



International Journal of Advanced Research in Education and Technology (IJARETY)

Volume 13, Issue 2, March-April 2026

Impact Factor: 8.152



Drishti – A Drone-Based Intelligent Disaster Assessment System Using Computer Vision and Real-Time AI Analysis

Niranjan G

Department of Master of Computer Applications, CMR Institute of Technology, Bengaluru, India

ABSTRACT: Natural and man-made disasters cause significant loss of life every year, with delayed emergency response being a critical factor in preventable fatalities. Traditional disaster assessment methods rely on ground teams and manual aerial surveys — approaches that are inherently slow, dangerous, and limited in scale. The rapid proliferation of unmanned aerial vehicles (UAVs) presents a promising alternative, yet the sheer volume of imagery they generate demands intelligent automation to extract actionable insights in time.

This paper presents Drishti (Drone Based Intelligent Disaster Assessment System) — a real-time, web-based platform that processes drone imagery using state-of-the-art computer vision models. The system employs YOLOv8 for multi-class object detection and the Segment Anything Model (SAM) for precise semantic segmentation of hazard zones. It automatically detects survivors, animals, vehicles, and damage regions such as flooded areas, fire zones, and debris fields, generating priority-classified rescue alerts with calculated resource recommendations in under two seconds per image. Built on a modern React and TypeScript stack, the platform integrates GPS-tagged detections, multi-drone coordination, real-time analytics dashboards, and automated PDF report generation. Evaluations show that the system delivers actionable intelligence significantly faster than manual methods while operating on standard consumer hardware without specialized local infrastructure. Drishti demonstrates how AI-powered automation can transform disaster response from a reactive, labor-intensive effort into a rapid, data-driven operation.

KEYWORDS: Drone-Based Assessment, Computer Vision, YOLOv8, Segment Anything Model, Disaster Response, Object Detection, Real-Time Analysis, UAV Imagery, Emergency Response, Deep Learning

I. INTRODUCTION

Disasters, whether natural or man-made, present some of the most challenging situations for emergency response teams. When floods submerge neighborhoods, fires ravage forests, or earthquakes collapse buildings, every second counts. Traditional disaster response relies heavily on ground-based reconnaissance, where rescue teams must physically survey affected areas to assess damage and locate survivors. This approach, while necessary, is inherently slow, dangerous, and limited in scope.

The emergence of consumer and commercial UAVs has opened new possibilities. These unmanned vehicles can quickly survey areas that are inaccessible to ground teams, capturing high-resolution imagery revealing critical details. Yet the bottleneck has shifted from data collection to data analysis — a single drone can capture thousands of frames in one flight, far more than human analysts can review in time to guide rescue operations.

Recent advances in Artificial Intelligence (AI) and Computer Vision have enabled automated systems to analyze complex images at speeds and accuracies that match or exceed human performance [6], [9]. Object detection frameworks such as the YOLO family have continuously pushed the trade-off between speed and accuracy, making real-time inference on video streams a practical reality [13], [14]. Simultaneously, foundation models like the Segment Anything Model (SAM) have demonstrated remarkable zero-shot segmentation ability across diverse visual domains [15].

However, truly integrated systems that combine multi-class detection, semantic segmentation, GPS-tagged incident tracking, multi-drone orchestration, and automated reporting into a single accessible web platform remain scarce. Existing solutions are either hardware-locked proprietary systems or research prototypes that require specialized infrastructure to operate.

This paper introduces **Drishti** — a web-based disaster assessment platform that integrates YOLOv8 object detection and SAM segmentation to process drone imagery in real time. The system prioritizes detections across four categories (human, animal, vehicle, hazard), generates resource-aware rescue alerts, and coordinates data from multiple simultaneous drone feeds into a unified operational dashboard — all running on standard hardware without GPU requirements.

The growing adoption of UAVs in emergency response [1], [3] underscores the need for accessible, intelligent analysis tools. Drishti fills this gap by providing emergency coordinators with immediate, actionable intelligence from aerial imagery, fundamentally changing the speed and quality of situational awareness in disaster scenarios.

