A Dual-Stage Approach to Deepfake Video Detection Employing ResNet and LSTM Networks

Nithish Kumar S.
Student
BMS College of Engineering Bengaluru

Akhila S.
Professor
BMS College of Engineering Bengaluru

ABSTRACT
Deepfake boom has emerged as greatest multimedia information authenticity threats. In this paper, in anticipation of this issue, we propose an end-to-end detection synergistically merged Residual Networks (ResNet) for spatial feature learning and a combination of Long Short- Term Memory (LSTM) and Convolutional Neural Network (CNN) for temporal sequence modeling. ResNet module effectively outputs rich facial and contextual data from one frame, and Long Short-Term Memory- Convolutional Neural Networks (LSTM-CNN) module tracks temporal dynamics to capture unusual facial movements and expressions between two frames. For enhancing the model's ability to generalize, we utilize transfer learning practices such as large dataset pre-training and fine-tuning on deepfake-specialized datasets. Experimental tests conducted on certain deepfake datasets validate the enhanced performance of the introduced framework based on accuracy, precision, and recall in comparison to other dominant state-of-the-art methods. The result validates the robustness of the framework and its applicability in real scenarios, which largely contributes to multimedia forensics as well as the fight against false digital propaganda.

General Terms
Artificial Intelligence, Generative Adversarial Networks

Keywords
Deepfake, Residual Networks, Long Short-Term Memory, Convolutional Neural Network

1. INTRODUCTION
The rapid advancement of artificial intelligence (AI) has made it possible to produce realistic synthetic media, or deepfakes. These AI-based images and videos create a multitude of challenges to the integrity and authenticity of online media, and problems arise in fields as diverse as media, politics, and cyber security. Conventional detection tools are usually not able to detect such advanced manipulations, more powerful and adaptable mechanisms need to be designed. In an attempt to counter this emerging menace, researchers have examined hybrid deep neural network structures that merge the modalities of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) towards more robust deepfake identification. A specific model of CNN such as ResNet, is effective at learning fine-grained spatial features from isolated images and detecting small changes introduced during the process of developing deepfakes. Conversely, LSTM networks are more effective in comprehension of temporal dependencies, enabling the analysis of sequential frame data to detect inconsistencies in motion and facial expressions over time. The combination of the above architectures enables end-to-end space and time attribute processing, enhancing detection and reliability of deepfake detection systems. Additionally, transfer learning techniques—i.e., pre-training models in large datasets and then fine-tuning them with deepfake-specific data—have proven effective in enhancing detection model generalizability. This paradigm allows models to counter disparate deepfake generation methods and databases, focusing on the adaptive needs of synthetic media assaults. This paper introduces a hybrid-based deepfake detecting system model with ResNet as the spatial feature extractor and LSTM network as the temporal analyzer. By integrating the above components and utilizing transfer learning techniques, the proposed model is expected to attain high accuracy and reliability in deepfake detection on different applications. Performance of the framework is tested on benchmarking datasets to prove the capability of the framework as a legitimate tool in pursuing the ongoing quest to preserve the integrity of digital content.

2. LITERATURE SURVEY
In [1], the authors introduce an Adaptive Manipulation Traces Extraction Network (AMTEN) capable of efficiently identifying manipulations traces in face images by trace extraction through manipulations. The model is constructed to adaptively highlight the discriminative traces produced by different manipulation processes, which are extremely small and hard to identify with traditional techniques.

Zhiqing Guo et.al., [2] discuss limitations of existing techniques in detecting deepfakes. They highlight the fact that existing techniques fail under adversarial conditions because of their dependence on static features. This dependency is exploited by the attackers through transferability techniques (black-box; in which attackers lack access to the model) or through iterative optimization (white-box; in which attackers have complete access to the model). The study highlights that even small, carefully crafted perturbations to deepfake images can lead to significant drops in detection accuracy. The paper emphasizes the importance of building more robust detection algorithms that can withstand adversarial manipulation. The author notes that current deepfake detectors may not be sufficiently reliable, especially in scenarios where adversaries actively attempt to bypass them.

