Real-Time Deepfake Detection Platform with Advanced ResNet and Temporal Analysis

Mrs. G. Sathee Lakshmi, Assistant Professor, Department of CSE Gayatri Vidya Parishad College of Engineering (A)

*B. S. S. Dvijesh, **D. Sai Srujana, **Ch. Avinash, **A. Gowtham Kumar

Department of Computer Science Engineering, GVPCE(A), Visakhapatnam

Abstract:
This project introduces a deepfake detection system, a web-based application designed to enhance the identification of AI-generated media. By leveraging deep learning techniques, the system analyzes video frames to detect subtle inconsistencies that indicate deepfake manipulation. The application integrates a pre-trained ResNext convolutional neural network (CNN) for feature extraction and a long short-term memory (LSTM) network for temporal sequence processing, ensuring accurate classification of video clips. With a user-friendly interface, the platform provides real-time classification results along with confidence scores, making deepfake detection accessible and efficient. Additionally, the system contributes to the ongoing efforts to combat misinformation by offering a robust and scalable solution for media authentication.

Keywords*: Deepfake Detection, Machine Learning, Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM), ResNext Model*

Introduction:

With the rapid advancement of artificial intelligence, deepfake technology has emerged as a serious threat to digital authenticity. Deepfakes, created using generative models like GANs[1]. and autoencoders, can manipulate video and audio to produce hyper-realistic but entirely fake content[2]. While this technology has positive applications in media and entertainment, it also poses significant risks, including political misinformation[3], identity theft[4], and cybercrimes[5].

To address this challenge, we propose a deep learning-based solution for detecting deepfake videos. Our method leverages a combination of ResNeXt[6]-based convolutional neural networks (CNNs) for frame-level feature extraction and Long Short-Term Memory (LSTM)[7] networks. This approach enables our model to distinguish between real and AI-generated videos effectively. Additionally, a user-friendly interface allows individuals to upload videos for verification, ensuring accessibility and ease of use.

Aim and Scope:

Aim

The aim of this project is to develop an AI-powered deepfake detection system that can accurately classify videos as real or fake using deep learning techniques. The model leverages a hybrid CNN-LSTM architecture, where ResNeXt-50 is used for feature extraction and LSTM for temporal sequence analysis. By ensuring high accuracy in detecting manipulated content, this system aims to contribute to combating misinformation, digital fraud, and media-based deception in real-time applications.

Scope

The scope of this project extends across various fields where deepfake detection is essential for ensuring digital security, media authenticity, and trust in online content. With the rapid advancement of AI-based deepfake generation techniques, manipulated videos are being used for misinformation, identity fraud, and cybercrimes. This system provides an AI-driven solution that can effectively identify deepfake videos by analyzing both spatial inconsistencies using ResNeXt-50 and temporal anomalies using LSTMs. By integrating deepfake detection into media platforms, law enforcement, and cybersecurity systems, this technology can help mitigate risks associated with manipulated videos and prevent the spread of deceptive content.

In addition to news verification and fraud prevention, this system can be utilized in forensic investigations, corporate security, and social media regulation. Authorities can use it to authenticate video evidence, while businesses can integrate it into their security infrastructure to prevent impersonation scams. The project can further be extended to detect full-body deepfakes, AI-generated speech manipulations, and real-time deepfake detection in live streams. Future implementations may include browser extensions, mobile applications, and enterprise-level API services to make deepfake detection more accessible and scalable. By continuously adapting to emerging deepfake techniques, this system contributes to ensuring the integrity of digital media and combating AI-driven misinformation.

Literature Survey:

The study "No One Can Escape[8]" introduces a novel approach for detecting forged images by leveraging edge-based feature extraction combined with deep learning. The method employs the Scharr operator to extract edge information, which is then transformed into a Gray-Level Co-occurrence Matrix (GLCM) for feature representation. By using GLCM, the study captures texture and spatial relationships within the image, which are then processed by a deep neural network to classify images as either real or forged. This approach demonstrates the effectiveness of combining edge detection with deep learning, as it enhances the model's ability to identify fine-grained inconsistencies in manipulated images. However, this method is primarily focused on static image forensics, making it less suitable for video-based deepfake detection, where temporal inconsistencies play a crucial role.

