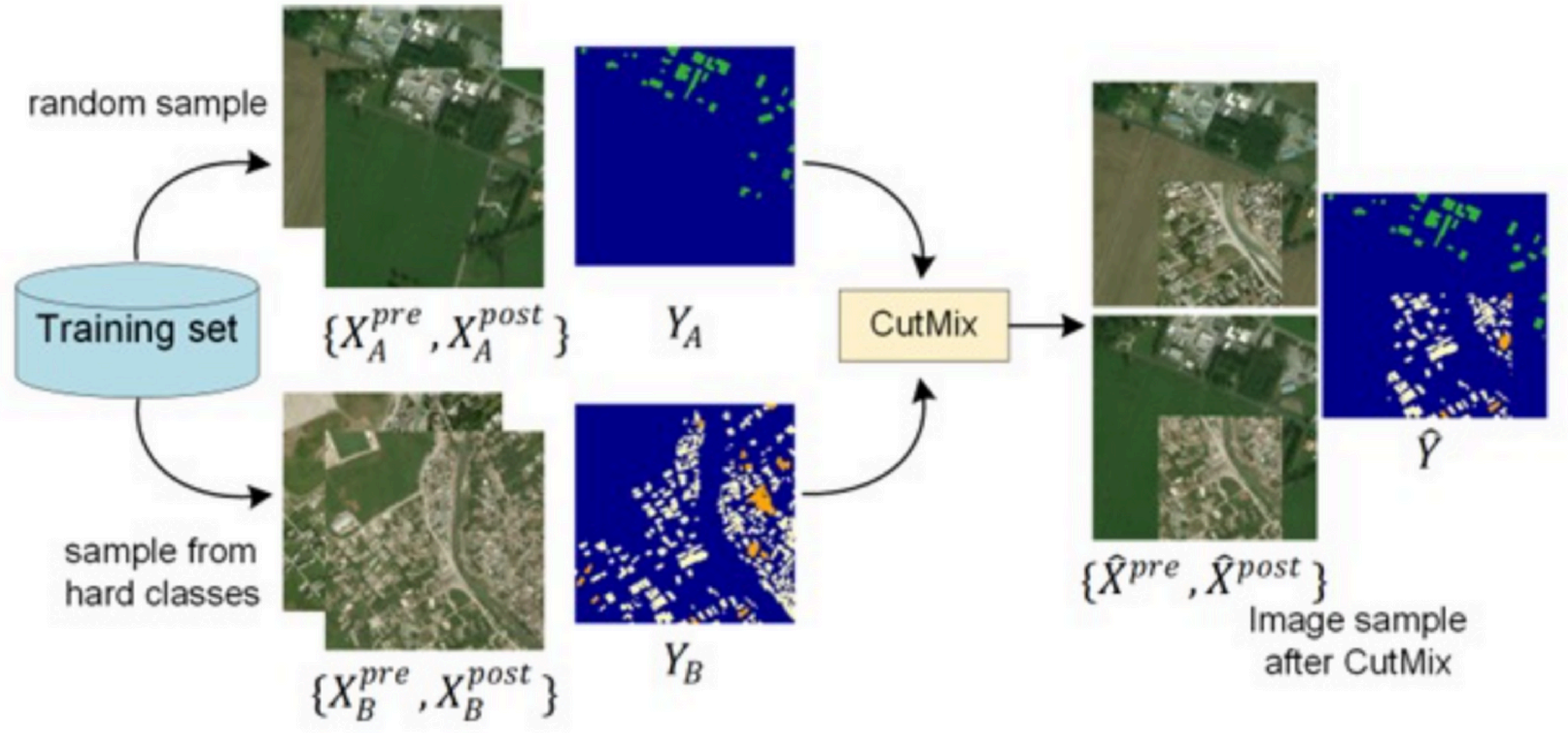


Fig. 4: Data augmentation with CutMix for difficult classes.

CUTMIX

Benefits of CutMix -

Balances Difficult Classes: By increasing the sample size of damage classes like minor and major, CutMix helps the model not to overfit to dominant classes (e.g., undamaged).
Improves Generalization: The model learns to recognize composite scenes, preparing it to detect multiple damage levels in real-world post-disaster images.

CUTMIX

Enhances Robustness: CutMix adds structural and contextual variation to training data, forcing the model to become more invariant to location, structure, and object blending.

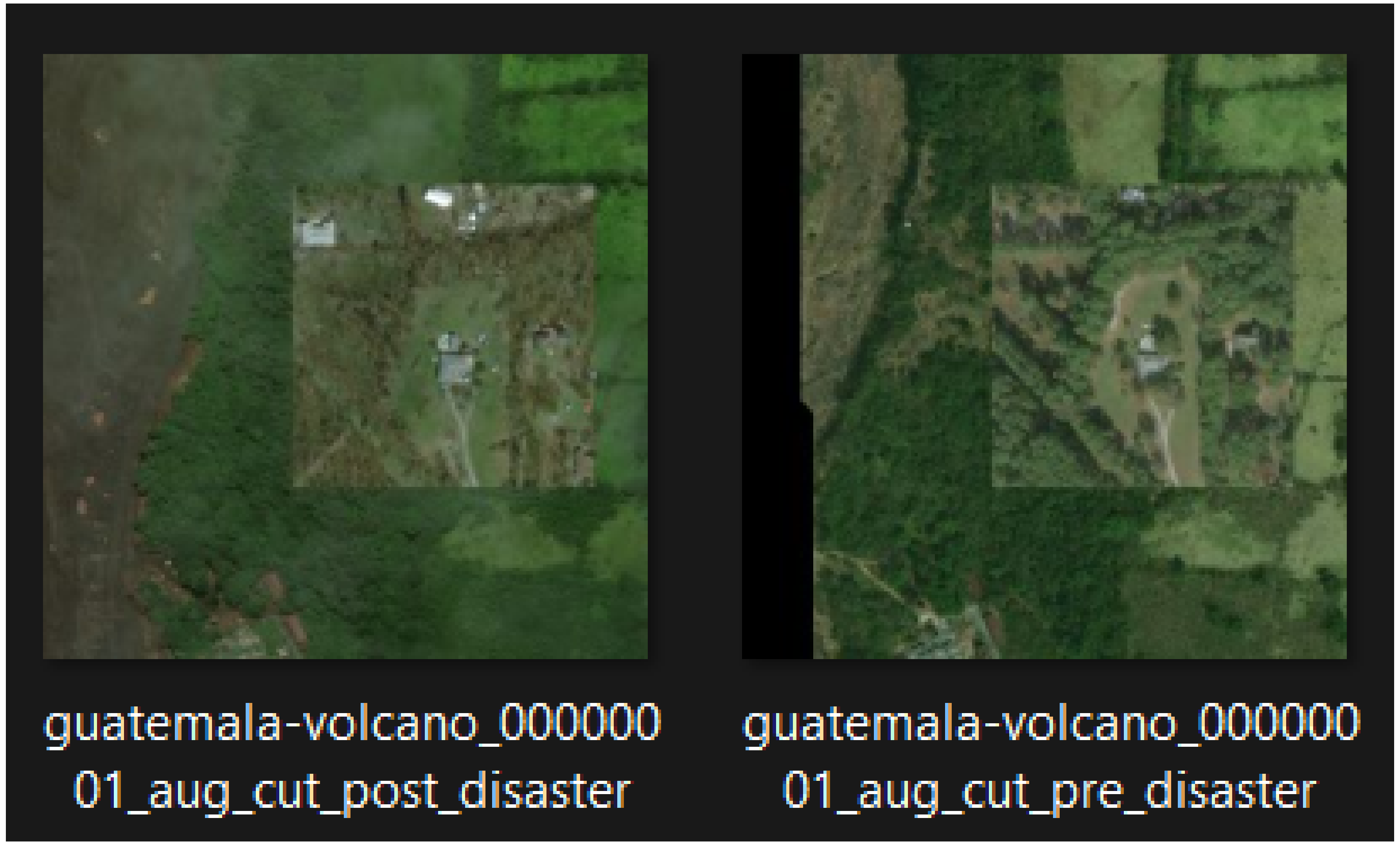
Realistic Simulation: Multiple damage zones in a single image mimic complex real-world post-disaster situations better than traditional augmentations.

CUTMIX

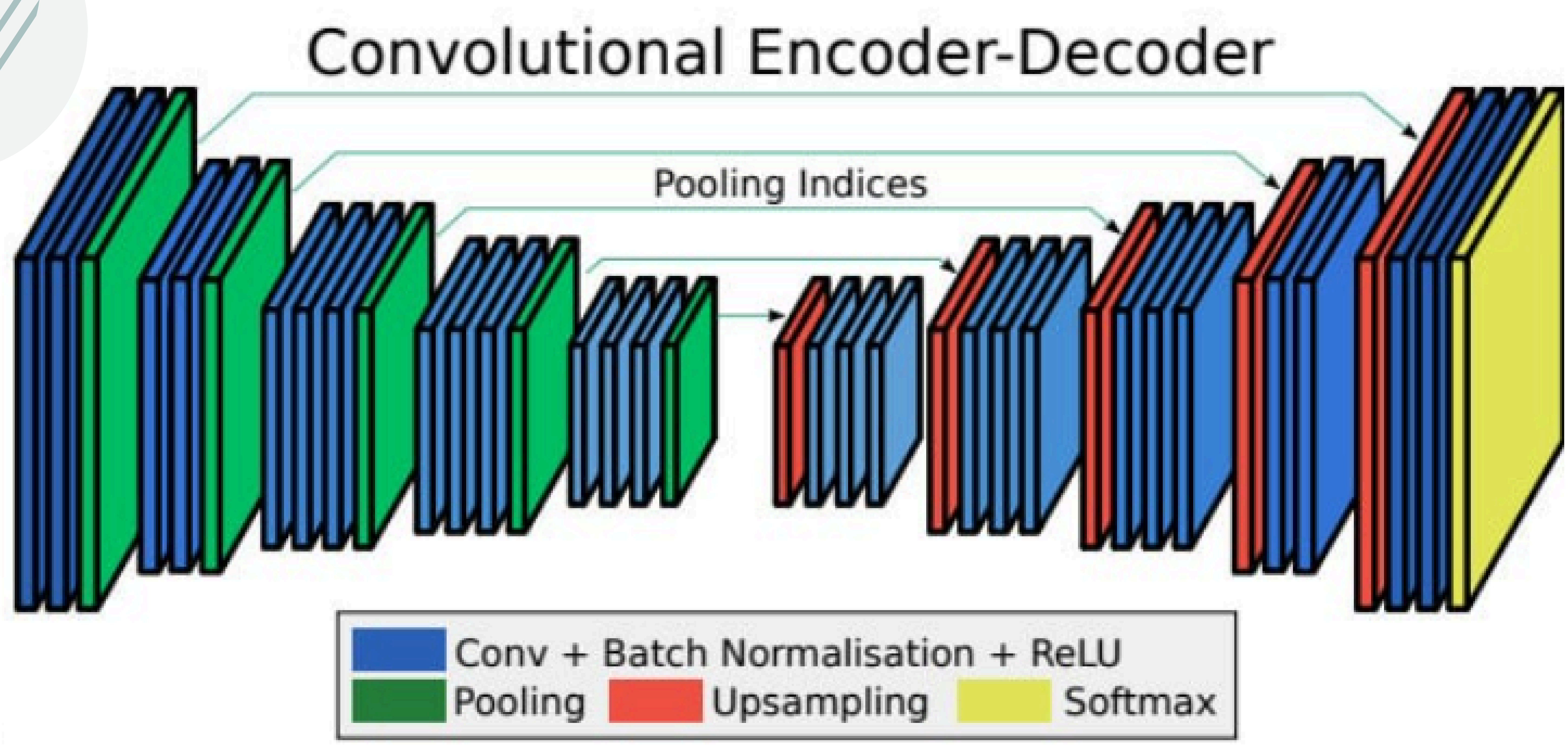
Implementation Notes:

The CutMix dataset was expanded to 175% of the original size. 75% consisted of selected difficult-class samples (non-white masks). The rest were random samples into which patches were inserted.
Only samples with visible damage were used for mixing to ensure the augmented images always contain meaningful content.

