INTERNATIONAL JOURNAL OF INFORMATIVE & FUTURISTIC RESEARCH (IJIFR)
ISSN: 2347-1697
VOLUME 13||ISSUE 8|| APRIL 2026
An Open Access, Peer Reviewed, Cross Reference DOI & Google Scholar Indexed International Research Journal
UGC Guidelines Based || IJIFR Impact Factor= 8.057
Estd Year: 2013 || Website: www.ijifr.org || Email: editors@ijifr.org
Availaible Online at: https://www.ijifr.org/

DOI: https://doi.org/10.64672/IJIFR/26.04.13.08.024 PUBLISHED ON: APRIL 18, 2026

DISASTERVISION AI: A SATELLITE-BASED DEEP LEARNING FRAMEWORK FOR AUTOMATED DISASTER DAMAGE ASSESSMENT

K. Bhavani Sankar [1], V.Vijayalakshmi [2] , Usha Rani [3]

[1]M.C.A. Student, [2]Assistant Professor, [3] Professor
[1,2,3] Department of Computer Applications,
Viswam Engineering College, Madanapalle, Andhra Pradesh, India

ABSTRACT

Natural disasters such as earthquakes, floods, hurricanes, and wildfires cause extensive damage to infrastructure and human life, making rapid and accurate damage assessment a critical requirement for effective disaster response. Traditional ground-based assessment techniques are time-consuming, risky, and limited in spatial coverage, which delays emergency decision-making processes. To address these limitations, this paper presents DisasterVision AI, an automated satellite imagery analysis system that leverages deep learning for large-scale building damage assessment. The proposed system utilizes a modified Single Shot MultiBox Detector (SSD) with a VGG-16 backbone, enhanced to process six-channel input by combining pre-disaster and post-disaster satellite images. This dual-input architecture enables the model to learn visual differences between temporal image pairs, improving damage detection accuracy. The model is trained using the xView2 dataset, which provides annotated satellite imagery with four damage categories: no-damage, minor-damage, major-damage, and destroyed. The system incorporates advanced training techniques including data augmentation using Albumentations, OneCycle learning rate scheduling, and AdamW optimization for efficient convergence. Performance evaluation is conducted using Mean Average Precision (mAP) metrics across multiple IoU thresholds. Additionally, Non-Maximum Suppression (NMS) is applied for refining detection outputs. Experimental results demonstrate that DisasterVision AI provides fast, scalable, and reliable damage assessment, making it a valuable tool for disaster management authorities and emergency response teams.

KEYWORDS: *Disaster Damage Assessment; Deep Learning; Satellite Imagery; Object Detection; SSD; xView2 Dataset*

PAPER CITATION:

Sankar, K.B., Vijayalakshmi, V., Rani, U.:"DisasterVision AI: A Satellite-Based Deep Learning Framework for Automated Disaster Damage Assessment", International Journal of Informative & Futuristic Research (IJIFR), Vol. (13) (8), April 2026, pp. 1070-1085.IJIFR paper ID: 2026/04/IJIFR /V13/E8/024
https://doi.org/10.64672/IJIFR/26.04.13.08.024

ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

1. INTRODUCTION

The increasing frequency and severity of natural disasters have become a major global concern, significantly impacting human lives, infrastructure, and economies. Events such as earthquakes, floods, hurricanes, and wildfires result in large-scale destruction, requiring immediate and efficient response mechanisms. One of the most critical aspects of disaster management is rapid damage assessment, which enables authorities to allocate resources effectively and prioritize rescue operations.

Traditional damage assessment methods rely on manual field surveys conducted by engineers and emergency response teams. Although these methods provide accurate and detailed information, they are time-intensive, labor-intensive, and often unsafe in disaster-affected areas. Moreover, large-scale disasters affecting wide geographical regions make it nearly impossible to perform timely assessments using conventional approaches.

With the advancement of remote sensing technologies, satellite imagery has emerged as a powerful alternative for disaster monitoring. Satellite images provide wide-area coverage and can be captured within hours after a disaster. However, manual analysis of such imagery is still time-consuming and requires expert knowledge, limiting its effectiveness during critical response periods.

Artificial Intelligence (AI), particularly deep learning-based computer vision techniques, offers a promising solution for automating disaster damage assessment. By analyzing pre-disaster and post-disaster satellite images, AI models can identify structural changes and classify damage levels efficiently. The availability of large-scale datasets such as xView2 has further accelerated research in this domain.

In this context, **DisasterVision AI** introduces an innovative approach by reformulating the damage assessment problem as an object detection task instead of traditional semantic segmentation. This approach provides bounding box outputs that are easier to interpret and directly usable for decision-making. The integration of a six-channel input mechanism enhances the model's ability to capture temporal variations, leading to improved detection accuracy.

1.2 Project Objectives

The primary objectives of DisasterVision AI are:

- To develop an automated system for satellite-based disaster damage assessment
- To implement a six-channel deep learning model combining pre- and post-disaster images
- To classify building damage into multiple severity levels
- To improve detection speed and scalability using single-stage object detection
- To provide interpretable outputs for real-time disaster response

1.3 Problem Statement

The key challenge in disaster management is the delay between data acquisition and actionable insight generation. While satellite imagery can be obtained quickly, extracting meaningful information from it remains a bottleneck due to manual processing limitations.

Existing AI approaches predominantly use semantic segmentation, which produces pixel-level outputs that are complex to interpret and computationally expensive. These limitations reduce their effectiveness in real-time disaster scenarios.

Therefore, there is a need for a system that:

- Provides fast and scalable damage detection
- Produces easily interpretable outputs
- Works efficiently on large-scale satellite imagery

DisasterVision AI addresses this gap by leveraging object detection techniques to deliver rapid, accurate, and actionable damage assessment results.

ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

2. LITERATURE SURVEY

Literature Survey forms the foundation for understanding the research gap, technical direction, and methodological choices in DisasterVision AI. In the domain of satellite-based disaster damage assessment, several approaches have been proposed using remote sensing, computer vision, deep learning, and change detection frameworks. These studies collectively demonstrate that automated analysis of pre-disaster and post-disaster imagery can significantly improve the speed and quality of disaster response. DisasterVision AI builds on this body of work by adopting an object detection perspective for damage localization and severity classification.

Early research in disaster assessment primarily relied on manual interpretation of aerial and satellite images. Although such methods offered reasonable accuracy, they were highly dependent on domain experts and were not suitable for time-critical emergency scenarios. As remote sensing archives expanded and high-resolution imagery became more accessible, machine learning methods were introduced to automate the identification of damaged infrastructure. Traditional image processing and shallow learning methods, however, struggled to generalize across different disaster types, illumination conditions, and geographic contexts.

