An Approach for Disaster Victim Detection Using ML

PROJECT TEAM MEMBERS:

21ME1A0597 - Md. Abubakar Siddque
21ME1A05B5 - S. Sai Ganesh
21ME1A05A0 - N. Ganesh
21ME1A0597 - D. Sai Siddarth

Mentor: Dr. K SWETHA SASTRY
Dept of CSE, Professor

Department of Computer Science and Engineering

RAMACHANDRA
COLLEGE OF ENGINEERING
AUTONOMOUS

Approved by **AICTE**, New Delhi
Permanently Affiliated to **JNTUK**
Recognized by **UGC 2(f) & 12(B)**
Accreditations
NAAC A+
NBA (EEE, Civil, ME, ECE & CSE)
ISO 9001 : 2015 Certified

CONTENTS:

- Introduction & Background
- Problem Statement & Objectives
- Literature Survey
- Existing / Traditional system
- Methodology / Proposed System
- Hardware & Software Requirements
- Design & Implementation
- Results & Observations
- Advantages & Applications
- Conclusion
- References

Introduction & Background

Disasters like earthquakes, floods, building collapse and wildfires often leave many people injured or trapped. Finding victims quickly is critical for saving lives, but traditional search-and-rescue efforts are slow and rely heavily on human effort, which can be inefficient in large-scale disasters. To solve this problem, we developed a machine learning-based system that automatically detects disaster victims in images. Our approach uses ResNet50, a powerful deep learning model, to analyse and extract important features from images. These features are then processed by a random forest classifier, which has been trained on a diverse dataset of images depicting various disaster scenarios. By leveraging this advanced technology, our system significantly accelerates the identification of individuals in distress, allowing rescuers to allocate resources more effectively. Random forest classifier, a machine learning technique that makes final predictions based on patterns in the data. To make the system accessible and easy to use, we deployed it as a Flask API, which means users can upload images through a web-based interface, and the system will quickly analyse and return results.

Key words:
Disaster response, victim detection, machine learning,image segmentation, Object detection, Random Forest,Resnet50.

Problem Statement & Objectives

During disasters, identifying victims quickly is a major challenge. Manually checking images and videos from disaster sites is slow and not always accurate. There is a need for an automated system that can analyze images and help detect whether a person is a victim or not, to support faster and more effective rescue operations.

Objectives:

- **To develop a machine learning model** capable of classifying images into "victim" and "non-victim" categories with high accuracy.
- **To leverage deep learning techniques**, particularly ResNet50, for efficient feature extraction from complex disaster scene images.
- **To design a user-friendly Flask web interface** that allows users to upload images and get real-time predictions
- **To enhance disaster response operations** by providing a tool that can assist in rapid victim identification

Literature Survey

Sl. NO	Reference	Objective	Method	Key Findings	Relevance
1	G Seeja, ASA Doss (2023) **A Novel Approach for Disaster Victim Detection Under Debris Environments**	To propose a new integration approach for human victim detection under debris environments.	Uses deep learning image recognisation features.	The approach improves the accuracy of detecting human victims under debris using machine learning classifiers.	Applicable for enhancing disaster victim detection under challenging debris conditions.
2	S Mahmud, JH Kim (2024) **A System for Enhancing Victim Detection in Robot-assisted Disaster Scenarios**	To develop a victim detection system integrated with robot assistance	deep learning (DL) with robotic systems.	detection in disaster scenarios using advanced robotics	integration of DL with robotics for disaster management.
3	Wong,Tham, BH Kwan, (2022) **An Optimized Multi-Task Learning Model for Disaster Classification and Victim Detection in Federated Learning Environments**	To enhance disaster classification and victim detection using multi-task learning (MTL)	Employs MTL and FL to analyze disaster-related data	Multi-task learning improves disaster classification and victim detection	Contributes to better coordination and efficiency in disaster victim identification.

Sl. NO	Reference	Objective	Method	Key Findings	Relevance
4	Sulistijono, T Imansyah, (2018) **Implementation of Victims Detection Framework on Post Disaster Scenario**	To evaluate victim detection frameworks in post-disaster scenarios using cameras.	Uses camera-based victim detection methodologies.	Camera-based frameworks effectively detect victims in post-disaster environments.	Applicable for rapid victim detection in post-disaster scenarios.
5	Valarmathi B, Kshitij J, Dimple R, (2023), **Human detection and action recognition for search and rescue in disasters using YOLOv3 algorithm**	To introduce a new disaster victim detection method using the YOLO algorithm	Integrates YOLO algorithm for rapid victim detection.	YOLO algorithm improves speed and accuracy in victim detection processes.	Focuses on advancements in victim detection technology using YOLO.
6	Hridy, RI Zaman (2022)**Multimodal human detection in disasters using AI & DL**	Develop AI and DL methods for multimodal human detection in disasters.	Combined thermal, visual, and audio modalities for detection.	Improved human detection efficiency in multi-modal disaster environments	Using AI and Dl for human detection by audio,video

Existing / Traditional System

1)Manual Image Analysis

Rescue teams rely on manually scanning through images and videos to identify victims. This process is slow and heavily dependent on human judgment.

