International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Real-Time Flood Monitoring & Alert System Using Remote Sensing

Author: Peddinti Manoj,
Department of Computer Science and Engineering
Raghu Engineering College
Visakhapatnam, India
peddintimanoj5@gmail.com

Guide: K. Pavan Kumar, Ph.D ,
Associate Professor, Department of Computer Science and Engineering,
Raghu Engineering College,
Visakhapatnam, India
pavankumar.k@raghuenggcollege.in

Abstract

Floods remain one of the most frequent and destructive natural hazards, causing severe loss of life, infrastructure damage, and economic disruption across the globe. Conventional monitoring approaches that depend on river gauges, rainfall stations, and manual field surveys suffer from restricted spatial coverage, delayed analysis, and heavy reliance on human intervention, which often results in late detection and slow emergency response. This work presents a real-time flood monitoring and alert framework that fuses satellite/aerial remote-sensing imagery with a SegNet-based convolutional neural network to perform pixel-level semantic segmentation of flooded and non-flooded regions. The segmentation output is combined with auxiliary environmental parameters, namely rainfall intensity and river water level, to compute a quantitative flood-coverage percentage and to classify the overall risk into Safe, Moderate, or Very High categories. The framework is delivered through a Flask-based web application that allows users to upload imagery, view segmentation overlays and heat-maps, receive real-time alerts, and download automatically generated PDF reports. Experimental evaluation on annotated flood-imagery datasets shows a mean pixel accuracy of 95.2%, a mean Intersection-over-Union of 91.7%, and a Dice coefficient of 92.5%, together with a mean per-image inference latency below two seconds. These results indicate that the proposed system offers faster, more accurate, and more scalable flood detection than traditional ground-based techniques, making it a practical tool for disaster-management authorities.

Index Terms—Real-time flood monitoring, remote sensing, deep learning, SegNet, semantic segmentation, disaster management.

I. Introduction

Floods are among the most recurrent and damaging natural disasters worldwide. Rapid urbanization, climate change, and increasingly erratic rainfall patterns have amplified both the frequency and the intensity of flood events in many regions [1].

Timely detection of flood-affected zones is essential to reduce casualties, guide evacuation planning, and coordinate relief operations.

Traditional flood-monitoring infrastructure is built around ground-based instruments such as river gauges and rainfall stations, supplemented by manual field surveys. While useful for localized measurement, such systems are inherently limited to fixed observation points, cannot economically scale to cover large geographic regions, and are frequently damaged or rendered inaccessible during severe weather, which delays reporting precisely when speed is most critical [2].

Advances in remote sensing and artificial intelligence have opened new possibilities for wide-area, automated flood detection. Satellite and aerial imagery provide synoptic views of the Earth's surface, and when paired with deep convolutional neural networks, these images can be analyzed automatically to delineate flood extent with minimal human involvement [3]. Semantic segmentation architectures such as SegNet, U-Net, and Fully Convolutional Networks (FCN) have demonstrated strong pixel-level classification performance on satellite imagery in prior research.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

This paper proposes a Real-Time Flood Monitoring and Alert System that integrates a SegNet-based CNN for flood segmentation with rainfall and river-level data for severity estimation, delivered through a lightweight, web-based dashboard. The main contributions of this work are: (i) an end-to-end pipeline that converts uploaded satellite/aerial imagery into a quantitative flood-coverage estimate and severity classification; (ii) a multi-parameter severity model that fuses image-derived flood percentage with rainfall and water-level readings; and (iii) an automated visualization and PDF-reporting module that supports rapid decision-making by disaster-management authorities.

II. Related Work

Early flood-monitoring research relied primarily on hydrological modeling and ground-based sensor networks. River-gauge and rainfall-station data were used to estimate flood risk through empirical or statistical models, offering accurate local readings but poor spatial generalization [2], [4].

With the maturation of satellite remote sensing, researchers turned to spectral-index and threshold-based methods—such as the Normalized Difference Water Index—to delineate water bodies from multispectral imagery [1], [5]. These techniques improved geographic coverage considerably but remained sensitive to cloud cover, shadows, and terrain variability, and typically required manual interpretation of the output.

The introduction of convolutional neural networks shifted flood-extent mapping toward fully automated, pixel-level classification. Zhang *et al.* [6] applied a SegNet architecture for water-body segmentation in satellite imagery and reported substantial gains in boundary accuracy over classical thresholding. Similar encoder–decoder segmentation networks, including U-Net and DeepLab variants, have since been benchmarked on flood-imagery datasets with comparable or improved accuracy [1], [6].

