Multimedia Tools and Applications
https://doi.org/10.1007/s11042-023-16196-x

Convolutional long short-term memory-based approach for deepfakes detection from videos

Marriam Nawaz[1] · **Ali Javed**[1] · **Aun Irtaza**[2]

Received: 7 April 2022 / Revised: 20 May 2023 / Accepted: 4 July 2023

Abstract
The great development in the area of Artificial Intelligence (AI) has introduced tremendous advancements in information technology. Moreover, the introduction of lightweight machine learning (ML) techniques allows the applications to work with limited storage and processing power. Deepfakes is among the most famous type of such applications of this era which generates a large amount of fake and modified audiovisual data. The creation of such fake data has introduced a serious risk to the security and confidentiality of humans all around the globe. Accurate detection and classification of actual and deepfakes content is a challenging task due to the progression of Generative adversarial networks (GANs) which produce such convincing manipulated content that it's impossible for people to recognize it through their naked eyes. In this work, we have presented deep learning (DL)-based approach namely the convolutional long short-term memory (C-LSTM) method for deepfakes detection from videos. More specifically, the spatial information from the input sample is calculated by employing various pre-trained models like VGG16, VGG19, ResNet50, XceptionNet, and GoogleNet, DenseNet. Further, we have proposed a novel feature descriptor called the Dense-Swish-Net121. Whereas the Bi-LSTM model is utilized to compute the temporal information. Lastly, the results are predicted based on both the frame level and temporal level information to make the final decision. A detailed comparison of all CNN models with the Bi-LSTM approach is performed and has confirmed through the reported results that the proposed Dense-Swish-Net121 with Bi-LSTM approach performs well for deepfakes detection.

Keywords CNN · Deepfakes · Bi-LSTM · Deep learning · Multimedia forensic

✉ Marriam Nawaz
marriam.nawaz@uettaxila.edu.pk

[1] Department of Software Engineering, UET Taxila, Punjab 47050, Pakistan

[2] Department of Computer Science, UET Taxila, Punjab 47050, Pakistan

Published online: 19 July 2023

1 Introduction

The affordable prices of digital devices like cell phones, tabs, cameras, laptops iPods, etc. have enabled mankind to easily keep them. The easier availability of these gadgets has urged people to save their data in digital format which ultimately causes to increase the multimedia content like images and videos in cyberspace [29, 33]. Meanwhile, the accessibility of internet services and usage of social sites like Facebook, Twitter, and Instagram has allowed people to connect around the globe and share their audio videos-based data. Such sites allow everyone to save this data and regenerate them even without the knowledge of others. At the same time, with the availability of easy-to-use Apps and tools, people can easily change or modify digital content without the need for any special expertise [26]. Moreover, the great development in the area of ML has introduced such convincing methods which can easily change the information conveyed through this digital content. Because of such reasons, researchers have given this era the name of "post-truth" where a piece of digital content (Image, Audio, and Video) is used by hateful actors to spread disinformation to alter the beliefs of the audience [48]. The main aim of people spreading such false narratives is to affect the reputation of famous people like politicians and celebrities [19, 39]. The processing of changing the audio-visual content by using ML-based approaches is known as deepfakes. Now, the deepfakes have become so convincing and powerful that even these manipulations can affect election campaigns and cause to initiate the war type situation in the countries [45]. The easy-to-use Apps like FaceApp [5], Zao [34], REFACE [3], etc. allow users to easily change their visual content. Because of such manipulations, now it has become very difficult for people to understand what to consider real or fake. Because of such reasons, multimedia data cannot be trusted to investigate criminal cases as the audio or video-based data used as proof in these cases must be trustworthy [30]. However, the convincing generation of deepfakes has made it a complex job to verify the truthfulness of multimedia data.

Deepfakes have many positive applications and can provide cost-effective solutions to many domain problems. The positive applications of deepfakes include generating speech for deaf people, and artists can use deepfakes to show their skills. Furthermore, film producers can use deepfakes to reshoot the scenes for which the actors or actresses are no more available. Even though there are multiple positive usages of deepfakes, however, its negative employment is more prominent [44]. Like in history, manipulated content was generated to make superstars notorious to their supporters, for example, in 2017 an actress was opposed in a deepfakes-based pornographic video [35]. Therefore, the negative creation of such content can easily cause the character assassination of celebrities. Moreover, intruders can use such fake content for money purposes by threatening people to spread false information about them on the internet. Likewise, deepfakes can have a considerable effect on stocks and businesses as well all around the world. Initially, deepfakes creation require extensive data, therefore, only famous people were the main target of them, however, now with the generation of the few-shot-based deepfakes creation approaches, its influence is spread to the general audience as well. An example of a few-shot-based deepfakes generation tool includes the Zao app [31] where the users can swap their faces with actors to see themselves acting in those shots. The usage of such apps and tools can easily result in intense privacy problems for not only celebrities but the general audience as well.

Although, extensive work has been presented by the research community for deepfakes detection, however, still there is a room for performance improvement. In this work, we have used the idea of using both the spatial and temporal information of the videos to detect

and classify the original and deepfakes content. More specifically, a two-stage network namely the C-LSTM approach is proposed where the spatial information from the videos is computed by using several pre-trained models like VGG16, VGG19, ResNet50, Xception-Net, and GoogleNet, DenseNet. Further, we have proposed a novel feature descriptor called the Dense-Swish-Net121. While the Bi-LSTM approach is used to measure the temporal information due to its effectiveness to learn the reliable keypoints and monitor the difference in the behavior of visual characteristics of a sample with the passage of time [10, 11]. Finally, the results are computed based on both the frame level and temporal level information to make the final decision. The following are the main contributions of our work:

- A spatial–temporal aware deepfakes detection framework by using the sequence of consecutive frames from videos is presented to classify the real and fake visual content.
- A novel feature descriptor called Dense-Swish-Net121 is proposed to acquire the more dense visual information of suspected samples.
- A general workflow for deepfakes identification and classification by employing several pre-trained models along with the temporally-aware Bi-LSTM network is presented to tackle the model over-fitting issue.
- Transfer learning-based technique for deepfakes detection in which heatmaps are generated to indicate the explainability of the proposed framework.
- Extensive experimentations including the performance analysis of several state-of-the-art DL-based approaches have been presented over challenging datasets to show the robustness of the proposed solution.

The rest of the paper is structured as follows: the latest research work related to deepfakes detection is explained in Section 2, while the demonstration of the proposed framework is elaborated in Section 3. The evaluation metrics along with the obtained results are discussed in Section 4, while the conclusion is discussed in Section 5.

2 Related work

Due to the devastating effects of deepfakes generation, the research community has focused its attention on the generation of such techniques which can locate the original and fake content. The generic methods used for deepfakes detection are broadly categorized into two types namely the hand-coded-based approaches or DL-based methods. In the case of hand-coded methods, Yang et al. [47] introduced an approach for identifying the manipulated visual content by estimating the 2D facial features for the 3D head pose estimation. The computed features were later used to train the SVM classifier to categorize the original and fake data. This work [47] shows better visual manipulation detection performance, however, the approach lacks to generalize well for the blur samples. Another approach was presented in [21] that employed the Image Quality Metric (IQM) for feature extraction. In the next step, the principal component analysis (PCA) approach was used to minimize the size of feature dimensions. Lastly, the extracted features were used to train the SVM classifier to discriminate between the original and deepfakes videos. The work in [21] works well for visual manipulation detection, however, the performance needs further enhancements. Another approach was introduced in [2] that was trained on the data of several celebrities. In the first step, the visual deepfakes of several celebrities were created by using the GAN approach. Then to locate the real and altered content, the OpenFace2 [7] toolkit was used

to capture the facial landmarks which were later applied for the SVM classifier training. The work shows better deepfakes detection results, however, the approach requires evaluation over a standard and challenging dataset. The hand-coded approaches have demonstrated better deepfakes detection results, however, these methods are not robust to post-processing attacks like under the occurrence of compression, noise, and blurring in the altered content. Moreover, these approaches lack to capture the in-depth information of content due to their limited feature extraction power. Another approach was discussed in [32] where the medical feature descriptor of human faces was computed via using the Dlib tool. The extracted fetaures were passed as keypoints vector to train the SVM, and ANN classifiers. The approach [32] performs well for deepfakes detection, however, the model was unable to locate the Face-Reenactment-based deepfakes.

