International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Deepfake Detection (A Deep Learning Approach)

Sahana Sharma M[1], *Vedant M Varadai*[2], *Tushar Bhosale*[3], *Pavan Jadhav*[4], *Gopal Koparde*[5], *Sameera J*[6]

[1]Dayananda Sagar Academy of Technology &amp; Management, Bengaluru
[2]Dayananda Sagar Academy of Technology &amp; Management, Bengaluru
[3]Dayananda Sagar Academy of Technology &amp; Management, Bengaluru
[4]Dayananda Sagar Academy of Technology &amp; Management, Bengaluru
[5]Dayananda Sagar Academy of Technology &amp; Management, Bengaluru
[6]Dayananda Sagar Academy of Technology &amp; Management, Bengaluru
Email: [1]sahana-csai@dsatm.edu.in , [2]1dt23ca412@dsatm.edu.in , [3]1dt23ca411@dsatm.edu.in, [4]1dt23ca405@dsatm.edu.in [5]1dt23ca402@dsatm.edu.in , [6]1dt23ca408@dsatm.edu.in

I. ABSTRACT

Deepfakes are synthetic media where a person's likeness is digitally altered using deep learning techniques. The increasing presence of deepfakes presents significant challenges in terms of privacy, ethics, and digital security. As a result, the field of deepfake detection has emerged as a crucial area of research. This review provides a structured analysis of the developments in deepfake detection technologies, including methodologies, datasets, and the associated challenges. With rapid advances in AI and deep learning, various tools have emerged for media manipulation. While some are used for beneficial purposes in entertainment and education, others exploit these technologies for harmful ends—such as spreading misinformation, creating political unrest, or personal harassment. The term "deepfake" has become widely recognized to describe these manipulations. To present a comprehensive overview of deepfake detection methods, this paper reviews 112 significant works from 2018 to 2020. These approaches are categorized into deep learning models, traditional machine learning techniques, statistical analysis, and blockchain-based methods. A comparative analysis shows that deep learning-based solutions tend to outperform others in accuracy and robustness. As deepfake generation methods continue to evolve, it is imperative to develop equally advanced detection techniques to safeguard information integrity and personal identity.

II. INTRODUCTION

Recent advancements in artificial neural networks (ANNs) have significantly enhanced the ability to manipulate digital media. Tools like FaceApp and FakeApp make it easy for users to change facial features and swap identities in images and videos. These tools fuel the creation of deepfakes—videos or images generated using deep learning to appear convincingly real. The term "deepfake" is a blend of "deep learning" and "fake," originally coined by a Reddit user in 2017 who applied AI techniques to swap faces in videos. Deepfake technology relies heavily on Generative Adversarial Networks (GANs), composed of two neural networks—a generator and a discriminator—that work in tandem to produce increasingly realistic fake images. Given the rise of deepfakes, research on their detection has become more pressing. Various surveys have explored detection methods and performance metrics. Notable projects have involved manipulating videos to synchronize lip movements with alternate audio, and the technology has even been misused to create explicit content involving celebrities. Major tech companies like NVIDIA have contributed to the development of GAN-based architectures. As of 2018, the presence of GAN-related research and content on platforms like arXiv and Google increased significantly. This underscores the need for a systematic review encompassing deepfake detection methods, datasets, and challenges.
This paper aims to:

- Review existing literature in the field of deepfake detection.
- Present a novel taxonomy categorizing current detection methods.
- Evaluate experimental results and performance metrics.
- Offer insights and recommendations for future research.

We also developed a user-facing application where videos can be uploaded, analyzed, and classified as real or deepfake using our model. Our approach combines temporal frame analysis with a pre-trained ResNeXt CNN and LSTM architecture to achieve accurate detection.

III. PROBLEM DEFINITION AND SCOPE

Digital image and video manipulation has existed for years, but recent innovations in deep learning have made the creation of highly realistic fake content—commonly referred to as deepfakes—easier and more convincing than ever. These AI-generated media are now widely used for spreading misinformation, inciting political instability, or even perpetrating personal attacks and extortion. While generating deepfakes has become increasingly accessible, detecting them remains a significant technical challenge. Notorious instances have demonstrated how such content can manipulate public opinion or defame individuals. As a result, developing reliable detection tools is essential to mitigate the harmful effects of deepfakes. Our approach employs a Long Short-Term Memory (LSTM) neural network to detect temporal inconsistencies in video frames and a ResNeXt CNN to extract frame-level visual features. This combination allows us to identify anomalies that reveal the presence of tampering. Traditional machine learning methods are advantageous due to their interpretability and ease of hyperparameter tuning. We explored various ML techniques before settling on a balanced deep learning approach. By training on both real and fake videos, we minimized model bias and improved generalization. After evaluating multiple training strategies and analysing different datasets, we selected the PyTorch framework for implementation due to its robust support for GPU acceleration, which is crucial for handling the computational load of video analysis.

