

# DeePhyNet: Towards Detecting Phylogeny in Deepfakes

Kartik Thakral, Harsh Agarwal\*, Kartik Narayan\*, Surbhi Mittal, Mayank Vatsa *Fellow, IEEE*,  
Richa Singh *Fellow, IEEE*

**Abstract**—Deepfakes have rapidly evolved from their inception as a niche technology into a formidable tool for creating hyper-realistic manipulated content. With the ability to convincingly manipulate videos, images, and audio, deepfake technology can be used to create fake news, impersonate individuals, or even fabricate events, posing significant threats to public trust and societal stability. The technology has already been used to generate deepfakes for a number of the above-listed applications. Extending the complexities, this paper introduces the concept of *deepfake phylogeny*. Currently, multiple deepfake generation algorithms can also be used sequentially to create deepfakes in a phylogenetic manner. In such a scenario, deepfake detection, ingredient model signature detection, and phylogeny sequence detection performances have to be optimized. To address the challenge of detecting such deepfakes, we propose DeePhyNet, which performs three tasks: it first differentiates between real and fake content; it next determines the signature of the generative algorithm used for deepfake creation to determine which algorithm has been used for generation, and finally, it also predicts the phylogeny of algorithms used for generation. To the best of our knowledge, this is the first algorithm that performs all three tasks together for deepfake media analysis. Another contribution of this research is the DeePhyV2 database to incorporate multiple deepfake generation algorithms including recently proposed diffusion models and longer phylogenetic sequences. It consists of 8960 deepfake videos generated using four different generation techniques. The results on multiple protocols and comparisons with state-of-the-art algorithms demonstrate that the proposed algorithm yields the highest overall classification results across all three tasks.

**Index Terms**—Deepfakes, Phylogeny, Deepfake detection

## 1 INTRODUCTION

THE rapid advancements in generative networks have ushered in an era where the generation and manipulation of digital media have become remarkably sophisticated and accessible. This technological progress has resulted in the proliferation of deepfake content that is convincingly altered to misrepresent reality. Deepfakes pose significant threats to our society, privacy, and even national security, and are capable of swaying public opinion, spreading false propaganda, and impersonating other people's identities.

Deepfake generation has evolved and several complex deepfake scenarios have emerged. Deepfakes can now be generated using low-resolution videos and with multiple subjects in a video [1], [2]. Multiple algorithms have been developed that are capable of generating high-quality deepfakes replicating pose, expression, and lighting of the target subject [3], [4], [5]. Extending the domain of deepfake creation, we introduce the novel problem of ‘deepfake phylogeny’, where deepfakes are generated in a “phylogenetic” manner by employing multiple generative algorithms in a sequential manner (Fig. 1).

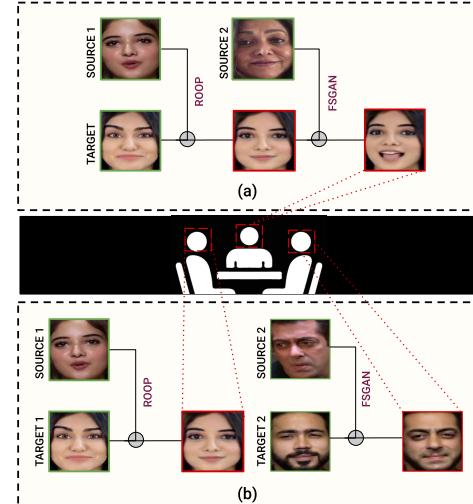


Fig. 1: Introducing the concept of deepfake phylogeny: (a) phylogenetic deepfake created with sequential use of multiple generative techniques to a single target and (b) phylogenetic deepfake created by replacing multiple targets in a video.

• K. Thakral, K. Narayan, S. Mittal, M. Vatsa, and R. Singh is with the Department of Computer Science and Engineering, Indian Institute of Technology, Jodhpur, India.

E-mail: thakral.1@iitj.ac.in, narayan.2@iitj.ac.in, mittal.5@iitj.ac.in, mvatsa@iitj.ac.in, richa@iitj.ac.in.

• H. Agarwal is with the Department of Electrical Engineering, Indian Institute of Technology, Jodhpur, India.

Email: agarwal.10@iitj.ac.in.

\*Equal Contributions by Agarwal and Narayan.

### 1.1 Defining Deepfake Phylogeny

The concept of “Deepfake Phylogeny” involves sequential application of various generative models to generate a deepfake video, as illustrated in Fig. 1. Similar to image phylogeny, this approach implies an evolutionary process for deepfakes and presents new challenges associated

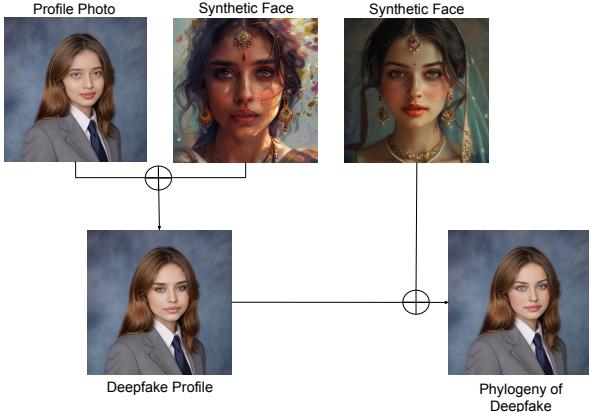


Fig. 2: Illustration of the generation of phylogenetic deepfakes and their application in real-world scenarios such as fake Instagram profiles and LinkedIn profile pictures.

with phylogeny in deepfakes. In deepfake generation, a ‘source’ represents a subject whose facial features are transferred to a ‘target’ subject’s video. In ‘deepfake phylogeny’ the generative algorithms are applied such that the generated video has multiple sources. This issue can arise in two scenarios. In the first scenario, as shown in Fig. 1(a), the facial features of a single target subject are manipulated sequentially using a series of deepfake generation techniques. Each technique leaves its unique generative signal on the target, leading to a layered effect of manipulations. In the second scenario, shown in Fig. 1(b), multiple *targets* in a video are manipulated using different *sources*. Each target receives facial features from a distinct source, resulting in a video where each individual exhibits characteristics from different generative models. Both scenarios lead to a mix of generative signals in the generated deepfake, creating a complex web of transformations that can be challenging to untangle. Deepfake phylogeny is nowadays popular in the dynamic landscape of social media where malicious actors are frequently employing a combination of techniques to generate and then alter fake images or videos (as shown in Fig. 2) – a trend particularly observed on platforms like Instagram [6], [7]. Conventional single-technique attribution models are inadequate in these scenarios due to a lack of training on such complex manipulations. With this concept, we introduce the following research questions:

**RQ1:** How well can a model differentiate between real and fake videos generated in a phylogenetic manner?

**RQ2:** Can a model extract the signature of each generative algorithm involved in the phylogenetic deepfake?

**RQ3:** Is the model able to predict the order of phylogeny in the deepfake video?

## 1.2 Research Contributions

This paper introduces the concept of the deepfake phylogeny and proposes the DeePhyV2 dataset, which contains 8960 phylogenetic videos<sup>1</sup> generated using four different

1. This paper extends our preliminary research published in the International Joint Conference on Biometrics 2022 [8].

techniques, namely - FSGAN [3], FaceShifter [5], FaceSwap [4], and diffusion model based Roop [9]. Focusing on the first scenario (Fig. 1(a)), the dataset is divided into three sections - “Succession 1”, “Succession 2”, and Succession 3” with 1120, 4480 and 3360 videos, respectively. Each succession utilizes a different generative algorithm, thereby enhancing the dataset complexity. Succession 1 consists of videos where the face of the target is swapped once. In Succession 2, the face of the target is swapped with two different sources. Similarly, in Succession 3, the face of the target is swapped with three different sources.