To overcome the challenges of the hand-coded methods, now the researchers are testing the ability of DL-based approaches for deepfakes detection. Xu et al. [46] proposed a supervised learning approach for identifying the changes made within visual content. The Xception framework along with a supervised constructive loss was used for deep features computation and classification. The work exhibits better deepfakes expandability power, however, the generalization ability of the model needs further evaluations on cross datasets. Another DL-based approach was presented in [20], where the fusion of landmark features with the deep feature was used to classify the original and fake content. The method presented in [20] shows better results for deepfakes detection, however, lacks to perform well for samples with dark light. Roy et al. [37] introduced a DL-based approach for locating the forensic changes made within multimedia content. Three types of networks namely the 3D ResNet, 3D ResNeXt, and I3D were used to compute the deep features and classify the visual content as being original or fake. The work in [37] attains the best performance for the 3D ResNeXt framework, however, exhibits lower performance for the unseen cases. In [41] a DL-based approach was introduced for deepfakes detection. The work used both the spatial and temporal information of videos to discriminate between the actual and manipulated content. The approach [41] shows better deepfakes classification performance, however, the classification accuracy degrades over the compressed video samples. Chen et al. [12] proposed a solution for identifying the deepfakes content from the original video samples. A two-stage model namely mask-guided detection and reconstruction was introduced to locate the manipulated content. Initially, the deep features were computed that were later used in an iterative manner to locate the altered content. The work [12] exhibits better deepfakes detection performance, however, not generalized to all sorts of adversarial attacks. Another approach was presented in [27] where the 3D CNN model was used to detect the deepfakes from the suspected videos. The method [27] works well for visual manipulation detection, however, suffering from a high computational cost. Masood et al. [25] proposed an approach where several pre-trained models were used to compute the deep features of the input videos. In the next step, the computed features were used for the SVM classifier training. The work [25] shows the best results for the DenseNet-169 model, however, at the charge of the increased computational burden.

Zhang et al. [49] discussed an approach for visual manipulation detection from videos by presenting a CNN model by incorporating the error level analysis. This work [49] improves the classification results, however, the approach lacks to show better performance results for highly compressed samples. A DL framework was introduced in [31] where an improved residual framework was proposed to locate the forensic manipulations introduced in the visual samples to spread disinformation. The approach performs well for deepfakes detection and better tackles the sample distortions, however, the generalization ability needs further enhancements. Ilyas et al. [17] introduced an approach called the AVFakeNet

model to locate the audiovisual manipulations from the investigated samples. Another DL approach was discussed in [18] that proposed a graph neural network-based model employing the multi-sized samples attributes to locate the real and fake visual samples. These approaches [17, 18] improved the classification results in the cross-corpus evaluation, however, with enhanced computing burden. Various works employing the ML and DL frameworks have been investigated by researchers for the effective recognition of original and manipulated input samples. However, the increased realism of deepfakes generated data is introducing new challenges which are imposing the need for more accurate frameworks capable of recognizing the altered samples and trace the manipulation signs to reliably verify the authenticity of samples.

3 Proposed method

In this part, we have described the details of the introduced approach. Figure 1 shows the workflow of our technique. The presented technique is comprised of a convolutional RNN model for handling frame sequences. The presented approach namely C-LSTM contains two main components which are as follows: i) the CNN unit that is responsible for computing the deep features at the frame level of videos, ii) a Bi-LSTM model to capture the temporal behavior to perform video sequence analysis over time. In the first component which is the convolutional part of the proposed approach, we have used several pre-trained

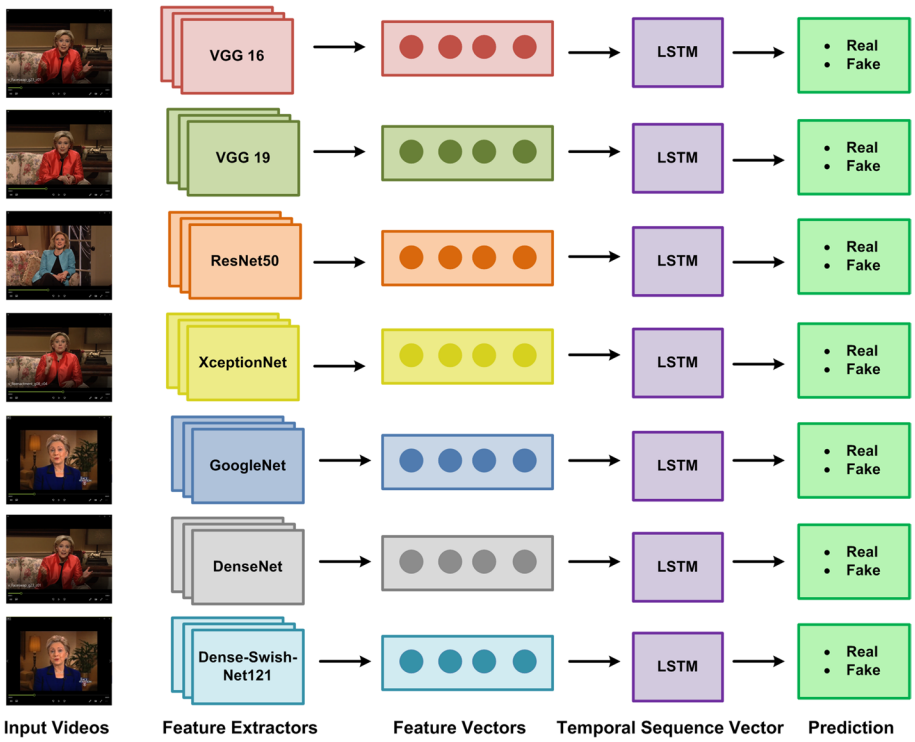

Fig. 1 Workflow of the proposed method

deep learning models namely the VGG16, VGG19, ResNet50, XceptionNet, GoogleNet, and DenseNet. The existing pre-trained models lack to capture the in-depth details of a visual sample due to the usage of the ReLU as the activation approach during the feature engineering phase. To overcome this issue, a novel feature descriptor called Dense-Swish-Net121 is proposed to acquire a more reliable set of visual features. For a given suspected visual sample, a set of discriminative set of features is computed by using various CNN models. Later, we combine the keypoints of several successive frames to use them as input to the Bi-LSTM unit for sequence analysis. Finally, the probability of whether a video is a deepfakes or original is computed to determine the final output.

3.1 C-LSTM

For a given test video (Fig. 1), a C-LSTM model is utilized to generate the temporal sequence feature vector for the visual alteration of the input frames. Employing the concept of end-to-end training, the combination of fully-connected layers is utilized to draw the high-dimensional Bi-LSTM framework to an output classification score. More descriptively, the proposed approach comprises two fully connected layers along with one dropout layer to reduce the model over-fitting problem. The proposed C-LSTM network comprised the CNN phase and a BI-LSTM network. The detailed description is discussed in the subsequent sections.

3.2 CNNs for deep feature computation

In this module of the proposed approach, the deep keypoints from the input video are computed which are later passed to the LSTM module to perform the final classification task (real, deepfake). In this work, we have taken five state-of-the-art DL-based models namely the VGG16, VGG19, ResNet50, XceptionNet, and GoogleNet. The main aim to use the pre-trained networks at the CNN module of the proposed C-LSTM approach is that these networks are trained on huge, openly available datasets like on the ImageNet database, and are robust to compute a more reliable set of features. In the training phase, the starting layers are responsible to learn low-level sample keypoints, whereas, the later layers detect and compute the task-specific features. As the pre-trained models have already gained significant knowledge and learned extensive image texture information, therefore, their training for a new job like employing them for deepfakes detection decreases the model training time and enhances the execution time and speed of these models. Tuning the pre-trained model for a new task is known as 'transfer learning'. A demonstration of transfer learning is shown in Fig. 2. The used frameworks compute reliable features from the video frames like face structure, nose position, eyes, and lips dimension, etc.

3.2.1 VGG

The VGG models like the VGG16 and VGG19 [8] are well-known CNN frameworks presented by the Visual Geometry Group and are known for their simple architecture and competence. The VGG models consist of numerous 3×3 convolutional layers along with the 2×2 max-pooling layers arranged in a sequence to make the models with either depth of 16 or 19. Both networks take the image with the input size of 224×224 and use the 3×3 filters in all convolution layers with a stride rate of 1. While max-pooling

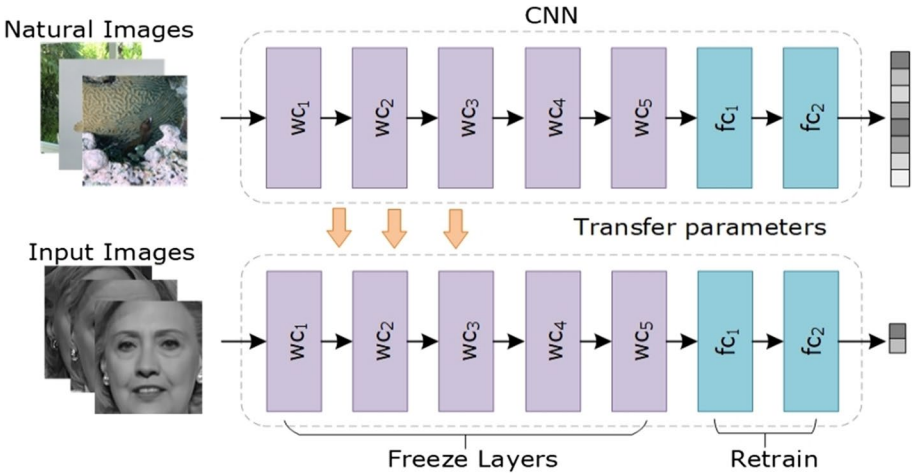

Fig. 2 Visual depiction of transfer learning

functions use the filter size of 2×2 with a stride rate of Moreover all layers use the ReLU method as an activation function which is explained in Eq. 1.

$$F(y) = \max(0, y) \tag{1}$$

Here, $F(y)$ is the output which is mapped to zero when the value of y is less than zero, otherwise it is mapped to y. The visual demonstration of both VGG16 and VGG19 is given in Figs. 3 and 4 respectively.