Deepfake images generated by neural networks often exhibit artifacts in their frequency domain, specifically in the high-frequency components. These artifacts are less noticeable in the spatial domain but become more apparent when analysed through Fourier transforms and other frequency-based techniques. The study in [3] highlights how deepfake generation procedures that introduce unnatural frequency patterns can serve as distinguishing features for detection.

Frank et.al., [4], demonstrated that the images generated by CNNs contain unique artifacts which are simple to identify through basic methods, including pixel-level inconsistencies and patterns. The authors deduce that although present CNN-generated images are quite simple to identify, the fast advancement of generative models would culminate in producing increasingly better fakes that are unidentifiable. This

study is a warning and a benchmark, which calls for relentless change in detection technologies in a bid to stay one step ahead of new deepfake technology.

Sheng-Yu Wang's [5] study on deepfake detection presents a wide array of methodologies and innovations aimed at identifying synthetic media. Key contributions in this domain include the utilization of saturation-based indicators to identify images generated by Generative Adversarial Networks (GAN), as well as the utilization of transfer learning with CNN models to enhance detection models adaptability.

A study published [6] in 2024 explored a hybrid model combining CNN, LSTM, and Transformer architectures for video deepfake detection. Evaluated on datasets like VoxCeleb2, DFD, Celeb-DF, and FF++, the model achieved an AUC of 90.82% with a single LSTM layer, indicating the importance of balancing model complexity to prevent overfitting.

A comparative study by Pu et.al., [7] performed in 2024 compared some of the best existing deepfake detection algorithms and identified one that utilizes ResNeXt-50 and the LSTM layers. Several deepfake detection models have shown strong performance in recent studies. XceptionNet achieved 95% accuracy on the FaceForensics++ dataset, while MesoNet reported 84% on Deepfake TIMIT. Capsule-based networks reached up to 96.6% accuracy by capturing spatial inconsistencies. Attention-based models, like multi-attention networks, achieved 92.1% on Celeb-DF v2. A method combining EfficientNet-B4 with LSTM reported 93.8% on DFDC. In 2024, Pu et al. found that a hybrid ResNeXt-50 with LSTM model achieved 94.3% on Celeb-DF v2, showing the value of combining spatial and temporal features. This model gave a precision of nearly 83% on the DFDC dataset and outperformed many competing models in the test.

Saikia et al. [8] presented a hybrid CNN-LSTM model that leverages optical flow features to capture motion inconsistencies in videos. The approach achieved accuracies of 91.21% on FF++, 79.49% on Celeb-DF, and 66.26% on DFDC datasets, highlighting the effectiveness of incorporating temporal motion features. Shrivathsa et.al., [9] introduced a deepfake detection framework utilizing XResNet, an optimized version of ResNet, alongside LSTM networks. The model focused on extracting 128 facial landmarks to capture intricate spatial details, with an 83.3% accuracy on the DFDC dataset.

A number of studies [14], [15] provide probabilistic models to explain performance metrics like precision, recall, and F- score. Advanced models like very deep convolutional neural networks and Inception-v4 have been applied effectively for facial recognition tasks. Recent advancements have introduced techniques such as Neural Radiance Fields (NeRFs), GANverse3D, and pose-guided image generation, which enable the realistic rendering of human subjects in varying poses and perspectives. These methods are further utilized to construct detailed neural avatars by integrating 3D geometry, texture synthesis, and deep generative models. Research also highlights specialized applications, including the animation of static images using First Order Motion Models and the translation of hand gestures via Gesture GAN. A number of systematic reviews compile these developments, offering an in-depth analysis of the latest techniques and the ongoing challenges in the deepfake detection landscape.

3. METHODOLOGY

The proposed system utilizes a hybrid deepfake detection model that integrates Residual Networks (ResNet) for spatial feature extraction with a combined Long Short-Term Memory and Convolutional Neural Network (LSTM-CNN) structure for temporal analysis. The ResNet module processes individual video frames to learn detailed facial features and contextual patterns. To capture motion-related anomalies across frame sequences, the LSTM-CNN module analyzes temporal relationships, allowing the detection of unnatural changes in facial expressions and movements. To improve the model's adaptability across varied data, transfer learning techniques are applied—starting with pre-training on large- scale datasets, followed by fine-tuning using datasets specifically designed for deepfake detection. This method ensures the model captures both static visual cues and temporal inconsistencies essential for accurate classification.

