Heliyon 10 (2024) e25757

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

An efficient deepfake video detection using robust deep learning

Abdul Qadir [a], Rabbia Mahum [a,*], Mohammed A. El-Meligy [b], Adham E. Ragab [c], Abdulmalik AlSalman [d], Muhammad Awais [e]

[a] Computer Science Department, UET, Taxila, Pakistan
[b] Advanced Manufacturing Institute, King Saud University, Riyadh, 11421, Saudi Arabia
[c] Industrial Engineering Department, College of Engineering, King Saud University, P.O. Box 800, Riyadh, 11421, Saudi Arabia
[d] Computer Science Department, King Saud University, Riyadh, Saudi Arabia
[e] Henan International Joint Laboratory of Laser Technology in Agriculture Sciences, Zhengzhou 450002, China

ARTICLE INFO

Keywords:
Video deepfakes
Multimedia forensics
Swish
Visual manipulation

ABSTRACT

The creation and manipulation of synthetic images have evolved rapidly, causing serious concerns about their effects on society. Although there have been various attempts to identify deep fake videos, these approaches are not universal. Identifying these misleading deepfakes is the first step in preventing them from spreading on social media sites. We introduce a unique deep-learning technique to identify fraudulent clips. Most deepfake identifiers currently focus on identifying face exchange, lip synchronous, expression modification, puppeteers, and other factors. However, exploring a consistent basis for all forms of fake videos and images in real-time forensics is challenging. We propose a hybrid technique that takes input from videos of successive targeted frames, then feeds these frames to the ResNet-Swish-BiLSTM, an optimized convolutional BiLSTM-based residual network for training and classification. This proposed method helps identify artifacts in deepfake images that do not seem real. To assess the robustness of our proposed model, we used the open deepfake detection challenge dataset (DFDC) and Face Forensics deepfake collections (FF++). We achieved 96.23% accuracy when using the FF++ digital record. In contrast, we attained 78.33% accuracy using the aggregated records from FF++ and DFDC. We performed extensive experiments and believe that our proposed method provides more significant results than existing techniques.

1. Introduction

In the modern world, falsifying images and videos threaten society. Any audio or video clip can be artificially created. Thanks to artificial intelligence (AI), especially machine learning (ML) methods, it isn't easy to distinguish altered photos and movies from the originals [1]. To alter photos and videos, a few conventional approaches are employed. Some are content altering-based, whilst others focus on computer-generated images like GIMP, Photoshop, and CANVA. Deepfake, which is based on Deep Learning (DL) [2–5], is a strong competitor among the procedures for customizing the content of videos. Deepfake is a phrase that evolved from the ideas of DL and fraud. DL networks (DNN) have accelerated and simplified the process of creating compelling synthetic images and movies. It involves altering a person's video or image with another person's image using DL algorithms such as the general adversarial network

* Corresponding author.
E-mail addresses: abdul.qadir@students.uettaxila.edu.pk (A. Qadir), rabbia.mahum@uettaxila.edu.pk (R. Mahum), melmeligy@ksu.edu.sa (M.A. El-Meligy), aragab@ksu.edu.sa (A.E. Ragab), salman@ksu.edu.sa (A. AlSalman), dr.muhammad.awais@henau.edu.cn (M. Awais).

https://doi.org/10.1016/j.heliyon.2024.e25757
Received 20 July 2023; Received in revised form 26 January 2024; Accepted 1 February 2024
Available online 9 February 2024

(GAN) [6,7]. Face swap is a crucial step in establishing a fake video as it involves the exchange of a source that challenges a victim's face while preserving the victim's initial movements and verbalization spoofs.

Generative Adversarial Networks (GANs) [8] are the primary driving force behind facial modification methods. As the use of StyleGAN [9] and StyleGAN2 [10] to synthesize the images increases, it becomes increasingly difficult for the human visual system to distinguish them. Many parodies, pranks, and other social channels exist on YouTube, Instagram, Twitter, Tiktok, and other streaming websites that use GAN-based face-swapping techniques. Commercial smartphone apps like ZAO2 [11] and FaceApp3 [12] would significantly speed up the adoption of these deepfake capabilities by making it natural for ordinary internet users to produce fake picture frames and clips. Deepfake records were first easily visible to the human eye, but as technology advances, they have become increasingly complex to differentiate from real images [13,14]. Because of this, there has been an increase in the frequency of its inappropriate use, leading to the formation of numerous pornographic images of politicians and celebrities to disseminate propaganda and fake news, leading to many societal issues. While they shouldn't be, they can influence a person's social standing and mental health. It has become so easy that mobile cameras, tiny devices, or other bugs are used to capture pictures or videos anytime. Then, it can be altered using professional image processing technology software, allowing one to create bogus pictures or films. Even some companies are specialists in offering such deepfakes services.

Various techniques have been proposed for deepfake detection; however, they are still unreliable for unseen videos. Moreover, the applicability of advanced DL methods for deepfake generation has made detection very difficult due to environmental changes, such as lighting, compression, change in scale and positions, etc. Hence, the increasing fraudulent activities using deepfakes have forced researchers to propose a more reliable method for deepfake detection. Therefore, to overcome the above glitches and shortcomings of previous efforts, we provide a robust approach for deepfake detection. We propose the innovative DL technology: ResNet-Swish-BiLSTM. The reason behind the Swish activation function is to utilize its smoothness as it combines the ReLU and Sigmoid functions. It is proven to be more efficient than ReLU and helps in learning complex patterns from videos achieving a high-recall rate. Furthermore, the extracted features from ResNet's layers are then classified using BiLSTM [15] layers that capture the meaningful attributes from input features and classify the frames effectively. The main contributions are as follows.

a. For deepfake detection, we offer ResNet-Swish-BiLSTM, a distinctive architecture based on Bi-LSTM and Residual Network. It is a more robust technique than previous deepfake detection methods due to its ability and reliability in feature extraction.
b. We analyzed the proposed model using DFDC, FF++, and Celeb-DF collections, which include a cross-compilation check along with a confirmatory assessment to assess the suitability and generalizability of the proposed ResNet-Swish-BiLSTM network for visual fraud detection.
c. The presented work works well against cyberattacks such as compression, noise, blurring, translation, and rotation variations, as ResNet-Swish and BiLSTM can propagate a more meaningful set of visual features within neurons and performs the classification task.

The remaining part of the research is organized as follows: Section 2 outlines the relevant deepfake detection techniques. Then, Section 3 describes the methodology, Section 4 explains the results and experiments used to analyze the performance, and Section 5 concludes the article.

2. Related work

The three types of modern deepfake records are Face Swap (FS), Lip Sync, and Puppeteer. The face of the target person is exchanged with the face of a source person to develop a fictional clip of the addressee in FS deepfakes. Deepfakes containing lip sync with audio are called lip sync deepfakes, so it appears they are saying the lyrics. When using Puppet Master, the source person's facial expressions remain on the target face while the target face is inserted into the original video to create a more convincing impersonation. Certain types of deepfake are the focus of most current detection methods; however, general methods capable of defeating all deepfakes are studied less often. A technique for detecting lip sync deepfakes has been released by Agarwal et al. [16]. This method took advantage of the discrepancies of both a phoneme (spoken word) and the viseme (shape of the mouth). Viseme-phoneme mapping was calculated in this study using manual and CNN-based approaches. This model works well for a specific collection of visual data.

Existing deepfake detection methods can be broadly divided into two groups: those based on hand-made features [17] and those using DL [18,19]. Yang et al. [19] trained an SVM (Support Vector Machine) classifier for recognition using 68-D face marking data. This approach worked well with high-quality movies from the UADFV [19] and DARPA MediFor datasets but had difficulties with low-quality movies. In addition, not all deepfakes were considered in the analysis of this study. Using 16-Dim texture-based eye and teeth characteristics, Massod et al. [20] were able to exploit visual flaws and detect deepfakes like FS and F2F. The most crucial aspect of this study was determining the various eye colors of a POI (Point of Interest) to detect FS deepfakes by filling in the blanks, such as using the reflection in the eye color. This study also uses attributes like the nose's top, face's rim, and eye retina color to distinguish F2F deepfakes. Unfortunately, only faces with clean teeth and open eyes can be identified with this approach [20], which is a disadvantage. Finally, only the FF++ collection of deepfake examples was used for this work's assessment evaluation. The intended affine warping artifacts that arise during the production of deepfakes have been proposed by Massod [20]; by focusing on specific objects, training cost has been reduced. However, as it becomes difficult to detect a deepfake with some advanced transforming characteristics, this artifact selection could weaken the resilience of this approach.

