

Identification and Classification of War-related Damages of Buildings Using Ground Level Images and State-of-the-art Models for Object Detection and Segmentation

Identifikation und Klassifizierung kriegsbedingter Gebäudeschäden unter Verwendung von terrestrischen Bildern und etablierten Modellen zur Objektdetektion und Segmentierung

Anand Gajaria, Basil Roth, Jonathan Banz, Andreas Wieser, Gesa Ziemer

Beyond the immediate human suffering, Russia's full-scale invasion of Ukraine has already inflicted massive damage on civilian infrastructure. The scale and nature of this destruction create a pressing need for rapid damage assessment to support recovery planning. Current assessment workflows depend heavily on manual terrestrial inspection, making them slow, resource-intensive, and difficult for local authorities to scale across multiple settlements in a consistent way. Satellite imagery is widely perceived as a useful basis for monitoring war-related damage, but satellite-based approaches, while offering synoptic coverage and scaling well to large areas, lack the spatial granularity and contextual detail needed to capture on-the-ground damage conditions at the level of individual buildings. Herein, we present a deep-learning-based damage assessment pipeline that structures and automates the extraction of damage information from terrestrial imagery. It is intended to be interoperable with other sensor data, such as satellite imagery, by later extension into a more comprehensive solution for automated derivation of robust spatio-temporal evidence of destruction. At its core, the pipeline consists of a YOLOv8-based object detector and a Segment Anything Model (SAM) for automatic extraction of impacted structures from terrestrial imagery, and categorisation as damaged walls, damaged windows, and debris. We used a patch-based data processing strategy to mitigate background dominance. We fine-tuned the models on a manually annotated dataset derived from 380 images. The two-stage architecture first detects and classifies damaged regions, then performs pixel-level segmentation, providing accurate localisation of structural damage. Within our study, we achieved a mean average precision (mAP) of 0.65 for the object detection and a Dice coefficient of 0.91 for the segmentation. This demonstrates that the proposed pipeline is a useful first step towards supporting spatial damage assessment and recovery planning with automated provision of quantitative and qualitative information. Despite the promising results, the current implementation is limited by the relatively small dataset. More research is needed to develop the pipeline into a solution that generalises well to unseen environments and extracts the relevant information in formats and with semantics

compatible with the requirements and legal constraints of public administration in Ukraine, or of other specific stakeholders.

Keywords: Damage assessment, buildings, damage classification, terrestrial imagery, semantic segmentation, Segment Anything Model (SAM), object detection, YOLOv8

Über das unmittelbare menschliche Leid hinaus hat Russlands Großinvasion in die Ukraine massive Schäden an ziviler Infrastruktur verursacht. Das Ausmaß und die Art dieser Zerstörung machen eine rasche Schadensbewertung dringend erforderlich, um den Wiederaufbau zu planen. Gängige Praktiken zur Bewertung beruhen stark auf manueller Inspektion vor Ort, wodurch sie langsam und ressourcenintensiv sind und es für lokale Behörden schwierig ist, sie einheitlich auf mehrere Siedlungen auszuweiten. Satellitenbilder gelten weithin als hilfreiche Grundlage zur Beobachtung kriegsbedingter Schäden. Satellitenbasierte Ansätze liefern zwar eine großräumige Übersicht und lassen sich gut auf große Gebiete skalieren, erreichen jedoch nicht die räumliche Granularität und kontextbezogenen Details, die erforderlich sind, um Schäden vor Ort auf der Ebene einzelner Gebäude zuverlässig abzubilden. In diesem Beitrag stellen wir eine auf Deep Learning basierende Pipeline zur Schadensbewertung vor, welche die Extraktion von Schadensinformationen aus terrestrischen Bildern strukturiert und automatisiert. Sie soll mit anderen Sensordaten, wie beispielsweise Satellitenbildern, kompatibel sein, sodass später durch die Erweiterung auf multimodale Eingaben eine umfassendere Lösung für die automatisierte Ableitung robuster räumlich-zeitlicher Beweise für Zerstörungen entwickelt werden kann. Im Kern kombiniert die Pipeline einen YOLOv8-basierten Objektdetektor mit dem Segment Anything Model (SAM) zur automatisierten Extraktion von betroffenen Strukturen aus terrestrischen Bildern und deren Kategorisierung als beschädigte Wände, beschädigte Fenster und Trümmer. Zur Reduktion der Hintergrunddominanz setzen wir eine patchbasierte Datenverarbeitungsstrategie ein. Wir haben die vortrainierten Modelle mit einem manuell annotierten Datensatz aus 380 Bildern fein abgestimmt. Die zweistufige Architektur erkennt und klassifiziert zunächst beschädigte Bereiche und führt anschließend eine Segmentierung auf Pixelebene durch, wodurch eine genaue Lokalisierung struktureller Schäden ermöglicht wird. In unserer Studie erreichten wir eine mittlere durchschnittliche Präzision (mAP) von 0,65 für die Objekterkennung und einen Dice-Koeffizienten von 0,91 für die Segmentierung. Dies zeigt, dass die vorgeschlagene Pipeline ein nützlicher erster Schritt zur Unterstützung der räumlichen Schadensbewertung und Wiederherstellungsplanung durch die automatisierte Bereitstellung quantitativer und qualitativer Informationen ist. Trotz der vielversprechenden Ergebnisse ist die aktuelle Implementierung durch die vergleichsweise geringe Bildanzahl eingeschränkt. Weitere Forschung ist erforderlich, um die Pipeline zu einer Lösung mit hoher Generalisierungsfähigkeit weiterzuentwickeln, die sich gut auf unbekannte Umgebungen übertragen lässt und die relevanten Informationen in Formaten und mit einer Semantik extrahiert, die mit den Anforderungen und rechtlichen Auflagen der öffentlichen Verwaltung in der Ukraine oder anderer spezifischer Interessengruppen kompatibel sind.

Schlüsselwörter: Gebäudeschäden, Schadensklassifikation, terrestrische Fotos, semantische Segmentierung, Segment Anything Model (SAM), Objektdetektion, YOLOv8

1 INTRODUCTION

Large-scale destruction of civilian infrastructure is among the most critical challenges emerging from modern conflicts /Robinson & Nohle 2016/. In Ukraine, Russia's full-scale invasion has destroyed or severely damaged thousands of homes, schools, hospitals, and other public facilities. This situation has created an urgent need for fast, objective, and accurate analysis of damage in support of national recovery efforts. Timely and consistent evaluations are

crucial for effective planning of reconstruction, for resource allocation, and for transparent compensation to those affected. Reliable damage assessment further enables authorities and humanitarian organisations to identify the most severely impacted areas and prioritise interventions where they are needed most.

