Deepfakes Examiner: An End-to-End Deep Learning Model for Deepfakes Videos Detection

Hafsa Ilyas
Software Engineering Department
University of Engineering and Technology
Taxila, Pakistan
hafsailyas97@gmail.com

Ali Javed
Software Engineering Department
University of Engineering and Technology
Taxila, Pakistan
ali.javed@uettaxila.edu.pk

Khalid Mahmood Malik
Computer Science and Engineering Department
Oakland University Rochester
MI, USA
mahmood@oakland.edu

Aun Irtaza
Computer Science Department
University of Engineering and Technology
Taxila, Pakistan
aun.irtaza@uettaxila.edu.pk

2022 16th International Conference on Open Source Systems and Technologies (ICOSST) | 978-1-6654-6477-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICOSST57195.2022.10016871

Abstract—**Deepfakes generation approaches have made it possible even for less technical users to generate fake videos using only the source and target images. Thus, the threats associated with deepfake video generation such as impersonating public figures, defamation, and spreading disinformation on media platforms have increased exponentially. The significant improvement in the deepfakes generation techniques necessitates the development of effective deepfakes detection methods to counter disinformation threats. Existing techniques do not provide reliable deepfakes detection particularly when the videos are generated using different deepfakes generation techniques and contain variations in illumination conditions and diverse ethnicities. Therefore, this paper proposes a novel hybrid deep learning framework, InceptionResNet-BiLSTM, that is robust to different ethnicities and varied illumination conditions, and able to detect deepfake videos generated using different techniques. The proposed InceptionResNet-BiLSTM consists of two components: customized InceptionResNetV2 and Bidirectional Long-Short Term Memory (BiLSTM). In our proposed framework, faces extracted from the videos are fed to our customized InceptionResNetV2 for extracting frame-level learnable features. The sequences of features are then used to train a temporally aware BiLSTM to classify between the real and fake video. We evaluated our proposed approach on the diverse, standard, and largescale FaceForensics++ (FF++) dataset containing videos manipulated using different techniques (i.e., DeepFakes, FaceSwap, Face2Face, FaceShifter, and NeuralTextures) and the FakeAVCeleb dataset. Our method achieved an accuracy greater than 90% on DeepFakes, FaceSwap, and Face2Face subsets. Performance and generalizability evaluation highlights the effectiveness of our method for detecting deepfake videos generated through different techniques on diverse FF++ and FakeAVCeleb datasets.**

Keywords—Bidirectional LSTM, Deepfakes Detection, FaceForensics++, FakeAVCeleb, InceptionResNetV2, Puppet-master, Face-swap.

I. Introduction

Deepfakes is the term used to represent manipulated videos or images generated using deep learning (DL) techniques such as Auto Encoders (AE) and Generative Adversarial Networks (GANs). Deepfake videos can be of identity swap, lip-synching, or puppet mastery types. In identity swap, a fake video is created by swapping the face of the source person with that of the target while retaining the expressions and background of the source person. Lip-synching refers to the techniques where the mouth of the target person is driven according to some arbitrary audio. In puppet mastery, the expressions, eye, and head movement of the target person is swapped with that of the source person. Such deepfakes techniques can be used to depict famous people performing and saying things they never did and said in order to tarnish their reputations [4].

Detection of deepfake videos and images has now become an important research area in the field of digital media forensics. In recent years, deepfakes generation techniques have achieved such a level of advancement that it becomes difficult for humans to identify fake content. Moreover, the accessibility of open-source tools (such as FaceSwap [1], DeepFaceLab [2]) and applications (i.e., Reface [3], Reflect, FakeApp) enables the users to generate manipulated videos without having any technical knowledge. So, such publicly available tools and applications make it easy to create and spread false information in cyberspace and thus lead to the loss of the public's trust and confidence in social media content [4].

Several works have been proposed for the detection of deepfakes to overcome the threats such as false pornography, disinformation, and defamation. But deepfakes detection is still a challenging task as the fake videos include temporal features that differ across the frames thus making it difficult to detect them. Moreover, frame-level visuals are also becoming more realistic due to a slight imperceptible modification in each frame. The detection of tiny modifications in each frame of a fake video is challenging. Also, various deepfakes detection methods are evaluated on fragmented datasets [5]. For instance, considering the FaceForensics++ (FF++) dataset, methods such as [14,16,18,25] are not evaluated on all the available subsets of the dataset. So, there is still a need for the development of effective and efficient deepfakes detection methods that can accurately detect the fake videos generated using different techniques.

