Deepfake Face Recognition using Pretrained Vision Transformers and LSTMs

M. Sowmya[1], D. Bandhavi[2], P. Vinaya Padma Sri Harshitha[3], S. Bhargav Rama Raju[4], M. Kapil Raj[5] and Arul Elango[6]
{ sowmyamuppalla09@gmail.com[1], divvelabandhavi20@gmail.com[2], harshithapettela2004@gmail.com[3], sbhargavramaraju007@gmail.com[4], Kapilraj090303@gmail.com[5], arulelango2012@gmail.com[6] }

Department of Advanced Computer Science & Engineering, Vignan's Foundation for Science, Technology and Research, Vadlamudi, Guntur, 522213, Andhra Pradesh, India[1, 2, 3, 4, 5, 6]

Abstract. The advancement in deepfake technology has significant issues with the authenticity and reliability of digital media empower in the production of hyper-realistic artificial facial images and videos by deep learning models. Deepfake technologies have extensive uses in entertainment and virtual reality but pose threats to misinformation, identity theft and cybersecurity. The paper is an in-depth analysis on the deepfake face detection methods leveraging state-of-the-art deep learning techniques. Particularly, we use Vision Transformers (ViT) and Long Short-Term Memory (LSTM) for both image and video-based detection, leveraging self-attention mechanisms to learn spatial and temporal dependencies. Our method features state-of-the-art feature extraction techniques, such as frequency domain analysis and attention-based representations, to enhance detection accuracy. We tested these models on benchmark datasets, assessing their adversarial robustness and generalizability across various deepfake generation methods. The system illustrates the capability to accurately predict and detect manipulated facial videos and images and offer real-time classification results via an interactive interface.

Keywords: Deepfake Detection, Vision Transformers (ViT), Long Short-Term Memory (LSTM), Cybersecurity, Identity Fraud, Misinformation.

1 Introduction

Deepfake technology has revolutionized the manipulation of digital media enables in the generation of highly realistic synthetic content. Traditional approach like image and video splicing techniques involve copying, modifying, or remove objects within an image or video, deepfake generation leverages advanced neural network architectures. There exist two prominent methods of neural networks commonly in usage: Deep Autoencoders and Generative adversarial networks (GANs) Deep autoencoders have an encoder and a decoder, where the encoder maps an input image to the representation in the latent space and the decoder is used to reconstruct the image. By having two autoencoders that share an encoder and two different decoders, deepfake content is produced by feeding an image of a person and reconstructing it as another. Generative adversarial networks (GANs) enhances in deep fake realism through an adversarial training process involves two components: a generator and a discriminator. The genera tor fabricates synthetic content from random noise, while the discriminator tries to differentiate between real and synthetic content. Through an iterative training the generator gets better at generating realistic outputs make deepfake videos look more real.

ICITSM-Part I 2025, April 28-29, Tiruchengode, India

DOI 10.4108/eai.28-4-2025.2357797

1.1 Types of Deepfake Face Synthesis

Deepfake face synthesis is majorly categorized into three types:

- Head Puppetry- Control of facial movements and expressions of a non-moving image.
- Face Swapping- Swapping one individual's face with another individual's in video.
- Lip-Syncing- Changing lip motions to accommodate different speech input.

Although their improvements, deepfake generation methods still create some artifacts that are exploitable for detection. These artifacts are resolution mismatch between face and background, facial and body feature discrepancy, and artificial facial muscle motion. These imperfections assist in deepfake detection via examination of pixel inconsistencies, motion glitches, and feature mismatches

2 Related works

Recent advancements in Deep fake face detection and GAN model have significantly contributed to generation of Deepfake Faces. Below is a summary of relevant articles that inform the direction of this paper.