At present, damage assessment workflows under the eRecovery mechanism (<https://getitback.in.ua>) are carried out by local authorities,

who primarily rely on in-situ field surveys. To address accessibility and safety constraints, damage assessment based on manual inspection of photographs instead of on-site assessment is permissible near the frontline or in other high-risk areas (Cabinet of Ministers of Ukraine, Resolution No. 815, 7 July, 2025). These manual evaluations are time-consuming, resource-intensive, and therefore difficult to scale.

More generalised assessments of damages and needs, as presented in, e.g., /Aimaiti et al. 2022/, use top-view remote-sensing data, such as high-resolution satellite images and aerial images. These approaches provide rapid, consistent, and synoptic coverage for large-area screening and change detection, do not expose data collectors to safety risks, and are particularly effective for identifying roof-level damage and large-scale collapse patterns. However, top-view observations can typically not resolve facade-level or fine-grained structural indicators needed to reliably characterise many on-the-ground damage conditions, particularly in dense urban settings.

Ground-level images can complement top views by providing close-up detail and evidence that is difficult to infer from above. For this reason, an automated system to analyse terrestrial images can play a vital role in improving and scaling the assessment process. Such a system can both support a more consistent understanding of the actual building condition and enable accurate analysis in areas where structures are partially obscured or difficult to observe from above. We focus on adding terrestrial imagery as an additional evidence layer – leveraging photographs that are already routinely collected, but not yet systematically integrated into scalable assessment workflows. The work presented here is conducted in the context of the Mapping Ukraine project (ETH Zurich, and Swiss Network with Ukraine), and the development of UNITAC's URPS (Urban Recovery and Planning System) tool. The terrestrial imagery used in this study was collected on site in Ukraine by some of the authors and by project partners working in the affected settlements. If implemented for operational application, the model would allow using abundantly available data, such as cell phone images from citizens, images acquired through coordinated large-scale efforts, or images from other sources, for the purpose of damage assessment. This would turn the proposal into a scalable solution for detailed and reliable damage mapping.

The majority of research on automated damage detection has focused on satellite or aerial imagery. For instance, /Wu et al. 2020/ developed a U-Net model with attention mechanisms for building damage detection from satellite images trained on pre- and post-disaster images. While their model showed promising results in large-scale mapping accuracy, it showed limited capability for identifying damages on a small scale. /Nabiee et al. 2022/ proposed a hybrid U-Net architecture for semantic segmentation based on high-resolution satellite imagery in war-affected areas, yet its performance depended on the spatial resolution of the satellite imagery and the need for bitemporal data. More recently, /Alisjahbana et al. 2024/ introduced DeepDamageNet, a two-stage deep-learning model that combines segmentation and classification for multi-disaster damage analysis using satellite imagery, but the solution had difficulties distinguishing visually similar damage levels and generalising across geographic regions. A significant limitation when

using satellite imagery is the limited availability and high cost of high-resolution datasets, as stated in /Cong et al. 2022/.

To address this problem, /Dietrich et al. 2025/ introduced a tool for destruction mapping using Sentinel-1 SAR imagery. While such SAR data are freely available with relatively high temporal resolution, the ground resolution of 10 meters severely limits the level of detail the analysis can provide. Overall, these studies highlight limitations of remote-sensing approaches in capturing local structural details which would likely be represented better by ground-level data.

Several studies have also examined the recognition of building components using street-level imagery. /Dai et al. 2021/ applied deep learning for residential infrastructure segmentation in complex urban environments, allowing for the detection of architectural elements such as windows, doors, and walls. Similarly, /Wang et al. 2022/ proposed an R-CNN based model that combines region-based feature extraction with hierarchical parsing to better identify and classify building components in dense urban scenes, helping to distinguish overlapping and irregularly shaped facade elements. More recently, /Wang et al. 2023/ introduced a model that combines a Vision Transformer backbone with a line-based refinement module to improve segmentation accuracy by integrating structural boundary information, demonstrating good performance under challenging urban conditions such as occlusions or irregular facade geometry. While these advancements have significantly improved facade-level analysis, research specifically targeting on-the-ground damage assessment remains scarce, particularly for multi-class segmentation in post-conflict environments.

We aim to address this gap by proposing a deep-learning-based framework for automated assessment of damaged infrastructure from terrestrial imagery. It utilises a curated dataset of manually annotated photographs of damaged buildings in Ukraine. It aims to classify damages into three classes: damaged walls, damaged windows, and debris. Because damaged regions often make up only a small portion of each image while background areas dominate, we employ a patch-based preprocessing strategy to increase the relative representation of fine-scale and partially visible damage components. The model architecture combines two complementary computer-vision techniques. We fine-tuned the Segment Anything Model (SAM) /Kirillov et al. 2023/, a vision transformer model which is inherently class-agnostic, to generate precise segmentation masks of damaged regions. To enable automated and class-specific segmentation, we integrated the YOLOv8-based object detection model /Jocher et al. 2023/. We let it produce bounding box prompts and class labels as input to SAM. This two-stage workflow eliminates the need for manual prompting, enabling a fully automated, multi-class segmentation process for terrestrial damage assessment.

While the model was trained on three visually distinctive classes – damaged walls, damaged windows, and debris – this labelling scheme should not be mistaken for a fixed or universal taxonomy of architectural elements or damages. E.g., even when a facade region is unambiguously perceived as a “wall”, this recognition rarely rests on a single visual cue. Instead, it emerges from a constellation of geometry, material appearance, facade context, and prior knowledge about building construction. These criteria vary across building typologies and historical periods. Similarly, regions properly labelled

as “wall” and “window” can overlap. In our dataset, this becomes visible in the changing wall-to-window ratios across images. For example, facades with lots of windows contrast sharply with massive masonry or plastered walls. In this study, the annotation scheme is therefore explicitly task-oriented: „damaged wall” denotes damages to opaque envelope surfaces, “damaged window” denotes damages to openings and glass, and “debris” denotes displaced material that no longer forms part of the building envelope.

This abstraction allows us to group heterogeneous architectural categories under three operational classes that are directly relevant for damage assessment and recovery planning. The above three classes are sufficient to test the viability of our approach. A more fine-grained architectural classification and an adaptation to the needs of the actors involved in damage assessment and reconstruction lies beyond the scope of this publication. The same holds for a combination with other sensor data for in-depth assessment of the structural conditions of buildings or for the detection of indications of hidden damages for rapid assessment as in the examples given by /Yakovenko et al. 2026/ and /Danyliv et al. 2026/. Such developments will have to be pursued later, along with an adaptation to the relevant national and international administrative regulations.