Similarly, the "Face Warping Artifacts[9]" study presents an approach for detecting face-warping inconsistencies in deepfake videos. The method identifies differences between AI-generated face areas and their surrounding regions using a Convolutional Neural Network (CNN) model. Since many deepfake techniques rely on blending manipulated face regions into the original frame, this approach effectively detects spatial artifacts in deepfake images. However, a significant limitation of this study is that it does not consider temporal analysis, which is essential for detecting inconsistencies in deepfake videos over time. As deepfake generation techniques continue to evolve, spatial artifacts alone may not be sufficient for robust detection, making temporal-based approaches more effective for video analysis.

Another notable technique, "Detection by Eye Blinking[10]," introduces an innovative approach to deepfake video detection by analyzing eye-blinking patterns. The method utilizes a Long-term Recurrent Convolutional Network (LRCN) to model eye movement behavior over time. Since deepfake algorithms often fail to generate natural eye blinking, this method identifies abnormal blinking rates as an indicator of forgery. However, with advancements in deepfake technology, newer models have improved their ability to generate more realistic eye blinks, reducing the effectiveness of this approach. To address this limitation, researchers suggest incorporating additional facial clues, such as teeth misalignment, unnatural wrinkle distribution, and incorrect eyebrow placement, to enhance detection accuracy.

The study "Deepfake Video Detection Using Recurrent Neural Networks[11]" further explores video-based deepfake detection by employing Recurrent Neural Networks (RNNs) to capture temporal inconsistencies across consecutive video frames. By modeling frame-by-frame relationships, RNNs can detect subtle motion artifacts that are often present in deepfake videos. However, a major challenge faced by this approach is the limited diversity of datasets used for training, which affects the model's ability to generalize to real-world deepfake variations. To improve the robustness of RNN-based deepfake detection, larger and more diverse datasets are required to ensure the model can handle different lighting conditions, camera angles, and video qualities.

Proposed System:

To combat the growing threat of deepfake videos, we propose a deep learning-based detection system that effectively analyzes both spatial and temporal inconsistencies in manipulated media. Our approach integrates a hybrid Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) architecture, leveraging ResNeXt-50 for feature extraction and Long Short-Term Memory (LSTM) networks for sequence analysis. This combination allows the model to capture both frame-level details and temporal inconsistencies, ensuring higher accuracy in detecting deepfakes. The system follows a structured pipeline involving data collection, preprocessing, feature extraction, classification, and deployment to ensure robustness and scalability.

The dataset for training and evaluation is curated from FaceForensics++[12] (FF), Deepfake Detection Challenge (DFDC)[13], and Celeb-DF[14], ensuring diversity and better generalization. To prevent model bias, the dataset is carefully balanced, containing 50% real and 50% fake videos. During preprocessing, frames are extracted from videos at 30 FPS, and only the face regions[15] are retained to focus on critical areas where deepfake artifacts are most apparent. These face crops[16][17] are resized to 112×112 pixels for uniformity. Additionally, audio-altered videos are removed to ensure that the system focuses solely on visual manipulations, and each video is limited to 150 frames to optimize computational efficiency while preserving essential temporal information.

For feature extraction, the ResNeXt-50 model is used to extract 2048-dimensional feature vectors from each video frame. This CNN architecture is optimized for deep feature learning and can detect fine-grained inconsistencies such as pixel blending errors, unnatural textures, and lighting mismatches commonly found in deepfake videos. To analyze the sequential nature of video data, these extracted features are passed into an LSTM network, which detects motion inconsistencies, unnatural facial expressions, blinking irregularities, and frame-to-frame distortions that are indicative of deepfake content.

The model is trained using cross-entropy loss, optimized with the Adam[18] optimizer (learning rate = 1e-5, weight decay = 1e-3), and fine-tuned to enhance accuracy. A batch size of 4 is used due to GPU memory constraints, and training is conducted for 20 epochs to ensure sufficient learning. To prevent overfitting, the model incorporates Leaky ReLU[19] activation, adaptive average pooling, and a SoftMax[20] classifier for final predictions. These elements help improve feature extraction, optimize classification, and enhance the interpretability of the model's decision-making process.

For real-world usability, the system is integrated into a Django-based web application, providing a user-friendly interface where users can upload videos for analysis. Once a video is uploaded, the system detects and extracts face regions, processes them using the trained ResNeXt-LSTM model, and outputs a classification result indicating whether the video is real or deepfake, along with a confidence score. This real-time processing capability ensures the system is applicable to media authentication, forensic analysis, and social media verification.

