Enhancing Authenticity Verification with Transfer Learning and Ensemble Techniques in Facial Feature-Based Deepfake Detection

Nadeem Qazi
Computer and Digitial Technolgies
University of East London
London,UK
n.qazi@uel.ac.uk

Iftikhar Ahmed
Research and Development
Tietoevry Finland Oy
Finland
iftikhar.ahmad@tietoevry.com

***Abstract*—Deepfake technology, facilitated by deep learning algorithms, has emerged as a significant concern due to its potential to deceive humans with fabricated content indistinguishable from reality. The proliferation of deepfake videos presents a formidable challenge, propagating misinformation across various sectors such as social media, politics, and healthcare. Detecting and mitigating these threats is imperative for fortifying defenses and safeguarding information integrity.**

This paper tackles the complexities associated with deepfake detection, emphasizing the necessity for innovative approaches given the constraints of available data and the evolving nature of forgery techniques. Our proposed solution focuses on leveraging facial features and transfer learning to discern fake videos from genuine ones, aiming to identify subtle manipulations in visual content. We systematically break down videos into frames, employ the Haar cascade algorithm for facial recognition, and utilize transfer learning to extract discriminative features. We evaluate multiple pre-trained models, including VGG16, ConvNeXt-Tiny, EfficientNetB0, EfficientNetB7, DenseNet201, ResNet152V2, Xception, NASNetMobile, and MobileNetV2, for feature extraction. Subsequently, we feed these features into a Deep Artificial Neural Network (DANN) for deepfake detection and employ ensemble learning to combine the strengths of the best-performing models for enhanced accuracy.

We found that the ensemble model comprising ConvNextTiny, EfficientNetB0, and EfficientNetB7 showed enhanced accuracy in detecting deep fakes compared to alternative models achieving up to 98% accuracy through ensemble learning.

***Index Terms*—Deepfake detection, video classification, Transfer learning, EfficentNetB0, DenseNet, Ensemble learning**

I. Introduction

Deepfakes are fake images and videos that are generated by deep learning algorithms. Deep Fake technology currently exists in three categories including Lip syncing, FaceSwap, and puppet-master. Lip syncing enables the manipulation of facial expressions and synchronized speech, achieving a remarkable level of realism. FaceSwap deep fake video technique replaces the original face with a target person's face to make a video, where the target person appears to do the activities performed by the source person. The original facial expression and actions remain unchanged, only the face is replaced, so that it appears to be authentic, though it is not entirely genuine.

Modern Deepfake technology employs deep neural networks in encoder-decoder or Generative Adversarial Networks (GANs) architecture on face images to automatically map the facial expressions of the source to the target. Generative Adversarial Networks (GANs), the recent trend for deep fake creation, consist of two neural networks: the generator and the discriminator. The generator attempts to create realistic content, such as images or videos, while the discriminator evaluates and distinguishes between authentic and generated content. This adversarial process continues iteratively until the generator produces convincingly realistic content. New Deep fake generation software utilizes these advanced algorithms to make fake content that looks just like the real thing.

Deepfake videos, particularly in the dynamic field of cybersecurity, are a considerable threat, for spreading misinformation across various domains such as social media, politics, and healthcare. These videos increase risks such as identity theft and phishing attacks, undermining digital trust. It can lead to misunderstanding, influencing decision-making, and posing significant threats to democracy, national security, and society. Humans are usually incapable of distinguishing these deepfakes from authentic content. It is, therefore, crucial to combat the spread of AI-powered misinformation and enhance the online environment by distinguishing real news, images, and videos from synthesized ones. This paper addresses the challenges in identifying deepfakes, highlighting the need for innovative solutions considering the data limitations and the ongoing evolution of deceptive techniques.

A. Challenges in Deep Fake Video Detection

The use of AI techniques in creating deep fake has made it difficult to spot fake videos and images, making it tricky to tell if a video or picture is genuine or fake. To detect these fake contents, we need intelligent methods that utilize a deep understanding of deepfake technology along with artificial intelligence. Existing AI techniques for detecting deep fake videos pose this as a binary classification challenge. These techniques utilize both hand-crafted features-based approaches and deep learning-based methods for identifying the deepfake.

