

# FORGERY CHECK

AI-Powered Image and Video  
Forgery Detection

REVIEW - 3

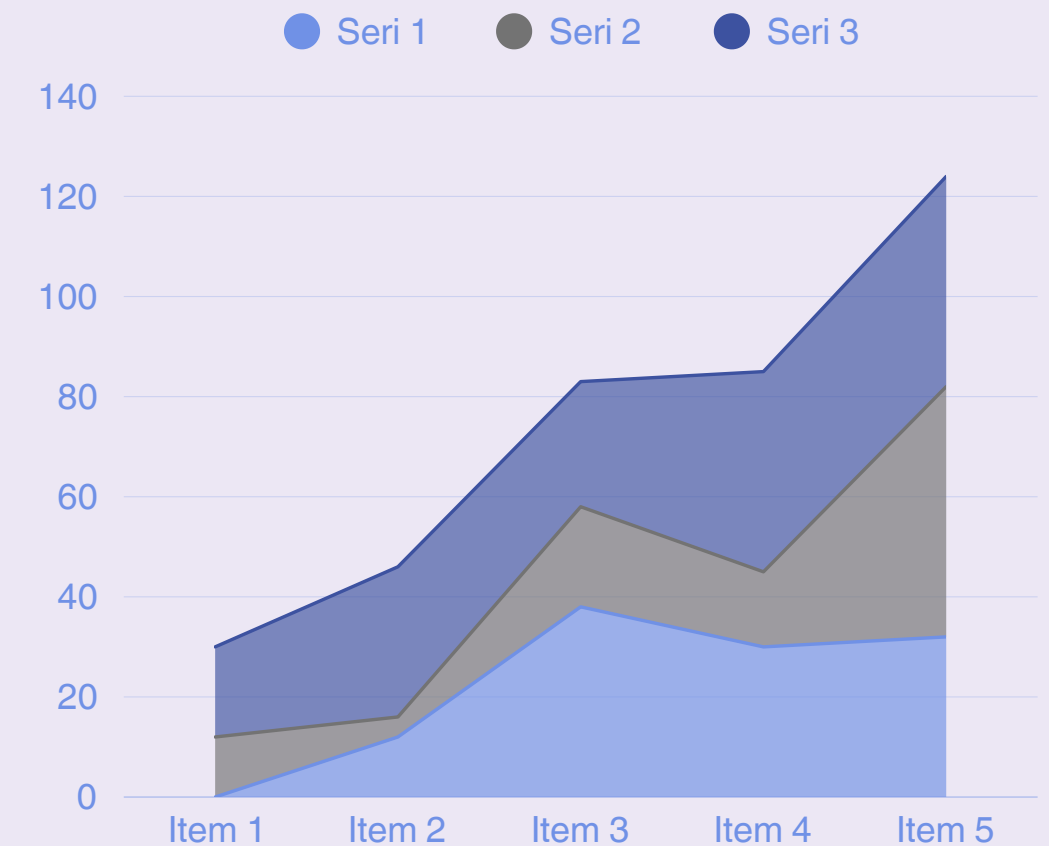


# INTRODUCTION

Forgery Check is an advanced tool designed to detect forgeries in images and videos using AI and machine learning. In a digital era where media manipulation is common, ensuring authenticity is crucial. Forgery Check provides a reliable solution for identifying manipulated media, covering a wide range of forgery types such as splicing, copy-move forgeries, and deep fakes. With its user-friendly interface, users can easily upload files, perform analyses, and receive detailed reports on media authenticity.

## India has witnessed an increase in the circulation of fake news and manipulated images/videos.

Forged media has contributed to societal unrest and misinformation, especially during elections. There is a growing need for AI-based systems to combat the impact of media manipulation. India is adopting AI-based solutions, but still lags behind other countries in terms of implementation. The proposed system can help in improving media integrity and fighting the spread of fake content.



# PROBLEM STATEMENT



- Increasing cases of fake images and videos spreading false information
- Difficulty in distinguishing between real and manipulated media
- Need for an automated system for detection and classification of forgeries
- Lack of accessible tools for forensic analysis of digital media

# PROJECT SCOPE

## Objective:

1. Deliver a robust solution for detecting digital forgeries in images and videos using AI. Ensure accurate and reliable analysis to maintain media integrity.

## Scope:

2. Analyze image and video files to detect splicing, copy-move forgeries, and deep fakes. Support diverse file formats for broad applicability across industries.



# PROJECT OVERVIEW

- 01** Django-based web application with responsive design
- 02** Dual-focused: Image and video forgery detection
- 03** Multiple detection algorithms for identifying forged regions
- 04** Detailed reporting with visual indicators of forgery

# EXISTING SYSTEM Vs. PROPOSED SYSTEM

## Existing System



These systems typically focus on either image or video content, not both. Additionally, many existing tools do not utilize deep learning models, resulting in lower accuracy and poor performance on compressed or low-resolution files.

## Proposed System



Our proposed system leverages deep learning techniques and AI algorithms to detect both image and video forgery. It uses convolutional neural networks (CNNs) and video analysis models to classify and detect forgeries.