Resolution	Images	Portrait	Landscape
4032 × 3024	208	77	131
4032 × 1816	94	94	0
4160 × 2080	49	49	0
4000 × 3000	15	15	0
1600 × 1200	11	8	3
3984 × 1840	3	2	1
Total	380	245	135

Tab. 1 | Summary of the dataset

We implemented the proposed pipeline in a software prototype that learns directly from human-labelled masks. The annotation protocol defines the operational classes but also, inevitably, encodes perceptual and cultural biases. Being as precise and consistent as possible is therefore crucial. At the same time, we suspect that there are sufficiently stable visual clues in the facade imagery for the model to reliably distinguish walls, windows, and debris, potentially based on patterns that remain only partially explicit to human annotators.

2 METHODOLOGY

2.1 Data and Annotation

In this study, we have used a dataset of 380 terrestrial images showing building damage in Ukraine caused by Russia’s full-scale invasion. The images capture a wide range of structural damage conditions of approximately 37 distinct buildings. The photographs were taken under diverse lighting conditions in the village of Kosarovychi and the suburbs of Kyiv in May 2022 and April 2024, using different cameras (Tab. 1) with different resolution and focal lengths, from different distances and viewing angles. The resulting visual diversity in the dataset is essential for robust model training.

All the images were manually annotated (see Fig. 1 for examples) using the LabelStudio annotation tool by /Tkachenko et al. 2025/. Pixel-wise annotations served as ground truth for the segmentation task and as the source for generating bounding boxes for the object-detection task. The bounding-box annotations were created automatically using a custom-developed script executed after the data preprocessing stage. Details of this step are described in Section 2.3.



Fig. 1 | Examples of original (left) and annotated (right) images. Damaged walls are shown in brown, damaged windows in red, and debris in blue. Background and areas that are not damaged are shown with the original colours.

2.2 Data Analysis and Preprocessing

2.2.1 Data Analysis

An exploratory data analysis was performed to understand the composition and class balance of the dataset prior to model training, with a focus on the distribution of foreground pixels, defined here as pixels belonging to annotated damage regions (damaged walls, damaged windows, and debris), coverage and the per-class distribution of damaged areas across all images.

Fig. 2 shows the distribution of the number of foreground pixels relative to the total number of pixels per image. The distribution is right-skewed, indicating that most images contain only a small or moderate amount of visible damage.

Fig. 3 provides insight into how the three damage types and background contribute to the total image area in the dataset. As

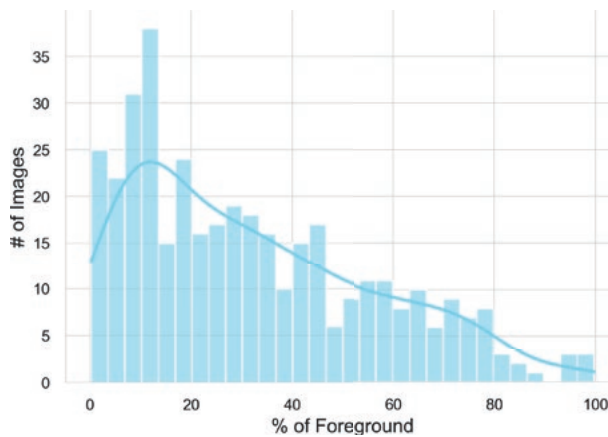


Fig. 2 | Distribution of foreground coverage per image

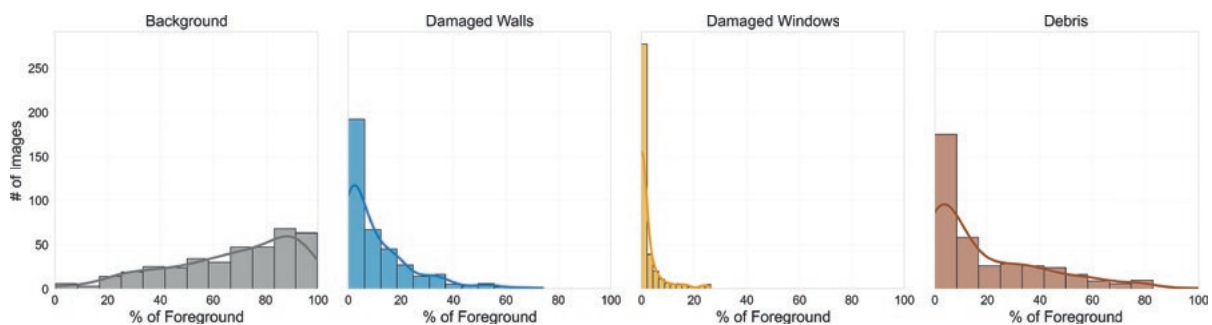


Fig. 3 | Distribution of coverage per class

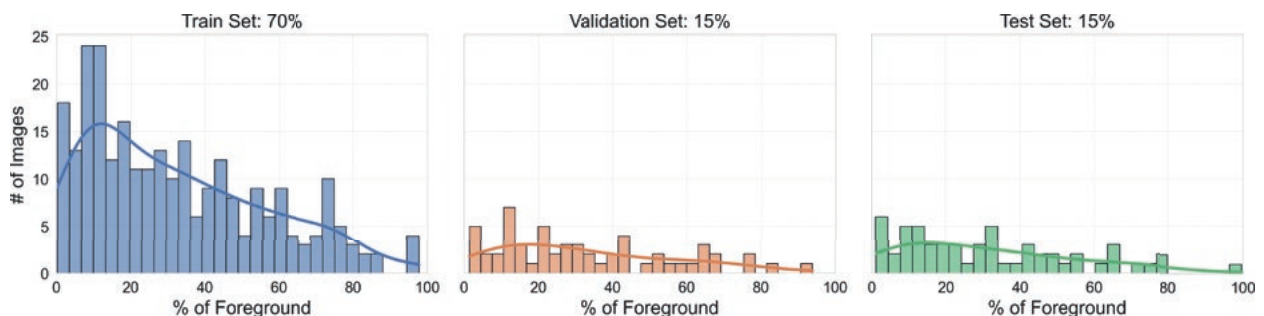


Fig. 4 | Distribution of foreground coverage per image across splits

expected, background pixels occupy the largest share of each image. In contrast, all three target classes exhibit right-skewed distributions, with strong peaks in the 0–10 % coverage bin. Among the damage classes, damaged walls and debris show relatively higher pixel coverage than damaged windows. This is consistent with basic facade composition: opaque wall surfaces usually occupy a larger share of the envelope than openings, so the relative area of windows and walls in the image directly reflects underlying architectural typologies represented in the dataset.