3.2.2 ResNet50

ResNet50 [42] is a renowned DL approach using skip connections with identity shortcut links that miss several layers for acquiring better classification results. Commonly, the conventional CNN approaches utilize the information from all proceeding layers to enhance the object classification performance. However, such network architectures are suffering from the issue of gradient vanishing during the training procedure [40]. To tackle the problems of such deep networks, the ResNet model presents the idea of employing skip links for deep model structures that skip the few layers and form the base of residual blocks (RBs). The main building block of the ResNet50 model is the RB and a visual demonstration is given in Fig. 5.The RB comprises frequent convolution layers that use the ReLU

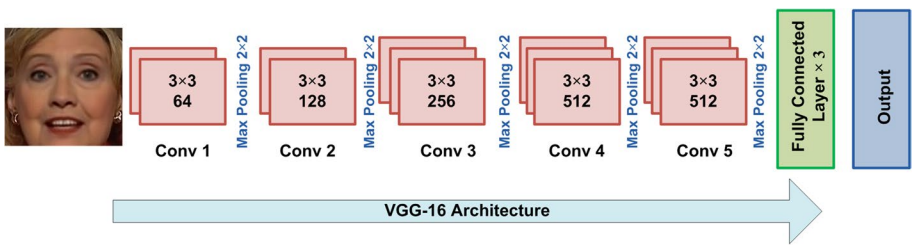

Fig. 3 Visual demonstration of VGG-16 architecture

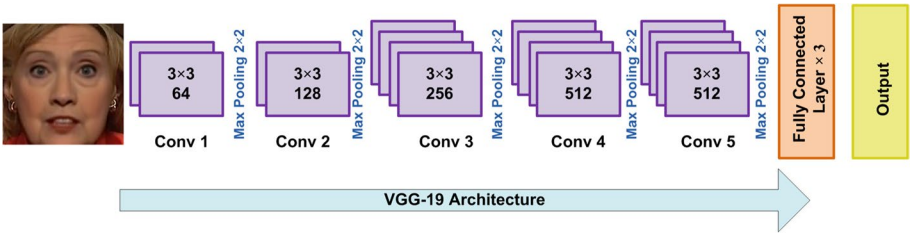

Fig. 4 Visual demonstration of VGG-19 architecture

activation method. Additionally, the ResNet50 model consists of the batch normalization layer together with the shortcut links. For all RBs, the stacked layers perform residual mapping by employing shortcut connections that implement identity mapping (i). The acquired values are combined with the resultant method of the stacked layers. The final result from the RB is expressed as:

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

Here, i shows the input, F demonstrates the residual method whereas Y is explaining the output attained from the residual method.

3.2.3 XceptionNet

Most of the CNN-based models use the concept of increasing the number of convolutional layers to enhance the classification performance. However, such deep network architectures cause to increase in the economic burden and result in model over-fitting problems. To

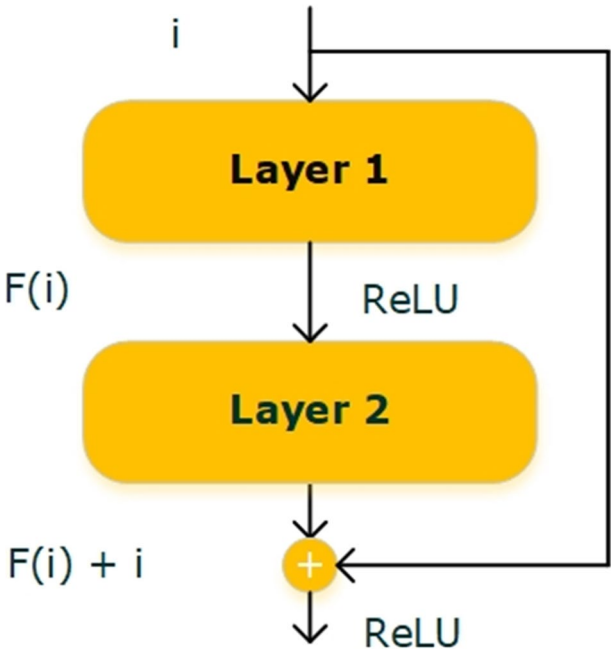

Fig. 5 The structural representation of RB

overcome the challenges of such models lightweight techniques are presented. One such model is the XceptionNet [14] which is presented to enhance the evaluation results both in terms of sample classification and computational time complexity. In this model, the depth-wise independent convolution layers are presented in place of the inception units which employ the point-wise convolutional layers. A set of convolution layers that are depth-wise independent from each other are used on all input samples. The point-wise convolutional layer (filter size of 1×1) maps the result of channels via a depth-wise convolution into the different channel spaces. A visual description of XceptionNet is given in Fig. 6.

3.2.4 GoogleNet

The GoogleNet model [6] is presented by Google organization in 2014 and is an extended form of the Inception model. The GoogleNet framework comprises a total of 22 convolution layers. This model contains fewer model parameters and can learn a more nominative set of frame features as this approach is solid and holds the entire frame at once. The main strength of this model is that it uses several Inception units, which permit it to select among convolutional filters of several sizes within each block. The Inception module holds these units on top of each other and introduced the max-pooling layers with the size of 2 to minimize the feature dimension sizes. A visual description of GoogleNet is given in Fig. 7.