# FEATURES

## Image Forgery

Image Forgery Detection utilizes Machine Learning Prediction model and other techniques like ELA, Noise Analysis, Copy-Move Detection to detect forged regions

## Web Application

Forgery Check provides a user-friendly interface for easy media uploads and forgery analysis, with responsive design for all devices

## Video Forgery

Video Forgery Detection utilizes frame-by-frame analysis and AI to detect inconsistencies and deep fakes, ensuring video content integrity

01

02

03



# WEB APPLICATION OVERVIEW



Users can upload  
images/videos  
for forgery  
detection

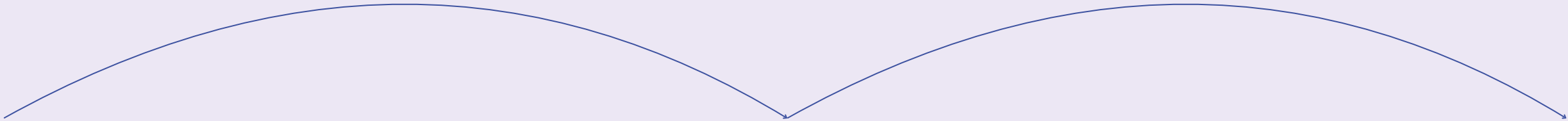


The backend  
processes the  
media using AI  
models



Results are  
displayed with  
confidence  
scores

# TECHNOLOGY STACK

A diagram showing three components of a technology stack: Frontend, Backend, and Machine Learning. Two blue curved lines connect the Frontend and Backend components, and another two blue curved lines connect the Backend and Machine Learning components, indicating their interdependence.

## Frontend

Utilizes HTML, CSS, and Bootstrap for responsive design.

## Backend

Built with Django, a high-level Python web framework.

## Machine Learning

Utilizes models trained on diverse datasets to enable accurate forgery detection.

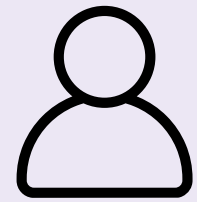
# IMAGE MODEL TRAINING

- **Input:** 128×128×3 preprocessed images
- **Model:** CNN for binary classification (Forged vs Authentic)
- **Training:** Tracked accuracy and loss over epochs
- **Evaluation:**
  - Accuracy: 87% on validation data
- **Testing:**
  - Checked individual images
  - Verified predictions with confidence scores

# VIDEO MODEL TRAINING

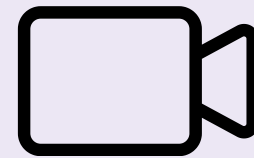
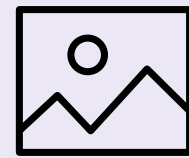
- **Data Loading:** Load the preprocessed training data (Xtrain, Ytrain) containing 3500 frames with dimensions of  $240 \times 320 \times 3$ .
- **Model Architecture:** Utilizes the ResNet50 architecture, a deep convolutional neural network pre-trained on ImageNet, for robust feature extraction and classification.
- **Model Training:** The ResNet50 model is trained on the training dataset (Xtrain) to recognize forged vs. original video frames.
- **Model Saving:** Once trained, the model is saved for future inference in the detection phase.

# WORKFLOW



## User Login

Users access the platform via login or registration



## Media Upload

Users upload images or videos for forgery analysis.



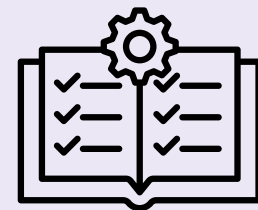
## Processing

System processes the media using machine learning models



## View Results

Users view analysis results, including confidence scores



## How it works

Users can explore how the system works, read FAQs, and contact support.



## User feedback

Users can submit feedback about their experiences and queries on contact us form.



# IMAGE FORGERY WORKFLOW



## Image Upload

User uploads a suspicious image through the web interface.



## Initial Analysis

The system performs ELA and runs the deep learning model to classify the image.



## Detailed Analysis

The user can select different analysis methods like ELA, Noise Analysis, and more.



## Results

The system displays analysis results with a confidence score and visual evidence.

# RESULTS

The system predicts the authenticity of the uploaded image using a trained deep learning model, classifying it as either 'Authentic' or 'Forged.' Forged regions are highlighted using specific analysis techniques.

- **Mask** : Black and white image that highlights the tampered areas.
- **Error Level Analysis (ELA)** : Detects compression inconsistencies.
- **Edge Detection** : Highlights boundary inconsistencies.
- **Noise Analysis** : Reveals inconsistent noise patterns.
- **Copy-Move Detection** : Identifies duplicated regions within the image.

# VIDEO FORGERY WORKFLOW



## Video Upload

User uploads a suspicious video through the web interface.



## Frame Processing

The video is split into frames and resized for consistent analysis.



## Model Prediction

A deep learning model evaluates all frames for potential forgery signs.



## Results

The system displays the final verdict along with forged frame count and probability scores.

# RESULTS

- **Video Authenticity:**

Authentic: No forged frames detected.

Forged: Presence of forged frames detected.

- **Number of Forged Frames:**

Total number of frames identified as forged.

- **Metadata:**

Additional details about the video file.

- **Total Frames Processed:**

Total number of frames analyzed in the video.



# FUTURE ENHANCEMENTS

- Extend functionality to analyze and detect tampering in PDF documents
- Implement secure password recovery using email/OTP-based verification for enhanced user experience.
- Launch a dedicated mobile application to allow forgery detection directly from mobile devices.
- Store and manage past analysis reports for users to track, download, or compare over time.
- Allow users to upload and analyze multiple files (images/videos/PDFs) simultaneously.
- Continuously refine the system based on user insights.



# CONCLUSION

Forgery Check is a tool designed to ensure the authenticity of digital media through advanced AI-driven analysis. By addressing challenges such as file handling and accuracy, the project has established a reliable platform for forgery detection. Future work aims to expand capabilities and integrate new technologies, enhancing the system's effectiveness and reach. As digital manipulation becomes more sophisticated, Forgery Check remains committed to providing users with trustworthy analysis and maintaining the integrity of digital communications.



# THANK YOU