To mitigate model bias, we trained our PyTorch based deepfake detection system using a balanced dataset comprising equal numbers of authentic(real) and manipulated(fake) videos. The architecture of the system is depicted in the accompanying figure. During this phase, we curated and pre-processed the dataset, focusing on extracting and utilizing face- cropped video segments for training.

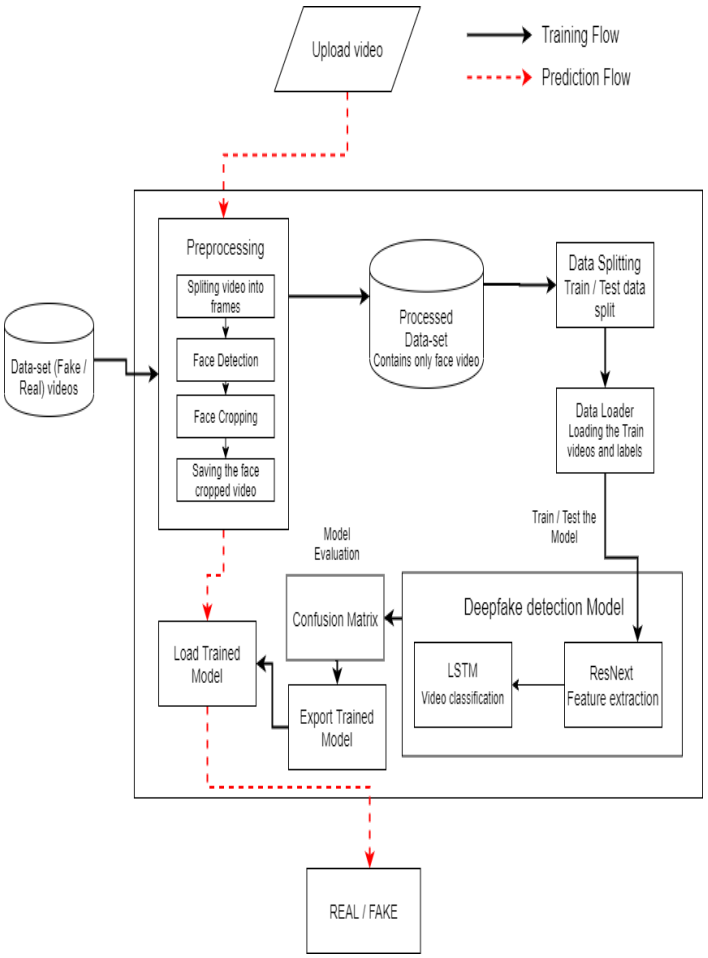

Fig 1. System Architecture

3.1 For creating deepfake videos

To effectively detect deepfake videos, it is crucial to comprehend the methodologies employed in their creation. Techniques such as autoencoders and Generative Adversarial Networks (GANs) are mainly used, where a target video and source image are provided as inputs. These procedures have the video divided into frames, locate the face region, and substitute the target face with the source face for each frame. The modified frames are then reassembled using various pre-trained models, which also enhance video quality by eliminating residual artifacts introduced during the deepfake generation process.

In our approach to deepfake detection, we adopt a similar methodology. Despite the high realism of deepfakes produced by pre-trained neural network models—making them nearly indistinguishable to the human eye—these synthetic videos often contain subtle artifacts or inconsistencies not easily perceptible without specialized analysis. The objective of this study is to identify these Imperceptible traces and distinguishable artifacts to accurately classify videos as either deepfake or authentic.