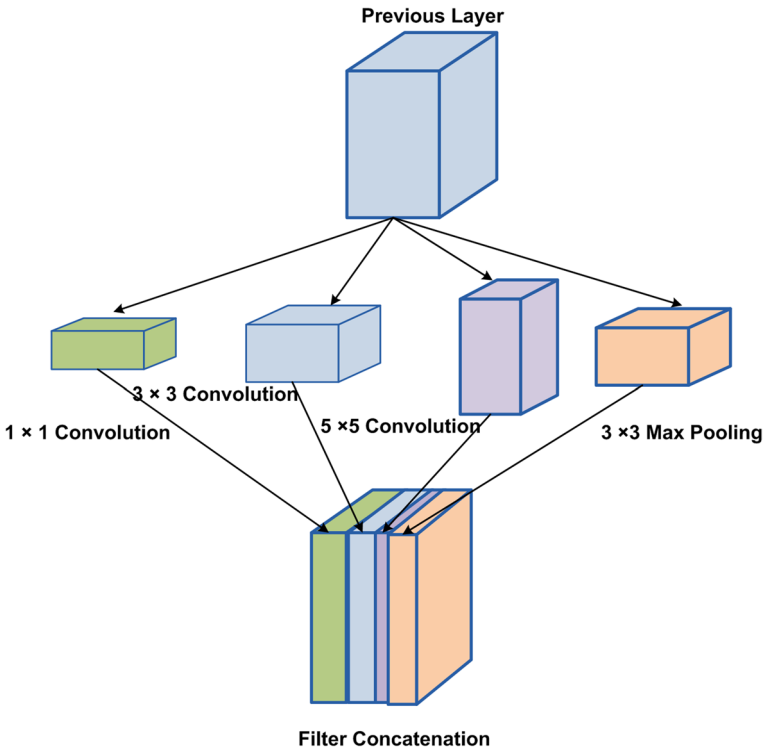

Fig. 6 Visual depiction of XceptionNet

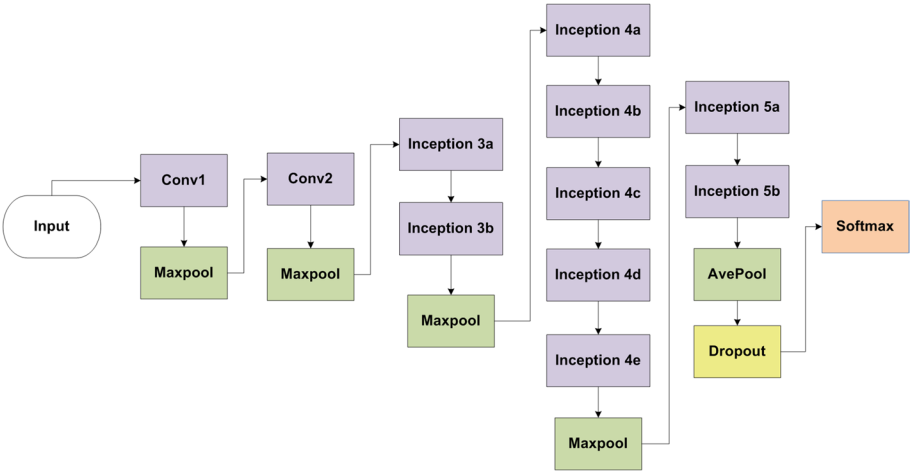

Fig. 7 Description of GoogleNet network

3.2.5 Dense-Swish-Net121

The above-mentioned CNN frameworks are not proficient to fully capture the detailed characteristics of a visual sample as these approaches use the ReLU activation approach which maps all the negative input values to zero during the phase of keypoints computation. Such architectures result in the loss of important details from an examined sample. To overcome such issues, and to incorporate an effective and reliable feature computation strategy for deepfakes detection, we have introduced a novel framework called the Dense-Swish-Net121 approach. We have modified the existing DenseNet-121 framework by introducing a more competent activation approach called the Swish activation method as an alternative to the ReLU method after all convolutional 2-D layers in the network description of the model. The swish activation approach represents the non-monotonic and more smooth behavior with unrestricted above and bounded beneath in the learning curve. Such aspects of the swish approach allow the Dense-Swish-Net121 to upgrade its learning capability and enhance the recognition power of the model by prohibiting the model over-learning issues. This activation technique is less complex by nature, and studies reveal that it exhibits promising results as compared to the ReLU function in accomplishing numerous object categorization tasks [25]. The fundamental cause for this upgraded performance of the swish method over the ReLU approach is that the ReLU activation approach does not allow the propagation of negative scores through the framework during keypoints computation step which results in the elimination of important visual characteristics of the examined samples. To overwhelm the problem of the conventional DenseNet-121 approach, we have employed the swish technique which permits the flow of a few negative computed scores inside the model and results in extracting a more dense and reliable set of sample features at the CNN level.

The Dense-Swish-Net121 approach comprises a total of four dense blocks with 121 layers. In each dense block, all block layers are strongly connected to each other, and sample characteristics are calculated from the previous layers as propagated to coming layers [4]. This method promotes reemployments of visual characteristics and strengthens the data flow throughout the model's structure, making it feasible to incorporate complex

video transformations for the effective recognition of visual modifications. The structural details of the proposed model are provided in Table 1.

The Dense-Swish-Net121 incorporates numerous Convolutional Layer (CnL), Dense Block (DnB), and Transition Layer (TsL). The DnB acts as the major element of Dense-Swish-Net121 as mentioned in Fig. 8. The z0 indicates the input layer incorporating f0 keypoints maps. The Hn(.) represents a mutual method performing 3 jobs which are Batch Normalization (BN), a Swish activation method, and a 3×3 Conv kernel. Each Hn(.) function generates f keypoints maps, forwarded to zn coming layers. As in Dense-Swish-Net121, all coming layers are receiving the visual characteristics calculated from the previous layers which resulted in high feature dimensional space. To tackle this, the TsL are introduced among all DnBs to minimize the keypoints space. For this reason, the Dense-Swish-Net121 comprehends a BN and 1×1 CvL accompanied by an average pooling layer, as elaborated in Fig. 8.

3.3 Temporal analysis using Bi-LSTM

After taking a sequence of deep features from the suspected frames of an input video, the main aim of the Bi-LSTM model is to classify the sequence as being original or manipulated by deploying a 2-node probability-based network. The major intuition to select the Bi-LSTM approach for temporal sequence analysis as compared to the other approach like optical flow field and 3D-CNN is due to its ability to better investigate the behavior of videos with time. As the optical flow contains only short-term motion information, adding it does not enable CNNs to learn long-term motion transitions, while 3D-CNNs compute richer information from a suspected sample, however, these are well suited to 3D image analysis. Comparatively, the Bi-LSTM approach better represents the characteristics of the quality features in time series, and for video forensic analysis, it is mandatory to learn the difference in the visual appearance of the subjects that appeared in the video sample,

Table 1 The architecture of Dense-Swish-Net121

	Layer	Operator	Stride
	Convolutional Layer	7×7 *conv*	2
	Pooling	3×3 avg_pool	2
	DnB1	$\left(\frac{1\times1 \text{ conv}}{3\times3 \text{ conv}}\right) \times 6$	1
TL1	Convolutional Layer	1×1 conv	
	Pooling Layer	2×2 avg_pool	
	DnB2	$\left(\frac{1\times1 \text{ conv}}{3\times3 \text{ conv}}\right) \times 12$	1
TL2	Convolutional Layer	1×1 conv	
	Pooling Layer	2×2 avg_pool	
	DnB3	$\left(\frac{1\times1 \text{ conv}}{3\times3 \text{ conv}}\right) \times 24$	1
TL3	Convolutional Layer	1×1 conv	
	Pooling Layer	2×2 avg_pool	
	DnB4	$\left(\frac{1\times1\text{conv}}{3\times3\text{conv}}\right) \times 16$	1
Classification Layer		7×7 avg_pool FC layer	

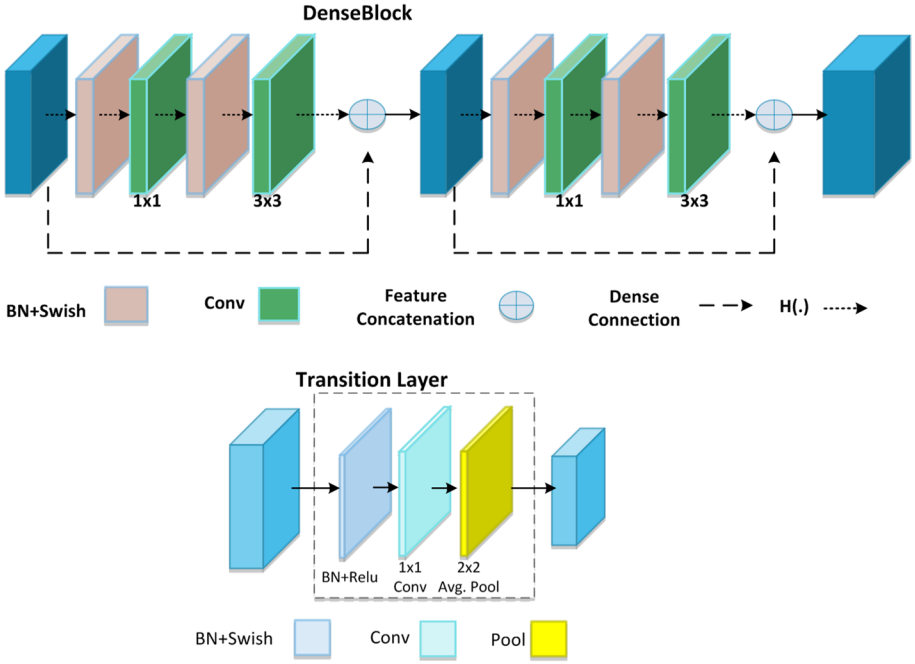

Fig. 8 The visual demonstration of Dense-Swish-Net121 with **a** Dense Block and **b** Transition Block

therefore, we nominated the Bi-LSTM approach for video temporal sequence analysis [9, 23, 24]. The main challenge for the Bi-LSTM model is to work in such an iterative manner that it can reliably process a sequence. To accurately handle this situation, we have employed the 1024-wide Bi-LSTM model with a drop-out layer of 0.5 to correctly attain what we require. Then, 512 fully-connected layers are introduced followed by the softmax layer of size two to calculate the likelihood of a frame sequence being original or deepfakes.

4 Results

In this section, we have discussed the evaluation metrics used to assess the classification performance of the proposed method. Moreover, a detailed description of the used dataset is also discussed. We have presented a detailed experimentation explanation to show the robustness of the proposed approach.

4.1 Evaluation metrics

To assess the deepfakes detection performance of the proposed approach, we have employed several standard metrics namely precision (P_r), recall (R_e), accuracy (A_c), and F1 score. The mathematical explanation of employed metrics is explained in Eqs. 1 to Eq. 4 respectively.

$$P_r = \frac{t'}{t' + r'} \tag{3}$$

$$R_e = \frac{t'}{t' + \eta} \tag{4}$$

$$A_c = \frac{t' + \partial}{t' + \partial + r' + \eta} \tag{5}$$

$$F1 = \frac{2 \times Pr \times Re}{Pr + Re} \tag{6}$$

Here, t' denotes the true positives (deepfakes videos) ∂ and shows the true negatives (real videos). While, r' demonstrates the false positives (negative real), and η shows false negatives (negative deepfakes) respectively.

