Appendix

A. Algorithm to Generate Synthetic Dataset

Algorithm 1 Algorithm to Generate Synthetic Dataset

Input: $\mathbf{x}_{\text{base}}$ and $\mathbf{x}_{\text{cut}}$ (base image and another image for CutMix from the EuroSAT dataset), p_1 and p_2 (probabilities to apply seasonal change and cloud cover), is_disaster (indicator to apply task-relevant change)

Output: $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_5\}$ (generated satellite image time series)

for $i \leftarrow 1$ to 4 **do** ▷ Apply seasonal change and cloud cover to mimic pre-disaster images
  $\mathbf{x}_i \leftarrow \text{ApplyRandomSeasonalChange}(\mathbf{x}_{\text{base}}, p_1)$
  $\mathbf{x}_i \leftarrow \text{ApplyRandomCloudCover}(\mathbf{x}_i, p_2)$
end for
if is_disaster **then**
  $\mathbf{M} \leftarrow \text{GenerateRandomSoftMask}$ ▷ Generate a random mask with the Gaussian filter for CutMix
  $\mathbf{x}_5 \leftarrow \text{CutMix}(\mathbf{x}_{\text{base}}, \mathbf{x}_{\text{cut}}, \mathbf{M})$
end if
$\mathbf{x}_5 \leftarrow \text{ApplyRandomSeasonalChange}(\mathbf{x}_5, p_1)$ ▷ Only apply seasonal change to the last image to mimic the post-disaster scenario

B. Implementation Details

The original data for the EuroSAT and RaVÆn datasets are based on Sentinel-2, an optical satellite that captures geo-referenced images across 13 spectral bands. For our experiments, we simplify the process by using only the RGB bands (bands 4, 3, and 2), which provide sufficient visual information for detecting extreme events. This choice not only aligns with common practices in optical satellite image analysis but also reduces memory usage during model training.

For the autoencoder setup described in Section 3.2, we employ a weight-shared Vision Transformer [10] as the encoder and a weight-shared transformer-based decoder [2, 15] to reconstruct the satellite patch time series. Inspired by the hyperparameter choices in [10] and [15], we set the embedding dimension to 256, patch size to 8, encoder depth to 4, number of heads to 8, and decoder depth to 4, considering our input patch size of 64×64.

Our method is implemented using PyTorch [22]. Adam with a weight decay of $1e^{-6}$ is used as the optimizer, and a grid search for learning rates within the range [$1e^{-5}$, $5e^{-5}$, $1e^{-4}$, $5e^{-4}$, $1e^{-3}$, $5e^{-3}$, $1e^{-2}$] revealed that $1e^{-4}$ was the most stable. Cosine annealing with a warmup epoch of 10 serves as the learning rate scheduler. The batch size and learning epoch are uniformly set to 64 and 200, respectively, for all experiments. Following standard machine learning protocols, hyperparameter selection is done using the validation set based on the primary evaluation metrics, Average Precision (AP). Based on extensive grid search, we set the loss balancing weights $\lambda = 0.5$ and $\mu = 0.5$ for the synthetic dataset, and $\lambda = 0.25$ and $\mu = 0.5$ for the real-world dataset. To complement the primary evaluation metric AP, which is threshold-independent and used for model optimization, we also report the F1 score to demonstrate the model's performance at a specific threshold. The threshold for the F1 score is selected on the testing set via grid search to identify the value that maximizes F1. It is important to note that this threshold selection is solely for demonstration purposes and does not influence model training or hyperparameter tuning. The primary evaluation remains based on the threshold-independent AP metric to ensure unbiased performance assessment. This procedure is applied consistently to both our method and all baseline methods to ensure fair comparisons.

All results reported in this paper are based on models trained three times with different random seeds: 42, 43, and 44, respectively. NVIDIA A100 GPUs were used for all experiments.

C. Rationale for Patch-Level Focus: Dataset Characteristics and Challenges

In this work, we focus on patch-level information rather than pixel-level information for extreme events detection. While pixel-level analysis is a common practice in remote sensing, our choice of patch-level focus stems from specific characteristics of the RaVÆn dataset used in this study.

