Journal of Computer and Electrical Sciences (JCES)- Volume-1 Issue-4, April 2026
ISSN: 3049-2602

Disaster Analysis and Classification Using AI

S. Nikesh[1] , Mr. N. Naga Vijaya Varma [2]

1 – IV B. Tech Department of Computer Science Engineering, Andhra Loyola Institute of Engineering and Technology, Vijayawada.

2 – Assistant Professor, Department of Computer Science Engineering, Andhra Loyola Institute of Engineering and Technology, Vijayawada.

ABSTRACT: Natural disasters such as cyclones, earthquakes, floods, and wildfires cause significant damage to human life and infrastructure. Traditional manual analysis of disaster-related images is time-consuming and inefficient during emergencies. This project presents an **AI-based Natural Disaster Image Classification System** that automatically identifies disaster types using deep learning. The system utilizes the **MobileNetV2** architecture with **transfer learning** to classify images into four categories: cyclones, earthquakes, floods, and wildfires. Developed using **TensorFlow, Keras, and Streamlit**, the system provides real-time predictions with confidence scores and offers safety precautions to enhance public awareness..

I. INTRODUCTION

Natural disasters are catastrophic events resulting from extreme weather or geological activities that lead to severe societal and economic consequences. While the growth of digital platforms has led to a surge in disaster-related visual data, manually processing these images to understand the severity of a situation is difficult. The goal of this project is to build an automated tool using Convolutional Neural Networks (CNNs) to identify and interpret visual patterns in disaster scenes. By utilizing MobileNetV2, a lightweight and efficient model, the system can assist authorities in rapid data analysis and response strategy improvement.

II. LITERATURE REVIEW

Recent research has moved from traditional image processing toward deep learning to handle the complexity of real-world disaster images.

- **Earlier Methods**: Relied on handcrafted features like edge detection and color segmentation, classified via **SVM** or **KNN**, which often failed to capture complex environmental variations.

- **Deep Learning Models**: Architectures like **AlexNet, VGGNet, and ResNet** significantly improved accuracy but often required high computational resources.

- **Proposed System Differentiation**: Unlike resource-heavy models, this project adopts **MobileNetV2**, which uses depthwise separable convolutions to maintain high performance with lower computational requirements, making it suitable for real-time deployment.

https://doi.org/10.5281/zenodo.19606246

III. PROPOSED SYSTEM

The system is designed as an end-to-end pipeline that transforms raw image data into actionable disaster intelligence.

- A. Data Acquisition Layer: A dataset of images representing cyclones, earthquakes, floods, and wildfires is collected from public research repositories.
- B. Image Preprocessing Module: Images are resized to a fixed dimension and normalized to ensure training stability and consistency.
- C. AI Processing Layer (MobileNetV2): The system utilizes **transfer learning** on a pretrained MobileNetV2 model. It extracts hierarchical features such as shapes, textures, and color distributions to distinguish disaster categories.
- D. Visualization Layer (Streamlit): An interactive web dashboard allows users to upload images and instantly view the predicted disaster type, confidence levels, and educational facts.

IV. METHODOLOGY

The methodology integrates deep learning and web technologies through a structured process:

1. **Dataset Preparation**: Organizing images into training and testing subsets.
2. **Data Augmentation**: Applying rotation, flipping, zooming, and brightness adjustments to increase model robustness and prevent overfitting.
3. **Model Training**: Using **TensorFlow and Keras** to train the MobileNetV2-based classifier for 30 epochs.
4. **Inference**: Processing user-uploaded images via the Streamlit backend.
5. **Output Generation**: Displaying the classification result along with specific **safety precautions** (e.g., "Stay indoors" for cyclones).