AFTER CUTMIX APPLICATION



SEMANTIC SEGMENTATION MODEL



RESNET-50

ResNet-50 is a 50-layer deep convolutional neural network that provides **Hierarchical Feature Extraction.**
Deeper layers progressively combine these into more abstract object parts (rooftops, shadows, rubble). It is excellent at extracting rich, generalizable features.
ResNet-50 serves as a powerful, off-the-shelf feature extractor that transforms raw six-channel satellite inputs into a compact, semantically rich representation.

ENCODER

ResNet-50 up through “Layer4”:

Stem

7×7 conv → batch-norm → ReLU → 3×3 max-pool (stride 2).

Input channels: 6 (stacked pre/post RGB).

Residual Stages

Layer1: 3 bottleneck blocks, output stride 4

Layer2: 4 bottleneck blocks, output stride 8

Layer3: 6 bottleneck blocks, output stride 16

Layer4: 3 bottleneck blocks, output stride 32

Output: feature tensor of shape (2048, 16, 16) from layer4.

DECODER

Simple two-stage up-sampling

Conv Block 1

2048→512 channels via 3×3 convolution + ReLU

Up-sample by factor 2 → (512, 32, 32)

Conv Block 2

512→256 channels via 3×3 convolution + ReLU

Up-sample by factor 16 → (256, 512, 512)

Final Head

1×1 convolution → 5 output channels (logits)

Output: 5-channel logits – channel 0 for background, 1–4 for rest

TRAINING PIPELINE

- **Loss function:** Cross-Entropy on all 5 channels, measures the difference between the model's predicted class scores (logits) and the true class labels on a per-pixel basis.

- **Optimizer**: (Learning rate: 2.02e-4, Weight decay: 1e-6), applies weight decay separately from the gradient update, which leads to more consistent regularization and better generalization

SOFTMAX

$$softmax(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}}$$

- z0: background vs. building
- z1 to z4: damage levels 1–4

So,

- p0 is the model’s confidence that the pixel is not damaged (i.e. background or undamaged).
- p1 to p4 are the confidences for each damage class(white, green, orange, red).

VALIDATION METRICS

- **Location Dice:** Measures overlap of predicted vs. true building footprints
  Dice = 2·TP / (2·TP + FP + FN) (=0.74)

- **Damage-Level F1** (per class 1–4): Evaluates precision & recall of each damage category within building mask and takes Harmonic mean of the four per-class F1 scores
  F1 = 2·TP / (2·TP + FP + FN)

- **Combined Validation Score:** Balances footprint accuracy (30 %) with damage-level performance (70 %)
  (= **0.3 × Location Dice + 0.7 × Aggregate Damage F1**) (=0.58)

Model Parameters

Train Samples: 7062

Test Samples: 1600

Epoch: 20

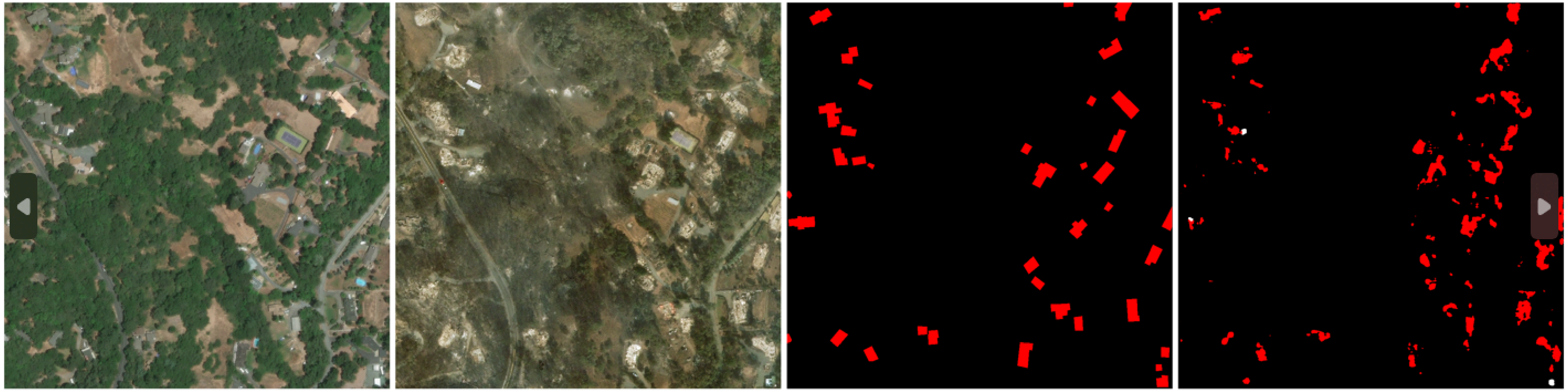
Batch Size: 2

Image Dimensions: 512 x 512

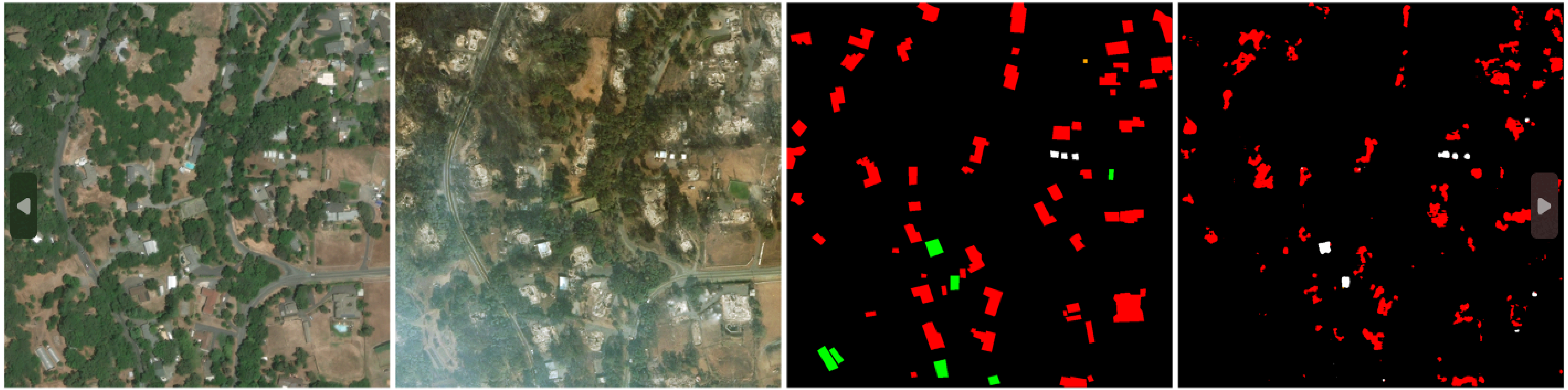
Training Time

Total time = 2370 ~ 1.6 Days

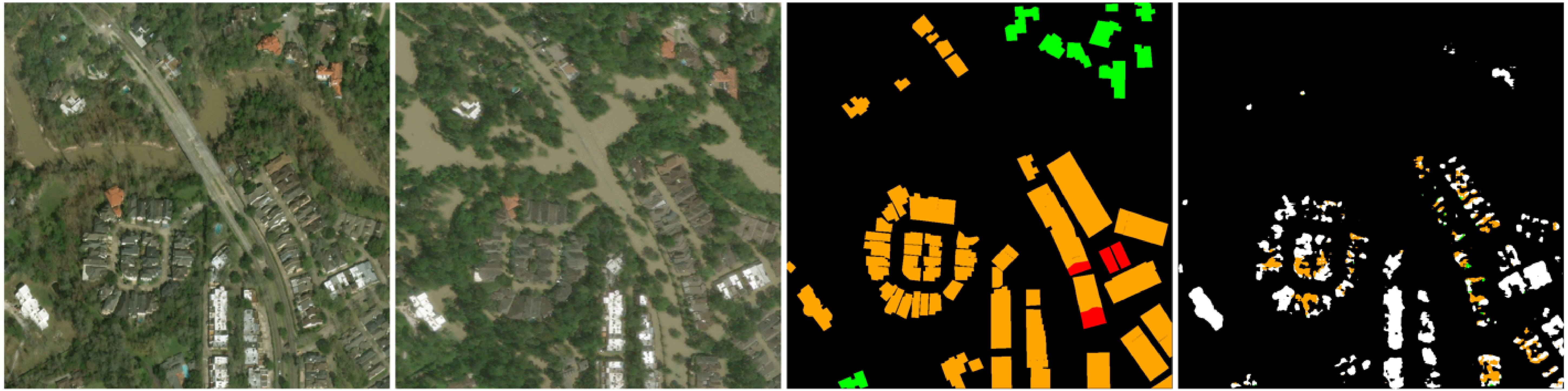
PREDICTIONS



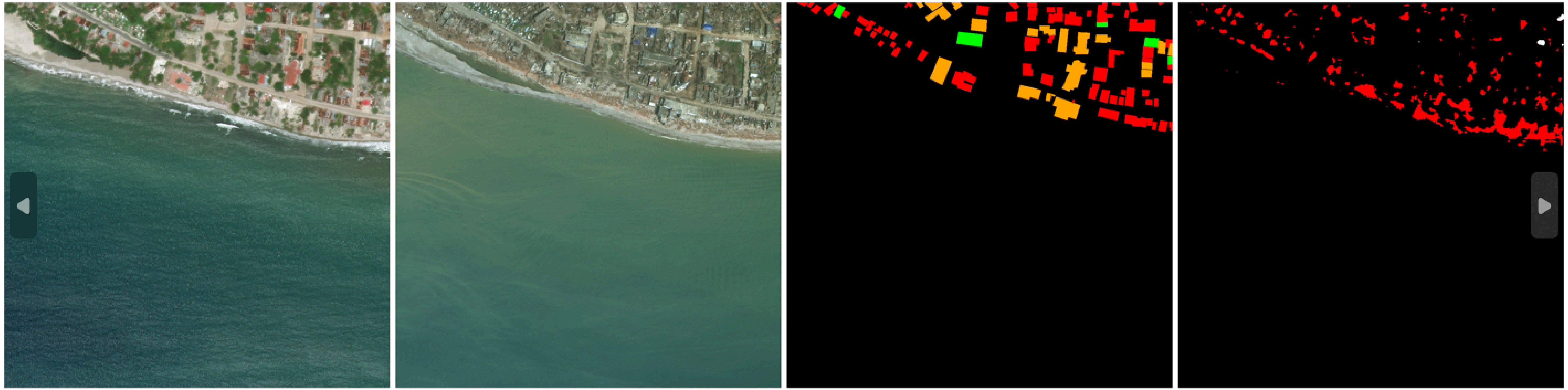
Pre-disaster | Post-disaster | Original Label | Predicted Label



Pre-disaster | Post-disaster | Original Label | Predicted Label



Pre-disaster Post-disaster Original Label Predicted Label



Pre-disaster Post-disaster Original Label Predicted Label

OBSERVATIONS

- **Accurate footprint localization**, even under heavy destruction; effectively detects large, contiguous severe-damage zones.
- **Challenges in mild vs. moderate damage classification** due to blurry boundaries, leading to occasional confusion.
- **Prediction issues:** small buildings may vanish in smoke/vegetation; some areas show over-segmentation or missed mild damage.
- **False positives** from noise, requiring threshold tuning, post-processing, and class-balanced fine-tuning to improve mild-damage detection.