The subject of deepfake detection Edwards et al. [1] has progressed significantly over the past few years due to advances in machine learning and deep learning, which have introduced novel approaches to address the societal threat posed by AI generated forgery videos. One notable study applied several algorithms to benchmark datasets and emphasized the importance of accurate detection techniques to prevent the spread of misinformation. In this study, depseudo videos were paired with 300 real videos to form an imbalanced dataset composed of 50% real and 50% synthetic content. The preprocessing pipeline included channel-wise subtraction of means, resizing video frames to 299×299 pixels, and normalizing the sequence length of inputs. A deep learning-Kadam et al. [2] based detection system combining ResNet18 with K-Nearest Neighbors (KNN) achieved an accuracy of 89.5% in identifying tampered images, underscoring the critical need to combat fake news on social media platforms .Rafique et al. [3] As a part of a move towards improvement in real-time detection, a novel CED-DCGAN-based system attained 98.23% in detecting high-fidelity deepfakes, vastly improving video communications security and resolving challenges brought by advancing forgery methods in digital media.

Sharma et al. [4] Yet another new method that combined Graph Neural Networks (GNNs) and Convolutional Neural Networks (CNNs) recorded an impressive accuracy of 99.3% across various datasets, highlighting its applicability in real time and requiring future studies on model robustness and interpretability .El-Gayar et al. [5] One such systematic review polled earlier deepfake detection research, and some major drawbacks were some identified limitations including computational expense, lack of real-world feasibility, and possible misidentification. The survey also ignited the need for developing more robust and resilient detection methods. Ramanaharan et al. [6] Convolutional Neural Networks were utilized to identify deepfakes in videos as well as images as well, where various training data sets enhanced detection and emphasized the need for ongoing innovation to keep up with the growing complexity of deepfakes. Rana et al. [7] An Xception-based architecture was used in a CNN model to achieve state-of-the-art accuracy for both the DFDC and FaceForensics++ datasets through key frame extraction to decrease computation complexity and thus be practical for social media use. Mitra et al. [8] Besides that, a revolutionary ResNet-Swish-BiLSTM model which combined CNNs

with Bidirectional LSTM layers achieved an accuracy of 96.23% using the FF++ dataset and was even better as it was resistant to an extremely large proportion of manipulation strategies.

Qadir et al. [9] Performance comparison of models like Xception, Multi-task, and Capsule Networks was done across datasets such as DF-TIMIT and Celeb-DF with excellent performance for facial warping and visual artifact detection based on accuracy value measures reported. Ramadhani and Munir [10] model integrating the CNN and RNN architectures also came into the limelight for detecting deepfake videos after identifying manipulation artifacts along temporal and spatial axes. This research employed multiple training sets and suggested a web-based system for real-time high-accuracy classification. Another research Potdar et al. [11] based on the Xception-based CNN model also demonstrated high accuracy on FaceForensics++ and DFDC datasets, once again highlighting the significance of effective feature extraction for social media application.Edwards et al. (2020) [12] proposed a deepfake video detection technique using a ConvLSTM model that can identify spatial and temporal inconsistencies in forgery content quite effectively. They created a dataset of 600 videos where each half has real and the other half synthetic clips. Frame resizing to 299×299 pixels, mean subtraction across channels, and normalization of sequence lengths were used in preprocessing. ConvLSTM network was trained to detect deepfake artifacts in frame sequences and achieved an incredible accuracy of 95.5%. The study highlights the importance of temporal modeling and robust preprocessing to enhance detection rate of AI-created fake media.

Nguyen et al. (2024) [13] opposed self-supervised vision transformers (ViTs) such as DINO and MAE to the conventional ConvNets to detect deepfakes. Nguyen et al. experimented with frozen and partially fine-tuned ViTs with simple classifiers. In different experiments carried out over different datasets, fine-tuned DINOv2 ViT-L/14-Reg had the best performance with a 94.38% accuracy and 5.63% EER. The experiment validated the greater explainability and generalizability of self-supervised ViTs, especially in the case of attention-based explanation, highlighting the significance of SSL and partial fine-tuning towards effective yet understandable detection. Neelima et al. (2024) [14] also proposed a deepfake face detection system using LSTM networks to learn temporal relationships among video frames. The model was trained on a variably sized dataset of real and fake videos by utilizing pre-processing operations including facial landmark extraction and optical flow. By applying LSTM to other machine learning models, the system maintained high accuracy and low false positives. The study demonstrates the strength of temporal modeling and ensemble approaches in building stronger deep fake detection systems. Karandikar et al. (2020) [15] proposed a deepfake detection model from CNN with transfer learning on top of the VGG16 network. The model is presented with video frames and identifies spatial inconsistencies caused by deepfake generation processes. Preprocessing includes face alignment and extraction, feature learning, and real/fake classification. The model was trained using the Celeb-DF dataset and achieved 70% accuracy. The research is focused on the ability of CNN for detection of spatial forgery and suggests adding temporal models to be used for performance improvement in the case of compressed and low-resolution media. Manish et al. [16] presented a deepfake detection model based on CNN and Xception, which was trained on a Kaggle dataset. The model is applied by using preprocessing, feature extraction, and classification for real and fake face images and videos. The best accuracy was reported by Xception with 99.3%, followed by CNN with 98.7%. Ensemble modeling, transfer learning, and metrics such as accuracy and F1-score are highlighted in research to make robust detection. The model is highly deployable but might need to be retrained for real-time adaptability. Arya Shah et al. (2024) [17] suggested a deepfake detection system using ResNeXt CNN for frame-level