While there have been significant research efforts on deepfake detection, no one has focused on the above mentioned research questions. In this research, we introduce a novel deepfake detection algorithm termed as “DeePhyNet”. The DeePhyNet model captures spatio-temporal inconsistencies introduced by the generative models and exploits them in order to discriminate between model signatures. DeePhyNet consists of three steps: (i) frame extraction and their division into packets, (ii) extraction of spatio-temporal features from the packets, and (iii) projection of these features to a frequency space. The frame extraction step maintains the temporal coherence by transforming the input video into a sequence of equidistant frames. The sequence is then broken into packets where local and global spatio-temporal features are captured as shown in Fig. 1(b). The features are then projected to a frequency space to amplify the extracted artifacts. Extensive experiments performed on deepfake detection, model attribution, and phylogeny sequence prediction tasks show state-of-the-art results. This is also a step towards explainable deepfake detection wherein the detection models can attribute a deepfake to a particular generative model.

## 2 RELATED WORK

Deepfakes are evolving rapidly, and so is the landscape of deepfake research. These are two major aspects of the research in deepfakes: generation and detection. Considering that this research involves deepfake dataset generation and detection, the literature review section is arranged as follows: (i) deepfake datasets, (ii) deepfake generation, (iii) deepfake detection, and (iv) model attribution and phylogeny.

### 2.1 Deepfake Datasets

Over several years, the research community has proposed multiple deepfake datasets. The early deepfake datasets UADFV [10] and DeepfakeTIMIT [11] had a small quantity of videos and were of poor quality. FaceForensics++ [12] was the first big milestone dataset that served as the foundation for deepfake detectors. It consists of 4,000 deepfake videos generated using four generative techniques and introduces the concept of deepfake detection under different compression levels. Dolhansky et al. [13] introduced the DFDC dataset, consisting of over 100,000 deepfake videos and 3,426 actors. The DFDC initiative significantly accelerated research in deepfakes, and many datasets followed after that. CelebDF [14] dataset contains 5,639 high-quality

TABLE 1: Summarizing the characteristics of the publicly available deepfake datasets.

Dataset	Real Videos	Deepfake Videos	Methods	Phylogeny
UADFV [10]	49	49	1	✗
FaceForensics [17]	1,004	2,008	1	✗
FaceForensics++ [12]	1,000	4,000	4	✗
DFDC [13]	23,654	104,500	8	✗
DeepFakeTIMIT [11]	320	640	2	✗
Deep Fakes Dataset [18]	70	70	NA	✗
CelebDF [14]	590	5,639	1	✗
KoDF [16]	62,166	175,766	6	✗
DF-Platter [2]	764	132,496	3	✗
DeePhy [8]	100	5,040	3	✓
DeePhyV2	100	8,960	4	✓

deepfake videos of celebrities. While existing datasets consisted of real videos captured in a controlled environment, Zi et al. [15] proposed WildDeepFake, comprising real videos captured in the wild to emulate real-world deepfakes. Recent datasets like KoDF [16], DF-Platter [2], and Open Forensics [1] have elevated deepfake research by introducing complex scenarios like occlusion, multiple faces in one image, and deepfake phylogeny and offering multiple annotations to enhance the dataset usability. These annotations facilitate various tasks, such as deepfake model attribution and face segmentation. The current development of deepfake datasets focuses on a deeper understanding of the multifaceted nature of deepfake technology and its implications. Table 1 provides a comprehensive list of publicly available deepfake and phylogeny datasets.

## 2.2 Deepfake Generation

Multiple techniques have been proposed for deepfake generation in the literature. Here, we focus on identity swap deepfakes in which one person’s face in the video is replaced with another person. Earlier works involving face swapping [19], [20] were proposed as a means of preserving the privacy of individuals. Faceswap [21] is a classical computer graphics-based technique. Face2Face [22] performs real-time facial reenactment, with consistent facial expression transfer using a dense photometric consistency measure. Thies et al. [23] adopt neural textures to perform facial reenactment. Korshunova et al. [4] treat face-swapping as a style-transfer problem. Garrido et al. [24] proposed an image-based solution for identity swapping, preserving the source face expression. Open-source software like DeepFaceLab [25] and DeepFakes [26] made the creation of deepfakes more accessible. The development of GAN resulted in significant improvement in the quality of deepfakes. Models like StyleGAN [27] enable face swap leveraging the latent space representation of faces. Nirkin et al. [3] proposed FSGAN, which can perform subject-agnostic face swapping and facial reenactment without re-training for each identity. FaceShifter [5] uses a two-stage framework for high-fidelity and occlusion-aware face swapping. Recently, Wang et al. [28] proposed HifiFace, which uses a 3D shape-aware identity to control the face shape with geometric supervision from 3DMM and 3D face reconstruction methods. In a recent study, Agarwal et al. [29] showed the impact of freely available face-swapping tools in fooling face recognition algorithms. Some works [9], [30] take advantage of diffusion models to generate highly realistic deepfake videos.

## 2.3 Deepfake Detection

Early works [31], [32], [33] utilize image classification networks to extract feature vectors and perform binary classification. However, these methods were prone to overfitting and did not generalize well on unseen data. Li et al. [34] proposed a deepfake detector based on detecting eye blinking in the videos. Chollet et al. [35] proposed XceptionNet based on separable convolutions with residual connections. Nguyen et al. [33] proposed a deepfake detection architecture based on capsule networks. Face Warping Artifacts with spatial pyramid pooling [36], Face X-Ray [37], and Spatial Phase Shallow Learning (SPSL) [38] are more generalized detectors and perform well on different manipulations. Sabir et al. [39] proposed a recurrent neural network that uses temporal cues for video forgery detection. Sun et al. [40] proposed a dual contrastive learning objective for general face forgery detection. Some works [41], [42] leverage the frequency information that provides clues for face forgery detection. In recent years, many deep-learning-based methods have been proposed [43], [44], [45], [46].

## 2.4 Model Attribution and Phylogeny

Early works in deepfake source detection were mainly image-based and exploited attributes of synthetic imagery such as GAN model fingerprints [47], [48]. Ciftci et al. [49] propose a GAN-based framework that leverages the variation in the heartbeat of deepfake videos to find the source method employed to generate the deepfake. Recently, Marra et al. [50] studied the GAN fingerprints based on photo-response non-uniformity pattern and demonstrated its effectiveness on GAN source identification. Jain et al. [51] propose a hierarchical CNN architecture to distinguish between retouched and GAN-generated images and identify the source GAN model. Yu et al. [48] can attribute fully synthetic images to their respective source GAN model. Zhang et al. [52] optimize over the source of entropy of each generative model to probabilistically attribute a deepfake to one of the models. Yang et al. [53] propose DNA-Net that identifies GAN fingerprints by employing patchwise contrastive learning and pre-training on image transformations. Inspired by the work on phylogeny in different domains [54], [55], [56], in 2022, for the first time, we proposed the application of phylogeny in the context of deepfakes [8].

## 3 DEEPHYV2: DEEPFAKE PHYLOGENY DATASET

The proposed DeePhyV2 dataset contains phylogeny sequences generated using FSGAN [3], FaceShifter [5], FaceSwap [21], and diffusion based Roop [9]. The dataset contains a total of 32 phylogeny sequences and 8960 deepfake videos created using three different kinds of model architectures. The new sequences have been highlighted in Table 2, which were not present in the predecessor. The dataset contains 9060 videos and it is 78.5% greater than its predecessor in terms of the total number of fake videos. The samples from the proposed dataset can be visualized in Fig. 3. The dataset is publicly available at <https://iab-rubric.org/deephv2-database>.