BDANET

Overview:

Two-Stage Framework:

1. **Stage 1 (Building Segmentation):**
   - Input: Pre-disaster image.
   - Output: Binary mask of building locations.
   - Backbone: U-Net with ResNet-50 encoder.
2. **Stage 2 (Damage Assessment):**
   - Input: Pre- and post-disaster images (two-branch U-Net).
   - Output: Damage level per pixel (4 classes + background).
   - Key Modules: MFF and CDA.

Stage 1: Building Segmentation

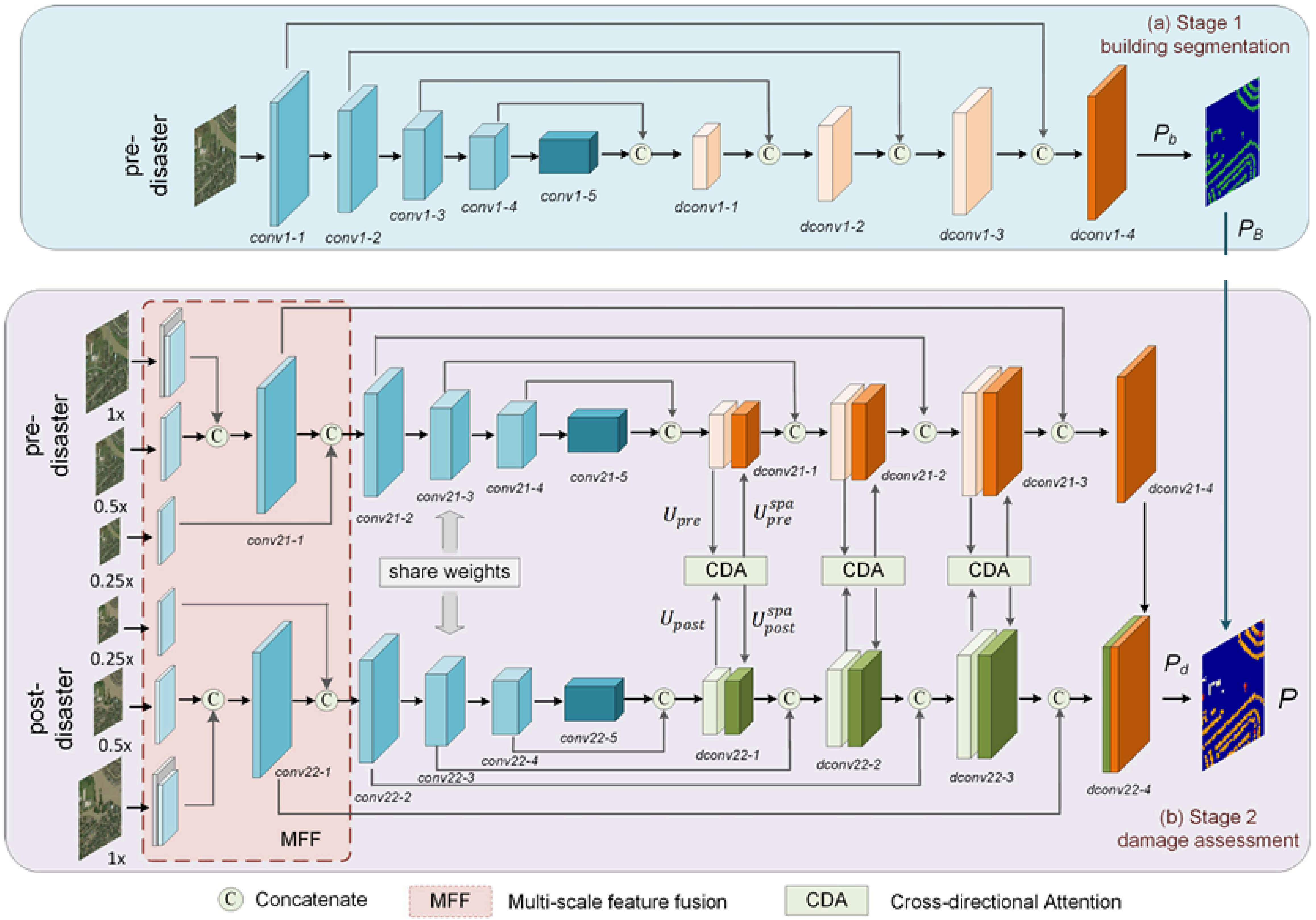
- Encoder: ResNet50 (ImageNet pretrained)
- Decoder: Upsamples features + skip connections (from encoder).
- Loss: Binary Cross Entropy
- Purpose: Identifies intact buildings in pre-disaster images to guide Stage 2.

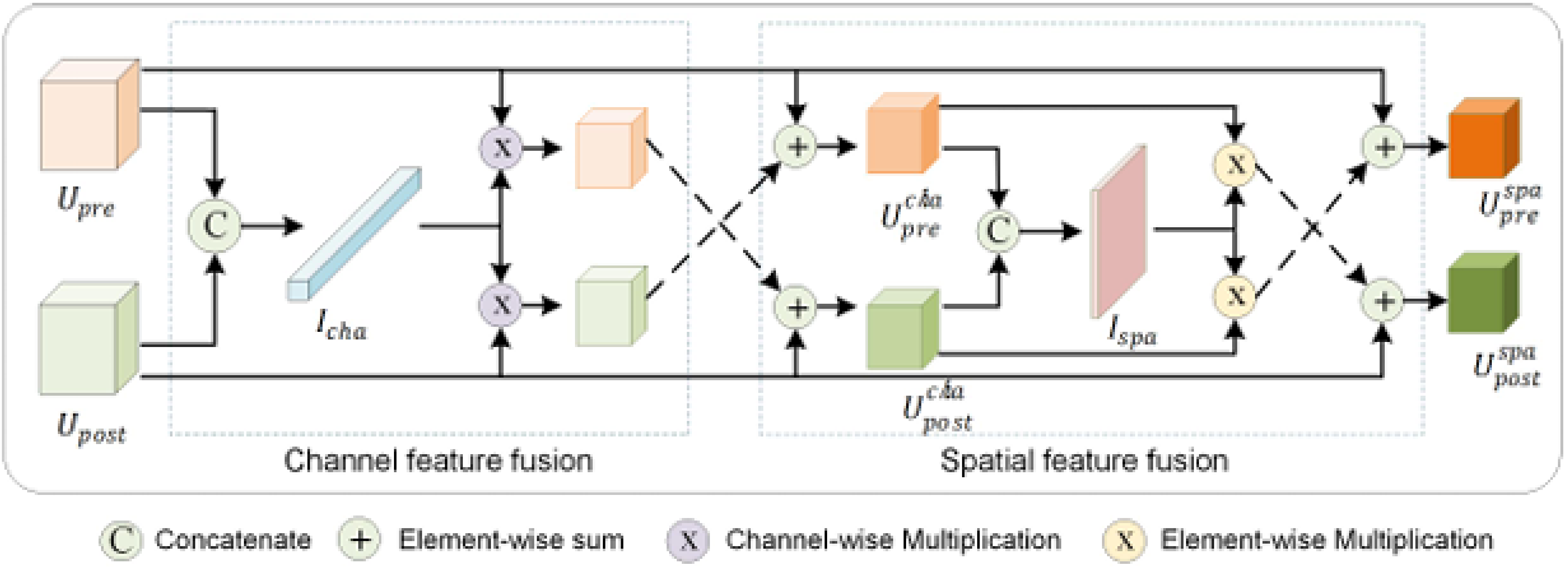
Stage 2: Damage Classification

- Architecture: Dual U-Net (pre & post image branches) + CDA modules
- Loss: Cross Entropy
- Output: Multi-class damage mask (5 levels)

Attention Mechanism (CDA):

- Captures spatial and channel-wise dependencies
- Fuses features between pre & post images to identify damage accurately


(a) Stage 1
building segmentation
pre-disaster
conv1-1
conv1-2
conv1-3
conv1-4
conv1-5
dconv1-1
dconv1-2
dconv1-3
dconv1-4
P_b
P_B
pre-disaster
post-disaster
1x
0.5x
0.25x
0.25x
0.5x
1x
conv21-1
conv21-2
conv21-3
conv21-4
conv21-5
dconv21-1
dconv21-2
dconv21-3
dconv21-4
share weights
U_{pre}
U_{pre}^{spa}
U_{post}
U_{post}^{spa}
CDA
CDA
CDA
conv22-1
conv22-2
conv22-3
conv22-4
conv22-5
dconv22-1
dconv22-2
dconv22-3
dconv22-4
P_d
P
MFF
(b) Stage 2
damage assessment
C Concatenate
MFF Multi-scale feature fusion
CDA Cross-directional Attention



U_{pre}
U_{post}
I_{cha}
U_{pre}^{cha}
U_{post}^{cha}
I_{spa}
U_{pre}^{spa}
U_{post}^{spa}
Channel feature fusion
Spatial feature fusion
C Concatenate
+ Element-wise sum
X Channel-wise Multiplication
X Element-wise Multiplication

Model Parameters

Train Samples: 7464 (pre & post images)

Test Samples: 1866 (pre & post images)