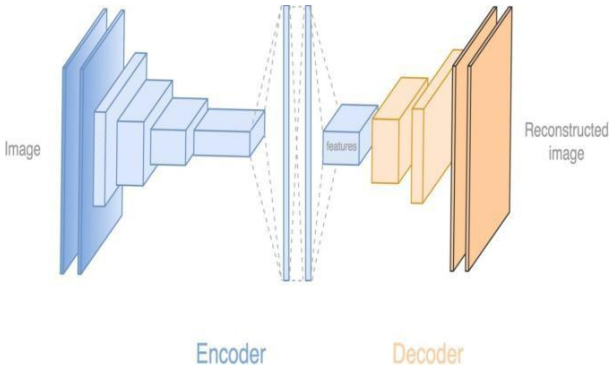

Fig 2. Deepfake Generation

3.2 Dataset Gathering

To optimize the model for real-time deepfake detection, a comprehensive dataset was assembled by integrating samples from various publicly available sources. This dataset was balanced, comprising equal numbers of authentic and manipulated videos, to mitigate potential training biases. Recognizing that certain datasets included videos with altered audio—elements beyond the scope of this study—a preprocessing step was implemented to exclude such instances, focusing solely on visual manipulations. The final curated dataset consisted of 6,000 videos, evenly split between real and deepfake content, thereby enhancing its capacity for generalization across diverse situations.

3.3 Pre – processing

To enhance the model's efficiency for real-time deepfake

detection, a comprehensive preprocessing pipeline was implemented. Initially, each video was decomposed into individual frames. Subsequently, facial regions were identified within these frames, and only the pertinent facial areas were retained, with extraneous content and noise being eliminated. Frames lacking detectable facial features were excluded from further processing. The extracted facial regions from each frame were then reassembled to form new video sequences, resulting in a curated dataset comprising face-centric videos.

In order to maintain consistency across the dataset and allow for the limitation of computational resources, a cutoff point was determined for frames per video. Based on these mean number of frames and limitations of available GPU resources, the first 150 frames from every video were taken. This method, as well as normalizing the input data, allows for proper utilization of Long Short-Term Memory (LSTM) networks by maintaining the temporal order of the frames. All videos were normalized to the frame rate of 30 frames per second and resolution of 112×112 pixels.

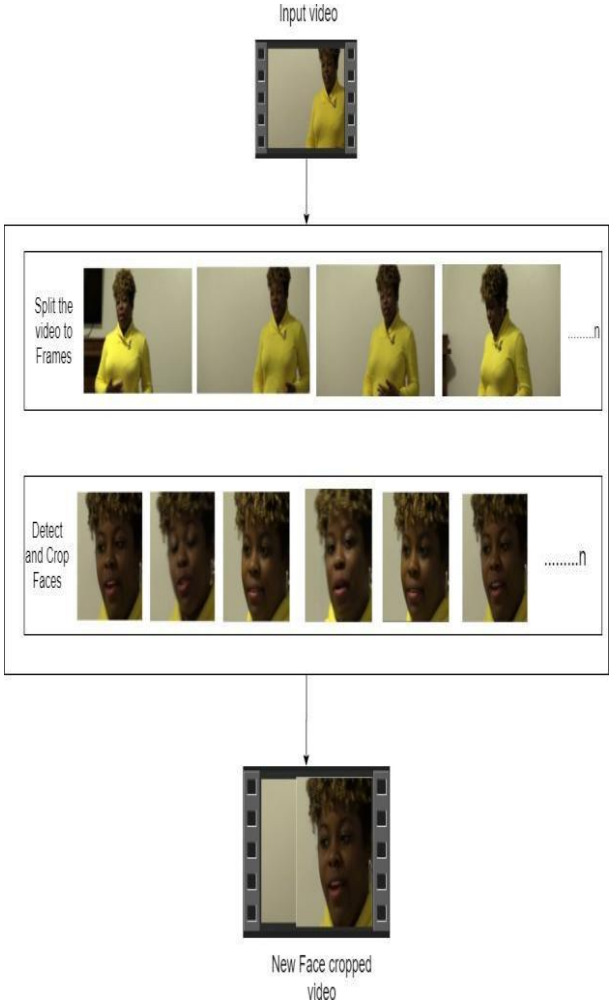

Fig 3. Pre – processing of video sample

3.4 Data - set splitting

To maintain appropriate training and testing of the deepfake detector model, the information was divided into test and train sets in 70:30 proportion. This type of splitting gave rise to 4,200 videos for training and 1,800 for testing. Both sets were kept balanced in a manner that both included equal proportions of real and manipulated videos and each set included 50% real and 50% deepfake content. Such balanced splitting is meant to avoid class imbalance issues and enhance the model's ability to generalize across other types of video content.