4.2 Dataset

In the introduced approach, we have employed two challenging databases to check the classification results of all employed models. First, we have utilized a challenging deepfakes database demonstrated in [1] and called the world leaders (WLDR) dataset. The employed database consists of both pristine and deepfakes visual samples of five subjects i.e., Barack Obama, Hillary Clinton, Bernie Sanders, Donald Trump, and Elizabeth Warren. The video samples of all subjects are of varying lengths from 10 s and 2.5 min. Furthermore, the videos are saved at 30 fps using an mp4-format at a relatively high quality of 20. The WLDR dataset was developed by taking video samples from YouTube in which all samples must meet the following requirements: the person of interest must face cameras, and talk during the entire video session. Moreover, it is ensured that the video capturing device is kept stationery and all samples must have a minimum length of 10 s. Further, we have employed another challenging dataset to check the performance of all employed models called the deepfakes detection challenge (DFDC) dataset consists of 1131 real and 4119 manipulated videos of different subjects. The deepfakes samples of the DFDC dataset are generated with 2 unknown methods. This data sample is online available and can be acquired from the Kaggle site [22].

4.3 Implementation details

The network is executed in Matlab 2021 version and runs on Nvidia GTX1070 GPU-based system. The dataset is divided randomly into 70/10/20 parts to produce three separate sets namely the training, validation, and test sets respectively. We have used an equal number of real and fake samples from all subjects to maintain the class balance. We have further performed the following settings to execute the deepfakes detection task:

i) Subtracting channel means from each channel.
ii) For the VGG16, VGG19, ResNet50, GooglNet, DenseNet121, and Dense-Swish-Net121 we have resized the frames to 224-by-224 dimensions, while for the XceptionNet, the video frames are set to the dimension of 299-by-299 as per model requirements.

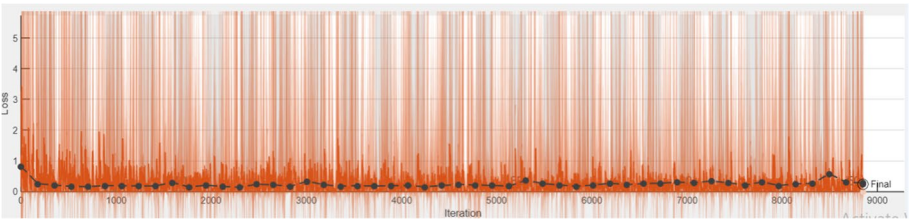

Fig. 9 Visual depiction of loss graph

iii) We have trained the model for 50 epochs and the learning rate is set to 0.0001.

For the presented approach, we have shown the visual representation of the optimal loss graph in Fig. 9. It is quite evident from Fig. 9 that the proposed solution attained an optimal value of 0.00021 at the epoch number of 50, which is showing the effective learning of our approach. Furthermore, we have attained the highest validation accuracy of 98.02% as shown in Fig. 10.

4.4 Results and discussion

In this section, we have performed the evaluation of the proposed model by using several experiments to show its robustness to deepfakes detection on both datasets. Initially, we performed the comparisons of all employed models via two types of experiments. First, we have compared the C-LSRM performance on the entire dataset. After this, we have further demonstrated the results of the proposed approach in terms of both class-wise and subject-wise results to show the in-depth evaluation of the presented technique. Then we have taken some latest approaches to evaluate the highest-performed C-LSTM algorithm against them. The details can be found in subsequent sections.

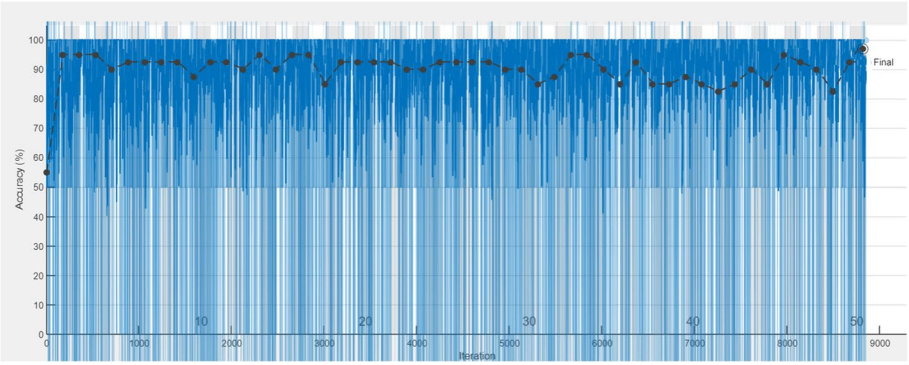

Fig. 10 Training graph representation

4.4.1 Evaluation of pre-trained and Dense-Swish-Net121-based Bi-LSTM models

In this section, we have discussed the obtained deepfakes classification results on both datasets to show the robustness of the proposed approach. We have evaluated the proposed C-LSTM model with several CNN-based approaches namely the VGG16, VGG19, ResNet50, XceptionNet, GoogleNet, DenseNet-121, and Dense-Swish-Net121. Several videos from the employed datasets are used to check the deepfakes detection performance of C-LSTM with mentioned CNN frameworks. We have used the standard evaluation metrics used in the field of video forensic analysis.

First, we have discussed the results attained for the WLDR database, and obtained results are shown in Fig. 11. The deepfakes detection performance in terms of accuracy evaluation metric for all CNN-based approaches for the C-LSTM framework are exhibited in Fig. 11a. It is quite evident from Fig. 11a that we have attained the best accuracy for the Dense-Swish-Net121-based Bi-LSTM approach with a value of 98.72%, while the second-highest classification accuracy value is shown by the DenseNet121-based Bi-LSTM approach. Moreover, the VGG16-based LSTM approach shows the lowest accuracy value of 90.02%.

In the field of multimedia forensics, the cost of misclassifying the forged content as real is much larger than the misclassification of the original sample as deepfakes. As mostly such content is used for legal claims investigation where a little mistake in classification can cause adverse damage to the victim. Hence, the main aim of the deepfakes recognition model is to reduce the rates of false negatives. To evaluate the model for this, we have calculated the recall rate of the C-LSTM approach for all CNN models,

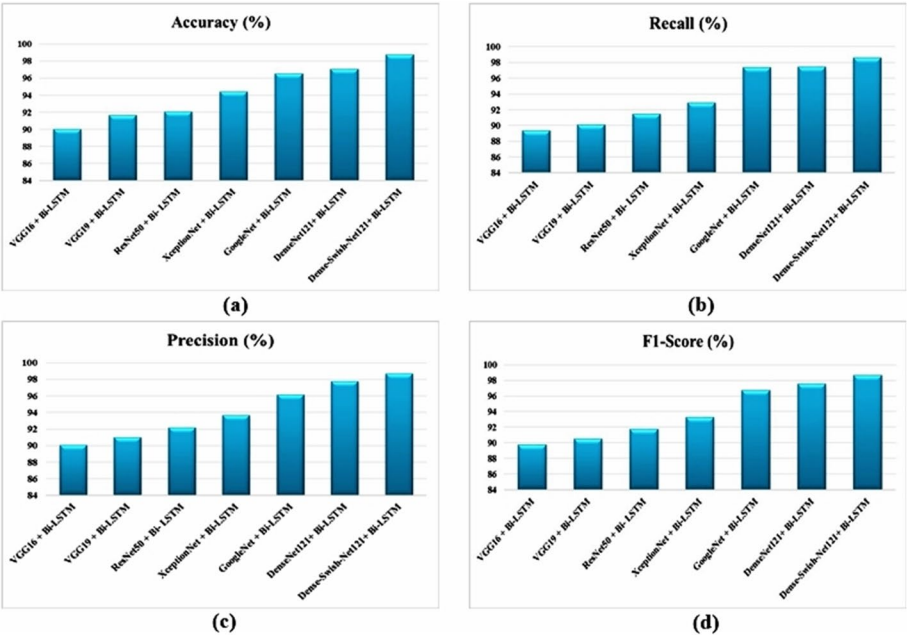

Fig. 11 Evaluation of the pre-trained and Dens-Swish-Net121-based Bi-LSTM models on the WLDR dataset

and obtained results are shown in Fig. 11b. The Dense-Swish-Net121-based Bi-LSTM approach shows the less false negative rates and attains the highest value of recall evaluation metric stated as 98.63%. While the lowest recall value is exhibited by the VGG16-based Bi-LSTM approach with a value of 89.37%.