IV. Hardware Resources Required

This project requires a system with high computational capabilities due to the nature of video and image processing. High-performance CPUs and GPUs are essential to handle batch processing and real-time video analysis efficiently.
Client-Side Requirements:
- Compatible web browser (any modern browser is sufficient)

V. Software Resources Required

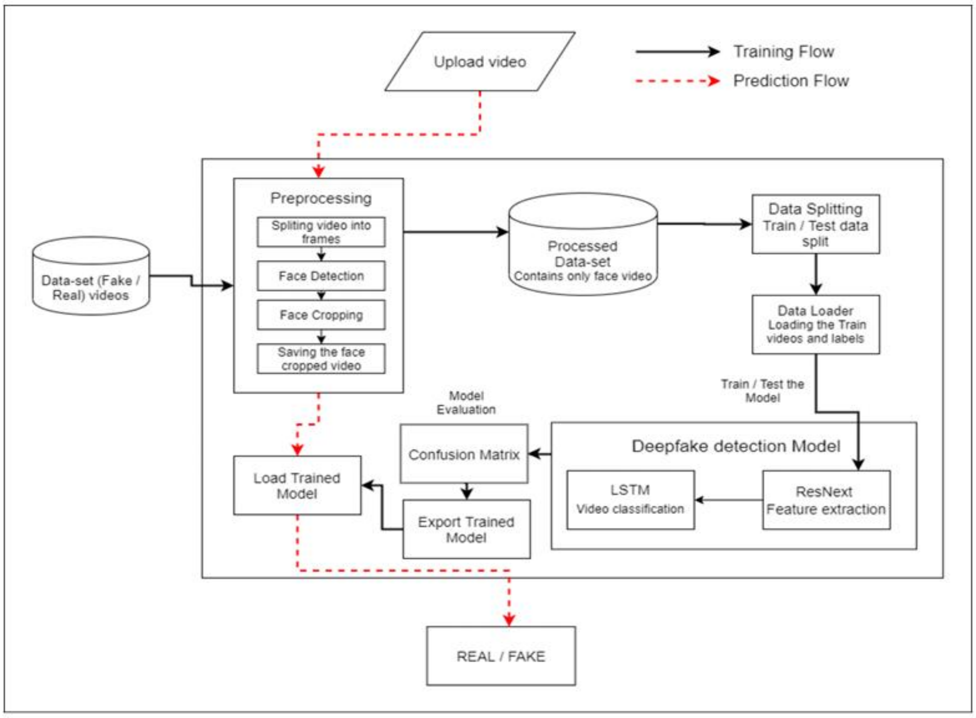
- Operating System: Windows 7 or later
- Programming Language: Python 3.0
- Frameworks: PyTorch 1.4, Django 3.0
- Cloud Platform: Google Cloud Platform
- Libraries: OpenCV, face-recognition

Parameters Used for Detection:

1. Eye blinking pattern
2. Teeth structure and enhancement
3. Eye spacing
4. Facial hair presence (moustaches)
5. Duplicate facial features (eyes, ears, nose)
6. Iris segmentation
7. Facial wrinkles
8. Head pose inconsistencies
9. Face orientation
10. Skin tone variation

VI. FLOWCHART

The operational structure of the system is based on a logical flow of data, starting from user input and culminating in the classification of a video. The data flow within the system is structured as follows:

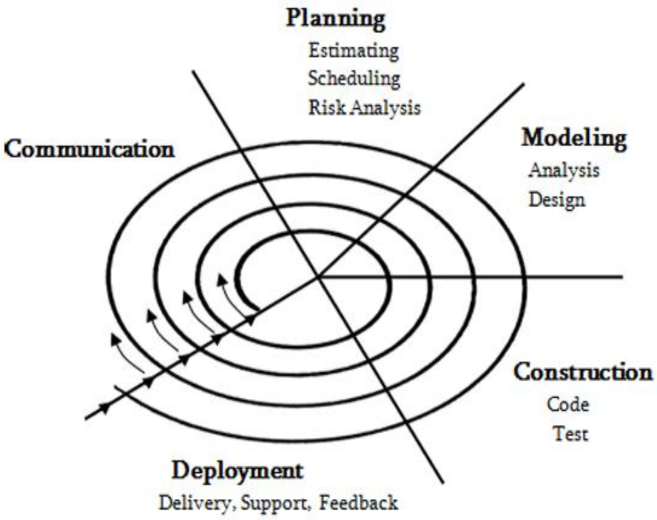

Upload video
Training Flow
Prediction Flow
Preprocessing
Spliting video into frames
Face Detection
Face Cropping
Saving the face cropped video
Data-set (Fake / Real) videos
Processed Data-set Contains only face video
Data Splitting Train / Test data split
Data Loader Loading the Train videos and labels
Train / Test the Model
Model Evaluation
Confusion Matrix
Deepfake detection Model
LSTM Video classification
ResNext Feature extraction
Load Trained Model
Export Trained Model
REAL / FAKE