3.5 Model Architecture

The proposed deepfake detection architecture blends a Recurrent Neural Network (RNN) and a Convolutional Neural Network (CNN) to address both spatial and temporal elements found within video sequences. Specifically, it leverages a pre-trained ResNeXt- 50_32x4d model to perform detailed, frame-wise feature extraction, capitalizing on its deep residual structure, which is optimized for advanced image processing tasks. Each video frame is passed through this ResNeXt backbone, resulting in the generation of a 2,048- dimensional feature vector derived from the final pooling stage. After obtaining these vectors for each frame, the sequence is organized and then input into a LSTM (Long

Short-Term Memory) network, which can simulate the dynamics and sequential dependencies present in video data. In particular, a pre-trained ResNeXt-50_32x4d model is utilized for frame-level feature extraction, taking advantage of its residual architecture optimized for deep learning tasks. Each video frame is processed through ResNet, yielding a 2048-dimensional feature vector from the final pooling layer. These features vectors sorted are then passed into an effective network for learning temporal dependencies is the Long Short-Term Memory (LSTM) network between frames.

The LSTM block consists of a single 2048 hidden unit layer

with dropout 0.4 to avoid overfitting. It makes use of an activation function Leaky ReLU, or Leaky Rectified Linear Unit for enabling non-linear transformation. This is followed by the fully connected linear layer that converts 2048-dimensional LSTM outputs into a two-dimensional space to enable binary classification task. A single 1×1 adaptive averaging pooling is applied for feature map normalization, and a SoftMax in the last layer is used to provide class probabilities. Model training is performed with a batch size of 4 to reduce computational resources.

It well captures spatial complexities and temporal dynamics in video segments and hence enables the model to better differentiate between real and fake material.

3.6 Model Details

Residual Network (ResNet) - In our approach, we utilize the ResNeXt-50_32x4d model, a pre-trained residual convolutional neural network comprising 50 layers and characterized by a cardinality of 32 with a bottleneck width of 4. This architecture facilitates efficient feature extraction, serving as a foundational component in our deepfake detection framework.

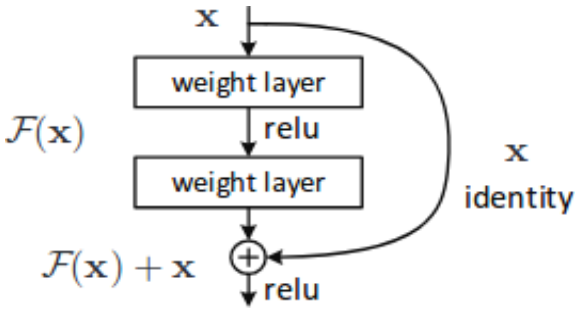

Fig 4. ResNet working

Sequential Layer - The Sequential model serves as a linear stack of layers, facilitating the orderly arrangement of modules. This configuration enables the systematic passage of feature vectors extracted by the ResNeXt model into the LSTM layer, thereby preserving the temporal dependencies inherent in the sequential data.

LSTM Layer - In our current architecture, a Long Short- Term Memory (LSTM) network compresses sequences of frames from videos and learns temporal dependencies and time variations in the process. Frames are considered as 2048-dimensional feature vectors, which are derived from the above said previous convolutional neural network, and are input into the LSTM layer one at a time. The LSTM structure is a single-layer network of 2048 units in the hidden layer and includes the inclusion to prevent overfitting, with a dropout rate of 0.4.