The open-source toolkit OpenFace2 [11] was developed by Agarwal et al. [21] to extract features from facial features. Several features were created based on the retrieved landmark features. Then, a binary SVM was trained for deepfake detection using these

derived and action unit traits. Five POIs were used in this approach, and a T-SNE plot showed that each POI could be linearly separated. However, the performance of this approach was severely impacted by the increased number of POIs brought about by the new datasets. DL-based approaches are also used for identifying deepfakes, in addition to some professional apps for altering facial content. To identify the deepfakes, Doke et al. [18] used a DL-based approach. This approach used a long short-term memory (LSTM) to learn the features after a CNN extracted them. The key contribution of this study was the use of deepfakes sequence and pattern inconsistencies for categorization. Still, the issue with this approach is lacking accuracy for the top three deepfake levels. Another DL model (MesoNet) was developed by Xia et al. [22] to identify deepfakes and F2F video manipulations, convolution and pooling layers for feature extraction followed by dense layers for classification in this unified architectural design. Instead of using a standard data set, these algorithms [22] were tested on digital media collected from different websites, raising questions about their sustainability over a sizeable and heterogeneous standard dataset. A bubble network was reported by Nguyen et al. [23] to uncover different types of manipulations in images and video clips. This framework is designed to identify computer-generated images, FS, and Natural Textures (NT). The performance of the proposed model has been improved by using dynamic routing and expectation maximization methods. The VGG-19 was used by the Nguyen bubble network to extract latent facial attributes, which are then categorized as legitimate or fake. Although the framework is sophisticated in computations, it effectively detects FS deepfakes in the FF++ dataset. Still, it has not been tested on lip-sync and puppet master deepfakes. Table 1 summarizes some recent developments in the domain of deepfakes.

3. Proposed methodology

The details of the proposed system are provided in this section. The proposed ResNet-Swish-BiLSTM deepfake video identification technique is shown in Fig. 1. This proposed approach utilizes the Swish activation function that minimizes the movement of negative values across the network while optimizing the model's learning behaviour due to its smoothness feature. To extract face landmark characteristics from the input video, the OpenFace 2.0 [11] toolkit was used. The Residual Blocks (RBs) extract the face's key features to categorize whether they are authentic or fake. We also utilized BiLSTM layers that focus on capturing the most useful features for classifying deepfake frames.

3.1. Data acquisition

For making the model efficient for real time prediction. We have gathered the data from different available datasets like FF++ [28] and DFDC [29]. Further we have mixed the dataset with the collected datasets and created our own new data set, to accurate and real time detection on different kind of videos. To avoid the training bias of the model we have considered 70% real and 30% fake videos. DFDC consist of certain audio alerted video, as audio deepfake are out of scope for this paper. We preprocessed the DFDC dataset and removed the audio altered videos from the dataset by running a python script. After preprocessing of the DFDC dataset, we have taken 1000 x Real and 1000 x Fake videos from the DFDC dataset and 1000 x Real and 1000 x Fake videos from the FF++ dataset. Which makes our total data set consisting of 2000 Real, 2000 fake videos datasets.

3.2. Pre-processing locating and cropping faces in videos

In this process system detect the faces from the frames after extracting them. In the facial area of frames, changes are mainly made for visual modification. Therefore, the procedure described here focuses primarily on the facial area. We used OpenFace 2.0, a face detector toolkit, to collect the faces. The method locates the facial region using 2 and 3-dimension facial area landmarks. As illustrated in Fig. 1, we have chosen seven (POI), the outward even edge (OE), the outward left edge (LE), the chin (C), the frontal head (FH), the outward proper cheek (PC), odd check (OC), and the center of the face (MF). The primary justification for using OpenFace 2.0 for face identification is that this method works well to find faces even when there are changes in face orientation, intensity fluctuations, and the position of the capturing device. The OpenFace toolkit's capabilities allow it to categorize face images from the collection of clips even when they go through considerable transformation changes.

Table 1
A Comparison of prior research on deepfake video recognition.

Research	DF Collections	CNN Attributes
A. Jaiswal [24].	FF++	Bidirectional recurrent neural networks (RNN) with DenseNet/ResNet50 are used to analyze the spatiotemporal properties of video streams.
P.Dongare [18]	Hollywood-2 Human Actions	It takes into consideration the deep-fake video's temporal irregularities. Inception-V3+LSTM
S. Lyu [25].	Closed Eyes in the Wild (CEW)	Used long-term recurrent convolution networks the frequency of eye blinking VGG16+LSTM + FC
A. Irtaza [14].	Fusion of datasets	Differences in facial structure, missing detail in the eyes and mouth, and a neutral network and logarithmic regression model
Nguyen [23].	Four major datasets	VGG-19 + Capsule Network
Hashmi [26].	DFDC whole dataset	CNN + LSTM Used facial landmarks and convolution features
Ganiyusufoglu [27].	FF++, Celeb-DF	Triplet architecture, Metric learning approach

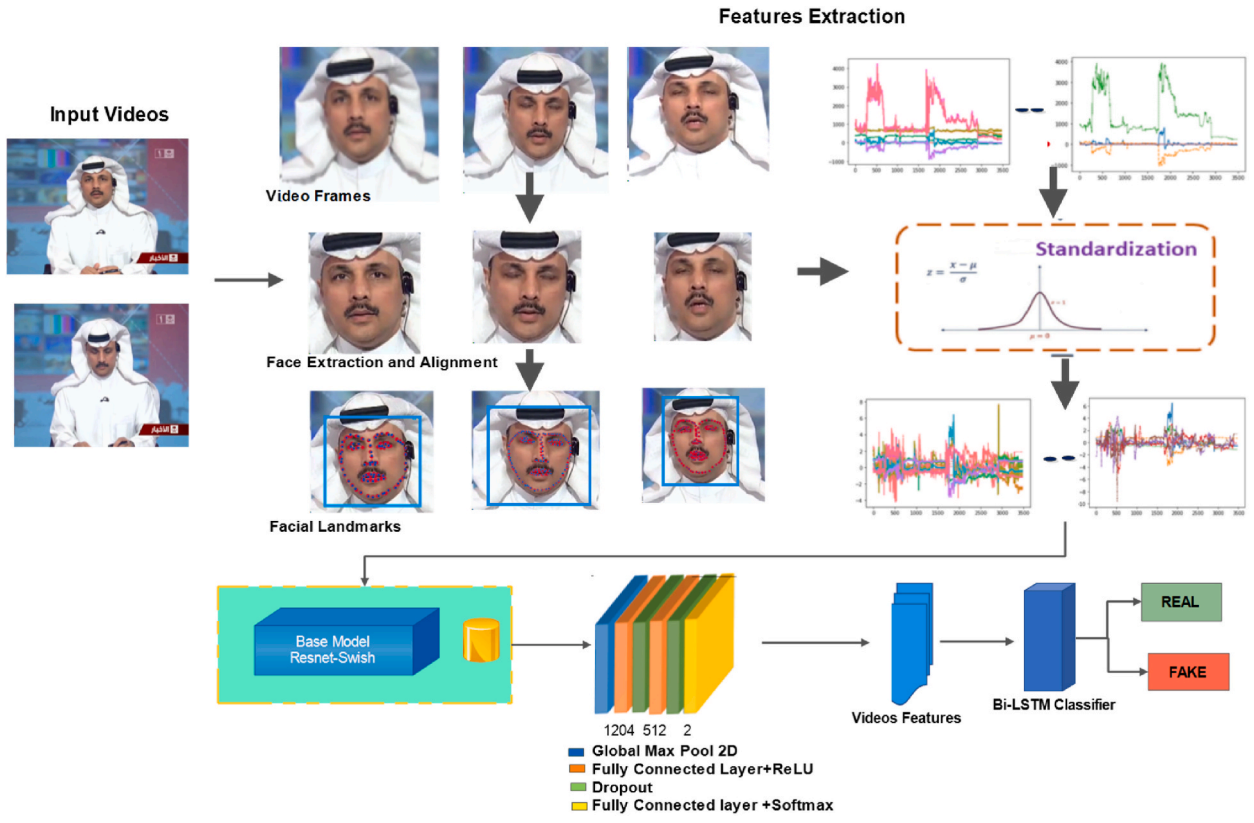

Fig. 1. Design and architecture of the proposed model.