These observations regarding the data distribution highlight the importance of adopting appropriate sampling and preprocessing strategies before model training. The dominance of background pixels and the right-skewness in all three target classes indicate that random splitting could lead to unbalanced subsets in which certain damage levels or classes are underrepresented. To mitigate this risk, a stratified sampling strategy, as proposed by Neyman 1934, was utilised to create the training, validation, and test split (70 %, 15 %, and 15 %, i. e., 266, 57 and 57 images).

This approach ensured that each subset preserved a similar proportion of images with light, moderate, and heavy damage, as well as a similar proportion of foreground (Fig. 4), maintaining uniform class representation across all splits. Such partitioning is crucial to enable the model to generalise across diverse damage scenarios rather than overfitting to the most frequent patterns.

2.2.2 Data Preprocessing

A class-aware patching strategy was implemented to address the challenge of background dominance and class imbalance observed during data analysis. Each image and its corresponding mask were divided into overlapping patches of 512×512 pixels with a stride

of 256 pixels. This helped to maintain the spatial details and visibility of localized damage regions.

During preprocessing, all background-only patches were excluded from the dataset, using a custom-developed script. This decision was based on the observation that the foreground patches already contained sufficient background context for the model to learn scene-level variation. Retaining background patches would have unnecessarily reinforced the already dominant background representation and further skewed the training process. By filtering these out, the resulting dataset comprised 19 348 patches for training, 4 004 for validation, and 4 109 for testing. These remaining patches represented a healthier balance between damaged and non-damaged regions, enabling the model to focus more effectively on learning meaningful structural features rather than redundant background information.

2.3 Bounding Box Creation

After processing the data, bounding-box annotations were generated from the corresponding segmentation. These annotations were created to support both the object detection and the segmentation stages of the workflow. A custom script was developed to extract connected components from each segmentation mask and derive the smallest enclosing rectangle from it. The coordinates of these bounding box rectangles were fetched and stored in a JSON file. The entries of this file were later used as training labels for the YOLOv8 object detection model and as prompts for the SAM during fine-tuning.

2.4 Data Augmentation

To improve the generalisation capability of the segmentation model, a set of data augmentations was implemented on the fly, using the Albumentations library by /Buslaev et al. 2020/. The augmentation pipeline helped in adding more variations to the data in terms of illumination, colour, and texture. The photometric transformation included Colour Jitter adjustments for brightness ($\pm 20\%$), contrast ($\pm 20\%$), saturation ($\pm 20\%$), and hue ($\pm 10\%$) with an implementation probability of 0.6. Additionally, RandomGamma (probability 0.2), Gaussian noise (variance limit = 10–50, probability = 0.2), and MotionBlur (limit = 3, probability = 0.2) were applied. To simulate partial occlusions, CoarseDropout with up to five holes of 40×40 pixels each (probability = 0.3) was implemented.

Additionally, for the object detection model, the augmentation settings provided by the Ultralytics YOLOv8 library were enabled within the training configuration. The contextual transformation included hue shifts (1.5%), saturation variations (70%), and brightness adjustments (40%). The geometric transformation included random rotations ($\pm 10^\circ$), translations (10%), and scaling (up to 50%). Finally, the techniques Mosaic (probability = 0.8) and MixUp (probability = 0.2) were employed to combine multiple images into composite training samples, enriching contextual variety and regularising the learning process.

2.5 Environment Setup

The training, validation, and test datasets were stored on an AWS S3 bucket to facilitate seamless integration with cloud compute resources. Model training was carried out on an Amazon SageMaker Jupyter instance of type ml.g5.2xlarge. This GPU-enabled instance provides 8 vCPUs, 32 GB of system memory, and a single NVIDIA A10G GPU with 24 GB GPU memory. The storage volume was provisioned with 50 GB disk space to accommodate datasets, intermediate files, and model checkpoints. The software stack used for training the model included Python 3.11.13 and PyTorch 2.6.0, running within an Amazon-provided Deep Learning Container image, ensuring compatibility with CUDA and Nvidia drivers for GPU acceleration.

2.6 Model Training

2.6.1 SAM – Segmentation Model

SAM /Kirillov et al. 2023/ is composed of three primary components: the image encoder, the prompt encoder, and the mask decoder. The image encoder converts the RGB image into a dense latent embedding that captures spatial and semantic details. The prompt encoder converts the input prompt into a dense positional embedding, representing the region of interest. The mask decoder then receives both the image embeddings and the prompt embeddings, integrating them through attention mechanisms to generate a binary segmentation mask corresponding to the specified region. In this study, the ViT-L (Vision Transformer-Large) variant of SAM was used due to its feature extraction capabilities.

The fine-tuning of SAM was performed by freezing the weights of both the image encoder and the prompt encoder, leaving only the mask decoder as trainable (see *Tab. 2* for hyperparameters). This approach is justified by the general nature of the pretrained encoders; they have already learned robust, universal representations of visual features and spatial relationships from their large-scale pre-training. Freezing these large components drastically reduces the number of trainable parameters, which in turn significantly decreases memory usage, shortens the training time, and minimises the risk of overfitting to the relatively small and specialised damage dataset. By focusing the training exclusively on the mask decoder, the model efficiently learns how to translate the existing, high-quality embeddings into our specific damage segmentation labels while preserving the general knowledge embedded through pre-training.

We constructed a custom instance-level dataset to fine-tune SAM. Each training example consisted of a patch image, its corresponding segmentation mask, and a bounding-box prompt representing the damaged region. These bounding boxes were obtained from the previously generated YOLOv8-format annotations, which defined the spatial location and class of each damage instance.

During dataset creation, individual images often contained several distinct damaged components, each annotated with its own class-labelled bounding box to represent different damage categories (e.g., walls, windows, debris), as illustrated in *Fig. 5*. Since SAM is trained to segment one object per prompt, the multi-class

data must first be converted into a series of single-instance entries. A custom script was developed that iterated through all annotations of an image and treated each bounding box as an independent training instance, while reusing the same image and mask paths. This approach expanded the effective dataset size, allowing the model to learn instance-level segmentation, where each prompt focuses on a single damaged region.

