

POLITECNICO DI TORINO

Master's Degree in Data Science and Engineering



Master's Degree Thesis

Deep Learning for Binary Post-Earthquake Building Damage Detection from Multi-Temporal VHR Imagery

Supervisors

Prof. ANDREA BOTTINO

Dott. LORENZO INNOCENTI

Dott. JACOPO LUNGO VASCHETTI

Candidate

PAOLO RIOTINO

OCTOBER 2025

Abstract

Rapid, reliable mapping of earthquake-induced building damage is essential for directing emergency response. This thesis tackles building-level damage detection from pre/post-event very-high-resolution satellite imagery, using the ESA Φ -Lab *AI for Earthquake Response Challenge* as context. The challenge targets automatic identification of damaged vs. undamaged buildings from image pairs and building polygons, leveraging historical Charter activations and multi-mission VHR data (e.g., Pléiades-1, WorldView-2/3, GeoEye-1, Kompsat-3/3A, Gaofen-2) to simulate real-world conditions.

We present an end-to-end pipeline designed for three persistent difficulties: severe class imbalance, image misalignment, and geographic domain shift. Supervision is restricted to building footprints to limit label noise, and a pixel-level weighting term (computed from the ratio of valid negatives to positives) is integrated into the loss to avoid discarding scarce positives. To mitigate effective resolution differences across sensors, we apply pan-sharpening where panchromatic band is available, fusing the high-resolution pan band with lower-resolution multispectral bands to enhance spatial detail before learning.

We adopt a Siamese architecture with a shared DINOv3-pretrained ConvNeXt encoder for the pre and post-event images; their embeddings are combined to predict building damage.

Experiments on held-out areas and events assess classification performance and robustness of the final system. We discuss limitations (cross-sensor variability, footprint quality, and generalization across geographies) and outline practical next steps toward deployment, including multi-class severity estimation, domain adaptation. Overall, the work delivers a reproducible pipeline and actionable insights for scaling satellite-based, post-event damage mapping in real response scenarios.

Table of Contents

List of Tables	v
List of Figures	vii
Acronyms	x
1 Introduction	1
2 The AI for Earthquake Response Challenge	5
2.1 Phase 1 – Training and Live Scoring	6
2.2 Phase 2 – Stress Test	6
3 Related Works	8
3.1 Stationary Wavelet Transform (SWT) for Pan-sharpening	8
3.2 U-Net	9
3.3 UPerNet	9
3.4 ChangeOS	10
3.5 ResNet	11
3.6 ConvNeXt	11
3.7 Vision Transformers (ViT)	12
3.8 Swin Transformer	13
3.9 DINO	13
3.9.1 DINOv3-distilled ConvNeXt	14
3.10 Uncertainty-Weighted Loss	14
3.11 Evaluation metrics	15
4 Dataset	16
4.1 ESA-Challenge Dataset	16
4.1.1 Image Sources	17
4.1.2 Train Dataset buildings	18
4.1.3 Test Dataset buildings	19

4.1.4	Challenges Related to the Imagery	19
4.1.5	Challenges Related to Annotations	23
5	Methodology	24
5.1	Dataset Preprocessing	24
5.1.1	Patch and Mask Generation	25
5.1.2	Benchmark Dataset	28
5.1.3	Deployment Dataset	28
5.2	Model Architectures	29
5.2.1	Decoders and Encoders	29
5.2.2	Fusion Strategies	30
5.3	Training Strategy	31
5.3.1	Data Augmentations	32
5.3.2	Loss Functions	33
5.3.3	Fine-Tuning Strategies	34
5.3.4	Prediction Strategies	35
6	Results	37
6.1	Implementation Details	37
6.2	Benchmark Dataset Results	38
6.2.1	Baseline	38
6.2.2	Alternative Encoders and Decoders	40
6.2.3	Fusion strategies	41
6.2.4	Prediction strategies	42
6.2.5	Loss Functions	45
6.2.6	Base vs. Large	46
6.2.7	Fine-Tuning of Encoders	48
6.3	Challenge Results	50
6.4	Summary of Results	51
7	Conclusions	52
	Bibliography	54

List of Tables

4.1	Train Data (6,408 buildings in total)	18
4.2	Phase 1 Test Data (60,050 buildings in total)	19
4.3	Phase 2 Test Data (465,740 buildings in total)	20
6.1	Baseline performance of ResNet50–U-Net on ID and OOD test sets. Scores are reported as global building-level F1, precision, and recall. The model was trained using early concatenation fusion technique, a pixel-wise loss, and the mean-probability rule for building-level prediction.	39
6.2	Approximate number of trainable parameters for the evaluated encoder architectures. The listed values refer to the base and large variants used in the experimental analysis.	40
6.3	Comparison of alternative encoder–decoder configurations. Scores are reported as building-level F1, precision, and recall. The model was trained using early concatenation fusion technique, pixel-wise loss and the mean-probability rule for building-level prediction. . . .	40
6.4	Comparison of fusion strategies. Scores are reported as building-level F1, precision, and recall. The model was trained using pixel-wise loss and the mean-probability rule for building-level prediction. . . .	42
6.5	Comparison of prediction strategies. Scores are reported as building-level F1, precision, and recall. The model was trained using late concatenation fusion technique and pixel-wise loss.	43
6.6	Comparison of loss functions for ConvNeXtV2. Scores are reported as building-level F1, precision, and recall. The model was trained using late concatenation fusion technique, mean-probability rule for building-level prediction.	45
6.7	Comparison of loss functions for DINOv3 under different prediction strategies. Scores are reported as building-level F1, precision, and recall. The model was trained using late concatenation fusion technique.	45

6.8	Comparison between base and large ConvNeXtV2 models. Scores are reported as building-level F1, precision, and recall. The model was trained using late concatenation fusion technique, mean-probability rule for building-level prediction.	47
6.9	Comparison between base and large DINOv3 models. Scores are reported as building-level F1, precision, and recall. The model was trained using late concatenation fusion technique.	47
6.10	Performance with partial fine-tuning of ConvNeXtV2 encoder backbones. Scores are reported as F1, precision and recall. The model was trained using late concatenation fusion technique and the mean-probability rule for building-level prediction.	49
6.11	Performance with partial fine-tuning of DINOv3 encoder backbones. Scores are reported as F1, precision and recall. The model was trained using late concatenation fusion technique and the mean-probability rule for building-level prediction.	49
6.12	Official challenge results for Phase 1 (ID) test sets. Scores refer to the global building-level F1 metric used for leaderboard ranking. Model: CNXv2–UPerNet architecture, late concatenation fusion technique, pixel-wise loss, mean-probability rule for building-level prediction and the encoder partially frozen (last 3 layers).	50
6.13	Official challenge results for Phase 2 (OOD) test sets. Scores refer to the global building-level F1 metric used for leaderboard ranking. Model: DINOv3–UPerNet architecture, late concatenation fusion technique, building-level loss, mean-probability rule for building-level prediction and the encoder partially frozen (last 3 layers).	51

List of Figures

3.1	Overview of the U-Net architecture, consisting of a contracting encoder path, an expanding decoder path, and symmetric skip connections that preserve spatial detail.	10
3.2	Schematic of a residual block, where the input \mathbf{x} is added to the learned transformation $F(\mathbf{x}; \theta)$ through an identity skip connection.	11
4.1	Antakya East, Türkiye — Comparison of pre-event images of the same building acquired by different very-high-resolution satellites at different times.	18
4.2	Histogram of damaged vs. undamaged buildings across cities. Note the strong dominance of undamaged samples and the over-representation of certain regions.	19
4.3	Wundwin, Myanmar — Example of a pre-event image where the low spatial resolution, combined with sensor-related color anomalies, makes it difficult to clearly distinguish the building footprint.	20
4.4	Osmaniye, Türkiye — Pre/post-event images acquired from different sensors showing variations in viewing geometry.	21
4.5	Kahramanmaraş, Türkiye — Example of a building that appears slightly degraded in the post-event image. However, according to the ground-truth mask it is labeled as <i>undamaged</i> , since the visible deterioration is due to pre-existing wear rather than earthquake damage. In the inference phase, such cases may nonetheless be predicted as damaged.	22
4.6	Antakya East, Türkiye — Example of a building where the pre-event image lacks a panchromatic band and the available multispectral data is of relatively low quality, resulting in reduced sharpness for damage assessment.	22
5.1	Cogo, China — Comparison of input imagery: (a) original RGB composite, (b) high-resolution panchromatic band, and (c) fused pan-sharpened image produced via SWT pan-sharpening.	26

5.2	(a) Pre-event; (b) post-event; (c) ground-truth mask. In the project, ground-truth masks are encoded as 0=undamaged, 1=damaged, 255=background; in images above the mask is visualized as an overlay on the post-event image (red=damaged, green=undamaged).	27
5.3	Siamese-like encoder–decoder architecture.	31
6.1	Example of the learning rate evolution over the training epochs following the OneCycleLR scheduling policy.	38
6.2	(a) pre-event; (b) post-event; (c) predicted damage activations heatmap (lighter = higher damaged probability); (d) thresholded prediction with $\tau = 0.5$ (red = damaged, green = undamaged); background outside building footprints is ignored.	44
6.3	Evolution of the two uncertainty parameters ($\log \sigma_{pixel}$ and $\log \sigma_{building}$) during training, illustrating how the model dynamically balances the pixel- and building-level loss contributions.	46

Acronyms

AI

Artificial Intelligence

BCE

Binary Cross-Entropy

CE

Cross-Entropy

CNN

Convolutional Neural Network

EO

Earth Observation

ESA

European Space Agency

FPN

Feature Pyramid Network

GPKG

GeoPackage

GSD

Ground sampling distance

ID

In-Distribution

LR

Learning Rate

MS

Multispectral

NIR

Near Infrared

PAN

Panchromatic

PPM

Pyramid Pooling Module

OOD

Out-of-Distribution

SWT

Stationary Wavelet Transform

TIFF

Tagged Image File Format

VHR

Very High Resolution

ViT

Vision Transformer

Chapter 1

Introduction

Major earthquakes turn cities into fast-changing, high-stakes environments where decisions made in the first hours can save lives. Satellite constellations now deliver very-high-resolution¹ (VHR) imagery within hours of an event, but turning these data into actionable building-level damage maps is still largely manual, slow, and inconsistent across places and sensors. This thesis addresses that gap by formulating post-earthquake building damage detection as a binary classification problem on pre/post-event VHR image pairs, using the ESA Φ -Lab “AI for Earthquake Response” Challenge as the operational frame of reference. The challenge’s data and protocol (multi-mission imagery, building polygons, strong class imbalance, misalignment and geographic transfer to unseen cities) mirror the realities of real response, and therefore provide a rigorous test ground for methods that must generalize beyond a single city or sensor.

Problem statement and approach.

Given a building footprint and the corresponding pre and post-event satellite image chips, the task addressed in this thesis is to determine whether that building has suffered damage or remains undamaged. This problem formulation captures the essence of post-disaster damage mapping: deriving a reliable binary decision at the building level directly from VHR imagery acquired before and after an earthquake. Achieving this goal, however, requires balancing several interdependent objectives. First, the proposed pipeline must operate under realistic and demanding conditions typical of emergency response. These include severe class imbalance,

¹Very High Resolution satellite imagery refers to optical remote sensing data with a ground sampling distance (GSD) of typically less than 1 m, allowing detailed observation of individual buildings and infrastructures.

where damaged buildings represent only a small fraction of all structures, alongside significant variations in sensor characteristics such as radiometric response, spatial resolution, and acquisition geometry. Second, the approach must demonstrate not only accuracy within the training domain but also the ability to generalize to unseen environments. This distinction is explicitly built into the challenge framework, which separates in-distribution (ID) evaluation (covering cities represented in the training data) from out-of-distribution (OOD) testing on entirely new urban areas. The latter represents the true operational test of a model’s robustness, as it reflects real-world deployment where no annotated examples from the affected city are available in advance. Finally, beyond model performance, the work aims to gain an ablative understanding of what factors most contribute to reliable predictions under these constraints. Controlled experiments compare different architectural components (encoders, decoders, and multi-temporal fusion strategies) as well as various loss formulations at both the pixel and building level. To pursue these aims, we build a reproducible pipeline around the Challenge dataset, which aggregates multi-mission VHR imagery and building footprints across recent earthquake events in Türkiye, Syria, Morocco, China, Myanmar, and Afghanistan. After aligning imagery to vector footprints and generating pre/post/mask triplets, we apply pan-sharpening where panchromatic bands are available. On the modeling side, we adopt a Siamese architecture with a shared encoder (tested across several backbones, including DINOv3-distilled ConvNeXt variants) and evaluate both U-Net and UPerNet decoders, as well as diverse feature-fusion schemes (early/late concatenation, signed and absolute differencing). Performance is primarily reported as building-level F1, following the challenge’s official leaderboard protocol, with systematic ID and OOD splits to evaluate generalization and robustness.

Our Contributions.

This thesis contributes to advancing building-level post-earthquake damage detection through a combination of methodological innovations, experimental rigor, and operational realism. The first contribution lies in the preparation of a refined dataset designed to meet the challenges of emergency response imagery. By aligning heterogeneous pre and post-event data and applying pan-sharpening where panchromatic bands are available, we constructed a consistent and alignment-aware dataset at building scale. This process also explicitly addresses missing or degraded acquisitions (issues frequently encountered in rapid-response contexts) thereby improving the usability of the data for both training and evaluation. A second contribution concerns the treatment of extreme class imbalance, a persistent limitation in damage mapping tasks where undamaged buildings vastly outnumber

damaged ones. Instead of discarding valuable positive samples through undersampling, we employ a weighting scheme that adapts to the ratio of valid negative to positive instances. This strategy ensures that the learning process remains sensitive to the minority damaged class without compromising statistical stability. Third, we conduct a systematic analysis of architectural and fusion strategies for change detection. Our experiments compare modern backbone encoders (such as ConvNeXtV2, DINOv3-distilled ConvNeXt, and SwinV2) combined with U-Net and UPerNet decoders within a Siamese framework (this design allows a shared encoder to process pre and post-event inputs symmetrically), while different fusion schemes (late concatenation, signed and absolute differencing) capture complementary aspects of scene change. A further contribution involves the formulation of a hybrid supervision strategy that integrates pixel-level and building-level loss terms. Finally, we provide empirical evidence of cross-city and cross-sensor generalization, validated through both in-distribution (ID) and out-of-distribution (OOD) evaluations as defined by the ESA Φ -Lab “AI for Earthquake Response” Challenge. These experiments quantify how model performance degrades (or remains stable) when transferred to new cities and disaster contexts, offering a concrete measure of robustness under operational deployment conditions.

Document structure.

Chapter 2 - The AI for Earthquake Response Challenge. Presents the evaluation setting (Phase 1 ID vs. Phase 2 OOD), data scale (from 6,408 labeled training buildings to 465k+ test footprints), and the leaderboard metric (global building-level F1). This chapter grounds the task in a realistic response scenario.

Chapter 3 - Related Works. Reviews pan-sharpening (with SWT), segmentation backbones/decoders (U-Net, UPerNet), modern CNNs/Transformers (ResNet, ConvNeXt, Swin/ViT), self-supervised pretraining (DINO/DINOv3), and multi-task uncertainty-weighted losses that inform our design choices.

Chapter 4 - Dataset. Details sources, coverage, and annotation structure; highlights practical challenges (sensor heterogeneity, artifacts, misalignment, temporal gaps, spectral availability inequalities, and severe class imbalance) that the pipeline must accommodate.

Chapter 5 - Methodology. Describes dataset refinement, patch/mask generation, augmentations, Siamese encoders and fusion, loss formulations (pixel/building-level; hybrid), optimization, and prediction rules that convert pixel scores into building labels.