Hand-crafted features focus on the traces left by computer programs that are used to create deepfakes, while deep learning methods automatically find unique features using convolutional networks to tell if a video is genuine or not. The deep learning solution, however, requires a huge amount of data for training AI algorithms to achieve a good performance level. Another challenge for deep fake detection is to achieve high model generalization, that measures the performance of algorithms on unknown datasets.

Related experiments [1] have proved that the generalization performance of existing deepfake detection algorithms is still insufficient for cross-dataset detection tasks. Furthermore, deep fake detection algorithms presented in the literature often employ frame-based binary classification. However, this technique has its drawbacks. On one hand, it is computationally intensive, making it unsuitable for real-time deep fake detection. On the other hand, the exceptionally realistic visuals in deepfake content present a considerable challenge, making it difficult to distinguish manipulated videos from authentic ones. Lastly, all the state-of-the-art deep fake detection models are based on black-box, models that lack interpretability and transparency.

This paper addresses these challenges associated with real-time deepfake video detection by proposing a transfer learning-based solution utilizing a convolution neural network on the facial features extraction and developing a deep learning neural network model for deepfake identification. The rest of the paper is organized in the following sections. The next section describes the related work, followed by the adopted methodology for deep fake video in Section 3. Section 4 presents the results and analysis of the experiments and finally, a conclusion is drawn in the last section of the paper.

II. RELATED WORK

Deep fake video detection research is in its early stages and is normally considered a binary classification problem, aiming to differentiate authentic and fake videos through deep learning algorithms. Researchers have employed multiple techniques that include eye blinking [2], visual and face warping artifacts [3] [4], Head pose, and temporal inconsistencies between adjacent frames of video, fed in machine learning algorithms for detecting the video forgery. Among the methods that have been suggested for deepfake detection, convolution neural networks (CNN) have been a popular choice. CNNs have shown great ability and scalability for applications regarding image and video processes when compared with other methods for supervised learning in computer vision. Various approaches have demonstrated the use of CNN along with other learning models like Recurrent Neural Networks (RNN) [5], Long Short-Term Memory Networks (LSTM) [6], and Capsule Networks [7] and have shown good results for deep fake detection. Sabir et al. [8] developed a dense recurrent convolution (RCN) model over the Face Forensics++ dataset, utilizing spatiotemporal features of video frames to detect deep fakes. Likewise, Guera and Delp [9] utilized time-based irregularities to propose a temporal-aware pipeline for fake video detection. Their proposed method consists of a fully connected network of CNN and long short-term memory (LSTM). They employed Frame-level features extracted through CNN and fed those into LSTM to form a temporal sequence descriptor for classifying doctored videos from real ones. Montserrat et al. [10] proposed an automatic weighting-based CNN-RNN framework. Wu et al. [11] proposed a novel manipulation detection framework, named SSTNet, exploiting both low-level artifacts and temporal discrepancies. The use of a biological signal [12], eye blinking [3], and heartbeat signals [13] are also demonstrated by researchers to detect deep fake content. Researchers [4] exploited artifacts observed during the face-warping phase of the deepfake generation, to detect the deepfake video. Their proposed method is evaluated on two deepfake datasets, namely the UADFV and DeepfakeTIMIT. Xinyi Ding et al. [14] employed transfer learning to identify face-swapped images. Their study was based on pre-trained the ImageNet model of ResNet-18 for object recognition, which was then fine-tuned on a public dataset for deepfake detection purposes. However, the stability of their presented approach was compromised due to the overfitting problem.