3.3. *Standardization and segmentation*

Different methods are used for the standardization of features. The extracted attribute is normalized during data loading by semi-training the distributed facial attributes. We use Equation (1) for the standardization.

$$z = \frac{X - \mu}{S}, \tag{1}$$

Where the mean and standard deviation of feature columns are represented by μ and S, respectively. The values of X are the input face attributes vector. Both the frames and the segment levels are functional with our solution. We created fragments with a frame duration of 100 and an overlapping of 25 frames to allow the slices to work planar. Projected resolution requirements are 25 frames per second to maintain computational complexity in our case.

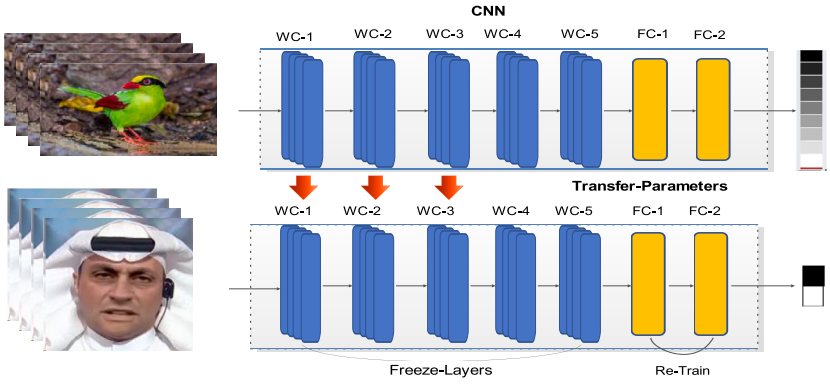

Fig. 2. Transfer Learning-based visual presentation for deepfake.

3.4. Feature computation

The next challenge is to compute features from the video images after extracting the faces. We adjusted the pre-trained ResNet18 CNN network by introducing the low-value elimination method in the current framework to solve this issue. The theory behind the Swish incitement design is that it eases the model's capability to scatter negative numbers through the neural network, helping to capture the complicated underlying patterns in visual perception. On the other hand, the supplementary RNN (Recurrent Neural Network) and additional coatings help select a representative set of support countenance that is passed for categorization. The starting layers of CNN Networks are responsible for learning the essential visual information, but the layers at the end focus more on computing the information required for each job. Therefore, using a pre-trained CNN network for deepfake detection improves the accuracy of spurious recognition while accelerating learning. The primary purpose of using a pre-trained model is to quantify more accurate sets of image attributes because it has been previously trained using the extensive ImageNet database of online datasets. Fig. 2 demonstrates the visual representation of the above task.

3.5. Base ResNet-18

We used the K. He et al. [30] approach to extract features. They designed an attention-based 2-D RB network that is both straightforward and efficient. The design of ResNet18 is described in Fig. 3. By deepening the network and addressing the problem of a limited training dataset, the architecture of ResNet18 [30] improves the performance of visual categorization. We selected the Residual Network model, which comprises eight fundamental RBs, a convolutional layer, and a totally connected layer. Each basic block has two convolutional layers, and subsequently, each convolutional layer is followed by batch normalization and a nonlinearity authorize function called ReLU [30]. Typically, the traditional CNN algorithms draw a thick collection of visual physiognomy from the information of all former slices to boost object identification accuracy [31]. However, due to the gradient vanishing problem in the learning phase, when the network thickness is increased, the algorithms with these types of design configurations experience a significant performance impact [32]. The Resnet approach suggested using hop connections for fully convolutional topologies that need to exclude some network layers to overcome the problems with existing Residual Neural network models. RBs are then built up on these connections. The resulting structure allows the keypoint mapping from the earlier layers to be reused, improving performance and simplifying training. The fundamental component of the Resnet model, the RB, is shown graphically in Fig. 4. Equation (2) presents the computation of residual function.

$$Y = F(x_i) + x_i, \tag{2}$$

In the above expression, attributes are fed into the formula symbolized by the letters x_i. The residual function through the letters F and Y represents the results obtained after applying the residual function and addition to x_i.

3.6. Proposed model ResNet-Swish-BiLSTM

Additionally, the use of full interconnects during input-to-state, and state-to-state transitions is enabled by the addition of Bi-LSTM in ResNet-18, which processes spatiotemporal data in FC-LSTM [33] rather than spatial information is coded. In contrast, Convolutional LSTM (ConvBi-LSTM) solves this problem by including 3D tensors with spatial (rows and columns) last two dimensions for all inputs. Table 2 provides the structural details of the proposed ResNet-Swish-BiLSTM models, and Fig. 5 illustrates the proposed network design from a comprehensive perspective. The 18 total layers reside in the proposed CNN Network framework and are further broken down into four phases with full connection layers. Each stage has several RBs stacked on top of each other.

Similarly, integrating the attention-based RB with a set of filters of Kernal size "3 × 3" can grab and extract the different vertices, which is very useful for classification during the final training, making the proposed model efficient and intelligent for pointing the POI. After that, by using the BiLSTM block, the network can accept a series of convolution feature vectors from the input frames and a 2-node neural network which predicts whether the sequence will come from a deepfake or original video clip. The main problem we must address is developing a logical model to process a sequence of features. To achieve this, we experiment with BiLSM, which

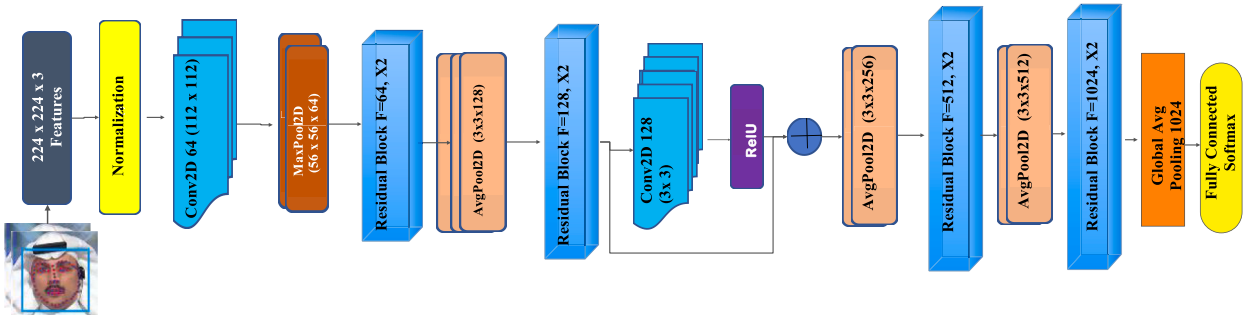

Fig. 3. ResNet-18's architecture.

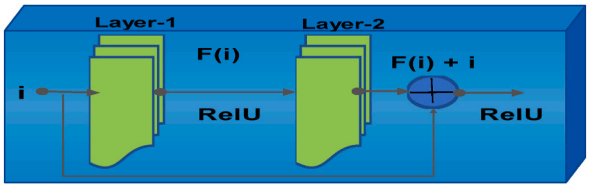

Fig. 4. Architectural description of RB.

Table 2
ResNet-Swish-BiLSTM structural description.