feature extraction and RNN-based LSTM for temporal video classification. The system is suggested to identify frame inconsistency and facial warping caused by deepfake generation. Implemented on a balanced dataset, the system was found highly accurate and implementable in real-time. The article acknowledges the intersection of spatiotemporal attributes in detecting deepfakes with high success, presenting a successful model in deterring manipulated media using deep learning methodology. Soudy et al. (2024) [18] presented a deepfake detection model combining CNNs and convolutional vision transformers (CViT). Three sub-models were used in the model to obtain features from the eyes, nose, and full face, respectively, fol lowed by majority voting for classification. Employing Face Forensics++ and DFDC datasets, the CNN model achieved 97% accuracy, while the CViT-based model achieved 85%. The research indicates multi-region analysis and model fusion enhance detection robustness, offering a robust solution for detecting tampered media under numerous conditions

3 Data Collection & Generation

Since deep learning models don't take raw videos directly, you generate structured data for model input

a. Extract Frames from Videos
   • Videos are converted into frames (images) using OpenCV.
   • You sample at 5 frames per second, and limit to 60 frames max.
   • Every frame is resized (224x224) and preprocessed (normalized, RGB scaled) for EfficientNet.
b. Generate Features
   • Utilize EfficientNetB0 (pre-trained) to obtain informative features (1280-d vectors per frame).
   • Now every video becomes a 60×1280 matrix - a time series of visual features

3.1 Our Dataset

Celeb-DF is a dataset full of these kinds of videos - some are real, and some are DeepFakes. It was made to help researchers and developers build tools that can tell the difference. Celeb-DF uses real video clips of 59 well-known public figures, collected from YouTube. Then, using advanced DeepFake tech, fake versions were created by swapping faces between them.

- Real videos: 590
- Fake videos: 5,639
- Total: Over 6,000 videos

3.2 Cleaning the Dataset

Cleaning ensures that the dataset is consistent, complete, and usable. The cleaning process has been implemented through the following steps

1. Ignore Missing/Corrupted Videos:
   - Videos that couldn't be read using CV2 were ignored.
   - VideoCapture().
   - For impossible-to-read videos, empty arrays of frames were returned.
2. Make Frame Counts Uniform:

- If video contained 60 frames → padded frame sequence with zero vectors.
- If video contained 60 frames → truncated at first 60 frames.
- That provides a regular input shape for the LSTM model.

3. Shuffle and Class Balance:
   - Videos were shuffled randomly to avoid bias during training.
   - Restricted to 2000 videos in order to maintain memory and processing reasonably.
   - If necessary, you may also balance real vs. fake video counts (if the dataset is imbalanced).
4. Save Cleaned Data:
   - Cleaned, processed data saved as NumPy arrays.

4 Methodology

In order to detect DeepFakes in both photos and videos, this study suggests a dual-pipeline deep learning system. In order to train robust models, it makes use of the Celeb-DF (v2) dataset, which provides high-resolution actual and fake celebrity videos with few artifacts. A collection of videos was processed for video analysis by extracting 5 frames per second, up to 60 frames. The extracted frames were then scaled and input into EfficientNetB0 in order to obtain deep feature embeddings. A bidirectional LSTM-based classifier was then used to predict authenticity and capture temporal trends using these embeddings. The processed features were saved as NumPy arrays for reuse and efficiency during training and testing.