III. METHODLOGY

Deep fake video detection can be viewed as a natural extension of image classification tasks. However, it poses greater challenges than image classification because it incorporates an additional temporal dimension. This dimension arises from the contextual relationship between the current and previous frame. Furthermore, like other machine learning classification tasks, effective deep fake video detection demands robust generalization techniques and a considerable amount of data for comprehensive training, which may not always be feasible. To tackle these challenges comprehensively, we developed a hybrid model that integrates transfer learning to enhance generalization and address the issue of data scarcity, along with ensemble learning to further improve the accuracy of deepfake video detection. The pipeline of our proposed hybrid model as shown in Fig. 1, includes several stages: video pre-processing, transfer learning for feature extraction, video classification through deep ANN models, and finally, ensemble learning for improving accuracy. We describe each of these stages in the following subsections.

A. Dataset

We utilized the publicly available FaceForsenic++ dataset [15] for our experiments with the proposed deepfake detection pipeline. FaceForensics++ is a forensic dataset comprising 1000 authentic video sequences. Four automated face manipulation techniques including Deepfakes, Face2Face, FaceSwap, and NeuralTextures were applied to these authentic videos generating 1000 manipulated videos for each category, providing a diverse set of manipulated content for research and development in the domain of facial image forensics.

B. Video Pre-processing

Our proposed methodology presented in Fig. 1, involves the classification of multiple frames in a video. The predic-

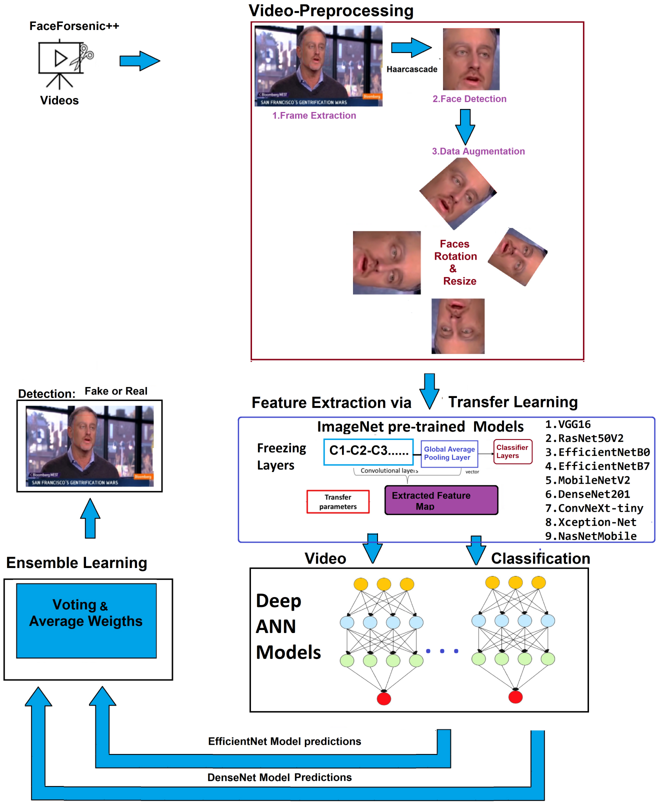

Fig. 1. Proposed Pipeline for Deepfake Detection.

tions from each frame are then combined to determine the authenticity of the entire video, distinguishing between real and fake content. During the video processing stage, frames are extracted from the video clips. For the FaceForsenic++ data set, a total of 17416 frames were extracted from each of the fake and real videos. However, this dataset is slightly imbalanced with 8,000 frames in the fake category and 9,416 frames in the real video category. Face extraction from each of these frames was then accomplished by utilizing the Haar Cascade algorithm [16].

To augment the dataset and improve the model's robustness and generalization, we implemented data augmentation techniques using the TensorFlow Keras library. This involved the random rotation of faces between 0 and 45 degrees, as well as horizontal and vertical flipping. Additionally, to introduce the variety in the images, we introduced a 20% zooming factor and a 10% shift in the width and height dimensions to make the images either small or bigger. Following the same procedure, lighting conditions were also randomly varied by adjusting brightness and contrast within the value range of 0.7 to 1.2. Subsequently, features were extracted from these manipulated faces through transfer learning, a process elaborated upon in the next section.