This structure is essential for fine-tuning SAM, as the model learns to generate segmentation masks conditioned on specific prompts. By training on individual bounding boxes, the model develops the ability to respond to precise spatial cues during inference, producing accurate, localised segmentations when given bounding-box prompts generated by the object detection model.

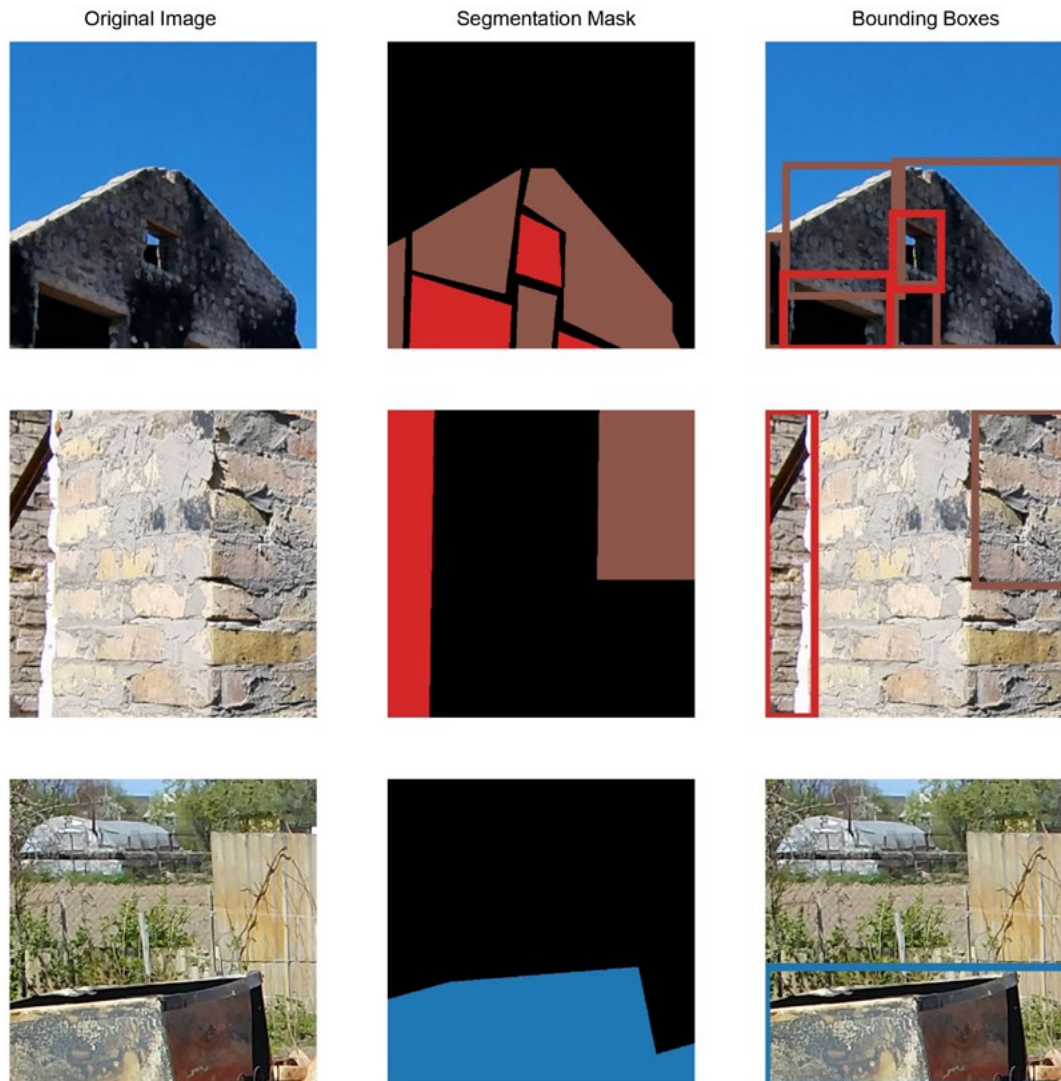


Fig. 5 | Example of annotation after preprocessing; original image (left), segmentation mask (centre; brown: damaged walls, red: damaged windows, blue: debris), and bounding boxes (right; colours match the mask colours)

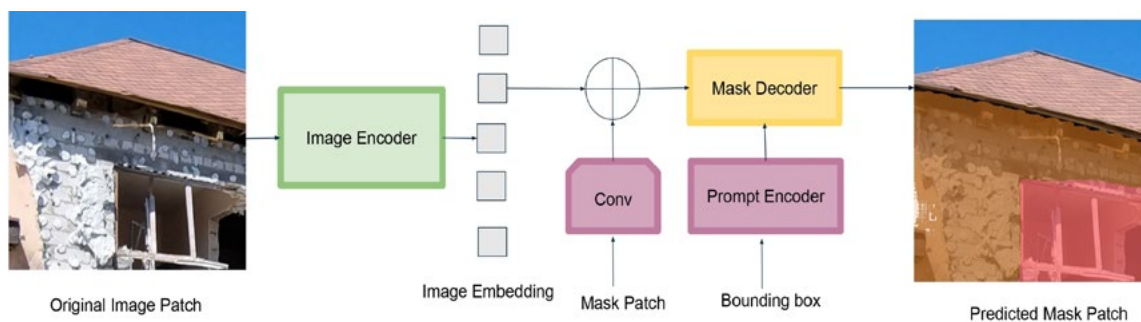


Fig. 6 | Segment anything model architecture overview

Parameter	Value
Optimiser	Adam
Learning Rate	1×10^{-5}
Batch Size	2
Loss Function	Dice-Focal Loss
Epochs	10

Tab. 2 | SAM Model Parameters

2.6.2 YOLOv8 – Object Detection Model

To perform object detection, the YOLOv8-Large architecture was employed. YOLOv8 is a single-stage object detector that consists of the CSPNet-based backbone by /Wang et al. 2020/ for feature extraction, a Feature Pyramid Network by /Lin et al. 2016/ combined with a Path Aggregation Network by /Liu et al. 2018/ for multi-scale feature fusion, and a lightweight, decoupled detection head for predicting object classes and bounding box coordinates. This architecture provides high detection accuracy and computational efficiency, making it well-suited for integration within the two-stage framework used in this study.

The object detection model was fine-tuned to identify the three damage classes using bounding-box annotations generated during the preprocessing stage. The YOLOv8-Large variant was initialized with weights pretrained on the COCO dataset, providing strong general-purpose visual feature representations. Fine-tuning on our task-specific dataset, using the hyperparameters listed in Tab. 3, enabled the model to adapt the pretrained features to the distinctive textures, colour variations, and structural patterns associated with the chosen damage classes and scenarios.

