IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Deepfake Video Detection Using Inception_ResNet_v2: A Convolutional Neural Network Approach

P. Priya nandini[1], Dr.M.Vikram[2]

[1]Research Scholar, Department of CSE, Sri Venkateswara College of Engineering, Tirupati

[2]Associate professor, Department of CSE, Sri Venkateswara College of Engineering, Tirupati

I. ABSTRACT:

II. Deepfake videos, produced by advanced artificial intelligence algorithms, present serious problems for the legitimacy and dependability of online audiovisual content. In this paper, we propose a deepfake video detection system based on the deep convolutional neural network known for its image classification task-winning Inception_ResNet_v2 architecture. A wide range of authentic and altered video clips make up the Deepfake Challenge Dataset, which is obtained from Kaggle and used to train and assess the system. To improve model generalization, we preprocess the dataset by removing frames from each movie and adding to them. After then, the Inception_ResNet_v2 model is refined by transfer learning, making use of ImageNet's pre-trained weights to hasten convergence and boost efficiency. We competition, as means of thorough testing and analysis, that our method effectively differentiates between real and deepfake videos. In terms of identifying deepfake films, the proposed strategy performs admirably, demonstrating its potential use in halting the spread of synthetic media and preserving the integrity of digital information. Our results highlight the need of utilizing cutting-edge machine learning methods to tackle the changing issues brought forth by deepfake technology.

III. **Keywords:**Deepfake, video detection, Inception_ResNet_v2, convolutional neural network, transfer learning, image classification.

1. INTRODUCTION:

The broad availability of strong editing tools and social media platforms has enabled the unprecedented transmission of multimedia material in the digital age. But despite all of this visual content, there is a rising lack of confidence in the veracity and authenticity of what we see online. The rise of deepfake technology, which uses sophisticated artificial intelligence algorithms to create incredibly realistic-looking but completely fake videos, is one particularly concerning example of this problem.

Deepfake films use deep learning algorithms to replace or superimpose other people's faces—often with stunning accuracy—over the faces of people in original footage. The possibility of false information, identity theft, and the decline in public confidence in digital media have all been brought up by this capability. Therefore, it is imperative to develop strong and trustworthy techniques to identify and stop the propagation of deepfake information on internet platforms.

The problem of deepfake detection has drawn the attention of computer vision [7] and machine learning researchers and practitioners in recent years, who have been actively investigating a number of different strategies. Convolutional neural networks (CNNs) [8], a family of deep learning models particularly suited for the analysis of visual input, are one well-known area of

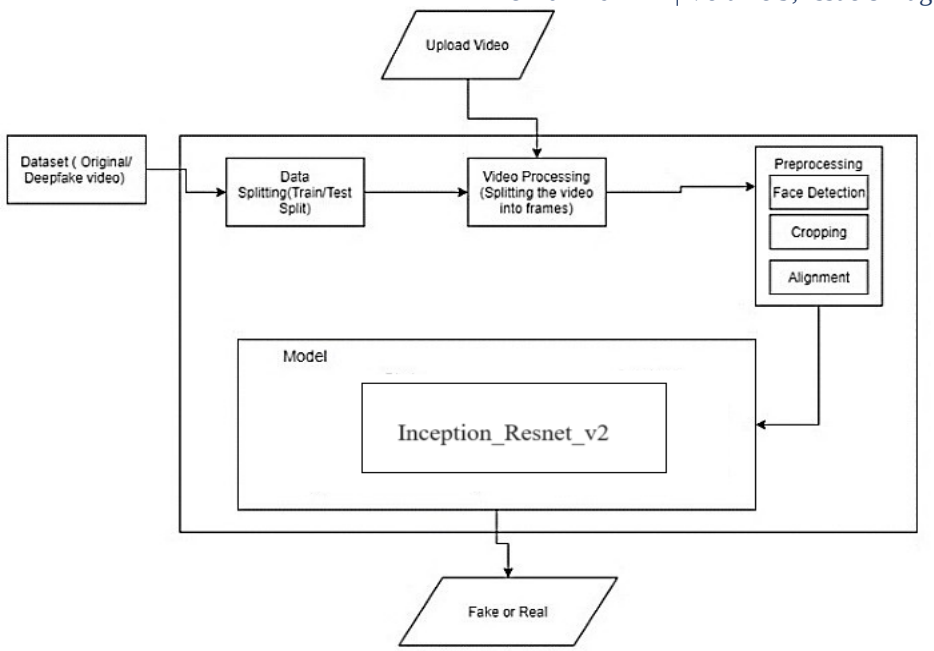
study. As CNNs have proven to be remarkably effective at tasks like object detection, facial recognition, and picture classification, they are an obvious candidate for identifying irregularities in video material.
The Inception_ResNet_v2 model is one of the CNN architectures that has drawn interest for deepfake detection [9]. Inception_ResNet_v2, which combines components of the Inception and ResNet architectures, provides an effective framework for learning hierarchical representations and extracting features. Its deep and complex design makes it especially good at identifying the minute artifacts that point to deepfake manipulation. It can also pick up on complex patterns and minute subtleties in visual data.
Using the Inception_ResNet_v2 architecture, we present a unique method for deepfake video detection in this paper [10]. Based on the ideas of transfer learning, we accelerate convergence and improve performance by using the pre-trained weights of Inception_ResNet_v2 on the ImageNet dataset. We experiment and evaluate extensively with the Deepfake Challenge Dataset, which is a large collection of authentic and modified video clips from Kaggle.
We enhance the dataset using preprocessing methods like frame extraction and augmentation to increase the generalization and resilience of the model. Using this expanded dataset, we optimize the Inception_ResNet_v2 model, improving its capacity to distinguish between real and deepfake films. By offering a workable and efficient way to combat the spread of synthetic media and protect the integrity of digital information on the internet, our method seeks to add to the expanding body of research on deepfake detection.