The proposed solution stands out due to its hybrid approach that combines CNN-based spatial analysis with RNN-based temporal analysis, achieving superior performance compared to traditional frame-wise deepfake detection methods. Additionally, training on multiple datasets ensures the model can generalize well across different deepfake generation techniques. The system is designed for scalability, with potential future applications including browser extensions, mobile apps, and cloud-based APIs for large-scale deployment. By integrating advanced deep learning techniques, sequence modeling, and real-time inference, this solution provides an effective, accurate, and scalable method for deepfake detection, addressing critical challenges in digital security, misinformation control, and AI-driven media verification.

Data Sets:

The FaceForensics++ (FF++)[12], Deepfake Detection Challenge (DFDC)[13], and Celeb-DF[14] datasets are among the most widely used resources for training and evaluating deepfake detection models. FaceForensics++ is a large-scale dataset that provides both real and manipulated videos created using multiple face-swapping techniques, including DeepFakes (DF), Face2Face (F2F), FaceSwap (FS), and NeuralTextures (NT). It offers videos in different compression levels to simulate real-world scenarios, making it an essential dataset for training robust deepfake detection systems. With over 1,000 original videos and their corresponding manipulated versions, FaceForensics++ serves as a benchmark dataset for deepfake research.

The Deepfake Detection Challenge (DFDC) dataset, developed by Facebook AI, contains a vast collection of deepfake videos generated using multiple AI-based techniques, including some with altered audio. It is designed to reflect real-world challenges, incorporating diverse backgrounds, lighting conditions, and individuals, making it highly suitable for training models to generalize across different deepfake creation methods. Meanwhile, Celeb-DF focuses on high-quality deepfake synthesis, reducing visible artifacts to mimic realistic video manipulations. It includes varied lighting conditions, multiple facial expressions, and diverse celebrities, making detection more challenging. Unlike other datasets, Celeb-DF videos often retain original audio, though some may have mismatched lip-syncing. Together, these three datasets provide a comprehensive training foundation, ensuring deepfake detection models can handle different manipulation techniques, compression levels, and real-world variations effectively.

Models:

The proposed deepfake detection system utilizes a hybrid deep learning model that combines ResNeXt-50 for spatial feature extraction and LSTM (Long Short-Term Memory) networks for temporal sequence analysis. The model is designed to effectively capture frame-wise inconsistencies as well as motion anomalies across video sequences, ensuring a high level of accuracy in classifying real and deepfake videos.

ResNeXt[6] for Feature Extraction

To extract spatial features from individual video frames, the model employs a pre-trained ResNeXt-50 (resnext50_32x4d), a residual CNN architecture optimized for deep learning applications. ResNeXt-50 consists of 50 layers with a 32 × 4-dimensional cardinality, enhancing its ability to learn hierarchical patterns from complex datasets. The final pooling layers of ResNeXt generate 2048-dimensional feature vectors that serve as inputs for the LSTM network. These extracted features highlight pixel inconsistencies and unnatural blending artifacts that are commonly found in deepfake videos.

stage	output	ResNeXt-50 (32×4d)
conv1	112×112	7×7, 64, stride 2
conv2	56×56	3×3 max pool, stride 2 [1×1, 128 3×3, 128, C=32 1×1, 256] ×3
conv3	28×28	[1×1, 256 3×3, 256, C=32 1×1, 512] ×4
conv4	14×14	[1×1, 512 3×3, 512, C=32 1×1, 1024] ×6
conv5	7×7	[1×1, 1024 3×3, 1024, C=32 1×1, 2048] ×3
	1×1	global average pool 1000-d fc, softmax
# params.		25.0×10^6

Fig. 1: ResNext Architecture

LSTM[7] for Temporal Processing

The extracted 2048-dimensional feature vectors are sequentially fed into an LSTM layer with 2048 hidden units and a dropout rate of 0.4 to enhance generalization. Unlike CNNs, which focus on spatial relationships within a single frame, LSTMs capture temporal dependencies by comparing the frame at time t with earlier frames at t-n. This allows the model to identify inconsistencies in facial movements, unnatural blinking patterns, and abrupt transitions, which are key indicators of deepfake videos.

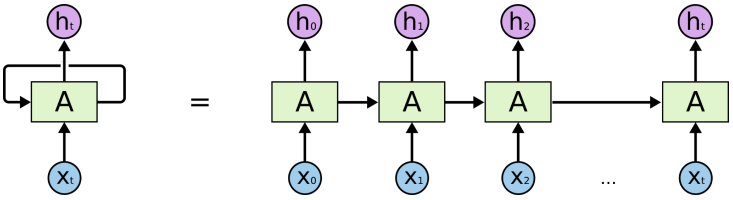

Fig. 2: LSTM Working

Sequential Layer

A Sequential Layer is utilized to store the feature vectors generated by ResNeXt-50 in an ordered sequence. This ensures that the frames are processed sequentially before being passed to the LSTM network. By maintaining the original temporal order of frames, the model effectively detects motion inconsistencies in deepfake videos.