The detector in this workflow is not an end in itself but serves as a supporting module that provides stable, class-labelled prompts for the segmentation model. Therefore, the fine-tuning strategy prioritised a careful balance between detection accuracy and inference speed, ensuring that bounding boxes remained consistent across heterogeneous damage conditions and varying viewing perspectives. This balance is crucial, as the detector effectively acts as the gatekeeper for the second stage: missed detections can result in incomplete segmentation, while overly permissive detections may generate false prompts and introduce downstream noise. Using this strategy, the model was able to generalise across diverse damage scenarios while maintaining fast inference performance, producing reliable bounding boxes for subsequent segmentation by SAM.

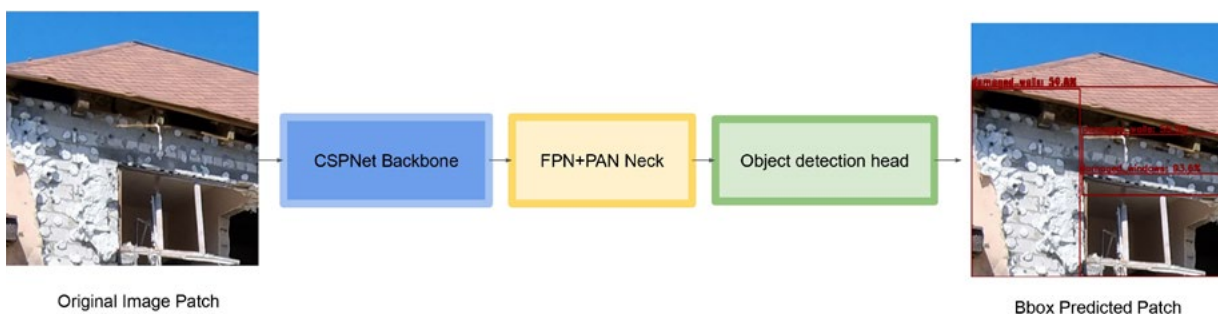


Fig. 7 | YOLOv8 Model Architecture Overview

Parameter	Value
Optimiser	Stochastic Gradient Descent
Learning Rate	0.01 (with cosine decay)
Batch Size	8
Loss Function	Composite loss (Bbox Regression + Classification)
Epochs	350

Tab. 3 | YOLOv8 Model Parameters

2.7 EVALUATION METRICS

To quantitatively evaluate the performance of the object detection and segmentation models, several evaluation metrics were used. These metrics assessed the accuracy of detection and localization, classification performance, and segmentation quality.

2.7.1 Object Detection

The object detection model was evaluated using three standard metrics: precision, recall /Oksuz et al. 2018/, and mean Average Precision (mAP, /Otani et al. 2022/) at IoU levels of 0.5 (mAP@0.5) and 0.5 to 0.95 (mAP@0.5:0.95).

2.7.2 Segmentation

The segmentation performance was evaluated using pixel-wise classification metrics, specifically precision, recall, and the Dice Coefficient /Li et al. 2020/ which measures the spatial overlap.

3 RESULTS AND EVALUATION

3.1 YOLOv8 Object Detection Performance

The YOLOv8 model demonstrated moderate performance across both validation and test datasets, as summarised in Tab. 4. The overall precision on the validation set and the test set indicates that the detector produced relatively few false positives, effectively distinguishing damaged and undamaged regions. The recall values suggest that a significant fraction of the damaged areas were successfully detected but there is potential for improvement.

Parameter	Data	Overall	DaWa	DaWi	De
Precision	V	0.74	0.68	0.76	0.78
	T	0.72	0.60	0.75	0.81
Recall	V	0.54	0.46	0.58	0.59
	T	0.58	0.44	0.65	0.65
mAP	V	0.62	0.53	0.66	0.69
	T	0.64	0.47	0.68	0.75
mAP95	V	0.46	0.37	0.47	0.47
	T	0.46	0.3	0.43	0.63

Tab. 4 | Performance of fine-tuned YOLOv8 model on validation(V) and test (T) data; DaWa = Damaged Walls, DaWi = Damaged Windows, De = Debris, mAP = mAP0.5, mAP95 = mAP0.5:0.95

Within the class-performance analysis, the highest precision and recall values suggest that the model reliably identifies debris. This might be due to its visual contrast, irregular textures, and larger spatial coverage in most images. Damaged windows and damaged walls are associated with slightly to considerably lower precision, recall and mAP values, reflecting their relatively more limited visual presence and structural similarity to undamaged regions.

The overall mAP@0.5 scores indicate good localisation capability at standard IoU thresholds, while the more stringent mAP@0.5:0.95 scores highlight the difficulty of achieving perfect spatial alignment across diverse object scales. Among all classes, debris again achieved the best localisation performance, whereas damaged walls lagged behind, likely due to their less distinct boundaries and similar texture to undamaged walls and sometimes even debris.

Overall, the results indicate that YOLOv8 generalises effectively across unseen samples and performs reliably for prominent and texture-rich parts of the images (here: debris). However, damaged windows and walls remain more challenging to detect and will require improvements of the models.

3.2 SAM Segmentation Performance

The fine-tuned Segment Anything Model achieved good segmentation results on both validation and test datasets, as shown in *Tab. 5*. The model demonstrated consistent generalisation with an overall

Parameter	Data	Overall	DaWa	DaWi	Debris
Precision	V	0.89	0.88	0.92	0.89
	T	0.90	0.87	0.91	0.91
Recall	V	0.93	0.93	0.93	0.93
	T	0.94	0.92	0.94	0.94
Dice Score	V	0.90	0.90	0.92	0.90
	T	0.91	0.89	0.93	0.92

Tab. 5 | Performance of fine-tuned SAM on validation(V) and test (T) data; DaWa = Damaged Walls, DaWi = Damaged Windows, De = Debris

Dice score of 0.91, meaning that, on average, the predicted damaged regions and their corresponding ground-truth areas overlap by 91 %, and with precision and recall exceeding 90 %. This indicates that the model not only avoids false detections but also captures nearly all relevant damaged areas. This indicates that the fine-tuning approach successfully adapted the pretrained model to the damage segmentation task.

Class-wise analysis reveals that the model achieved the highest Dice score, accompanied by precision, and recall in segmenting damaged windows. Segmentation of debris was also performed robustly. This could be due to its irregular yet visually distinctive texture patterns and larger coverage areas.