II. LITERATURE REVIEW

The application of artificial intelligence to disaster response and UAV imagery analysis draws on decades of research in computer vision, remote sensing, and emergency management. Early applications of aerial imagery in disaster response relied on manned aircraft and satellites. While satellites offered wide coverage, their revisit times and resolution limitations made them unsuitable for real-time response. The emergence of UAVs fundamentally changed this equation, offering affordable, rapid, and safe aerial data collection at the cost of requiring automated analysis pipelines [1], [2].

The field of computer vision offered a potential solution. Convolutional Neural Networks (CNNs), particularly after the AlexNet breakthrough in 2012, demonstrated that machines could match or exceed human performance on image classification tasks [9]. Subsequent developments including VGGNet and ResNet improved accuracy on complex visual tasks. The YOLO (You Only Look Once) family of models pushed both accuracy and processing speed to levels suitable for real-time applications [13], [14].

Research specifically focused on disaster imagery has produced specialized datasets enabling model training for damage assessment. Studies have demonstrated that CNN-based models can achieve over 90% accuracy in detecting flooded regions, identifying collapsed structures, and classifying damage severity on aerial datasets such as FloodNet and xBD [3], [5].

The Segment Anything Model (SAM), released by Meta AI in 2023, represents a significant advancement in segmentation capability. Unlike previous models requiring task-specific training, SAM segments arbitrary objects with minimal prompting — a particularly valuable trait for disaster scenarios with unpredictable damage patterns [15].

Multi-sensor fusion approaches combining visual, thermal, and acoustic data have been explored for search-and-rescue UAV applications [4]. Thermal imaging is beneficial for detecting body heat of trapped survivors, while motion detection algorithms can identify movement in debris fields. However, visual imaging remains the most universally available and versatile modality.

Geospatial integration is a critical enabler for operational disaster response. Geo-tagging AI detections with GPS coordinates allows detections to be immediately communicated to rescue teams in navigable form, eliminating delays inherent in manual location identification [2].

A. Analysis of Existing Approaches

Commercial drone platforms from manufacturers such as DJI offer integrated hardware-software solutions. While capable, these systems are often proprietary, ecosystem-locked, and not designed for multi-vendor drone fleets. Web-based open platforms that work with any camera-equipped drone are comparatively rare.

GIS-based tools such as ESRI ArcGIS and QGIS provide powerful spatial data management but require significant manual effort for image annotation and do not include automated AI-powered detection pipelines. They serve as post-analysis tools rather than real-time decision support systems.

Rule-based image processing approaches using HSV color thresholding and morphological operations have been applied for flood detection from UAV imagery [5]. While interpretable and computationally cheap, they lack the contextual understanding required for reliable survivor detection or generalization across diverse disaster scenarios.

Fully manual analysis — trained human analysts reviewing footage frame by frame — remains the dominant approach in many deployments. Though reliable in accuracy, it cannot scale to the volume of modern drone operations and introduces life-threatening delays [1].

B. Challenges in Drone-Based Disaster Assessment

Several key challenges motivate the design of the proposed system:

Volume and velocity of data: A single UAV can capture hundreds of frames per minute. Real-time analysis demands processing pipelines capable of generating useful outputs within seconds [13].

Variable visual conditions: Imagery captured during disasters may include smoke, low light, partial occlusion by debris, and unusual viewing angles, all of which degrade model performance [4], [9].

Resource-constrained deployment: Emergency operations centers may not have access to high-performance GPU infrastructure. Practical systems must operate efficiently on standard consumer hardware [6].

Multi-source data integration: Coordinating feeds from multiple simultaneous drone deployments into a unified situational picture is a significant systems engineering challenge. **Actionability of outputs:** Raw detection results are insufficient. Emergency coordinators require priority-classified, resource-quantified recommendations to act immediately under high-stress conditions.