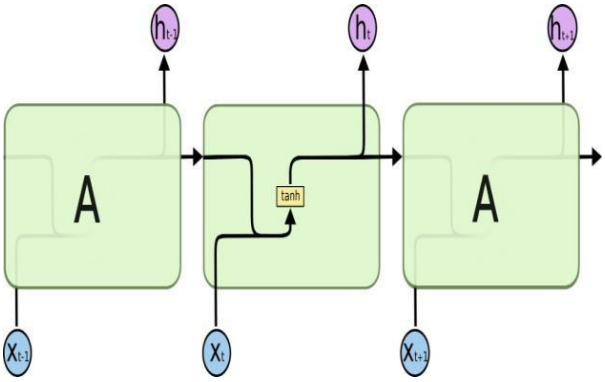

Fig 5. Internal LSTM Architecture

The configuration supports the capture of temporal behavior by comparing the current frame at time step t against previous frames at time steps t–n, where n is the number of past frames up to t. This kind of structure enables the network to learn and recognize temporal patterns in the video sequence well.

Rectified Linear Unit (ReLU) – The ReLU activation function is given by f(x)=max (0, z), producing zero for negative inputs and a linear identity for positive inputs. This non-linear function introduces sparsity in neural networks by activating only a subset of neurons, thereby enhancing computational efficiency and mitigating the vanishing gradient problem commonly associated with sigmoid functions. ReLU's simplicity and computational efficiency make it particularly advantageous for training deep neural networks, as it accelerates convergence during backpropagation without the need for complex exponential calculations. These properties have led to its widespread adoption in various deep learning applications, including computer vision and natural language processing.

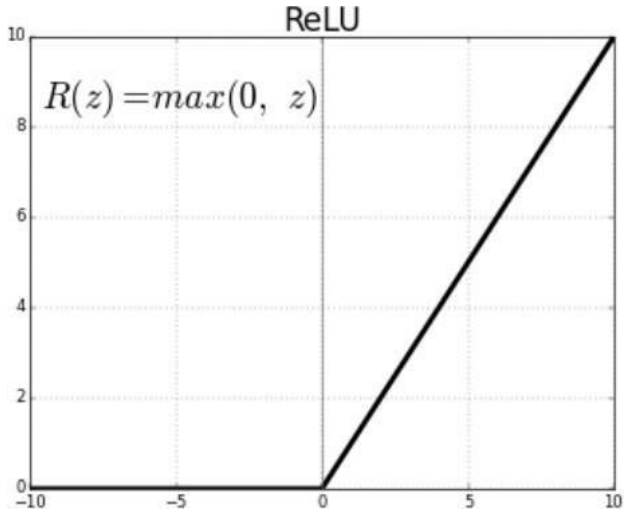

Fig 6. ReLU Activation Function

Dropout Layer - In the proposed neural network architecture, a dropout layer of 0.4 dropout rate is added to avoid overfitting and enhance the model's generalization ability. In training, this method randomly shuts down some of the neurons by effectively setting their output to zero. This stochastic deactivation avoids over- reliance by neurons on certain features and limits complex co- adaptation among neurons, but encourages learning more stable and generalized features.

This incorporation of dropout also affects the process of backpropagation. By randomly dropping out some neurons during training, the network imposes an effect of noise that forces spreading of weight updates among the active neurons. This generalizes and balances the learning process, as the model cannot rely on any specific neuron and has to spread the learning across various paths.

In summary, the use of the 0.4 operating rate dropout layer at a rate of 0.4 is a good and efficient regularization technique method that helps the model generalize to unseen data and with well-trained dynamics.

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 7. ResNet Architecture

Adaptive Average Pooling Layer - The adaptive avera ge pooling layer is added to the model in order to normalize the output dimension in order to save computational complexity and suppress variance between feature maps. That is, a two-dimensional adaptive average pooling is used, which splits the input into sub-regions and then calculates the average along each of the sub-regions, so that discriminative low-level features can be extracted from localized neighborhoods.

Model Training Details - The database was split into the test and training sets in the proportion of 70:30, comprised 4,200 and 1,800 videos, respectively. There was an equal proportion of fake and real videos in each subset for proper representation.

A data loader was used to facilitate the loading of video data and their respective labels with an optimal batch size of four. It is trained for 20 epochs with a learning rate of 1e-5 and a weight decay parameter of 1e-3 with the Adam algorithm optimized.