2. RELATED WORK:

There is a growing need for video analysis, identification, and intervention because to the serious concerns that the emergence of deepfake films poses to democracy, justice, and public confidence. In "Exposing DF Videos by Detecting Face Warping Artifacts," for example, one method uses a specialized Convolutional Neural Network (CNN) to recognize facial artifacts in created faces in comparison to their environment. This technique reveals shortcomings in the state-of-the-art deepfake algorithms, which frequently yield low-resolution images that require additional processing in order to match the original faces.
Another method, described in "Uncovering AI Created Fake Videos by Detecting Eye Blinking," is concerned with identifying physiological indicators such as the absence of eye blinking in deepfake footage. Although promising, this approach just takes into account the lack of blinking, which leaves opportunity for development by taking into account other factors like the look of the teeth.
As covered in [3], capsule networks provide a another method for identifying photos and movies that have been altered. Though initially successful, issues with noise in training data cast doubt on the system's suitability for real-time implementations. To tackle this, we present a novel approach that uses a real-time, noiseless dataset for training.
"Detection of Synthetic Portrait Videos" presents a novel method that achieves high accuracy in identifying synthetic portrait movies without depending on particular video properties. Instead, it makes use of biological signals. Nevertheless, developing differentiable loss functions in the absence of a discriminator presents difficulties.
Even in unsupervised situations, Durall et al.'s [5] use of frequency domain analysis and a straightforward classifier yields remarkable results. The success of their strategy is demonstrated by the perfect classification accuracy they are able to attain with a small number of labeled instances.
On the other hand, Johansson's work [6] employs LSTM and CNN architectures to recognize manipulated video information, including several deepfake approaches. Although efficient, it is not as good as the most sophisticated detectors.
Our research expands on previous investigations by putting forth a thorough deepfake detection system built on the Inception_ResNet_v2 architecture. By using preprocessing techniques and transfer learning to the Deepfake Challenge Dataset, we hope to improve model generalization and help fight the spread of synthetic media.

3. PROBLEMSTATEMENT:

In digital forensics, detecting deepfakes is a major difficulty since attackers may easily produce convincing fake movies by seamlessly superimposing one person's face onto another. Let's represent a real picture or video frame as I_real, its deepfake counterpart as I_fake, and the output of an image obtained by a facial recognition system as F(I). The challenge is to develop a trustworthy deep learning model M that, through analysis of the facial features extracted by F, can discern between I_real and I_fake. The objective is to create M in such a way that the patterns it finds in F(I_real) and F(I_fake) are distinctly different, enabling precise identification. .. The challenge is keeping up with the increasing complexity of deepfake methods, which calls for a strategy that combines sophisticated deep learning with facial recognition to improve the detection system's dependability and efficiency.



e

Figure1:SystemArchitecture

4. METHODOLOGY:

Here, we describe the process used to identify deepfake videos using the Inception_ResNet_v2 architecture, as illustrated in Figure 1. We present the transfer learning approach used to fine-tune the model, as well as the preprocessing steps employed to the Deepfake Challenge Dataset. By doing thorough testing and analysis, we demonstrate how well our method works to distinguish between real and fake video content, underscoring its possible importance in halting the spread of fake media.