C. Feature Extraction through Transfer Learning

Transfer learning is a machine learning technique that involves leveraging the feature representation learned by a pre-trained model, often trained on extensive datasets like ImageNet. ImageNet constitutes a vast dataset containing over 14 million annotated images categorized into more than 21,000 groups or classes. The pre-trained Imagenet model trained on one task is reused with a few or more weights adjusted, along with the addition or removal of some new layers, for a second related task. This process significantly reduces the model development time and improves the performance compared to an isolated learning model.

TABLE I
IMAGENET PRETRAINED MODELS

Model	ModelSize	Parameters	Top1-Accuracy
VGG16	528	138.4	71.30
NASNetLarge	343	88.9	82.5
EfficientNetB7	256	66.7	84.30
ResNet152V2	232	60.4	78.00
InceptionResNetV2	215	55.9	80.30
ConvNeXtTiny	109.42	28.6	81.30
ResNet50	98	25.6	74.90
Xception	88	22.9	79.00
DenseNet201	80	20.2	77.30
NASNetMobile	23	5.3	74.40
EfficientNetB0	29	5.3	77.10
MobileNetV2	14	3.5	71.30

Researchers have identified two types of transfer learning: inductive and transductive transfer learning. Inductive transfer learning focuses on applying knowledge across different tasks, where the target task is distinct from the source task. On the other hand, transductive transfer learning involves scenarios where the tasks remain the same, but the datasets employed for these tasks are different. Both these approaches utilize the pre-trained Imagenet models, which are convolution neural network models consisting of several convoluted layers and one fully connected layer. Several such pre-trained CNN models have been proposed such as AlexNet [17], GoogleNet [18], and VGG [19].

Table 1 shows the characteristics of Imagenet models. In choosing a pre-trained model for image classification, we focused on performance, considering top-1 accuracy for precise predictions and the number of parameters for computational efficiency. Additionally, we factored in the deployment platforms (web or mobile) to ensure compatibility. As can be seen from the Table 1, the architecture of the EfficientNetB0 model is well suited for real-time application due to its balance between model accuracy and computational efficiency represented through model parameters and size. On the other hand, the CNN architecture of DenseNet201 is parameter-efficient due to its dense connections and consequently facilitates effective feature extraction. DenseNet is thus a good choice for learning complex patterns and capturing facial cues in manipulated facial images of the deepfake content.

Following this strategy, we chose eight pre-trained models including Xception, VGG16, RasNet50V2, EfficientNetB0, EfficientNetB7, MobileNetV2, DenseNet201, and ConvNeXt-Tiny, in a transductive transfer learning approach and tested their performance as feature extractors on FaceForsenic++

dataset. The final or fully connected layers of both of these pre-trained models were removed and all the remaining layers were set to a frozen state. We employed max pooling due to its ability to capture the most discriminative features for the given input face images. Feature vectors were thus constructed by fine-tuning these pre-trained models over the training and testing data set, achieved by dividing the pre-processed data with a ratio of 0.3. The generated feature vectors extracted from each of these models were then separately fed into a deep neural network model for binary classification, described in the next section.