To address the above-mentioned limitations of the existing methods including computational complexity and evaluation on fragmented datasets, we introduce a hybrid deep learning

978-1-6654-6477-2/22/$31.00 ©2022 IEEE

model called InceptionResNet-BiLSTM that employs the customized InceptionResNetV2 as a front-end feature extractor and Bidirectional Long-Short Term Memory (BiLSTM) network as a back-end classifier. Our proposed model is robust to the variation in illumination conditions, different ethnicities and also addresses intra-frame, temporal, and visual inconsistencies among the frames in a video. In our proposed model, we extract the features from the frames of the videos using our customized InceptionResNetV2 and then pass the feature vectors to the temporally aware Bidirectional LSTM, which simulates the class dependency in forward and backward directions. Finally, a fully connected dense layer with a softmax activation function performs the classification task. The major contributions of our work are:

- We propose a novel deep learning model InceptionResNet-BiLSTM that exploits both the visual and temporal artifacts for the accurate detection of deepfake videos.
- Our deepfakes detector model is capable of detecting all types of deepfakes i.e., identity swap, lip-synching, and puppet mastery.
- We present an effective and efficient deepfakes detection model that is robust against different illumination conditions, races of people, and videos generated via several deepfakes techniques such as DeepFakes, FaceSwap, Face2Face, FaceShifter, NeuralTextures, DeepFaceLab, and FSGAN.
- We employ different augmentation techniques to address the class imbalance problem of the FakeAVCeleb dataset.
- We performed extensive experiments on diverse datasets including the close-set and cross-set evaluations to demonstrate the generalizability of our proposed model.

II. Related Work

Existing detection approaches can be divided into two groups: (i) those that explore temporal artifacts among the video frames [6,19,24,27], and (ii) approaches that exploit visual inconsistencies in the video frames. Approaches that identify the temporal inconsistencies are mainly based on recurrent neural networks (RNNs) to classify the videos as fake or real [5]. For instance, to detect the temporal artifacts among the video's frames, a deep learning architecture [27] combining ResNext and LSTM was evaluated on a custom dataset of 6000 videos. These videos were gathered from FF++, Celeb-DF, and deepfakes detection challenge (DFDC) datasets. Non-face frames were discarded, and the model achieved an accuracy of 95.5%. Haliassos et al. [6] created a LipForensics model that was trained and tested on grayscale cropped images of the mouth region only. This model fails in scenarios where the mouth is not manipulated or has limited movement, as only the mouth area is considered for deepfakes detection. In [19], a long-term recurrent convolution network (LRCN) is presented that exposed the deepfakes via detecting irregular or slow eye blinking in a video. This model [19] fails to detect NeuralTextures-generated videos and is also unable to detect fake videos in case of closed eyes visuals and/or rapid eye blinking rate.

Existing methods for detecting the visual artifacts can be categorized as (i) deep feature learning-based approaches [7,14,25], and (ii) hand-crafted feature-based approaches [9,10,11,12]. For example, Ciftci et al. [10] developed a hand-crafted feature-based approach for detecting manipulated videos by exploiting medical signal features such as heart rate, extracted from the face region in a video. Spatial and temporal facial features are computed and passed to the convolutional neural network (CNN) and SVM for the discrimination of fake and pristine videos. This approach [10] is computationally complex and has a huge feature vector space. Moreover, the detection performance decreases while reducing the dimensionality of the vector. Nguyen et al. [15] presented Capsule Forensics for digital media forensics problems including the detection of fully computer-generated images, computer-manipulated images, and presentation attacks. Capsule Forensics demonstrated good performance against other competitive approaches and introduced the capability of capsule networks for the development of generic deepfakes detection systems [5]. Amerini et al. [16] extracted the optical flow fields between the consecutive frames through PWC-Net and then used them to train the CNN models (ResNet50 and VGG16) for detecting deepfake videos. Only preliminary results are reported in [16] for the Face2Face subset of the FF++ dataset.

III. Proposed Method

This section provides the details of the proposed deepfakes detection framework (Fig. 1).