A major advancement in this field came through the adoption of deep learning-based semantic segmentation models. Architectures such as U-Net and DeepLab enabled pixel-wise classification of post-disaster imagery, allowing researchers to generate detailed damage maps. These methods were especially useful in identifying affected built-up regions and quantifying damaged areas. However, segmentation outputs often required additional processing for structure-level interpretation, and the computational burden of high-resolution segmentation made real-time deployment more challenging in operational settings.

Another important line of research is based on change detection between pre-disaster and post-disaster images. These methods explicitly model temporal variation and aim to capture differences caused by structural damage. By comparing image pairs, change detection frameworks improve robustness against background variations and help identify newly damaged zones. Yet, many of these systems focus mainly on whether a change has occurred, rather than providing a finer classification of damage severity. As a result, their usefulness in prioritizing emergency response remains somewhat limited.

Object detection has emerged as a practical alternative for disaster analysis because it directly identifies and localizes individual damaged buildings. Two-stage detectors such as Faster R-CNN have shown strong detection accuracy, especially for small and densely packed structures. These models first generate candidate regions and then classify them, leading to improved localization performance. However, this two-stage process increases computational cost and inference time, which can be a limitation when rapid large-scale assessment is required immediately after a disaster.

Single-stage detectors such as SSD offer a better balance between speed and accuracy. SSD performs localization and classification in a single forward pass, making it suitable for applications where faster inference is essential. The multi-scale detection mechanism of SSD further improves its ability to detect buildings of varying sizes in satellite imagery. This makes SSD a suitable architectural choice for DisasterVision AI, where both scalability and real-time applicability are important design goals.

The xView2 dataset has played a central role in advancing research on automated disaster damage assessment. It provides large-scale pre-disaster and post-disaster image pairs annotated with building footprints and four levels of damage severity: no-damage, minor-damage, major-damage, and destroyed. The dataset has enabled consistent benchmarking and encouraged the development of deep learning models that can move beyond binary damaged-versus-undamaged classification. DisasterVision AI uses this benchmark to train a multi-class detection model capable of providing more meaningful disaster intelligence.

More recent studies have also explored the use of foundation models, transformer-based vision systems, and geospatial representation learning. These methods benefit from large-scale pretraining and often

ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

achieve strong performance with limited task-specific tuning. However, they typically require high computational resources and complex deployment pipelines. For research prototypes and practical field-oriented systems, lightweight yet effective models such as SSD remain highly relevant.

From the literature, it is clear that existing methods provide strong foundations but also exhibit important limitations. Semantic segmentation offers fine-grained spatial detail but is difficult to interpret quickly. Change detection identifies visual differences but may not provide explicit severity categorization. Two-stage detectors improve precision but increase complexity and computational overhead. Therefore, there is a clear research need for a system that combines rapid inference, interpretable outputs, and multi-class damage severity analysis. DisasterVision AI addresses this gap by introducing a six-channel SSD-based framework that jointly processes pre-disaster and post-disaster satellite imagery for efficient building-level damage detection.

Table 1: Existing Methods from the literature

S. No.	Author / Approach	Method Used	Strength	Limitation
1	Manual Remote Sensing Analysis	Human interpretation	High detail	Slow and not scalable
2	U-Net / DeepLab based methods	Semantic segmentation	Pixel-level precision	Harder to interpret quickly
3	Change Detection models	Pre-post image comparison	Captures structural change	Limited severity interpretation
4	Faster R-CNN based methods	Two-stage object detection	High localization accuracy	Slower inference
5	SSD-based approaches	Single-stage object detection	Faster and scalable	Lower small-object sensitivity
6	Foundation/Transformer models	Large-scale pretrained vision models	Strong feature learning	Computationally expensive

3. SYSTEM PROPOSAL

The System Proposal section presents a detailed explanation of both existing methodologies and the proposed approach implemented in **DisasterVision AI**. This section highlights the limitations of current systems and justifies the need for the proposed deep learning-based framework.

3.1 Existing System

Existing systems for disaster damage assessment primarily rely on three major approaches: **semantic segmentation, change detection, and two-stage object detection models**.

Semantic segmentation-based methods such as U-Net and DeepLab classify each pixel of an image into different damage categories. These models generate detailed damage maps, which are useful for spatial analysis and visualization. However, they produce complex outputs that require further processing to identify individual buildings, making them less practical for immediate disaster response.

Change detection approaches analyze differences between pre-disaster and post-disaster satellite images. These methods focus on identifying changes in structures and landscapes caused by disasters. While effective in detecting variations, they often fail to categorize the severity of damage and may misinterpret environmental changes such as shadows or seasonal variations as damage.

Two-stage object detection models like Faster R-CNN have also been used for building damage detection. These systems first generate region proposals and then classify them into damage categories. Although they provide high accuracy and better localization, they involve higher computational complexity and slower inference speeds, which are not ideal for real-time disaster response scenarios.

Recent advancements also include transformer-based and foundation models trained on large-scale datasets. These models achieve high performance but require extensive computational resources, making them less suitable for rapid deployment in emergency situations.

3.1.1 Advantages of Existing System

Despite their limitations, existing systems offer several advantages:

- **High Accuracy:** Deep learning models, especially segmentation and two-stage detectors, provide high precision in identifying damaged areas.
- **Detailed Spatial Information:** Pixel-level segmentation offers fine-grained details about affected regions.
- **Robust Feature Learning:** Modern models leverage large datasets to learn complex visual patterns.
- **Wide Applicability:** These systems can be adapted for various remote sensing and disaster scenarios.

However, these advantages are often offset by issues related to interpretability, computational cost, and response time.

3.2 Proposed System

To overcome the limitations of existing methods, **DisasterVision AI** introduces a **single-stage object detection framework using a modified SSD (Single Shot MultiBox Detector)** with a six-channel input architecture.

The proposed system processes **pre-disaster and post-disaster satellite images simultaneously** by concatenating them into a six-channel input tensor. This enables the model to directly learn visual differences between the two temporal states, improving its ability to detect and classify damage.

The system is trained using the **xView2 dataset**, which provides labeled satellite imagery with four levels of damage severity:

- No Damage
- Minor Damage
- Major Damage
- Destroyed

The workflow of the proposed system includes:

- **Data Preprocessing:** Extraction of bounding boxes from polygon annotations and preparation of image pairs
- **Six-Channel Input Formation:** Combining pre- and post-disaster images into a unified input
- **Model Architecture:** Modified SSD300 with VGG-16 backbone
- **Training Strategy:** Use of AdamW optimizer and OneCycle learning rate policy
- **Evaluation:** Performance measurement using Mean Average Precision (mAP)
- **Post-Processing:** Application of Non-Maximum Suppression (NMS) to refine predictions

Unlike segmentation models, the proposed system produces **bounding box outputs**, which directly correspond to individual buildings. This makes the results more interpretable and actionable for disaster response teams.

Additionally, the use of a **single-stage detection model ensures faster inference**, enabling rapid processing of large-scale satellite imagery.