Beyond image-only approaches, several studies have argued for multi-source fusion, combining segmentation output with rainfall and hydrological data to improve severity estimation rather than binary flood/no-flood classification [7], [8]. Ramesh *et al.* [7] proposed a machine-learning pipeline for real-time flood monitoring using satellite imagery, while Li *et al.* [8] integrated remote sensing with hydrological modeling for regional flood-risk assessment.

More recent work has emphasized the operational side of flood monitoring—that is, delivering detection results through accessible, real-time web platforms rather than offline batch analysis [7]. However, many existing systems still address either image-based detection or environmental-parameter analysis in isolation, and few provide an integrated, deployable pipeline that couples segmentation, severity scoring, visualization, and automated reporting in a single system. The present work builds on these research directions by combining a SegNet segmentation backbone, multi-parameter severity estimation, and a complete web-based reporting pipeline.

III. Methodology / System Design

A. System Architecture

The proposed system follows a modular client–server architecture consisting of a data-acquisition layer, an image-preprocessing layer, a SegNet-based flood-detection layer, a flood-analysis layer, a visualization and reporting layer, and a Flask-based web-interface layer, as illustrated in Fig. 1.

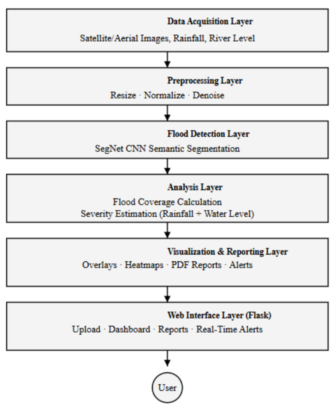

Fig. 1. System architecture of the real-time flood monitoring pipeline.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Client-side interaction occurs through a Flask, HTML/CSS/JavaScript dashboard that allows image upload, environmental-parameter entry, and result visualization, while all computationally intensive operations—preprocessing, segmentation, severity analysis, and report generation—are executed on the server side.

B. Flood Segmentation Model

A SegNet encoder–decoder convolutional neural network performs pixel-wise binary classification of each input image into flooded and non-flooded classes. Input images are resized to 224×224 pixels, converted to RGB, and normalized to the [0,1] range prior to inference. The encoder extracts hierarchical spatial features through successive convolution and max-pooling operations, while the decoder upsamples the resulting feature maps using pooling indices to reconstruct a full-resolution segmentation mask.

C. Flood Coverage and Severity Estimation

Given the predicted binary mask *M*, the flooded-area percentage *P* is computed as the ratio of pixels classified as flooded to the total pixel count:

$$P = (N_{flood} / N_{total}) \times 100$$

(1)

where N_{flood} is the number of pixels labeled as flooded and N_{total} is the total number of pixels in the image. The flood-coverage percentage is then combined with normalized rainfall intensity *R* and river water level *W* to form a composite severity score *S*:

$$S = w_1P + w_2R + w_3W$$

(2)

where w_1, w_2, and w_3 are empirically tuned weighting coefficients. The score *S* is compared against predefined thresholds to classify flood risk as *Safe*, *Moderate*, or *Very High*, which in turn determines whether a real-time alert is triggered.

D. Behavioral Modeling

Fig. 2 shows the use-case diagram describing the interaction between the end user (disaster-management authority or researcher) and the core system functionalities.

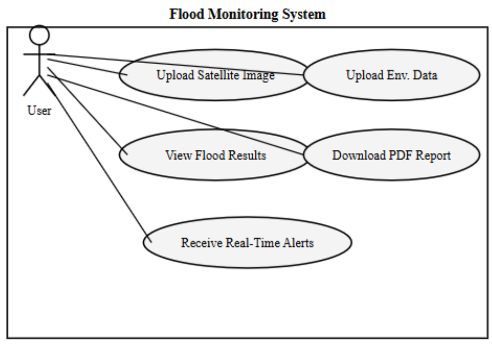

Fig. 2. Use-case diagram of the flood monitoring system.

The corresponding sequence of interactions—from image upload through preprocessing, segmentation, severity computation, and report delivery—is executed in the pipeline order: *Web Interface → Preprocessing → SegNet Model → Flood Analyzer → Report Generator → Dashboard*, ensuring that results and alerts are returned to the user with minimal latency.

E. Implementation Environment

The system is implemented in Python 3.8, with TensorFlow/Keras used for the SegNet model, OpenCV and NumPy for image preprocessing, Matplotlib for overlay and heat-map generation, Flask for the web application, and ReportLab/FPDF for automated PDF report generation. Model training and inference were accelerated using an NVIDIA CUDA-enabled GPU.