Epoch: 20

Batch Size: 4

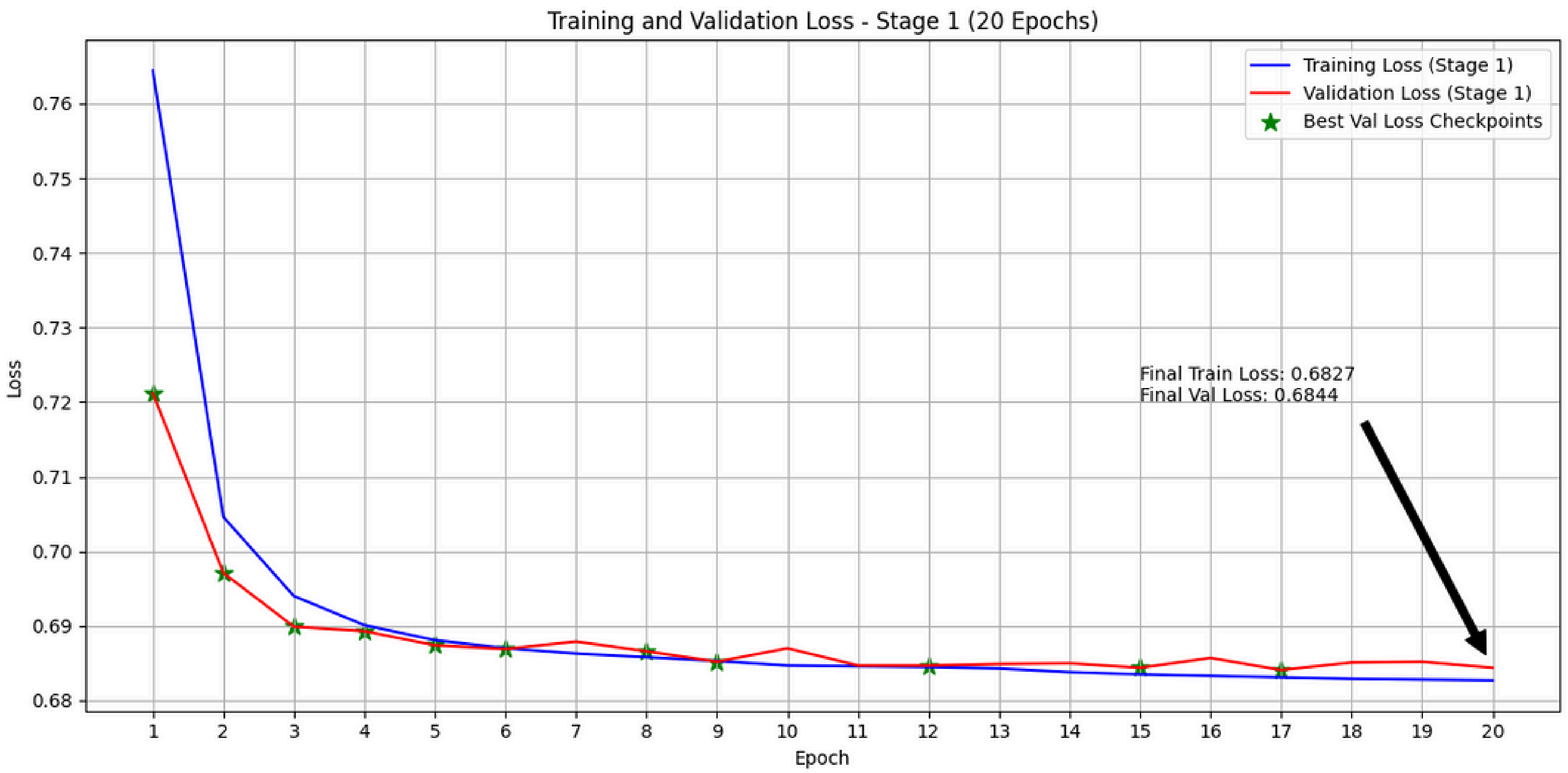
Image Dimensions: 512 x 512

Training Time

Stage 1: 460 mins ~ 7.6 hrs

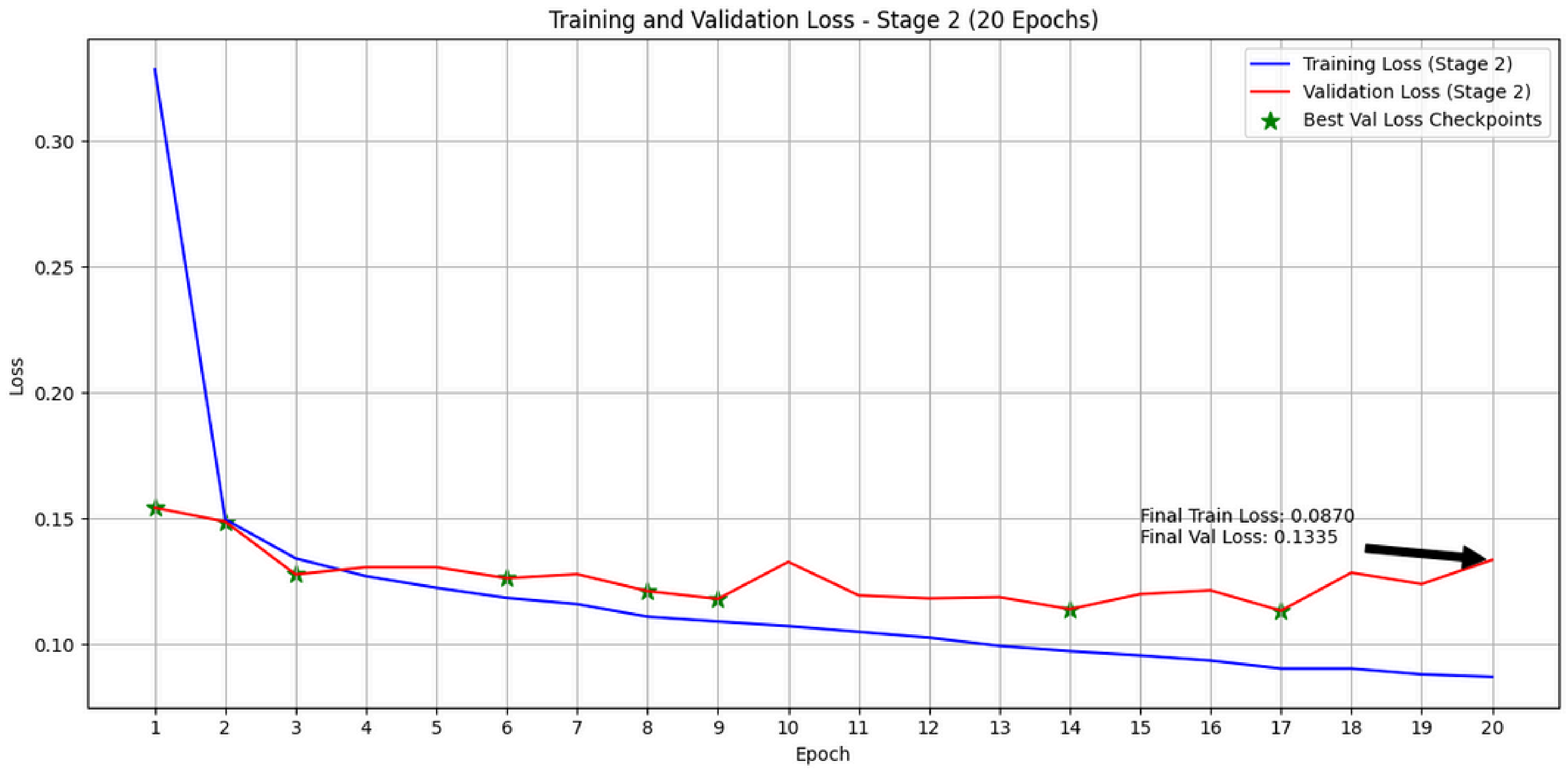
Stage 2: 3740 mins ~ 62.3 hrs

Total time ~ 3 Days

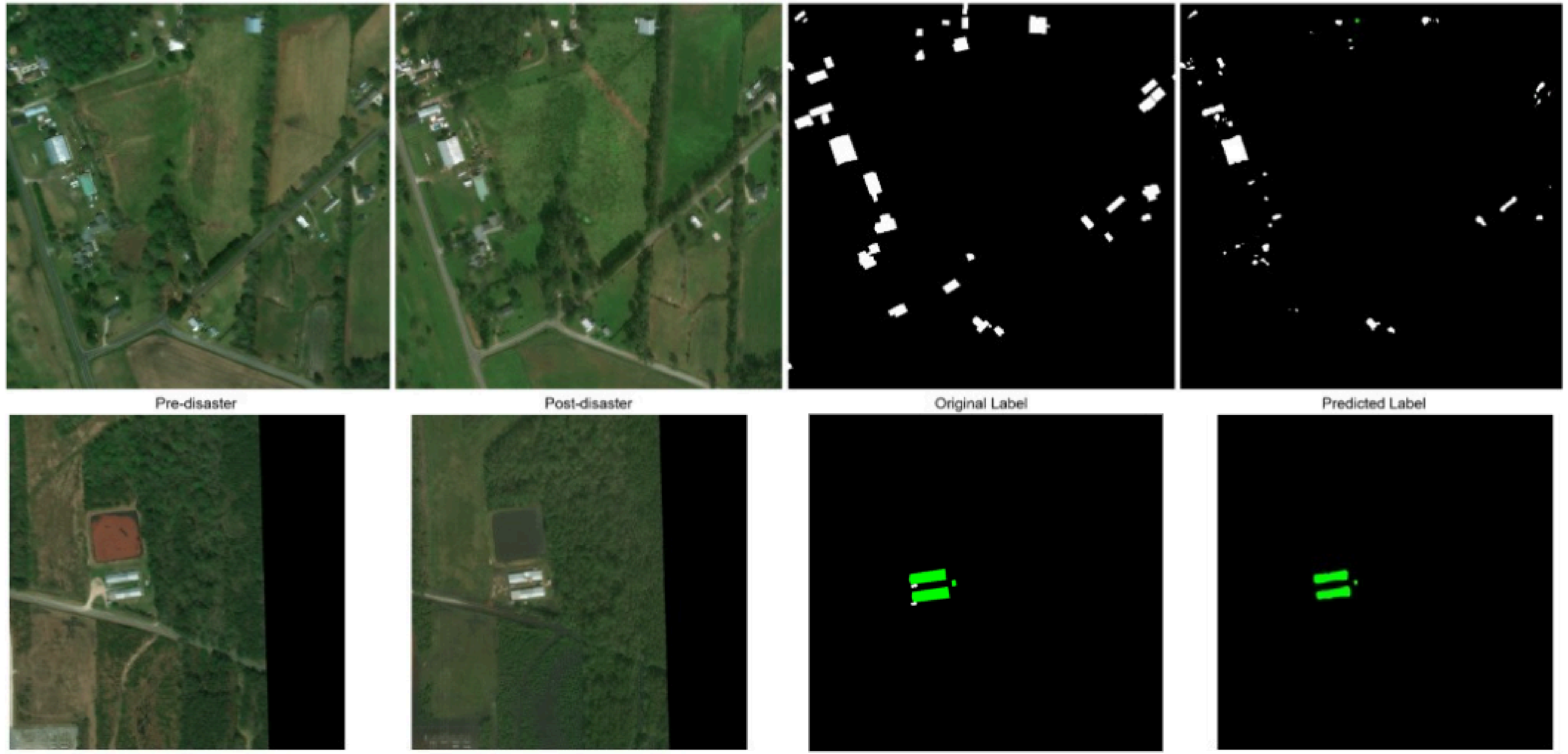


Min validation loss
Epoch 17
Stage 1
Train Loss: 0.6831, Val Loss: 0.6841

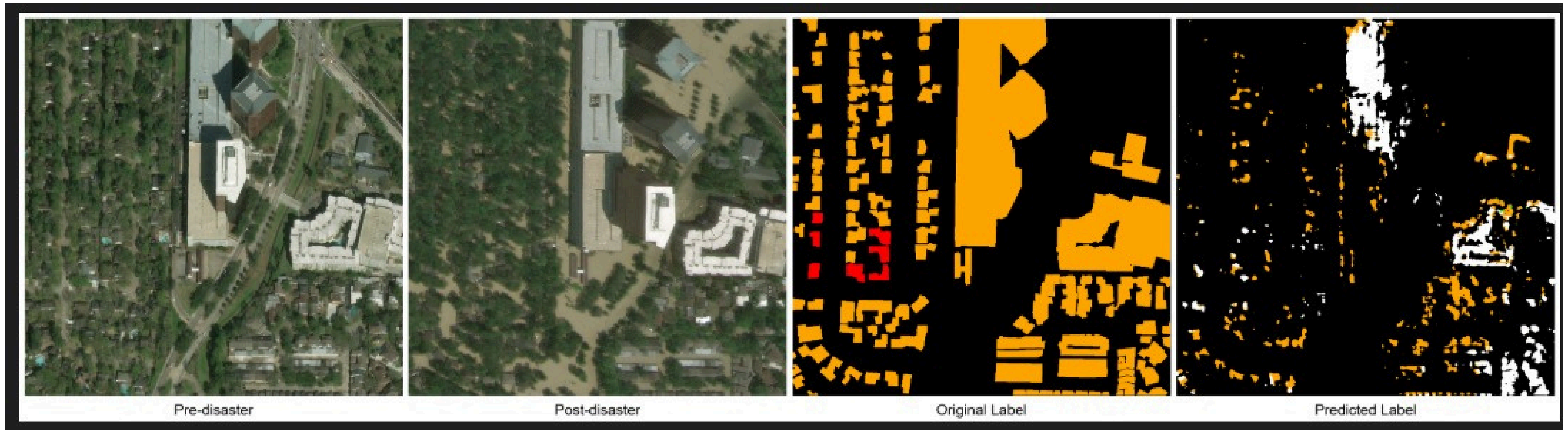
Min validation loss
Epoch 17
Stage 2
Train Loss: 0.0903, Val Loss: 0.1133



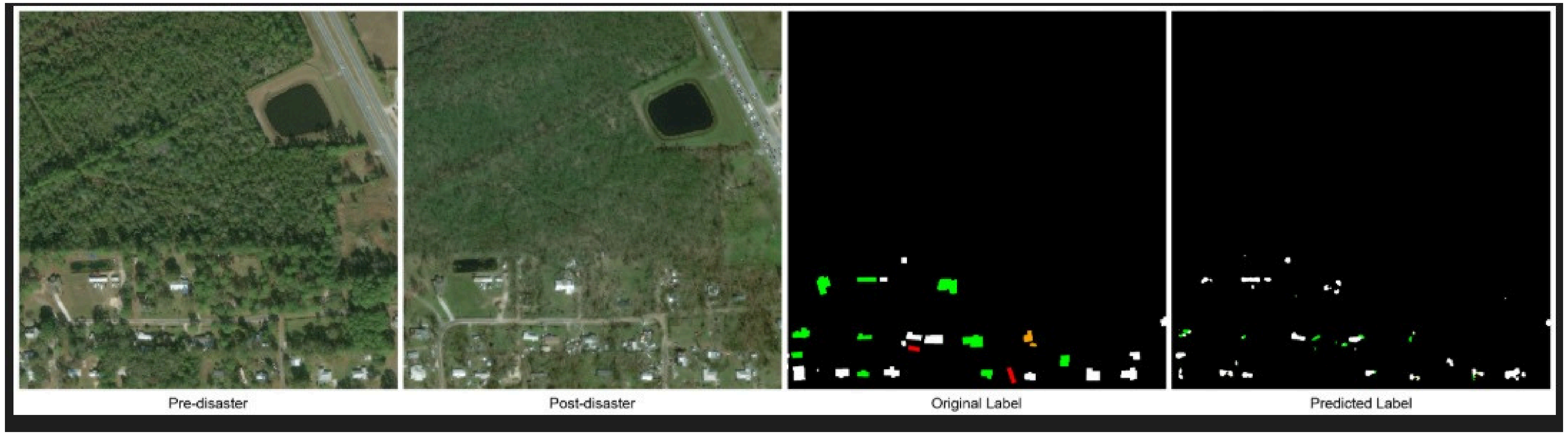
Predictions



Predictions



Predictions



Reasons for wrong predictions:

1) Class Imbalance (Primary Cause)
2) Poor Feature Discrimination for Similar Classes
3) Ineffective Fusion of Pre/Post-Disaster Features
4) Insufficient Data Augmentation
5) Localization Errors from Stage 1

Siamese U-Net with Alpha Blending

- Automated disaster damage assessment is a crucial yet challenging task. Manual inspection is time-consuming, error-prone, and unsafe in many situations.
- Here we use a Siamese U-Net architecture with Alpha Blending, capable of comparing pre-disaster and post-disaster imagery for precise pixel-level damage classification.

Siamese U-Net with Alpha Blending: Architecture Overview

- The Siamese U-Net with Alpha Blending is a specialized deep learning architecture designed for tasks requiring comparison between paired images, such as disaster damage assessment. This architecture combines three powerful concepts:
- Siamese Networks: For comparative analysis of image pairs
- U-Net Architecture: For precise pixel-level segmentation
- Alpha Blending: For adaptive feature fusion
- This architecture is particularly effective for disaster damage assessment because it can directly compare "before" (pre-disaster) and "after" (post-disaster) images to identify, classify, and segment damaged areas.

Architectural Components

1. Siamese Network Concept

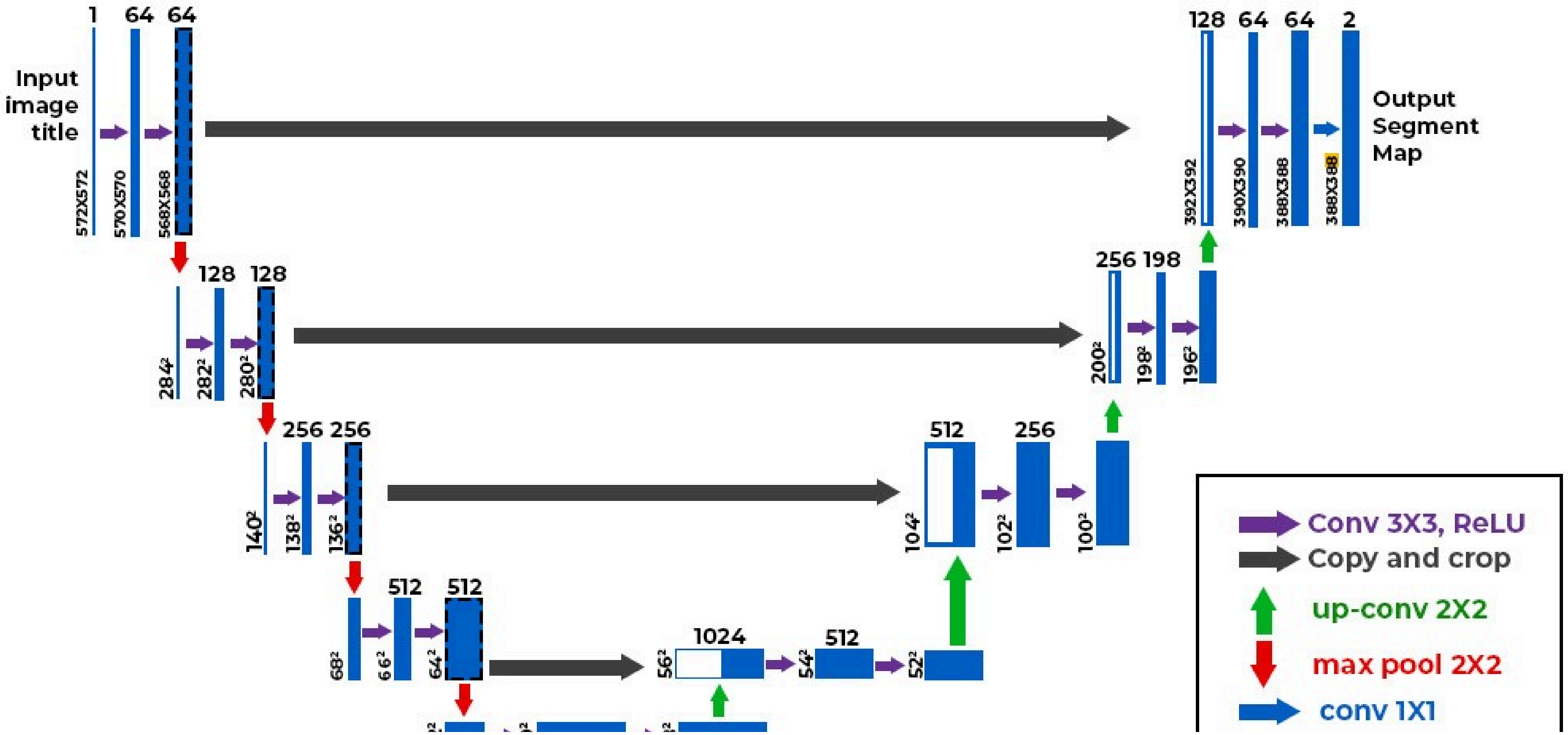