Chapter 6 - Results. Reports ablations (baselines, encoders/decoders, fusion, prediction strategies, losses, model size, fine-tuning regimes), ID/OOD performance, and official challenge outcomes, distilling what most improves robustness under operational constraints.

Chapter 7 - Conclusions. Summarizes findings, limitations (cross-sensor variability, footprint quality, geographic generalization), and outlines next steps such as multi-class severity estimation and domain adaptation toward deployable systems.

In sum, this thesis aims to close part of the distance between rapidly collected satellite imagery and equally rapid, reliable building-level situational awareness. By combining careful data preparation with change-aware architectures and principled training, we seek models that are not only accurate in familiar cities but also resilient when the next earthquake strikes somewhere new.

Chapter 2

The AI for Earthquake Response Challenge

The research presented in this work was inspired by and developed in the context of the *AI for Earthquake Response Challenge* [1], organized by the Φ -Lab of the *European Space Agency* (ESA) in collaboration with the *International Charter "Space and Major Disasters"*. The initiative was launched to explore the potential of artificial intelligence methods for rapid damage assessment after major seismic events. Timely and reliable damage mapping is a critical component of disaster response, as it enables authorities and humanitarian organizations to allocate resources, plan rescue operations, and prioritize areas most affected by the event. The challenge was motivated by the growing availability of Very High Resolution (VHR) satellite imagery in the aftermath of disasters, made possible through emergency tasking by international space agencies. While such data can capture the full extent of urban devastation within hours, manual interpretation remains slow and resource-intensive, often taking days or weeks. This gap between data availability and actionable information has been repeatedly observed in past disasters, as observed, for instance, after the 2023 Türkiye–Syria earthquake. By automating the mapping process, AI-driven approaches promise to significantly accelerate humanitarian response and reduce the time needed to mobilize rescue and relief operations.

The central objective of the challenge was to develop machine learning models capable of classifying the structural status of buildings (damaged or undamaged) based on pairs of VHR satellite imagery acquired before and after an earthquake. This formulation not only represents a highly relevant real-world application but also constitutes a demanding benchmark problem due to the variability of the data, the heterogeneity of urban landscapes, and the limited availability of labeled samples across different geographical contexts. In addition to the scarcity of

labeled data, the dataset poses unique technical difficulties: imagery originates from multiple satellite platforms with heterogeneous spatial resolutions and spectral properties, often acquired under different viewing angles and illumination conditions. Pre and post-event images are rarely perfectly co-registered, introducing spatial misalignment. Moreover, annotations are based on manual interpretation under time pressure, leading to ambiguities in borderline cases (e.g., partially collapsed vs. heavily damaged buildings). Finally, the data suffer from strong class imbalance, with the majority of structures remaining undamaged. Together, these factors make the problem substantially more challenging than conventional computer vision benchmarks.

2.1 Phase 1 – Training and Live Scoring

In the first phase, participants were provided with annotated training data and were allowed to train their models accordingly. Submissions were evaluated automatically on a hidden test set consisting of previously unseen buildings from the same cities included in the training dataset. For evaluation, an F1-score was computed separately for each city, yielding a performance score specific to that geographic area. The final global F1 was then obtained by aggregating all building-level predictions across cities, thus reflecting performance on the entire test set as a whole. This phase thus emphasized the ability of the models to generalize within the same geographic areas but across different buildings.

- Training set: 6,408 annotated buildings across multiple countries, including Türkiye, Syria, Morocco, China, Myanmar, and Afghanistan.
- Test set: 60,050 buildings from the same set of cities, unseen during training.

2.2 Phase 2 – Stress Test

The second phase constituted the core difficulty of the competition. Here, the models developed in Phase 1 were evaluated on an entirely new dataset, consisting of buildings located in cities not included in the training set. This phase tested the capability of models to perform domain adaptation, i.e., to generalize to previously unseen urban contexts with different architectural styles, building materials, and post-disaster imagery conditions.

- Test set: 465,740 buildings across two newly introduced cities in Türkiye and Myanmar.

This cross-city evaluation makes the benchmark particularly significant: unlike typical computer vision challenges that focus on closed-world generalization (e.g., ImageNet, COCO¹), here the main difficulty lies in geographic transferability.

Beyond its competitive format, the challenge also served as a community benchmark. It established a new, standardized, and large-scale reference dataset tailored to earthquake response² from satellite imagery. This resource fosters reproducible comparisons between methods and contributes to the consolidation of Earth Observation and AI4EO³ research, alongside similar efforts for flood or wildfire mapping. Through this structure, the challenge provided a rigorous benchmark for the evaluation of AI approaches to earthquake damage detection. By combining intra-city evaluation (Phase 1) with cross-city generalization (Phase 2), the competition emphasized the importance of robustness, scalability, and transferability qualities that are indispensable for real-world disaster response scenarios.

The work carried out in this thesis builds directly upon the experience gained during the challenge. In particular, the methods and experiments presented here aim not only to reproduce the competitive setting but also to extend it. By experimenting with advanced backbone architectures, exploring alternative loss functions, and assessing different prediction strategies, this work addresses several limitations revealed during the competition.

¹In datasets such as ImageNet or COCO, training and test sets are drawn from the same distribution of images. The main difficulty is avoiding overfitting, while in the earthquake damage detection case the challenge lies in transferring models to completely new geographic domains with different architectures, materials, and acquisition conditions.

²Earlier efforts such as xBD [2] already provided large-scale building-damage annotations across multiple disasters, including earthquakes. However, the ESA Φ -Lab challenge is distinctive in being specifically curated from International Charter activations and designed to benchmark cross-city generalization.

³AI4EO stands for *Artificial Intelligence for Earth Observation* [3], a research field that applies machine learning and computer vision methods to problems in remote sensing and geoscience.

Chapter 3

Related Works

Research in remote sensing-based damage assessment has drawn on a wide spectrum of methods, ranging from traditional image fusion techniques to modern deep learning architectures. In this chapter, we review the most relevant works and technical components that support our methodology. The discussion spans both classical approaches and recent advances, highlighting how the field has evolved and which ideas have proven most influential for building-level damage mapping. By examining these contributions, we set the stage for understanding the choices made in this thesis and how they relate to the broader research landscape.

3.1 Stationary Wavelet Transform (SWT) for Pan-sharpening

Pan-sharpening aims to combine the spatial detail of a high-resolution panchromatic (PAN) image with the radiometric content of lower-resolution multispectral (MS) bands. Multiresolution approaches are well suited to this goal because they separate low-frequency structure from high-frequency detail across scales [4, 5]. The Stationary Wavelet Transform (SWT)—an undecimated, translation-invariant variant of the discrete wavelet transform—preserves spatial alignment of coefficients, which helps limit ringing and shift-dependent artifacts during detail transfer [6]. A widely used wavelet-domain strategy keeps the MS low-frequency content while injecting PAN high-frequency information [7]:

$$(H_j^F, V_j^F, D_j^F) = (H_j^M, V_j^M, D_j^M) + \gamma(H_j^P, V_j^P, D_j^P), \quad j = 1, \dots, J,$$

followed by inverse SWT to obtain the fused band. Here (H, V, D) denote horizontal/vertical/diagonal detail subbands at scale j . They are obtained by convolving the image I with orientation-specific high-pass filters:

$$H_j = I * h_H^{(j)}, \quad V_j = I * h_V^{(j)}, \quad D_j = I * h_D^{(j)},$$

while the approximation coefficients (low-frequency content) are extracted with the low-pass filter $l^{(j)}$. Thus, H_j captures horizontal edge responses, V_j vertical ones, and D_j diagonal structures. Superscripts M, P, F indicate MS, PAN, and fused signals, J is the number of scales, and $\gamma > 0$ modulates the sharpness-versus-spectral-fidelity trade-off. By preserving the MS approximations and injecting only PAN details, the method enhances edges and texture while limiting large luminance shifts [7, 5].

3.2 U-Net

U-Net [8] is a fully convolutional encoder-decoder architecture with symmetric skip connections that transmit high-resolution feature maps from the contracting path to the expanding path. The encoder aggregates context through successive downsampling, while the decoder progressively recovers spatial resolution via up-convolutions (transposed convolutions). The lateral skips provide high-frequency detail and short gradient paths, jointly improving localization quality and training stability—especially for fine structures and fragmented boundaries (see Figure 3.1). Despite its compactness, U-Net achieves strong dense predictions with limited training data when paired with heavy augmentation, a property that underpins its broad adoption in segmentation tasks, including overhead imagery with large scenes and thin man-made structures. Modern implementations often replace the plain encoder with a pretrained backbone (e.g., residual or modern ConvNets) while retaining the U-shaped decoder and skip topology [9, 10]. Overall, U-Net’s combination of multi-scale context, precise localization via skips, and modularity (backbone-agnostic encoder, flexible decoder) makes it a standard baseline for per-pixel labeling and a strong foundation upon which many contemporary segmentation systems build.

3.3 UPerNet

Unified Perceptual Parsing Net (UPerNet) [11], is an encoder-agnostic segmentation architecture that fuses multi-level features through a Feature Pyramid Network (FPN) [12] and augments them with global context via a Pyramid Pooling Module (PPM) [13]. The FPN builds a top-down pathway with lateral connections from successive encoder stages, aligning spatial resolutions and merging semantics across scales; the PPM aggregates scene-level context through multi-scale pooling to mitigate local ambiguities. This combination yields rich, scale-robust representations while remaining decoupled from the specific backbone, allowing straightforward adoption with modern CNNs or Vision Transformers (ViT) [14]. UPerNet has

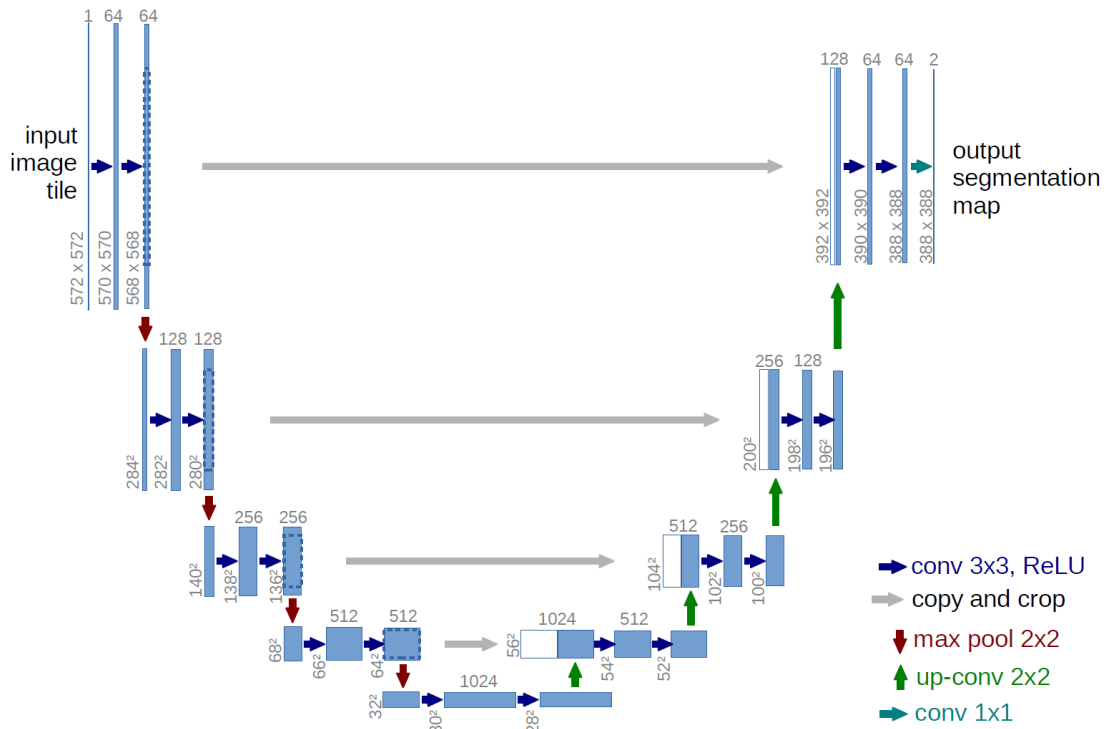


Figure 3.1: Overview of the U-Net architecture, consisting of a contracting encoder path, an expanding decoder path, and symmetric skip connections that preserve spatial detail.

become a common architecture for dense prediction due to its strong accuracy–efficiency trade-off and compatibility with high-resolution inputs. In overhead imagery, its ability to reconcile fine details with large receptive fields is particularly useful for urban scenes containing small structures alongside city-scale context.

3.4 ChangeOS

ChangeOS [15] casts building damage assessment as object-centric semantic change detection. The pipeline first localizes building instances, typically using either an object detector (e.g., bounding boxes) or an instance segmentation model that outputs building masks. From these proposals, aligned pre/post-event image chips are extracted for each object, passed through a shared-weight (Siamese) encoder, and the two feature streams are fused (e.g., concatenation and/or differencing) before a per-object change classifier. By predicting change at the instance level rather than per pixel, ChangeOS aligns supervision and evaluation with the decision unit of interest (buildings), mitigates boundary noise from imperfect co-registration,

and produces operational outputs (per-building labels) suitable for rapid disaster mapping. The framework is backbone-agnostic and can be paired with standard encoders, while remaining compatible with either detector or segmentation-based instance proposals.

3.5 ResNet

Residual Networks (ResNet) [9] address the optimization degradation that arises when stacking many layers by introducing identity skip connections that learn residual functions. A residual block (see Figure 3.2) adds a learned transformation $F(\cdot)$ to its input,

$$\mathbf{y} = \mathbf{x} + F(\mathbf{x}; \theta),$$

which preserves an unhindered gradient path through the identity term and stabilizes training in very deep models. In practice, ResNet employs either basic (3×3) blocks or bottleneck blocks ($1 \times 1 \rightarrow 3 \times 3 \rightarrow 1 \times 1$) with projection shortcuts when dimensions change. The architecture yields strong, transferable features and a favorable accuracy–efficiency trade-off, which has made ResNet backbones a standard choice for dense prediction when paired with encoder–decoder heads (e.g., U-Net) or multi-scale necks (e.g., FPN/UPerNet), as well as for shared-weight (Siamese) encoders in change detection.

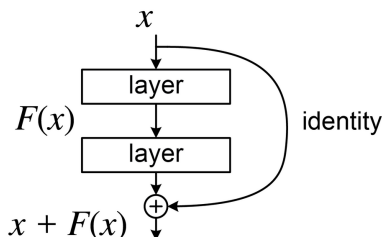


Figure 3.2: Schematic of a residual block, where the input \mathbf{x} is added to the learned transformation $F(\mathbf{x}; \theta)$ through an identity skip connection.

3.6 ConvNeXt

ConvNeXt [16] was introduced as a modernized convolutional architecture that rethinks ResNet design in light of the lessons learned from Vision Transformers. While keeping the overall residual stage structure of ResNets, it incorporates several key changes: large 7×7 depthwise separable kernels to capture wider spatial context, a patchify stem that replaces the initial 7×7 convolution, Layer Normalization

instead of Batch Normalization for more stable optimization, and GELU activations in place of ReLU. These adjustments bring ConvNeXt closer in spirit to ViT-style models while retaining convolutional inductive biases such as locality and translation equivariance. Building on this design, ConvNeXtV2 [10] co-designs the architecture with self-supervised pretraining via a fully convolutional masked autoencoder (FCMAE), and introduces Global Response Normalization¹ (GRN) to improve optimization and channel interaction. Compared to the original ConvNeXt, V2 emphasizes scalable pretraining and stable training dynamics, yielding strong transfer to dense prediction benchmarks (e.g., ADE20K²) while maintaining efficiency and deployment friendliness on conv-optimized hardware. In practice, both ConvNeXt and ConvNeXtV2 serve as drop-in encoders for segmentation heads (e.g., UPerNet or U-Net) and for shared-weight Siamese encoders in change detection. They offer a favorable accuracy–efficiency trade-off, making them competitive alternatives to transformer backbones in high-resolution overhead imagery.