Adaptive learning rate has been utilized by utilizing an adam optimizer feature to improve convergence in training.

For purposes of task classification, cross-entropy loss function was utilized to establish the difference between labelled data and estimated probabilities. The last layer of the network also had a SoftMax activation function that would transform raw output scores into normalized probabilities over the two classes—real and fake—producing a measure of confidence in all predictions. As an estimate of model precision, a confusion matrix was used to count true positives, false positives, true negatives, and false negatives. The matrix is a source of information regarding the model's precision for predictions and the type of any errors, thereby indicating possible areas for improvement. The classification task, the cross-entropy loss function was are used in a way that calculates the difference between the predicted and actual labels. The last layer utilized a SoftMax activation function, converting raw output scores into normalized probabilities across the two classes—real and fake—thus providing a measure of confidence for each prediction.

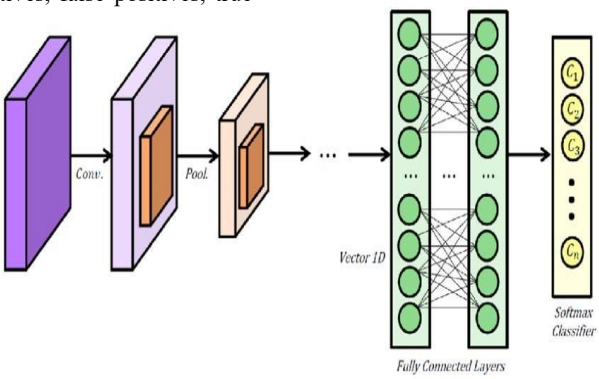

Fig 9. Softmax Layer

4. RESULTS

The figure 9, 10, 11, 12 shows the procedures to be chosen in depicting whether the video is a real or a fake.

The proposed deepfake detection framework delivered high performance across key metrics, including accuracy, precision, and recall, confirming its capability to reliably detect manipulated video content. Due to its strong generalization and detection capabilities, the model is well-suited for practical use in areas such as monitoring online media platforms, supporting forensic analysis, validating video-based legal evidence, securing identity verification processes, and countering the spread of misleading digital content.

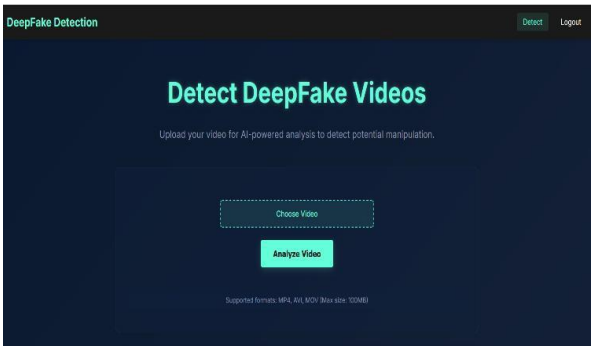

Fig. 9 Choose the video which needs to depict a video is a real or a fake

Fig.10 The analysed video will be giving an analysis result from the video analysed

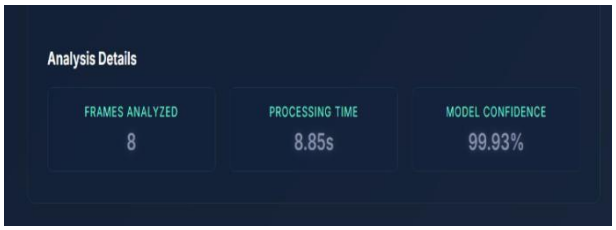

Fig. 11 The analysis result will be analyzed such as frames analyzed, processing time, and a model confidence

Fig. 12 The video will be anlayzed as the frames shown

From the above figure we can conclude that the obtained video detection using ResNet and LSTM approach has been employed with a confidence result of real detection by providing a 99.93%. The frame analysis that is of 8 frames has been analyzed with a processing time mentioned of 8.85 seconds and with the model confidence of 99.93%.