Another main objective of deepfakes detection approaches is to minimize the false positive rate as well which ultimately causes to reduce the chances of misclassification of original content as deepfakes. For this reason, we have evaluated the C-LSTM model for all used CNN frameworks by computing their precision, and the acquired results are shown in Fig. 11c. In the case of precision evaluation metric, again the Dense-Swish-Net121-based Bi-LSTM approach shows the highest value of 98.68%, while the second better result is demonstrated by the DenseNet121-based Bi-LSTM approach with the value of 97.75%. The VgGG16 and VGG19-based LSTM models exhibit the occurrence of more false positives values and attain precision values of 90.07%, and 90.98% respectively.

To further assess the deepfakes detection performance of the proposed approach, we have computed the F1-score as it better elaborates the recognition power of the model. The higher a model shows the F1 score, the higher its recognition ability. The obtained F1 score for all employed CNN models with the Bi-LSTM approach is shown in Fig. 11d. The largest and lowest F1 scores are attained by the Dense-Swish-Net121-based Bi-LSTM and VGG16-based BiLSTM approaches with the values of 98.72%, and 89.72% respectively.

Next, we measured the performance of all models with the Bi-LSM approach for the DFDC dataset, and obtained values for all performance metrics are given in Table 2. For all measures, all employed DL frameworks perform effectively in recognizing the visual manipulations which are indicating the robustness of such spatiotemporal frameworks for locating the signs of alterations introduced in videos to spread false information. More in-depth analysis, we can see from the values given in Table 2 that the new proposed approach named the Dense-Swish-Net121 along with the Bi-LSTM framework shows the highest results for all performance metrics. The Dense-Swish-Net121 approach with the Bi-LSTM framework has attained an accuracy of 98.11%, with precision, recall, and F1 values of 97.98%, 97.97%, and 97.97% respectively that is clearly signifying its effectiveness for deepfakes detection. The DenseNet121 and GoogleNet models also perform better with the Bi-LSTM technique and attained accuracy values of 98.11%, and 97.91%. Further, the VGG16-based Bi-LSTM approach shows the least performance results with accuracy, precision, recall, and F1 of 89.64%, 88.12%, 87.74%, and 87.93%.

Table 2 Performance comparison of all DL models with the Bi-LSTM approach over the DFDC dataset

Models	DFDC			
	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
VGG16 + Bi-LSTM	89.64	88.12	87.74	87.93
VGG-19 + Bi-LSTM	90.03	87.99	86.82	87.40
ResNet50 + Bi-LSTM	97.88	96.08	96.66	96.37
XceptionNet + Bi-LSTM	97.91	96.61	96.59	96.60
GoogleNet + Bi-LSTM	97.99	96.97	95.85	96.41
DenseNet121 + Bi-LSTM	98.11	97.98	97.97	97.97
Dense-Swish-Net121 + Bi-LSTM	99.31	99.24	98.35	98.79

Table 3 Architectural comparison of employed CNN models

Model	Layers	Parameters (Million)
VGG16	16	138
VGG19	19	144
ResNet50	50	25.6
XceptionNet	71	22.9
GoogleNet	22	7
DenseNet121	121	7.2
Dense-Swish-Net121	121	7.2

Moreover, we have compared the CNN models in terms of architecture, and the comparison is shown in Table 3. It is quite clear from Table 3 that the VGG19 model has the highest number of model parameters 138 million. Whereas, GoogleNet, DenseNet121, and Dense-Swish-Net121 frameworks have the lowest model parameters at 7 million only. Therefore, based on the results obtained by all CNN models with the Bi-LSTM approach (Fig. 11), and the model description demonstrated in Table 3, we can conclude that the Dense-Swish-Net121-based Bi-LSTM approach is more robust to deepfakes detection of both in terms of model complexity and classification results. The distinguishing characteristic of the Dense-Swish-Net121-based Bi-LSTM that allows it to attain the highest classification results is the inclusion of the swish activation method which enables the approach to learn a more competent set of visual characteristics under varying sample capturing conditions and better recognize the alterations of videos.

Moreover, we have presented the heatmaps with the help of Grad-Cam [38] corresponding to the last layer of all pre-trained models and proposed Dense-Swish-Net121 to visualize the inner working of all frameworks. The red color in Fig. 12 is signifying the potential

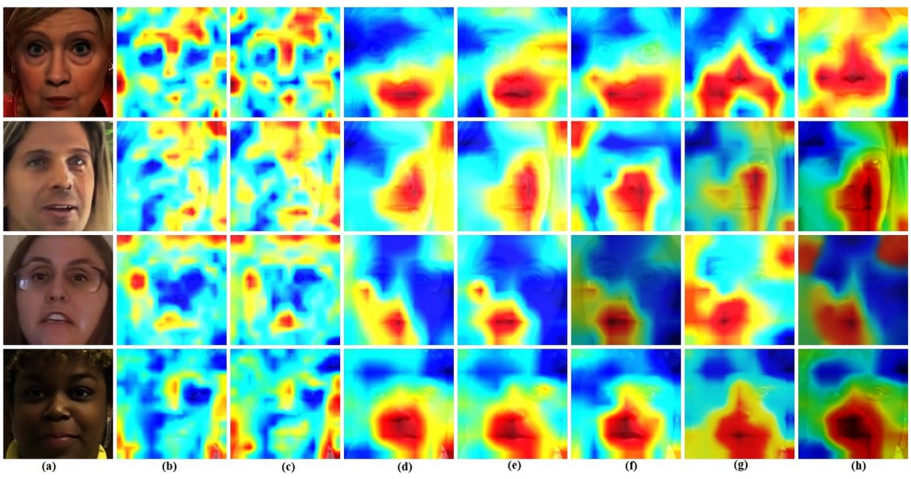

Fig. 12 Visual representation of heatmaps where **a** presents sample, **b** shows heatmaps for VGG-16, **c** for VGG-19, **d** for ResNet50, **e** for XceptionNet, **f** for GoogleNet, **g** for DenseNet121, and **h** for Dense-Swish-Net121

Table 4 Class-wise comparative analysis of proposed approaches over the WLDR dataset

Model	Real (Accuracy %)	Fake (Accuracy %)
VGG16 + Bi-LSTM	91.04	89.01
VGG19 + Bi-LSTM	92.66	90.61
ResNet50 + Bi-LSTM	92.24	91.89
XceptionNet + Bi-LSTM	94.86	93.89
GoogleNet + Bi-LSTM	96.6	96.47
DenseNet121 + Bi-LSTM	97.34	96.74
Dense-Swish-Net121 + Bi-LSTM	98.97	98.47

areas where such manipulations are introduced. From Fig. 12, it is quite evident that among all approaches, the Dense-Swish-Net121 is more focused on the altered regions of visual samples where such modifications are introduced in the human faces and clearly prove the robustness of our approach. So, based on the visual results, we can say that the major cause for the improved classification results of the Dense-Swish-Net121-based Bi-LSTM is due to its effective keypoints extraction capability which enhances its recognition power to discriminate the real and fake samples.

4.4.2 Class-wise evaluation

Next, in this section, we have reported the class-wise performance of all employed DL approaches with the Bi-LSTM technique for both the WLDR and DFDC datasets to further provide a detailed comparison of all approaches.

Initially, the results of the C-LSTM approach are indicated for the WLDR dataset by performing two types of experiments. Initially, we evaluated the class-wise manipulation detection results of the C-LSTM model with all employed CNN-based approaches namely the VGG16, VGG19, ResNet50, XceptionNet, GoogleNet, DenseNet121, and Dense-Swish-Net121. To show the class-wise evaluation performance for all CNN-based Bi-LSTM models, we have selected the accuracy metric as it is a standard metric employed in the area of the image classification field and the obtained results are shown in Table 4. It is quite evident that all employed CNN models with the Bi-LSTM approach are proficient to detect both real and fake data. The highest results are reported by the

Table 5 Subject-wise comparative analysis of proposed approaches

Model	BO	HC	BS	DT	EW
	AUC				
VGG16 + Bi-LSTM	0.94	0.90	0.91	0.89	0.88
VGG19 + Bi-LSTM	0.92	0.93	0.91	0.92	0.9
ResNet50 + Bi-LSTM	0.93	0.92	0.93	0.91	0.91
XceptionNet + Bi-LSTM	0.95	0.94	0.94	0.96	0.98
GoogleNet + Bi-LSTM	0.99	0.98	0.97	0.97	0.99
DenseNet121 + Bi-LSTM	0.99	0.99	0.97	0.98	0.99
Dense-Swish-Net121 + Bi-LSTM	1	1	1	0.99	0.99

Dense-Swish-Net121-based Bi-LSTM approach for both the original and manipulated classes. Whereas, the rest of the models have also exhibited comparative results.