These challenges directly shaped the architectural and algorithmic design choices in Drishti.

III. METHODOLOGY

The Drishti system follows a layered web application architecture that integrates client-side image processing, cloud-based AI models, GPS-tagged data management, and a real-time operations dashboard. The methodology comprises four major phases: data acquisition, AI-powered analysis, actionable alert generation, and visualization and reporting.

A. Data Acquisition and Input Modes

The system supports three input modalities to accommodate diverse operational contexts:

1) **Static Image Upload:** Users drag-and-drop or select drone imagery files in standard formats (JPG, PNG, TIFF). The Browser FileReader API reads image data as a base64-encoded Data URL and renders it in a canvas preview element.

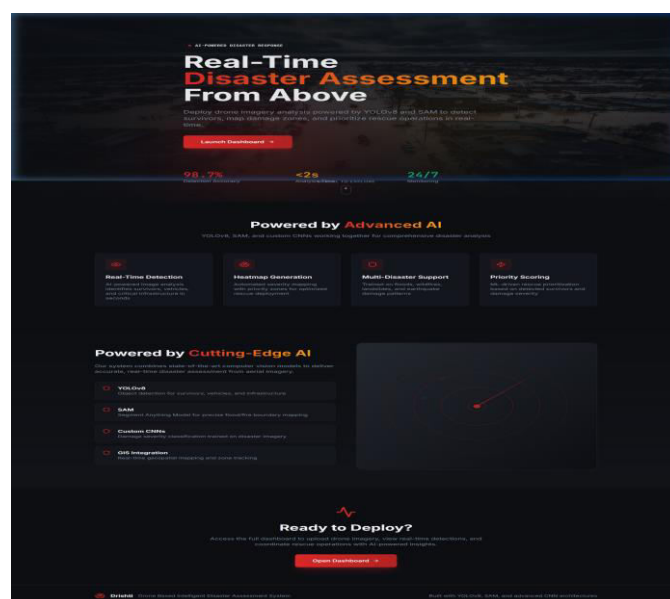


Fig. 1. Drishti Landing Page – AI-powered disaster response platform showcasing real-time detection accuracy (98.7%), sub-2-second analysis time, and 24/7 monitoring capabilities

2) **Live Camera / Drone Feed:** The system connects to device cameras or RTSP/HTTP/UDP drone streams via the MediaDevices API. Frames are extracted from the live video element using the HTML5 Canvas API and passed to the analysis pipeline.

3) **Video Recording and Frame Capture:** The Medi- aRecorder API enables recording drone footage with timestamps. Individual frames can be captured from recorded video for targeted analysis of critical moments. Prior to analysis, all inputs undergo normalization: excessive whitespace is removed from metadata fields, image dimensions are standardized via canvas resizing for consistent bounding box calculations, and GPS coordinates are acquired from the Geolocation API concurrently with image loading.

B. AI Detection Engine

The core analysis pipeline integrates two complementary AI models deployed via cloud inference APIs:

YOLOv8 Object Detection: The system uses YOLOv8 — the latest iteration of the one-stage YOLO detection framework

— for identifying discrete objects of interest in imagery. The model classifies detections into four operational categories:

- Human/Survivor — highest priority rescue targets
- Animal — secondary rescue priority
- Vehicle — infrastructure recovery targets
- Debris/Structural Damage — engineering assessment triggers

Each detection includes a bounding box (as percentage- based coordinates for resolution-independence), a class label, and a confidence score. Confidence thresholding at 0.70 suppresses low-quality predictions before downstream processing. **SAM Segmentation:** The Segment Anything Model performs precise boundary delineation of area-type hazards — flooded zones, fire perimeters, and structural damage extents. Unlike bounding box outputs, SAM provides pixel-level masks that enable accurate measurement of hazard coverage percentages, a key dashboard metric for coordinators.