A. Pre-Processing

In the pre-processing step, frames are extracted from the videos using the OpenCV library [8] followed by face detection. For face detection, we employed the Multi-Task Cascaded Convolution Neural Network (MTCNN) [20]. We focus only on the faces in the video frames because these are the areas where the actual manipulation takes place. The

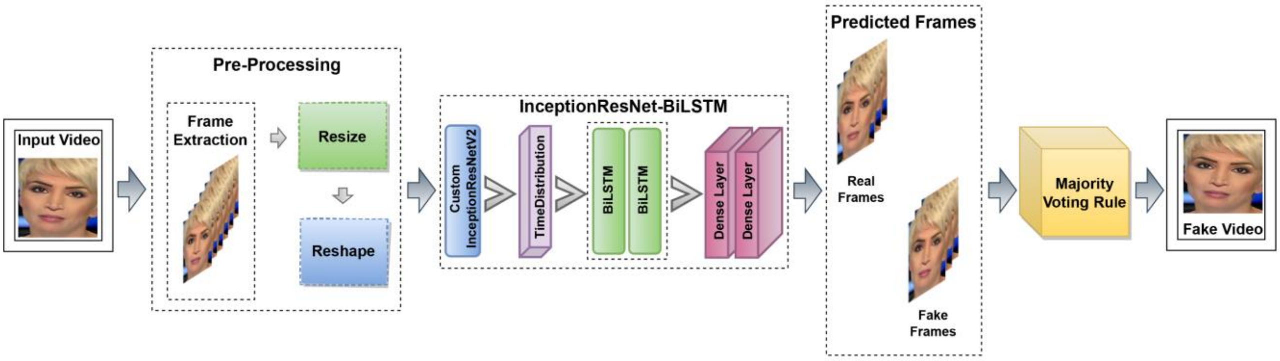

Fig. 1. Architecture of deepfakes examiner.

extracted faces are then resized to 300 × 300 with three channels. These resized facial images are fed to our customized InceptionResNet-BiLSTM model to compute effective deep features and later classify the video as real or fake.

B. *Architecture Details*

The proposed model is a hybrid deep learning framework comprising InceptionResNetV2 with Bidirectional LSTM. InceptionResNetV2 is a 164-layer deep network trained on the ImageNet dataset. It is derived from the combination of inception architecture with residual networks. Residual links are mixed with different-sized convolution filters in each Inception-Resnet block [17]. The use of residual links not only decreases the training time but also helps to solve the degradation problem. Bidirectional LSTM captures the temporal inconsistencies and remembers the long-term class dependencies. It also propagates the dependencies in both forward and backward directions and thus helps to improve detection accuracy.

In our proposed model, we employed the customized InceptionResNetV2 for exploiting the visual artifacts that exist in the video frames. InceptionResNetV2 consists of the following blocks: Stem, Inception-ResNet1-A, Reduction-A, Inception-ResNet1-B, Reduction-B, Inception-ResNet1-C, Average Pooling, Dropout, and Softmax [17]. We used InceptionResNetV2 up to the block Inception-ResNet1-C and then introduced two dense layers with 128 units to discover further hidden features. We also froze all the layers of custom InceptionResNetV2 excluding the last four, which helps in the reduction of training time as it backpropagates the gradient and updates the weights of only the last four layers. The freezing of layers also enables our model to become computationally efficient. The feature vector sequences are then flattened using a TimeDistributed layer and later passed to Bidirectional LSTM layers for exploiting the temporal patterns appearing among the frames. We used two Bidirectional LSTM layers one with 128 units and the other with 64. BiLSTM enhances the feature extraction potential of InceptionResNetV2 due to its ability to learn the patterns for a longer time period. After the Bidirectional LSTM layers, we employed a dense layer with a ReLU activation function. Finally, a fully connected dense layer with a softmax activation function is used to classify the frame as real or fake. The detailed architecture of our InceptionResNet-BiLSTM model is shown in Fig. 2.

C. *Majority Voting Rule for Video Classification*

Our proposed framework classifies the frames of the videos individually as either real or fake. To classify if the video is real or fake, we applied the majority voting rule, as follows:

$$V_t = max\{ R, F\} \qquad (1)$$

where R and F represent the counters for real and fake frames, and V_t indicates the category assigned to the video.