Layer Name	Activation	Learnable
Image Input	224 × 224 × 3	–
Convolution	112 × 112 × 64	Weights 7x7x3x64 Bias 1 × 1 × 64
Batch Normalize	112 × 112 × 64	offset 1 × 1 × 64 Scale 1 × 1 × 64
Swish Activation	112 × 112 × 64	
Multi Plan	112 × 112 × 64	
Convolution	56 × 56 × 64	Weights 3 × 3 × 64 × 64 Bias 1 × 1 × 64
Batch Normalize	56 × 56 × 64	Offset 1 × 1 × 64 Scale 1 × 1 × 64
Convolution Block	28 × 28 × 128	Weights 3 × 3 × 128 × 128 Bias 1 × 1 × 128
Identity Block	28 × 28 × 128	
Convolution Block	14 × 14 × 256	Weights 3 × 3 × 258 × 256 Bias 1 × 1 × 256
Identity Block	14 × 14 × 256	
Convolution Block	7 × 7 × 512	Weights 3 × 3 × 128 × 512 Bias 1 × 1 × 512
Identity Block	7 × 7 × 512	
Convolution Block	3 × 3 × 1024	Weights 3 × 3 × 1024 × 1024 Bias 1 × 1 × 1024
Identity Block	1 × 1 × 1024	
Residual Block-1	1 × 1 × 1024	
AP (Average Pooling)	1 × 1 × 1024	
Residual Block-2	3 × 3 × 1024	
Featureinput	9216	
BiLSTM1	500	inputWeights:500 × 9216,
FC1	200	Weights:200 × 500,Bias: 200 × 1
BiLSTM2	200	inputWeights:2500 × 9216, Bias:2500 × 9216
Dropout 25%	200	–
FC2 2 Fully Connected	2	Weights:2 × 200, Bias:2 × 1
SoftMax	2	–
Classification	2	–

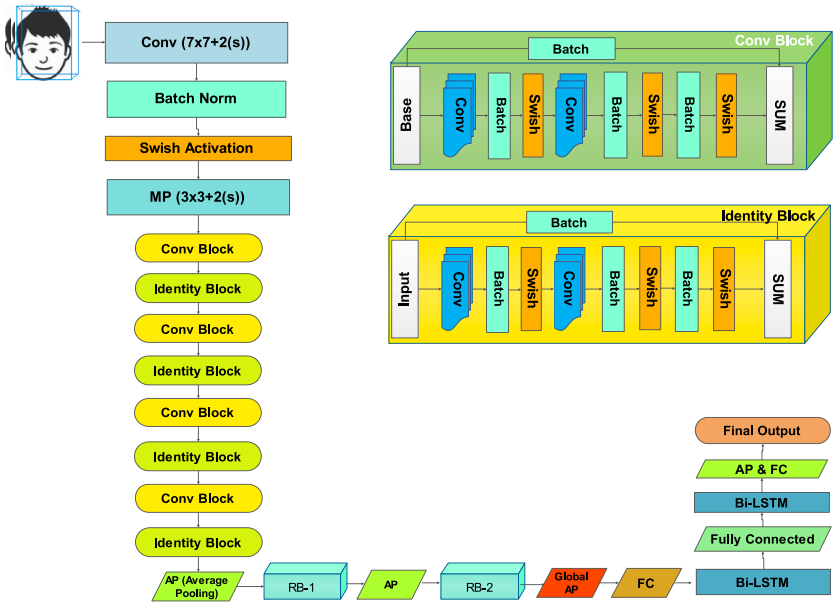

Fig. 5. The ResNet-Swish-Bi-LSTM architectural description.

produces positive results, like resorting to a 9216-wide BiLSTM unit with a prediction dropout error of 0.5. This addition in the parameter has a positive outcome, which we want. More specifically, 9216-D spatial feature vectors are fed into the BiLSTM model to train it for evaluation. Further, we added in the proposed model a fully connected 512 layers with a 0.5 percent dropout probability at the end of the proposed network to determine the probability that the frame sequence is either authentic or deepfake. We used a softmax layer for making the final classification.

4. Experiments and results

This section describes the experiments used to measure the detection and classification accuracy of the proposed model. We also highlighted the delicate aspects of the dataset used. Extensive experiments are also described to elucidate the suitability of our strategy. Subsection 4.1 describes the evaluation metrics we used to measure the accuracy, and Subsection 4.2 discusses the dataset in depth. The details of the experimental design are presented in Subsection 4.3. Subsections 4.4 to 4.7 cover the various trials we conducted to evaluate the effectiveness of the suggested technique.

4.1. Evaluation metric

Several common metrics, including Precession (PR), Recall (RC), Accuracy (AC), and F1-Score, were used to assess how well the suggested method behaved on the deepfakes detections. Equations (3)–(7) are the metrics used for the performance evaluation.

$$Ac = \frac{D^{'} + \prod}{D^{'} + \prod + r + R}, \tag{3}$$

$$Pr = \frac{D^{'}}{D^{'} + r}, \tag{4}$$

$$Re = \frac{D^{'}}{D^{'} + R}, \tag{5}$$

$$F1 - \ score = 2^{*} \ \frac{Precision^{*}Recall}{Precision + Recall}, \tag{6}$$

Where $D^{'}$ denotes the positive (deepfakes values), and $\prod$ refers to negative values. Similarly, r stands for the mistakenly positive results (untrue bonafide), and R stands for the mistakenly adverse (false deepfakes), respectively. Some evaluation measurements using these equations on the proposed classification model are detailed in the following subsections.

4.2. Dataset

Our proposed model is evaluated using two datasets: DFDC provided by Facebook [29] and the FF++ [28]. The Facebook dataset contains 4119 modified visual examinations and 1131 unique records. The Kaggle competition website allows users to obtain the DFDC data, an open and free online dataset [29]. A variety of different AI-based techniques, including Deepfakes: FS [20], F2F [34], and NT [35], were used to create the FF++ dataset. 1000 authentic and 1000 altered records are included for each control process. Modifying recordings using simple, light, and high compression settings is also common nowadays. Table 3 shows some of the abandoned sets that DL used for deepfake. The datasets we used during the experiment are randomly split into 70% and 30% ratios to learn the proposed model for the training and testing, respectively. Since the modified recordings in the FF++ sample set were created with an extended deepfake algorithm, this is specifically used for the evaluation [36]. This calculation creates a high-quality visual record that closely resembles the real-world environment. The datasets usually contain actual and false examples, as shown in Fig. 6.

4.3. Implementation details

We used PowerEdge R740, Intel Xeon Gold 6130 2.1G, 16C, 128 GB RAM and 10 TB HDD with Windows 11 OS, A 64 GB T4 GPU

Table 3
A list of collections that include modified and original videos.

DF collections with Origin	Year	Pristine vs Modified
UADFV [17]- YouTube	Nov-2018	49 vs. 49
Deep Fake-TIMIT [37] - YouTube	Dec-2018	550 vs. 620
FF++ [38] - YouTube	Jan-2019	1000 vs. 4000
Google DFD [39] - Actors	Sep-2019	363 vs. 3068
DFDC [29]- Actors	Oct-2019	23,654 vs. 104,500
Celeb-DF [40]- YouTube	Nov-2019	890 vs. 5639
DeeperForensics [41]- Actors	Jan-2020	10,000 vs. 50,000

Fig. 6. Samples from the used collections, with instances in the first record that are real and in the second record that have been modified.

and a GeForce RTX 2060 (Realistic Card). The proposed DL-based network is developed using the latest version of Python 3.9.0 language, with Python libraries like Keras, Tensorflow, OpenFace2, Sklearn, Numpy, Arbitrary, OS and PIL. To implement the proposed model and perform the required task successfully, the following requirements must be met.

a. The OpenFace2 toolbox is used to perform standardization, segmentation, and feature extraction of facial features.
b. As the model requires, the patterns of identified faces are sized to 224 × 224 dimensions.
c. We have trained the model for 60 epochs, 0.000001 learning rate, 35 batch sizes, and stochastic gradient descents (SGD as the hyperparameters.

4.4. Assessment of the proposed model

Accurately detecting visual changes is a prerequisite for a trusted forensic analysis model. To illustrate the power of deepfake detection, we've evaluated several common measures. The results are presented in Fig. 7 to highlight the classification results of the recommended approach for the FF++ deepfake collections. Regarding identifying FS, F2F, and NT deepfakes, the data in Fig. 7 shows that the proposed framework has achieved reliable performance. More specifically, the suggested approach achieved PR, RC, and AC 99.25%, 97.89%, and 98.99%, respectively, for the FS Deepfake. Similarly, we achieved 98.23%, 97.78%, and 98.08%, respectively, for the F2F Deepfakes. Moreover, for the NT Deepfake, the obtained results were 95.45%, 97.66%, and 96.23%, respectively. The above results demonstrate the resilience of our proposed method. The ResNet-Swish-BiLSTM model is significant enough in picking and selecting the features that enable the framework to present complex visual patterns and utilize them for classification effectively.