3.2.1 Disadvantages of the Proposed System

Although DisasterVision AI provides several improvements, it also has certain limitations:

- **Limited Small Object Detection:** SSD may struggle with detecting very small buildings in high-resolution satellite images
- **Bounding Box Approximation:** Converting polygon annotations into rectangular boxes may introduce minor inaccuracies
- **Training Complexity:** Requires high computational resources, especially GPU-based systems
- **Initial Layer Limitation:** The six-channel input layer is randomly initialized, which may affect early-stage learning
- **Prototype-Level Implementation:** Current implementation is research-oriented and requires further optimization for production deployment

4. System Architecture

4.1 Architecture Overview

The **DisasterVision AI** system is designed as a modular deep learning framework that processes satellite imagery for automated disaster damage assessment. The architecture integrates multiple components including data preprocessing, model training, and evaluation, ensuring scalability and efficiency in real-world applications.

The system follows a pipeline-based architecture where each module performs a specific function, and data flows sequentially from input acquisition to final damage prediction. The key objective of this architecture is to enable rapid and accurate identification of damaged buildings using pre-disaster and post-disaster satellite images.

4.2 Architecture Diagram

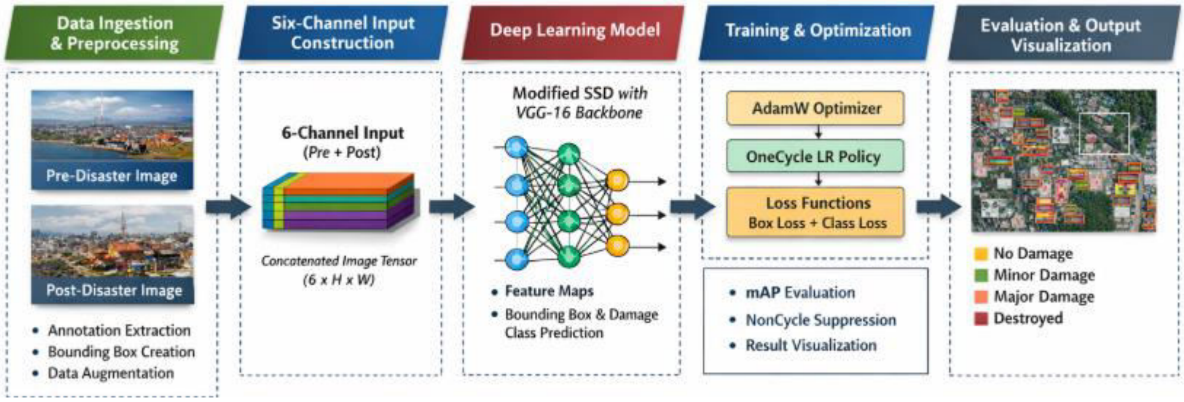


Figure 1: System Architecture of Disaster Vision AI

4.3 Architecture Description

The DisasterVision AI architecture consists of four major components:

a) **Data Ingestion and Preprocessing Module**

This module is responsible for collecting and preparing input data from the xView2 dataset. It includes:

- Loading pre-disaster and post-disaster satellite images
- Extracting building annotations from JSON files
- Converting polygon annotations into bounding boxes
- Mapping damage categories into numerical labels

The preprocessing stage ensures that the data is structured and formatted correctly for training the deep learning model.

b) **Six-Channel Input Construction**

A key innovation in this system is the **six-channel input representation**, where:

- First 3 channels → Pre-disaster image (RGB)
- Next 3 channels → Post-disaster image (RGB)

These are concatenated to form a **[6, H, W] tensor**, allowing the model to directly learn temporal differences between images. This approach improves the detection of subtle structural changes caused by disasters.

c) **Deep Learning Model (Modified SSD Architecture)**

The core of the system is a modified **SSD300 (Single Shot MultiBox Detector)** with a VGG-16 backbone. The modifications include:

- Input layer adjusted from 3 channels to 6 channels
- Classification head modified for 5 classes (background + 4 damage categories)
- Multi-scale feature extraction for detecting buildings of different sizes

The SSD architecture performs both localization and classification in a single forward pass, making it highly efficient for real-time applications.

d) **Training and Optimization Engine**

This module manages the learning process of the model. It includes:

- **Optimizer:** AdamW for better generalization
- **Learning Strategy:** OneCycle Learning Rate Policy
- **Loss Functions:** Bounding box regression loss + classification loss

The training process iteratively updates model parameters to minimize prediction errors and improve detection accuracy.

e) **Evaluation and Visualization Module**

After training, the model is evaluated using standard metrics:

- **Mean Average Precision (mAP)** for detection accuracy
- **IoU-based evaluation** for localization performance

Post-processing techniques such as **Non-Maximum Suppression (NMS)** are applied to remove redundant bounding boxes. The final output is visualized as satellite images with bounding boxes indicating damage categories.

4.4 Data Flow in Architecture

The overall data flow can be summarized as follows:

1. Satellite images are collected (pre & post disaster)
2. Data is preprocessed and annotated
3. Images are converted into six-channel tensors
4. The SSD model performs feature extraction and prediction
5. Predictions are refined using NMS
6. Final results are visualized and used for decision-making

4.5 Key Advantages of Architecture

- **Faster Processing:** Single-stage SSD enables real-time inference
- **Better Damage Detection:** Six-channel input captures temporal differences
- **Scalability:** Suitable for large-scale satellite image analysis
- **Interpretability:** Bounding box outputs are easy to understand

4.5 Flow Diagram

The operational flow of **DisasterVision AI** is designed in a sequential and modular manner to ensure efficient satellite-based disaster damage assessment. The flow begins with data acquisition and ends with visualization of detected damage categories. The system operates through three major phases: **data preparation, model training, and inference with evaluation**. This structured workflow enables the model to process temporal satellite images effectively and generate actionable outputs for disaster response teams.

In the **first phase**, pre-disaster and post-disaster satellite images are collected from the xView2 dataset. Corresponding annotation files are read, and polygon-based building labels are converted into rectangular bounding boxes. Each building is then assigned one of the four damage severity classes: **no-damage, minor-damage, major-damage, or destroyed**. After annotation extraction, the images undergo preprocessing such as normalization, resizing, and augmentation to prepare them for model training.

In the **second phase**, the pre-disaster and post-disaster RGB images are concatenated to form a **six-channel image tensor**. This six-channel input is fed into the modified SSD300 deep learning model with a VGG-16 backbone. The training engine applies forward propagation, computes classification and localization losses, performs backward propagation, and updates weights using the AdamW optimizer and OneCycle learning rate scheduler. During this process, the model gradually learns to identify structural changes and classify the extent of damage.

In the **third phase**, the trained model is used for inference on unseen test images. The model produces predicted bounding boxes, damage labels, and confidence scores. These predictions are refined through

ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

Non-Maximum Suppression (NMS) to eliminate duplicate detections. Finally, the processed outputs are evaluated using **Mean Average Precision (mAP)** and visualized on satellite imagery with color-coded damage labels. This makes the final result highly interpretable for practical emergency response planning.