3.7 Vision Transformers (ViT)

Vision Transformers (ViT) [14] treat an image as a sequence of fixed-size patches, each flattened and linearly projected to a token embedding; optional class and positional embeddings are added, and the sequence is processed by a stack of Transformer [18] encoder blocks. Self-attention models long-range dependencies with

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V,$$

where Q, K, V are learned projections of the token embeddings and d is the key dimension. Each encoder block couples multi-head self-attention with a position-wise feed-forward network: a small fully connected network applied independently to each token, wrapped with residual connections and layer normalization for stable optimization. When pretrained at scale and adapted with segmentation necks/heads, ViT backbones transfer strongly to dense prediction thanks to their global receptive fields. Practical considerations include the quadratic cost of attention in the number of tokens (salient for very high-resolution inputs), the patch-size vs. resolution trade-off, and the benefits of large-scale pretraining for sample efficiency.

¹GRN measures how strong each feature channel is across the whole image, rescaling channels relative to one another, and adding a residual connection. This encourages competition between channels and stabilizes training [10].

²ADE20K [17] is a large-scale scene parsing dataset with 150 semantic classes spanning indoor and outdoor scenes.

3.8 Swin Transformer

Swin Transformer [19] was introduced as a hierarchical vision transformer that partitions the image into non-overlapping windows and applies local self-attention within each window. To allow cross-window information flow, the windows are shifted between consecutive layers (shifted window attention). This design reduces the quadratic complexity of global self-attention to nearly linear in image size, while maintaining strong representation power. Moreover, the hierarchical structure progressively merges patches across stages, producing multi-scale feature maps analogous to CNN backbones—an advantage for dense prediction. Swin Transformer V2 (SwinV2) [20] builds directly on this design and introduces techniques to scale both model capacity and input resolution. Specifically, it stabilizes very deep or large models with residual post-normalization³, and adopts a scaled cosine attention, where attention logits are computed from cosine similarity with a learnable temperature and an additive relative position bias. It also proposes a log-spaced continuous relative position bias (CPB) that transfers more reliably from pre-training at low resolution to fine-tuning at higher resolutions. These innovations enable training models up to billions of parameters and images up to 1536×1536 , with strong results on classification, detection, and semantic segmentation benchmarks (e.g., ADE20K). In practice, both Swin and SwinV2 serve as drop-in backbones for dense prediction heads (e.g., UPerNet) and for shared-weight Siamese encoders in change detection. They offer a favorable accuracy–efficiency trade-off for high-resolution overhead imagery while retaining the multi-scale feature hierarchy central to Swin’s design.

3.9 DINO

Distillation with No Labels (DINO) [21] is a family of self-supervised learning methods based on knowledge distillation. It trains a student network to match the output distribution of a teacher network, where the teacher is updated as an exponential moving average of the student. Without using any labels, this formulation encourages the emergence of semantically meaningful representations: features cluster by object or scene type, and dense patch embeddings transfer well to downstream tasks such as detection and segmentation. DINO has been particularly successful with Vision Transformers, where self-distillation stabilizes training and yields strong global and local features. Building on this foundation, DINOv3 [22]

³Layer normalization is applied after each residual branch (post-norm) rather than before (pre-norm), normalizing the block output before it merges back to the main path. This curbs the accumulation of activation magnitudes in deep models and stabilizes training.

scales both data and model size and introduces Gram anchoring⁴ to maintain high-quality dense features during long pretraining. It relies on a curated large-scale mixture (e.g., the LVD-1689M⁵ web corpus) and a refined teacher–student formulation that combines global and local objectives. The resulting backbones transfer strongly to dense tasks such as semantic segmentation without task-specific fine-tuning. The report also demonstrates domain portability, including models trained for satellite imagery, supporting applicability to overhead scenes.

3.9.1 DINOv3-distilled ConvNeXt

Beyond ViT backbones, DINOv3 provides ConvNeXt students (from Tiny to Large) distilled from a ViT-7B teacher [23]. These ConvNeXt variants aim to retain the teacher’s dense feature quality while offering convolutional efficiency and quantization-friendliness; they cover a wide compute range and perform competitively on dense prediction benchmarks. Ready-to-use checkpoints (e.g., `convnext_base.dinov3_lvd1689m`) make these models practical as drop-in encoders for UPerNet or Siamese change-detection pipelines in remote sensing.

3.10 Uncertainty-Weighted Loss

Balancing multiple training objectives (e.g., pixelwise segmentation loss and auxiliary object-level loss) via fixed weights is brittle. Homoscedastic uncertainty weighting [24] learns those weights from task-dependent noise parameters and yields a composite loss derived from maximum-likelihood principles:

$$\mathcal{L} = \sum_{t=1}^T \frac{1}{2\sigma_t^2} \mathcal{L}_t + \log \sigma_t,$$

where \mathcal{L}_t is the per-task loss and $\sigma_t > 0$ is a learned scalar capturing task noise (aleatoric, but assumed input-independent). For classification terms (e.g., cross-entropy on logits), this corresponds to a temperature-scaled likelihood; for regression terms (e.g., ℓ_1/ℓ_2), it is the Gaussian (or Laplace) negative log-likelihood with unknown variance. The $\log \sigma_t$ term regularizes the solution, preventing trivial divergence of σ_t .

⁴In DINOv3, Gram anchoring is a refinement objective that aligns the student’s patch-wise Gram matrix (pairwise dot products of token features) with that of an early “Gram teacher,” helping preserve dense/local features over long training schedules.

⁵LVD-1689M is a curated web-scale dataset of 1.689 billion images selected with hierarchical clustering and balanced sampling; it serves as the pretraining corpus for DINOv3.

3.11 Evaluation metrics

The assessment of model performance in building damage detection requires metrics that effectively balance precision in identifying damaged structures with robustness to the strong class imbalance typically present in post-disaster datasets. Among the various metrics used in the literature, the F1-score is one of the most common, as it jointly captures the trade-off between precision and recall and remains less sensitive to skewed class distributions. It is defined as the harmonic mean of precision and recall:

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}},$$

where

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN}.$$

Here, TP , FP , and FN represent true positives, false positives, and false negatives, respectively. While precision reflects the ability to avoid false alarms, recall quantifies the model's sensitivity to actual damage. The F1-score thus provides a balanced summary of these two aspects, making it particularly suitable for applications where damaged buildings constitute only a small fraction of the total.

Chapter 4

Dataset

This chapter describes the dataset employed in the *AI for Earthquake Response Challenge*, focusing on its sources, structure, and the main challenges encountered in its preparation for machine learning tasks. We first present the raw data provided by the organizers, which consist of vVHR satellite images acquired from multiple commercial and open-access sensors, along with vector annotations of building footprints. Particular attention is given to the diversity of sensors, spectral modalities, and geographic regions covered, as well as to the issues that arise from this heterogeneity, such as differences in resolution, temporal gaps, and annotation imbalance.

4.1 ESA-Challenge Dataset

The dataset provided constitutes the raw material from which all subsequent processing was derived. It consists of a collection of VHR satellite images in GeoTIFF¹ format, acquired from multiple commercial and open-access sensors. The temporal coverage spans from 2019 to 2025 and includes six regions that were recently struck by major seismic events: Türkiye, Syria, Morocco, China, Myanmar, and Afghanistan. Alongside the imagery, the training data also includes vector annotations in GeoPackage (GPKG) format, containing building footprints labeled as either *damaged* or *undamaged*. These annotations form the basis for supervised learning, but they are affected by class imbalance and geographic skew, which present significant modeling challenges. The combination of heterogeneous image sources, diverse spectral modalities, and uneven annotation distributions defines

¹GeoTIFF is a geospatial raster data format based on the standard Tagged Image File Format (TIFF), which embeds georeferencing metadata to allow accurate placement of the image on the Earth's surface.

both the richness and the complexity of this dataset.

4.1.1 Image Sources

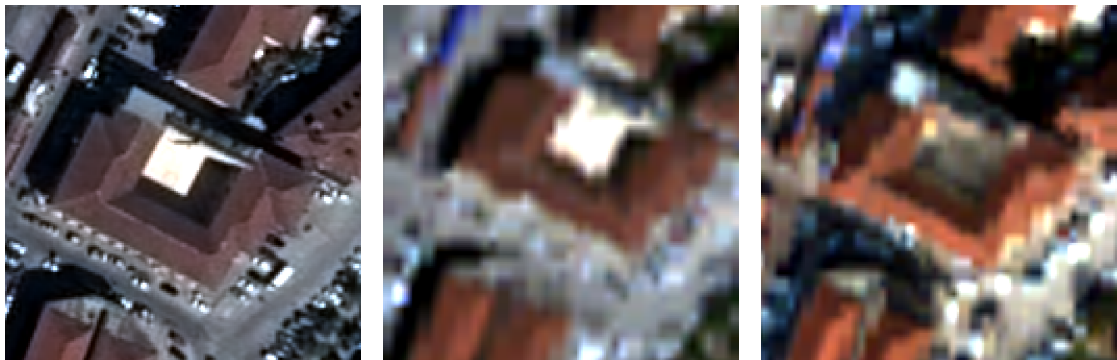
The images originate from a diverse set of optical satellites, each with specific spatial, spectral, and radiometric characteristics:

- Beijing SuperView Gaofen (Chinese VHR constellation)
- Pléiades-1A and Pléiades-1B (Airbus Defence and Space, France)
- WorldView-2 and WorldView-3 (Maxar Technologies, USA)
- Kompsat-3 and Kompsat-3A (Korea Aerospace Research Institute, South Korea)
- GeoEye-1 (Maxar Technologies, USA)
- Gaofen-2 PMS2 (China High-resolution Earth Observation Program)

Each image includes at least the three visible bands² (Red, Green, Blue). In some cases, additional bands are available, such as Near Infrared (NIR), Coastal, Yellow, Red Edge, and Panchromatic. The latter is particularly valuable since it enables pan-sharpening (see Section 3.1), thereby increasing the spatial resolution of multispectral images and improving the visibility of fine details relevant for building damage detection. The diversity of sources introduces variability not only in resolution but also in radiometric calibration and acquisition geometry. For example, Pléiades and WorldView satellites provide sub-meter panchromatic imagery (0.3–0.5 m GSD³) and 1.2–2 m in multispectral mode [25, 26], whereas Kompsat and Gaofen products are typically limited to 0.7–0.8 m in the panchromatic band and about 2.8–3.2 m in multispectral mode [27, 28]. Such heterogeneity is a key factor influencing both the visual detectability of building damage and the transferability of models across regions, as illustrated in Figure 4.1.

²Spectral bands correspond to specific portions of the electromagnetic spectrum (e.g., Red, Green, Blue, Near-Infrared, Coastal, Yellow, Red Edge). Each band provides complementary information about surface materials and conditions.

³Ground Sampling Distance indicates the distance between pixel centers measured on the ground. A smaller GSD corresponds to higher spatial resolution and finer detail in the imagery.



(a) Pléiades-1B, 2020

(b) WorldView-2, 2021

(c) WorldView-2, 2022

Figure 4.1: Antakya East, Türkiye — Comparison of pre-event images of the same building acquired by different very-high-resolution satellites at different times.

4.1.2 Train Dataset buildings

The training dataset also includes vector annotations in GPKG format, consisting of building footprints with binary labels: 0=*undamaged*, 1=*damaged*. In total, 6,408 buildings were annotated, data distributions are summarized in Tables 4.1. It is worth noting that the Afghanistan cities dominates the dataset in terms of

Area	City	Damaged	Undamaged	Total
Türkiye	Adiyaman	87	277	364
Türkiye	Antakya East	209	496	705
Türkiye	Gaziantep	10	410	420
Türkiye	Osmaniye	37	872	909
Türkiye	Kahramanmaraş	183	954	1,137
Syria	Latakia	19	113	132
Morocco	Marrakesh	11	768	779
Afghanistan	Zinda Jan	689	129	818
Afghanistan	Kharaba	334	160	494
China	Cogo	212	253	465
Myanmar	Wundwin	18	167	185

Table 4.1: Train Data (6,408 buildings in total)

damaged samples (over 1,000 instances), while other cities, such as Marrakesh or Latakia, contribute very few damaged buildings. This imbalance across geographic areas compounds the overall label imbalance, and raises the risk that models might learn context-specific biases rather than damage-relevant features, as illustrated in Figure 4.2.

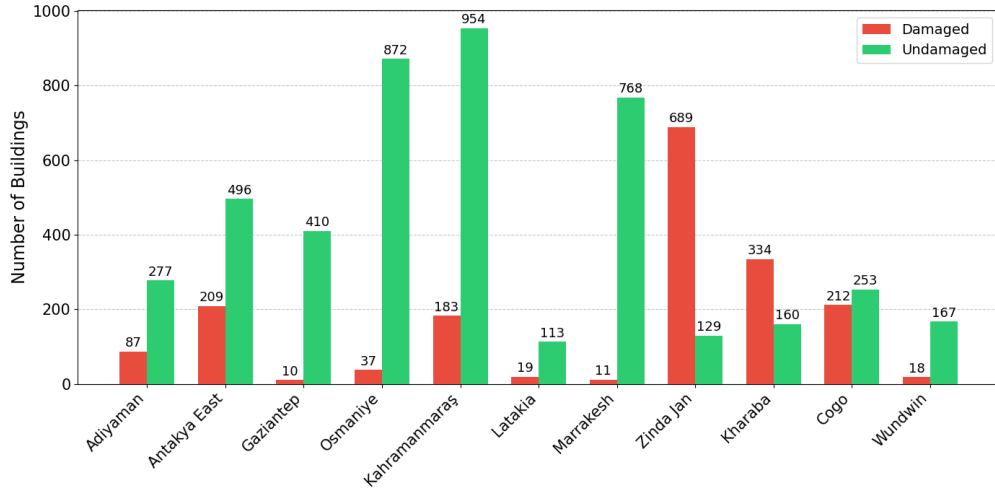


Figure 4.2: Histogram of damaged vs. undamaged buildings across cities. Note the strong dominance of undamaged samples and the over-representation of certain regions.

4.1.3 Test Dataset buildings

The test set comprises unlabeled building-footprint layers in GPKG format and it is used exclusively for submissions during the challenge phases. Data distributions are summarized in Tables 4.2 and 4.3. The considerable scale of the Phase 2

Area	City	Total
Türkiye	Adiyaman	10,987
Türkiye	Antakya East	15,334
Türkiye	Kahramanmaraş	33,429
China	Cogo	300

Table 4.2: Phase 1 Test Data (60,050 buildings in total)

evaluation (over 465,000 buildings) illustrates the operational ambition of the challenge: unlike typical academic benchmarks with thousands of samples, here the test sets approximate real-world mapping workloads where millions of buildings may need to be assessed within hours after a disaster.

4.1.4 Challenges Related to the Imagery

The satellite imagery itself introduced a number of difficulties that directly affected both human annotation and model training. These challenges stemmed from the

Area	City	Total
Türkiye	Antakya West	18,945
Myanmar	Mandalay	446,795

Table 4.3: Phase 2 Test Data (465,740 buildings in total)

heterogeneity of sources, acquisition conditions, and temporal coverage, and can be grouped into four main categories:

Heterogeneity of image sources

Multiple sensors produced images of different resolutions (see Figure 4.1), acquisition geometries, and spatial coverages. In some cases, a single city was covered by multiple images, requiring the combination of different sources to achieve complete coverage.