4.5. Assessment of the Swish function

Knowing about activation functions is very important because they play a major role in training neural networks. So, it is better to be aware of the pros and cons of different activation functions beforehand. Exploring the activation functions compared in this paper, Swish consistently performed better than others. Although Swish put up an intense fight with the Mish activation function, Mish has some implementation problems. For this reason, we used Swish AF to solve deepfake problems with the proposed network model. Table 4 shows the results of some activation functions on both DFDC and FF++ datasets.

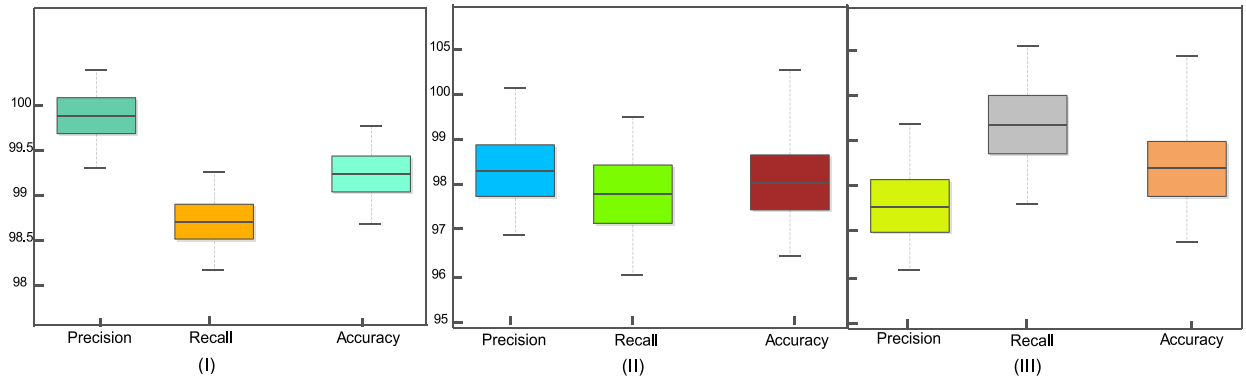

Fig. 7. Execution of the proposed approaches is examined for preciseness, recall, and accuracy across the FF++ dataset for the detection of (I) FS, (II) F2F, and (III) NT deepfakes.

ReLU is still better than Leak_ReLU (L-ReLU), but the difference in accuracy has significantly decreased with the number of epochs; L-ReLU is performing better than ReLU now. For longer training, L-ReLU performed better than ReLU, and Swish and Mish performed way better than other activation functions. Swish is more accurate than Mish in such types of applications. So, based on these observations, we can conclude that Swish outperformed other activation functions in the list. The other main reason for the selection is the non-monotonic nature of the Swish activation method, which allows the computed values to fall still even if the input rises. Hence, the result improves the computed values storage capacity of the proposed approach. Similarly, employing the Swish activation method optimizes the model behavior by improving the proposed approach's feature selection power and recall ability. Hence due to these factors, the proposed ResNet-Swish-BiLSTM model presents the highest performance results for classifying visual manipulations.

4.6. Comparing the improved performance of ResNet-18 to the base model

In this subsection, we compared the effectiveness of the proposed deepfake detection method with the original ResNet-18 model. Table 5 illustrates a comparison, and Fig. 9 shows the confusion matrix of the proposed deepfake detection model on the FF++ test set, clearly showing its effectiveness. We attained the best results for FS, F2F, and NT deepfakes when evaluating all performance metrics using the ResNet-18 model. We have attained performance gains of 4.72%, 5.57%, 5.16%, and 6.24% for the FS deepfakes detection PR, RC, F1, and AC, respectively. In the PR, RC, F1, and AC measurements for F2F, we gained improvements of 5.81%, 4.73%, 3.93%, and 6.99%, respectively. For the NT deepfake's PR, RC, F1, and AC measurements, we achieved 4.72%, 5.75%, 4.33%, and 6.33%, respectively. In addition, we compared the AUC (area under the curve) with ROC (receiver operating characteristic curve) for the FS, F2F, and NT deepfakes in Fig. 8 (a, b, and c), which clearly shows how robust our method is comparable to the base model. The analysis shows that the enhanced version, ResNet-Swish-BiLSTM, outperforms the base model over the FF++ dataset. Although, our proposed technique has more parameters (18.2 million) than ResNet-18 (13.5 million). However, it is quite clear from the presented results that the proposed technique has reliable performance for deepfake detection. Integrating Swish activation methods into the model, which produces precise feature computation, is the main factor contributing to the improved performance of the proposed solutions in recognition. Furthermore, the additional Bi-LSTM layers help the model deal with network overfitting data more effectively, ultimately improving performance.

4.7. Comparison with existing DL models

Creating high-quality datasets is crucial, as current techniques based on DL rely heavily on large amounts of data. We contrasted the suggested technique with several current CNN network methodologies evaluating similar deepfake collections to understand better how well they behave for detecting deepfakes. As deepfake algorithms become more prevalent, new datasets must be created to provide more robust algorithms that counter new manipulation tactics.

We first compared the accuracy of our deepfake detector with VGG16 [32], VGG-19 [42], ResNet101 [43], InceptionV3 [44], InceptionResV2 [45], XceptionNet [46], DenseNet-169 [15], MobileNetv2, EfficientNet [47] and NasNet-Mobile [48], respectively. The comparative findings are shown in Fig. 10. The results show that the proposed technique works better than the alternative options. VGG16 got the lowest AC at 88%. The VGG19 model took the second lowest place with 88.92%. The developed Res-Swish-BiLSTM model got the best AC with 98.06%. The average accuracy score when comparing is 93.84%, however, our strategy has a score of 98.06%. As a result, a 4.22% average performance increase in the accuracy metric is achieved. Fig. 10 shows a comparison using the DFDC dataset based on accuracy, recall, precision, and F1-score with the existing DL techniques. Additionally, using the FF++ dataset, we evaluated the proposed technique against existing DL models including XceptionNet [46], VGG16 [32], ResNet34 [49], InceptionV3 [44], VGGFace, and EfficientNet, and the results are shown in Fig. 10. From the results, it is clear that the Res-Swish-BiLSTM model achieved more accurate classification results than other DL methods for all classes: FS, F2F, and NT. When finding low-quality movies versus high-quality videos, detection algorithms often show a decrease in effectiveness, as shown in Table 6.

Hence, Figs. 10 and 11, Table 6 and 7 demonstrate that the provided ResNet-Swish-BiLSTM model is more robust to visual tampering detection than previous DL techniques for all specified assessment parameters. The novel proposed method works consistently well due to the improved feature computation capability of the proposed model, which allows it to learn the key points more effectively. Furthermore, the Resnet18 model's ability to deal with the video's visual changes, such as lighting and background variations, has been improved by employing the Swish activation approach, which results in superior deepfakes detection performance.

Table 4
Comparative analysis of Swish and other activation functions over DFDC and FF++

Activation Function	Accuracy (%)	Avg training time (sec)	Avg time (sec) classification	Remarks
Sigmoid	94.0	1110	2549	Can't work for Boolean gates simulation
Swish	**98.0**	**1166**	**3057**	**Worth giving a try in very deep networks**
Mish	98.35	1155	3524	It has very few implementations, not matured
Tanh	90.0	1173	2950	In the recurrent neural network
Relu	97.0	1050	2405	Prone to the dying ReLU" problem
Leak_Relu	97.5	1231	2903	Use only if expecting a dying ReLU problem

Table 5
Comparison of the core with proposed models over FF++ deep fake collections.