4.1 Dataset

Deep Learning is mostly dependent on learning from data, which means that in order to achieve high learning quality and precise predictions, dataset preparation must be done with care. We use the Deep Fake Detection Challenge (DFDC) datasets [11] for our experiments, which include both real and fake video clips. The 400 movies in the training dataset include 400 testing videos, a CSV file, and a metadata.json file with 323 fictitious and 77 genuine videos. The dataset includes a range of settings, such as people facing or not facing the camera, standing or sitting, and with different backgrounds, lighting conditions, and video quality.

The training films are 1920 × 1080 pixels in horizontal mode, or 1080 × 1920 pixels in vertical mode. The dataset additionally includes 4000 private movies for testing, with a total of 119,146 labeled recordings in the training set and 400 videos in the validation set without labels. The Kaggle framework is used to evaluate these private test movies, and models are rated according to log-likelihood loss. Because of the significant penalties for false positives and false negatives, this loss function reflects doubts in the authenticity of the videos.

Figure 2: Sample dataset

More difficulties arise from the private test set since it simply offers binary labels without indicating the kind of alteration (e.g., face, audio, or both). Furthermore, videos may include multiple people with selectively altered faces, making detection even more difficult. In order to overcome this, our approach only considers video data, which could result in noisy labels when modifications are made just with audio. We use all 50 numbered segments of the labeled dataset in our training method. Of the dataset, 80% is used for training and 20% is used for testing. Our method seeks to alleviate the inherent problems presented by the features and assessment metrics of the dataset by means of meticulous preprocessing, model selection, and training procedures.

4.2 Preprocessing:

The first step in our pre-processing pipeline is to split the video into frames. After that, we use these frames for face recognition, removing the frames that have faces identified in them. We compute the mean number of frames in the dataset and build a new dataset with frames equal to this mean in order to guarantee consistency in the quantity of frames throughout the dataset. This method makes efficient model training possible while also assisting in maintaining uniformity.

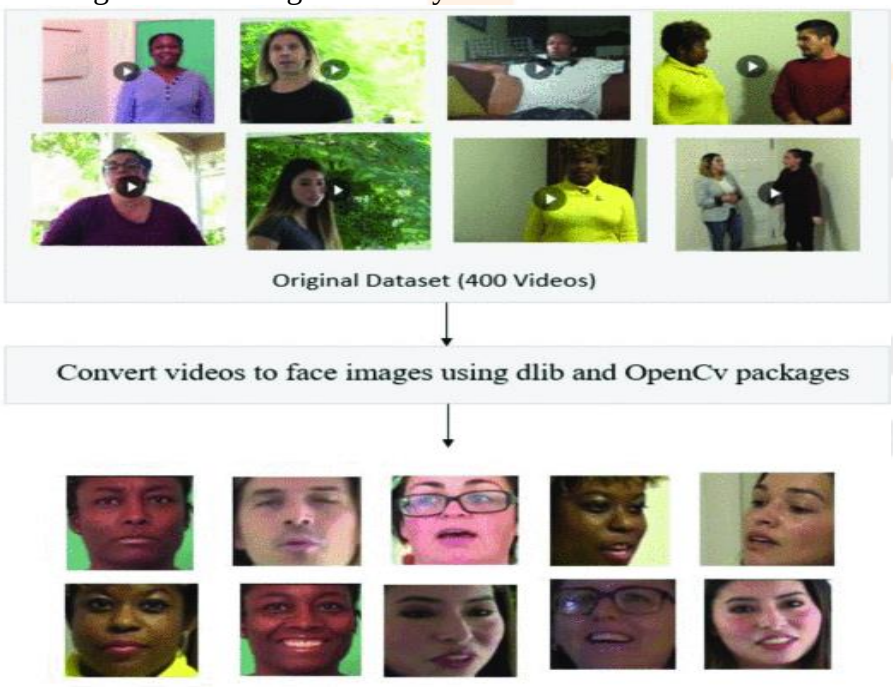


Figure 3: Dataset's creation process

It is inefficient to extract characteristics from the full frame because faces only take up a tiny amount of each frame and visual changes usually take place within facial regions. Rather, we concentrate on obtaining features exclusively from areas where faces are visible. In order to reduce computing overhead and retain enough information for analysis, we suggest training the model on a subset of frames—for example, 150 frames—in order to minimize computational loads.