IV. Datasets

For the performance evaluation of our model, we selected a largescale, standard, and diverse FaceForensics++ [13] and FakeAVCeleb [22] datasets. These datasets comprise videos having different illumination conditions, and individuals belonging to different ethnic backgrounds. *It is important to mention that the testing set videos are unseen during the training of our proposed model for both datasets.*

The FF++ contains 1000 pristine/original videos collected from the youtube-8M dataset having frontal faces without any occlusions. From these pristine videos, manipulated videos are generated using deep learning and computer graphics-based approaches. Overall, the FF++ dataset has five subsets of manipulated videos named as DeepFakes, FaceSwap, Face2Face, FaceShifter, and NeuralTextures each containing 1000 videos. The dataset is available in three quality levels: (i) uncompressed, (ii) low compression, and (iii) high compression [13]. We evaluated our model on a low compression level. To evaluate our model, we split each subset of the FF++ dataset into training and testing sets (70:30).

FakeAVCeleb is an audio-visual deepfakes dataset containing real videos of celebrities from YouTube. Real videos are gathered from the voxCeleb2 dataset whereas fake face-swapped videos are generated using different deepfakes tools such as DeepFaceLab [2], FaceSwap [1], and FSGAN [23]. This dataset [22] contains four subsets (RealAudio RealVideo, FakeAudio FakeVideo, RealAudio FakeVideo, and FakeAudio RealVideo) having a total of 500 real and 20,000 plus fake videos with different ethnicity. This dataset includes videos with faces captured at different angles. Each video in a dataset has an average duration of 7 seconds containing a single individual. We split the dataset into an 80:20 ratio for evaluation.

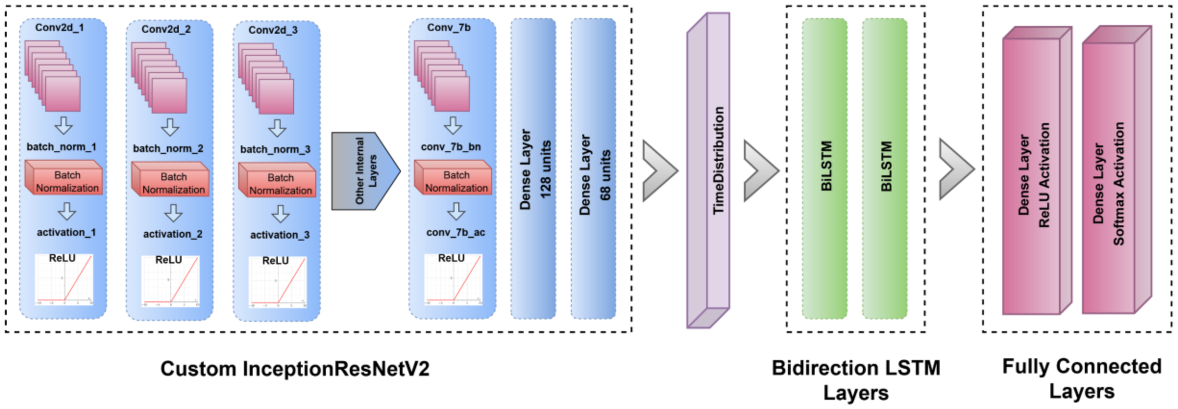

Fig. 2. InceptionResNet-BiLSTM model.

V. Experiments And Results

Performance of our model is evaluated using the accuracy, area under curve (AUC) score, true positive rate (TRP), and true negative rate (TNR). Further, implementation details and experiments are discussed in the subsequent subsections.

A. *Implementation Details*

The proposed model is implemented using a Python module Keras TensorFlow. We trained the proposed model on the extracted frames using the Adam optimizer and binary cross-entropy loss. We also employed an early stopping regularization technique on validation accuracy to avoid the model from overfitting. Our proposed model stops the training when there is no improvement in the validation accuracy for four consecutive epochs. Other training parameters are as follows: batch size = 10, learning rate = le-5, beta_1 = 0.9, beta_2 = 0.999, epsilon = none. After detailed experimentation, we tuned our model to these settings as we attained the best results on these parameter values. The model is executed on a high-performance computing machine having 192 GB RAM, 48 CPU Cores, and 4 NVIDIA Tesla V100 16G GPUs.