- **Input:** Users upload a video through the interface.
- **System Processing:** The video is decomposed into frames. Each frame undergoes feature extraction using a pre-trained ResNeXt CNN model. These extracted features are passed to an LSTM model to analyze temporal dependencies and detect inconsistencies.
- **Output:** The system returns a classification (real or deepfake) along with a confidence score.

This logical flow ensures accuracy in detection and enables scalability and integration into broader applications.

VII. PROJECT MODEL ANALYSIS

The Spiral Model has been chosen as the development framework due to its iterative design and risk mitigation capabilities. It combines the flexibility of agile development with systematic stages of traditional models.



Reasons for Using the Spiral Model:

- Iterative Progression: Each phase of development includes planning, risk analysis, engineering, and evaluation, which ensures continuous improvement.
- Risk Management: The model is designed to identify and address risks at each stage.
- Client Involvement: Stakeholders are involved throughout the project, allowing feedback and refinements.

As the project includes several interconnected modules (feature extraction, frame analysis, model prediction, UI), the Spiral Model is an ideal fit to handle these dynamics and evolving requirements.

VIII. USE CASE VIEW

This section outlines the project's structural design and functionality from a user and developer perspective.

Primary Stakeholders:

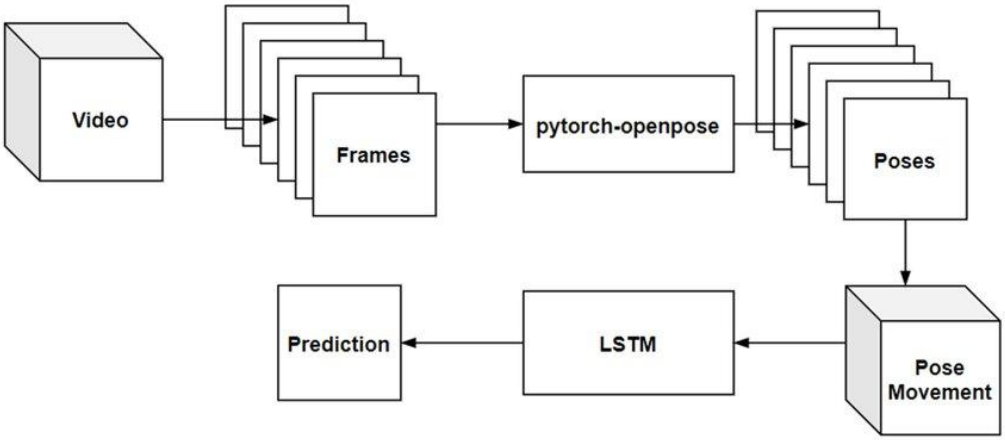
- Developers (current and future)
- Project sponsors and mentors

System Features:

- **Video Upload:** Users can upload video files for analysis.
- **Automated Analysis:** Videos are automatically broken into frames and processed by deep learning models.
- **Prediction Output:** The system displays whether the video is real or fake with a confidence percentage.

Documentation Includes:

- **Use Case Diagram:** Illustrates actor-system interactions.
- **Activity Diagram:** Outlines workflow from input to output.
- **Data Flow Diagram (DFD):** Represents the transformation of data across modules.
- **Functional Requirements:** Core functionalities such as model accuracy, real-time processing, and easy interface.
- **Non-Functional Requirements:** Reliability, scalability, and cross-platform compatibility.
- **Risk Mitigation:** Version control, regular updates, and testing across datasets.
-



This comprehensive blueprint ensures the project remains scalable and adaptable while staying aligned with core objectives.

IX. PROJECT IMPLEMENTATION

Deepfake technologies have been widely misused to create misleading content involving public figures, such as altered videos of political leaders and celebrities. These manipulated media can incite confusion and spread false narratives across social platforms. Thus, there's an urgent need for effective detection mechanisms. Our project addresses this issue using a dual-model architecture:

- A pre-trained Res Next Convolutional Neural Network (CNN) for framelevel feature extraction.
- A Long Short-Term Memory (LSTM) network to capture temporal inconsistencies across frames.