Taking still images from the video is the first step in this process. For this work, we make use of the dlib[12] and OpenCV packages, utilizing their picture identification, video recording, and interpretation capabilities, which include face and object recognition. Face regions taken from both fake and actual content are included in the final dataset, which is necessary for training.
We flatten the images and eliminate noise to boost algorithm performance because it has been demonstrated that clearer images greatly increase model accuracy. Figure 3 shows how the dataset was created, highlighting the frame capture from the source videos.
Moreover, the input layer of the model can be optimized and its performance enhanced by applying pre-processing techniques such image scaling and data normalization [13]. These actions can improve desirable characteristics or lessen artifacts that could distort the network. Effective pre-processing of image inputs is made possible by the use of datastores and functions, which allow for effective data modification prior to model training.
Videos & Video Processing: A video is a picture that is transferred through a communication channel. Every video in this dataset has a duration of 10 seconds and has only undergone one change. The metadata.json file contains information that is used to categorize the dataset into two folders: real and fake. Ten frames are taken from each movie, marked as real or fake, and the video file name and frame number (e.g., fake_aagfhgtpmv_1.png) are added. These are then saved in the appropriate directories.
Frames and Frame Extraction: A frame is an individual picture among a series of images that together make up a single second of video. Generally speaking, videos consist of 24 or 30 frames per second (FPS). Every frame has an image and the amount of time it was visible. Reducing extraneous frames and processing time while effectively representing the video content with the fewest possible frames is the aim of frame extraction. Ten frames are taken from each video using OpenCV, which is used to extract frames from videos. Images with human faces are extracted from these frames and saved in the.png file extension [14].

Figure 4: Extracted frame

Resizing the Image: The pixels in an image's height and width are referred to as its resolution. Every frame in a video has the same resolution, for example, (500x480) or (620x540). Resizing an image is adjusting its dimensions to other resolutions, such as 35x48 or 47x50. Standardized resolutions make processing easier, and each frame's dimensions are set to 128 by 128 pixels so that the classification system may use them as input.
Labeling and Data Splitting: Model training and testing are the two phases in which the deep learning algorithm runs. Typical picture features are extracted during training, and a unique label is given to each group. During the testing stage, these feature-space partitions help classify picture features. CNN models are used in supervised learning, which depends on labeled data for effective learning. Real and phony photographs make up the two categories into which the collection is split. Data is divided into frames from actual and fake videos for training. But labeling is saved for assessing the model's performance, not for labeling during testing.

4.3 Model:-

Using the preprocessed data as input, modeling is an important step. Using pre-trained models, like InceptionResNet-v2 [15], is an example of transfer learning—a strategy that makes use of previously learned information to effectively address related issues. By removing the requirement to start from scratch, transfer learning makes it possible to adapt learnt features to new computer vision tasks.
For object detection and image feature extraction in our investigation, we used InceptionResNet-v2, a convolutional neural network (CNN) [16] trained on the ImageNet database, which contains over a million

photos divided into 164 categories. This pretrained model's last layer synthesizes knowledge from previous levels to categorize input into the appropriate classes.Our goal was to leverage this model's ability to discern broad markers of deepfake videos, discerning telltale signs of manipulated images.

InceptionResNet-v2 architecture:
InceptionResNet-v2 is built using both the ResNet and Inception frameworks. This network, which is 164 layers deep, is excellent at classifying photos into 1000 item categories that include different items, animals, and more. Its architecture combines residual connections with multi-scale convolutional filters inside inception resnet blocks. These residual connections reduce training time in half and tackle the problem of degradation in deep structures.
Recent advances in image recognition performance have been made possible by the ResNet and Inception designs, which provide remarkable results at little computing cost. By incorporating residual connections into the Inception framework, the Inception-ResNet design leverages the best features of both architectures.
This combination improves the model's effectiveness and performance, making it a potent tool for jobs like deepfake identification.
The ultimate deep learning model for image recognition tasks is the Inception_ResNet_v2, which is a combination of the Inception and ResNet designs. By combining the advantages of both architectures, it performs better across a range of computer vision applications.
• Architecture Fusion: Inception_ResNet_v2 combines the residual connections of ResNet with the multi-branch architecture of Inception. This fusion mitigates the vanishing gradient issue and enables effective feature extraction over several paths.
• Deep Representation Learning: Inception_ResNet_v2's deep layers enable it to recognize complex patterns and features at many levels of abstraction. The model can recognize minute details in photos thanks to this depth, which is important for tasks like deepfake detection.
• Pre-Trained Weights: The model gains a deep grasp of visual features by pre-training on extensive image datasets like ImageNet. Its effectiveness in the target job is further increased by fine-tuning on domain-specific data, such as face features retrieved for deepfake detection.
• Hierarchical Feature extraction: This technique, which extracts features at various scales and resolutions, is made possible by the inception modules of the architecture. The model is strong at identifying deepfake manipulations because of its ability to recognize both local and global patterns in the input data thanks to its hierarchical representation.
• Scalability and Efficiency: Inception_ResNet_v2 is scalable and efficient in terms of memory footprint and processing resources, even with its depth and complexity. Because of its efficiency, it may be implemented in settings with limited resources, guaranteeing scalability for practical uses.
By adding the Inception_ResNet_v2 model, the deepfake detection system can distinguish between real and fake content more effectively, offering a strong barrier against the spread of fake media.