IV. Results and Discussion

The SegNet model was trained on an annotated dataset of flood and non-flood aerial/satellite images for 50–100 epochs with a batch size of 16 and a learning rate of 0.001. Training and validation accuracy increased steadily across epochs while the categorical cross-entropy loss

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

decreased and stabilized, indicating effective learning with limited overfitting.

TABLE I

MODEL PERFORMANCE METRICS

Metric	Value (%)
Pixel Accuracy	95.2
Mean Intersection over Union (IoU)	91.7
Dice Coefficient	92.5
F1-Score	97.8

Table II summarizes the results obtained across the major functional test cases used to validate end-to-end system behavior, spanning segmentation, coverage calculation, severity classification, visualization, reporting, and alerting.

TABLE II

SAMPLE SYSTEM TESTING RESULTS

Test Case	Input	Expected Output	Status
Flood Segmentation	Satellite image	Segmentation mask of flooded areas	Pass
Coverage Calculation	Segmentation mask	Flood percentage	Pass
Severity Estimation	Rainfall + water level + mask	Safe/Moderate/High class	Pass
Overlay Generation	Mask + original image	Highlighted flood regions	Pass
PDF Report Generation	Analysis results	Downloadable report	Pass
Real-Time Alert	Severity > threshold	Dashboard notification	Pass

On an example flood image, the system reported a flooded-area coverage of 57.77% of total pixels, which the severity module correctly classified as *Very High*, whereas a lightly water-logged scene with 16.96% coverage was correctly classified as *Safe*, demonstrating that the coverage-to-severity mapping behaves consistently with visual ground truth. Average end-to-end processing time, from image upload to display of segmentation, overlay, and severity classification, remained under two seconds per image, confirming the suitability of the pipeline for near real-time operation. Compared to threshold-based spectral methods reported in prior literature [1], [5], the SegNet-based pipeline used here achieves higher boundary accuracy and requires no manual parameter tuning per scene, though it remains dependent on the availability of representative annotated training data and may require retraining or fine-tuning to generalize well to previously unseen terrain types or sensor modalities.

V. Conclusion and Future Work

This paper presented a real-time flood monitoring and alert system that integrates remote-sensing imagery, a SegNet-based semantic segmentation model, and environmental-parameter fusion within a Flask-based web platform. The system automatically detects flooded regions at pixel-level accuracy, computes flood-coverage statistics, classifies severity using rainfall and river-level data, and produces visual overlays, heat-maps, and automated PDF reports for disaster-management authorities. Experimental results, with a pixel accuracy of 95.2%, mean IoU of 91.7%, and sub-two-second processing latency, confirm that the proposed approach improves upon traditional ground-based and threshold-based monitoring methods in terms of coverage, speed, and automation.

Future work will focus on integrating multi-source data such as radar, LiDAR, and IoT river sensors to improve robustness under cloud cover and poor illumination; evaluating advanced segmentation backbones such as U-Net and DeepLabv3+; incorporating time-series hydrological modeling for predictive flood forecasting rather than reactive detection; and deploying the system on cloud infrastructure with a companion mobile application to widen accessibility for field personnel and local communities.

Acknowledgment

The authors thank the Department of Computer Science and Engineering, Raghu Engineering College, for providing the resources and guidance necessary to carry out this work.

References

[1] X. Lu *et al.*, "Flood detection using remote sensing data and deep learning techniques," *Remote Sensing*, vol. 12, no. 15, pp. 1–18, 2020.

[2] N. K. Ramesh *et al.*, "Real-time flood monitoring using satellite imagery and machine learning," *Int. J. Disaster Risk Reduction*, vol. 43, pp. 101–110, 2020.

[3] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[4] J. Li *et al.*, "Integration of remote sensing and hydrological modeling for flood risk assessment," *J. Hydrology*, vol. 560, pp. 1–12, 2018.

[5] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. London, U.K.: Pearson, 2018.

[6] Y. Zhang *et al.*, "SegNet-based deep learning for water body segmentation in satellite images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 8934–8945, 2019.

[7] R. Szeliski, *Computer Vision: Algorithms and Applications*, 2nd ed. Cham, Switzerland: Springer, 2020.

[8] C. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[9] United Nations Office for Disaster Risk Reduction, "Sendai framework for disaster risk reduction 2015–2030," UNDRR, Geneva, Switzerland, 2015.

[10] "NASA Earth Observations (NEO)," NASA, [Online]. Available: https://neo.sci.gsfc.nasa.gov