Sensor-related artifacts

In addition to differences in resolution and coverage, some images exhibited artifacts such as geometric distortions, striping, compression effects, or color anomalies due to sensor calibration issues (see Figure 4.3). These artifacts not only reduce the visual quality of the imagery but also risk misleading automated models, which may mistake such anomalies for real damage patterns.

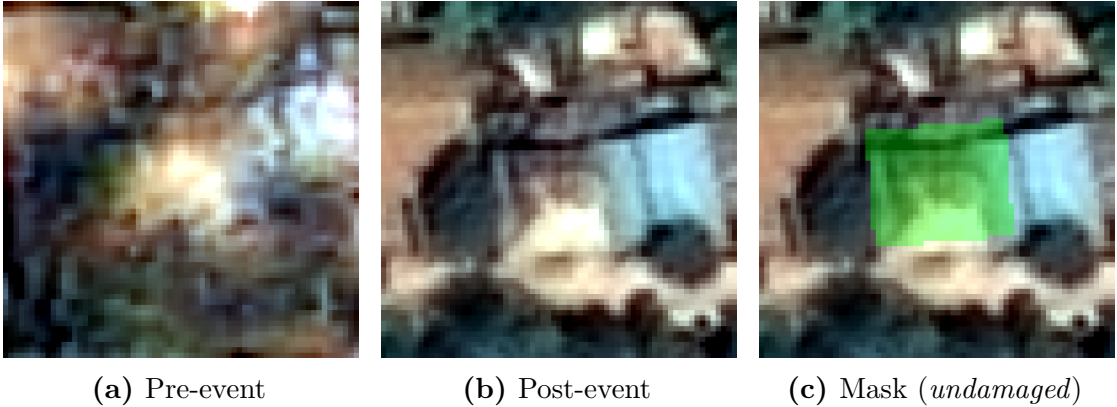
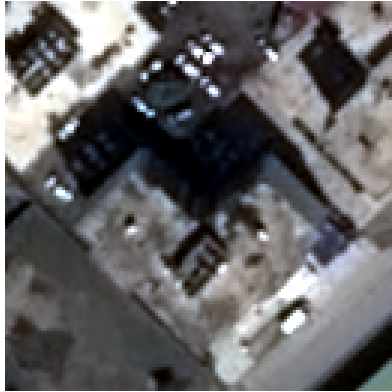


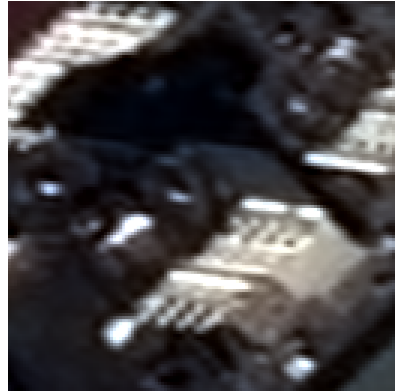
Figure 4.3: Wundwin, Myanmar — Example of a pre-event image where the low spatial resolution, combined with sensor-related color anomalies, makes it difficult to clearly distinguish the building footprint.

Misalignments

This misalignment arises from differences in acquisition times, sensors, and atmospheric or illumination conditions. Variations in viewing geometry (off-nadir angles⁴) and lighting can introduce residual shifts, particularly noticeable as parallax effects along roof edges and in shadowed areas, as illustrated in Figure 4.4.



(a) Pre-event (Pléiades-1A)



(b) Post-event (Pléiades-1B)

Figure 4.4: Osmaniye, Türkiye — Pre/post-event images acquired from different sensors showing variations in viewing geometry.

Temporal gaps

In some cases, the pre-event image was acquired several years before the earthquake (e.g., pre-disaster 2019 vs. post-disaster 2025). As a result, urban development unrelated to the earthquake (e.g., new construction or demolition) introduced confounding factors. In practical terms, this means that some “damaged” labels may correspond to buildings that were not even present in the pre-disaster image, or conversely, that newly constructed undamaged buildings might be misclassified as damaged simply because they appear only in the post-event image. Another possible source of noise is that buildings may show signs of wear, aging, or pre-existing partial degradation that are not related to the disaster (see Figure 4.5) but could still be interpreted as earthquake damage by the model. Handling this type of temporal noise is a non-trivial open problem for EO-based change detection.

⁴In Earth observation, nadir is the direction straight down from the sensor to the ground; off-nadir means viewing at a tilt, which can introduce parallax, foreshortening, and apparent shifts.

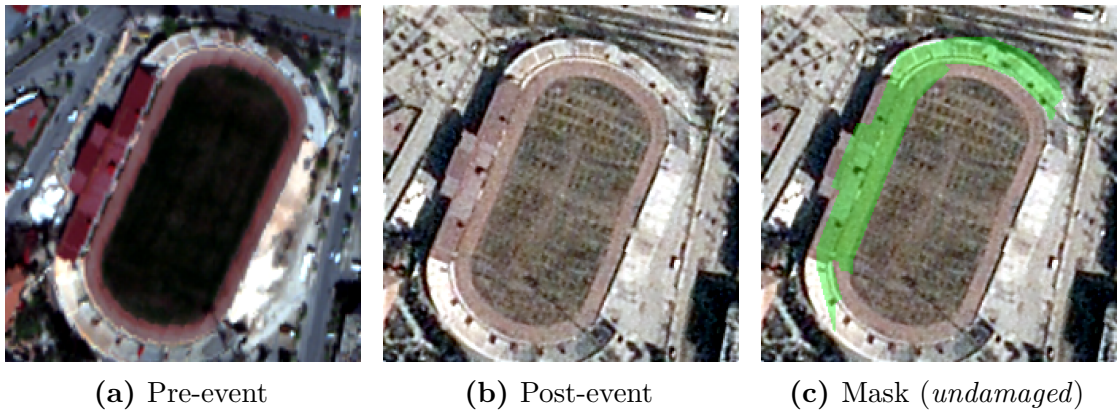


Figure 4.5: Kahramanmaraş, Türkiye — Example of a building that appears slightly degraded in the post-event image. However, according to the ground-truth mask it is labeled as *undamaged*, since the visible deterioration is due to pre-existing wear rather than earthquake damage. In the inference phase, such cases may nonetheless be predicted as damaged.

Unequal spectral availability

Panchromatic bands were available only for a subset of pre-event images, while many post-event images were limited to lower-resolution multispectral data, resulting in lower sharpness for damage assessment (see Figure 4.6).

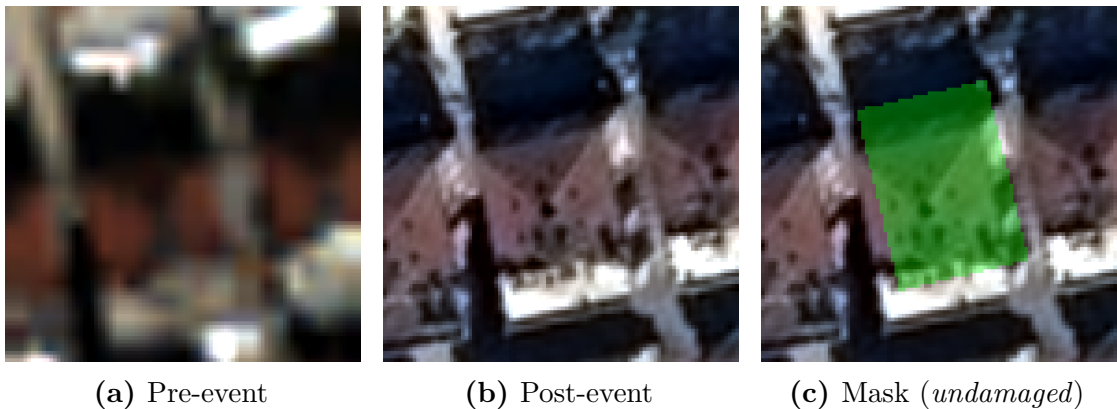


Figure 4.6: Antakya East, Türkiye — Example of a building where the pre-event image lacks a panchromatic band and the available multispectral data is of relatively low quality, resulting in reduced sharpness for damage assessment.

Incomplete coverage in Phase 2

Approximately 80,000 buildings in the second-phase test set lacked pre-event imagery, leaving only post-event images for evaluation.

4.1.5 Challenges Related to Annotations

Beyond the intrinsic difficulties of satellite imagery, the quality and nature of the annotations introduced additional sources of complexity. In particular, we identified three main challenges:

Severe class imbalance

Of the total annotated buildings, only 1,809 ($\approx 28\%$) were labeled as damaged, while the remaining majority were undamaged. This imbalance complicates model training, as most models tend to overfit toward the majority class (see Table 4.1 and Figure 4.2).

Train-test distribution shift

The training set contained a relatively small number of annotated buildings from a limited number of cities, while the test sets included hundreds of thousands of buildings, often located in previously unseen areas. This substantially increased the difficulty of generalization to new geographic domains and urban morphologies. For instance, in the case of Myanmar, the training set provided only 185 annotated buildings from Wundwin, whereas the Phase 2 test set required predictions for more than 446,000 buildings from Mandalay, a city with markedly different urban structure and architectural styles.

Annotators Subjectivity

Annotators must infer structural damage from roof-level imagery, which may obscure internal collapse or partial damage. This subjectivity can lead to label noise, particularly in borderline cases (e.g., cracked roofs vs. complete collapse).

Chapter 5

Methodology

This chapter details the end-to-end pipeline for detecting building damage from multi-temporal very-high-resolution satellite imagery. Building on the previous chapter’s data description, we first curate a refined dataset that addresses heterogeneous sensor resolutions, misalignment between imagery and vector annotations, and incomplete city coverage. We then formalize a preprocessing pipeline that standardizes patch creation and normalization, and introduces targeted augmentations to improve robustness to radiometric variability and residual co-registration errors. On top of this data foundation, we investigate models that pair modern convolutional encoders with U-Net and UPerNet decoders, including Siamese-like configurations that better exploit ImageNet pretraining by processing pre and post-event imagery separately. We next describe the training strategy—combining pixel and building-level supervision, class-imbalance mitigation via dataset-level positive weighting, and a hybrid uncertainty-weighted loss—together with optimization choices and fine-tuning regimes.

5.1 Dataset Preprocessing

In order to address the challenges described above, a series of processing steps were applied to produce a consistent and reliable dataset, minimizing inconsistencies and noise while maximizing usability for model training. Specifically, we established a resolution hierarchy, applied alignment to the GPKG, performed patch and mask generation, and defined both a benchmark and a deployment dataset.

Resolution Hierarchy. For each available image, we associated it with the corresponding GPKG that it covered. Thus, each GPKG polygon set was linked to one or more images belonging to different satellite passages. A ranking hierarchy was then established, ordering the images from highest to lowest spatial resolution,

separately for pre and post-event data. The presence of a panchromatic band (available only in pre-event imagery) was also considered as an important factor in the ranking, since it enables pan-sharpening.

Alignment to the GPKG. Many images were misaligned with respect to the building footprints in the GPKG annotations, mainly due to differences in sensor acquisition geometry. To correct this, we performed manual alignment by identifying tie points across the imagery and the GPKG polygons. When the misalignment consisted of a simple shift, a linear correction was applied, while in cases with small geometric distortions, a polynomial warping algorithm was employed. This ensured accurate correspondence between imagery and building annotations.

5.1.1 Patch and Mask Generation

Once the resolution hierarchy was established and the images aligned, the effective dataset generation was carried out:

Building coverage. For each building in the GPKG, suitable pre and post-event images were selected, ensuring complete coverage of the building polygon. Buildings without pre-event imagery were discarded at this stage to avoid introducing noise or biased predictions during training. For evaluation during the challenge submission (Phase 1 and Phase 2), those buildings were assigned as undamaged.

Spectral stacking. The three RGB bands were stacked to form a three-channel image. If available, the panchromatic band was fused with the RGB channels using stationary wavelet transform (SWT) pan-sharpening (see Section 3.1). This method was chosen because wavelet-based fusion preserves both high-frequency spatial details from the panchromatic band and low-frequency spectral information from the multispectral channels, reducing spectral distortion compared to traditional intensity-hue-saturation (IHS) or principal component analysis (PCA) approaches. Preliminary visual inspection confirmed that pan-sharpened images facilitated clearer identification of building outlines and roof structures, both by human annotators and downstream models (see Fig. 5.1). However, pan-sharpening also risks introducing artifacts (e.g., spectral distortion at sharp edges), meaning its impact had to be carefully validated during training.

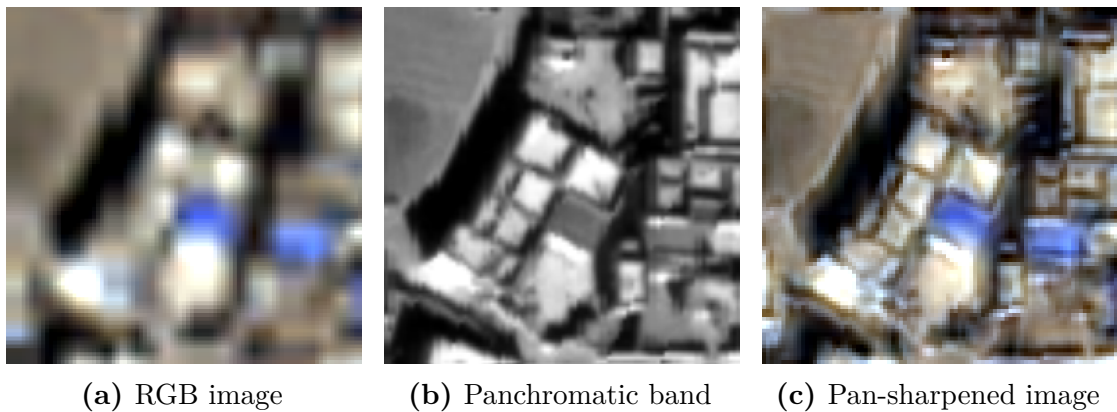


Figure 5.1: Cogo, China — Comparison of input imagery: (a) original RGB composite, (b) high-resolution panchromatic band, and (c) fused pan-sharpened image produced via SWT pan-sharpening.

Context padding. Around each building polygon, a context window of 20 pixels per edge was added, and the corresponding pre and post-event patches were cropped.

Mask generation. For the training set, a binary mask of the same size as the patch was created: pixels inside the building polygon were assigned the label (0 = *undamaged*, 1 = *damaged*), while background pixels were assigned a value of 255 (*ignored*). Figure 5.2 illustrate three canonical examples of pre/post/mask triplets, which we take as representative of the standard format used throughout the dataset.



Figure 5.2: (a) Pre-event; (b) post-event; (c) ground-truth mask. In the project, ground-truth masks are encoded as 0=undamaged, 1=damaged, 255=background; in images above the mask is visualized as an overlay on the post-event image (red=damaged, green=undamaged).

5.1.2 Benchmark Dataset

The Benchmark Dataset was designed to replicate the evaluation setting of the ESA challenge, enabling controlled and reproducible experimentation with both in-distribution (ID) and out-of-distribution (OOD) domains. All building patches (pre, post, and mask) were grouped by city and GPKG identifier. This dataset was explicitly split (stratified¹ by label to mitigate class imbalance) into four subsets:

- **Out-of-Distribution (OOD) set** – consisting of all buildings from Cogo (China) and Myanmar, entirely excluded from training/validation, to simulate the geographic generalization challenge of the ESA competition. Myanmar was selected because it was also part of the hidden test set used in Phase 2 of the official challenge, ensuring direct comparability with that evaluation setting. Conversely, Cogo was included to balance the overall size of the OOD split, as its exclusion removed a moderate number of samples from training while maintaining a similar scale between the ID and OOD domains.