We have performed another analysis on the WLDR dataset, in which we have compared the performance of all CNN-based Bi-LSTM models in terms of all five subjects mentioned in the dataset description. For this reason, we have taken the AUC metric and obtained results are shown in Table 5. The results demonstrated in Table 5 are clearly depicting that the C-LSTM approach is capable of differentiating several subjects effectively which is showing the recognition power of our approach. Moreover, the results are clearly showing that the Dense-Swish-Net121-based Bi-LSTM technique has a better recall ability to differentiate and recognize the various subject with high proficiency. The second highest results are depicted by the DenseNet121-based Bi-LSTM model. While the other approaches also show better results.

Next, we have discussed the classification results for all networks over the DFDC dataset to check the recognition ability of all C-LSTM modules in differentiating the real and fake videos. The attained accuracy values for all models are given in Table 6 which clearly depicts the accurateness of all frameworks. In a precise way, the highest class-wise results are shown by the Dense-Swish-Dense121 approach along with the Bi-LSTM framework to categorize both real and fake videos with scores of 99.38%, and 99.24%. Whereas, the Bi-LSTM-oriented DenseNet-121 and GoogleNet approaches also perform effectively, where the initial model reports accuracy scores of 98.21%, and 98.01% for the original and altered videos samples which are 98.18%, and 98.80% for the later approach and clearly indicating the efficacy of spatiotemporal-based sequence analysis for deepfakes detection. The VGG16-based Bi-LSTM framework shows the lowest classification results with accuracy scores of 90.23%, and 89.05%.

4.4.3 Comparison with state-of-the-art

The results reported in the above sections are clearly showing that the Dense-Swish-Net121-based Bi-LSTM approach has shown the highest manipulation detection results in comparison to all other employed models. Therefore, to compare the deepfakes detection results with other latest approaches, we have chosen the C-LSTM model with the Dense-Swish-Net121-based network. We have evaluated the results of our approach with several new techniques for both datasets named the WLDR and DFDC datasets.

Initially, we have compared the results of our approach attained with the Dense-Swish-Net121-based Bi-LSTM approach for new studies in terms of the WLDR dataset. For this

Table 6 Class-wise comparative analysis of proposed approaches over the DFDC dataset

Model	Real (Accuracy %)	Fake (Accuracy %)
VGG16 + Bi-LSTM	90.23	89.05
VGG19 + Bi-LSTM	90.58	89.48
ResNet50 + Bi-LSTM	98.00	97.76
XceptionNet + Bi-LSTM	98.05	97.78
GoogleNet + Bi-LSTM	98.18	97.8
DenseNet121 + Bi-LSTM	98.21	98.01
Dense-Swish-Net121 + Bi-LSTM	99.38	99.24

Table 7 Subject-wise comparative analysis of Dense-Swish-Dense121-based Bi-LSTM model with the latest approaches over the WLDR dataset

Subject	AUC			TPR		
	[1]	[32]	Proposed	[1]	[32]	Proposed
BO	0.99	0.98	1	0.97	0.99	0.99
HC	0.95	1	1	0.89	0.94	0.96
BS	0.96	1	1	0.92	0.93	0.98
DT	0.90	0.99	0.99	0.74	0.80	0.98
EW	0.98	0.99	0.99	0.92	0.94	0.98

reason, we have performed two types of performance evaluations, where initially, we compared the results of our method in terms of all subjects, and then, we compared our results with new techniques in terms of entire dataset results. For the subject-wise evaluation of the proposed approach, we have selected the study given in [1, 32] and obtained results are shown in Table 7. We have selected two evaluation metrics namely AUC and TPR for this reason. It is quite evident from the results shown in Table 7 that our work is more accurate in terms of both the AUC and TPR as compared to the works presented in [1, 32]. More descriptively, the works in [1, 32] shows an average AUC value of 0.974 which is 0.996 for our case, hence presenting an average performance gain of 2.2%. Similarly, for the TPR metric, the approach in [1, 32] shows an average value of 0.904 which is 0.978 for our work. Therefore, for the TPR evaluation measure, we have acquired an average performance gain of 7.4% which is clearly demonstrating the effectiveness of our work.

To further assess the deepfakes detection performance of our work over the WLDR dataset, we have compared the obtained results against other well-known DL-based approaches namely ResNet [28] and InceptionNet approach, the acquired comparison is exhibited in Fig. 13. Figure 13 is clearly showing that our technique has outperformed the other methods. More clearly, the comparative methods show an average TPR value of 88.45% which is 97.80% for our method. Hence, we have provided an average performance gain of 9.35%.

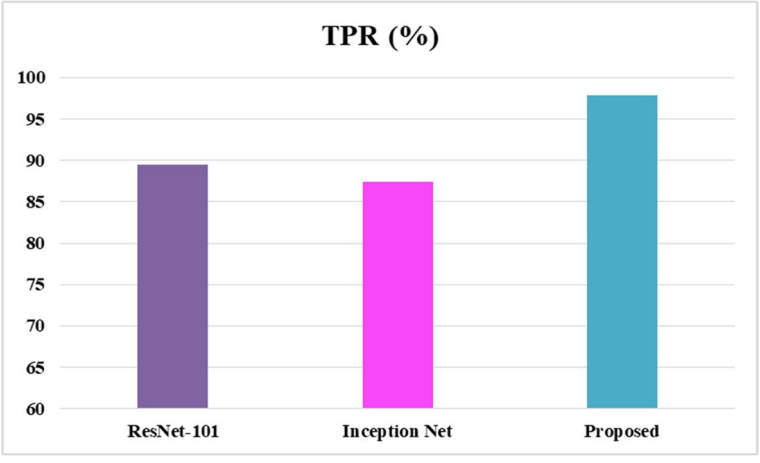

Fig. 13 Comparision of Dense-Swish-Dense121-based Bi-LSTM model with state-of-the-art approaches over the WLDR dataset

The major reason for the better performance of the proposed solution is because of the better face recognition ability of the Dense-Swish-Net121-based Bi-LSTM model which assists in effectively detecting real and manipulated faces.

Next, the performance results attained with the Dense-Swish-Net121-based Bi-LSTM approach are compared with several approaches [13, 15, 16, 31, 36, 43] for the DFDC data sample, and obtained comparison is given in Table 8. The scores in Table 8 are proving that we have attained the highest results as compared to the techniques given in [13, 15, 16, 36, 43] for all reported measures. Ranjan et al. [36] used a DL approach for videos-based deepfakes classification and reported an accuracy score of 84.70%, while the work in [13] utilized both the pixel and temporal information of video samples and reported an accuracy number of 97.94%. While the method [43] has secured the AUC of 92.44%, whereas, the approaches in [15, 16, 31] are showing accuracy numbers of 95.42%, 94.40%, and 99.26% respectively, In comparison, we have exhibited the highest accuracy and AUC values of 99.31%, and 99.39%. In a more brief manner, the comparative approaches have shown average accuracy and AUC scores of 94.34%, and 97.22%, while, we have shown average values for accuracy, and AUC measures with numbers of 99.31%, and 99.39% and reported performance gains of 4.97%, and 2.17%.

The performance analysis performed on both challenging datasets in comparison to other latest approaches has clearly proven the proficiency of our approach in better recognizing the manipulated visual samples. The leading attribute of the Dense-Swish-Net121 approach to propagate the negative scores in the process of features computation allows it to extract a more dense and nominative group of visual characteristics which causes to enhance the classification results of our approach in comparison to other comparative techniques.

5 Conclusion

In this work, we have presented a DL-based approach namely C-LSTM to detect the real or deepfakes samples from input videos. More descriptively, we have employed both the spatial and temporal information of the visual samples to locate the forensic changes. We have used several CNN models namely VGG16, VGG19, ResNet50, XceptionNet, GoogleNet, and DenseNet121 to compute the frame-level information. Further, a novel DL-based feature extractor named the Dense-Swish-Net121 is also presented. While for the temporal sequence analysis, the Bi-LSTM approach is used. For performance evaluation, we have utilized two challenging datasets named the WLDR, and DFDC. We have gained

Table 8 A comparative analysis of the Dense-Swish-Dense121-based Bi-LSTM model with the latest approaches over the DFDC dataset

Method	Accuracy (%)	AUC (%)
[36]	84.70	-
[13]	97.94	-
[43]	-	92.44
[15]	95.42	98.93
[16]	94.40	98.20
[31]	99.26	99.31
Proposed	99.31	99.39

the highest accuracy for the Dense-Swish-Net121-based Bi-LSTM approach with values of 98.72%, and 99.31% over the WLDR, and DFDC databases respectively. In the future, we plan to extend the approach to other challenging datasets and test other CNN models with the Bi-LSTM approach to further improve the deepfakes detection performance.

Acknowledgements This work was supported by grant of Punjab HEC of Pakistan via Award No. (PHEC/ARA/PIRCA/20527/21). We would like to thank Prof. Hany Farid from the University of California Berkeley to provide us with their World Leaders Dataset for performance evaluation

Declarations

Conflict of interest The authors declare that there is no conflict of interest between them.