B. *Performance Evaluation*

To evaluate the performance of our model, we conducted two experiments. In the first experiment, we evaluated the performance of each subset of the FF++ dataset. For this purpose, we designed a multi-stage experiment where the real class contains the pristine videos, and the fake class contains the videos from one of the subsets of the FF++ dataset. In the first stage, we used the 1000 real videos and 1000 DeepFakes subset fake videos from the FF++ dataset for evaluation. Similarly, in the next four stages, we used the 1000 real videos against the 1000 fake videos of the remaining classes separately and computed the accuracy for each subset. For each stage of this experiment, we split our dataset into training and testing sets (70-30). The training set is used to train the model while the testing set is utilized for the evaluation of our model. From the training set, 20 % of the data is used for validation purposes. We stored the best model weights and used them for testing. The results in terms of accuracy, TPR, and TNR are shown in Table I. TPR indicates the probability that the model predicts the fake videos as fake, whereas TNR is the probability of the model predicting real videos as real.

TABLE I. Results on FF++ dataset.

Subsets of FF++ Dataset	Accuracy (%)	TPR (%)	TNR (%)
DeepFakes	93.39	96.07	90.71
FaceSwap	93.01	92.81	93.21
Face2Face	92.11	92.09	92.14
FaceShifter	84.91	72.56	97.14
NeuralTextures	78.67	61.87	95.36

From Table I, it can be seen that the proposed model performs very well in detecting the fake videos manipulated using DeepFakes, FaceSwap, and Face2Face methods. For videos manipulated using the FaceShifter method, accuracy, and TPR is reasonably good but for the NeuralTextures forged videos, the accuracy drops to 78% while TPR drops to 61.8%. This is attributed to the fact that in NeuralTextures manipulated videos, there is an imperceptible semantic variation because only the mouth region is modified to change the expressions of a person, thus makes it a challenging task to detect fake videos manipulated using the NeuralTextures algorithm. From the TPR values in Table I, it can also be inferred that our model can detect the fake videos manipulated using the DeepFakes generation technique more accurately.

In the second experiment, we evaluated the performance of our model on the FakeAVCeleb dataset. We utilized the RealAudio RealVideo (RaRv) and FakeAudio FakeVideo (FaFv) subsets of the FakeAVCeleb dataset for evaluation. The total number of real videos is 500 while there are 10,000 fake videos. This dataset has a class imbalance problem as it contains more fake videos than real ones. Therefore, we applied various augmentation techniques (such as sharpening, blurring, translation, noise, etc.) to increase the number of real videos. Our proposed model attained an accuracy of 76.57%, AUC score of 77.2%, TPR of 67.61%, and TNR of 86.97% on the FakeAVCeleb dataset. Our method achieved slightly lower performance on the FakeAVCeleb dataset as compared to the FF++ dataset. This could be due to the fact that FakeAVCeleb is a more challenging dataset as compared to others because it contains videos having angled faces along with individuals belonging to five different ethnic backgrounds (i.e., American, African, South Asian, East Asian, European).

C. *Comparison with Existing Methods*

To evaluate the performance of our proposed method against existing deepfakes detection models, we designed two experiments. In the first experiment, we compared the detection results of our model with these methods [14, 16, 18, 21, 25, 26] on the FF++ dataset. The experimentation protocol is the same as mentioned for the multi-stage experiment in Section V-B. The comparison in terms of accuracy is shown in Table II. The missing values in Table II indicate that the result for the subset is not provided by the respective methods.