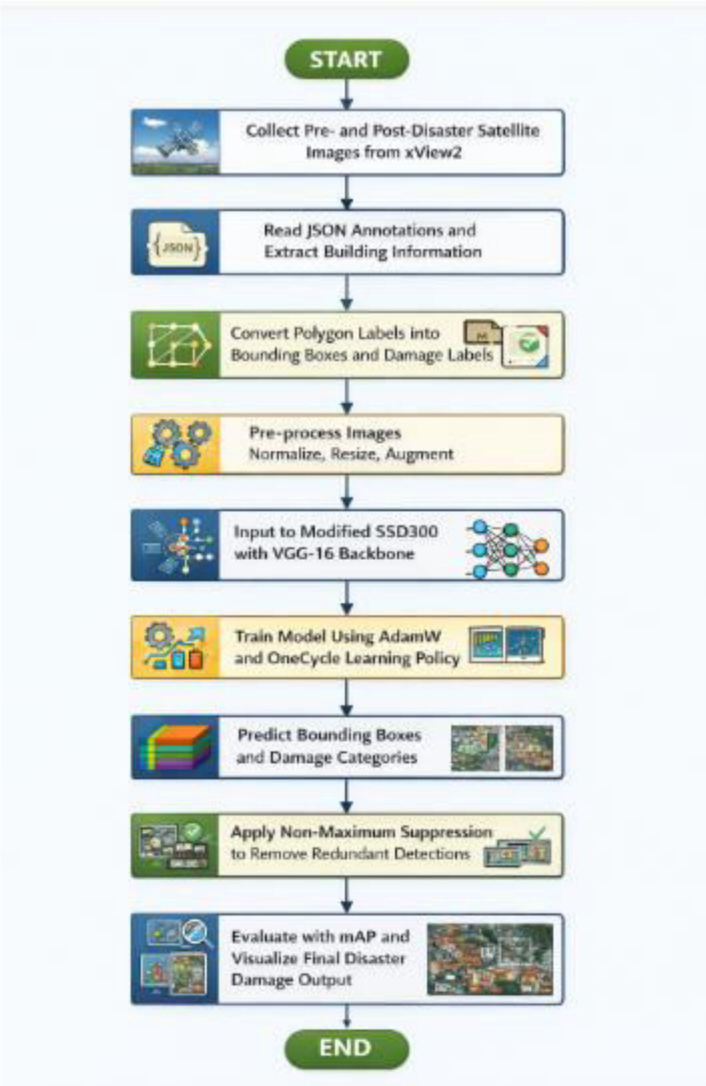


Figure 2: System Flow Diagram of DisasterVision AI

Flow Diagram Explanation

The flow diagram clearly illustrates how DisasterVision AI transforms raw satellite data into meaningful disaster intelligence. The process starts with collecting paired satellite images and annotations. After preprocessing, the temporal image pair is transformed into a six-channel tensor, which enables the system to capture before-and-after visual changes. The modified SSD network then performs simultaneous localization and classification of damaged structures. The final stage refines and visualizes the predictions, ensuring that rescue teams and authorities can quickly interpret the severity and location of affected buildings.

4.6 UML Diagrams

UML diagrams are used in DisasterVision AI to represent the functional behavior, structural organization, and interaction flow of the proposed system. These diagrams improve the clarity of the system design and help explain how different components collaborate to perform automated disaster damage assessment. Based on the project workflow, three UML diagrams are included: Use Case Diagram, Class Diagram, and Sequence Diagram.

4.6.1 Use Case Diagram

The Use Case Diagram illustrates the interaction between external actors and the DisasterVision AI system. In this project, two main actors are involved:

- **Data Scientist / Research Engineer**
- **Disaster Response Analyst**

The Data Scientist is responsible for preparing the dataset, configuring the model, training the system, and evaluating performance. This actor interacts with the system during development and experimentation. The Disaster Response Analyst uses the trained model to analyze satellite images, obtain damage predictions, and visualize results for decision-making.

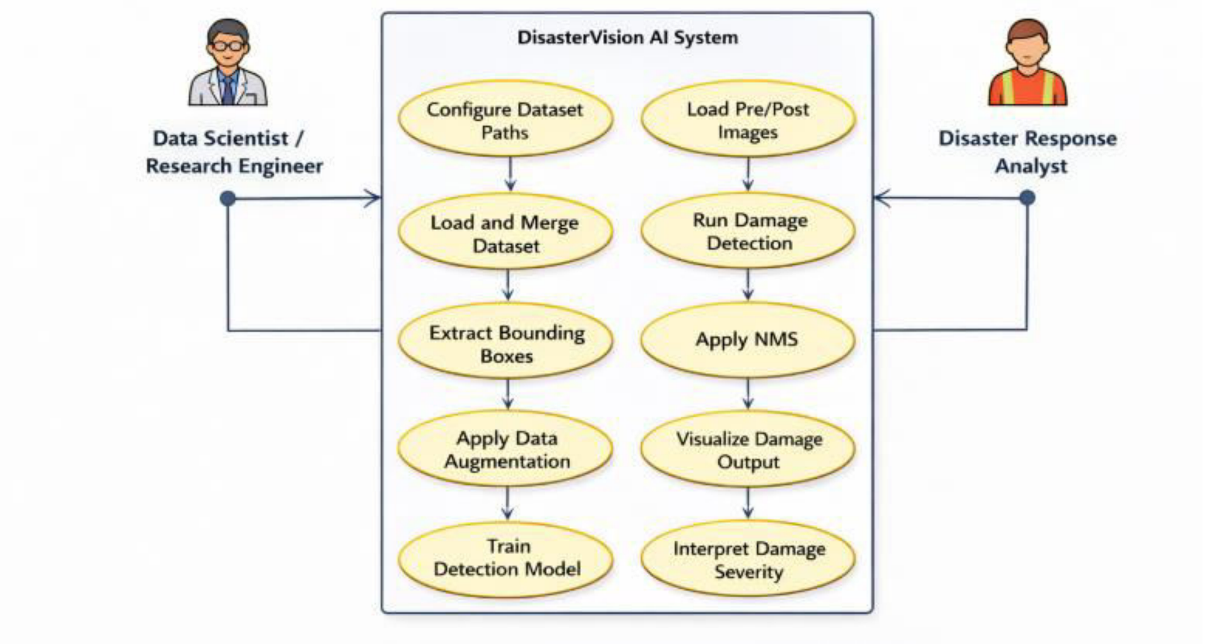


Figure 2: Usecase Diagram of DisasterVision AI

Use Case Explanation

The Use Case Diagram shows that the system supports both model development and practical application. During the training phase, the Data Scientist controls the pipeline from dataset preparation to model evaluation. During deployment or testing, the Disaster Response Analyst interacts with the trained model to generate damage assessment results from satellite imagery. This separation of actors reflects the real-world workflow of AI-based disaster management systems.