A Siamese network consists of twin neural networks that share weights, allowing them to process two different inputs while maintaining the same transformation logic.

Key benefits for disaster assessment:

- Identical feature extraction from both pre and post-disaster images
- Weight sharing reduces parameter count and helps prevent overfitting
- Enforces consistent feature representation across time points

2. U-Net Architecture

U-Net is an encoder-decoder architecture with skip connections, originally developed for biomedical image segmentation but now widely used for various segmentation tasks:



3.Alpha Blending Mechanism

A alpha blending module is introduced for feature fusion. At each level of the encoder-decoder bridge, features from pre- and post-disaster branches are blended as follows:

$$blended = \alpha \times post\_features + (1 - \alpha) \times pre\_features$$

- α is a learnable parameter for each layer.
- α is passed through a sigmoid to constrain it to the [0, 1] range.
- The blending enables adaptive focus on either the pre- or post-disaster image based on context.

This module is key to the model's interpretability and performance in change detection.

Key components:

- Encoder Path (Contracting): Uses DoubleConv and Down modules to extract features at increasing levels of abstraction
- Decoder Path (Expanding): Uses Up modules to upsample and recover spatial information
- Skip Connections: Connect corresponding levels of encoder and decoder to preserve spatial details

Implementation Details

The implementation includes several optimizations:

- Progressive Resizing: Starting with smaller images (256×256) and gradually increasing to final size (512×512)
- Mixed Precision Training: Using FP16 calculations where possible to speed up training
- Gradient Accumulation: Accumulating gradients over multiple batches to effectively increase batch size
- Class Weighting: Assigning higher weights to rare damage classes

These optimizations help the model train efficiently and effectively, even with limited data.