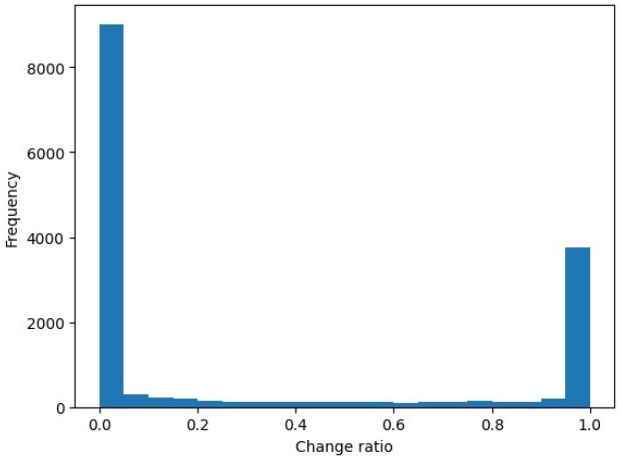
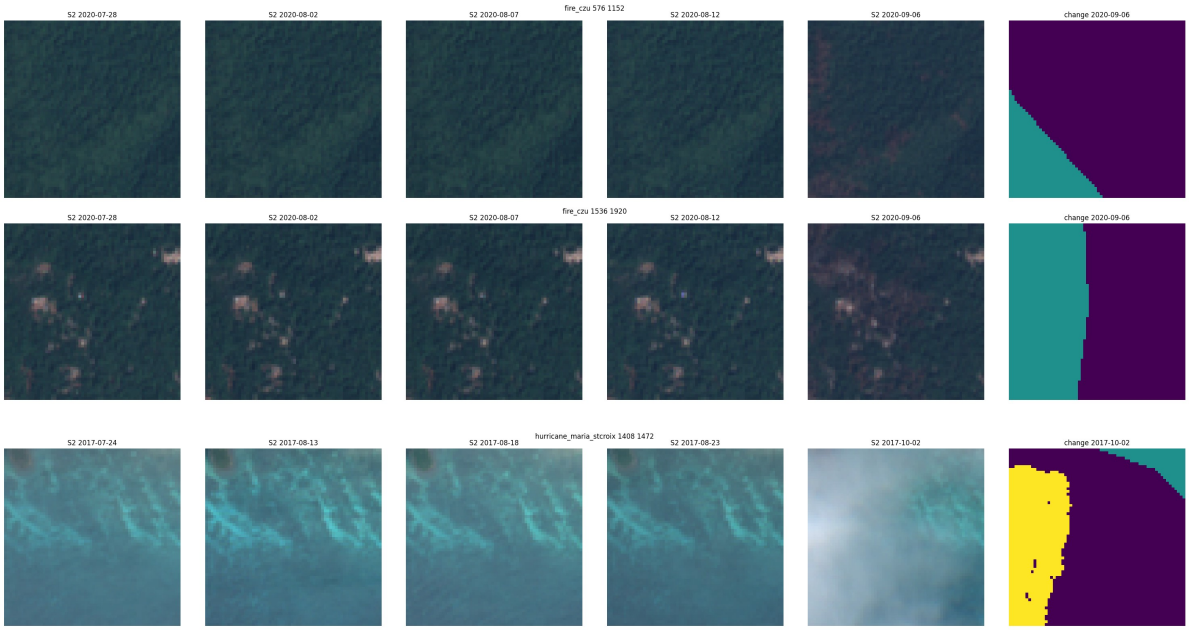


Figure 5. Histogram of patch *change ratios*, where the *change ratio* is defined as the proportion of changed pixels relative to the total pixels in each patch. The binary distribution shows that most patches are either entirely unchanged (change ratio close to 0) or fully affected by extreme events (change ratio close to 1). This supports the rationale for a patch-level focus in our approach.

(a) Good annotation examples (flooding): Fine-grained and meaningful pixel-level annotations.



(b) Bad annotation examples: Coarse and unclear pixel-level annotations, making them difficult to interpret.

Figure 6. Examples of pixel-level annotations in the dataset. Good annotations 6a are meaningful and accurately reflect ground truth, particularly for flooding events, while bad annotations 6b are coarse and challenging for model interpretation.

This section highlights the reasoning behind this approach and its implications for future research.

A key characteristic of the RaVÆn dataset is that most patches are either entirely unchanged or entirely affected by extreme events. This binary distribution makes patch-level focus a reasonable choice for effective detection. Figure 5 illustrates the histogram of patch statistics, highlighting the dominance of fully-changed and unchanged patches.

Another factor influencing our patch-level approach is the quality of the pixel-level annotations in the dataset. While some annotations, particularly for flooding events, are meaningful and accurately reflect ground truth (e.g., Figure 6a), many annotations for other disaster types are coarse and often difficult to interpret (e.g., Figure 6b), making it challenging for the model to learn from pixel-level annotations, further motivating a patch-level focus.

In summary, the choice of patch-level focus is driven by the dataset's binary patch characteristics and limitations in pixel-level annotations. Addressing these dataset-specific challenges highlights the need for large-scale multi-disaster datasets with fine-grained, consistent annotations. Such datasets would better reflect real-world disaster response needs, enabling more robust and generalizable models for future research.