V. Proposed System Hardware And Software Results

I. User Interface & Upload

The system features a clean, web-based interface built with **Streamlit**. Users are greeted with an upload prompt for disaster images in JPG, JPEG, or PNG formats.

https://doi.org/10.5281/zenodo.19606246

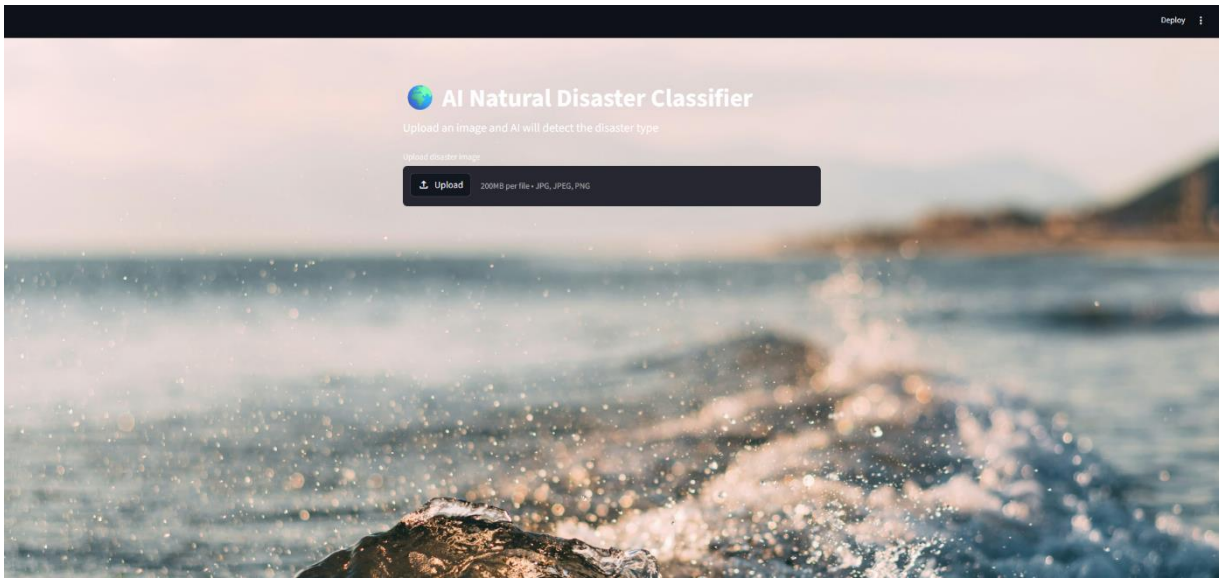


Fig. 1: User Authentication Interface

II. Prediction & Analysis

Once an image is uploaded, the model analyzes the visual features and displays the result card:

- **Disaster Type**: Clearly labels the detected disaster (e.g., FLOOD, WILDFIRE).
- **Confidence Score**: Shows the model's certainty percentage.
- **Interactive Content**: Displays "Did you know?" facts during analysis to keep the user engaged