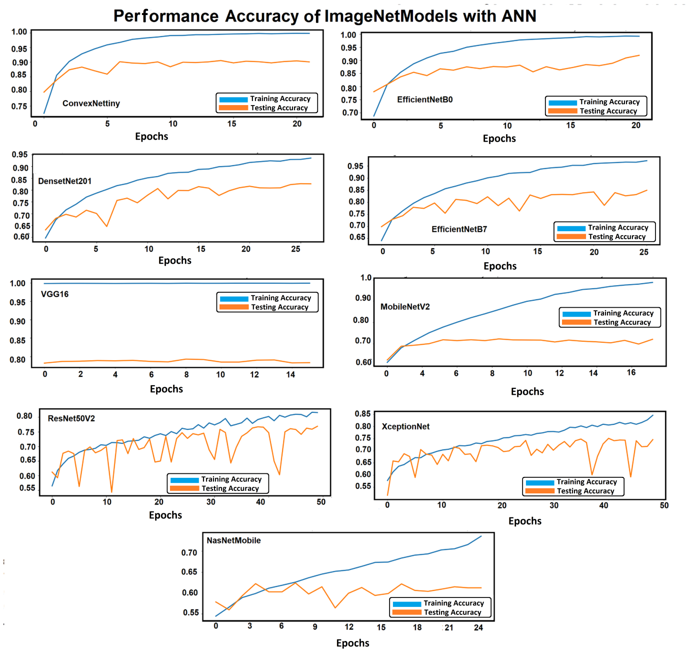

Fig. 2. Performance of Chosen ImageNet Models as Feature Extractor

D. Deep ANN Model

The Deep artificial neural network ANN classifier was constructed with two hidden layers having 50 hidden neurons and one classification layer with a sigmoid function. To avoid overfitting and minimize the loss function, we employed adaptive Moment Estimation or Adam optimizer for the training. The choice of the Adam optimizer was made because of its fast convergence and adaptive learning rate. An early stopping technique was used to avoid overfitting. The accuracy of these deep ANN models was evaluated on confusion matrix, F1-score, precision, and recall rate. The training accuracy of these models along with the confusion matrix is shown in Fig. 2 and Fig. 3 respectively.

E. Ensemble Learning

The final phase of our proposed methodology focuses on improving the accuracy of the deep fake detection classifiers. We employed ensemble learning to improve the predictive performance by leveraging the strengths of the individual models. Researchers have shown more than one approach to implementing ensemble learning, which includes Bagging, Boosting, Stacking, Voting, Blending, Adaptive boosting, etc. However, in this work, we utilized the model aggregation approach and implemented soft voting and weighted techniques to effectively combine the predictions of individual classifiers.

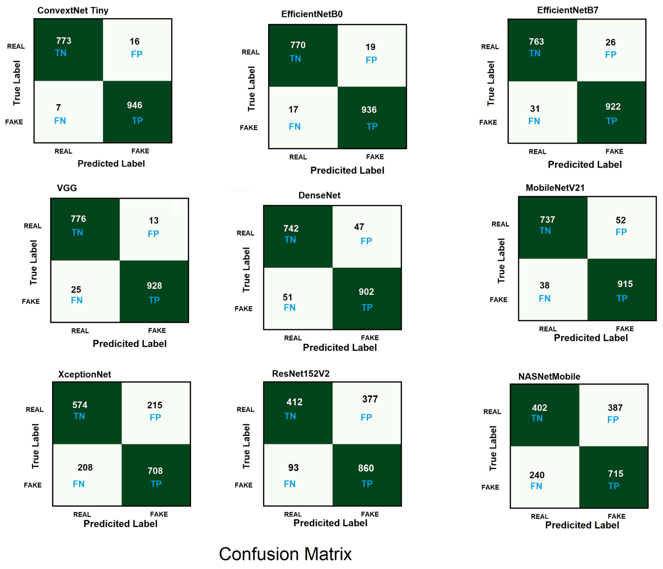

Fig. 3. Confusion Matrix of Chosen ImageNet Models as Feature Extractor

In the soft voting approach, we combined predictions of individual classifiers leveraging the probability estimates provided by each classifier for every class. The ensemble aggregates these probabilities through averaging. The final prediction is determined by selecting the class with the highest aggregated probability.

In another approach of weighted soft computing, we assigned weights to each of the classifiers based on their reliability. The weight was chosen by looking at the testing accuracy of the model, the model that attained the highest testing accuracy was weighted most, and consequently, the model with the lower testing accuracy was given the lower weight. The final prediction was made by combining the contributions from each classifier based on their weights.

The final ensemble predictions from soft voting and weighted approaches were evaluated on an unseen test dataset. The next section describes the performance accuracy of all the trained models both before and after the ensembling.