**Phylogeny Procedure:** The DeePhyV2 dataset contains videos generated using four different generative techniques



Fig. 3: Illustration of phylogenetic samples of the DeePhyV2 dataset from different successions.

and comprises three successions, namely - “Succession 1”, “Succession 2” and “Succession 3” with 1120, 4480 and 3360 videos, respectively. Different generative techniques employed three different generative architectures namely - “Convolutional Autoencoders”, “GANs” and “Diffusion”. In Succession 1 videos, the target’s face is interchanged once with that of the source. Following this, the target’s face is switched with two distinct sources in Succession 2 videos and similarly with three different sources in Succession 3 videos. For each Succession, swapping does not necessarily involve the same generative methodology. The generation methods utilized in each Succession are sequentially listed in Table 2. The previously generated deepfake from Successions 1 and 2, respectively, is subjected to the application of the generation methodologies in Successions 2 and 3. Each row of Table 2 indicates the cumulative sequence in which the generation techniques are used to generate phylogeny. For example, row 1 depicts that in “Succession 1”, face swapping is done using Roop. In “Succession 2”, a phylogeny is produced by swapping the face of an existing deepfake face (made using FSGAN). By employing the FaceSwap technique to switch faces in an already phylogenetic deepfake face (created with Roop followed by FSGAN), “Succession 3” deepens the phylogeny even further. A set of 280 deepfake videos, their accompanying generating methods, and their order are rendered by each block in Table 2.

**Dataset Statistics:** The DeePhyV2 dataset comprises a total of 100 real videos and 8960 deepfake videos made with several succession of face swapping. These videos have variations in the generation technique in each succession. Videos filmed in controlled environments, where the variance of various lighting situations is not fully caught, are found in the majority of existing deepfake datasets. Additionally, in contrast to real-life situations, there is little diversity in background, position, and emotion in controlled environment settings. The real videos of the people in the DeePhyV2

TABLE 2: Methods applied to various dataset successions. Succession 2 and 3 represent the application of deepfake techniques twice and thrice, respectively.

	Succession 1	Succession 2	Succession 3
FSGAN	FaceSwap	FaceShifter	
	FaceShifter	FaceSwap	
	FSGAN	<b>Roop</b>	
	<b>Roop</b>	FSGAN	FaceShifter
FaceSwap	FSGAN	FaceShifter	
	FaceShifter	FSGAN	
	FaceSwap	<b>Roop</b>	
	<b>Roop</b>	FSGAN	FaceSwap
FaceShifter	FaceSwap	FSGAN	
	FaceShifter	<b>Roop</b>	
	<b>Roop</b>	FSGAN	FaceShifter
		FaceSwap	
<b>Roop</b>	FaceShifter	FSGAN	
	<b>Roop</b>	FaceShifter	
	FSGAN	FaceShifter	
	FaceSwap	FSGAN	

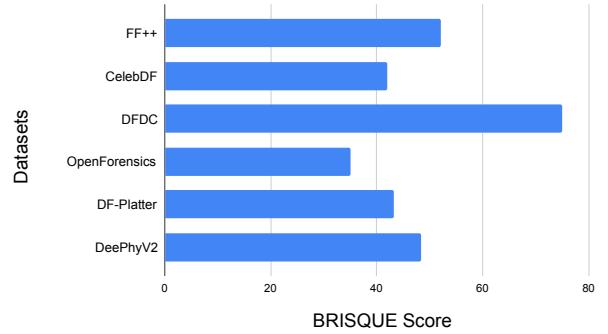


Fig. 4: Comparing BRISQUE scores of DeePhyV2 with existing deepfake detection datasets. The lower the BRISQUE score, the higher the visual quality of the dataset.

dataset are taken from YouTube<sup>2</sup>.

**Visual Quality Assessment:** The visual quality of the proposed DeePhyV2 dataset is evaluated using the BRISQUE score [57]. The complete DeePhyV2 dataset exhibits an average BRISQUE score of 48.28. When analyzed succession-wise, the BRISQUE scores for the first, second, and third successions are 44.39, 48.94, and 51.52, respectively. We also analyzed the BRISQUE of existing datasets, including FaceForensics++, DFDC, CelebDF, OpenForensics, and DF-Platter, and the results are summarized in Fig. 4. The BRISQUE scores for these datasets are approximated based on Narayan et al. [8]. These scores emphasize the high quality of the proposed dataset, indicating its complexity due to multiple deepfake iterations.

**Size and Format:** The extended dataset is around 50 GBs in size. Videos in the dataset are around 20 seconds in duration on average and are in 720p resolution. They have 25 frames per second and are stored in MPEG4.0 format.

## 4 PROPOSED DEEPHYNET

Deepfake videos often display local artifacts around the lips, eyes, and ear regions that occur over a sequence of frames.

2. The research is approved by the institutional ethical review committee.

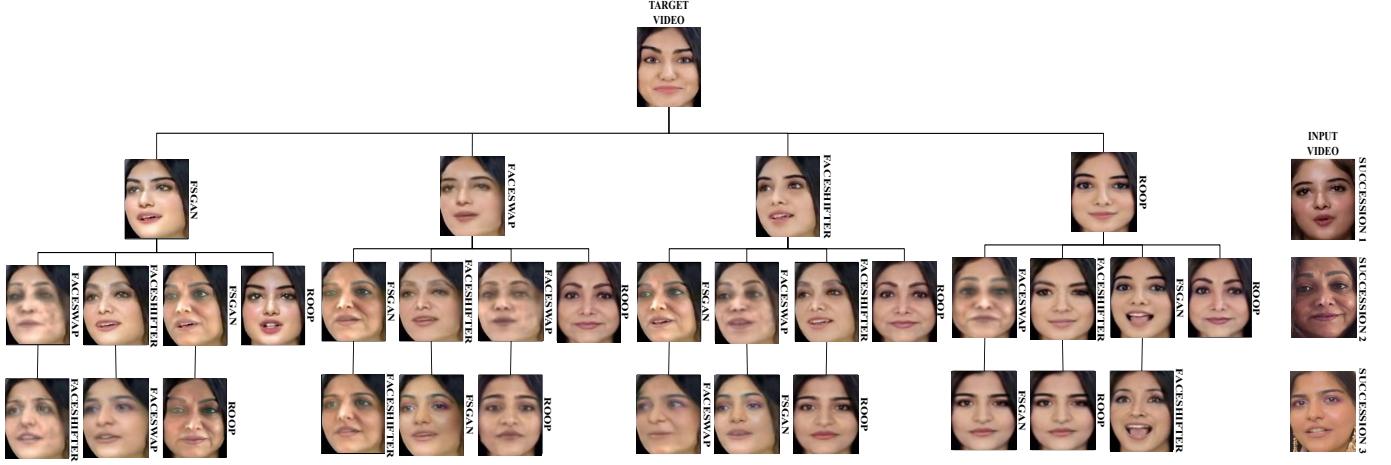


Fig. 5: Visualization of the complete generation process of phylogenetic deepfakes using four generation techniques, namely, FSGAN, FaceSwap, FaceShifter, and Roop. The phylogeny tree illustrates the output generated when a target face is swapped onto the input video using each of the aforementioned generation techniques.

In addition, deepfake videos also retain the residual signature of the generative model involved in their generation. These inconsistencies occurring over a sequence of frames can be utilized to detect deepfakes as well as to identify the corresponding generative model [48].

In this work, we propose an approach capable of elucidating these artifacts present in deepfake videos introduced due to the succession of multiple phylogenies. For this, we focus on capturing the spatial artifacts as well as temporal inconsistencies in features. This is achieved by utilizing a combined spatio-temporal feature extractor  $\mathcal{F}$  with weights  $\theta = \{\phi, \zeta, \psi\}$  responsible for processing features in the spatial domain and then further identifying temporal inconsistencies. Given a phylogenetic fake video  $\mathcal{V}$  with  $n$  frames  $\mathcal{V} = \{f\}_{i=1}^n$ , it is first divided into  $p$  packets  $v = \{f\}_{i=1}^m$ , where each packet sequence consists of  $m$  frames. Throughout the training process, every sequence of video frames  $v = \{f\}_{j=1}^m$  assumes a critical role in discerning local temporal inconsistencies, whereas the analysis of the entire video  $\mathcal{V} = \{v\}_{i=1}^p$  is essential for understanding global temporal inconsistencies. The extracted spatio-temporal features are then projected into the frequency space to further enhance the detection of phylogeny signatures in a video. The entire procedure is performed in three steps discussed below.