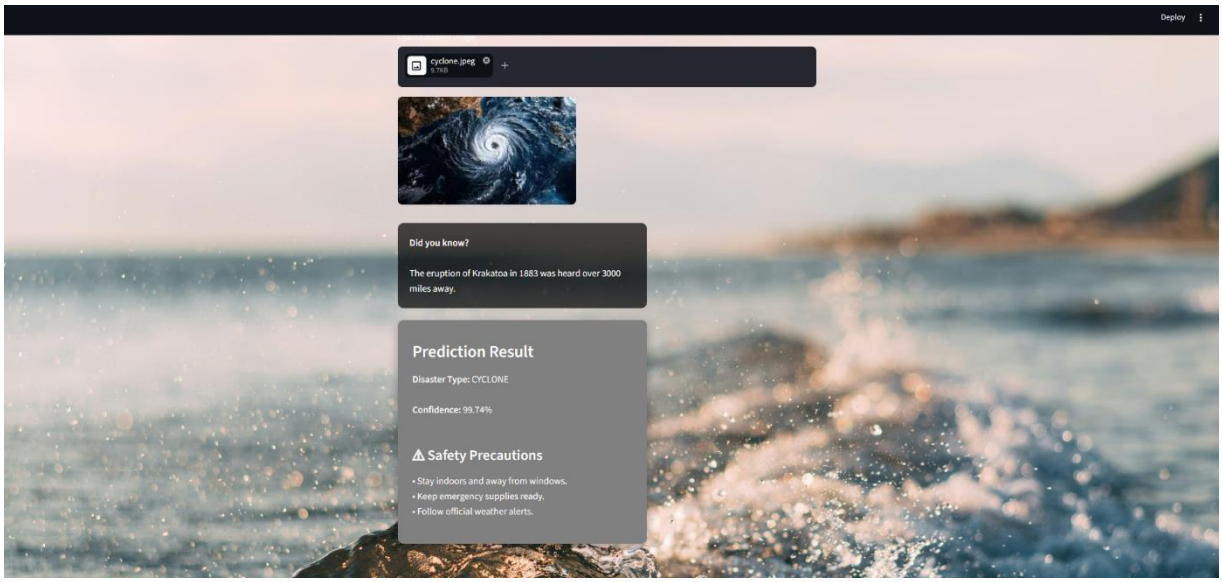


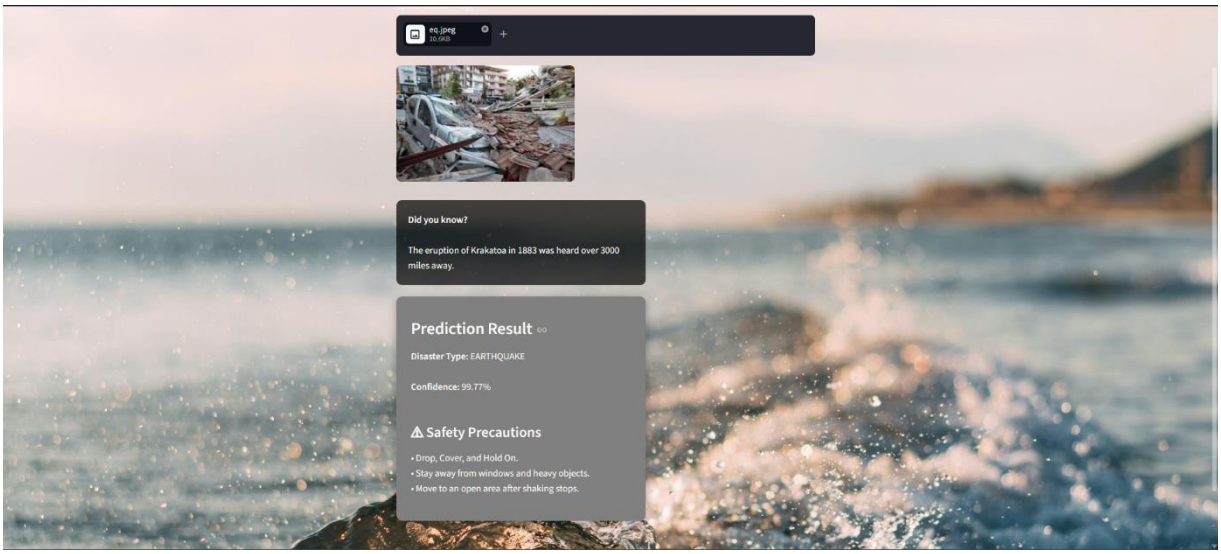
Fig. 2: Real-Time Prediction

https://doi.org/10.5281/zenodo.19606246

III. Safety Information Output

Based on the classification, the system dynamically provides tailored **Safety Precautions**:

- **Earthquake**: "Drop, Cover, and Hold On".
- **Flood**: "Move to higher ground immediately".
- **Wildfire**: "Evacuate immediately if advised".



VI. CONCLUSION

The **AI-Based Natural Disaster Image Classification System** successfully automates the identification of catastrophic events using **MobileNetV2** and transfer learning. By integrating a high-accuracy deep learning model with an accessible **Streamlit** web interface, the system provides a reliable tool for real-time disaster analysis. This solution not only assists management authorities in quick decision-making but also serves as an educational platform to improve public preparedness for natural disasters.

VII. Reference

1. Chollet, F., *Deep Learning with Python*, Manning Publications, 2017.
2. Sandler, M., et al., "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
3. Krizhevsky, A., Sutskever, I., & Hinton, G., "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems*, 2012.

https://doi.org/10.5281/zenodo.19606246

4. Chollet, F., "Keras: Deep Learning Library for Python," *GitHub Repository*, 2015.
5. TensorFlow Documentation, Google AI, https://www.tensorflow.org
6. Streamlit Documentation, https://streamlit.io
7. OpenCV Documentation, https://opencv.org

https://doi.org/10.5281/zenodo.19606246