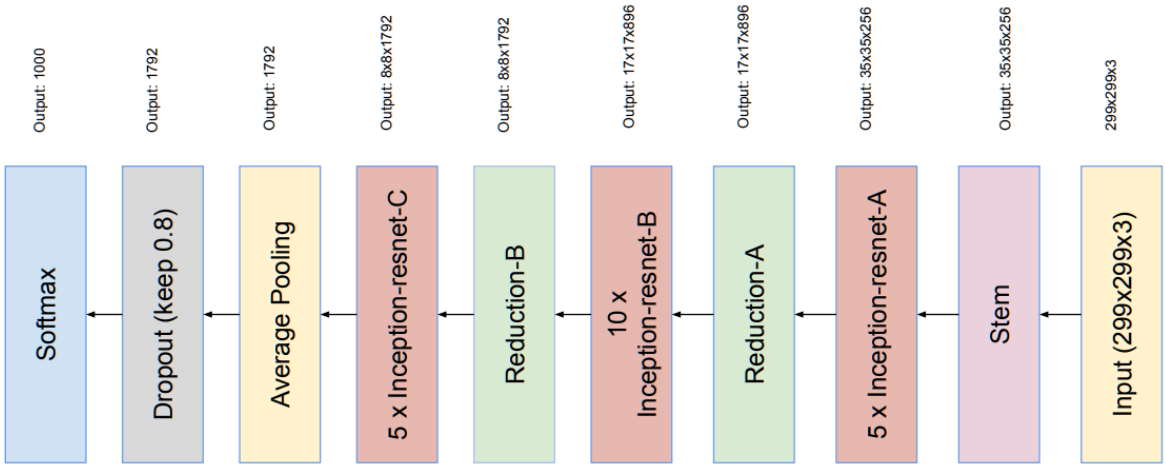


Figure 5: The basic architecture of Inception-Resnet-v2 [17]

4.4 TrainingProcess:-

Deepfake detection involves three essential components: validation, testing, and training. Our suggested model's training component is where learning takes place. Deep learning models must be carefully designed and adjusted in order to be tailored to certain problem domains. In order to guarantee that the model learns efficiently, our goal during training is to identify the dataset's ideal parameters. Similar to training, validation involves fine-tuning the model and monitoring its advancement and accuracy

in identifying deepfakes. It assists in keeping an eye on the model's performance and making the required modifications.

On the other hand, testing entails assigning a class to the faces that have been taken from particular videos and classifying them. This element is essential for assessing the model's efficacy and helps us accomplish our study goals.

Our model relies heavily on feature learning (FL) in addition to classification. FL uses convolutional processes to extract learnable features from facial photos. The FL component, which is modeled after the Inception-ResNetV2 architecture, is devoid of fully connected layers and prioritizes feature extraction over face classification.

Fully linked layers are added with variable weight initialization, and Inception-ResNetV2 is initialized with pre-trained weights. Using the fully connected layer's expectations, the network is trained end-to-end using the binary cross-entropy (BCE) loss function [18]. Using clipped faces from randomly chosen video frames, BCE loss is calculated, depending on the likelihood that the films are being altered. Using the ADAM optimizer, this loss function modifies all of the model's weights—aside from those in the Inception-ResNetV2 layers. [19]

This comprehensive training approach ensures effective feature learning and classification for deepfake detection.

$$LogLoss = -\frac{1}{n}\sum_{i=1}^{n}[y_i log(\hat{y}_i) + log(1 - y_i) log(1 - \hat{y}_i)]$$

5 *RESULTS:*

5.1 Implementation:

First, frames from movies are taken and saved as images. Next, OpenCV is used for identification and processing. Before being fed into the InceptionResNetV2 model for training, these frames go through pre-processing. The deepfake detection loss output layer, an output layer created especially for identifying deepfake loss, takes the role of the loss layer in this model. Limiting deviations from the dataset or anticipated performance is made easier with precise network tuning.

To maximize learning on the training dataset, the model is trained for 20 epochs using a learning rate of 0.00001 and a batch size of 100. The neural network performs better when the Sigmoid activation function is used, which maps input data to values between 0 and 1. Confusion matrix is used for additional assessment, giving information about the effectiveness of the model.

Using successive batches of frames with sizes of 224 × 224 × 3, the network's weights are optimized during training using the ADAM optimizer's default values of β1 = 0.9 and β2 = 0.999. Table 1 provides a summary of the parameters used in the Inception-ResNet-V2 model. The goal of this thorough training procedure is to efficiently train the model for precise deepfake identification.