5. CONCLUSION AND FUTURE WORK

The study proposed a deepfake detection framework using ResNet and LSTM networks. The ResNet block is better suited for the extraction of fine-grained spatial information from one frame, and the LSTM layer preserves temporal relationships between the frames of the video. Evaluation on standard benchmark datasets indicates that the proposed architecture

significantly improves detection accuracy and robustness when compared to conventional CNN-based approaches. The inclusion of spatial and temporal data allows the model to detect subtle manipulations and inconsistencies typical of deepfake material.

Although the proposed method exhibits promising performance, multiple potential directions for future research remain open for exploration. Firstly, the model can be scaled up to bigger and more heterogeneous sets of authentic social media posts of different resolutions and different compression rates. Secondly, optimization methods like pruning, quantization, and distillation of knowledge may be employed to facilitate deployment onto resource- constrained devices. Thirdly, integration of Explainable AI (XAI) methods may introduce interpretability into the detection process, thus enhancing transparency and trust in the system. In addition, the model's adversarial robustness would have to be tested to improve its reliability in adversarial situations. Lastly, multi-modal techniques integrating visual, auditory, and behavioural information might be explored to enhance detection accuracy in more challenging situations.

6. REFERENCES

[1] N. M. Alnaim, Z. M. Almutairi, and M. S. Alsuwat, "DFFMD: A Deepfake Face Mask Dataset for Infectious Disease Era with Deepfake Detection Algorithms," 2023.

[2] Z. Guo, G. Yang, J. Chen, and X. Sun, "Fake face detection via adaptive manipulation traces extraction network," *Comput. Vis. Image Underst.*, vol. 204, 2021.

[3] N. Carlini and H. Farid, "Evading deep-fake-image detectors with white- and black-box attacks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops*, 2020.

[4] J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, and T. Holz, "Leveraging frequency analysis for deep fake image recognition," 2020.

[5] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, "CNN-generated images are surprisingly easy to spot...for now," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020.

[6] M. Masood, M. Nawaz, K. M. Malik, A. Javed, and A. Irtaza, "Deepfakes generation and detection: State-of-the-art, open challenges, countermeasures, and way forward," *arXiv preprint arXiv:2103.00484*, 2021.

[7] J. Pu, N. Mangaokar, L. Kelly, P. Bhattacharya, K. Sundaram, M. Javed, B. Wang, and B. Viswanath, "Deepfake videos in the wild: Analysis and detection," *arXiv preprint arXiv:2103.04263*, 2021.

[8] S. M. Zobaed, M. F. Rabby, M. I. Hossain, E. Hossain, S. Hasan, A. Karim, and K. M. Hasib, "DeepFakes: Detecting forged and synthetic media content using machine learning," *arXiv preprint arXiv:2109.02874*, 2021.

[9] S. Degadwala and V. M. Patel, "Advancements in deepfake detection: A review of emerging techniques and technologies," *ResearchGate*, 2024.

[10] Y. S. El-Din, M. N. Moustafa, and H. Mahdi, "Deep convolutional neural networks for face and iris presentation attack detection: Survey and case study," *IET Biometrics*, 2020.

[11] L. Lv *et al*., "Combining dynamic image and prediction ensemble for cross-domain face anti-spoofing," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2021.

[12] B. Zhang, B. Tondi, and M. Barni, "Attacking CNN-based anti-spoofing face authentication in the physical domain," in *Proc. Int. Conf. Multidiscip. Eng. Appl. Sci. (ICMEAS)*, Abuja, Nigeria, 2023, pp. 1–6.

[13] J. N. Kundu, N. Venkat, M. V. Rahul, and R. V. Babu, "Universal source-free domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020.

[14] U. A. Ciftci, I. Demir, and L. Yin, "FakeCatcher: Detection of synthetic portrait videos using biological signals," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jul. 15, 2020, doi: 10.1109/TPAMI.2020.3009287.

[15] K. Roy *et al*., "Bi-FPNFAS: Bi-directional feature pyramid network for pixel-wise face anti-spoofing by leveraging Fourier spectra," *Sensors*, vol. 21, no. 8, 2021.