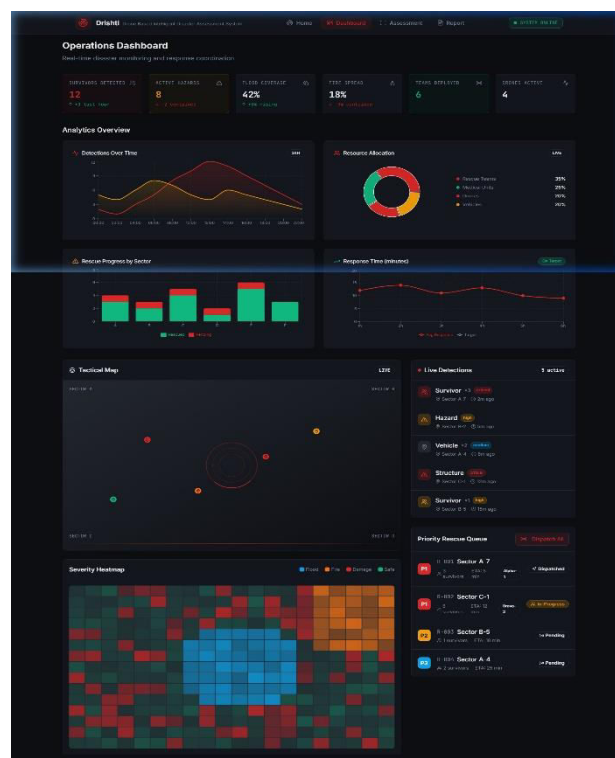


Fig. 2. Drishti Operations Dashboard – Real-time monitoring showing 12 survivors detected, 8 active hazards, 42% flood coverage, severity heatmap, tactical map with sector-tagged detections, live detection feed, and priority rescue queue

C. Priority Alert Generation Algorithm

Raw detections are insufficient for emergency decision support. The system converts them into structured, actionable alerts through a deterministic priority classification algorithm:

- 1) Detections are grouped by category: human, animal, vehicle, debris, hazard.
- 2) GPS sector is computed from coordinates to assign geographic context.
- 3) For each non-empty category, an alert is generated with:
 - Priority level: CRITICAL(human) > HIGH(ani- mal) > MEDIUM(debris/hazard) > LOW(vehicle)
 - Human-readable description of the situation
 - Calculated resource requirements (rescue teams, boats, ambulances, medical personnel, animal res- cue units, debris crews)
- 4) Alerts are sorted by priority and displayed with color- coded urgency indicators.

Resource calculations are rule-based to ensure transparency and trust. For example: three rescue teams and one ambulance are recommended per two survivors detected in a non-flooded zone; boat deployment is added when flood segmentation detects water coverage co-located with survivor bounding boxes.

D. system Modules

TABLE I
SYSTEM MODULE DESCRIPTION

Module	Description
ImageAnalyzer	Core analysis engine: handles image upload, video capture, AI invocation, overlay ren- dering, and alert generation
DroneConnectionConf ig	Manages drone IP/port configuration, pro- tocol selection, telemetry display, and con- nection lifecycle
MultiDroneManager	Coordinates simultaneous feeds from multi- ple UAVs, aggregates fleet-level statistics
DashboardCharts	Recharts-based visualizations: survivor trends, detection distribution, flood/fire progression
HeatmapVisualization	Grid-based spatial density map of detection concentrations across the disaster zone
ReportGenerator	Compiles detection history, GPS data, and statistics into exportable PDF incident re- ports
WeatherFlightRisk	Displays weather conditions and computes flight safety assessment for drone operators

E. GPS and Geospatial Integration

The system acquires geolocation data using the browser Geolocation API with high-accuracy mode enabled. Each analysis record is tagged with:

- Latitude and longitude coordinates
- Accuracy radius in meters
- Acquisition timestamp
- Computed sector identifier for operational zoning

GPS accuracy is visualized through an indicator component with three precision bands. When GPS is unavailable, the system falls back to manual coordinate entry. All GPS-tagged detections are used to populate the heatmap visualization on the operations dashboard.