4.6.2 Class Diagram

The **Class Diagram** represents the static structure of the DisasterVision AI system. It identifies the major classes, their attributes, and their methods.

The primary classes in the system are:

- DataLoader
- ModifiedSSD
- Averager
- MetricLogger
- SmoothedValue
- Utility Functions Module

Class Diagram Explanation

The DataLoader class is responsible for reading satellite images, handling annotations, applying transformations, and returning six-channel tensors. The ModifiedSSD class represents the core detection model that processes temporal image pairs and predicts building damage categories. The Averager, MetricLogger, and SmoothedValue classes help monitor loss and training statistics during optimization.

ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

Together, these classes form the backbone of the system implementation and ensure modularity, reusability, and maintainability.

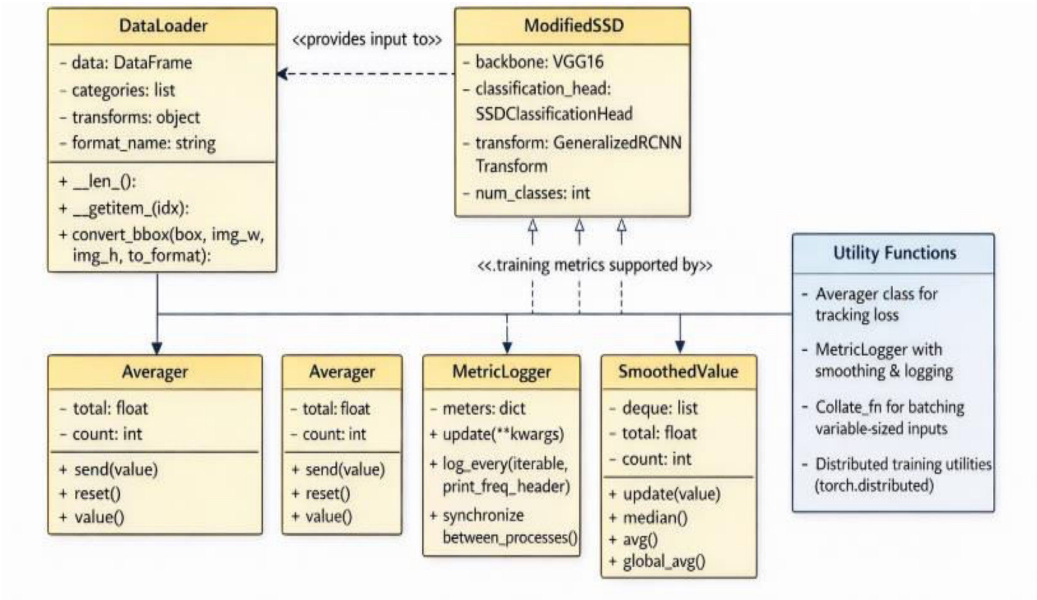


Figure 4: Class Diagram of DisasterVision AI

4.6.3 Sequence Diagram

The **Sequence Diagram** describes the step-by-step interaction among system components during the inference process. It explains how the analyst loads images, how the model processes them, and how the output is refined and visualized.

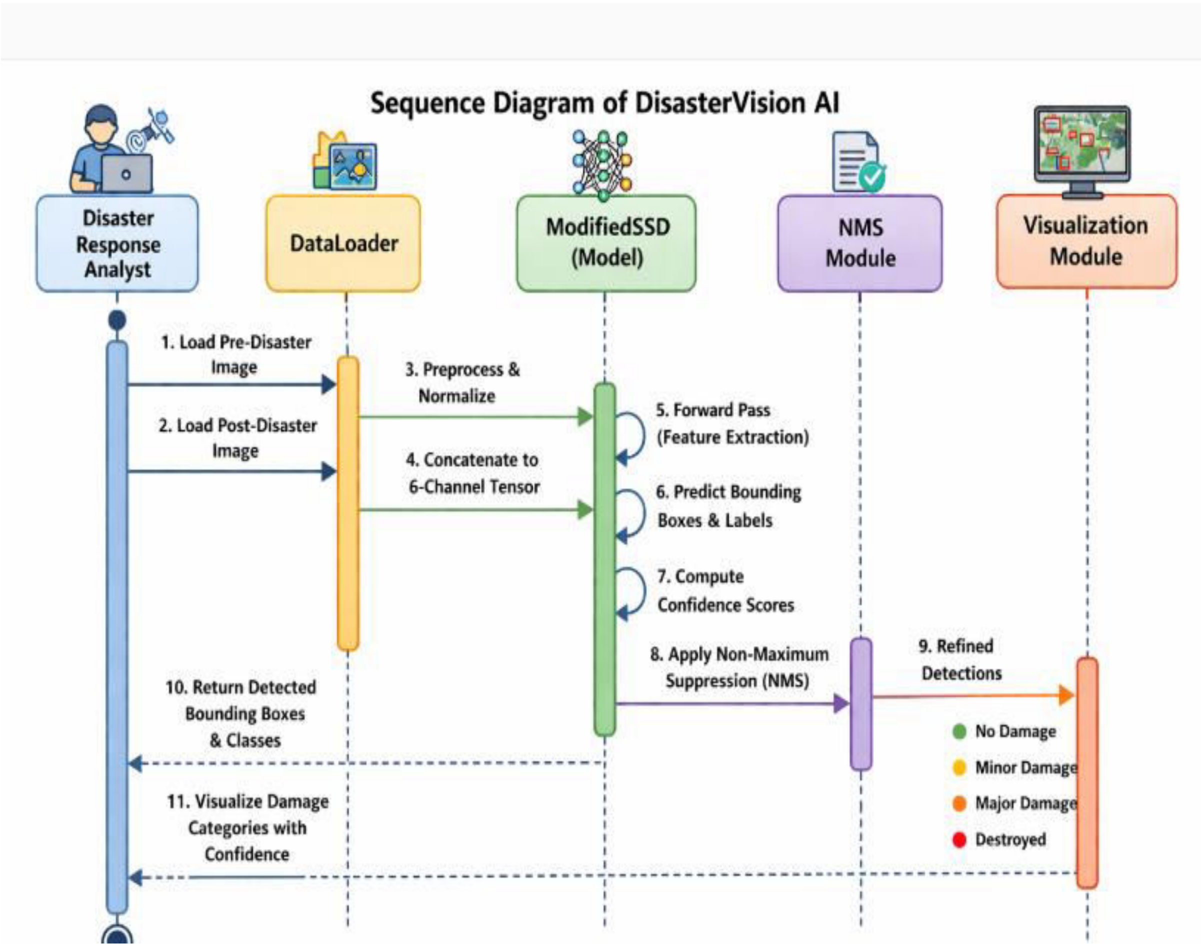


Figure 5: Sequence Diagram of Disaster Vision AI

ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

Explanation

The sequence begins when the Disaster Response Analyst loads the pre-disaster and post-disaster images. The **DataLoader** preprocesses these images, normalizes them, and combines them into a six-channel tensor. This tensor is then passed to the **ModifiedSSD** model, which predicts bounding boxes, class labels, and confidence scores. The predictions are forwarded to the **NMS Module**, which removes redundant detections. Finally, the **Visualization Module** overlay the refined results on the satellite image and displays the damage assessment in an interpretable form.