And with the remaining data:

- **Training set (70%)** – used for model fitting.
- **Validation set (15%)** – used for hyperparameter tuning.
- **In-Distribution (ID) set (15%)** – held-out data to report final scores under controlled conditions.

Although derived from the same source imagery and annotations as the deployment setup described next, this configuration explicitly isolates ID and OOD data to mirror the hidden test conditions of the challenge in its two phases. The Benchmark Dataset was primarily used to compare experimental variants under consistent and repeatable conditions including ablation studies, encoder–decoder combinations, and fusion strategies, so that performance differences could be attributed to model design rather than data variability. This setup thus serves as the reference framework for reporting and discussing quantitative results throughout this work.

5.1.3 Deployment Dataset

In parallel, a second configuration, termed the Deployment Dataset, was created using the same underlying imagery and building annotations as the Benchmark Dataset, but organized to better reflect an operational or production-oriented use

¹Stratification ensures that the proportion of damaged and undamaged buildings remains consistent across all dataset splits.

case. Here, all available data from the same cities were pooled together and divided into only two subsets:

- **Training set (80%)** – comprising the majority of available buildings to maximize exposure to diverse examples.
- **Validation set (20%)** – held out for model selection and monitoring.

Unlike the Benchmark Dataset, this configuration omits explicit ID/OOD test partitions, allowing a larger portion of the data to be used for model optimization. It was adopted to train the final submission models, leveraging more data to improve generalization and stability before evaluation on the challenge’s hidden test sets. This dataset therefore prioritizes robustness and completeness over controlled comparability, serving as the final step between experimental analysis and deployment-ready models.

With the datasets thus standardized and organized for both benchmarking and deployment, the next step involved defining the learning architectures capable of exploiting them effectively. The following section introduces the model design adopted in this work, describing how different encoder–decoder combinations were structured and compared to identify the most suitable configuration for building-level damage classification.

5.2 Model Architectures

The design of the proposed models followed a modular strategy, combining alternative encoder–decoder pairs and different fusion schemes for handling pre and post-event imagery. This section outlines the architectural components adopted and the fusion mechanisms used to integrate temporal information. Theoretical details of the individual architectures are discussed in Chapter 3, while here the focus is on their practical configuration and role within the experimental framework.

5.2.1 Decoders and Encoders

Before discussing the specific backbones, it is useful to outline the architectural paradigms that guided the design of our models. Two well-established segmentation architectures were considered:

- **U-Net.** Used as the baseline configuration, providing a lightweight and efficient framework well suited for large-scale inference.
- **UPerNet.** Explored as a richer alternative capable of fusing multi-scale contextual information through pyramid pooling and feature aggregation.

In this work, these decoding schemes were paired with alternative encoder backbones to evaluate how feature extraction capacity and pretraining strategy influence model performance. The following encoders were considered:

- **ResNet-50.** Served as a conventional baseline with stable optimization and moderate complexity.
- **ConvNeXtV2.** Represented modern convolutional hierarchies with improved scalability and efficiency.
- **DINOv3-distilled ConvNeXt.** Included to test self-supervised pretraining and its transferability to post-disaster domains.
- **SwinV2.** A transformer-based alternative capturing long-range spatial dependencies and texture variations.

This modular formulation enabled a systematic combination of backbones and decoders, from the standard U-Net with ResNet-50 baseline to advanced ConvNeXtV2, DINOv3-distilled ConvNeXt, and SwinV2 encoders coupled with UPerNet decoders. Through this setup, the study assessed how backbone capacity, pretraining paradigm, and decoding complexity jointly affect generalization across unseen cities and sensors.

5.2.2 Fusion Strategies

A central design choice in the training pipeline concerned how to represent and feed the pre and post-event images into the network. Since both inputs are needed to capture temporal differences, two alternative strategies were investigated: one that directly concatenates them into a joint representation, and another that processes them separately through a shared encoder before fusing their features.

1. **Early concatenation.** Pre and post-event images were stacked along the channel dimension and passed jointly through the encoder and decoder.
2. **Siamese-like encoder.** Pre and post-event images are processed by the same encoder; features f_{pre} and f_{post} are fused before being passed to the decoder (see Figure 5.3). Three fusion schemes were considered:

- *Absolute difference:*

$$f = |f_{\text{pre}} - f_{\text{post}}| \quad (5.1)$$

This symmetric formulation captures the magnitude of change regardless of direction, emphasizing regions with strong visual discrepancy between pre and post-event features. It is particularly suitable when change intensity matters more than the specific direction of variation.

- *Signed difference*:

$$f = f_{\text{pre}} - f_{\text{post}} \quad (5.2)$$

This asymmetric variant preserves the sign of feature variation, allowing the network to learn directional cues (e.g., appearance loss due to destruction versus appearance gain due to debris accumulation). While more sensitive to feature misalignment, it can provide richer semantic signals in well-registered data.

- *Late concatenation*:

$$f = [f_{\text{pre}} \parallel f_{\text{post}}] \quad (5.3)$$

Here, both representations are concatenated along the channel axis, letting the subsequent decoder layers learn how to compare them implicitly. This configuration maximizes the information available to the decoder but doubles the channel dimensionality, increasing computational and memory requirements.

These fusion approaches allowed a more effective exploitation of pretrained encoders, which are designed for three-channel RGB inputs, while explicitly modeling temporal differences between pre- and post-event observations.

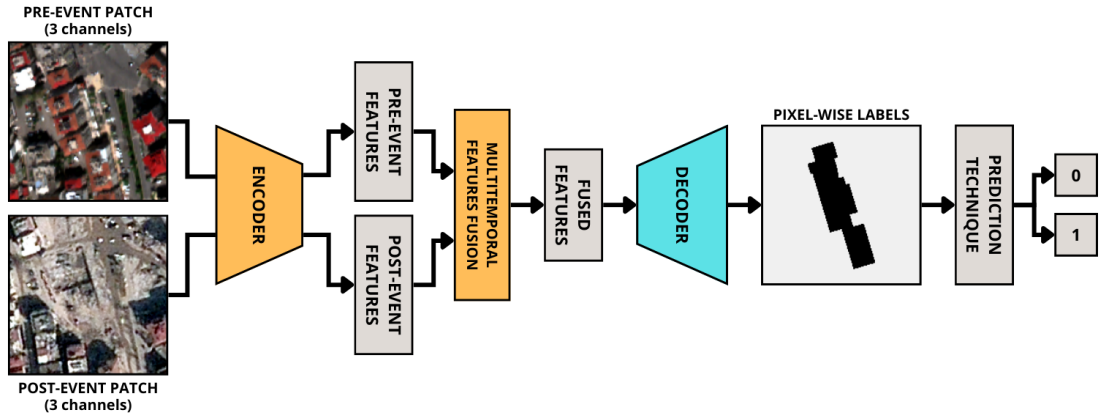


Figure 5.3: Siamese-like encoder–decoder architecture.

5.3 Training Strategy

This section details the end-to-end learning pipeline adopted to train the proposed architectures for building damage detection. Training begins with the application of targeted data augmentations and normalization procedures designed to improve generalization and robustness to acquisition variability. We then describe the

optimization setup used in the experiments, including the loss formulations that supervise learning at both pixel and building level, the fusion techniques that define how pre and post-event information is represented and adapted during training and fine-tuning strategies. Finally, we outline the prediction rules applied at inference time to derive building-level decisions from pixel-wise probabilities.

5.3.1 Data Augmentations

Given the standardized and aligned pre/post patches described in Section 5.1, the first step of the training pipeline is to increase data diversity and reduce overfitting through on-the-fly augmentations applied consistently to the paired inputs (and masks). These transformations target the dominant nuisance factors in this problem, such as viewing geometry, illumination, atmospheric effects, and residual co-registration errors, so that subsequent supervision (Section 5.3) acts on features that are robust to such variability.

- **Geometric transformations:** random rotations, horizontal and vertical flips, translations, and isotropic scaling were applied simultaneously to the pre-event, post-event, and mask patches to preserve spatial alignment. In addition, coarse dropout was employed to randomly occlude rectangular regions, simulating missing data or cloud cover and forcing the network to rely on broader contextual cues rather than isolated details. These operations collectively encouraged invariance to scene orientation, minor positional shifts, and local occlusions.
- **Photometric transformations:** brightness, contrast, gamma, and color-jitter adjustments were independently applied to the pre- and post-event images. Additional effects such as random shadows, random fog, and Gaussian blur were included to mimic illumination changes, atmospheric scattering, and sensor-induced blur. Together, these transformations simulated variability due to differences in sensors, acquisition time, or environmental conditions, thereby improving robustness to radiometric inconsistencies.
- **Residual misalignment simulation:** despite the manual alignment step described in Chapter 4, small random shifts and slight rotations were introduced to pre-event patches only. This emulated inevitable misalignments between multi-temporal acquisitions and trained the models to be less sensitive to imperfect co-registration.

After the above transformations, all patches were standardized as follows:

$$\mu = \frac{1}{N} \sum_{i=1}^N x_i, \quad \sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2}, \quad x' = \frac{x - \mu}{\sigma}, \quad (5.4)$$

where x_i denotes the intensity of each valid pixel in the training set, and N is the total number of such pixels. The computed mean μ and standard deviation σ were stored and reused to ensure consistent normalization across the training, validation, and test sets. Finally, all inputs were resized to a fixed resolution of 256×256 pixels before being fed into the models. Image patches were resampled using area-based interpolation to minimize aliasing, while binary masks were resized using nearest-neighbor interpolation to preserve discrete label boundaries.

5.3.2 Loss Functions

Training was formulated as a binary classification problem, where the objective is to discriminate between damaged and undamaged building pixels. The basic loss used throughout this work was the binary cross-entropy (BCE), defined as

$$\mathcal{L}_{BCE}(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^N [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \quad (5.5)$$

where $y_i \in \{0,1\}$ denotes the ground-truth label of pixel i , $\hat{y}_i \in [0,1]$ the predicted probability, and N the number of valid pixels.

Formulations

On top of the BCEWithLogits formulation, different strategies were implemented to exploit both pixel-level and building-level supervision:

- **Pixel-wise loss.** The loss was applied to valid pixels only (mask value $\neq 255$), using a positive class weight:

$$w_{pos}^{pixel} = \frac{N_{neg}^{pixel}}{N_{pos}^{pixel}},$$

$$\mathcal{L}_{pixel} = -\frac{1}{N} \sum_{i \in \mathcal{V}} \left[w_{pos}^{pixel} y_i \log \sigma(z_i) + (1 - y_i) \log(1 - \sigma(z_i)) \right], \quad (5.6)$$

where \mathcal{V} denotes the set of valid pixels, and $\sigma(z_i)$ is the sigmoid activation applied to the model output at pixel i . The positive class weight w_{pos}^{pixel} is defined as the ratio between the total number of valid negative and positive pixels across all buildings in the dataset, where N_{pos}^{pixel} and N_{neg}^{pixel} represent, respectively, the total counts of valid positive and negative pixels. This formulation ensures that damaged pixels (typically underrepresented) receive proportionally higher importance during optimization. In addition, we evaluated a *soft Dice* term which directly optimizes overlap between positive predictions and ground

truth while being robust to class imbalance. We report results for (i) *BCE* ($\mathcal{L}_{pixel} = \mathcal{L}_{pixel-BCE}$) and (ii) *BCE+Dice*, using a simple convex combination:

$$\mathcal{L}_{pixel} = \lambda \mathcal{L}_{pixel-BCE} + (1 - \lambda) \mathcal{L}_{pixel-Dice}, \quad \lambda \in [0,1], \quad (5.7)$$

with λ fixed in our experiments (default $\lambda = 0.5$ unless stated otherwise).

- **Building-level loss.** To extend supervision from individual pixels to entire building instances, predictions were aggregated within each polygon via mean probability and compared to the building label:

$$\hat{y}_b = \frac{1}{|P_b|} \sum_{i \in P_b} \sigma(z_i), \quad w_{pos}^{building} = \frac{N_{neg}^{building}}{N_{pos}^{building}},$$

$$\mathcal{L}_{building} = -\frac{1}{B} \sum_{b=1}^B \left[w_{pos}^{building} y_b \log \hat{y}_b + (1 - y_b) \log(1 - \hat{y}_b) \right], \quad (5.8)$$

where P_b is the set of pixels belonging to building b , and \hat{y}_b denotes its aggregated damage probability. The positive class weight $w_{pos}^{building}$ compensates for class imbalance at the instance level and is computed as the ratio between the total number of undamaged and damaged buildings in the dataset, where $N_{pos}^{building}$ and $N_{neg}^{building}$ indicate the number of damaged and undamaged buildings, respectively. Also this weight was precomputed and stored to ensure that each building contributes proportionally to the training signal, preventing dominance of the majority (undamaged) class.

- **Uncertainty-weighted hybrid loss.** In addition to the individual formulations, a combined strategy was explored following Kendall *et al.* (see Section 3.10), the two terms are combined with learnable task uncertainties:

$$\mathcal{L} = \frac{1}{2\sigma_{pixel}^2} \mathcal{L}_{pixel} + \frac{1}{2\sigma_{building}^2} \mathcal{L}_{building} + \log(\sigma_{pixel}\sigma_{building}). \quad (5.9)$$

where σ_{pixel} and $\sigma_{building}$ are learnable parameters that automatically balance the contributions of the two terms during training.

5.3.3 Fine-Tuning Strategies

Since all encoders were initialized from models pretrained on large-scale datasets, a key question was how to best adapt these representations to the task of post-disaster damage assessment. To this end, different fine-tuning strategies were explored, reflecting varying degrees of reliance on the pretrained features versus task-specific adaptation:

- **Frozen encoder.** In this setting, all encoder weights were kept fixed and only the decoder parameters were updated. This strategy drastically reduces the number of trainable parameters and lowers computational requirements, which is particularly useful when working with limited labeled data or constrained hardware. It also allows testing the transferability of generic ImageNet features to the disaster-response domain without any task-specific adaptation.
- **Partial unfreezing.** Here, only the deeper layers of the encoder (closer to the semantic representations) were updated, while the shallower layers (capturing low-level textures and edges) were kept frozen. The motivation is that low-level filters learned from natural images often transfer well to remote sensing data, whereas deeper layers may need adaptation to capture domain-specific cues such as rubble patterns, roof collapses, or urban texture variations. This strategy provides a balance between computational efficiency and task specialization.

5.3.4 Prediction Strategies

Since the models produced pixel-level probabilities, a post-processing step was required to aggregate these into building-level predictions. Two complementary decision rules were evaluated, each reflecting a different balance between sensitivity to small damaged regions and robustness against noisy pixel predictions:

- **Mean probability.** In this strategy, the average probability of all valid pixels inside a building polygon was computed, and the building was classified as damaged if this value exceeded a threshold τ . This rule is simple and computationally efficient, providing a smooth aggregation of pixel-level information. It reduces the influence of isolated noisy pixels, but may underestimate damage in cases where only a small portion of the building is affected.
- **Minimum-area rule.** In this strategy, a building was classified as damaged if at least a fraction α of its valid pixels (e.g., $\alpha = 0.05$) exceeded the fixed threshold $\tau = 0.5$. This approach is more sensitive to localized damage, since even a small but consistent damaged area can trigger a positive prediction. At the same time, it prevents spurious single-pixel activations from dominating the decision, requiring a minimal spatial extent of evidence to confirm damage.
- **Any rule.** In this strategy, a building was classified as damaged if at least one of its valid pixels exceeded the fixed threshold $\tau = 0.5$. This rule maximizes sensitivity, ensuring that even very small damaged regions are detected, and thus favors recall—an especially critical property in disaster assessment, where missing damaged buildings can have severe consequences. However, it also

increases the risk of false alarms, since a single noisy or spurious pixel can trigger a positive prediction.