From Table II, it can be observed that the detection accuracy of our model is improved as compared to methods [14, 18, 21, 25, 26] for the FaceSwap subset, and achieved an average accuracy gain of 24%. For the DeepFakes subset, our proposed model slightly improves the detection accuracy when compared with [14, 18, 26]. Whereas, for the Face2Face subset, our model achieved better accuracy than [14, 16, 25, 26] but slightly lower than [21]. Our method outperforms this method [18] for the detection of FaceShifter-generated videos by achieving an accuracy gain of 38.91%. In the case of NeuralTextures subset of FF++ dataset, our method performs far better than [14, 18, 21, 26] except from [25]. This method [25] provides good detection results only for the NeuralTextures subset as [25] is a one-class anomaly detection method trained only on real images. Since the NeuralTextures images have very few semantic changes only at the mouth area, thus making such fake images closer to the real ones. In order to attain the desired results, the methods [18] followed the approach of removing non-face frames from the training and testing set. While our proposed framework achieved good classification results even in the presence of frames having non-face regions, thus demonstrating the effectiveness of our method. It can also be seen from Table II that existing methods like [14, 18, 21, 25, 26] reported the performance on three or four subsets of the FF++ dataset while [16] stated the results on only one subset. It is a difficult task to obtain good detection results on all subsets of the FF++ dataset, especially in the presence of challenging conditions such as non-face frames, varying illumination conditions, people belonging to different races and faces having accessories (i.e., glasses, etc.). As per Table II, our framework has an edge over other approaches [14, 16, 18, 25, 26] as it shows good detection results on all subsets and is capable of distinguishing between the real and fake videos generated via different manipulation techniques.

TABLE II. ACCURACY COMPARISON WITH EXISTING METHODS ON THE FF++ DATASET.

Models	**Subsets of FF++ Dataset**				
	DeepFakes	***FaceSwap***	***Face2Face***	***FaceShifter***	***NeuralTextures***
Khalifa et al.[21]	93.30%	91.96%	93.75%	--	77.10%
CviT [18]	93%	69%	--	46%	60%
Amerini et al. [16]	--	--	81.61%	--	--
Xie et al. [14]	93.08%	74.67%	91.61%	--	65%
OC-FakeDect [25]	86.20%	86.10%	71.20%	--	95.30%
Demir et al. [26]	93.28%	91.62%	59.69%	--	57.02%
Proposed Model	93.39%	93.01%	92.11%	84.91%	78.6%

In the second experiment, we compared the performance of our model on the FakeAVCeleb dataset with existing methods using the AUC score as adopted by the comparative methods. The experimental protocols are the same as mentioned for the second experiment of Section V-B. The comparison results in Table III show that the existing methods have lower AUC scores on the FakeAVCeleb dataset which indicates the complexity and challenging nature of this dataset. Among the existing methods, Mesoinception4 provides the highest AUC of 77.8% while our method attains 77.2% AUC which is comparatively equivalent to the Mesoinception4 method. This shows the effectiveness of the proposed model on the FakeAVCeleb dataset compared to the existing methods.

TABLE III. COMPARISON WITH EXISTING METHODS ON THE FAKEAVCELEB DATASET.

Models	**AUC (%)**
HeadPose [22]	49.2
Capsule [22]	73.1
VA-logReg [22]	65.4
Xception-comp [22]	73.4
Mesoinception4 [22]	77.8
Proposed Model	77.2

D. Generalizability Evaluation

To analyze the generalizability aspect of our model, we performed cross-set and close-set evaluations. In cross-set evaluation, we trained the model on four subsets of the FF++ dataset and tested it on the remaining unseen subset, while in close-set evaluation, we trained our model on all the subsets of the FF++ dataset and evaluated it on the testing set of each subset separately. The details of these experiments are given in the subsequent sections.

1) Close-Set Evaluation. In close-set evaluation, we examined the accuracy of our model on the FF++ dataset to show the model's generalizability. For this reason, we performed an experiment where the real class contains pristine videos, and the fake class contains the videos from all the subsets of the FF++ dataset. So, the real class has 1000 videos while the fake class consists of 5000 videos. We split our dataset into training and testing sets. For this experiment, we stored the best model weights. There are two variants of the testing set for this experiment. The first testing set variant has real and fake subsets where the fake subset contains the fake videos of all subsets in the FF++ dataset. On this testing set, our model achieved an overall accuracy of 82.64%. The second testing set variant is the same as used for the first multi-stage experiment in Section V-B. The results shown in Table IV revealed that for close-set evaluation, accuracy values are comparatively lower than in the multi-stage experiment. The class imbalance problem introduced in this experiment may be the reason for a decrease in accuracy.

TABLE IV. CLOSE-SET EVALUATION ON FF++ DATASET.