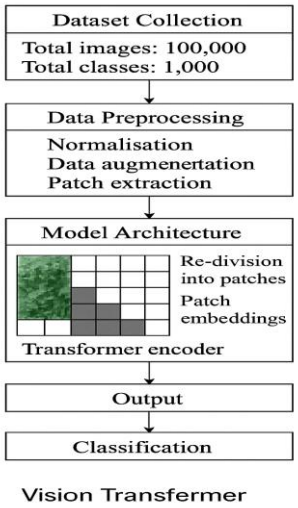

Fig. 1. ViT architecture.

An ensemble of Distilled Transformer (DeiT) and Vision Transformer (ViT) models was refined for picture classification in order to identify modified faces. Majority voting and confidence averaging were used to create the ensemble predictions, with" uncertain" serving as a fallback if model confidence fell below 70%. To offer a user-friendly interface for uploading and

evaluating media, a Flask web application was developed. It manages problems such as unreadable files or unsupported formats and incorporates backend logic to route inputs to the image or video pipeline according to file type. Early Stopping and ReduceLROnPlateau callbacks were employed during model training in order to enhance convergence and avoid overfitting. For the LSTM layers to maintain consistent input lengths, every video sequence was either padded or trimmed. Celeb-DF samples were used to fine-tune the image models after they were initialized with pretrained weights on ImageNet. With improved loading and preprocessing, the complete system facilitates real-time inference. Because it makes use of both temporal and spatial information, this hybrid method is resistant to different manipulation techniques.

The Vision Transformer (ViT) (fig 1) model begins with a dataset of 100,000 images of 1,000 classes. Data preprocessing is done through normalization, augmentation, and patch extraction. Images are divided into patches, embedded, and fed into a transformer encoder. The patches are fed into the encoder to capture spatial relationships and context. The output is then fed into a classification layer to make predictions for the class of the image.

The flowchart (Fig 2) illustrates an LSTM-based image classification process. It begins with gathering a dataset of 100,000 images divided into 1,000 classes. The data is preprocessed by normalization, augmentation, and sequence generation. Every image is reshaped into 224×224 pixels and divided into four frames. These frames are input into LSTM layers to extract sequence information. Finally, the output is forwarded into a classification layer to classify the image class.

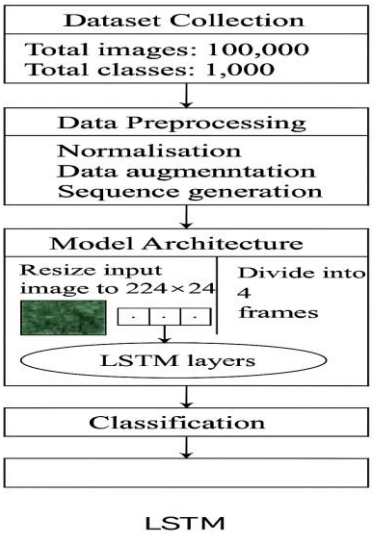

Fig. 2. LSTM architecture.

5 Flow Chart

Below are the flowcharts for Vision Transformer (ViT) and Long Short-Term Memory (LSTM):

The flowchart (fig 3) shows step-by-step execution for identifying if the uploaded image is real or forged through the usage of an ensemble model transformer. First, an image is uploaded by the user, then the image goes through preprocessing from Vision Transformer (ViT)-based and Data efficient Image Transformer (DeiT)-based feature extractors.