Table 1. Parameters for inception-resnet-v2

S.No.	*Parameters*	*Values*
1	Number of layers	467
2	Learning Rate	0.00001
3	Number of Epochs	20
4	Batch Size	100
5	Optimizer	Adam

Using the Google Colab cloud platform to implement the Inception-ResNet-V2 model in Python has a number of benefits, including effective cooperation and access to strong computational resources. We may use key libraries for data processing, model development, and visualization by importing them, including Sklearn, NumPy, Pandas, TensorFlow, Keras, and Matplotlib.

Utilizing a GPU to run the model speeds up calculations, increasing productivity and cutting down on training time. We are able to efficiently handle big datasets and intricate deep learning models because to our 128GB RAM, 32GB GPU RAM, and 4TB hard drive capacity.

5.2 Parameters for Evaluation

The confusion matrix and accuracy are two of the most important metrics used in the assessment of the deepfake detection algorithm.

Confusion Chart: As seen in Figure 6, a confusion matrix is a table that characterizes a classification system's performance [20]. It groups the predictions into four primary categories, which serve to summarize the output of a classification algorithm:

• True Positive (TP): Information that is appropriately categorized as positive even when it is not expected to be positive.

• False Positive (FP): Information that should be categorized as negative but is mistakenly assigned as positive.

• True Negative (TN): Information that is appropriately categorized as negative even because it should be negative.

• False Negative (FN): Information that should be categorized as negative but is mistakenly assigned a positive value.

Numerous assessment metrics may be computed thanks to the confusion matrix, which offers insightful information about the classifier's performance.

We show the distribution of predictions in our confusion matrix representation by using different color tones within each class. Larger numbers are shown by lighter values, and lesser numbers are indicated by darker values. Notably, the model produces less mistakes and more accurate predictions when darker patches are found in the false positive and false negative sectors.

Interpreting the model's performance and pinpointing areas for development is made easier with the help of this depiction. We can evaluate the efficacy of the deepfake detection model and adjust its parameters by looking at the confusion matrix.

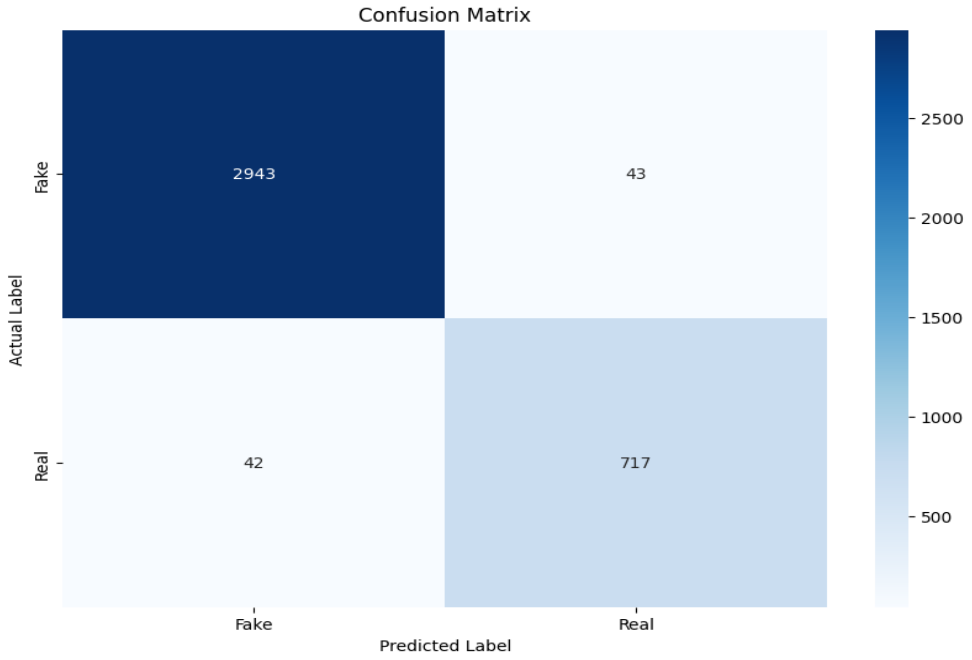


Figure 6: Confusion matrix

Accuracy

Accuracy serves as a fundamental metric for assessing classification models, quantifying the ratio of correctly predicted instances to the total number of predictions [21]. Formally, accuracy is computed using the equation:

ACCURACY=(TP+TN)/(TP+FN+FP+TN)

where:

TP (True Positive): Correctly predicted positive instances.

TN (True Negative): Correctly predicted negative instances.

FP (False Positive): Incorrectly predicted positive instances.

FN (False Negative): Incorrectly predicted negative instances.