IV. RESULT & DISCUSSION

In evaluating the performance of our models, we employed training and testing accuracy as the key metrics, along with additional measures such as F1-score, precision, recall, and Area under Curve(AUC). The precision, recall, and F1-score were calculated using equations 1,2, and 3, where TP, FP, and FN represent True positive, False positive, and False negative respectively. The AUC as a performance metric was chosen because it comprehensively evaluates a model's capacity, particularly for an imbalanced dataset to distinguish between positive and negative classes. Its value ranges from 0 to 1 with an AUC value of 1 indicating the perfect classification and a value less than 0.5 is considered random performance, and the model is considered a bad classifier. The training and testing curves of these models as obtained during the training of the models are shown in Fig. 2(a) and Fig. 2(b) respectively, while other performance measures are shown in Fig. 4.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3}$$

It can be seen from the Fig. 2(a) that no single model emerged as the unequivocal winner across all performance criteria. Significantly, ConvNeXtTiny and VGG16 exhibited outstanding performance on the training dataset, achieving an impressive accuracy of 99%. Following closely, EfficientnetB0 and MobileNetV2 attained a training accuracy of 97% and 96% respectively.EfficientNetB7 and DenseNet201 yielded accuracies of 93%. The models Xception and RasNet50V2, however, exhibited relatively lower training accuracy of 84% and 80% respectively and NASNetMobile turned out to be the least training efficient with training accuracy of 74%.

The testing accuracy, which assesses the performance of these models on unseen data, is illustrated in Fig. 2(a). Intriguingly, ConvexNextTiny and EfficientNetB0 emerged as victors, achieving testing accuracies of 91% and 90%, respectively, followed by EfficientNetB7 and DenseNet201 at 81%, and VGG16 at 80%. On the other hand, ResNet152V2, MobileNetV2, and Xception demonstrated approximately 70% testing accuracy, while NASNetMobile exhibited the lowest accuracy at 64%.

The confusion matrix illustrated in Fig. 2(b) presents a comparative analysis of the performance of multiple models concerning their ability to correctly identify true negatives (TN) and true positives (TP) in distinguishing between real and fake videos. As can be seen from this figure ConvNeXtTiny demonstrated the highest TN score of 773, followed by EfficientNetB0 and EfficientNetB7, indicating their superior accuracy in identifying real videos. This trend was similarly observed for TP, with ConvexNeXtTiny leading with a value of 946, followed by EfficientNetB0 and EfficientNetB7 with scores of 928 and 915, respectively.

These results suggest that all three models exhibit a strong capability in detecting fake videos from real ones, with ConvexNextTiny emerging as the top performer, followed by EfficientNetB0 and EfficientNetB7. VGG16 and DenseNet201 models also showed competitive performance, albeit slightly lower than the top three. On the other hand, NasNetMobile demonstrated the least efficiency in this task.

Figure Fig. 4 shows the performance matrix of these models in terms of precision, recall, AUC, and F1-score. ConvNeXtTiny also exhibited the highest recall rate of 0.95 succeeded by EfficientNetB0, EfficientNetB7, DenseNet201, VGG16, and MobileNetV21, with values of 0.942, 0.937, 0.916, 0.934, and 0.955, respectively. Regarding precision, VGG16, ConvextNeXtTiny, and EfficientNetB0 emerged as top performers, each achieving a precision value of 0.94, closely pursued by EfficientNetB7, MobileNet, and DenseNet201, with a precision rate of 0.93,0.92,91 respectively.

Concerning the AUC metric, the ConvexNeXtTiny model once more outperformed others, achieving a value of 0.95, closely followed by EfficientNet B0, EfficientNet B7, VGG16, and MobileNetV2. In contrast, DenseNet exhibited an AUC value of 0.90. As expected, XceptionNet, NasNetMobile, and ResNet152V2 showed values nearing 0.7.

In conclusion, these results emphasize the significance of choosing suitable models for video authentication purposes, particularly highlighting the promising performance of smaller models such as ConvexNextTiny and various versions of EfficientNet in discerning real from fake videos.