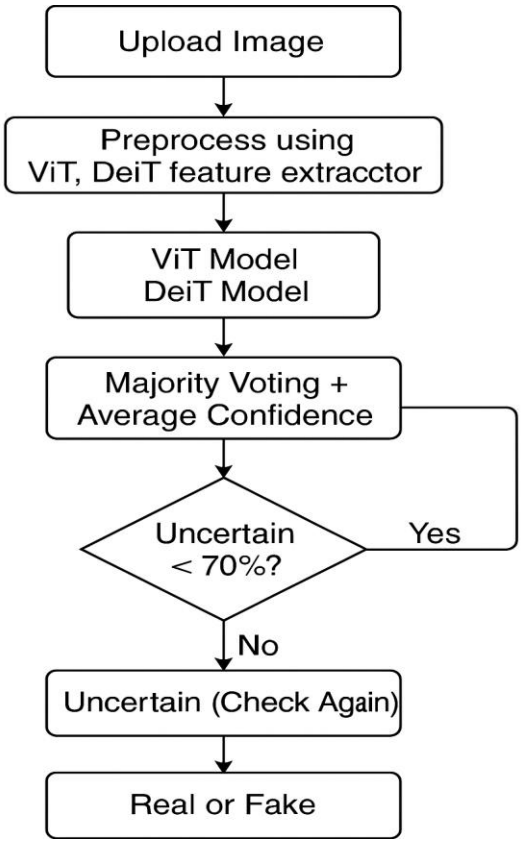

Fig. 3. ViT process.

Preprocessed features are input separately into the models ViT and DeiT in order to conduct classification. The outputs of both models are then combined through a majority voting system with average confidence scoring to arrive at a collective decision. If the average confidence is less than 70%, the output is labeled as uncertain, and a re-evaluation is triggered. If the confidence is more than 70%, the model goes ahead to make a final prediction, labeling the image as real or fake.

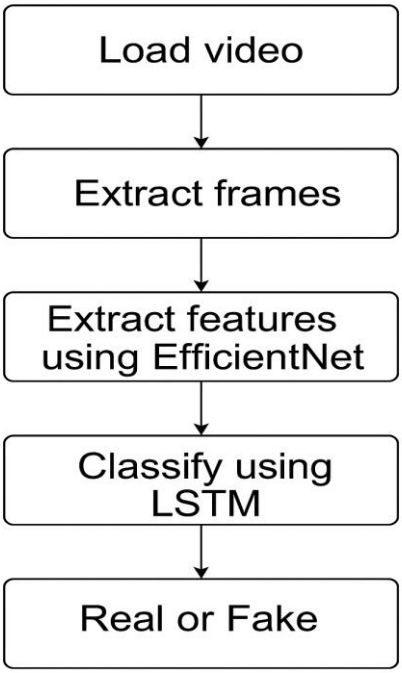

Fig. 4. LSTM process.

Fig 4 explains the process starting with the uploading of a video, which is decomposed into frames separate images that characterize the video at different moments in time. Every frame is run through feature extraction by EfficientNet, a fast-convolutional neural network that is widely known as fast and accurate in image processing tasks. Extracted features preserve spatial information from all the frames. The frame feature sequence is then fed into an LSTM network, having acquired temporal patterns and temporal dependency - an inherent requirement for video data processing. The model then, considering this temporal analysis, labels the video as authentic or not, most often in deepfake cases.

6 User Interfaces

The purpose of the image upload (fig 5) area is to allow users to submit facial images for verification of their authenticity.

Functionality: The backend feeds the submitted image to the Vision Transformer (ViT) and DeiT models for inference when the Detect button is hit.

Procedure:

- A preprocessed image is used.
- Each model examines the image separately.
- Ensemble logic (majority voting or average confidence) is used to integrate the results.

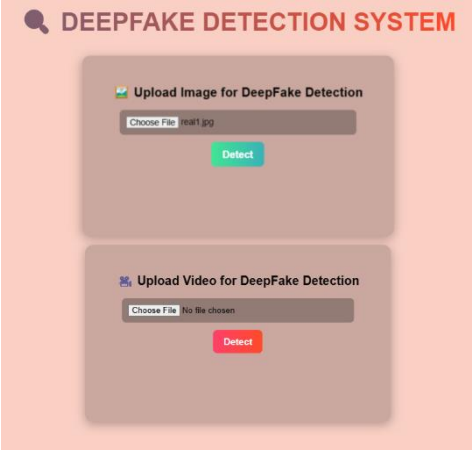

Fig. 5. User Interface While Uploading Image.

The goal of the video upload section is to allow individuals to submit short video clips for DeepFake detection.