After 20 epochs during the training phase, the model's precision value of 0.9890 demonstrated a high degree of accuracy in differentiating between actual and fraudulent videos. This suggests that the algorithm has a 98.90% accuracy rate in differentiating between actual and bogus videos. Furthermore, the model achieved a validation accuracy of 0.977, which means that on a validation dataset that was not used for training, it accurately classified deepfakes with an accuracy of 97.7%. This illustrates how well the model generalizes to new data.

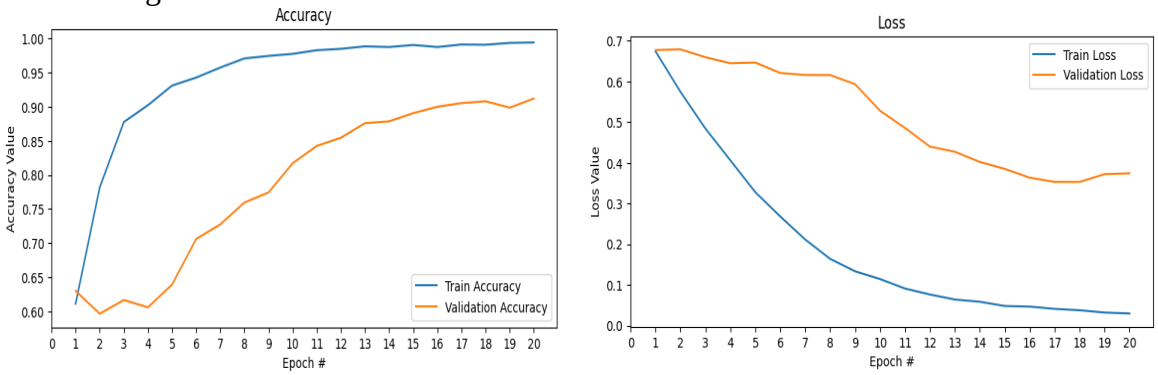


Figure 7: Training And Validation Accuracy and Loss

In terms of loss values, during training the model produced a loss value of 0.0292, which shows that deepfake detection is successful. A low loss number indicates that the model is learning the dataset effectively. Nevertheless, a score of 0.3744 for the validation loss was obtained, suggesting some differences from the training set. This disparity points to possible areas for development, such broadening the dataset's variety or improving the model's architecture.
The model's performance during the various stages of training and validation is shown in Figure 7, which also shows the link between training and validation accuracy and loss. It is imperative to keep an eye on these indicators in order to assess the model's efficacy and pinpoint areas in need of improvement.

Table 2: Performance of the proposed model

Model	**Performance criteria**			
	Accuracy	**Recall**	**Precision**	**F1-Score**
InceptioResNet-V2	97.73	98.5	98.6	98.6

As shown in Table 2, The proposed approach achieves accuracy of 97.73 %, recall of 98.5 %, precision of 98.5 %, and F1 score of 98.6 %.

5.1 Comparative Analysis

We used our dataset to train different models, and after a thorough analysis, we discovered that the ResNet50 + LSTM model consistently yields the best results during both the training and testing stages. But we found that this network had a hard time accurately identifying alterations in low-quality photos taken in dimly lit or hazy environments. In spite of these difficulties, the model performs exceptionally well at detecting manipulations in high-caliber recordings.

The accuracy rates attained by our research in comparison to other deepfake detection strategies covered in the related works section are shown in Table 3.

Table 3: The accuracy of different approach

Model	***Train Data***	***Test Data***
Custom Model	0.8523	0.8057
ResNet50+LSTM	0.9795	0.9463
MesoNet	0.9568	0.8997
DenseNet121	0.9699	0.9181
InceptionResNet-v2	**0.9951**	**0.9773**

InceptionResNet-v2 clearly provides better accuracy than some of the proposed methods in other researches.

6 CONCLUSION:

It is commonly known that face tampering in videos occurs frequently, and that this poses problems as well as opportunities in a variety of industries, including virtual reality, robotics, digital media, education, and more. Although these developments seem promising, there is a chance that they will upset and compromise social order.

We included a neural network-based technique to identify films as real or deepfake as part of our design strategy, and we also evaluated the model's confidence. Our suggested approach uses the Inception-ResNetV2 architecture for frame-level detection and subsequent video classification, taking inspiration from the process of creating deepfakes with Generative Adversarial Networks (GANs) and Autoencoders. Our approach achieves an average accuracy of 97.73% in identifying deepfake movies on a variety of internet sites by carefully adjusting the parameters. Our goal is to achieve comparable precision in real-time implementations, which will allow for quick and dependable identification of deepfakes. Deep learning provides a flexible framework for solving problems and enables us to tackle complicated problems without requiring extensive previous hypotheses. But it's important to comprehend the fundamental ideas and constraints of these kinds of solutions. As a result, we invested a great deal of time and energy into illustrating how our network functions inside, highlighting the importance of elements like mouth and eyes in differentiating between real and deepfake appearances.