**Step 1: Packet Generation.** The proposed approach is designed to process a sequence of frames as a video input. To extract features from both the spatial and temporal domains for an input video  $\mathcal{V}$ , we devise a frame extraction method to extract a total of  $k$  frames, which are further divided into  $p$  packets with  $m$  frames each. Thus, given a video input,  $\mathcal{V} = \{f\}_{i=1}^n$  with  $n$  frames, it is transformed into output video  $\mathcal{V}'$  consisting of  $k$  frames extracted at equal intervals.

$$\mathcal{V}' = \left\{ \{f\}_{i=1}^n \mid i \in \left[ i \times \left( \frac{n-2}{k-1} \right) \right], 1 \leq i \leq n, 1 \leq k \leq n \right\} \quad (1)$$

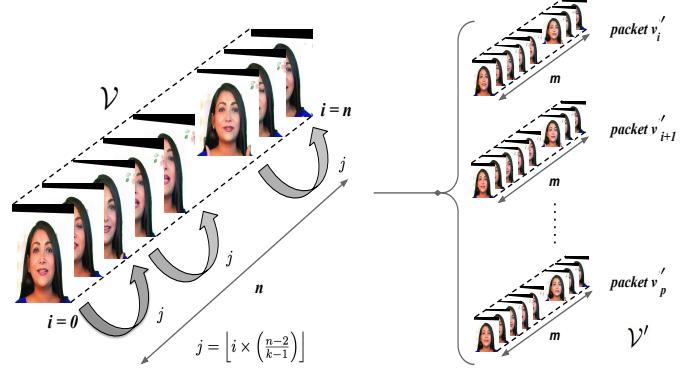


Fig. 6: Pictorial representation of the transformation of input video  $\mathcal{V}$  to the sequence of packets  $\mathcal{V}'$ .

The output video  $\mathcal{V}'$  is then broken down into  $p$  packets where  $\mathcal{V}' = \{v'\}_{i=1}^p$  and packet  $p$  is  $v' = \{f\}_{i=1}^m$  with constraints  $p \leq k$  and  $m \leq n$ . Hereafter, each packet  $\{v'\}_{i=1}^p$  serves as an input to  $\mathcal{F}(\cdot; \theta, \delta)$ , where  $\delta$  are the weights of the fully connected layer trained for a target task. The transformation of  $\mathcal{V}$  to  $\mathcal{V}'$  in this step not only ensures temporal consistency but also reduces computational overhead throughout the training process. This process is pictorially represented in Fig. 6.

**Step 2: Spatio-Temporal Feature Extraction.** In the context of deepfake detection, capturing both spatial and temporal features is essential. This is because spatio-temporal inconsistencies could emerge across various facial regions spanning over multiple frames at discrete intervals. For deepfake detection, we utilize  $\mathcal{F}(\cdot; \phi, \zeta)$ , designed to extract intricate spatial artifacts and nuanced temporal inconsistencies embedded within the input deepfake video  $\mathcal{V}$ .

The feature extraction overall is a two-phase process. The first phase is employed to extract only spatial features  $z_l^i = \mathcal{F}(v'_i; \phi)$  for each sequence frame present in a packet. Given an  $i^{th}$  packet  $v'_i$ , each frame is fed to a robust backbone network, parameterized by weights  $\phi$ . Subse-

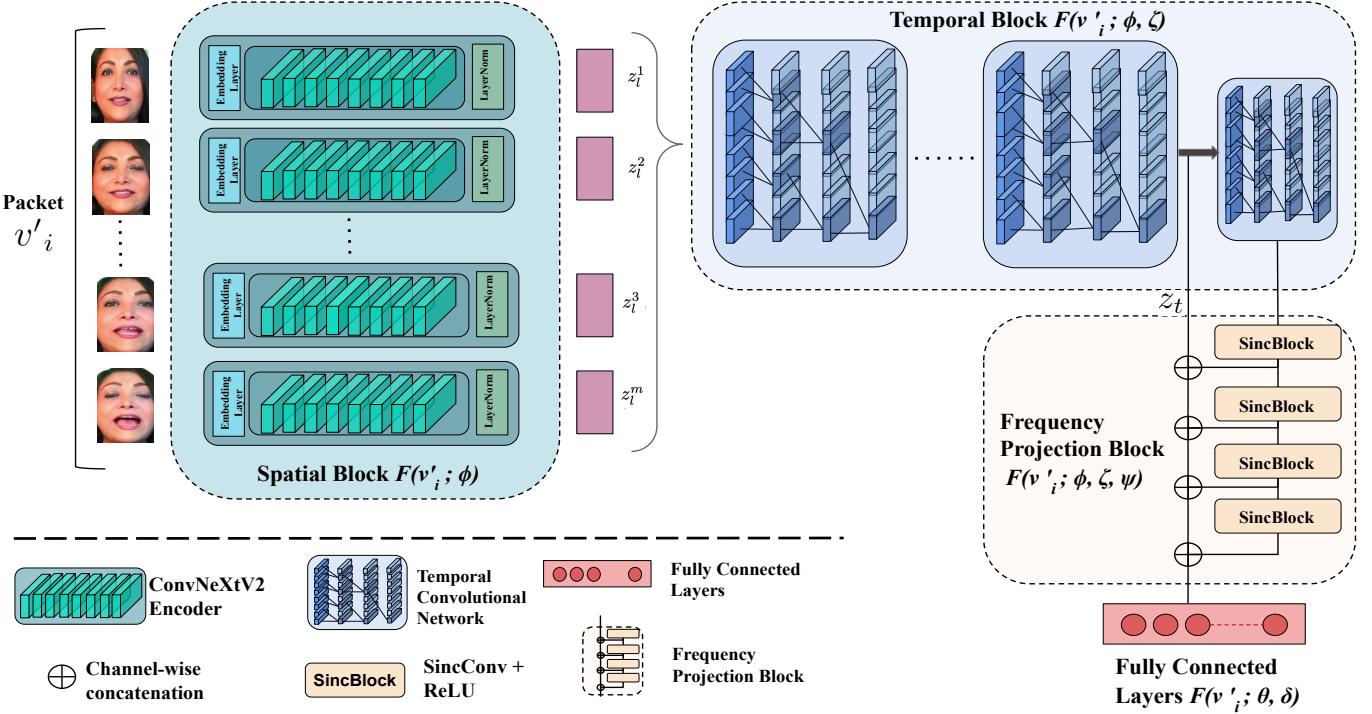


Fig. 7: Illustration of the complete training process of the proposed approach for video deepfake phylogeny detection.

quently, in the second phase, to encapsulate the temporal features intrinsic to a specific packet, output spatial features  $z_l^i = \mathcal{F}(v'_i; \phi)$  are concatenated and are then processed through a temporal network, parameterized by weights  $\zeta$ . The output features  $z_t = \mathcal{F}(v'_i; \phi, \zeta)$  from the temporal network ensures the incorporation of both spatial artifacts and temporal inconsistencies in the final feature embeddings. These resultant embeddings are then utilized in the final step of the training of the detection task.

In other words, the network is trained individually for each packet, but the inclusion of all  $p$  packets from an input video equips the network with a comprehensive temporal perspective. This allows for an understanding of the temporal dynamics across the entire video sequence. Therefore, our proposed setup not only processes the spatial features from each frame but also captures local temporal information through each packet and global temporal information through all packets. This approach provides the network with significantly more locations to extract artifacts from. The spatial and temporal blocks utilized in this step are represented with green and blue blocks in Fig. 7, respectively.