References

1. Agarwal S, Farid H, Gu Y, He M, Nagano K, Li H(2019) Protecting World Leaders Against Deep Fakes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 38–45
2. Agarwal S, Farid H, Gu Y, He M, Nagano K, Li H (2019) Protecting World Leaders Against Deep Fakes. In CVPR workshops, vol. 1
3. Albahli S, Nawaz M (2022) DCNet: DenseNet-77-based CornerNet model for the tomato plant leaf disease detection and classification. Front Plant Sci, 13
4. Albahli S, Nazir T, Irtaza A, Javed A (2021) Recognition and Detection of Diabetic Retinopathy Using Densenet-65 Based Faster-RCNN. Comput Mater Contin 67:1333–1351
5. Albattah W, Nawaz M, Javed A, Masood M, Albahli S (2022) A novel deep learning method for detection and classification of plant diseases. Compl Intell Syst, pp. 1–18
6. Ballester P, Araujo RM (2016) On the performance of GoogLeNet and AlexNet applied to sketches. In Thirtieth AAAI Conference on Artificial Intelligence
7. Baltrušaitis T, Robinson P, Morency L-P (2016) Openface: an open source facial behavior analysis toolkit. In 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1–10: IEEE
8. Carvalho T, De Rezende ER, Alves MT, Balieiro FK, Sovat RB (2017) Exposing computer generated images by eye's region classification via transfer learning of VGG19 CNN. In 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 866–870: IEEE
9. Chen C, Li S, Wang Y, Qin H, Hao A (2017) Video saliency detection via spatial-temporal fusion and low-rank coherency diffusion. IEEE Trans Image Process 26(7):3156–3170
10. Chen C, Wang G, Peng C, Fang Y, Zhang D, Qin H (2021) Exploring rich and efficient spatial temporal interactions for real-time video salient object detection. IEEE Trans Image Process 30:3995–4007
11. Chen C, Wang G, Peng C, Zhang X, Qin H (2019) Improved robust video saliency detection based on long-term spatial-temporal information. IEEE Trans Image Process 29:1090–1100
12. Chen Z, Xie L, Pang S, He Y, Zhang B (2021) MagDR: Mask-guided Detection and Reconstruction for Defending Deepfakes. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9014–9023
13. Chintha A, Rao A, Sohrawardi S, Bhatt K, Wright M, Ptucha R (2020) Leveraging edges and optical flow on faces for deepfake detection. In 2020 IEEE international joint conference on biometrics (IJCB), pp. 1–10: IEEE
14. Chollet F (2017) Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258
15. Ganguly S, Ganguly A, Mohiuddin S, Malakar S, Sarkar R, (2022) ViXNet: Vision Transformer with Xception Network for deepfakes based video and image forgery detection. Expert Syst Appl. 118423
16. Hernandez-Ortega J, Tolosana R, Fierrez J, Morales A (2020) Deepfakeson-phys: Deepfakes detection based on heart rate estimation. arXiv preprint arXiv:.00400
17. Ilyas H, Javed A, Malik KM (2023) AVFakeNet: A unified end-to-end Dense Swin Transformer deep learning model for audio–visual deepfakes detection. Appl Soft Comput 136:110124
18. Khalid F, Javed A, Ilyas H, Irtaza A (2023) DFGNN: An interpretable and generalized graph neural network for deepfakes detection. Expert Syst Appl 222:119843

19. Kohli A, Gupta A (2021) Detecting DeepFake, FaceSwap and Face2Face facial forgeries using frequency CNN. Multimed Tools Appl 80(12):18461–18478
20. Kolagati S, Priyadharshini T, Rajam VMA (2022) Exposing deepfakes using a deep multilayer perceptron–convolutional neural network model. Int J Inf Manage Data Insights 2(1):100054
21. Korshunov P, Marcel S (2018) Deepfakes: a new threat to face recognition? assessment and detection. *arXiv preprint arXiv:.08685*
22. Dolhansky et al., (2020) The DeepFake Detection Challenge Dataset. *arXiv preprint* arXiv:2006.07397
23. Li Y, Li S, Chen C, Hao A, Qin H (2019) Accurate and robust video saliency detection via self-paced diffusion. IEEE Trans Multimedia 22(5):1153–1167
24. Li Y, Li S, Chen C, Hao A, Qin H (2020) A plug-and-play scheme to adapt image saliency deep model for video data. IEEE Trans Circuits Syst Vid Technol 31(6):2315–2327
25. Masood M, Nawaz M, Javed A, Nazir T, Mehmood A, Mahum R (2021) Classification of Deepfake Videos Using Pre-trained Convolutional Neural Networks. In 2021 International Conference on Digital Futures and Transformative Technologies (ICoDT2), pp. 1–6: IEEE
26. Masood M, Nawaz M, Malik KM, Javed A, Irtaza A, Malik H (2023) Deepfakes Generation and Detection: State-of-the-art, open challenges, countermeasures, and way forward. Appl Intell 53(4):3974–4026
27. Mehta V, Gupta P, Subramanian R, Dhall A (2021) FakeBuster: A DeepFakes Detection Tool for Video Conferencing Scenarios. In 26th International Conference on Intelligent User Interfaces, pp. 61–63
28. Nawaz et al., (2021) Melanoma localization and classification through faster region-based convolutional neural network and SVM. Multimed Tools Appl, pp. 1–22
29. Nawaz M et al (2021) Single and multiple regions duplication detections in digital images with applications in image forensic. J Intell Fuzzy Syst 40(6):10351–10371
30. Nawaz M et al (2021) Image Authenticity Detection Using DWT and Circular Block-Based LTrP Features. CMC-Comput Mater Cont 69(2):1927–1944
31. Nawaz M, Javed A, Irtaza A (2022) ResNet-Swish-Dense54: a deep learning approach for deepfakes detection. Vis Compu, pp. 1–22
32. Nawaz M, Masood M, Javed A, Nazir T (2022) FaceSwap based DeepFakes Detection. Int Arab J Inf Technol 19(6):891–896
33. Nazir T, Irtaza A, Javed A, Malik H, Mehmood A, Nawaz M (2021) Digital Image Forensic Analysis using Hybrid Features. In 2021 International Conference on Artificial Intelligence (ICAI), pp. 33–36: IEEE.
34. Nazir T, Nawaz M, Masood M, Javed A (2022) Copy move forgery detection and segmentation using improved mask region-based convolution network (RCNN). Appl Soft Comput 131:109778
35. Nguyen TT, Nguyen CM, Nguyen DT, Nguyen DT, Nahavandi S (2019) Deep learning for deepfakes creation and detection: A survey. *arXiv preprint arXiv:.11573*
36. Ranjan P, Patil S, Kazi F (2020) Improved generalizability of deep-fakes detection using transfer learning based CNN framework. In 2020 3rd international conference on information and computer technologies (ICICT), pp. 86–90: IEEE
37. Roy R, Joshi I, Das A, Dantcheva A (2022) 3D CNN Architectures and Attention Mechanisms for Deepfake Detection. ed
38. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017) Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE international conference on computer vision. pp. 618–626
39. Shelke NA, Kasana SS (2021) A comprehensive survey on passive techniques for digital video forgery detection. Multimed Tools Appl 80(4):6247–6310
40. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint* arXiv:1409.1556
41. Sun Z, Han Y, Hua Z, Ruan N, Jia W (2021) Improving the Efficiency and Robustness of Deepfakes Detection through Precise Geometric Features. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3609–3618
42. Theckedath D, Sedamkar R (2020) Detecting affect states using VGG16, ResNet50 and SE-ResNet50 networks. SN Comput Sci 1(2):1–7
43. Trinh L, Tsang M, Rambhatla S, Liu Y (2021) Interpretable and trustworthy deepfake detection via dynamic prototypes. In Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp. 1973–1983
44. Uga B (2019) Towards Trustworthy AI: A proposed set of design guidelines for understandable, trustworthy and actionable AI," ed

45. Wang G, Chen C, Fan D-P, Hao A, Qin H (2021) From semantic categories to fixations: A novel weakly-supervised visual-auditory saliency detection approach. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 15119–15128.
46. Xu Y, Raja K, Pedersen M (2022) Supervised Contrastive Learning for Generalizable and Explainable DeepFakes Detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 379–389
47. Yang X, Li Y, Lyu S (2019) Exposing deep fakes using inconsistent head poses. In ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 8261–8265: IEEE
48. Zhang T (2022) Deepfake generation and detection, a survey. Multimed Tools Appl, pp. 1–18
49. Zhang W, Zhao C, Li Y (2020) A novel counterfeit feature extraction technique for exposing face-swap images based on deep learning and error level analysis. Entropy 22(2):249