Chapter 6

Results

This chapter reports the outcomes of the experimental evaluation, following the methodology outlined in Chapter 5 and model performance is assessed using the evaluation criteria discussed in Section 3.11, which ensure comparability with standard practices in damage estimation literature. We first establish performance on the benchmark dataset using a standard baseline model to provide a reference point. We then progressively analyze model variants, ablation experiments, and prediction strategies. Finally, each set of quantitative results is complemented by qualitative assessments and an error analysis, discussed alongside every comparison.

6.1 Implementation Details

Training was conducted for a maximum of 200 epochs. The AdamW optimizer was employed, as it provides improved generalization compared to the standard Adam algorithm. A OneCycleLR learning-rate scheduler (Figure 6.1) was used to accelerate convergence during the initial epochs while preventing premature stagnation in local minima. Batch size and learning rate were systematically varied to balance convergence stability with GPU memory constraints. In addition, the learning rate was initialized using the *LR finder* utility of PyTorch Lightning, which identifies a range of values yielding stable and efficient convergence. The final value was selected as a compromise between stability (avoiding divergence) and sufficient aggressiveness to escape local minima. In all cases, gradient clipping and weight decay regularization were additionally employed to stabilize training and improve efficiency. All experiments were implemented using the PyTorch Lightning framework, which provided a structured and modular training pipeline and facilitated reproducibility. Training was performed on an NVIDIA GeForce RTX 2080 Ti (11 GB VRAM). Runtime varied significantly across models, from approximately 5 hours for the ResNet-50 U-Net baseline to around 36 hours for the

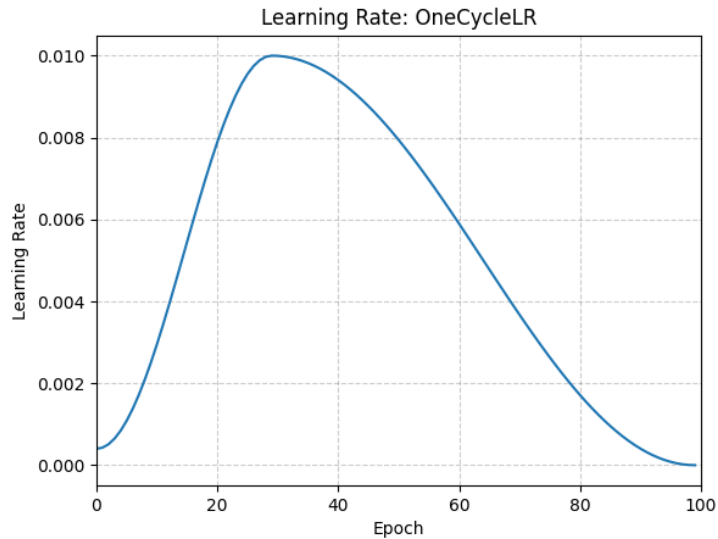


Figure 6.1: Example of the learning rate evolution over the training epochs following the OneCycleLR scheduling policy.

fine-tuned DINOv3 ConvNeXt-Large. For SwinV2 experiments, we considered two window configurations ($window=8$ and $window=16$), trading off detail resolution and memory/computation: smaller windows favor fine roof patterns and debris textures; larger windows provide broader context for block-scale damage cues. To ensure comparability, a fixed random seed was used throughout all experiments, controlling for randomness in data shuffling, and augmentation. The input data were stored and accessed in `.npy` format to minimize disk I/O overhead and memory consumption during training, enabling faster loading and more efficient GPU utilization.

6.2 Benchmark Dataset Results

To assess the generalization ability of the proposed methods, we evaluated models on both in-distribution (ID) and out-of-distribution (OOD) test sets derived from the benchmark dataset.

6.2.1 Baseline

As a starting point, we implemented a standard ResNet50-U-Net architecture, which combines a convolutional encoder for hierarchical feature extraction with a symmetric decoder for dense spatial prediction. The ResNet50 backbone serves as

a robust feature extractor pretrained on large-scale natural image datasets, while the U-Net decoder progressively restores spatial detail through upsampling and skip connections, enabling precise localization of damaged areas within building footprints. In this baseline configuration, the encoder was kept frozen throughout training, ensuring that only the decoder parameters were updated. Input fusion was performed by concatenating pre and post-event images at the pixel level before feature extraction (early-concatenation), so that the encoder received a six-channel combined representation. Training optimized a pixel-wise BCE loss, directly supervising each valid pixel within the building masks. Optimization followed the AdamW strategy with a weight decay of 1×10^{-2} , using a learning rate of 2×10^{-4} (determined with LR finder utility) for the decoder while keeping the encoder frozen. Training was carried out for 200 epochs with a batch size of 64, under a OneCycleLR scheduler to dynamically adapt the learning rate. This baseline provides a fair and reproducible reference point against which all subsequent architectural and methodological improvements can be compared. The results, reported separately for the ID and OOD test sets, are summarized in Table 6.1.

Test set	F1-score	Precision	Recall
In-distribution (ID)	0.6832	0.6078	0.7799
Out-of-distribution (OOD)	0.5921	0.4642	0.8174

Table 6.1: Baseline performance of ResNet50–U-Net on ID and OOD test sets. Scores are reported as global building-level F1, precision, and recall. The model was trained using early concatenation fusion technique, a pixel-wise loss, and the mean-probability rule for building-level prediction.

The baseline results highlight several key aspects of the task difficulty. On the ID test set, the model achieves a balanced F1-score of 0.68, with moderately high recall but lower precision, indicating a tendency to over-predict damage within known domains. The OOD performance drops to an F1 of 0.59, revealing a notable generalization gap across unseen cities. Nevertheless, the recall remains high (0.82), suggesting that the model retains sensitivity to damaged regions even under distribution shifts, albeit at the cost of more false positives. Overall, these findings confirm that while a simple concatenation-based ResNet50–U-Net can effectively capture damage cues when domain conditions are similar, its transferability to new urban contexts is limited, justifying the need for improved encoder adaptation and more robust fusion strategies explored in subsequent sections.

6.2.2 Alternative Encoders and Decoders

To evaluate the impact of architectural choices, we extended the baseline configuration by replacing the ResNet50 encoder and U-Net decoder with more advanced counterparts introduced in Chapter 5. Specifically, we considered the ConvNeXtV2, DINOv3-distilled, and SwinV2 architectures as alternative encoders, each coupled with the UPerNet decoder, which integrates multi-scale features through a feature pyramid and global context via a pyramid pooling module (an overview of the model sizes is provided in Table 6.2). Unless otherwise specified, all the encoders were used in their base versions. These experiments aim to assess whether stronger visual backbones and a more context-aware decoder can enhance both in-distribution and cross-domain generalization performance (see Table 6.3).

Encoders	Parameters
ResNet50	23 M
ConvNeXtV2 Base	88.7 M
ConvNeXtV2 Large	198 M
DINOv3-distilled ConvNeXt Base	87.6 M
DINOv3-distilled ConvNeXt Large	196.2 M
SwinV2 Base	87.9 M

Table 6.2: Approximate number of trainable parameters for the evaluated encoder architectures. The listed values refer to the base and large variants used in the experimental analysis.

Encoder ¹	Decoder	ID Test			OOD Test			AVG F1
		F1	p	r	F1	p	r	
ResNet50	U-Net (baseline)	0.6832	0.6078	0.7799	0.5921	0.4642	0.8174	0.6376
CNXv2	UPerNet	0.7162	0.6193	0.8491	0.6602	0.5273	0.8826	0.6882
DINOv3	UPerNet	0.7479	0.6804	0.8302	0.6010	0.4781	0.8087	0.6744
SwinV2-w8	UPerNet	0.6895	0.6302	0.7610	0.5066	0.4482	0.5826	0.5980
SwinV2-w16	UPerNet	0.7188	0.6133	0.8679	0.6514	0.5024	0.9261	0.6851

Table 6.3: Comparison of alternative encoder–decoder configurations. Scores are reported as building-level F1, precision, and recall. The model was trained using early concatenation fusion technique, pixel-wise loss and the mean-probability rule for building-level prediction.

¹Unless otherwise specified, all encoders are considered completely frozen and in their base version.

Results in Table 6.3 highlight the benefits of employing more expressive and modern encoders. Across both test regimes, models equipped with UPerNet decoders systematically outperform the ResNet50–U-Net baseline, confirming the importance of multi-scale and global context integration for post-disaster damage assessment. Among the tested variants, DINOv3-distilled + UPerNet achieves the best overall ID performance ($F1 = 0.75$), combining strong discriminative capacity with a balanced precision–recall trade-off. This suggests that self-supervised visual features, distilled from large-scale pretraining, transfer effectively to the damage detection domain. Conversely, ConvNeXtV2 + UPerNet yields the most robust OOD performance ($F1 = 0.66$), outperforming all others in cross-city generalization. Its hierarchical convolutional structure and efficient feature scaling appear to better preserve transferability under domain shifts. The SwinV2 variants show mixed results: while the wide (window size 16) version maintains competitive recall, both configurations tend to underperform in precision. Overall, these results indicate that encoder choice substantially influences both within-domain accuracy and out-of-domain robustness, and that architectures combining strong visual priors with structured multi-scale decoding (such as DINOv3–UPerNet and ConvNeXtV2–UPerNet) provide the most favorable balance for real-world deployment.

6.2.3 Fusion strategies

To investigate how the integration of pre and post-event information affects model performance, we compared multiple fusion strategies introduced in Chapter 5. Specifically, we evaluated four configurations for combining temporal inputs: *early concatenation*, *difference*, *absolute difference*, and *late concatenation*. These strategies differ in where and how the two image streams are merged—either before feature extraction (early), through direct pixel or feature-level differencing, or after independent encoding paths (late). The goal of this comparison is to determine which fusion scheme best captures structural changes while preserving robustness across domains (see Table 6.4).

Results in Table 6.4 demonstrate that the choice of fusion strategy has a pronounced impact on both in-domain accuracy and cross-domain robustness. For all encoders, the *late concatenation* approach consistently achieves the best ID performance, with up to 0.78 F1 for ConvNeXtV2 and 0.76 for DINOv3. This strategy allows the encoder to process pre and post-event images independently before feature-level integration, enabling the model to preserve semantic representations for each time step and fuse them in a more discriminative manner at the decoder stage. In contrast, the *early concatenation* method shows slightly lower ID accuracy but often stronger OOD generalization, particularly for ConvNeXtV2 ($F1 = 0.66$) and SwinV2-w16 ($F1 = 0.65$). This suggests that early fusion encourages the model to

Encoder ¹	Fusion	ID Test			OOD Test			AVG F1
		F1	p	r	F1	p	r	
CNXv2	early-concat	0.7162	0.6193	0.8491	0.6602	0.5273	0.8826	0.6882
CNXv2	diff	0.6951	0.7260	0.6667	0.4645	0.5104	0.4261	0.5798
CNXv2	abs-diff	0.6797	0.7075	0.6541	0.3333	0.3363	0.3304	0.5065
CNXv2	late-concat	0.7758	0.7485	0.8050	0.6287	0.5735	0.6957	0.7023
DINOv3	early-concat	0.7479	0.6804	0.8302	0.6010	0.4781	0.8087	0.6745
DINOv3	diff	0.7143	0.6780	0.7547	0.4778	0.4161	0.5609	0.5961
DINOv3	abs-diff	0.6905	0.6554	0.7296	0.4855	0.4161	0.5826	0.5880
DINOv3	late-concat	0.7590	0.6782	0.8616	0.5383	0.5138	0.5652	0.6487
SwinV2-w16	early-concat	0.7188	0.6133	0.8679	0.6514	0.5024	0.9261	0.6851
SwinV2-w16	diff	0.7016	0.7329	0.6730	0.6237	0.5976	0.6522	0.6627
SwinV2-w16	abs-diff	0.6228	0.5943	0.6541	0.2801	0.3220	0.2478	0.4515
SwinV2-w16	late-concat	0.7556	0.7628	0.7484	0.3949	0.7381	0.2696	0.5753

Table 6.4: Comparison of fusion strategies. Scores are reported as building-level F1, precision, and recall. The model was trained using pixel-wise loss and the mean-probability rule for building-level prediction.

learn joint spatial patterns that generalize across domains, at the expense of finer temporal discrimination. Differencing strategies (*difference* and *absolute difference*) perform less effectively overall, especially in cross-domain settings. While they simplify the learning problem by emphasizing explicit change information, they also discard useful contextual and radiometric cues, leading to reduced precision and unstable transfer performance. Overall, the results indicate that late fusion provides the strongest within-domain representations, whereas early fusion offers more balanced performance under distribution shifts.

6.2.4 Prediction strategies

To assess the impact of building-level aggregation on final performance, we compared the three prediction rules introduced in Section 5.3.4: *mean probability*, *minimum-area rule*, and *any-pixel rule*. These post-processing strategies differ in how they summarize pixel-wise probabilities within each building polygon—ranging from smooth averaging to more localized or binary criteria. The goal of this comparison is to quantify the trade-off between sensitivity to small damaged regions and robustness against noisy pixel predictions (see Table 6.5).

The minimum-area rule achieves comparable performance to the mean approach on the ID test set and, in some cases, slightly outperforms it for the DINOv3 model. Its higher recall values indicate stronger sensitivity to small but spatially consistent damaged regions. However, this comes at the expense of slightly lower precision, as small prediction clusters may trigger false alarms in more heterogeneous or noisy contexts. The any-pixel rule yields the highest recall values (up to 0.87) but at

Encoder ¹	Prediction	ID Test			OOD Test			AVG F1
		F1	p	r	F1	p	r	
CNXv2	mean	0.7758	0.7485	0.8050	0.6287	0.5735	0.6957	0.7023
CNXv2	min-area	0.7464	0.6957	0.8050	0.5775	0.4852	0.7130	0.6620
CNXv2	any-pixel	0.7124	0.6136	0.8491	0.5802	0.4364	0.8652	0.6463
DINOv3	mean	0.7590	0.6782	0.8616	0.5383	0.5138	0.5652	0.6487
DINOv3	min-area	0.7675	0.6919	0.8616	0.5389	0.4199	0.7522	0.6532
DINOv3	any-pixel	0.7289	0.6994	0.7610	0.5219	0.4497	0.6217	0.6254

Table 6.5: Comparison of prediction strategies. Scores are reported as building-level F1, precision, and recall. The model was trained using late concatenation fusion technique and pixel-wise loss.

the cost of a substantial precision drop, reflecting its extreme sensitivity to even minimal predicted damage. While this may be desirable in scenarios prioritizing exhaustive detection (e.g., rapid response mapping), it tends to overestimate damage in cross-domain conditions, where model confidence is less stable. Overall, the mean probability rule provides the most balanced and generalizable performance, while the minimum-area rule offers a valuable alternative for high-recall use cases. The observed differences confirm that the aggregation stage is not merely a post-processing detail but a key determinant of operational performance, especially under variable data distributions.

Qualitative examples

Figure 6.2 illustrate representative examples of the model outputs and their corresponding prediction maps. Each set of panels shows the pre and post-event images, the predicted damage probability heatmap, and the final binary prediction after thresholding. These visualizations highlight the model’s ability to localize damaged areas with high spatial precision.