We believe that in the future, technological developments will improve our organizational capacities, making them stronger, more effective, and more skilled at handling the challenges posed by the spread of deepfakes. We can embrace deep learning's revolutionary potential while preventing its misuse by iteratively improving our techniques and being alert to new risks.

REFERENCES:

[1] Yuezun Li, Siwei Lyu, “ExposingDF Videos By Detecting Face Warping Artifacts,” in arXiv:1811.00656v3.

[2] YuezunLi,Ming-ChingChangandSiweiLyu “ExposingAICreatedFakeVideosbyDetecting Eye Blinking” in arxiv.

[3] Huy H. Nguyen , Junichi Yamagishi, and Isao Echizen “ Using capsule networks to detect forged images and videos ”.

[4] Umur Aybars Ciftci, ˙Ilke Demir, Lijun Yin “Detection of Synthetic Portrait Videos using Biological Signals” in arXiv:1901.02212v2.

[5] R. Durall, M. Keuper, F.-J. Pfreundt and J. Keuper, *Unmasking DeepFakes with simple Features*, Nov. 2019, [online] Available: http://arxiv.org/abs/1911.00686.

[6] E. Johansson, *Detecting Deepfakes and Forged Videos Using Deep Learning*.

[7] Zheng, Nanning, George Loizou, Xiaoyi Jiang, Xuguang Lan, and Xuelong Li. "Computer vision and pattern recognition." (2007): 1265-1266.

[8] Yamashita, R., Nishio, M., Do, R.K.G. et al. Convolutional neural networks: an overview and application in radiology. Insights Imaging 9, 611–629 (2018). https://doi.org/10.1007/s13244-018-0639-9.

[9] MadallahAlruwaili, Abdulaziz Shehab, Sameh Abd El-Ghany, "COVID-19 Diagnosis Using an Enhanced Inception-ResNetV2 Deep Learning Model in CXR Images", Journal of Healthcare Engineering, vol. 2021, Article ID 6658058, 16 pages, 2021. https://doi.org/10.1155/2021/6658058.

[10] Szegedy, Christian & Ioffe, Sergey &Vanhoucke, Vincent & Alemi, Alexander. (2016). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. AAAI Conference on Artificial Intelligence. 31. 10.1609/aaai.v31i1.11231.

[11] Dolhansky, Brian & Howes, Russ & Pflaum, Ben & Baram, Nicole & Ferrer, Cristian. (2019). The Deepfake Detection Challenge (DFDC) Preview Dataset.

[12] King, Davis. (2009). Dlib-ml: A Machine Learning Toolkit. Journal of Machine Learning Research. 10. 1755-1758. 10.1145/1577069.1755843.

[13] Zhou, Weibin& Ma, Xiaotong & Zhang, Yong. (2020). Research on Image Preprocessing Algorithm and Deep Learning of Iris Recognition. Journal of Physics: Conference Series. 1621. 012008. 10.1088/1742-6596/1621/1/012008.

[14] R. R. Schultz and R. L. Stevenson, "Extraction of high-resolution frames from video sequences," in IEEE Transactions on Image Processing, vol. 5, no. 6, pp. 996-1011, June 1996, doi: 10.1109/83.503915.

[15] S. Guefrechi, M. B. Jabra and H. Hamam, "Deepfake video detection using InceptionResnetV2," 2022 6th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), Sfax, Tunisia, 2022, pp. 1-6, doi: 10.1109/ATSIP55956.2022.9805902.

[16] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 2017, pp. 1-6, doi: 10.1109/ICEngTechnol.2017.8308186.

[17] https://www.geeksforgeeks.org/inception-v4-and-inception-resnets/

[18] Ruby, Usha &Yendapalli, Vamsidhar. (2020). Binary cross entropy with deep learning technique for Image classification. International Journal of Advanced Trends in Computer Science and Engineering. 9. 10.30534/ijatcse/2020/175942020.

[19] Kingma, Diederik & Ba, Jimmy. (2014). Adam: A Method for Stochastic Optimization. International Conference on Learning Representations.

[20] Karimi, Zohreh. (2021). Confusion Matrix.

[21] Sokolova, Marina &Japkowicz, Nathalie &Szpakowicz, Stan. (2006). Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation. AI 2006: Advances in Artificial Intelligence, Lecture Notes in Computer Science. Vol. 4304. 1015-1021. 10.1007/11941439_114.

IV.