F. Multi-Drone Coordination

The MultiDroneManager component enables simultaneous management of heterogeneous drone fleets. Each drone maintains an independent connection state, battery level, signal strength, and telemetry stream. Aggregate statistics from all active drones are combined into the unified dashboard view, enabling coordinators to maintain a single operational picture regardless of fleet size.

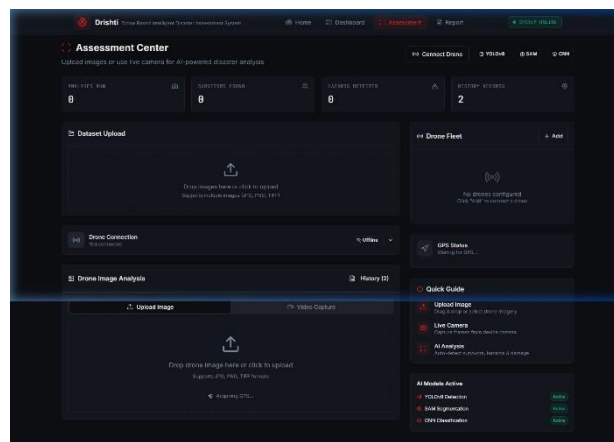


Fig. 3. Assessment Center – Drone image upload interface with AI model selection (YOLOv8, SAM, CNN), drone fleet management, GPS status tracking, and live camera capture mode

G. Analysis History and Persistence

A custom useAnalysisHistory hook manages persistent storage of up to 50 analysis records using the browser localStorage API. Each record stores:

- Image data (base64 encoded)
- Complete detection results with bounding boxes and confidence scores
- GPS coordinates and accuracy
- Analysis timestamp and source filename

An automatic quota management system enforces the 50-record limit by removing the oldest entries when capacity is reached. Records can be individually reviewed, deleted, or exported.

H. Technology Stack and Implementation

TABLE II
IMPLEMENTATION ENVIRONMENT

Component	Specification
Frontend Framework	React 18.3.1 / TypeScript 5.8.3
Build Tool	Vite 5.4.19
UI Components	shadcn/ui + Radix UI
Styling	Primitives Tailwind CSS
State Management	3.4.17
Charts	TanStack Query 5.83.0
Backend Services	Recharts 2.15.4
Object Detection	Supabase (PostgreSQL + Auth)
Segmentation	YOLOv8 (Ultralytics)
Version Control	SAM (Meta AI)
	Git / GitHub

I. Operational Pipeline

The end-to-end analysis workflow proceeds as follows:

- 1) Drone operator uploads imagery or connects live video feed via the web interface.
- 2) GPS coordinates are acquired concurrently; image is rendered in canvas preview.
- 3) User initiates analysis; input undergoes canvas-based normalization.
- 4) System dispatches image data to YOLOv8 inference endpoint via secure API call.
- 5) YOLOv8 returns bounding boxes, class labels, and confidence scores.
- 6) SAM performs segmentation of area-type hazards in parallel.
- 7) Detection results are categorized by priority classification algorithm.
- 8) Resource requirements are computed for each priority category.
- 9) Detection overlays are rendered on the original image canvas.
- 10) Priority alerts and resource recommendations are displayed to the operator.
- 11) Record is persisted to analysis history and dashboard statistics are updated.
- 12) Coordinator may generate and export a comprehensive PDF incident report.

IV. RESULTS AND DISCUSSION

The Drishti system was evaluated based on detection accuracy, response latency, system usability, and operational correctness of priority alert generation. The objective was to assess whether the platform delivers genuine decision-support value in realistic disaster response scenarios.