Activation Functions

ReLU Activation Function[19]

The Rectified Linear Unit (ReLU) activation function is used throughout the model to introduce non-linearity while preventing the vanishing gradient problem. ReLU outputs 0 for negative inputs and returns the input itself for positive values, mimicking biological neuron behavior. This enables the model to learn complex patterns efficiently, improving training speed and convergence, especially in deep networks.

SoftMax Activation Function[20]

The SoftMax function is applied in the final classification layer to convert raw output logits into a probability distribution, assigning values between 0 and 1 for each class. The sum of all outputs equals 1, allowing the model to generate confidence scores for classification decisions. This function is essential for interpreting model predictions, ensuring that deepfake videos are classified with an associated level of certainty.

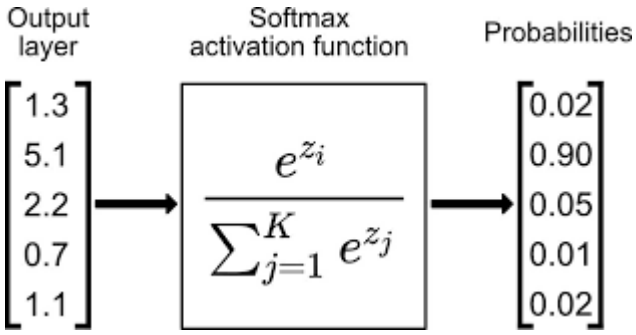

Fig. 3: Softmax activation function

Dropout Layer for Overfitting Prevention[21]

A Dropout layer with a probability of 0.4 is incorporated to reduce overfitting by randomly deactivating 40% of neurons during training. This forces the model to learn more generalized and robust patterns instead of memorizing specific features, leading to improved performance on unseen data. Dropout also modifies weight updates during backpropagation, making the model more stable for real-world deepfake detection scenarios.

Adaptive Average Pooling Layer

To optimize feature extraction and reduce computational complexity, the model includes a 2D Adaptive Average Pooling Layer. This layer adjusts the pooling window size dynamically to retain essential spatial information while reducing variance. By aggregating features from different parts of the frame, it ensures that low-level and high-level feature representations are efficiently captured, improving the model's overall robustness in deepfake detection.

Flow Diagram:

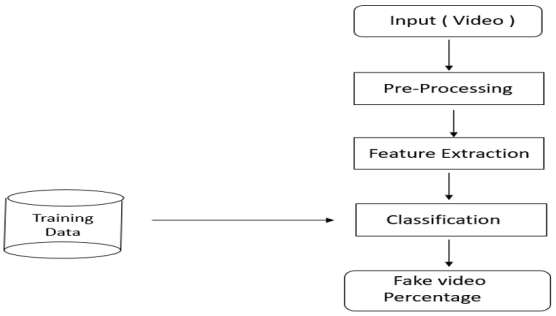

Fig. 4: Predicting user input video

The deepfake detection system begins with the user providing a video as input, which undergoes a pre-processing stage where frames are extracted at a fixed rate, focusing only on faces to eliminate unnecessary background information. These frames are then passed through ResNeXt-50, a powerful convolutional neural network (CNN), to extract meaningful spatial features that highlight pixel inconsistencies and manipulation artifacts. The extracted feature vectors are sequentially fed into a Long Short-Term Memory (LSTM) network, which analyzes temporal dependencies between frames to detect motion inconsistencies and unnatural transitions. The model then computes a fake video probability score, determining how likely the video is deepfake. If the score exceeds a predefined threshold (e.g., 50%), the video is classified as fake, and the final result is displayed to the user through a user-friendly interface, ensuring transparency and ease of interpretation.

Performance Measure:

A confusion matrix is a performance evaluation tool that provides a detailed summary of a classification model's predictions. It presents the count of correct and incorrect predictions, categorized by each class, offering valuable insights into the model's performance. In the context of Deepfake Video Detection, the confusion matrix helps identify instances where the model misclassifies real and fake videos, highlighting areas where it struggles. Beyond measuring accuracy, it also reveals the types of errors made, such as false positives and false negatives, allowing for targeted improvements. By analyzing the confusion matrix, we can assess model reliability, fine-tune parameters, and enhance deepfake detection accuracy effectively.