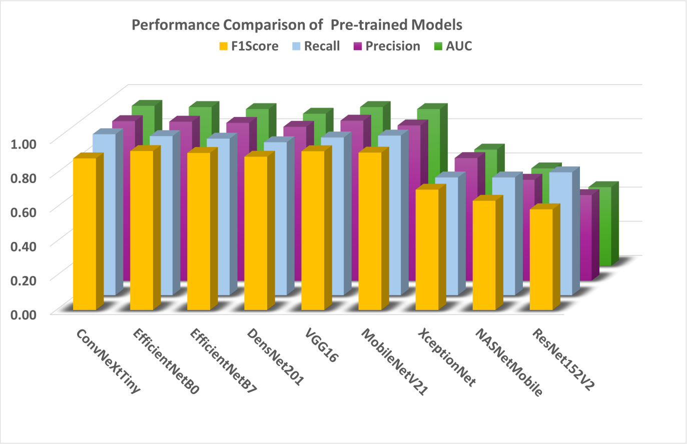

Fig. 4. Performance Comparision of the Trained Models

After analyzing performance and striving to enhance testing accuracy, we adopted an ensemble strategy. This involved combining ConvNextTiny with EfficientNetB0 and EfficientNetB7, pairing VGG16 with DenseNet201, and aligning MobileNetV21 and Xception with NASNetMobile and ResNet152V2, respectively, using a Voting technique outlined in the final section. The outcome was an improved testing accuracy, as depicted in Table 2. It can be seen from Table 2 that both of the adopted ensemble learning produced a positive effect in increasing the testing accuracy along with the improvement in precision, recall, and AUC.

It can be seen from Table 2.0 The ensemble model comprising ConvNextTiny, EfficientNetB0, and EfficientNetB7 has enhanced accuracy to over 98% through soft voting and over 90% through average weight technique, proving to be notably more effective by consistently improving across all metric measurements.

We also compared the performance of the proposed approach with [20]. [20] employed the EfficientNet model on FaceForsenic++ and achieved an accuracy of 85.84% and ACU of 72.17 for their proposed EfficientNet to detect the deep fake. However, the performance of EfficientNet models both B0 and B7 surpassed that of [20] achieving over 90% accuracy.

TABLE II
PERFORMANCE COMPARISION OF ENSEMBLING TECHNIQUE

Model	Accuracy	F1score	Recall	Precision	AUC
ConvNexttTiny	.89	0.91	0.95	0.94	0.95
EfficientNetB0	.87	0.94	0.94	0.94	0.94
EfficientNetB7	.81	0.93	0.93	0.93	0.93
Soft Voting	.98	0.96	0.96	0.95	0.95
Weighted.Voting	.94	0.97	0.97	0.96	0.96
VGG16	.81	0.94	0.93	0.95	0.95
DenseNet201	.80	0.91	0.91	0.91	0.90
MobileNetV21	.72	0.93	0.95	0.92	0.93
Soft Voting	.83	0.94	0.93	0.96	0.92
Weighted Voting	.85	.95	0.94	0.96	0.93
XceptionNet	.72	0.72	0.70	0.73	0.69
NASNetMobile	.64	0.65	0.70	0.60	0.58
ResNet152V2	.70	0.60	0.73	0.51	0.47
Soft Voting	.72	0.72	0.73	0.70	0.65
Weighted Voting	.72	0.73	0.72	0.71	0.66

V. CONCLUSION

In conclusion, our study was dedicated to addressing the challenge of detecting deep fake videos. We introduced a solution involving transfer learning to overcome the limitations posed by insufficient training data and to facilitate feature extraction. The evaluation of our method, employing various performance metrics such as accuracy, precision, recall, AUC, and F1-score, yielded promising results. The individual models, ConvexNextTiny, EfficientNetB0 EfficientNetB7, and DenseNet201, demonstrated robust performance across multiple metrics, showcasing high precision, recall, and F1-score values. Moreover, our ensemble methods, including the voting and weighting approach, surpassed the individual models, achieving even greater precision, recall, and F1-score. Notably, consistent, and high AUC values across all models and ensembles underscored their excellent discriminatory ability. We found the ensemble model consisting of ConvNextTiny, EfficientNetB0, and EfficientNetB7 improved the deep fake detection accuracy as compared to other models.