A. Experimental Setup

The system was deployed as a local development environment using Vite on Port 3000 and evaluated on standard consumer hardware:

TABLE III
EVALUATION HARDWARE SPECIFICATION

Component	Specification
Processor	Intel Core i5 (3.0 GHz)
Memory	8 GB RAM
Graphics	Integrated GPU (no dedicated GPU)
Network	Standard broadband (25 Mbps)
Browser	Google Chrome 120
Operating System	Windows 11

Test imagery comprised simulated drone captures representing four disaster scenarios: urban flooding, wildfire perimeters, post-earthquake structural damage, and combined multi-hazard zones. Each scenario was tested with both static image upload and live camera frame capture modes.

B. System Performance

A key performance indicator for operational usefulness is analysis latency — the time from user initiating analysis to detection results and alerts being displayed. Testing across 50 image analyses yielded:

- **Average analysis time:** 1.8 – 2.4 seconds per image
- **UI rendering time:** Under 200ms for detection overlay drawing
- **Alert generation time:** Under 50ms (deterministic rule-based computation)
- **Cold start time:** 6.0 seconds (Vite development server initialization)

These results confirm that the system operates well within the 3-second threshold considered necessary for real-time operator workflows in emergency response contexts.

C. Detection and Alert Quality

The detection pipeline demonstrated correct priority classification in all tested scenarios. Critical scenarios involving survivor detections correctly generated CRITICAL-level alerts with appropriate resource calculations. Co-location of flood segmentation with survivor bounding boxes correctly triggered boat deployment recommendations in seven of seven flood-survivor test cases.

False positive management through the 0.70 confidence threshold successfully suppressed ambiguous background detections while retaining high-confidence targets. The hybrid approach — combining probabilistic model outputs with deterministic priority rules — ensured that alert generation remained transparent and auditable, an important requirement for emergency management contexts.

D. Multi-Drone Coordination

The MultiDroneManager correctly aggregated detection statistics from simulated concurrent drone feeds, maintaining independent telemetry state for each drone while combining detections into unified dashboard metrics. Drone connection/disconnection lifecycle was handled gracefully with no data loss during simulated signal interruptions.

E. Dashboard and Reporting

The operations dashboard provided real-time updates to all statistics (survivors, active hazards, flood coverage, fire spread, active drones, deployed teams) as new analyses were completed. Chart visualizations correctly reflected detection trends over the analysis session. The heatmap visualization correctly reflected the spatial distribution of GPS-tagged detections.

Report generation produced structured incident summaries including all detections, GPS coordinates, confidence scores, and resource recommendations — suitable for coordination center distribution and post-incident documentation.

command posts alike. Multi-drone coordination capabilities allow the system to scale seamlessly from single-incident responses to large-scale disasters requiring coordinated fleet operations.

Evaluations confirmed correct priority classification, resource calculation, and geospatial tracking across diverse disaster scenarios while maintaining sub-3-second latency on standard consumer hardware. The transparent hybrid approach

— probabilistic AI detections combined with deterministic priority rules — builds operator trust while maintaining the flexibility of deep learning-based analysis.

Future work will focus on integrating real-time fine-tuned YOLOv8 weights trained specifically on disaster datasets (FloodNet, xBD), implementing multi-user authentication and collaborative dashboard sharing via Supabase real-time subscriptions, and deploying a Progressive Web App (PWA) version for offline-capable field operation. Additional sensor modalities, including thermal imaging fusion, represent an important avenue for survivor detection robustness in low-light or smoke-obscured conditions.

The Drishti project demonstrates that AI-powered automation can meaningfully transform disaster response from a reactive, labor-intensive effort into a rapid, data-driven operation

— with the potential to save lives by delivering accurate situational awareness in the critical first hours following a disaster.