Functionality:

The system breaks a video into frames by frame once you have submitted it and pressed Detect.

Method:

- pulls out five frames per second for up to 60 frames.
- A pretrained EfficientNetB0 is used to extract features from each resized frame.
- A bidirectional LSTM model is fed these features to examine temporal patterns and determine if the video is authentic or not.
- A confidence score for the final decision is returned.

7 Result

7.1 Results for Image Deepfake Recognition

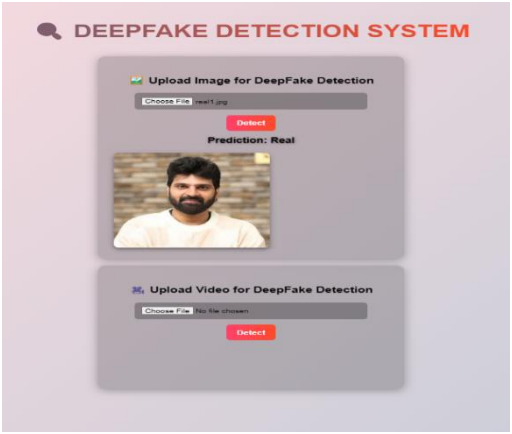

Fig. 6. ViT image result.

After examining (fig 6) an uploaded image called real1.jpg, the DeepFake recognition System's output is shown in the image. An ensemble model comprising Vision Transformer (ViT) and DeiT architecture was applied to the image. To ascertain authenticity, these models evaluated facial artifacts, visual patterns, and discrepancies.

The system gave back the following result: the prediction is Real.

Behind the Scenes: The image that was uploaded was:

- resized and preprocessed.
- input into individual prediction models for ViT and DeiT.
- The end prediction was determined by averaging the output.

Results for video deepfake recognition: After analyzing (fig 7) a file called id9 _id17 0000.mp4, the image shows the DeepFake recognition System's video analysis result. Five frames per second, up to a maximum of sixty frames, were extracted from the uploaded video. To extract feature embeddings, each frame was run through a pretrained EfficientNetB0 model. A bidirectional LSTM-based neural network that was intended to identify temporal patterns in video frames was then fed these features.

7.2 Technical Insight

The algorithm can identify minute temporal irregularities that could point to DeepFake alterations since it employs sequential learning with LSTMs.

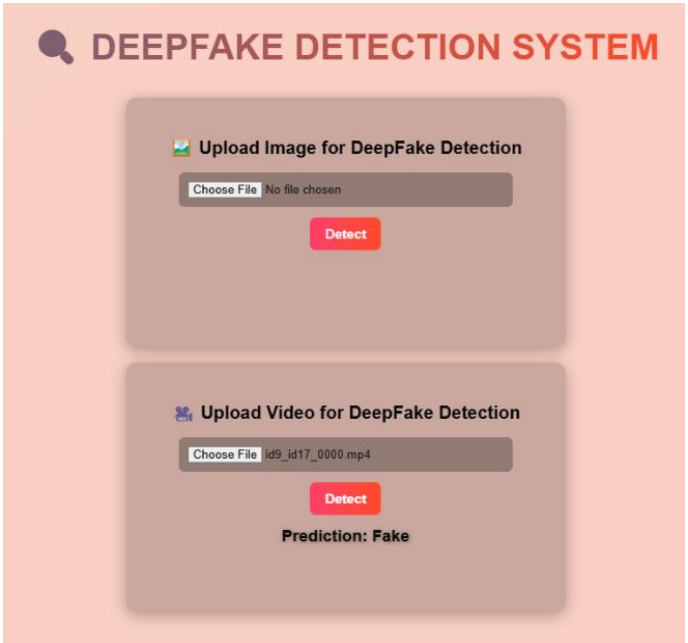

Fig. 7. LSTM video result.

8 Conclusion

A thorough and effective solution to discover and labeling modified media material- and more and more so currently in today's digital environment- is provided by the Deepfake Detection System constructed for this project. The solution is a bottom-to-top solution that realizes the interface ce between artificial intelligence and contemporary web technology by marrying a deep-learning-driven solid backend with a responsive, aesthetically impressive frontend. Within designing a responsive and user-centric design that works flawlessly on all devices, technologies like HTML5, CSS3 (with gradient background animations and glassmorphism effects), and JavaScript are utilized on the user interface side. Basic and informative user experience is delivered by the frontend's well-defined spaces for uploading photos and videos, with instant feedback, result showing, and previewing features.