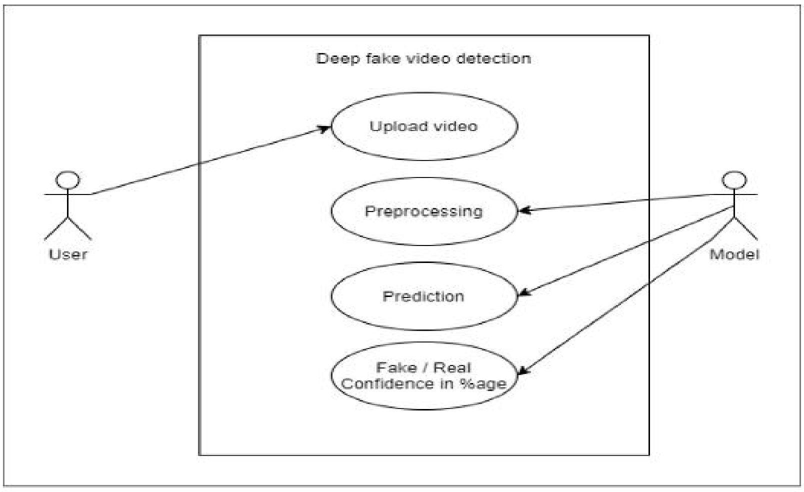
Key Technologies:

- **Res Next CNN (resnext50_32x4d):** A 50-layer network used for extracting detailed visual features from each frame.
- **Sequential Layer:** Stores the ordered output of frame features, preparing it for temporal analysis.
- **LSTM:** Processes sequential data to detect abrupt or unnatural transitions between frames.
- **ReLU Activation:** Applied for non-linearity and efficient training.
- **Dropout Layer (0.4):** Prevents overfitting by randomly deactivating neurons during training.

Training Strategy:

- **Dataset Split:** 70% for training and 30% for testing (balanced split: 50% real, 50% fake in each set).
- **Batch Loader:** Loads video batches along with labels.

- **Optimizer:** Adam optimizer with a learning rate of 1e-5 and weight decay of 1e-3.
- **Loss Function:** Cross-Entropy for classification.
- **Soft max Layer:** Converts logits into interpretable class probabilities.



This architecture ensures the model is capable of processing multiple frames per second and providing real-time predictions with high confidence.

X. CONCLUSION & FUTURE SCOPE

This project presents a neural network-based approach to detect deepfake videos, combining frame-level analysis with sequential modeling. By integrating a ResNeXt CNN with an LSTM network, our system effectively captures visual and temporal discrepancies to distinguish between real and synthetic media. The solution is capable of analysing short clips (up to 10 frames per second) and delivering accurate classification results. Testing on public datasets like Face Forensics++, Celeb-DF, and DFDC validated its performance and adaptability. The model was further integrated into a user-friendly interface to allow nontechnical users to upload videos and receive real-time predictions, bridging the gap between advanced AI and practical usability.

Future Scope:

- **Model Generalization:** Future models should be resilient to newer, more sophisticated deepfake generation techniques by incorporating diverse datasets and evolving training protocols.
- **Real-time Integration:** Expanding the tool's usage in live systems, such as social media filters, video conferencing platforms, and surveillance networks.
- **Multimodal Analysis:** Incorporating audio-visual synchronization for detecting inconsistencies in sound and lip movement.
- **Explainable AI:** Implementing techniques to explain detection outcomes, increasing user trust in critical sectors like journalism, forensics, and governance.
- **Blockchain & Watermarking:** Partnering detection tools with blockchain or digital watermarking to verify media authenticity at the point of creation.
- **Public Education Tools:** Leveraging this technology for awareness campaigns that help users identify and report suspicious content.

With these advancements, we can move toward a future where the integrity of digital information is protected against the growing threat of synthetic media.

XII. REFERENCES

1. Meenakshi Sundaram & C. Nandini (2017). "ASRD: Algorithm for Spliced Region Detection in Digital Image Forensics."
2. Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Niebner, M. "Face Forensics++: Learning to Detect Manipulated Facial Images." *arXiv preprint*, arXiv:1901.08971.
3. Kaggle. "Deepfake Detection Challenge Dataset." Available at:
4. https://www.kaggle.com/c/deepfake-detection-challenge/data (Accessed March 26, 2020).
5. Li, Y., Yang, X., Sun, P., Qi, H., & Lyu, S. "Celeb-DF: A Large-scale Challenging Dataset for Deep Fake Forensics." *arXiv preprint*, arXiv:1909.12962.
6. Creative Blog. "10 Deepfake Examples That Terrified and Amused the Internet." Available at: https://www.creativebloq.com/features/deepfakeexamples(Accessed March 26, 2020).
7. TensorFlow. "TensorFlow Machine Learning Framework." Available at: https://www.tensorflow.org/(Accessed March 26, 2020).