CNN-Model	PR (%)			RC (%)			F1 (%)			AC (%)		
	FS	F2F	NT	FS	F2F	NT	FS	F2F	NT	FS	F2F	NT
ResNet-18	94.15	89.42	94.33	90.19	92.5	94.22	91.1	88.5	85.23	89.2	92.5	86.6
Propose Model	**99.25**	**98.23**	**95.48**	**97.89**	**97.8**	**97.66**	**96.3**	**92.9**	**89.56**	**98.9**	**98.0**	**96.2**

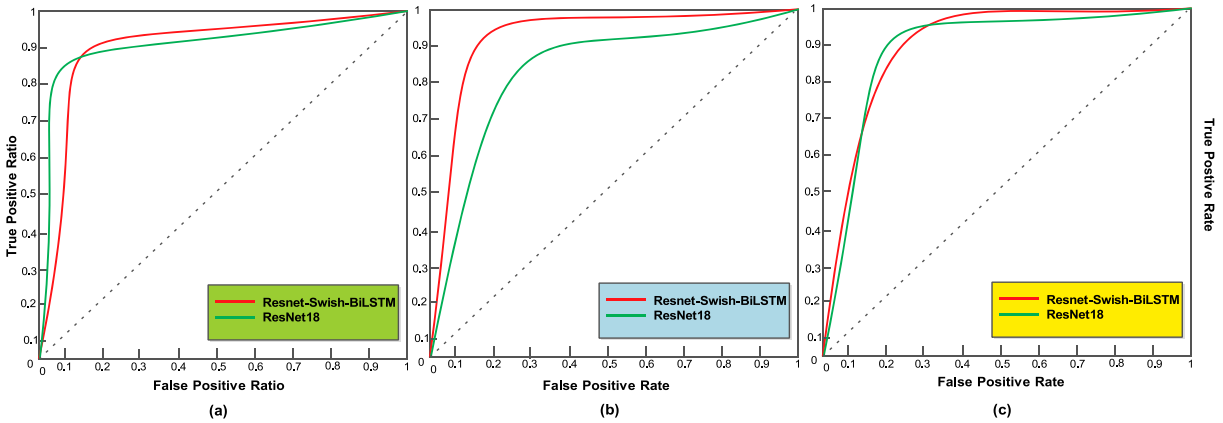

Fig. 8. AUC with ROC plots against the proposed and underlying CNN network on the FF++ deepfake collections (FS, F2F, and NT, respectively).

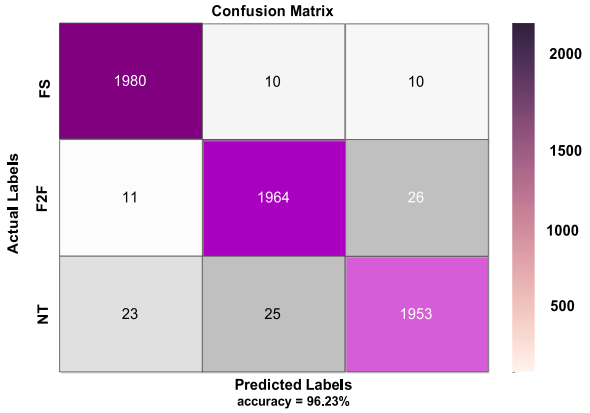

Fig. 9. The confusion matrix for the proposed model over FF++ dataset.

4.8. Cross-validation

We conducted an experiment to measure the effectiveness of the proposed deepfake detection method in a cross-corpus scenario. For this challenge, we chose the Celeb-DF data set. The collection includes 950 edited videos and 475 original videos. Unfortunately, the tiny visual distortions make it difficult to detect tampering.

The ResNet-Swish-BiLSTM model was tested for the situations presented in Table 8, where it was trained on the FF++ (FS, F2F, and NT) deepfake collections and assessed comparatively on the FF++, DFDC, and Celeb-DF deepfakes. The statistics in Fig. 12 show unequivocally that in a cross-corpus scenario, the performance of the proposed technology has decreased compared to the database-internal evaluation situation. The main reason for this degradation in performance is that the method used does not account for temporal variations that have evolved within frames throughout training, which could have helped the method to capture the underlying manipulated biases more accurately. Table 8 clarifies that in the first case, the suggested DL model network AUC values for the DFDC and Celeb-DF collections are 71.56% and 70.04%, respectively, when trained on the FF++(FS, F2F, and NT) deepfake collections. Suggested CNN network ResNet-Swish-BiLSTM produced the AUC values for the FF++ and Celeb-DF deepfake data sources are 70.12% and 65.23%, respectively, when trained on the DFDC dataset. These results show how well the model holds up to unobserved cases from whole new deepfake collections.

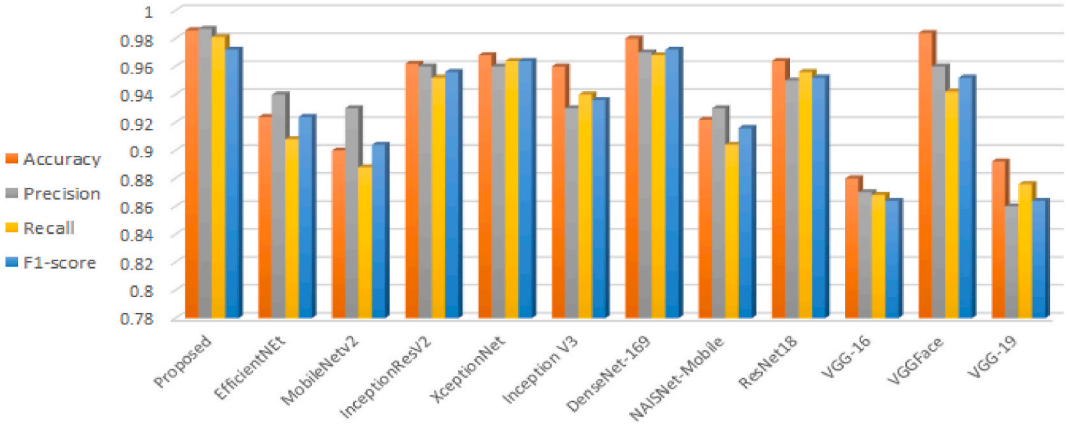

Fig. 10. Chart showing the different DL network's Efficiency by utilizing the DFDC dataset.

Table 6
Comparative analysis with existing methods using various DF datasets.

Study	Method	Dataset	Performance (AC)
E.D. Cannas [50]	Group of CNN	FF++(c23)	84%
Sabir [24]	CNN + GRU + STN	FF++, DF	96.9%
		FF++, F2F	94.35%
		FF++,FS	96.3%
J.C. Neves [51]	3D head poses	UADFV	97%
F. Juang	Eyeblink + LRCN	Self-made dataset	97.5%
Ciftci [52]	Biological signals	Self-made deep fakes dataset	91.07%
Tarasiou [53]	A lightweight architecture	DFDC	78.76%
Keramatfar [54]	Multi-threading using Learning with Attention	Celeb-DF	70.2%
Nirkin [55]	FACE X-RAY	Celeb-DF	81.58%
Ciftci [52]	Bio Identification	Celeb-DF	90.50%
Proposed Method		**FF++, FS**	**99.13%**
		FF++,F2F	**98.08%**
		FF++,NT	**99.09%**

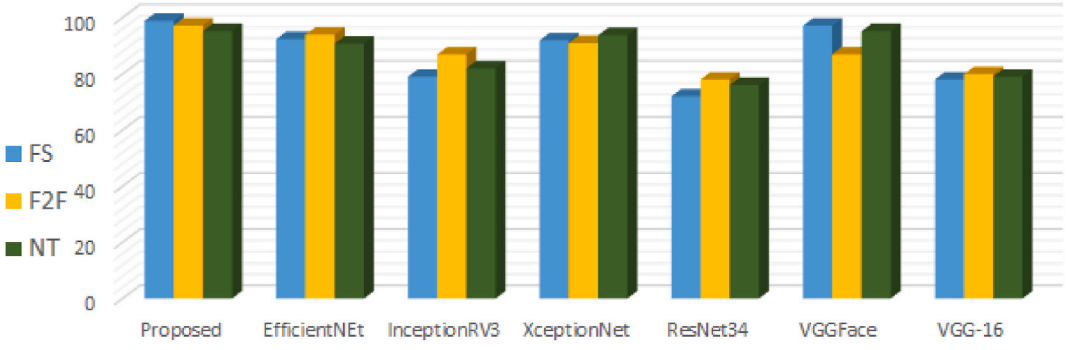

Fig. 11. Assessing the accuracy of the DL techniques using the FF++ dataset.