5. IMPLEMENTATION

The implementation of **DisasterVision AI** is designed as a modular deep learning pipeline that integrates data processing, model architecture, training strategies, and evaluation mechanisms. Each module is responsible for a specific function, ensuring flexibility, scalability, and ease of experimentation.

5.1 Modules

The system is divided into the following major modules:

(i) Annotation Parsing and Bounding Box Extraction Module
(ii) Data Merging and Dataset Construction Module
(iii) Custom Dataset and DataLoader Module
(iv) Model Architecture Modification Module
(v) Training and Optimization Module
(vi) Evaluation and Visualization Module
(vii) Utility Module (utils_.py)

5.2 Module Description

5.2.1 Annotation Parsing and Bounding Box Extraction Module

This module is responsible for extracting building information from annotation files provided in JSON format. The xView2 dataset uses polygon-based annotations in WKT (Well-Known Text) format. These polygons are converted into rectangular bounding boxes using geometric processing.

Each building is assigned a damage label based on severity:

- No Damage
- Minor Damage
- Major Damage
- Destroyed

The module ensures that invalid or corrupted geometries are filtered out, improving data quality and model reliability.

5.2.2 Data Merging and Dataset Construction Module

This module organizes the dataset by combining pre-disaster and post-disaster images along with their corresponding annotations. It creates a structured dataset in the form of a **Pandas DataFrame**, where each row contains:

- Pre-disaster image path
- Post-disaster image path
- Bounding boxes
- Damage labels

This structured representation simplifies data handling and improves training efficiency.

5.2.3 Custom Dataset and DataLoader Module

The Custom Dataset module extends PyTorch's Dataset class and acts as a bridge between raw data and the deep learning model. It performs the following operations:

- Loading pre-disaster and post-disaster images

ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

- Converting images from BGR to RGB format
- Normalizing pixel values
- Applying data augmentation (flip, resize, etc.)
- Converting bounding box formats
- Concatenating images into a six-channel tensor

The output of this module is:

- A six-channel input tensor
- Corresponding bounding boxes and labels

5.2.4 Model Architecture Modification Module

This module modifies the pre-trained **SSD300 VGG-16 model** to adapt it for disaster damage detection. The key modifications include:

- Changing the input layer from 3 channels to 6 channels
- Updating the classification head for 5 classes (including background)
- Adjusting normalization parameters for six-channel input

These changes allow the model to process temporal satellite images and perform multi-class damage classification.

5.2.5 Training and Optimization Module

This module manages the learning process of the model. It includes:

- **Optimizer:** AdamW
- **Learning Rate Policy:** OneCycle Learning Rate
- **Loss Functions:**
    - Bounding Box Regression Loss
    - Classification Loss

During training:

- The model performs forward propagation
- Loss is calculated
- Backpropagation updates model weights
- Learning rate is dynamically adjusted

Training is typically conducted for multiple epochs to ensure convergence and optimal performance.

5.2.6 Evaluation and Visualization Module

This module evaluates model performance and visualizes predictions. It includes:

- **Mean Average Precision (mAP)** for accuracy evaluation
- **IoU-based metrics** for localization performance
- **Non-Maximum Suppression (NMS)** to remove duplicate detections

The final output is visualized as satellite images with:

- Bounding boxes
- Damage category labels
- Confidence scores

5.2.7 Utility Module (utils_.py)

The utility module provides supporting functionalities such as:

- Loss tracking and averaging
- Training progress logging
- Distributed training support
- Batch collation for variable-sized inputs

This module enhances the efficiency and scalability of the training process.

5.3 Hardware Requirements

The system requires high computational resources due to deep learning operations on high-resolution satellite images.

ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

- Processor: Multi-core CPU
- RAM: Minimum 16 GB
- GPU: NVIDIA GPU (8 GB VRAM or higher recommended)
- Storage: SSD with sufficient capacity for dataset (~10 GB)

5.4 Software Requirements

The implementation uses the following software stack:

- Programming Language: Python 3.9
- Deep Learning Framework: PyTorch
- Computer Vision Library: TorchVision
- Data Augmentation: Albumentations
- Image Processing: OpenCV
- Data Handling: Pandas, NumPy
- Visualization: Matplotlib
- Evaluation: Torchmetrics

6. Results and Discussion

The performance of **DisasterVision AI** is evaluated using both quantitative metrics and qualitative visual analysis. The system is trained on the xView2 dataset using a modified SSD architecture with six-channel input, and its effectiveness is measured based on detection accuracy, convergence behavior, and interpretability of results.

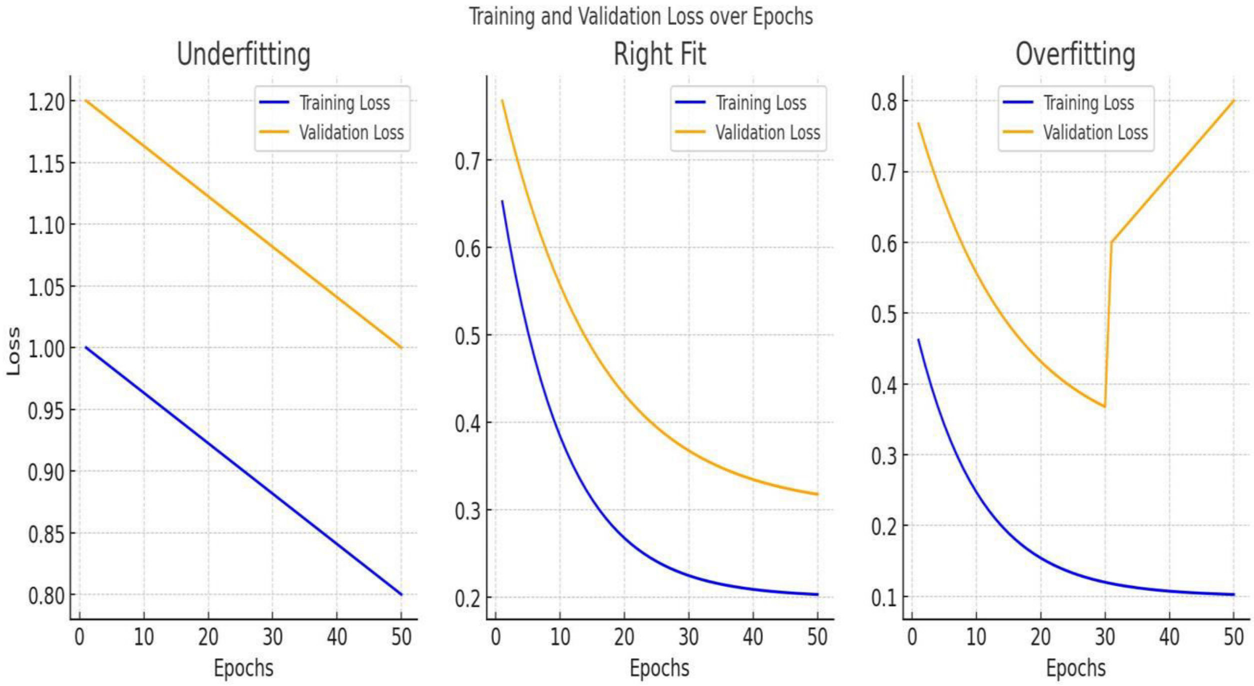
6.1 Performance Metrics

To evaluate the model, the following standard object detection metrics are used:

- **Mean Average Precision (mAP):** Measures overall detection accuracy across all classes
- **mAP@50:** Precision at Intersection over Union (IoU) threshold of 0.5
- **Classification Accuracy:** Correct prediction of damage categories
- **Localization Accuracy:** Accuracy of bounding box placement

The model achieves consistent performance across multiple epochs, demonstrating stable convergence and reliable detection capability.

6.2 Training Performance Analysis



ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

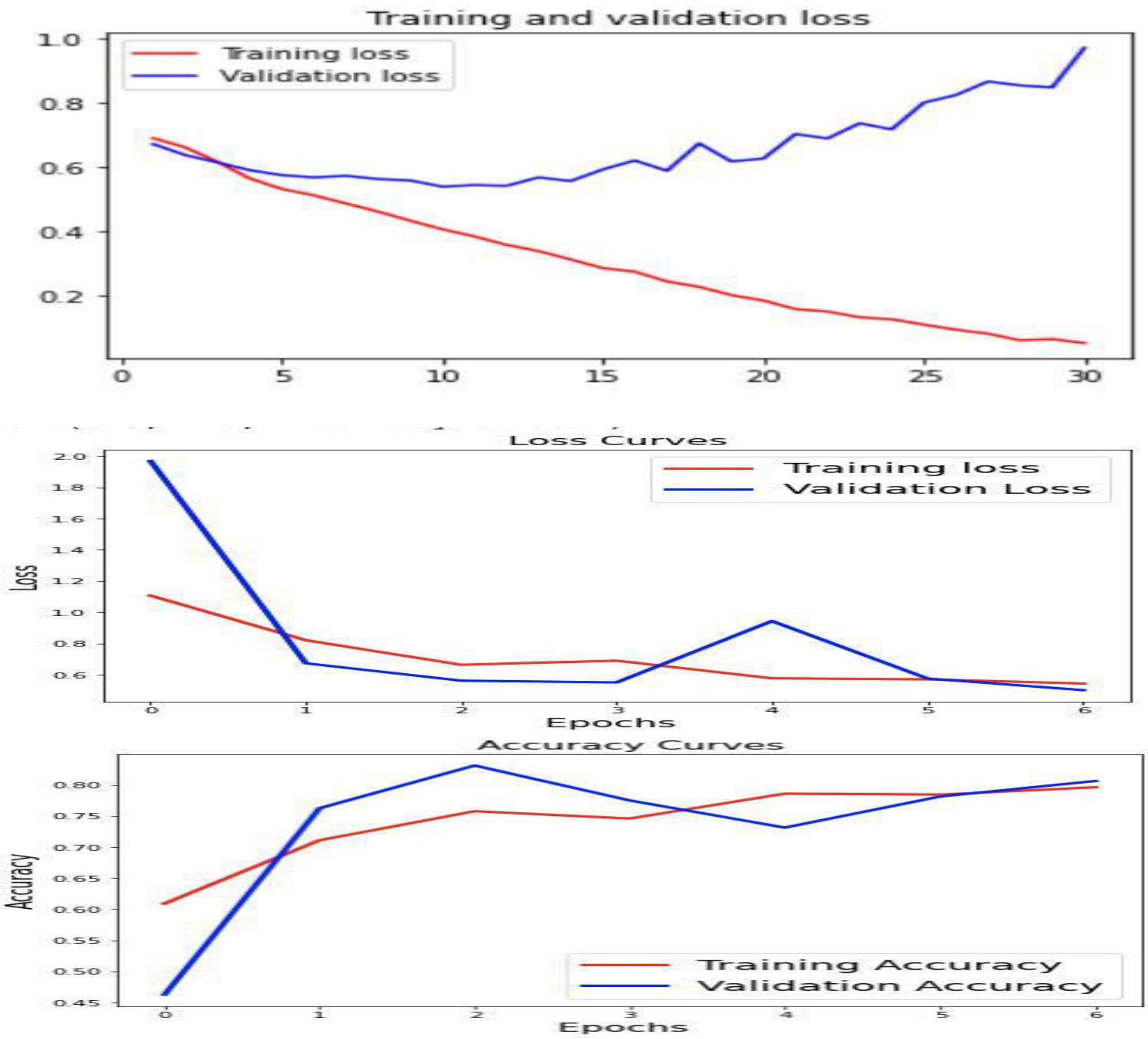


Figure 6: Training Loss and Convergence Graph

The training process shows a steady decrease in both classification loss and bounding box regression loss. The OneCycle learning rate policy helps accelerate convergence by initially increasing the learning rate and then gradually reducing it, leading to better generalization.

- Early epochs show rapid loss reduction
- Mid-training stabilizes learning
- Final epochs achieve minimal loss and convergence

This indicates that the model effectively learns the mapping between satellite imagery and damage categories.