- Input: Paired pre-disaster and post-disaster images (RGB, 512×512 px)
- Output: Pixel-wise 5-class damage segmentation map

Dataset Structure

Train and test Dataset comprises matched triplets of images:

- Pre-disaster images (baseline state)
- Post-disaster images (showing damage)
- Ground truth damage masks (labeled by experts)

Combined Loss Function for Optimal Segmentation

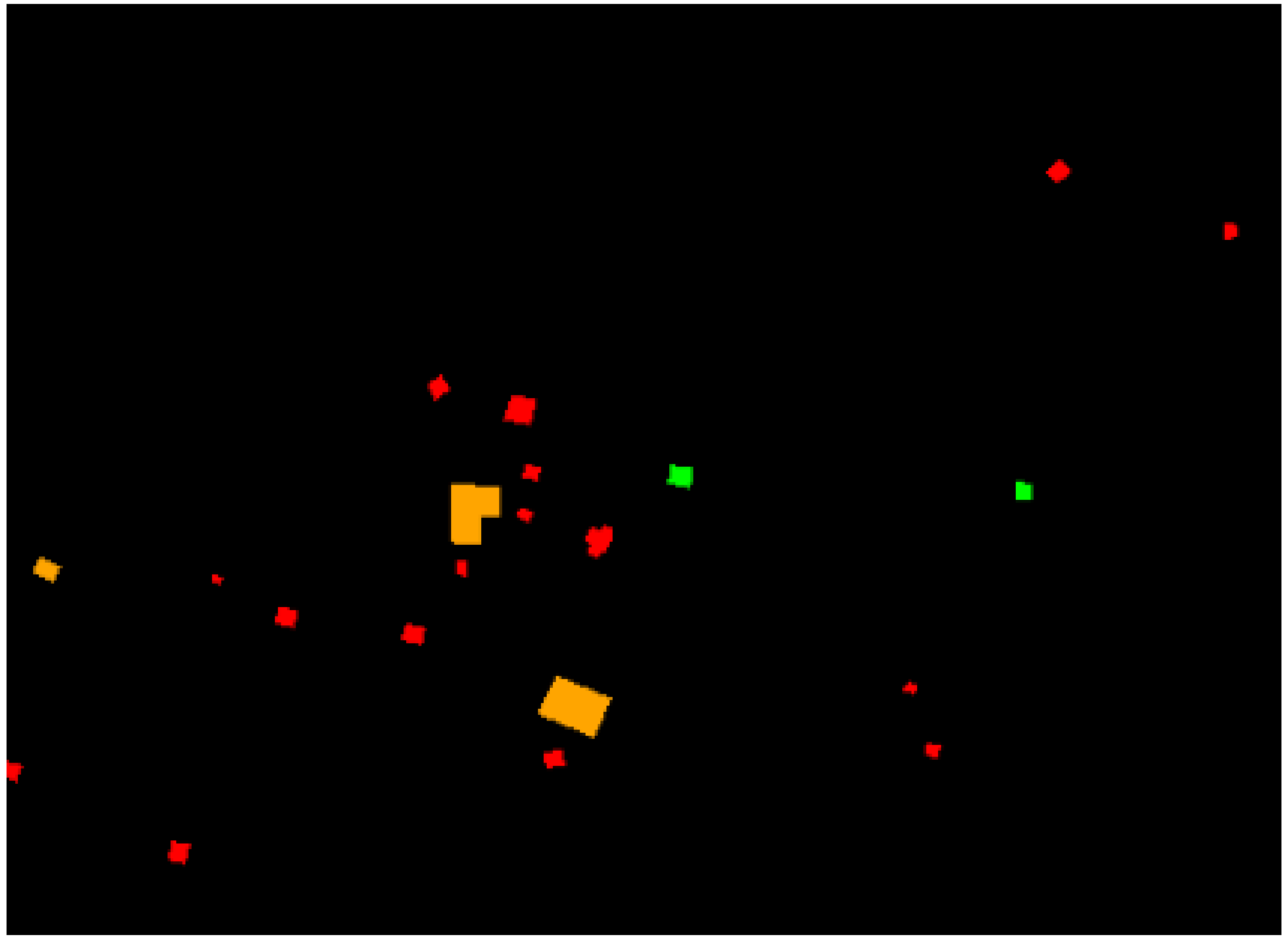
- Multi-component loss combining:

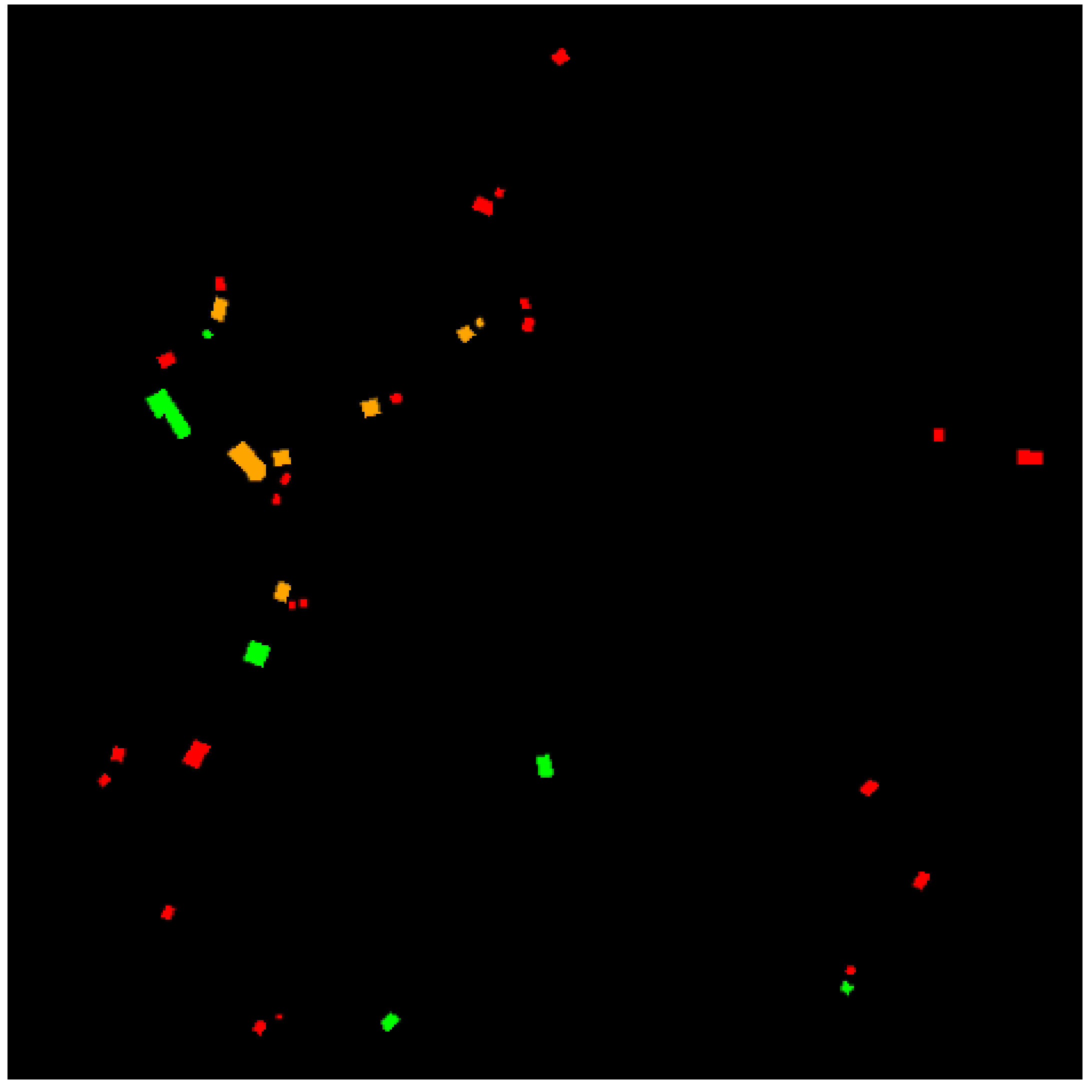
- Cross-entropy loss: Good for multi-class categorization
- Dice loss: Better for handling class imbalance and boundary precision

- Class weighting to address highly imbalanced data:
- Lower weights for common classes (background: 0.1)
- Higher weights for rare classes (destroyed: 2.5)

Damage Classification System

- **Class 0:** Background (Black) - Non-target areas or areas outside the region of interest
- **Class 1:** No Damage (White) - Structures present but unaffected by the disaster
- **Class 2:** Minor Damage (Green) - Visible damage but structure remains largely intact
- **Class 3:** Major Damage (Orange) - Significant structural damage but not completely destroyed
- **Class 4:** Destroyed (Red) - Complete destruction or collapse of structures.





Performance Metrics

Class-wise metrics for balanced evaluation:

- Dice coefficient: Measures overlap between prediction and ground truth
- IoU (Intersection over Union): Measures segmentation quality

The Dice coefficient (also called F1-score or Sørensen–Dice coefficient) measures the overlap between two sets - in this case, between predicted segmentation and ground truth :

Dice = (2 × Intersection) / (Sum of areas)

Properties:

- Range: 0.0 (no overlap) to 1.0 (perfect overlap)
- Insensitive to imbalanced data
- Focuses on intersection relative to total area

IoU (Intersection over Union) / Jaccard Index :
IoU measures the overlap between predicted segmentation and ground truth relative to their union:

IoU = Intersection / Union
= Intersection / (Area_A + Area_B - Intersection)

Properties:

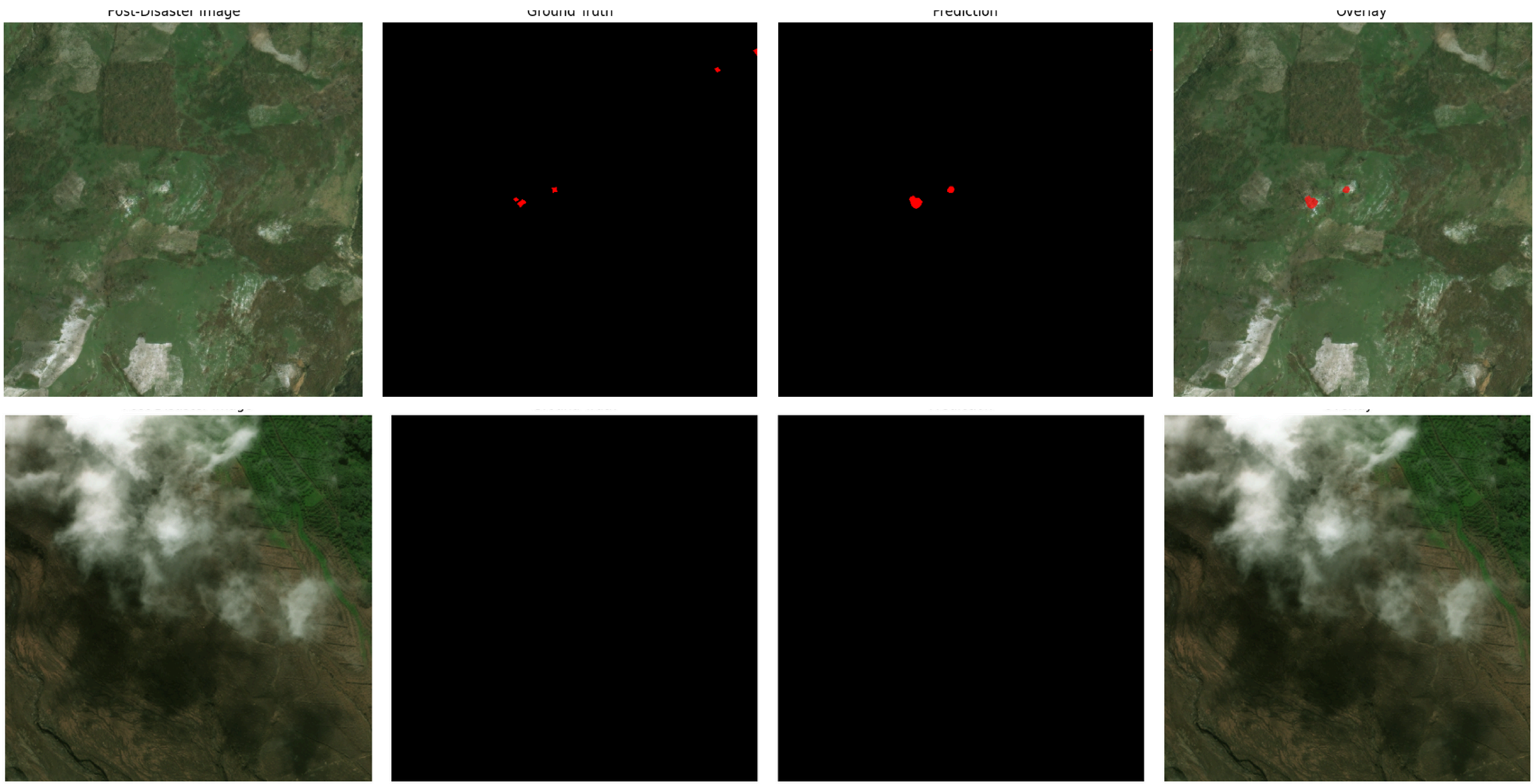
- Range: 0.0 (no overlap) to 1.0 (perfect overlap)
- More stringent than Dice (punishes any non-overlapping areas more heavily)
- Standard metric in segmentation tasks