In contrast, the model achieved slightly lower Dice scores and precision, with recall remaining high, for damaged walls. This suggests that the model occasionally over-segmented surrounding regions due to the complex visual similarity between damaged and undamaged wall surfaces, shadows, and, in some cases, due to annotation errors.

The overall results demonstrate that the SAM model effectively captures fine-grained structural details while maintaining strong generalisation between validation and test sets. The high recall values across all classes indicate that most damaged areas were correctly identified, while the consistent Dice performance confirms precise boundary localisation. Together, these findings validate the effectiveness of using SAM's fine-tuned mask decoder for automated segmentation of terrestrial damage imagery.

4 MODEL INFERENCE

During inference, the two trained models, YOLOv8 and SAM, operate sequentially to generate structured predictions from unseen terrestrial images. The workflow begins with the input of a raw image, captured by a digital camera or a smartphone.

Each input image is divided into overlapping patches of the size 512×512 . This ensures consistency with the training resolution and enables the models to handle high-resolution images efficiently without loss of spatial details. Each patch is processed independently through the detection and segmentation pipeline, allowing localised analysis of damages across large or complex scenes.

The YOLOv8 detector analyses each patch to identify and localise potentially damaged components, according to the three predefined classes. For each detection, the model outputs a bounding box, a class-label, and a confidence score. Only predictions exceeding a predefined confidence threshold (we use 0.5) are retained to reduce the number of false positives.

The fine-tuned SAM model then receives each of these bounding boxes as spatial prompts, along with the corresponding image patch for segmentation. The mask decoder of SAM integrates these inputs to produce the binary segmentation masks.

Since we perform inference at the patch level, we implemented a post-processing step to reconstruct complete segmentation maps from the individual patch predictions. This step reassembles these patch outputs into their original spatial arrangement by grouping all patch files belonging to the same image based on their encoded coordinates in the filenames. Overlapping areas are merged by

direct pixel replacement to ensure spatial continuity and visual integrity. The reconstructed images are saved as RGB overlays, producing high-resolution, visually interpretable segmentation maps.

5 SUMMARY AND CONCLUSIONS

Damage assessment at large scale as, e. g., necessary in Ukraine to support planning reconstruction, requires data and automation. Reliance on experts in the field is not scalable to large areas, and top view data as provided by sensors on satellites or aircraft, do not have sufficient spatial resolution and 3d coverage to enable detailed damage assessment at the scale of individual buildings. The use of terrestrial imagery like photographs collected by local authorities, civilians, or aid workers using smartphones in a participatory manner can complement other data sources in this respect.

We proposed a framework for damage assessment using such images, herein. The quantitative results achieved herein are encouraging and indicate that the framework can serve as a foundational component for analytical and decision-support applications in recovery planning. By automating the localisation and pixel-level delineation of damaged regions in terrestrial imagery, it reduces reliance on slow, resource-intensive manual inspection and supports assessment in contexts where access and safety constraints limit in-situ surveys. In a hybrid setting, overhead data can be used to identify hotspots and monitor change at settlement scale or larger scales, while the proposed pipeline provides the fine-grained information required for detailed damage characterisation at building scale.

Once damaged structures and debris are detected and classified, their attributes can be extracted, structured, and exported in standardised, GIS-compatible formats for higher-level analysis – for example, to support clearance priority mapping based on debris occurrence and extent, and to enable consistent cross-settlement comparison over time. From an administrative perspective, these

structured outputs can also support verification and compensation workflows by providing a transparent and repeatable basis for documenting damage classes and indicative severity.

Future work should focus on refining requirements, capture protocols, classification schemes and output formats for practical use under active legal regulations, on expanding and diversifying the dataset and improving annotation consistency, in close collaboration with Hromadas (local authorities in Ukraine) and operational partners, as well as on quality control of information extracted from citizen-supplied images.

While the pipeline is demonstrated here for war-related building damage in Ukraine, the underlying solution architecture is transferable to other geographic regions, other damage scenarios, and other types of infrastructure.

REFERENCES

- Aimaiti, Y.; Sanon, C.; Koch, M.; Baise, L. G.; Moaveni, B. (2022): War Related Building Damage Assessment in Kyiv, Ukraine, Using Sentinel-1 Radar and Sentinel-2 Optical Images. In: *Remote Sensing* 14(2022), 6239.
- Alisjahbana, I.; Li, J.; Zhang, Y. (2024): DeepDamageNet: A two-step deep-learning model for multi-disaster building damage segmentation and classification using satellite imagery. *arXiv preprint*, <https://arxiv.org/abs/2405.04800>.
- Buslaev, A.; Iglovikov, V. I.; Khvedchenya, E.; Parinov, A.; Druzhinin, M.; Kalinin, A. A. (2020): Albumentations: Fast and Flexible Image Augmentations. In: *Information* 11(2020), 125.
- Cong, Y.; Khanna, S.; Meng, C.; Liu, P.; Rozi, E.; He, Y.; Burke, M. M.; Lobbell, D. B.; Ermon, S. (2022): SatMAE: Pre-training Transformers for Temporal and Multi-Spectral Satellite Imagery. In: *Proceedings 36th Conference on Neural Information Processing Systems (NeurIPS 2022)*, 197–211.
- Danyliv, N.; Savchyn, I.; Serebrianskyi, D.; Austin, T. (2026): Methodology for Collecting and Processing Data to Diagnose the Technical Condition of a Building: A Case Study of the State Tax University in Irpin (Ukraine). In: *allgemeine vermessungs-nachrichten (avn)* 133(2026)1, 16–25.