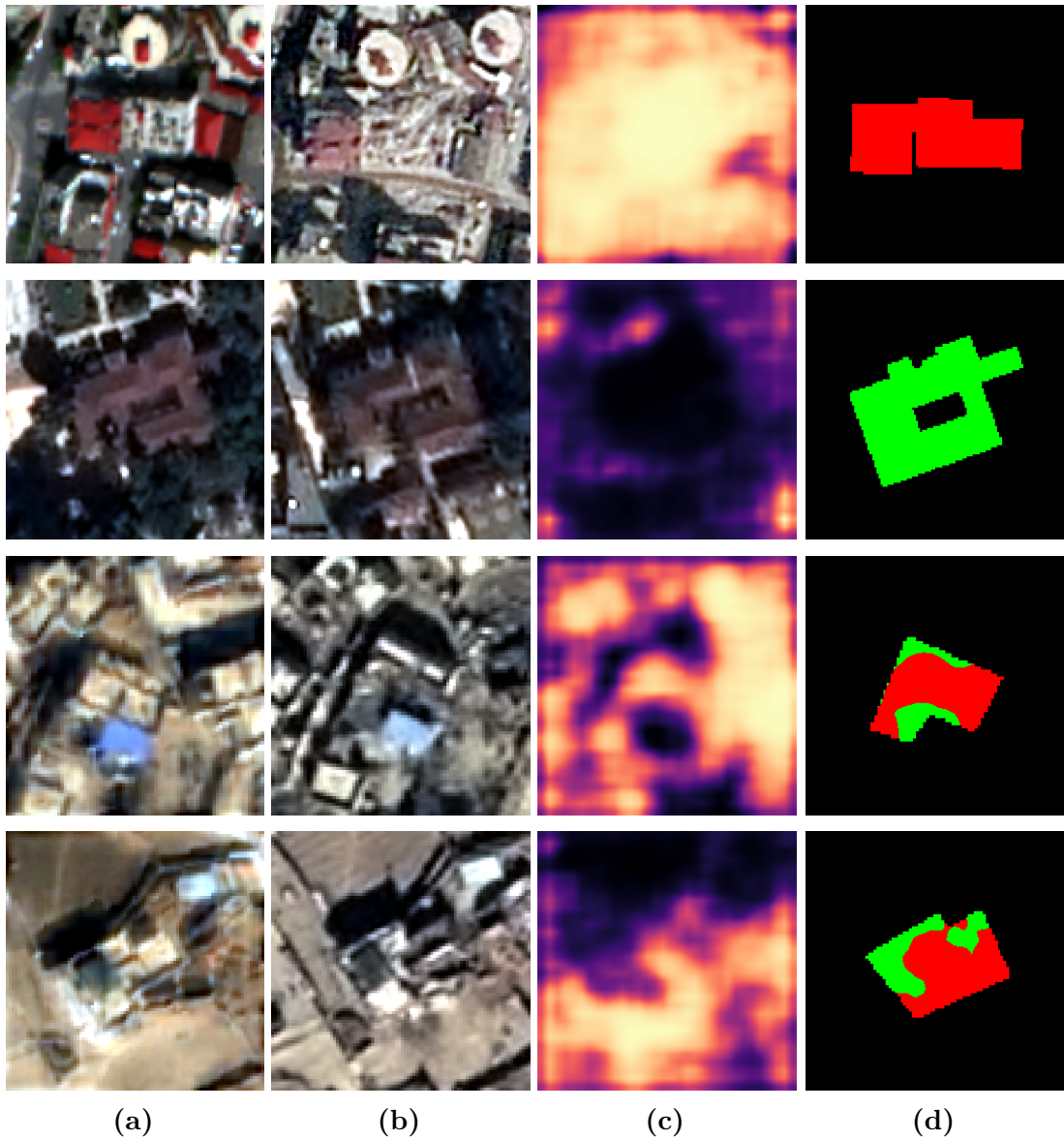


Figure 6.2: (a) pre-event; (b) post-event; (c) predicted damage activations heatmap (lighter = higher damaged probability); (d) thresholded prediction with $\tau = 0.5$ (red = damaged, green = undamaged); background outside building footprints is ignored.

6.2.5 Loss Functions

To examine how the choice of objective function influences training stability and generalization, we evaluated several loss formulations introduced in Chapter 5. Beyond the standard pixel-wise binary cross-entropy (BCE), we considered composite objectives combining pixel-level and region-level supervision. Specifically, we tested (i) *pixel with dice* losses, which jointly optimize class balance and overlap; (ii) *building-level* losses, directly aggregating predictions over building masks during training; and (iii) *hybrid* variants that combine pixel and building-level terms, optionally with an additional dice regularizer. These experiments were conducted using the late-concatenation fusion strategy and compared across both ID and OOD test sets (see Tables 6.6 and 6.7).

Encoder ¹	Loss	ID Test			OOD Test			AVG F1
		F1	p	r	F1	p	r	
CNXv2	pixel	0.7758	0.7485	0.8050	0.6287	0.5735	0.6957	0.7758
CNXv2	pixel w/dice	0.7884	0.7312	0.8553	0.6117	0.5285	0.7261	0.7001
CNXv2	building	0.8487	0.8034	0.8994	0.6242	0.5914	0.6609	0.7365
CNXv2	hybrid	0.8000	0.7614	0.8428	0.6379	0.6055	0.6739	0.7190
CNXv2	hybrid w/dice	0.8059	0.7569	0.8616	0.5891	0.5182	0.6826	0.6975

Table 6.6: Comparison of loss functions for ConvNeXtV2. Scores are reported as building-level F1, precision, and recall. The model was trained using late concatenation fusion technique, mean-probability rule for building-level prediction.

Encoder ¹	Loss	Prediction	ID Test			OOD Test			AVG F1
			F1	p	r	F1	p	r	
DINOv3	pixel	mean	0.7590	0.6782	0.8616	0.5383	0.5138	0.5652	0.6487
DINOv3	pixel w/dice	mean	0.7611	0.6816	0.8616	0.5372	0.5587	0.5174	0.6492
DINOv3	building	mean	0.7906	0.7444	0.8428	0.5768	0.5592	0.5957	0.6837
DINOv3	hybrid	mean	0.7908	0.7263	0.8679	0.5150	0.4760	0.5609	0.6529
DINOv3	hybrid w/dice	mean	0.8071	0.7640	0.8553	0.5212	0.5083	0.5348	0.6642
DINOv3	building	min-area	0.7192	0.6171	0.8616	0.5730	0.4175	0.9130	0.6461
DINOv3	hybrid	min-area	0.7466	0.6587	0.8616	0.5477	0.4613	0.6739	0.6472

Table 6.7: Comparison of loss functions for DINOv3 under different prediction strategies. Scores are reported as building-level F1, precision, and recall. The model was trained using late concatenation fusion technique.

The results reported in Tables 6.6 and 6.7 show that incorporating structural priors through building-level or hybrid supervision leads to substantial improvements over standard pixel-wise optimization. For ConvNeXtV2, the building-level loss achieves the highest in-distribution score (F1 = 0.85), with a marked gain in both precision and recall. This demonstrates that directly supervising the aggregated building response encourages more coherent and object-consistent predictions, reducing

fragmented detections and false negatives within partially damaged structures. The hybrid formulation also provides the strongest OOD performance ($F1 = 0.64$), suggesting that joint spatial and semantic regularization improves resilience to domain shifts. While the addition of a dice component marginally stabilizes training, it offers limited benefits in this setup and occasionally reduces precision due to overemphasis on overlap regions. For DINOv3, a similar trend is observed: hybrid and building-level losses outperform pure pixel-based objectives, with the hybrid with dice configuration achieving the highest overall ID F1 (0.81). When evaluated under the minimum-area prediction rule, the building loss yields particularly strong recall (0.91), indicating that this formulation better captures subtle or spatially sparse damage patterns. However, its precision remains lower in cross-domain tests, implying some sensitivity to false alarms in heterogeneous scenes. The training dynamics in Figure 6.3 further illustrate this behavior, showing how the logarithmic uncertainty parameters ($\log \sigma_{pixel}$ and $\log \sigma_{building}$) evolve to modulate the contribution of each loss component throughout training. This analysis underscores the value of aligning the training objective with the target evaluation granularity (in this case the building level) which ultimately improves both interpretability and generalization in post-disaster damage mapping.

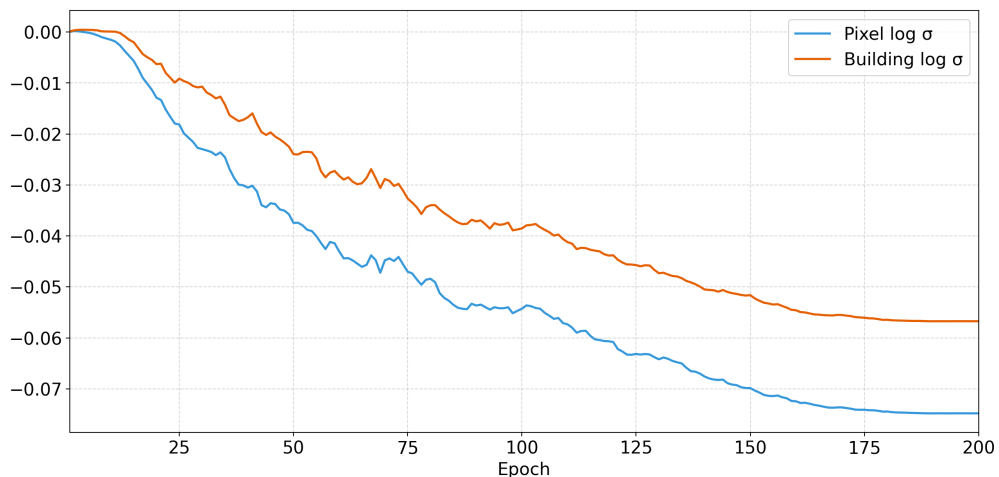


Figure 6.3: Evolution of the two uncertainty parameters ($\log \sigma_{pixel}$ and $\log \sigma_{building}$) during training, illustrating how the model dynamically balances the pixel- and building-level loss contributions.

6.2.6 Base vs. Large

To investigate the effect of model scale on performance, we compared the *base* and *large* variants of the ConvNeXtV2 and DINOv3 encoders. Larger models

typically provide higher representational capacity and improved feature abstraction but may also exhibit overfitting or instability when training data are limited or heterogeneous. This experiment therefore aimed to quantify the trade-off between model complexity, in-distribution accuracy, and cross-domain robustness. All models were evaluated under consistent late-concatenation and loss configurations, as summarized in Tables 6.8 and 6.9.

Encoder ²	Loss	ID Test			OOD Test			AVG F1
		F1	p	r	F1	p	r	
CNXv2-B	building	0.8487	0.8034	0.8994	0.6242	0.5914	0.6609	0.7365
CNXv2-L	building	0.7964	0.7600	0.8365	0.5022	0.5136	0.4913	0.6493
CNXv2-B	hybrid	0.8000	0.7614	0.8428	0.6379	0.6055	0.6739	0.7190
CNXv2-L	hybrid	0.7824	0.7348	0.8365	0.5195	0.5512	0.4913	0.6510
CNXv2-B	hybrid w/dice	0.8059	0.7569	0.8616	0.5891	0.5182	0.6826	0.6975
CNXv2-L	hybrid w/dice	0.7588	0.7763	0.7421	0.5163	0.6095	0.4478	0.6376

Table 6.8: Comparison between base and large ConvNeXtV2 models. Scores are reported as building-level F1, precision, and recall. The model was trained using late concatenation fusion technique, mean-probability rule for building-level prediction.

Encoder ²	Loss	Prediction	ID Test			OOD Test			AVG F1
			F1	p	r	F1	p	r	
DINOv3-B	building	mean	0.7906	0.7444	0.8428	0.5768	0.5592	0.5957	0.6837
DINOv3-L	building	mean	0.8025	0.7879	0.8176	0.5802	0.5508	0.6130	0.6914
DINOv3-B	building	min-area	0.7192	0.6171	0.8616	0.5730	0.4175	0.9130	0.6461
DINOv3-L	building	min-area	0.7114	0.5885	0.8994	0.5920	0.4341	0.9304	0.6517
DINOv3-B	hybrid	mean	0.7908	0.7263	0.8679	0.5150	0.4760	0.5609	0.6529
DINOv3-L	hybrid	mean	0.8034	0.7344	0.8868	0.6032	0.5644	0.6478	0.7033
DINOv3-B	hybrid	min-area	0.7466	0.6587	0.8616	0.5477	0.4613	0.6739	0.6472
DINOv3-L	hybrid	min-area	0.7538	0.6364	0.9245	0.6468	0.5228	0.8478	0.7003
DINOv3-B	hybrid w/dice	mean	0.8071	0.7640	0.8553	0.5212	0.5083	0.5348	0.6642
DINOv3-L	hybrid w/dice	mean	0.8000	0.7330	0.8805	0.5272	0.5284	0.5261	0.6636

Table 6.9: Comparison between base and large DINOv3 models. Scores are reported as building-level F1, precision, and recall. The model was trained using late concatenation fusion technique.

The comparison between base and large model variants reveals that larger architectures do not consistently outperform their smaller counterparts. For ConvNeXtV2, the base version achieves superior results across all tested loss configurations, reaching an F1 of 0.85 on the ID test and 0.64 on the OOD set with the hybrid loss. In

²B: Base version, L: Large version

contrast, the large variant shows a noticeable degradation in both precision and recall, suggesting overfitting to the training distribution and less stable generalization under cross-city shifts. This performance gap indicates that the available dataset may not provide sufficient diversity to fully exploit the additional capacity of the large encoder. For DINOv3, the large model achieves slightly higher or comparable ID scores in several settings (e.g., hybrid and building losses with mean probability prediction) and shows consistent gains in recall under the minimum-area rule for the OOD test (up to 0.93). However, these improvements are often marginal and accompanied by higher variability, implying that while larger self-supervised models can leverage richer representations, their benefits depend strongly on the loss design and prediction strategy. Overall, these findings confirm that “larger” is not inherently “better.” While large encoders can offer marginal advantages in specific regimes (particularly for recall-oriented objectives) they tend to require more regularization, data, and tuning to achieve stable performance. In contrast, base models provide a more reliable balance between accuracy, efficiency, and generalization, making them a more practical choice for scalable disaster response systems.

6.2.7 Fine-Tuning of Encoders

Finally, to evaluate the contribution of pretrained representations and the optimal degree of adaptation to the target domain, we investigated different fine-tuning regimes for the encoder backbones. As detailed in Chapter 5, two configurations were tested: (i) *Fully Frozen (FF)*, where pretrained weights remain fixed and only the decoder is updated; (ii) *Partially Frozen (PF)*, where the last n encoder stages are unfrozen ($PF(1-3)$) These experiments aim to clarify whether gradual unfreezing enhances task adaptation or instead leads to overfitting, particularly under cross-city domain shifts. The results are summarized in Tables 6.10 and 6.11.

The results in Tables 6.10 and 6.11 reveal that full fine-tuning of pretrained encoders is not necessarily advantageous (particularly in data-limited or distribution-shifted settings). For ConvNeXtV2, the fully frozen (FF) models consistently achieve the best or near-best performance across both losses and evaluation regimes, reaching an F1 of 0.85 on the ID test and 0.64 on the OOD test. Partial unfreezing of the last one to three stages yields minor gains in precision but generally leads to reduced stability and poorer cross-domain generalization, as reflected by the sharp decline in OOD F1 beyond $PF(1)$. This suggests that the pretrained features of ConvNeXtV2 already capture semantically transferable structures, and aggressive

³FF: Full Frozen, PF: Partially Frozen (number of last unfrozen layers.)