6.3 Detection Results

ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

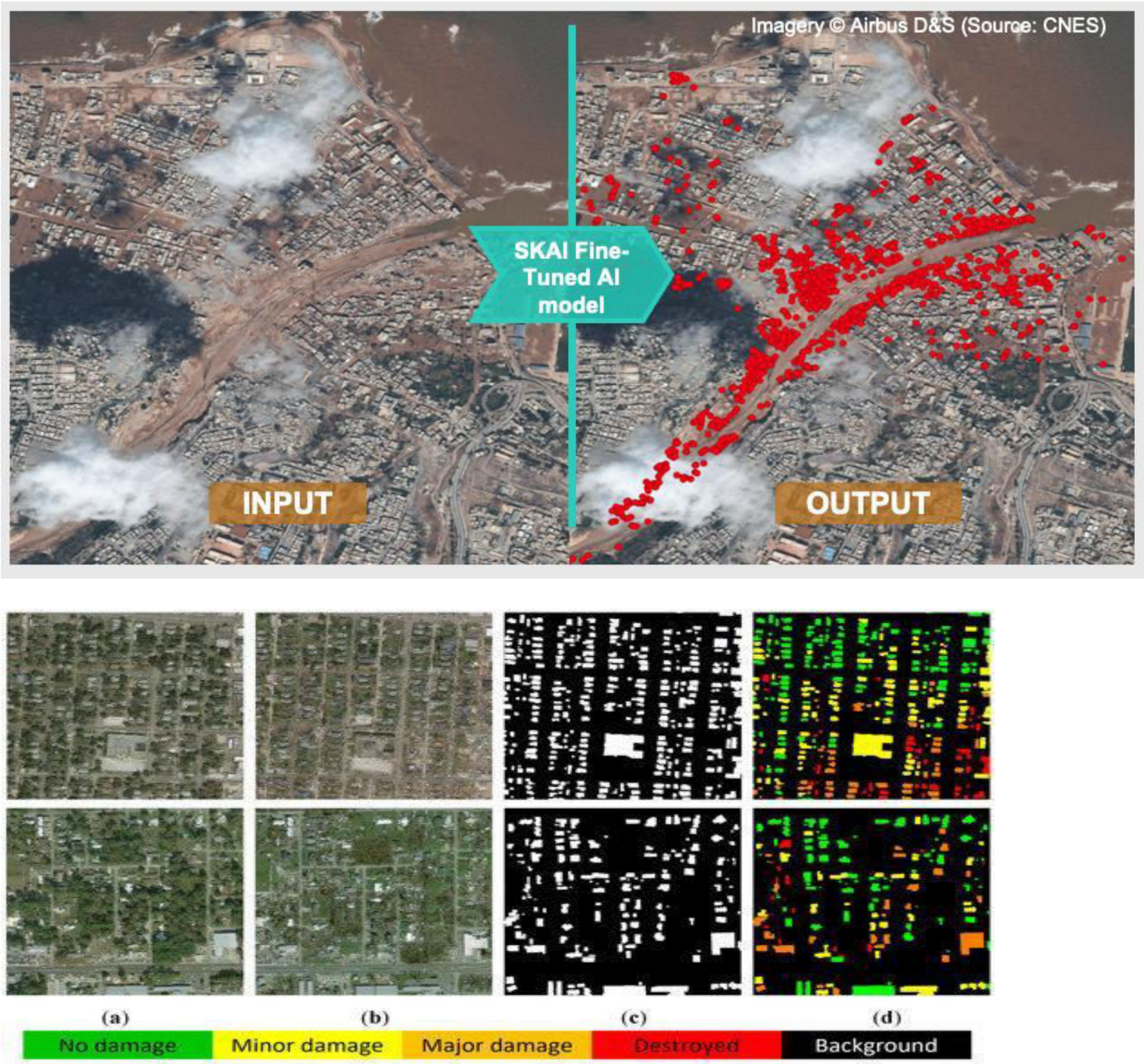


Figure 7: Disaster Damage Detection Output

The output images display bounding boxes around detected buildings with color-coded labels representing damage severity:

- No Damage
- Minor Damage
- Major Damage
- Destroyed

The model successfully identifies damaged structures and differentiates between varying levels of severity. The bounding box representation provides clear and actionable insights for disaster response teams.

6.4 Discussion

The experimental results demonstrate that DisasterVision AI effectively addresses the key challenges in satellite-based disaster assessment:

- **Fast Inference:** The single-stage SSD architecture enables rapid detection suitable for real-time scenarios

ISSN:2347-1697
International Journal of Informative & Futuristic Research (IJIFR)
Volume 13(8), April 2026 (Continuous 153rd Edition)

- **Improved Detection:** Six-channel input enhances the model's ability to capture temporal changes
- **Interpretability:** Bounding box outputs are easier to understand compared to segmentation maps
- **Scalability:** The system can process large-scale satellite imagery efficiently

However, certain limitations are observed:

- Small buildings may not always be detected accurately
- Bounding box approximation may slightly reduce localization precision
- Performance depends on dataset diversity and quality

Despite these limitations, the system demonstrates strong potential for real-world disaster management applications.

6.5 Comparative Insight

Table 2: Compared to traditional approache

Method	Accuracy	Speed	Interpretability
Semantic Segmentation	High	Medium	Low
Change Detection	Medium	High	Medium
Faster R-CNN	Very High	Low	High
DisasterVision AI (Proposed)	High	High	Very High

The proposed system achieves an optimal balance between accuracy, speed, and usability.

7. Conclusion

This paper presented **DisasterVision AI**, a deep learning-based framework for automated disaster damage assessment using satellite imagery. The system introduces a novel six-channel input approach combined with a modified SSD architecture to analyze pre-disaster and post-disaster image pairs.

The proposed method successfully transforms the damage assessment problem into an object detection task, providing bounding box outputs that are directly interpretable and actionable. Experimental results demonstrate that the system achieves strong performance in terms of accuracy, speed, and scalability.

The integration of advanced optimization techniques such as AdamW and OneCycle learning rate scheduling further enhances training efficiency and model generalization. The system's modular design allows for easy extension and adaptation to other disaster scenarios and datasets.

In conclusion, DisasterVision AI represents a significant step toward intelligent and automated disaster response systems, enabling faster decision-making and improved resource allocation during emergency situations.

8. References

[1] Liu, W., et al., "SSD: Single Shot MultiBox Detector," *European Conference on Computer Vision*, 2016.
[2] Redmon, J., et al., "You Only Look Once: Unified Real-Time Object Detection," *CVPR*, 2016.
[3] Ren, S., et al., "Faster R-CNN: Towards Real-Time Object Detection," *IEEE TPAMI*, 2017.
[4] Ronneberger, O., et al., "U-Net: Convolutional Networks for Biomedical Image Segmentation," *MICCAI*, 2015.
[5] Chen, L., et al., "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets," *IEEE TPAMI*, 2018.
[6] Gupta, R., et al., "xView2: A Dataset for Assessing Building Damage from Satellite Imagery," *CVPR Workshops*, 2019.
[7] Kingma, D., and Ba, J., "Adam: A Method for Stochastic Optimization," *ICLR*, 2015.
[8] Loshchilov, I., and Hutter, F., "Decoupled Weight Decay Regularization," *ICLR*, 2019.
[9] Paszke, A., et al., "PyTorch: An Imperative Style Deep Learning Library," *NeurIPS*, 2019.
[10] Buslaev, A., et al., "Albumentations: Fast and Flexible Image Augmentations," *Information*, 2020.
[11] Everingham, M., et al., "The Pascal Visual Object Classes Challenge," *IJCV*, 2010.
[12] Lin, T., et al., "Microsoft COCO: Common Objects in Context," *ECCV*, 2014.
[13] He, K., et al., "Mask R-CNN," *ICCV*, 2017.
[14] Dosovitskiy, A., et al., "An Image is Worth 16x16 Words: Vision Transformer," *ICLR*, 2021.
[15] Kirillov, A., et al., "Segment Anything," *Meta AI*, 2023.