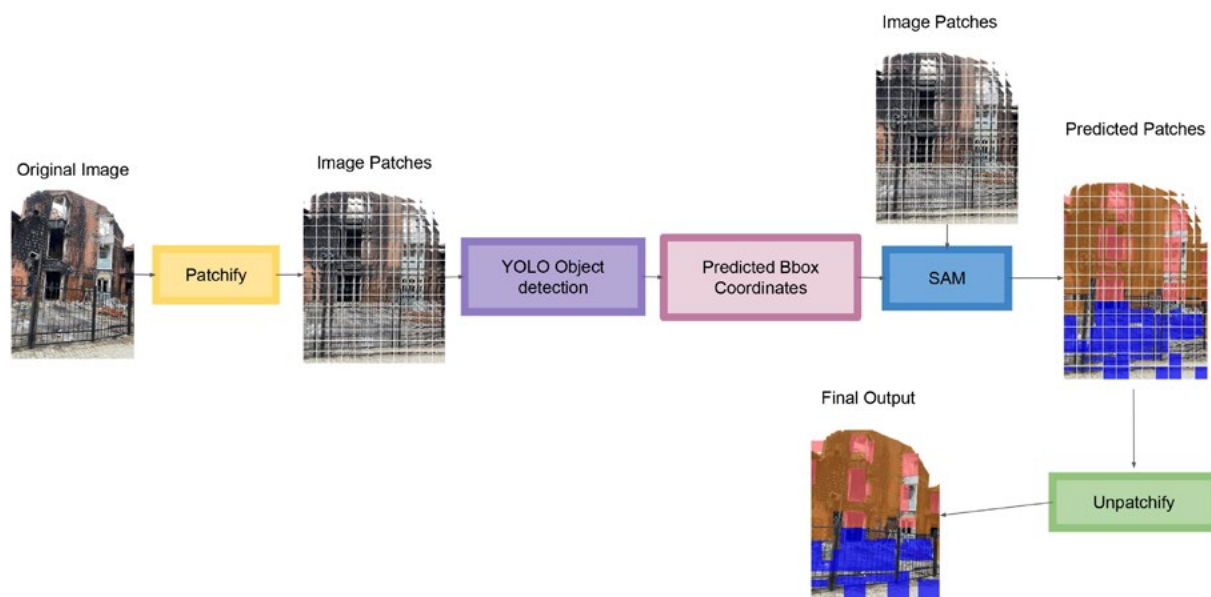


Fig. 8 | Model Inference Architecture (in the predicted patches and output, brown indicates damaged walls, blue represents debris, and red denotes damaged windows)

Dai, M.; Ward, W. O. C.; Meyers, G.; Tingley, D. D.; Mayfield, M. (2021): Residential building facade segmentation in the urban environment. In: *Building and Environment* 199(2021), 107921.

Dietrich, O.; Peters, T.; Sainte Fare Garnot, V.; Sticher, V.; Whelan, T.; Schindler, K.; Wegner, D. (2025): An open-source tool for mapping war destruction at scale in Ukraine using Sentinel-1 time series. In: *Communications Earth & Environment* 6(2025), 215.

Jocher, G.; Chaurasia, A.; Qiu, J. (2023): Ultralytics YOLOv8 (Software). <https://github.com/ultralytics/ultralytics> (05.01.2026).

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; Dollár, P.; Girshick, R. (2023): Segment Anything. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 4015–4026.

Li, X.; Sun, X.; Meng, Y.; Liang, J.; Wu, F.; Li, J. (2020): Dice Loss for Data-imbalanced NLP Tasks. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 465–476.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. (2016): Feature Pyramid Networks for Object Detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2117–2125.

Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. (2018): Path Aggregation Network for Instance Segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 8759–8768.

Nabiee, S.; Harding, M.; Hersh, J.; Bagherzadeh, N. (2022): Hybrid U-Net: Semantic segmentation of high-resolution satellite images to detect war destruction. In: *Machine Learning with Applications* 9(2022), 100381.

Neyman, J. (1934): On the two different aspects of the representative method: The method of stratified sampling and the method of purposive selection. In: *Journal of the Royal Statistical Society* 97(1934)4, 558–625.

Oksuz, K.; Cam, B. C.; Akbas, E.; Kalkan, S. (2018): Localization recall precision (LRP): A new performance metric for object detection. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, 520–536.

Otani, M.; Saito, S.; Taniguchi, R. (2022): Optimal correction cost for object detection evaluation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 16782–16791.

Tkachenko, M.; Malyuk, M.; Holmanyuk, A.; Liubimov, N. (2025): Label Studio: Data labeling software. <https://github.com/HumanSignal/label-studio> (04.01.2026).

Wang, B.; Zhang, J.; Zhang, R.; Li, Y.; Li, L.; Nakashima, Y. (2023): Improving facade parsing with Vision Transformers and line integration. In: *Advanced Engineering Informatics* 60 (2024), 102463.

Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y. M. (2020): CSPNet: A new backbone that can enhance learning capability of CNN. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*.

Wang, S.; Kang, Q.; She, R.; Tay, W. P.; Navarro, D. N.; Hartmannsgruber, A. (2022): Building facade parsing R-CNN. *arXiv preprint*, <https://arxiv.org/abs/2205.05912>.

Wu, C.; Zhang, F.; Xia, J.; Xu, Y.; Li, G.; Xie, J.; Du, Z.; Liu, R. (2020): Building damage detection using U-Net with attention mechanism from pre- and post-disaster remote sensing datasets. In: *Remote Sensing*, 13(2020)5, 905.

Yakovenko, M.; Nesterenko, O.; Stelmakh, D.; Babik, K.; Zorin, Y.; Ben, I.; Horkovchuk, I. (2026): Assessing the Technical Condition of a War-Damaged Residential Building in Ukraine Based on Integrated Geodetic and Photogrammetric Surveys. In: *allgemeine vermessungs-nachrichten (avn)* 133(2026)1, 3–15.

AUTHORS



Anand Gajaria

UNITED NATIONS INNOVATION
TECHNOLOGY ACCELERATOR
FOR CITIES (UNITAC)

HAFENCITY UNIVERSITY HAMBURG

Hongkongstraße 8 | 20457 Hamburg | Germany
anand.gajaria@un.org | ORCID: 0009-0002-4760-2140



Basil Roth

SWISS NETWORK WITH UKRAINE

Gladbachstr. 44 | 8044 Zürich | Switzerland
basil.roth@swissnetworkwithukraine.org | ORCID: 0009-0001-4177-9845



Jonathan Banz

SWISS NETWORK WITH UKRAINE

Gladbachstr. 44 | 8044 Zürich | Switzerland
jonathan.banz@swissnetworkwithukraine.org |
ORCID: 0009-0003-3994-7461



Prof. Dr. Andreas Wieser

ETH ZÜRICH

INSTITUTE OF GEODESY
AND PHOTOGRAMMETRY

Stefano-Francini-Platz 5 | 8093 Zürich | Switzerland
andreas.wieser@geod.baug.ethz.ch | ORCID: 0000-0001-5804-2164



Prof. Dr. Gesa Ziemer

UNITED NATIONS INNOVATION
TECHNOLOGY ACCELERATOR
FOR CITIES (UNITAC)

HAFENCITY UNIVERSITY HAMBURG

Hongkongstraße 8 | 20457 Hamburg | Germany
gesa.ziemer@un.org | ORCID: 0000-0002-1491-0991