**Step 3: Feature Extraction in Frequency Domain:** Media, when generated synthetically, captures the signature of the generating model. Recent research suggests these signatures can be pronounced when visualized in frequency space [48], [58]. To detect and identify the presence of the signature of generative models in the phylogenetic deepfake, we project the embeddings  $z_t$  to the frequency space by first transforming  $z_t$  into  $1-D$  space via temporal block and then utilize them further for the target task. For this, a Frequency Projection Block (FP Block)  $\{\psi\}_{i=1}^q$  with  $q$  layers is used, and

output from each layer is concatenated with embeddings  $z_t$ ,

$$z_{\nu} = \{z_t \oplus \mathcal{F}(z_t; \{\psi\}_{j=1}^i)\}_{i=1}^q \quad (2)$$

where  $\oplus$  is the concatenation operation and  $z_{\nu}$  are the final embeddings from  $\mathcal{F}(V'; \phi, \zeta, \psi)$  which can be utilized for the target task in deepfake phylogeny.

In our implementation, we used ConvNeXtv2 [59] as the backbone ( $\phi$ ) to extract spatial features  $(\{z_l\}_{i=1}^p)$  and Temporal Convolution Networks [60] ( $\zeta$ ) to extract temporal features ( $z_t$ ). ConvNeXt inherits its components from ConvNets as well as Vision Transformers, enriching it with the capabilities to extract spatial artifacts as well as local temporal inconsistencies. Since  $n$  and  $k$  can vary for a deepfake video  $V$ , we utilize Temporal Convolution Networks (TCNs) as their performance is unaffected (unlike other temporal architectures) when  $n$  and  $k$  are scaled. To extract features from the frequency domain and pronounce the signature of the generative model(s) used in the generation, we utilize Sinc Convolutions [61] ( $\{\psi\}_{i=1}^q$ ) and project the overall features ( $z_{\nu}$ ) to the frequency space. In Fig. 7, this step is represented with the yellow block.

**Inferencing:** The proposed algorithm is evaluated with the standard video detection procedure in which the prediction is computed as  $\hat{y} = \mathcal{F}(\{f'_i\}_{i=1}^p; \theta, \delta)$ , where  $\delta$  denotes the weights of the classification layer. In other words, during the testing phase, the video is divided into 50 packets, and the first frame from each packet is fed into the model for final prediction.

## 5 EXPERIMENTAL RESULTS AND ANALYSIS

We conduct various experiments on the proposed DeePhyV2 dataset to establish the benchmark performance and evaluate the performance of the proposed algorithms under different protocols. We further compare the performance of different baseline algorithms with the proposed approach and perform ablation experiments to investigate the impact of each component of the proposed approach.

### 5.1 Evaluation Protocols

The dataset is divided into a train, validation, and test set with a 70-10-20 split. For the baseline models, ten frames are extracted from each video. The split of the identities in the train and test set is subject-disjoint. The baseline models and the proposed approach are evaluated for the following protocols:

**Protocol 1:** This protocol concerns with detecting deepfakes i.e. real vs fake on the proposed dataset. In this experiment, we select the fake videos from succession 1 and all real videos to keep this protocol consistent with the conventional real vs fake deepfake detection task. We train the deepfake detection models in a binary-classification setting to classify an input data sample as “real” or “fake”. We report both class-wise accuracy and overall accuracy for the baseline models on the test set of this experiment.

**Protocol 2:** In this protocol, the task is to determine if a specific generative algorithm was used in the sequence of algorithms that produced the deepfake. The training set of DeepPhyv2 is used to train a model in a multi-label classification setting. The performance of various models is evaluated based on their ability to accurately predict all the deepfake generation techniques used in the creation of a phylogenetic deepfake. The sequence in which the generation techniques were used to create the deepfake is not reflected in this protocol’s performance. The overall accuracy, which indicates the model’s ability to correctly identify all the generation techniques used, regardless of their order, is reported.

**Protocol 3:** This protocol presents a complex model attribution task where the model is required to determine the specific order in which generative techniques were applied to create a phylogenetic deepfake. The dataset for this protocol was generated by sequentially applying all four generative techniques—FSGAN, FaceShifter, FaceSwap, and Roop—to create deepfake videos, resulting in 32 unique sequences (as outlined in Table 2), each representing a distinct class in the multi-class classification task. The model is trained to recognize and classify these sequences by identifying the subtle signatures and artifacts left by each technique. The evaluation of the model is based on its accuracy in predicting the correct sequence of generative techniques applied to unseen deepfake videos, thereby demonstrating its ability to handle the complexities of phylogenetic deepfake generation.

In the above-mentioned Protocol-2 and Protocol-3, we carry out two experiments. The first experiment involves training the model on the training set of the proposed DeepPhyv2 dataset and evaluating its performance on the test set, referred to as the Train experiment. In the second experiment, we use existing datasets that contain deepfake

samples created using FSGAN, FaceSwap, and FaceShifter to pre-train a model, which is referred to as the Finetune experiment. For this purpose, we collect samples of FaceSwap and FaceShifter-generated deepfakes from the FaceForensics++ dataset [12]. The FSGAN deepfakes are created using raw videos from the CelebDF [14] dataset. The model is pre-trained on these samples in a multi-label classification setting. Following this, the models are fine-tuned on the training set of our proposed dataset, and their performances are evaluated.

### 5.2 Comparison Algorithms and Evaluation Metrics

Detection of phylogeny can be viewed from the lens of spatial as well as temporal domain. For this, the following ConvNet-based algorithms and Transformer-based backbones are trained for the baseline and comparison experiments. Their performance is evaluated based on the protocols discussed earlier.

**MesoNet:** Afchar et al. [31] proposed a ConvNet-based deepfake detection method with two variants, namely, Meso4 and Inception modules-based [36] MesoInception4.

**FWA:** Li et al. [36] explore the artifacts in the deepfake videos for fake detection with a ResNet50 [62] backbone. These artifacts are caused due to affine face-warping transformations.

**DSP-FWA:** Li et al. [36] extended FWA with the introduction of a dual pyramid strategy at both image and feature levels.

**CapsuleNet:** Nguyen et al. [63] proposed a CapsuleNet which builds over Capsule architecture [64] with VGG19 [65] backbone. It utilizes the spatial relationships of features in the input image.

**XceptionNet:** Chollet et al. [35] proposed an InceptionNet-based architecture termed extreme Inception-Net (Xception-Net). It consists of separable convolutions with residual connections across the network.

**Vision Transformer:** Dosovitskiy et al. [66] proposed an architecture for applying a Transformer-encoder on image-recognition tasks. The architecture utilizes self-attention only without the use of convolution operations.

**MobileViT:** Mehta et al. [67] proposed a light-weight and general-purpose ViT for mobile devices. It combines the strengths of ConvNets and ViTs by using convolutions to learn local representations and transformers to learn global representations.

**SwinViTv2:** Liu et al. [68] proposed a vision backbone that uses shifted windows and hierarchical structure to compute local and global self-attention. It introduces several techniques to improve training stability, resolution transfer, and memory efficiency.

**ConvNeXtv2:** Woo et al. [59] is another recent vision backbone that aims to modernize the standard ConvNet by incorporating some design principles from Transformers. It consists of four stages, each with a number of blocks that have a shifted window self-attention layer and a feed-forward network layer.

**Evaluation Metrics:** In the first protocol, we report both the classwise accuracy and the overall video-wise accuracy for the deepfake detection task. The second protocol involves evaluating the performance of various detection models in a multi-label classification setting. The overall video-wise

accuracy is defined as the number of instances where all employed generation techniques are correctly identified, excluding cases where only a subset of these techniques are accurately predicted.