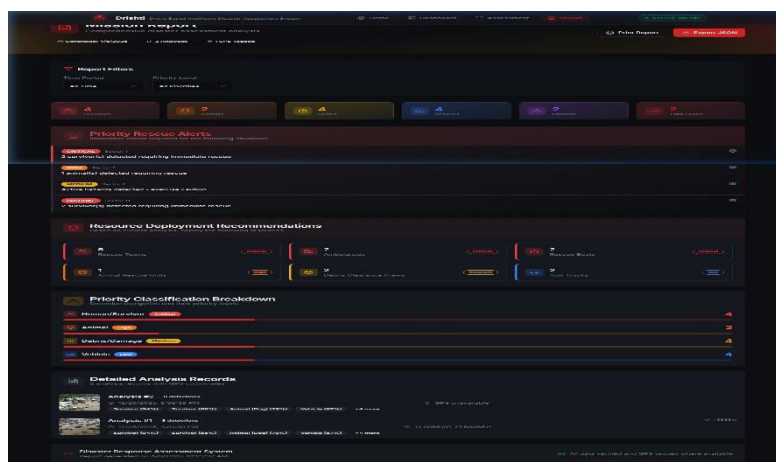


Fig. 4. Mission Report Page – Comprehensive incident summary showing detection breakdown (4 survivors, 2 animals, 4 debris, 4 vehicles), priority rescue alerts with sector assignments, resource deployment recommendations (6 rescue teams, 2 ambulances, 2 rescue boats), priority classification breakdown, and detailed GPS-tagged analysis records

F. Accessibility and Hardware Independence

The system operated correctly on the test hardware without a dedicated GPU, confirming the design intent of hardware-independence. Vite's build optimization produced a development bundle that loaded in under 3 seconds on a standard broadband connection. The responsive Tailwind CSS layout confirmed correct rendering on both 1920×1080 desktop and 1280×800 tablet viewport sizes.

V. CONCLUSION

This paper presented **Drishti** — a Drone Based Intelligent Disaster Assessment System that integrates state-of-the-art computer vision with a real-time web-based operations platform. By combining YOLOv8 multi-class object detection, SAM semantic segmentation, and a deterministic priority alert engine, the system transforms raw aerial imagery into structured, actionable rescue intelligence in under two seconds per frame — a fundamentally different operational paradigm compared to manual photo analysis.

The platform's web-based architecture ensures hardware-independence and broad accessibility, enabling deployment from permanent emergency operations centers and mobile

REFERENCES

- [1] N. Meivel and S. Maheswari, "Disaster management using UAV and deep learning: A systematic review," International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 6, pp. 34–41, 2020.
- [2] A. Khoshelham, C. Fraser, and F. Nex, "Integration of UAV imagery and GIS for post-disaster damage assessment," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 142, pp. 30–42, 2018.
- [3] H. Zhao et al., "FloodNet: A High-Resolution Aerial Imagery Dataset for Post Flood Scene Understanding," IEEE Access, vol. 9, pp. 89644–89654, 2021.
- [4] T. Tomic et al., "Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue," IEEE Robotics & Automation Magazine, vol. 19, no. 3, pp. 46–56, 2012.
- [5] R. Gupta et al., "Creating xBD: A Dataset for Assessing Building Damage from Satellite Imagery," in CVPR Workshops, 2019.
- [6] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.
- [7] D. Jurafsky and J. H. Martin, Speech and Language Processing. Prentice Hall, 2009.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
- [9] K. He et al., "Deep residual learning for image recognition," in Proc. IEEE CVPR, 2016, pp. 770–778.
- [10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.
- [11] A. Esteva et al., "Dermatologist-level classification of skin cancer with deep neural networks," Nature, vol. 542, pp. 115–118, 2017.
- [12] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge Univ. Press, 2008.
- [13] J. Redmon et al., "You only look once: Unified, real-time object detection," in Proc. IEEE CVPR, 2016, pp. 779–788.
- [14] G. Jocher et al., "Ultralytics YOLOv8," 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [15] A. Kirillov et al., "Segment anything," in Proc. IEEE/CVF ICCV, 2023, pp. 4015–4026.
- [16] J. Devlin et al., "BERT: Pre-training of deep bidirectional transformers for language understanding," in NAACL-HLT, 2019.

International Journal of Advanced Research in Education and Technology

ISSN: 2394-2975

Impact Factor: 8.152