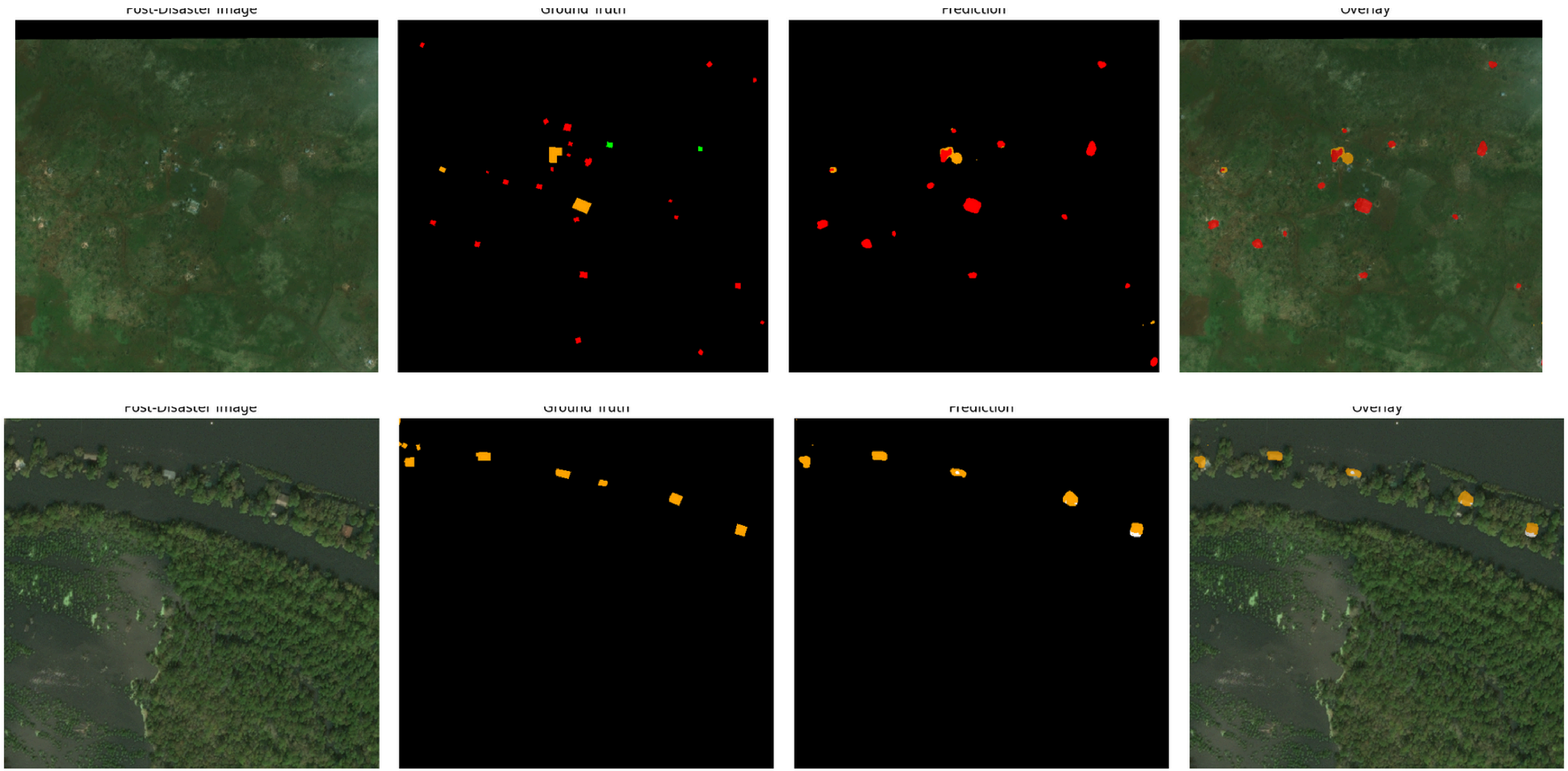
```
ing:  99%|████████████████████████████████████████| 926/933 [24:04<00:09,  1.37s/it]Res
o (512, 512)
ing:  99%|████████████████████████████████████████| 927/933 [24:06<00:08,  1.50s/it]Res
o (512, 512)
ting:  99%|███████████████████████████████████████| 928/933 [24:08<00:07,  1.52s/it]Resi
o (512, 512)
ting: 100%|███████████████████████████████████████| 929/933 [24:09<00:05,  1.47s/it]Resiz
o (512, 512)
ting: 100%|███████████████████████████████████████| 930/933 [24:12<00:05,  1.94s/it]Resiz
o (512, 512)
ting: 100%|███████████████████████████████████████| 931/933 [24:14<00:03,  1.87s/it]Resizi
o (512, 512)
sting: 100%|██████████████████████████████████████| 932/933 [24:15<00:01,  1.72s/it]Resizin
to (512, 512)
sting: 100%|██████████████████████████████████████| 933/933 [24:17<00:00,  1.63s/itTesting:
to (512, 512)
sting: 100%|██████████████████████████| 933/933 [24:17<00:00,  1.56s/it]

ass 0 - Dice: 0.9788, IoU: 0.9611
ass 1 - Dice: 0.5152, IoU: 0.4335
ass 2 - Dice: 0.7385, IoU: 0.7385
ass 3 - Dice: 0.2993, IoU: 0.2731
ass 4 - Dice: 0.5844, IoU: 0.5536
erall Metrics - Dice: 0.6232, IoU: 0.5919
sting completed in 1457.09 seconds
ocessed 933 test images
```

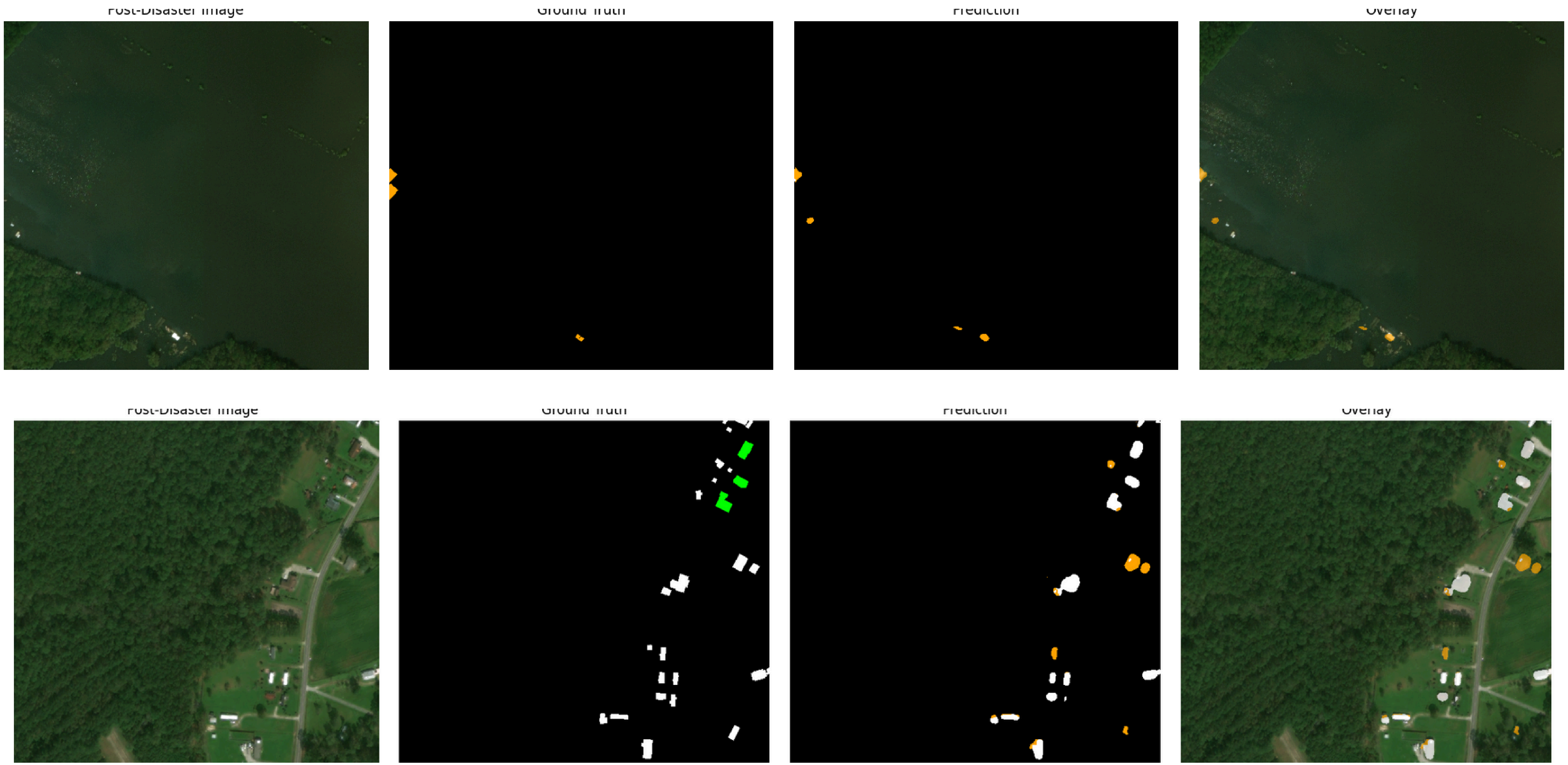
Results Visualization

Multi-view visualization system:

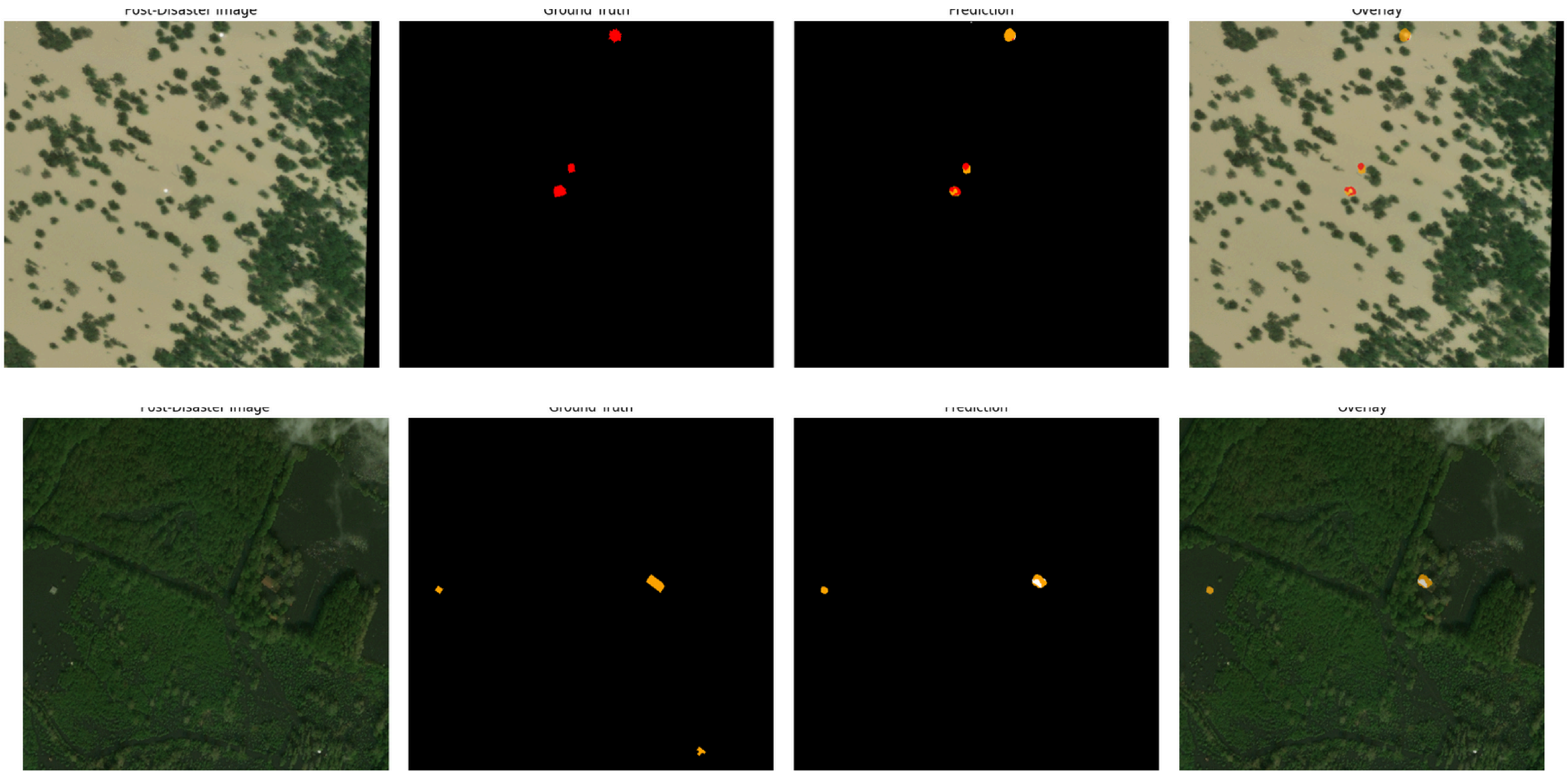
- Post-disaster image: Original state after disaster
- Ground truth: Expert-labeled damage mask
- Model prediction: Generated damage segmentation
- Overlay: Transparent prediction overlay on post-disaster image


Post-Disaster Image
Ground Truth
Prediction
Overlay


Post-Disaster Image
Ground Truth
Prediction
Overlay
Post-Disaster Image
Ground Truth
Prediction
Overlay



Post-Disaster Image
Ground Truth
Prediction
Overlay
Post-Disaster Image
Ground Truth
Prediction
Overlay


Post-Disaster Image
Ground Truth
Prediction
Overlay
Post-Disaster Image
Ground Truth
Prediction
Overlay

Technical Challenges and Solutions

Small dataset challenges:

- Automatic batch size adjustment
- Gradient accumulation optimization
- Enhanced data augmentation

Class imbalance:

- Weighted loss function
- Combined CE and Dice loss

Computational efficiency:

- Progressive resizing strategy
- Mixed precision training
- Optimized model depth (3 instead of 4-5)

Checkpoint management:

- Robust save/resume system with metadata tracking
- Best model tracking based on validation performance

Future Work and Potential Improvements

1. Architecture enhancements:
   Attention mechanisms beyond alpha blending
   Integration of transformer modules for global context

2. Data improvements:
   Multi-source data fusion (optical, SAR, LiDAR)
   Domain adaptation for different disaster types

3. Deployment considerations:
   Model pruning and quantization for edge devices
   Cloud-based processing pipeline
   Integration with disaster response systems

COMPARATIVE ANALYSIS OF SEMANTIC SEGMENTATION MODELS

SIMILARITIES ACROSS ALL MODELS

Feature	ResNet- UNet	BDANet	SiameseNet
CNN-based Architecture	Yes	Yes	Yes
Encoder-Decoder Structure	UNet	Enhanced UNet-based decoder	Specialized decoder
Skip Connections	Yes	Yes	Yes
ResNet Backbone	ResNet50	ResNet101	Optimized CNN

DIFFERENCES BETWEEN MODELS

Difference Criteria	ResNet- UNet	BDANet	SiameseNet
Backbone Complexity	Moderate (ResNet50)	High (ResNet101)	Moderate (specialized CNN)
Prediction Scope	Single Class (severe)	Multi-class (Full Severe range)	Single Class (Mild)
Attention Mechanism	None	Boundary-driven attention	None

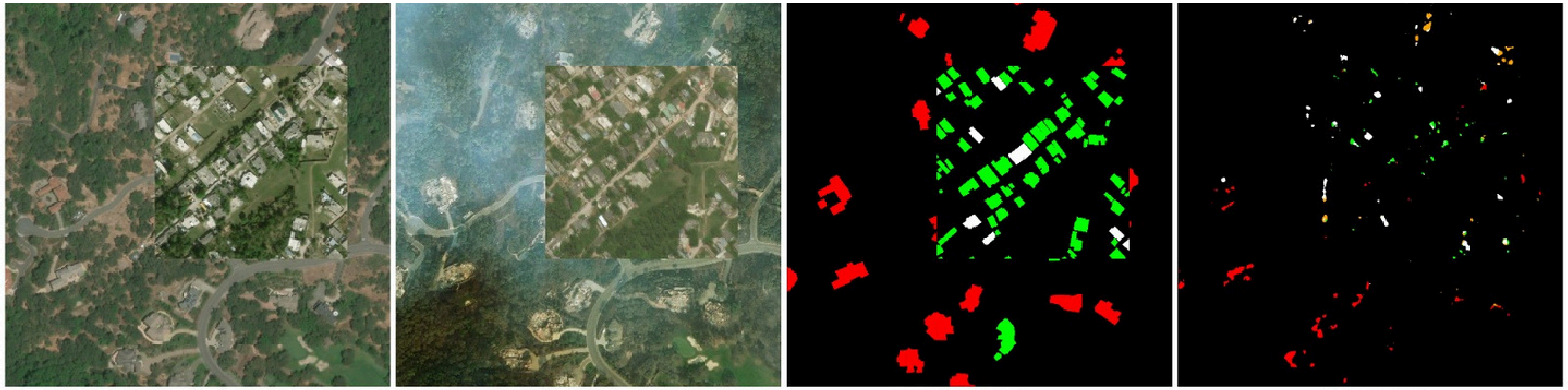
INITIAL MODEL SELECTION CRITERIA

Criteria	ResNet-UNet	BDANet	SiameseNet
Accuracy	Moderate	High	High
Computational Efficiency	High	Moderate	High
Multi-class Capability	No	Yes	No
Robustness	Good	Excellent	Good

DATASETS ACROSS ALL MODELS

Models	Dataset	Model Failure	Model Sucess
ResNet- UNet	Traditional Augmentation	No	Yes
BDANet	Raw	No	Yes
SiameseNet	Raw	No	Yes
BDANet	Cut-mix	Yes	No

PREDICTION OF BDANET ON CUT-MIX DATASET



Pre-disaster | Post-disaster | Original Label | Predicted Label

Very Low Accuracy

Final Decision: Reject Model

ADVANTAGES AND LIMITATIONS COMPARISON

Model	Advantages	Limitations	Severity Prediction
ResNet-UNet	Efficient, robust severe region prediction	Lower accuracy for complex boundaries, one-class biasness	Mostly and Dominatingly Severe disaster regions (Red), Sometimes White
BDANet	High accuracy, multi-class, precise	Higher computational demands	Dominating any-one class, Mostly Mild disaster regions (Orange)
Siamese U-Net	High accuracy in mild regions	Limited to mild region prediction	White, Orange, Red but not Green

FINAL MODEL RESULT COMPARISION

- ResNet-UNet shows high accuracy in detecting severe damage (red) but consistently underperforms in recognizing mild or moderate damage classes, often defaulting to red. It overpredicts red regions, making it unsuitable for cases where damage severity varies spatially.
- BDANet demonstrates a peculiar bias—its predictions strongly favor the dominant damage class in the input. For instance, if mild damage is prevalent, other classes are suppressed and orange dominates the output. This makes BDANet unreliable in real-world cases where multiple severity levels often coexist.
- Siamese U-Net, although slightly lower in raw performance metrics, is the only model that maintains consistent multi-class predictions across a range of severity levels. It reliably segments white (no disaster), green (low), orange (moderate), and red (severe) regions—even when they coexist. Its only notable weakness is in detecting green/no-damage zones.

FINAL CONCLUSION

Criterion	ResNet-UNet	BDANet	Siamese U-Net
Multi-class Prediction Handling	Poor – Dominated by red	Unstable – Dominated by one class	Good – All classes reasonably detected
Performance on Mixed Severities	Weak	Inconsistent	Strong
Class Bias	High (Red)	High (Varies with dominant class)	Low (Except green)
Practical Utility	Limited to severe-only cases	Limited to uniform-class dominance	Suitable for real-world disasters
Green Class Detection	Moderate	Inconsistent	Weak

FINAL DECISION

In real-world disaster imagery, multiple levels of damage often coexist. Therefore, a model's practical utility is not only determined by its overall accuracy but by its ability to correctly differentiate and represent all severity classes in the same image.

Based on these models, we select **Siamese U-Net** as the primary segmentation model.

- Best balance in practical scenarios where multiple damage severities coexist.
- Despite low green class accuracy, it outperforms others in real-world utility.
- Recommended improvements include fine-tuning for green detection and leveraging ensemble post-processing.

THANK YOU