5. Conclusion

This study has provided a comprehensive strategy to identify all three forms of deepfakes based on the fusion of our unique facial features. Unlike many other systems, the proposed model is easy to use, understandable, efficient, and resilient at the same time. We introduced a novel Resnet-Swish-BiLSTM deepfake detection model. The utility of the proposed visual tampering detection technique was extensively tested on the deepfake data source FF++, DFDC and Celeb-DF. We evaluated the proposed method across deepfake collection to demonstrate its generalizability for unusual scenarios. We found that the suggested modelling technique can distinguish between the modified and the unmodified digital footage with a high-recall rate and recognizes various visual modifications, including FS, NT and F2F. The FF++ and DFDC datasets, which show the highest AU values of 0.9623 and 0.9876 have been used to evaluate the

Table 7
Performance Comparison of DL network's over the DFDC dataset.

DL-Models	AC	PR	RC	F1
Proposed	0.986	0.99	0.992	0.988
EfficientNEt	0.924	0.94	0.908	0.924
MobileNetv2	0.9	0.93	0.888	0.904
InceptionResV2	0.962	0.96	0.952	0.956
XceptionNet	0.968	0.96	0.964	0.964
Inception V3	0.96	0.93	0.94	0.936
DenseNet-169	0.98	0.97	0.968	0.972
NAISNet-Mobile	0.922	0.93	0.904	0.916
ResNet18	0.964	0.95	0.956	0.952
VGG-16	0.89	0.87	0.868	0.864
VGGFace	0.984	0.96	0.942	0.952
VGG-19	0.882	0.86	0.876	0.864

Table 8
Cross-data validation accuracy corresponding to the suggested convolution network.

Training Dataset	Class		Proposed Model-ACU				
	Real	Fake	FS (%)	F2F (%)	NT (%)	DFDC (%)	C-DF (%)
FS	988	983	99.36	61.7	68.71	64.22	49.13
F2F	988	978	65.23	99.53	65.9	75.22	70.05
NT	988	950	47.98	84.82	99.5	71.19	80.39
DFDC	1000	980	49.68	74.25	78.55	99.36	65.23
C-DF	475	950	55.88	77.22	80.23	71.56	92.22

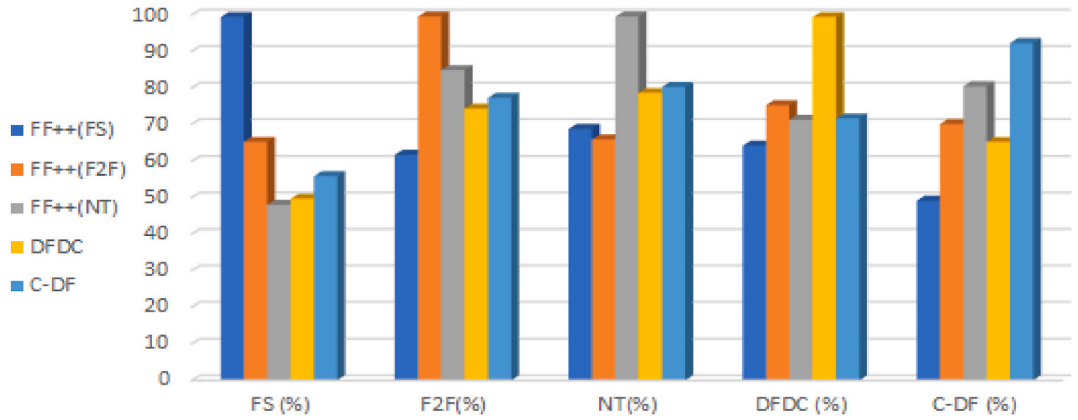

Fig. 12. Cross-dataset assessment of the proposed hypothesis using numerous datasets.

proposed technique in detail, respectively. Therefore, after thoroughly examining the ResNet-Swish-BiLSTM at the statistical and digital media levels, we can conclude that our work in the field of advanced digital investigation, such as criminal forensics, is potentially beneficial. Since the model cannot yet capture the temporal patterns of the faked material over time, our ultimate aim in the future is value addition in form of temporal pattern analytics and reasoning to further improve the discovery, inference, and adaptability capabilities of the projected approach.

Data availability statement

Data sharing does not apply to this article as authors have used publicly available datasets, whose details are included in the "experimental results and discussions" section of this article. Please contact the authors for further requests.

CRediT authorship contribution statement

Abdul Qadir: Writing – original draft, Methodology, Data curation, Conceptualization. **Rabbia Mahum:** Investigation, Formal analysis, Data curation. **Mohammed A. El-Meligy:** Software, Resources, Project administration. **Adham E. Ragab:** Validation, Supervision, Funding acquisition. **Abdulmalik AlSalman:** Writing – original draft, Visualization, Muhammad Awais, Software, Project

administration.

Funding statment

This research was funded by the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia through the project no. (IFKSUOR3-582-6).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The authors present their appreciation to Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia for funding this research through the project no (IFKSUOR3-582-6).