Encoder ¹	FL ³	Loss	ID Test			OOD Test			AVG F1
			F1	p	r	F1	p	r	
CNXv2	FF	building	0.8487	0.8034	0.8994	0.6242	0.5914	0.6609	0.7365
CNXv2	PF (1)	building	0.8094	0.7582	0.8679	0.5972	0.5539	0.6478	0.7033
CNXv2	PF (2)	building	0.8257	0.8036	0.8491	0.3333	0.6023	0.2304	0.5795
CNXv2	PF (3)	building	0.8129	0.7596	0.8742	0.4232	0.6348	0.3174	0.6181
CNXv2	FF	hybrid	0.8000	0.7614	0.8428	0.6379	0.6055	0.6739	0.7190
CNXv2	PF (1)	hybrid	0.7977	0.7473	0.8553	0.5789	0.5417	0.6217	0.6883
CNXv2	PF (2)	hybrid	0.8155	0.7740	0.8616	0.3562	0.4294	0.3043	0.5859
CNXv2	PF (3)	hybrid	0.8379	0.8155	0.8616	0.3832	0.4834	0.3174	0.6106

Table 6.10: Performance with partial fine-tuning of ConvNeXtV2 encoder backbones. Scores are reported as F1, precision and recall. The model was trained using late concatenation fusion technique and the mean-probability rule for building-level prediction.

Encoder ¹	FL ³	Loss	ID Test			OOD Test			AVG F1
			F1	p	r	F1	p	r	
DINOv3	FF	building	0.7906	0.7444	0.8428	0.5768	0.5592	0.5957	0.6837
DINOv3	PF (1)	building	0.8232	0.7988	0.8491	0.4365	0.4866	0.3957	0.6299
DINOv3	PF (2)	building	0.8276	0.8250	0.8302	0.2222	0.3077	0.1739	0.5249
DINOv3	PF (3)	building	0.8427	0.7978	0.8931	0.2261	0.3391	0.1696	0.5344
DINOv3	FF	hybrid	0.7908	0.7263	0.8679	0.5150	0.4760	0.5609	0.6529
DINOv3	PF (1)	hybrid	0.8261	0.8160	0.8365	0.4525	0.4717	0.4348	0.6393
DINOv3	PF (2)	hybrid	0.8051	0.8182	0.7925	0.2464	0.2772	0.2217	0.5258
DINOv3	PF (3)	hybrid	0.8369	0.8193	0.8553	0.2553	0.3288	0.2087	0.5461

Table 6.11: Performance with partial fine-tuning of DINOv3 encoder backbones. Scores are reported as F1, precision and recall. The model was trained using late concatenation fusion technique and the mean-probability rule for building-level prediction.

fine-tuning may overfit to the training distribution’s specific texture and lighting characteristics. In the case of DINOv3, partial fine-tuning improves in-distribution scores modestly, especially for deeper unfreezing levels ($PF(3)$), which achieve F1 scores above 0.84. However, these improvements are accompanied by a drastic drop in OOD performance, indicating a loss of generalization once the pretrained representation is excessively altered. Fully frozen configurations, while slightly less accurate on the ID test, remain significantly more robust under domain shifts, underscoring the strength and stability of self-supervised visual features when transferred directly. Overall, these findings confirm a consistent pattern across architectures:

- Frozen encoders yield the most reliable and generalizable performance.
- Partial fine-tuning can provide limited in-domain gains but increases the risk

of overfitting.

The optimal strategy depends on the available data diversity and the target deployment scenario, with full freezing emerging as a safer and more practical choice for real-world disaster response applications.

6.3 Challenge Results

This section presents the results obtained during participation in the official damage assessment challenge, following the training and evaluation procedures described in Chapter 5. Unlike the benchmark experiments discussed in Section 6.2, which allowed for extensive testing and tuning, the models submitted to the challenge were constrained by practical limitations inherent to the competition framework. In particular, participants were allowed only one submission per day during Phase 2 and two submissions per day during Phase 1, which substantially limited the number of experiments that could be validated on the official leaderboard. Moreover, due to the tight schedule and the need for rapid iteration, the architectures selected for submission did not correspond to the best-performing configurations later identified in the benchmark analysis. Several improvements—such as enhanced loss formulations, refined fusion strategies, and partial fine-tuning schemes—were developed only after the challenge deadline. It is also important to note that the official test labels used by the organizers are not publicly available, preventing retrospective evaluation of these improved models on the same hidden dataset. As a result, the results reported in this section reflect the performance of the models actually submitted during the competition period, providing a fair representation of their effectiveness under realistic time and resource constraints.

Test set	Rank	F1-score
Phase 1 (ID)	4 th	0.775

Table 6.12: Official challenge results for Phase 1 (ID) test sets. Scores refer to the global building-level F1 metric used for leaderboard ranking. **Model:** CNXv2–UPerNet architecture, late concatenation fusion technique, pixel-wise loss, mean-probability rule for building-level prediction and the encoder partially frozen (last 3 layers).

The results highlight a consistent gap between in-distribution and out-of-distribution performance, reflecting the inherent difficulty of generalizing across unseen geographic areas. In Phase 1, the ConvNeXtV2–UPerNet configuration achieved a strong 4th-place ranking with an F1 of 0.775, confirming the model’s solid within-domain generalization. In contrast, Phase 2 proved considerably more challenging:

Test set	Rank	F1-score
Phase 2 (OOD)	6 th	0.342

Table 6.13: Official challenge results for Phase 2 (OOD) test sets. Scores refer to the global building-level F1 metric used for leaderboard ranking. **Model:** DINOv3-UPerNet architecture, late concatenation fusion technique, building-level loss, mean-probability rule for building-level prediction and the encoder partially frozen (last 3 layers).

the DINOv3-UPerNet submission reached 6th place with an F1 of 0.342, consistent with the drop in cross-city transferability observed in the benchmark experiments. Overall, these outcomes validate the experimental findings of Section 6.2: while modern pretrained encoders can achieve high accuracy under known conditions, robust adaptation to unseen domains remains a major challenge and a central focus for future work.

6.4 Summary of Results

Across all experiments, the proposed framework demonstrated consistent and interpretable behavior under varying architectural, training, and evaluation conditions. The benchmark analysis showed that performance depends strongly on the interplay between encoder design, fusion strategy, and loss formulation. Architectures such as ConvNeXtV2-UPerNet and DINOv3-UPerNet achieved the best overall balance between accuracy and robustness, while hybrid or building-level losses further improved object consistency and domain transferability. At the same time, experiments on fine-tuning revealed that extensive adaptation of pretrained backbones often leads to overfitting, confirming that frozen or lightly tuned encoders offer the most stable and generalizable representations for post-disaster damage assessment. The challenge results provided a realistic validation of these findings under operational constraints. Despite limited training time and submission frequency, the submitted models achieved competitive leaderboard rankings in both phases, confirming the soundness of the overall methodological design. Nevertheless, the performance gap observed between in-distribution and cross-city test sets emphasizes that domain generalization remains the principal open problem in large-scale disaster mapping and motivates future efforts toward adaptive and uncertainty-aware learning strategies.

Chapter 7

Conclusions

Natural disasters such as earthquakes cause large-scale destruction, posing immediate challenges for emergency response and recovery planning. Rapid and reliable damage assessment is therefore essential to identify the most affected areas, allocate resources efficiently, and support decision-making in the critical hours following an event. Traditional field surveys, while accurate, are time-consuming, costly, and often infeasible in the aftermath of major earthquakes due to accessibility and safety constraints. In this context, remote sensing and machine learning methods provide a powerful alternative, enabling scalable and timely extraction of damage information directly from satellite or aerial imagery. Nevertheless, despite remarkable progress in deep learning, building-level damage classification after earthquakes remains challenging due to the heterogeneity of urban environments, the variability of sensor resolutions and acquisition conditions, and the limited availability of reliable annotated datasets. This thesis has addressed these challenges by developing, evaluating, and comparing deep learning models and methodological strategies for automated building damage assessment from pre and post-event imagery in earthquake scenarios.

Our Contributions. The work presented in this thesis has made several contributions to the field of remote sensing-based damage assessment. First, a refined and consistent dataset was created from heterogeneous sources, integrating geospatial data, multispectral imagery, and building footprints through a robust alignment and patch generation pipeline. This preprocessing stage addressed critical inconsistencies that typically hinder model generalization and reproducibility. Second, multiple deep learning architectures were implemented and analyzed, emphasizing the role of encoder design, fusion strategies, and loss formulations. Third, a comprehensive evaluation protocol was designed, encompassing both pixel-level and building-level metrics, along with qualitative visual analyses. This allowed a balanced assessment of performance not only in terms of quantitative indicators

but also in terms of interpretability and real-world reliability.

Future Works. Building upon the findings of this thesis, several directions can be pursued to further advance earthquake damage assessment through remote sensing and deep learning. A primary objective should be the expansion and diversification of annotated datasets. Increasing the quantity and geographic variety of labeled examples would significantly enhance model generalization and reduce biases arising from class imbalance and uneven spatial distributions. Another promising direction involves the integration of multimodal and multitemporal data. The combination of optical and synthetic aperture radar (SAR) imagery could improve robustness in adverse conditions such as cloud coverage or variable illumination. Likewise, incorporating temporal sequences from satellite constellations would enable monitoring of damage evolution, supporting both rapid response and long-term reconstruction analysis. From a modeling perspective, further investigation into uncertainty estimation and domain adaptation techniques could strengthen the interpretability and transferability of predictions across different disaster events. Incorporating geospatial priors, such as building typology or local seismic vulnerability, may also improve contextual understanding and reduce false detections in heterogeneous urban settings.

Bibliography

- [1] European Space Agency (ESA) Φ-Lab and International Charter "Space and Major Disasters". *AI for Earthquake Response Challenge*. <https://platform.ai4eo.eu/ai-for-earthquake-response>. 2025 (cit. on p. 5).
- [2] Ritwik Gupta, Richard Hosfelt, Sandra Sajeew, and et al. «Creating xBD: A Dataset for Assessing Building Damage from Satellite Imagery». In: *CVPR Workshops*. 2019. URL: https://openaccess.thecvf.com/content_CVPRW_2019/papers/cv4gc/Gupta_Creating_xBD_A_Dataset_for_Assessing_Building_Damage_from_Satellite_CVPRW_2019_paper.pdf (cit. on p. 7).
- [3] European Space Agency. *AI4EO Platform – Artificial Intelligence for Earth Observation*. <https://platform.ai4eo.eu/>. Accessed: 2025-09-18. 2024 (cit. on p. 7).
- [4] Karl Amolins, Yong Zhang, and Paul Dare. «Wavelet Based Image Fusion Techniques—An Introduction, Review and Comparison». In: *ISPRS Journal of Photogrammetry and Remote Sensing* 62.4 (2007), pp. 249–263 (cit. on p. 8).
- [5] Gemine Vivone, Luciano Alparone, Jocelyn Chanussot, Mauro Dalla Mura, Andrea Garzelli, Giorgio A. Licciardi, Rocco Restaino, and Lucien Wald. «A Critical Comparison Among Pansharpening Algorithms». In: *IEEE Transactions on Geoscience and Remote Sensing* 53.5 (2015), pp. 2565–2586. DOI: 10.1109/TGRS.2014.2361734 (cit. on pp. 8, 9).
- [6] Guy P. Nason and Bernard W. Silverman. «The Stationary Wavelet Transform and Some Statistical Applications». In: *Wavelets and Statistics*. Vol. 103. Lecture Notes in Statistics. Springer, 1995, pp. 281–300 (cit. on p. 8).
- [7] J. Nunez, X. Otazu, O. Fors, A. Prades, V. Pala, and R. Arbiol. «Multiresolution[†] based image fusion with additive wavelet decomposition». In: *IEEE Transactions on Geoscience and Remote Sensing* 37.3 (1999), pp. 1204–1211. DOI: 10.1109/36.763274 (cit. on pp. 8, 9).

- [8] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. 2015. arXiv: 1505.04597 [cs.CV]. URL: <https://arxiv.org/abs/1505.04597> (cit. on p. 9).
- [9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. «Deep Residual Learning for Image Recognition». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2016 (cit. on pp. 9, 11).
- [10] Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. *ConvNeXt V2: Co-designing and Scaling ConvNets with Masked Autoencoders*. 2023. arXiv: 2301.00808 [cs.CV]. URL: <https://arxiv.org/abs/2301.00808> (cit. on pp. 9, 12).
- [11] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. *Unified Perceptual Parsing for Scene Understanding*. 2018. arXiv: 1807.10221 [cs.CV]. URL: <https://arxiv.org/abs/1807.10221> (cit. on p. 9).
- [12] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. *Feature Pyramid Networks for Object Detection*. 2017. arXiv: 1612.03144 [cs.CV]. URL: <https://arxiv.org/abs/1612.03144> (cit. on p. 9).
- [13] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. *Pyramid Scene Parsing Network*. 2017. arXiv: 1612.01105 [cs.CV]. URL: <https://arxiv.org/abs/1612.01105> (cit. on p. 9).
- [14] Alexey Dosovitskiy et al. «An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale». In: *International Conference on Learning Representations (ICLR)*. 2021 (cit. on pp. 9, 12).
- [15] Zhuo Zheng, Yanfei Zhong, Junjue Wang, Ailong Ma, and Liangpei Zhang. «Building damage assessment for rapid disaster response with a deep object-based semantic change detection framework: From natural disasters to man-made disasters». In: *Remote Sensing of Environment* 265 (2021), p. 112636. ISSN: 0034-4257. DOI: <https://doi.org/10.1016/j.rse.2021.112636>. URL: <https://www.sciencedirect.com/science/article/pii/S0034425721003564> (cit. on p. 10).
- [16] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. «A ConvNet for the 2020s». In: *CVPR*. 2022 (cit. on p. 11).
- [17] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. «Scene Parsing through ADE20K Dataset». In: *CVPR*. 2017 (cit. on p. 12).

- [18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. «Attention Is All You Need». In: *NeurIPS*. 2017 (cit. on p. 12).
- [19] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. «Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows». In: *ICCV*. 2021 (cit. on p. 13).
- [20] Ze Liu et al. «Swin Transformer V2: Scaling Up Capacity and Resolution». In: *CVPR*. 2022 (cit. on p. 13).
- [21] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. *Emerging Properties in Self-Supervised Vision Transformers*. 2021. arXiv: 2104.14294 [cs.CV]. URL: <https://arxiv.org/abs/2104.14294> (cit. on p. 13).
- [22] Oriane Siméoni et al. *DINOv3*. 2025. arXiv: 2508.10104 [cs.CV]. URL: <https://arxiv.org/abs/2508.10104> (cit. on p. 13).
- [23] Oriane Siméoni et al. *DINOv3 ConvNeXt models and training notes*. Hugging-Face model cards: convnext_base.dinov3_lvd1689m. 2025 (cit. on p. 14).
- [24] Alex Kendall, Yarin Gal, and Roberto Cipolla. *Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics*. 2018. arXiv: 1705.07115 [cs.CV]. URL: <https://arxiv.org/abs/1705.07115> (cit. on p. 14).
- [25] Airbus / Geospatial Intelligence. *Pléiades-1A/1B Technical Specifications: 0.5 m Panchromatic, 2 m Multispectral at Nadir*. <https://www.geoint.global/products/pleiades-0-5-m/> (cit. on p. 17).
- [26] Maxar Technologies. *WorldView-2/3 Satellite Imagery Specifications*. <https://earth.esa.int/eogateway/missions/worldview-3#instruments-section> (cit. on p. 17).
- [27] KARI / SI Imaging Services / Geopera. *KOMPSAT-3: 70-cm Panchromatic, 2.8-m Multispectral at Nadir*. <https://geopera.com/sensors/kompsat-3/> (cit. on p. 17).
- [28] CNSA / SatimagingCorp / Apollo Mapping. *Gaofen-2 Satellite Imagery: 80-cm Panchromatic and 3.2-m Multispectral (Nadir)*. <https://apollomapping.com/gaofen-2-satellite-imagery> (cit. on p. 17).