Outputs:

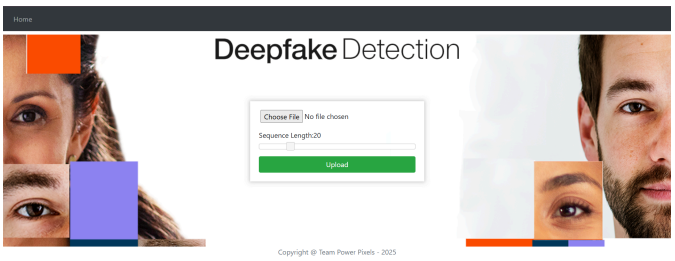

Fig. 5: Home page

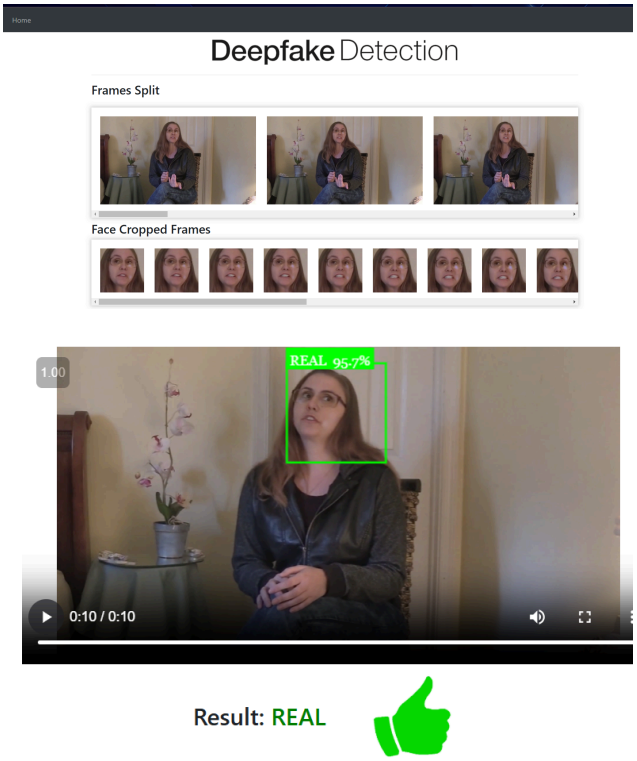

Fig. 6: Prediction for real video

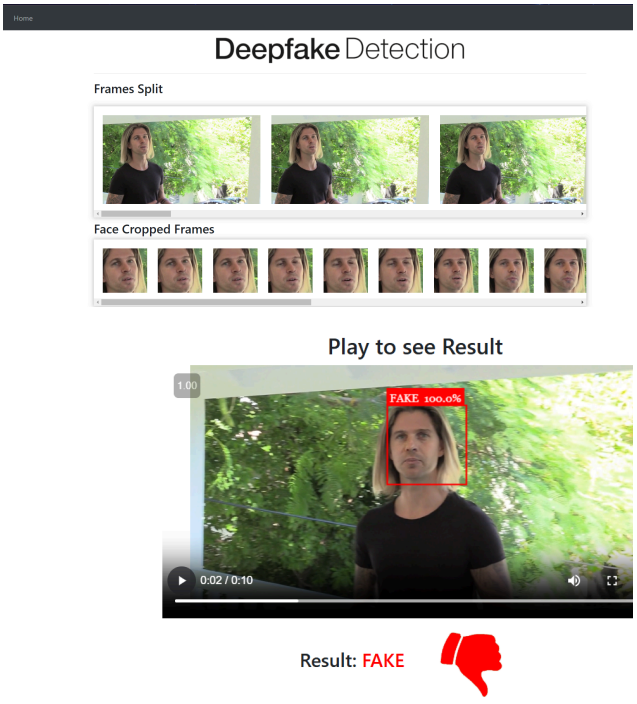

Fig. 7: Prediction for fake video

Conclusion:

In this project, we developed a deep learning-based framework for detecting deepfake videos by classifying them as either real or fake, along with a confidence score. Our approach integrates Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), where ResNeXt-50 extracts spatial features from individual frames, and an LSTM network analyzes temporal inconsistencies across sequences. This hybrid methodology enhances detection accuracy by leveraging both frame-level and motion-based inconsistencies, making it more effective than traditional single-frame analysis techniques.