In the third protocol, we assess the model’s video-wise accuracy in a multi-class classification context, treating each unique sequence as a separate class. For all these protocols, video-wise accuracy is determined by majority prediction. Specifically, if a certain label is predicted for 5 or more out of 10 frames with a threshold of 0.5, that label is considered the final output.

### 5.3 Implementation Details

We next discuss the generation details of the DeePhyV2 dataset and implementation details of the proposed DeePhyNet approach for reproducibility:

**Generation Details:** The dataset’s source videos were collected from YouTube and feature subjects of Indian origin. The videos were generated using FaceSwap<sup>3</sup>, FaceShifter<sup>4</sup>, FSGAN<sup>5</sup>, and Roop<sup>6</sup> techniques with their open-source codes. For FaceSwap, each video was created after 8 hours of training on two Nvidia DGX A100 systems with 8 GPUs of 80GB memory each. Similarly, videos were generated by utilizing pre-trained weights of FaceShifter with default parameters on two Nvidia RTX 3090 GPUs of 24 GB memory each. Additionally, we fine-tuned the re-enactment generator of FSGAN for each source video and performed inferencing using Nvidia DGX A40 with 48 GB memory. To generate the deepfake videos using Roop, their open-source model was utilized, and generation was performed on four Tesla V-100 GPUs. The dataset generation process took over 1400 hours with parallel use of the above-mentioned GPUs.

**Benchmarking Details:** The baseline experiments for the proposed dataset were conducted on a Nvidia DGX station with four Tesla V-100 GPUs, each consisting of 32 GB memory. The frames from videos were extracted using DSFD [69]. The number of frames in videos varies between 500 to 600, and we extract 450 frames and divide them into 10 packets with 50 frames in each. Protocol 1 and protocol 2, are trained using binary cross-entropy loss, whereas protocol 3 is trained using the cross-entropy loss. While training for each protocol, the proposed algorithm and baseline models were trained for 30 epochs with early stopping using Adam Optimizer, having a learning rate of 0.0001 and keeping the rest of the parameters as specified in the respective papers.

**Architecture Details:** DeePhyNet employs ConvNextv2 encoder as the Spatial Block with standard 8 ConvNeXt stages preceded by a positional embedding layer and followed by a LayerNorm layer. A standard ConvNeXt stage primarily comprises a depthwise 2D convolutional layer, GeLU activation function, and Global Response Normalization layer. The temporal Block consists of 8 sub-temporal convolutional network blocks with kernel size 7, stride 1, and dilation rates increasing from 1 to 128. The padding size is adjusted according to the kernel size and dilation rates. The number

TABLE 3: Comparing DeePhyNet with state-of-the-art algorithms for protocol 1 (deepfake detection task) on the DeePhyV2 dataset.

Model Type	Models	Accuracy (%)		
		Real	Fake	Overall
ConvNet based	XceptionNet	35.71	85.29	66.66
	MesoNet	45.84	91.17	74.07
	MesoInceptionNet	35.88	94.11	72.22
	FWA	45.85	94.11	75.92
	DSP-FWA	85.62	82.35	83.33
	CapsuleNet	85.00	91.17	88.88
Transformer based	MobileViT	90.84	91.17	90.74
	ViT	80.57	76.47	77.77
	SwinT	85.45	88.23	87.03
	ConvNeXt	90.05	91.92	90.74
<b>DeePhyNet (Proposed)</b>		80.18	99.99	<b>93.33</b>

of input and output features is kept at 128 for all these blocks except the first one, for which the number of input features is the same as spatial feature dimensions, which is 768. This is followed by a small temporal convolutional network, which serves as a bridge between the transformation of 128 features to a single feature for each  $i^{th}$  frame in a packet. These sub-temporal blocks further utilize the ReLU activation function and a dropout with a probability rate of 0.2. The input of the frequency projection block is thus of a single channel and packet sequence length. The frequency projection block comprises four SincConv networks, which consist of a sinc convolutional layer with an audio sample rate equal to packet sequence length followed by ReLU activation function and dropout with a probability rate of 0.1. The final classification layer adapts a channel-wise concatenation of temporal block output and all four sinc Network outputs and has therefore, 132 as the number of input features. The softmax activation function is employed for the final model prediction.

**Training and Inference Details:** DeePhyNet is trained by extracting 9 packets from each video, where each packet contains 50 frames. This process results in a total of 450 frames. During testing, for an input video, we maintain the same frame count (50 frames), consistent with the training protocol. Specifically, we extract 50 packets from the original 450 frames and select the first frame from each of these 50 packets, yielding a total of 50 frames. DeePhyNet then concatenates these frames sequentially to make predictions. As a baseline comparison, we adhere to the standard protocol of thresholding with 5 or more frames for classification.

### 5.4 Results and Discussion

To evaluate the robustness of the DeePhyNet against phylogenetic deepfakes, the DeePhyV2 dataset is used. As discussed in section 5.1, we evaluate the proposed approach and baseline algorithms through three protocols.

**Protocol 1** In this protocol, we evaluate the baseline models and the proposed algorithm for the binary classification problem of distinguishing between real and fake videos. We compute class-wise and overall video accuracy and report them in Table 3. The results indicate that the proposed approach achieves state-of-the-art performance in terms of video accuracy, with a score of 93.33%. ConvNeXt and CapsuleNet follow with an accuracy of 90.74% and 88.88%,

3. <https://github.com/deepfakes/faceswap>

4. <https://github.com/Heonozis/FaceShifter-pytorch>

5. <https://github.com/YuvalNirkin/fsgan>

6. <https://github.com/s0md3v/roop>

	Succession 1	Succession 2	Succession 3
Data used			
Accuracy	93.33 %	92.38 %	93.42 %

Fig. 8: The performance of DeePhyNet when trained on each successions, individually.

respectively. We observe that the majority of the models achieve lower class-wise accuracy for the real class than for the fake class. This is attributed to the fact that the dataset is skewed against the real class, reflected in the performance. This behaviour also explains the diminished class-wise performance of DeePhyNet on the real class. The proposed algorithm processes the entire video as a single sample, which further accentuates the imbalance between the real and fake classes.

The accuracy of the baseline models in detecting deepfakes in the DeePhyV2 dataset is comparable to that of other existing deepfake datasets [12], [14], [11], [17], [15]. Since the proposed deepfake dataset with 8960 videos generated using four different techniques is designed to facilitate research on deepfake detection and phylogeny prediction, we suggest that a subset of the DeePhyV2 dataset (i.e., Succession 1 deepfakes) can be included in existing datasets for a robust training of the models. This will also ensure a better representation of Indian origin, under-represented in existing deepfake datasets.

We also evaluate the performance of DeePhyNet on binary classification between real and fake data from each succession under Protocol 1. For this, DeePhyNet was trained and tested individually for each succession. The results are presented in the Figure 8. These results demonstrate that DeePhyNet maintains consistent and high accuracy across all three successions, with a slight variation in performance. This suggests that the complexity introduced by phylogenetic deepfakes in succession 2 and 3 does not significantly affect the model's ability to differentiate between real and fake content. The comparable performance across successions indicates that phylogenetic deepfakes, despite involving more complex manipulations, are effectively detected by the proposed method.

**Protocol 2** This protocol assesses the proposed algorithm's ability to detect the signatures of deepfake generation algorithms. In this, two experiments are conducted: in the first (termed "Train" in Table 4), the algorithms are evaluated directly after being trained on the train set of the DeepPhyV2 dataset. In the second experiment (termed "Finetune"), the algorithms are first pre-trained on a curated deepfake dataset (details discussed in section 5.1) and then fine-tuned on the DeePhyV2 dataset for evaluation. The results of both experiments are reported in Table 4. The table shows that the proposed algorithm outperforms existing algorithms by a significant margin in both experiments, achieving an accuracy of 96.93% and 97.98% for Train and Finetune

TABLE 4: Comparing DeePhyNet with state-of-the-art algorithms for protocol 2 (model attribution task) on the DeePhyV2 dataset. Video-wise accuracy (in %) is reported for the Train and Finetune experiment.