Flask micro-framework is utilized by the backend for file processing, routing, and model execution. Advanced transformer-based models, including Vision Transformer (ViT) and Data-efficient Image Transformer (DeiT), deployed in PyTorch and Hugging Face's Transformers library, are the backbone of image-based deepfake detection. Uploaded images, before submitting to the models for inference, undergo preprocessing, which includes scaling, normalization, and tensor conversion. To improve accuracy and consistency, the predictions are averaged over models, and the results are given a confidence percentage reflecting how likely manipulation is. The system uses OpenCV to read frames from the input video file in order to perform video-based detection.

A trained LSTM-based model constructed using TensorFlow and Keras processes each frame. This model is particularly suited to detect minute movements and inconsistencies that are usually characteristic of deepfakes since it captures temporal patterns between frames. The model gives an overall summary of the results, and the final classification is made by the overall results of all frames. This model is very efficient because of its two-model architecture, which tackles both temporal and spatial issues of media manipulation, such as frame-based inconsistencies in movies and static modification in photos. Moreover, scalability, future development, and adaptability are readily made simple by the neat separation of frontend and backend. The project is a useful weapon in the future war against synthetic media and online disinformation from a practical standpoint. Applications in the real world could include forensic analysis, verification of the authenticity of news, and validation of social media content.

References

[1] P. Edwards, J.-C. Nebel, D. Greenhill, and X. Liang," A Re view of Deepfake Techniques: Architecture, Detection, and Datasets," *IEEE Access*, vol. 12, pp. 154718–154742, 2024, doi: 10.1109/AC CESS.2024.3477257.

[2] M.Kadam, S. Kate, V. Chavare, and S. Bhoite," Comparative Analysis of Deep Learning Techniques for Deepfake Detection: Evaluating Threats and Opportunities," in *Proceedings of the International Conference on Emerging Trends in Engineering and Technology*, vol. 10, pp. 1340–1348, 2024. doi: [Insert DOI if available].

[3] R. Rafique, R. Gantassi, R. Amin, et al.," Deep fake detec tion and classification using error-level analysis and deep learning," *Scientific Reports*, vol. 13, no. 7422, 2023. [Online]. Available: https://doi.org/10.1038/s41598-023-34629-3

[4] Sunil Kumar Sharma, Abdullah AlEnizi, Manoj Kumar, Osama Alfarraj, Majed Alowaidi, Detection of real-time deep fakes and face forgery in video conferencing employing generative adversarial networks, Heliyon, Volume 10, Issue 17, 2024, e37163, ISSN 2405-8440,

https://doi.org/10.1016/j.heliyon.2024.e37163.
https://www.sciencedirect.com/science/article/pii/S2405844024131945

[5] El-Gayar, M.M., Abouhawwash, M., Askar, S.S. et al. A novel approach for detecting deep fake videos using graph neural network. J Big Data 11, 22 (2024). https://doi.org/10.1186/s40537-024-00884-y

[6] Ramcharan Ramanaharan, Deepani B. Guruge, Johnson I. Agbinya, Deepfake video detection: Insights into model generalization — A Systematic review, Data and Information Management, 2025, 100099, ISSN 2543-9251, https://doi.org/10.1016/j.dim.2025.100099. https://www.sciencedirect.com/science/article/pii/S2543925125000075

[7] M. S. Rana, B. Murali and A. H. Sung," Deepfake Detection Using Machine Learning Algorithms," 2021 10th International Congress on Advanced Applied Informatics (IIAI-AAI), Niigata, Japan, 2021, pp. 458-463, doi: 10.1109/IIAI-AAI53430.2021.00079. keywords: Training; Deep learning; Machine learning algorithms; Computational modeling; Feature extraction; Task analysis; Informatics; Deepfake; Deep Learning; Machine Learning; Face Manipulation

[8] Mitra, Alakananda & Mohanty, Saraju & Corcoran, Peter & Kougianos, Elias. (2021). A Machine Learning Based Approach for Deepfake Detection in Social Media Through Key Video Frame Extraction. SN Computer Science. 2. 10.1007/s42979-021-00495-x.