2)Sources of Data

Visual data is collected from various sources like drones, CCTV footage, and smartphones. These images are often unorganized and need to be sorted manually.

3)Time-Consuming Process

Reviewing large sets of images can take hours or even days, which delays rescue operations and reduces the chances of finding survivors in time.

4)Human Error and Fatigue

In high-stress disaster situations, human observers can easily miss important visual cues due to fatigue, emotional strain, or poor image quality.

Methodology

1. Data Collection & Preprocessing

- Collected images of disaster scenes categorized into **‘victim’** and **‘non-victim’** folders.
- Performed **image resizing**, **normalization**, and **augmentation** to improve model performance.

2. Feature Extraction using ResNet50

- Used **ResNet50**, a deep CNN model pre-trained on Images, to extract high-level visual features from images.
- Removed the top layer and used it as a fixed feature extractor.

3. Classification using Random Forest

- Extracted features are passed to a **Random Forest classifier** to categorize images as "victim" or "non-victim".
- Chosen for its accuracy and ability to handle high-dimensional data.

Methodology

4. Model Training & Evaluation

- Split data into **training**, **validation**, and **testing** sets.
- Evaluated model performance using metrics like **accuracy**, **precision**, **recall**, and **F1-score**.

5. Web Interface with Flask

- Developed a simple **Flask web app** for users to upload images.
- The model processes the image and displays the prediction result in real-time.

Hardware & Software Requirements

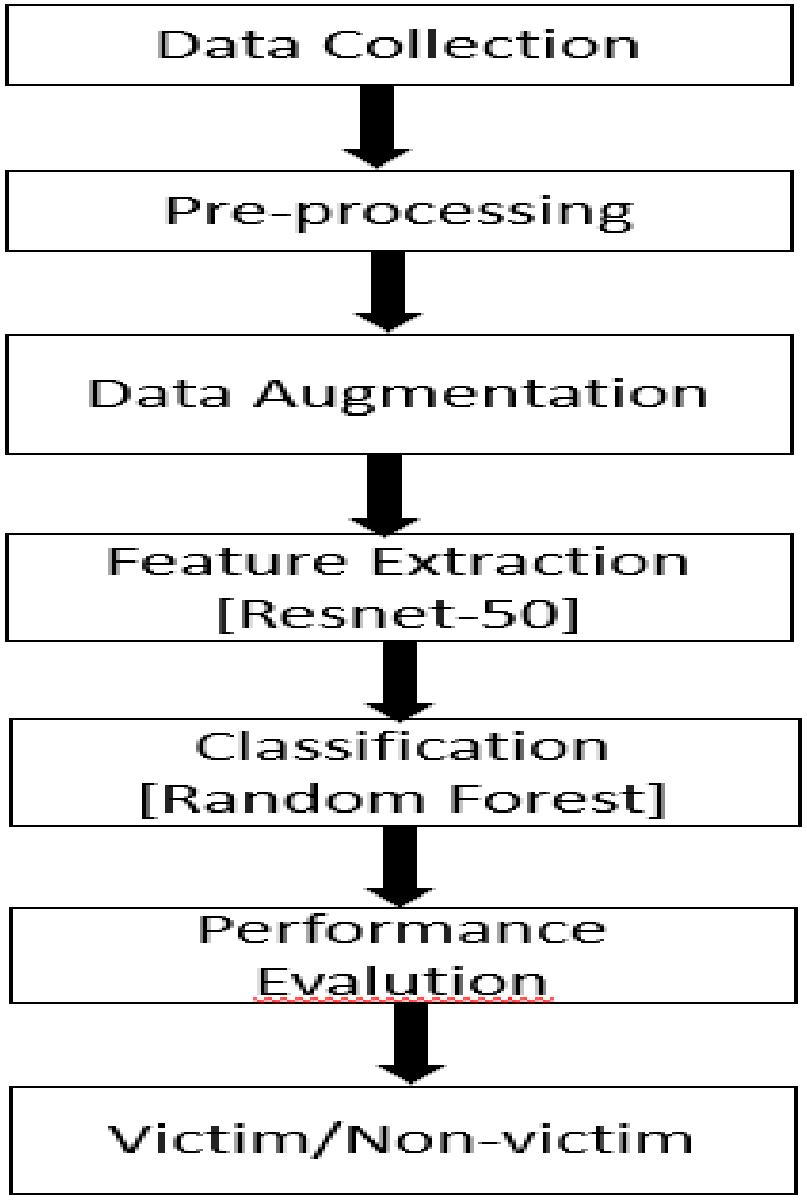
Hardware Requirements:

- **Processor**: Intel i5 or higher (i7/i9 recommended for training)
- **RAM:** Minimum 8 GB (16 GB or more preferred)
- **Storage:** 100 GB (for datasets, models, and logs)
- **Internet:** Required for downloading pre-trained models and dependencies

Software Requirements:

- **Operating System:** Windows
- **Programming Language:** Python 3.10
- **Frameworks & Libraries:**
  - TensorFlow / Keras (for ResNet50)
  - Scikit-learn (for Random Forest)
  - NumPy, Pandas, Matplotlib
  - OpenCV (for image processing)
- **Web Framework:** Flask (for web interface)
- **IDE / Tools:** Jupyter Notebook / VS Code / PyCharm

Design & Implementation



1)Dataset

Images are organized into 'victim' and 'non-victim' folders across training, validation, and testing sets. This structure helps in supervised learning and performance evaluation.

2)Feature Extraction

Pre-trained ResNet50 is used to extract deep features from each image. The final fully connected layers are removed, and the output of the convolutional layers is used as input for classification.

3)Classification Model

A Random Forest classifier is trained on the extracted features. It provides high accuracy, handles overfitting well, and works effectively with complex data.

4)Web Interface

A Flask-based web application is built where users can upload an image. The backend processes the image, makes predictions, and displays the result on the same page.

5)Result Display

Once an image is uploaded, the system shows whether the image contains a victim or not — providing a fast and user-friendly experience for rescue teams or volunteers.

Results & Observations

1)Performance Metrics

- **Accuracy:** Measures the overall correctness of predictions.
- **Precision**: Indicates that most images classified as "victim" are correct.
- **Recall**: The model successfully identifies most actual victims.
- **F1-Score**: Balanced performance across both metrics.

2)Observations

- ResNet50 provided strong feature extraction, even in complex backgrounds.
- Random Forest worked effectively on extracted features with minimal overfitting.
- Real-time prediction using Flask was fast and responsive.

Results & Observations

Performance Metrics and Results:

Evaluation Metric	Values
Accuracy	100.00%
Precision	1.00%
Recall	1.00%
F1-score	1.00%

Advantages & Applications

Advantages:

- **Fast Detection**: Automatically detects victims in images within seconds.
- **Reduces Manual Work**: Minimizes the need for human image analysis.
- **High Accuracy**: Uses deep learning for reliable results.
- **User-Friendly**: Simple web interface for easy image uploads and predictions.
- **Scalable**: Can be extended to mobile apps or integrated into rescue drones.

Applications:

- **Disaster Response**: Helps rescue teams quickly identify victims in large disaster zones.
- **Drone Surveillance**: Can be integrated with drone systems to analyze aerial images in real time.
- **NGOs and Government Agencies**: Useful for organizations involved in relief and recovery operations.
- **Military and Defense**: Assists in search-and-rescue missions in conflict or remote zones.

Conclusion

- This project provides an **AI-powered solution** for detecting disaster victims from images, improving the speed and accuracy of rescue efforts.
- It uses **ResNet50** for deep feature extraction and a **Random Forest classifier** for reliable image classification.
- The system offers a **quick and efficient way to identify victims**, reducing the need for manual analysis.
- A user-friendly **Flask web interface** allows users to easily upload images and receive real-time predictions.
- Overall, the project demonstrates the **power of machine learning** in supporting emergency response systems and saving lives in critical situations.

References

[1] G. Seeja, A. S. A. Doss, V. B. Hency, "A Novel Approach for Disaster Victim Detection Under Debris Environments Using Decision Tree Algorithms With Deep Learning Features," *Journal Article*, 2022.

[2] R. K. Gali, R. Nakka, "A Novel Approach for Disaster Victim Detection Under Debris Environments Using Decision Tree Algorithms With Deep Learning," *International Journal of HRM and Organizational Studies*, 2022.

[3] S. Mahmud, A. A. Fime, J. H. Kim, "ATR HarmoniSAR: A System for Enhancing Victim Detection in Robot-assisted Disaster Scenarios," *IEEE Conference Proceedings*, 2022.

[4] Y. J. Wong, M. L. Tham, B. H. Kwan, E. M. A. Gnanamuthu, "An Optimized Multi-Task Learning Model for Disaster Classification and Victim Detection in Federated Learning Environments," *IEEE Journal*, 2022.

[5] I. A. Sulistijono, T. Imansyah, M. Muhajir, "Implementation of Victims Detection Framework on Post Disaster Scenario," *IEEE Conference Proceedings*, 2018.


THANK
YOU