Model Type	Models	Train	Finetune
ConvNet based	MesoNet	76.95	77.23
	MesoInceptionNet	77.68	78.18
	FWA	88.82	87.80
	DSP-FWA	79.13	84.28
	Xception	86.76	87.24
	CapsuleNet	92.84	86.96
Transformer based	MobileViT	91.18	91.35
	ViT	93.28	95.46
	SwinT	93.06	94.93
	ConvNeXt	93.56	93.79
<b>DeePhyNet (Proposed)</b>		<b>96.93</b>	<b>97.98</b>

TABLE 5: Comparing DeePhyNet with existing state-of-the-art algorithms for protocol 3 (phylogeny sequence prediction task) on the DeePhyV2 dataset. Video-wise accuracy (in %) is reported for the Train and Finetune experiment.

Model Type	Models	Train	Finetune
ConvNet based	MesoNet	2.62	3.41
	MesoInceptionNet	35.00	33.64
	FWA	65.68	65.82
	CapsuleNet	79.69	75.16
	Xception	66.77	63.78
	DSP-FWA	81.20	83.44
Transformer based	ViT	79.58	79.75
	MobileViT	82.94	86.85
	ConvNeXt	81.48	83.27
	SwinT	83.50	83.72
<b>DeePhyNet (Proposed)</b>		<b>90.93</b>	<b>90.79</b>

experiments, respectively.

For the proposed approach, the primary factor contributing to the significant improvement in detecting generative model signatures is the *frequency\_block*, which projects the input into frequency space, making residue signatures more apparent for the classifier to detect. This is consistent with the recent research that shows that the signature of generative models can be amplified in the frequency domain [58], and its contribution can also be observed in the ablation study of the proposed algorithm discussed in section 5.4. We also notice that ConvNet-based models, CapsuleNet and XceptionNet, achieve performances of 92.84% and 88.76% for the Train experiment and 75.16% and 63.78% for the Finetune experiment, respectively. Transformer-based architectures like SwinT and MobileViT achieve a performance of 81.48% and 82.94% for the Train experiment and 83.72% and 86.85% for the Finetune experiment, respectively.

**Protocol 3** is a task to predict the sequence of generative techniques used to create a phylogenetic deepfake. The performance of the proposed approach and different baseline algorithms on this task is shown in Table 5. The significant drop in the performance obtained by the baseline algorithms like MesoNet, MesoInceptionNet, FWA, and XceptionNet illustrates the challenging nature of this protocol. While the existing algorithms struggle, the proposed approach achieves state-of-the-art performance for both Train and Finetune experiments with an accuracy of 90.93% and 90.79%, respectively. Since DeePhyNet processes the video as a sequence, it searches for spatial artifacts while extract-

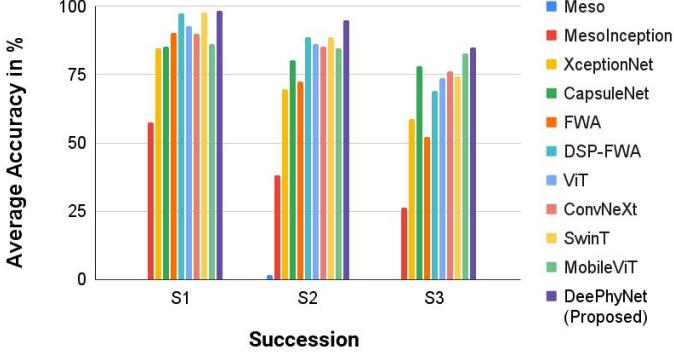


Fig. 9: Comparison of Succession-wise performance for different algorithms. Here, the models are evaluated for Protocol 3 without fine-tuning and in terms of average accuracy over each succession.

ing temporal inconsistencies. The spatio-temporal nature of the proposed approach enables it to capture the subtle artifacts in the face around the lip, nose, eyes, and ears and movements over time (i.e., over the sequence of frames) that are introduced by different generative techniques. As a result, DeePhyNet can accurately identify the order of the generation methods used to create a phylogenetic deepfake.

From Table 5, we also observe that the performance of DeePhyNet is followed by Swin-Transformer with a video accuracy of 83.50% and 83.72%, and DSP-FWA algorithm with a video accuracy of 81.20% and 83.44% for the Train and Finetune experiments, respectively. We also observe that the Meso and MesoInception along with the other remaining could barely detect the sequence of phylogeny, due to the challenging nature of this protocol.

In order to study the impact of successions on phylogenetic deepfakes, we evaluated the performance of various algorithms across all sequences, averaging the results for each succession. The average accuracy of these comparison algorithms for each succession is illustrated in Fig. 9. It is evident that as the number of deepfake successions increases, the performance of the models diminishes. This suggests that generating phylogeny through different deepfake algorithms makes the task more complex. However, it is noteworthy that while the proposed algorithm also follows this trend over successions, it consistently achieves the highest performance for each succession, with average accuracy of 98.40%, 95.02%, and 84.90% for successions 1, 2, and 3, respectively.

**Evaluation on Different Post-Processing Techniques:** We evaluate the proposed DeePhyNet on different post-processing operations, such as blurring, resizing, and compression on protocol 3, and compute the performance. In Table 6, we observe that minor changes in the input data do not significantly affect the performance of the proposed DeePhyNet. However, substantial modifications, such as resizing the image to almost half its size, blurring with  $\sigma = 0.7$ , and applying hard compression, lead to a decrease in performance when evaluated on the challenging Phylogeny sequence prediction task. These findings offer valuable insights into the algorithm's behavior. For instance, a reduction in image size to 128x128 alone leads

TABLE 6: Evaluation of DeePhyNet on different post-processing methods.

Operation		Accuracy
Resizing	224x224 (original)	90.93
	180x180	87.75
	150x150	79.92
	128x128	62.08
Blur	$\sigma = 0$ (original)	90.93
	$\sigma = 0.3$	90.88
	$\sigma = 0.5$	85.68
	$\sigma = 0.7$	53.57
Compression	raw (original)	90.93
	c23	64.54
	c40	12.41

TABLE 7: Performance comparison between DeePhyNet and current state-of-the-art algorithms on the Celeb-DFv2 dataset and Set C of the DF-Platter (DF-P) dataset in an out-of-domain setting. The reported results for the comparative algorithms are obtained from published literature. Here, FF++/DF-P Set A means the model is trained on FaceForensics++ (FF++) to evaluate performance on Celeb-DFv2 and trained on DF-Platter Set A to assess performance on DF-Platter Set C.

Algorithm	Trained on	Celeb-DFv2	DF-P Set C
MesoNet (2018) [31]	DF-P Set A	-	0.690
MesoInception (2018) [31]	DF-P Set A	-	0.690
FWA (2018) [36]	FF++/DF-P Set A	0.569	0.640
DSP-FWA (2018) [36]	FF++/DF-P Set A	0.693	0.770
Xception (2019) [?]	FF++	0.737	0.710
Capsule (2019) [63]	DF-P Set A	-	0.810
HICL (2022) [70]	FF++ (c23)	0.790	-
FTCN (2021) [71]	FF++	0.869	-
RealForensics (2020) [72]	FF++ (c23)	0.857	-
ICT (2022) [73]	FF++ (c23)	0.857	-
SBI (2022) [74]	FF++	0.931	-
SBI (2022) [74]	Private Data	0.870	-
SSPSL (2022) [75]	FF++	0.922	-
LSDA (2024) [76]	FF++	0.911	-
<b>DeePhyNet (ours)</b>	DeePhyV2	0.892	0.876

to a performance drop. A performance decline is observed only when the blurring magnitude increases significantly. At  $\sigma = 0.7$ , we note an accuracy of 53.57%, which can be attributed to the removal of deepfake artifacts by the blurring operation. A similar trend is observed with compression. Hard compression of c40 results in significant performance degradation, with an accuracy of just 12.41%. This decline can be attributed to the loss of deepfake artifacts during the blurring, resizing, or compression processes. This experiment suggests that the algorithm may not exhibit complete robustness against various post-processing tasks, particularly when the input is significantly compressed.