[9] Abdul Qadir, Rabbia Mahum, Mohammed A. El-Meligy, Adham E. Ragab, Abdulmalik AlSalman, Muhammad Awais, an efficient deepfake video detection using robust deep learning, Heliyon, ISSN 2405-8440, Volume 10, Issue 5, 2024, e25757, https://doi.org/10.1016/j.heliyon.2024.e25757. (https://www.sciencedirect.com/science/article/pii/S2405844024017882)

[10] K. N. Ramadhani and R. Munir," A Comparative Study of Deepfake Video Detection Method," 2020 3rd International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, 2020, pp. 394-399, doi: 10.1109/ICOIACT50329.2020.9331963. keywords: Tion; Feature extraction; Pattern Visualiza recognition;Information and communication technology;Detection algorithms;Videos;Information integrity;deepfake;Generative Adversarial Networks;autoencoder;deep learning

[11] Potdar, R., Gidd, A., Kulkarni, S., Chavan, R., & Nikam, N. (2021). Deepfake video detection using deep learning. International Research Journal of Modernization in Engineering Technology and Science, 3(7), 1730–1734. https://www.irjmets.com

[12] P. Theerthagiri and G. b. Nagaladinne," Deepfake Face Detection Using Deep Inception Net Learning Algorithm," 2023 IEEE International Students' Conference on Electrical, Electronics doi: and Computer Science (SCEECS), Bhopal, India, 2023, pp. 1-6, 10.1109/SCEECS57921.2023.10063128. keywords: Measurement;Deep learning;Computer science;Deepfakes;Computerarchitecture;Transformers;Recording;Deepfake;Inception net;CNN(Convolutional Neural Network);Vision Transformers,

[13] H. H. Nguyen, J. Yamagishi and I. Echizen," Exploring Self-Supervised Vision Transformers for Deepfake Detection: A Comparative Analysis," 2024 IEEE International Joint Conference on Biometrics (IJCB), Buf falo, NY, USA, 2024, pp. 1-10, doi: 10.1109/IJCB62174.2024.10744497. keywords: Training;Deepfakes;Computer vision;Limiting;Semantic segmentation;Neural networks;Training learning;Feature extraction;Transformers, data;Self-supervised

[14] T. Vignesh, P. H. Tarun, R. Parthav and V. Bhargavi," Deepfake Face Detection using Machine Learning with LSTM," 2024 10th International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 2024, pp. 1633 1638, doi: 10.1109/ICCSP60870.2024.10544299. Deepfakes;Visualization;Convolution;Transfer keywords: learning;Neural networks;Motion pictures;Data models;Deepfake detection;Long Short-Term Memory (LSTM);Kaggle;Residual next convolution neural network (Xception CNN); Image manipulation,

[15] Karandikar, A., Deshpande, V., Singh, S., Nagbhidkar, S., & Agrawal, S. (2020). Deepfake video detection using convolutional neural network. International Journal of Advanced Trends in Computer Science and Engineering, 9(2), 1311–1315. https://doi.org/10.30534/ijatcse/2020/62922020

[16] Manish, B., Manish, C., & Reddy, B. S. K. (2024). Deepfake de tection on face images & videos using deep learning. International Journal of Creative Research Thoughts (IJCRT), 12(5), 753–766. http://www.ijcrt.org/

[17] Shah, A., Thakur, A., Kale, A., Bothara, H., & Pardeshi, D. C. (2024). Review paper on deepfake video detection using neu ral networks. International Journal of Advanced Research in Sci ence, Communication and Technology (IJARSCT), 4(1), 140–143. https://doi.org/10.48175/IJARSCT-16924

[18] Wodajo2021DeepfakeVD,Deepfake Video Detection Using Convolutional Vision Transformer, Deressa Wodajo and Solomon Atnafu,ArXiv,2021,abs/2102.11126, https://api.semanticscholar.org/CorpusID:231986176