Our findings highlight the effectiveness of our approach, emphasizing the potential of combining transfer learning and ensemble methods for robust deep fake video detection with a specific emphasis on facial features. We specifically demonstrated that the resource-efficient nature of ConvexNextTiny, EfficientNetB0 EfficientNetB7, and DenseNet201, make it suitable for dynamic video detection. In the context of the critical issue of deep fake video detection, our study contributes to the establishment of a trustworthy digital environment by showcasing the efficacy of transfer learning in this domain.

ACKNOWLEDGMENT

This research is funded by TAILOR Connectivity Fund through Technische Universiteit Eindhoven, implemented as Financial Support to Third Parties (FSTP), under the rules of the European Commission Horizon 2020 program.

REFERENCES

[1] P. Korshunov and S. Marcel, "Improving generalization of deepfake detection with data farming and few-shot learning," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 4, no. 3, pp. 386–397, 2022.

[2] Y. Li, M.-C. Chang, and S. Lyu, "In ictu oculi: Exposing ai generated fake face videos by detecting eye blinking," *ArXiv*, vol. abs/1806.02877, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:47013913

[3] T. Jung, S. Kim, and K. Kim, "Deepvision: Deepfakes detection using human eye blinking pattern," *IEEE Access*, vol. 8, pp. 83 144–83 154, 2020.

[4] Y. Li and S. Lyu, "Exposing deepfake videos by detecting face warping artifacts," *arXiv preprint arXiv:1811.00656*, 2018.

[5] V. Abdul Jamsheed and B. Janet, "Deep fake video detection using recurrent neural networks," *International Journal of Scientific Research in Computer Science and Engineering*, vol. 9, no. 2, pp. 22–26, 2021.

[6] L. Rahunathan, D. Sivabalaselvamani, A. PriyaDharshini, M. Vignesh, and G. VinithKumar, "Cannotation measureup to detect deepfake by face recognition via long short-term memory networks algorithm," in *Intelligent Communication Technologies and Virtual Mobile Networks*. Springer, 2023, pp. 475–487.

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

[8] E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, and P. Natarajan, "Recurrent convolutional strategies for face manipulation detection in videos," *Interfaces (GUI)*, vol. 3, no. 1, pp. 80–87, 2019.

[9] D. Güera and E. J. Delp, "Deepfake video detection using recurrent neural networks," in *2018 15th IEEE international conference on advanced video and signal based surveillance (AVSS)*. IEEE, 2018, pp. 1–6.

[10] D. M. Montserrat, H. Hao, S. K. Yarlagadda, S. Baireddy, R. Shao, J. Horváth, E. Bartusiak, J. Yang, D. Guera, F. Zhu *et al.*, "Deepfakes detection with automatic face weighting," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 668–669.

[11] X. Wu, Z. Xie, Y. Gao, and Y. Xiao, "Sstnet: Detecting manipulated faces through spatial, steganalysis and temporal features," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 2952–2956.

[12] 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.

[13] U. A. Çiftçi, I. Demir, and L. Yin, "Deepfake source detection in a heart beat," *The Visual Computer*, pp. 1–18, 2023.

[14] X. Ding, Z. Raziei, E. C. Larson, E. V. Olinick, P. Krueger, and M. Hahsler, "Swapped face detection using deep learning and subjective assessment," *EURASIP Journal on Information Security*, vol. 2020, no. 1, pp. 1–12, 2020.

[15] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics: A large-scale video dataset for forgery detection in human faces," *arXiv preprint arXiv:1803.09179*, 2018.

[16] H. Cevikalp, B. Triggs, and V. Franc, "Face and landmark detection by using cascade of classifiers," in *2013 10th IEEE international conference and workshops on automatic face and gesture recognition (FG)*. IEEE, 2013, pp. 1–7.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[18] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[20] S. Suratkar and F. Kazi, "Deep fake video detection using transfer learning approach," *Arabian Journal for Science and Engineering*, vol. 48, no. 8, pp. 9727–9737, Aug 2023. [Online]. Available: https://doi.org/10.1007/s13369-022-07321-3