As AI-generated deepfake content becomes increasingly sophisticated, deepfake detection has become a critical tool for media authentication, cybersecurity, and social media platforms. Our model provides a scalable and reliable solution to mitigate the risks posed by manipulated videos and combat the spread of misinformation. Furthermore, this study contributes to ongoing deepfake detection research by demonstrating the advantages of combining spatial and temporal feature analysis. By capturing both local inconsistencies within frames and global inconsistencies across sequences, our approach improves model robustness and enhances its ability to detect even highly realistic deepfakes.

Acknowledgement:

We express our heartfelt gratitude to Gayatri Vidya Parishad College of Engineering (Autonomous) for providing a conducive environment for research and study. We sincerely thank my guide, Dr. G. Sathee Laxmi, and my coordinator, Dr. Ch. Sita Kumari, for their invaluable guidance, continuous support, and insightful feedback throughout this study. Her encouragement and expertise have been instrumental in the successful completion of this research.

References:

[1] G. Antipov, M. Baccouche, and J.-L. Dugelay. Face aging with conditional gen erative adversarial networks. arXiv:1702.01983, Feb. 2017

[2] Deepfake Video of Mark Zuckerberg Goes Viral on Eve of House A.I. Hearing : https://fortune.com/2019/06/12/deepfake-mark-zuckerberg/ Accessed on23 February, 2025

[3] 10 deepfake examples that terrified and amused the internet : https://www.creativebloq.com/features/deepfake-examples Accessed on23 February, 2025

[4]Deepfakes, Revenge Porn, And The Impact On Women : https://www.forbes.com/sites/chenxiwang/2019/11/01/deepfakes-revenge-porn-and the-impact-on-women/

[5]The rise of the deepfake and the threat to democracy : GHRCEM-Wagholi,Pune, Department of Computer Engineering 2019-2020 51 Deepfake Video Detection https://www.theguardian.com/technology/ng-interactive/2019/jun/22/the-rise-of the-deepfake-and-the-threat-to-democracy(Accessed on23 February, 2025)

[6] ResNext Model : https://pytorch.org/hub/pytorch_vision_resnext/ accessed on 06 April 2020

[7] Understanding LSTM Networks : https://colah.github.io/posts/2015-08-Understanding-LSTMs/

[8] KEJUN ZHANG,YULIANG, JIANYI ZHANG, AND XINXIN LI1, “No One Can Escape: A General Approach to Detect Tampered and Generated Image” , No One Can Escape: A General Approach to Detect Tampered and Generated Image, August-September 2019, DOI:

10.1109/ACCESS.2019.2939812
https://ieeexplore.ieee.org/document/8826269
[9] Yuezun Li, Siwei Lyu, “ExposingDF Videos By Detecting Face Warping Arti facts,” in arXiv:1811.00656v3.
[10] Yuezun Li, Ming-Ching Chang and Siwei Lyu “Exposing AI Created Fake Videos by Detecting Eye Blinking” in arXiv:1806.02877v2.
[11] D. Güera and E. J. Delp, "Deepfake Video Detection Using Recurrent Neural Networks," 2018 15th IEEE International Conference on Advanced Video and Sig nal Based Surveillance (AVSS), Auckland, New Zealand, 2018, pp. 1-6.
[12] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies,Matthias Nießner, “FaceForensics++: Learning to Detect Manipulated Facial Images” in arXiv:1901.08971.
[13] Deepfake detection challenge dataset : https://www.kaggle.com/c/deepfake-detection challenge/data Accessed on23 February, 2025
[14] Yuezun Li , Xin Yang , Pu Sun , Honggang Qi and Siwei Lyu “Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics” in arXiv:1909.12962
[15] J. Thies et al. Face2Face: Real-time face capture and reenactment of rgb videos. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2387–2395, June 2016. Las Vegas, NV.
[16] Face app: https://www.faceapp.com/ (Accessed on 23 February, 2025)
[17] Face Swap : https://faceswaponline.com/ (Accessed on 23 February, 2025)
[18] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. arXiv:1412.6980, Dec. 2014.
[19]An Introduction to the ReLU Activation Function - https://builtin.com/machine-learning/relu-activation-function (Accessed on 23 February, 2025)
[20]Softmax Activation Function: Everything You Need to Know - https://www.pinecone.io/learn/softmax-activation/ (Accessed on 23 February, 2025)
[21] Dropout layer - http://keras.io/api/layers/regularization_layers/dropout/ (Accessed on 23 February, 2025)