References

[1] P.S.Q. Yeoh, K.W. Lai, S.L. Goh, K. Hasikin, Y.C. Hum, et al., Emergence of deep learning in knee osteoarthritis diagnosis, Computational intelligence and neuroscience 2021 (2021).
[2] K. Bjerge, H.M. Mann, T.T. Høye, Real-time insect tracking and monitoring with computer vision and deep learning, Remote Sensing in Ecology and Conservation 8 (3) (2022) 315–327.
[3] N. Le, V.S. Rathour, K. Yamazaki, K. Luu, M. Savvides, Deep reinforcement learning in computer vision: a comprehensive survey, Artificial Intelligence Review (2022) 1–87.
[4] A. Bouguettaya, H. Zarzour, A.M. Taberkit, A. Kechida, A review on early wildfire detection from unmanned aerial vehicles using deep learning-based computer vision algorithms, Signal Processing 190 (2022) 108309.
[5] P. Shukla, R. Aluvalu, S. Gite, U. Maheswari, Computer Vision: Applications of Visual AI and Image Processing, vol. 15, Walter de Gruyter GmbH & Co KG, 2023.
[6] T.T. Nguyen, C.M. Nguyen, D.T. Nguyen, D.T. Nguyen, S. Nahavandi, Deep learning for deepfakes creation and detection 1 (2) (2019) 2, *arXiv preprint arXiv: 1909.11573*.
[7] Marek Kowalski, Faceswap [Online]. Available :https://github.com/deepfakes/faceswap, , Jan 2020. (Accessed 19 January 2021).
[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, et al., Generative adversarial networks, Communications of the ACM 63 (11) (2020) 139–144.
[9] C. Bravo-Prieto, J. Baglio, M. Cè, A. Francis, D.M. Grabowska, et al., Style-based quantum generative adversarial networks for Monte Carlo events, Quantum 6 (2022) 777.
[10] P. Zhu, R. Abdal, Y. Qin, J. Femiani, P. Wonka, Improved Stylegan Embedding: where Are the Good Latents?, 2020 *arXiv preprint arXiv:2012.09036*.
[11] T. Baltrusaitis, A. Zadeh, Y.C. Lim, L.-P. Morency, Openface 2.0: facial behavior analysis toolkit, in: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), IEEE, 2018.
[12] F. Schroff, D. Kalenichenko, J. Philbin Facenet, A unified embedding for face recognition and clustering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.
[13] A. Radford, L. Metz, S. Chintala, Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, 2015 *arXiv preprint arXiv: 1511.06434*.
[14] S. Suwajanakorn, S.M. Seitz, I. Kemelmacher-Shlizerman, Synthesizing obama: learning lip sync from audio, ACM Transactions on Graphics (ToG) 36 (4) (2017) 1–13.
[15] R. Mahum, S. Aladhadh, Skin lesion detection using hand-Crafted and DL-based features fusion and LSTM, Diagnostics 12 (12) (2022) 2974.
[16] S. Agarwal, H. Farid, O. Fried, M. Agrawala, Detecting deep-fake videos from phoneme-viseme mismatches, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020.
[17] S. Kolagati, T. Priyadharshini, V.M.A. Rajam, Exposing deepfakes using a deep multilayer perceptron–convolutional neural network model, International Journal of Information Management Data Insights 2 (1) (2022) 100054.
[18] Y. Doke, P. Dongare, V. Marathe, M. Gaikwad and M. Gaikwad, "Deep Fake Video Detection Using Deep Learning." Journal homepage: www.ijrpr.com ISSN, vol. 2582, pp. 7421..
[19] X. Yang, Y. Li, S. Lyu, Exposing deep fakes using inconsistent head poses, in: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)., IEEE, 2019.
[20] M. Masood, M. Nawaz, K.M. Malik, A. Javed, A. Irtaza, et al., Deepfakes Generation and Detection: State-Of-The-Art, Open Challenges, Countermeasures, and Way Forward, Applied Intelligence, 2022, pp. 1–53.
[21] S. Agarwal, H. Farid, Y. Gu, M. He, K. Nagano, et al., Protecting world leaders against deep fakes, in: CVPR Workshops, 2019.
[22] Z. Xia, T. Qiao, M. Xu, X. Wu, L. Han, et al., Deepfake video detection based on MesoNet with preprocessing module, Symmetry 14 (5) (2022) 939.
[23] H.H. Nguyen, J. Yamagishi, I. Echizen, Capsule-forensics: using capsule networks to detect forged images and videos, in: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019.
[24] E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, et al., Recurrent convolutional strategies for face manipulation detection in videos, Interfaces (GUI) 3 (1) (2019) 80–87.
[25] Y. Li, S. Lyu, Exposing Deepfake Videos by Detecting Face Warping Artifacts, 2018 *arXiv preprint arXiv:1811.00656*.
[26] M.F. Hashmi, B.K.K. Ashish, A.G. Keskar, N.D. Bokde, J.H. Yoon, et al., An exploratory analysis on visual counterfeits using conv-lstm hybrid architecture, IEEE Access 8 (2020) 101293–101308.
[27] I. Ganiyusufoglu, L.M. Ngô, N. Savov, S. Karaoglu, T. Gevers, Spatio-temporal Features for Generalized Detection of Deepfake Videos, 2020 *arXiv preprint arXiv: 2010.11844*.
[28] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, et al., Faceforensics: A Large-Scale Video Dataset for Forgery Detection in Human Faces, 2018 *arXiv preprint arXiv:1803.09179*.
[29] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, et al., The Deepfake Detection Challenge (Dfdc) Dataset, 2020 *arXiv preprint arXiv:2006.07397*.
[30] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: Computer Vision–ECCV 2016: 14th European Conference, Springer, Amsterdam, The Netherlands, 2016. October 11–*14, 2016, Proceedings, Part IV 14*.
[31] M.J. Akhtar, R. Mahum, F.S. Butt, R. Amin, A.M. El-Sherbeeny, et al., A robust framework for object detection in a traffic surveillance system, Electronics 11 (21) (2022) 3425.

[32] M. Nawaz, Z. Mehmood, M. Bilal, A.M. Munshi, M. Rashid, et al., Single and multiple regions duplication detections in digital images with applications in image forensic, Journal of Intelligent & Fuzzy Systems 40 (6) (2021) 10351–10371.
[33] A. Graves, Generating Sequences with Recurrent Neural Networks, 2013 *arXiv preprint arXiv:1308.0850*.
[34] A. Kohli, A. Gupta, Detecting DeepFake, FaceSwap and Face2Face facial forgeries using frequency CNN, Multimedia Tools and Applications 80 (2021) 18461–18478.
[35] J. Thies, M. Zollhöfer, M. Nießner, Deferred neural rendering: image synthesis using neural textures, ACM Transactions on Graphics (TOG) 38 (4) (2019) 1–12.
[36] T. Jung, S. Kim, K. Kim, Deepvision: deepfakes detection using human eye blinking pattern, IEEE Access 8 (2020) 83144–83154.
[37] P. Korshunov, S. Marcel, Deepfakes: a New Threat to Face Recognition? Assessment and Detection, 2018 *arXiv preprint arXiv:1812.08685*.
[38] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, et al., FaceForensics++: Learning to Detect Manipulated Facial Images, 2019 *arXiv preprint arXiv: 1901.08971*.
[39] Y. Hua, R. Shi, P. Wang, S. Ge, Learning patch-channel correspondence for interpretable face forgery detection, IEEE Transactions on Image Processing (2023).
[40] Y. Li, P. Sun, H. Qi, S. Lyu, Toward the creation and obstruction of deepfakes, in: Handbook of Digital Face Manipulation and Detection: from DeepFakes to Morphing Attacks, Springer International Publishing Cham, 2022, pp. 71–96.
[41] H. Chi, M. Peng, Toward robust deep learning systems against deepfake for digital forensics, in: Cybersecurity and High-Performance Computing Environments, Chapman and Hall/CRC, 2022, pp. 309–331.
[42] Y.S. Taspinar, M. Dogan, I. Cinar, R. Kursun, I.A. Ozkan, et al., Computer vision classification of dry beans (Phaseolus vulgaris L.) based on deep transfer learning techniques, European Food Research and Technology 248 (11) (2022) 2707–2725.
[43] D. Theckedath, R. Sedamkar, Detecting affect states using VGG16, ResNet50 and SE-ResNet50 networks, SN Computer Science 1 (2) (2020) 1–7.
[44] G. Chugh, A. Sharma, P. Choudhary, R. Khanna, Potato leaf disease detection using inception V3, Int. Res. J. Eng. Technol (IRJET) 7 (11) (2020) 1363–1366.
[45] M.M. Rahman, A.A. Biswas, A. Rajbongshi, A. Majumder, Recognition of local birds of Bangladesh using MobileNet and Inception-v3, International Journal of Advanced Computer Science and Applications 11 (8) (2020).
[46] A. Biswas, D. Bhattacharya, K.A. Kumar, DeepFake detection using 3D-xception net with discrete fourier transformation, Journal of Information Systems and Telecommunication (JIST) 3 (35) (2021) 161.
[47] G. Marques, D. Agarwal, I. de la Torre Díez, Automated medical diagnosis of COVID-19 through EfficientNet convolutional neural network, Applied soft computing 96 (2020) 106691.
[48] F. Saxen, P. Werner, S. Handrich, E. Othman, L. Dinges, et al., Face attribute detection with mobilenetv2 and nasnet-mobile, in: 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA), IEEE, 2019.
[49] R. Roy, I. Joshi, A. Das, A. Dantcheva, 3D CNN architectures and attention mechanisms for deepfake detection, in: Handbook of Digital Face Manipulation and Detection, Springer, Cham, 2022, pp. 213–234.
[50] N. Bonettini, E.D. Cannas, S. Mandelli, L. Bondi, P. Bestagini, et al., Video Face Manipulation Detection through Ensemble of CNNs, arXiv e-prints, 2020 arXiv: 2004.07676.
[51] J.C. Neves, R. Tolosana, R. Vera-Rodriguez, V. Lopes, H. Proença, et al., Ganprintr: improved fakes and evaluation of the state of the art in face manipulation detection, IEEE Journal of Selected Topics in Signal Processing 14 (5) (2020) 1038–1048.
[52] U.A. Ciftci, I. Demir, L. Yin, Fakecatcher: Detection of Synthetic Portrait Videos Using Biological Signals, IEEE transactions on pattern analysis and machine intelligence, 2020.
[53] W. Zhang, C. Zhao, Y. Li, A novel counterfeit feature extraction technique for exposing face-swap images based on deep learning and error level analysis, Entropy 22 (2) (2020) 249.
[54] A. Keramatfar, H. Amirkhani, A. Jalaly Bidgoly, Multi-thread hierarchical deep model for context-aware sentiment analysis, Journal of Information Science 49 (1) (2023) 133–144.
[55] Y. Nirkin, L. Wolf, Y. Keller, T. Hassner, Deepfake detection based on discrepancies between faces and their context, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (10) (2021) 6111–6121.