Subsets of FF++ Dataset	**Accuracy (%)**
DeepFakes	70.71
FaceSwap	68.99
Face2Face	72.4
FaceShifter	67.14
NeuralTextures	64.8

2) Cross-Set Evaluation. To analyze the generalization ability of our proposed model from identity swap to puppet mastery and vice versa, we designed an experiment to perform the cross-set evaluation on the subsets of the FF++ dataset. This experiment is carried out in five phases where for each phase, we trained the model on four subsets and evaluated it on the testing set of the remaining one unseen subset of the FF++ dataset. For instance, in the first phase, we trained the model on the combined training set of DeepFakes, FaceSwap, Face2Face, and NeuralTextures subsets and used the FaceShifter subset for evaluation. We utilized the best model weights for the evaluation of the model on the testing sets. The cross-set evaluation results in terms of accuracy are listed in Table V.

TABLE V. CROSS-SET EVALUATION ON FF++ DATASET.

Training set	**Testing set**	**Accuracy (%)**
DF-FS-FF-NT	FaceShifter	62.32
FSh-NT-DF-FS	Face2Face	62.5
FS-FSh-FF-NT	DeepFakes	65.89
NT-FSh-DF-FF	FaceSwap	53.92
FS-FSh-FF-DF	NeuralTextures	63.65

Despite achieving good detection results on the individual subsets of FF++ dataset, the performance of our model is not as well on the cross-set experiment. This is due to the fact that there exists huge dissimilarity among the subsets of FF++ dataset involved in the cross-set evaluation. The generative techniques used to create the videos for different subsets are completely distinct and diverse from each other. For example, FaceSwap and Face2Face are computer graphics-based techniques while DeepFakes, FaceShifter, and NeuralTextures are DL-based techniques for generating fake videos. Moreover, differences in the manipulation types also exist such as FaceSwap and DeepFakes subsets have identity-swapped videos while Face2Face, FaceShifter, and NeuralTextures contain videos having puppet mastery manipulation. The variations among the different subsets in terms of deepfakes type and generative techniques make the cross-set evaluation more challenging. Regardless of the above-mentioned diversity between subsets, it can be noted that our model attained the highest accuracy of 65% on the DeepFakes subset. The detection accuracy above 60% for all the subsets except FaceSwap is encouraging and demonstrates the generalization power of our model even in the presence of such huge diversity among the subsets of the FF++ dataset.

VI. Conclusion

In this paper, we have presented an effective and efficient InceptionResNet-BiLSTM model to detect all types of deepfakes such as identity swap, puppet mastery, and lip-synching. Our proposed model can identify both the temporal and visual artifacts among the frames of deepfake videos by employing the InceptionResNetV2 and Bidirectional LSTM. We evaluated our proposed model on all subsets of the FF++ dataset and the FakeAVCeleb dataset. The results indicate the remarkable detection capability of our model for videos generated through the techniques such as FaceSwap, Face2Face, DeepFakes, FaceShifter, NeuralTextures, DeepFaceLab, and FSGAN. We also performed extensive experimentation to analyze the generalizability aspect of our model and exhibit the cross-set and close-set evaluation on the FF++ dataset. In the future, we intend to further improve the robustness of our model against the close-set and cross-set evaluation. Moreover, we also plan to evaluate our model on all compression levels of FF++.

Acknowledgment

This work was supported by the grant of Punjab HEC of Pakistan via Award No. (PHEC/ARA/PIRCA/20527/21).

References

[1] FaceSwap, https://faceswap.dev/ , last accessed 2022/05/20.

[2] DeepFaceLab, https://github.com/iperov/DeepFaceLab/ , last accessed 2022/05/20.

[3] Reface, https://hey.reface.ai/ , last accessed 2022/05/20.

[4] M. Masood, M. Nawaz, K. M. Malik, A. Javed, A. Irtaza, and H. Malik. "Deepfakes Generation and Detection: State-of-the-art, open challenges, countermeasures, and way forward". Applied Intelligence, pp. 1-53, 2022.

[5] T. T. Nguyen, Q. V. H. Nguyen, D. T. Nguyen, D. T. Nguyen, T. Huynh-The, S. Nahavandi, T. T. Nguyen, Q. V. Pham and C. M. Nguyen. "Deep learning for deepfakes creation and detection: A survey". Computer Vision and Image Understanding, 223, 103525, 2022.

[6] A. Haliassos, K. Vougioukas, S. Petridis, and M. Pantic. "Lips don't lie: A generalisable and robust approach to face forgery detection". In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 5039-5049, 2021.

[7] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen. "Mesonet: a compact facial video forgery detection network". InIEEE international workshop on information forensics and security (WIFS), pp. 1-7, 2018.

[8] OpenCV, https://opencv.org/ , last accessed 2022/07/25.

[9] F. Matern, C. Riess, and M. Stamminger. "Exploiting visual artifacts to expose deepfakes and face manipulations". In IEEE Winter Applications of Computer Vision Workshops (WACVW), pp. 83-92, IEEE, 2019.

[10] U. A. Ciftci, I. Demir, and L. Yin. Fakecatcher: "Detection of synthetic portrait videos using biological signals". IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020.

[11] S. Agarwal, H. Farid, Y. Gu, M. He, K. Nagano, and H. Li. "Protecting World Leaders Against Deep Fakes". In CVPR workshops, vol. 1, p. 38, 2019.

[12] T. Jung, S. Kim, and K. Kim. "Deepvision: Deepfakes detection using human eye blinking pattern". IEEE Access, 8, pp. 83144-83154, 2020.

[13] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner. "Faceforensics++: Learning to detect manipulated facial images". In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1-11, 2019.

[14] D. Xie, P. Chatterjee, Z. Liu, K. Roy, and E. Kossi. "Deepfake detection on publicly available datasets using modified AlexNet". In IEEE symposium series on computational intelligence (SSCI), pp. 1866-1871, IEEE, 2020.

[15] H.H. Nguyen, J. Yamagishi, and I. Echizen. Capsule-forensics: "Using capsule networks to detect forged images and videos". In ICASSP IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2307-2311, 2019.

[16] I. Amerini, L. Galteri, R. Caldelli, and A. Del Bimbo. "Deepfake video detection through optical flow based CNN". In Proceedings of the IEEE/CVF international conference on computer vision workshops, pp. 0-0, 2019.

[17] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. "Inception-v4, inception-resnet and the impact of residual connections on learning". In: Thirty-first AAAI conference on artificial intelligence, 2017.

[18] D. Wodajo, and S. Atnafu. "Deepfake video detection using convolutional vision transformer". arXiv preprint arXiv:2102.11126, 2021.

[19] Y. Li, M. C. Chang, and S. Lyu. "In ictu oculi: Exposing ai created fake videos by detecting eye blinking". In IEEE International Workshop on Information Forensics and Security (WIFS), pp. 1-7, 2018.

[20] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. "Joint face detection and alignment using multitask cascaded convolutional networks". IEEE Signal Processing Letters, 23(10), pp.1499-1503, 2016.

[21] A.H. Khalifa, N. A. Zaher, A. S. Abdallah, and M. W. Fakhr. "Convolutional Neural Network Based on Diverse Gabor Filters for Deepfake Recognition". IEEE Access, 10, pp. 22678-22686, 2022

[22] H. Khalid, S. Tariq, M. Kim, and S. S.Woo. "FakeAVCeleb: a novel audio-video multimodal deepfake dataset". arXiv preprint arXiv:2108.05080, 2021.

[23] Y. Nirkin, Y. Keller, and T. Hassner. "Fsgan: Subject agnostic face swapping and reenactment". In: Proceedings of the IEEE/CVF international conference on computer vision, pp. 7184-7193, 2019.

[24] D. M. Montserrat, H. Hao, S. K. Yarlagadda, S. Baireddy, R. Shao, J. Horváth, E. Bartusiak, J. Yang, D. Guera, F. Zhu, and E.J. Delp. "Deepfakes detection with automatic face weighting". In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 668-669, 2020.

[25] H. Khalid, and S.S. Woo. "OC-FakeDect: Classifying deepfakes using one-class variational autoencoder". In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 656-657, 2020.

[26] I. Demir, and U. A. Ciftci. "Where do deep fakes look? synthetic face detection via gaze tracking". In ACM Symposium on Eye Tracking Research and Applications, pp. 1-11, 2021.

[27] Y. Al-Dhabi, and S. Zhang. "Deepfake Video Detection by Combining Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN)". In IEEE International Conference on Computer Science, Artificial Intelligence and Electronic Engineering (CSAIEE), pp. 236-241, 2021.