**Evaluation in Out-of-Domain Setting:** We assess the performance of the proposed DeePhyNet in an out-of-domain context. Specifically, we train DeePhyNet on the first succession data of DeePhyv2 for real versus fake classification under protocol 1 and subsequently test it on the Celeb-DFv2 [14] and set C of the DF-Platter [2] dataset. The performance achieved is compared with popular and state-of-the-art algorithms and is reported in Table 7. We observe that DeePhyNet consistently outperforms these state-of-the-art methods in terms of AUC on the DF-Platter dataset while maintaining competitive AUC scores on the Celeb-DFv2 dataset, further highlighting its robustness in detect-

	FaceShifter	FaceSwap	FSGAN	Roop
Trained on				
Accuracy on other generation methods	84.04 %	66.35 %	80.13 %	68.28 %

Fig. 10: Performance of DeePhyNet in an open-set setting. The algorithm is trained on the data of one generation algorithm and tested on rest three for protocol 1.

ing deepfakes across diverse unseen samples. Additionally, we have computed the accuracy of DeePhyNet on the Celeb-DFv2 dataset, observing a performance of 86.28%, which surpasses SSPSL [75], which achieved 84.60%. We also evaluated in-domain performance of DeePhyNet, along with existing algorithms, on Set A of the DF-Platter dataset. The AUC for these methods ranged from 0.59 to 0.83, with DeePhyNet achieving a leading AUC of 0.84.

**Evaluating for Open-Set Generalization:** We conduct empirical evaluations to assess the generalization capability of DeePhyNet. Specifically, we trained DeePhyNet using data from one deepfake generation algorithm (e.g., FaceShifter) in succession 1. We then tested the trained model on a combination of data from the remaining three algorithms (FaceSwap, FSGAN, and Roop). This process was repeated for each generation method to ensure a comprehensive evaluation. The results for Protocol 1 are presented in Figure 10. Notably, the model achieves its highest generalization performance when trained on FaceShifter, achieving an accuracy of 84.03% on previously unseen deepfake generation methods. FSGAN closely follows with an accuracy of 80.13%. However, performance is comparatively lower when the model is trained on FaceSwap and Roop, achieving accuracies of 66.35% and 68.28%, respectively. This performance disparity is likely due to the shared GAN-based nature of FaceShifter and FSGAN, allowing a model trained on one GAN-based algorithm to generalize more effectively to others, consistent with the observations in the literature [77]. While open-set generalization remains a challenging task, these results suggest that incorporating a broader range of generative techniques during training could help improve performance.

These findings highlights the inherent challenges in open-set generalization, particularly when different generative techniques are at play. To further enhance generalization, consider incorporating a diverse set of generation methods during the training process.

**Effects of Thresholding on Baseline Algorithms:** In the literature on deepfake detection methods, the standard practice is to assign the majority label based on the majority of frames. To test the impact of thresholding on phylogeny prediction, we conduct experiments and analyze the performance of baseline algorithms such as ViT and SwinTransformer. Specifically, we varied the number of frames within each packet for Protocol 3, as summarized in Table 8. We observe that an increase in the number of frames led to a decline in performance. This decrease can be attributed

TABLE 8: Effect of different thresholds for baseline algorithms when evaluated on protocol 3.

Number of Frames (Threshold)	Accuracy (%)	
	ViT	SwinT
3	87.08	85.85
5	82.94	83.50
7	74.88	75.72

TABLE 9: Performance comparison of DeePhyNet with baseline methods on threshold independent metrics for Protocol 3.

Detection Method	AUC	AP
ViT	0.812	0.798
MobileViT	0.849	0.830
ConvNeXt	0.860	0.825
SwinT	0.865	0.828
<b>DeephNet (ours)</b>	<b>0.958</b>	<b>0.919</b>

to the heightened challenge of maintaining consistent predictions across a larger frame set. Despite this challenge, achieving consistent predictions over more frames would imply a higher level of confidence in the prediction.

We also compute threshold-independent metrics such as AUC (Area Under the ROC Curve) and AP (Average Precision) to compare the performance of baseline algorithms with the proposed DeePhyNet algorithm. The baseline algorithms are executed (using a standard threshold of 5 frames) and evaluated for Protocol 3 of the phylogeny-sequence prediction task. The results are reported in Table 9. Even when considering threshold-independent metrics, we observe that DeePhyNet significantly outperforms the baseline algorithms on the proposed dataset.

**Ablation Analysis** To evaluate the contribution of each component of the proposed DeePhyNet approach, we perform the ablation, where we test the performance achieved by each component of the algorithm individually. This experiment is performed for protocol 3 of the sequence prediction task of the proposed algorithm, and results are reported in Table 10. To systematically evaluate the key components of our approach, we conduct ablation experiments on protocol 3, i.e., the sequence prediction task of the proposed dataset. We analyze the proposed algorithm from two aspects and present our analysis.

- **Effect of extracting spatio-temporal features:** Training the backbone on spatial features alone resulted in a performance of 81.48%. However, when the

TABLE 10: Ablation of the DeePhyNet with its different components plugged in. The performance is evaluated for protocol 3 (Train experiment) of the sequence prediction task on the proposed dataset with ConvNeXtV2 as the backbone and video-wise accuracy (in %) is reported.

Component	Model Description	Accuracy
Spatial Learning + Backbone	$\mathcal{F}(\{f_i\}_{j=1}^k; \phi, \delta)$	81.48
Spatial-Temporal Learning + Backbone + TCN	$\mathcal{F}(\{v'_i\}_{j=1}^p; \phi, \zeta, \delta)$	88.12
Spatial-Temporal Learning + Backbone + TCN + Frequency block (Proposed)	$\mathcal{F}(\{v'_i\}_{j=1}^p; \phi, \zeta, \psi, \delta)$	90.54

backbone was trained to learn from both spatial and temporal features, we observed a significant increase in performance by approximately 7%. This suggests that while the backbone extracts spatial artifacts, but when trained with packets and TCN network, the model identifies local and global irregularities. These irregularities extracted over the sequence aid the model in achieving better performance.

- **Effect of projecting the features to frequency space:** Since multiple generation algorithms are involved in phylogenetic deepfakes, extraction of residual signals of generative algorithms should aid in the prediction of phylogeny. From Table 10, it can be observed that the projection of features to the frequency domain leads to an improvement of approximately 2.5%. This suggests that our observation is consistent with the recent research [48], [58] and model signatures are more discriminative in frequency space.

**Limitations:** The proposed algorithm views deepfake videos as sequences and identifies spatiotemporal inconsistencies throughout the video for prediction. This method provides a substantial improvement in performance, but it necessitates additional computational time to extract frames and identify faces. This is because it processes each video as a single sequence, requiring an equivalence between real and fake data. Furthermore, the concurrent use of two networks to extract spatio-temporal features considerably escalates the computational time requirements.

## 6 CONCLUSION

In this research, we introduce the problem of deepfake phylogeny: the study of the evolutionary relationships between deepfakes created using different generation techniques. We propose DeePhyV2, a novel Deepfake Phylogeny dataset consisting of 8960 deepfake videos generated using four generation techniques sequentially for up to three successions. The DeePhyV2 dataset is designed to challenge deepfake detection models to identify the sequence of generation techniques used to create a deepfake image or video. We benchmark the dataset of various popular deepfake detection algorithms and proposed a novel approach termed DeePhyNet that achieves state-of-the-art performance in all three protocols of the proposed dataset. DeePhyNet processes video frames by extracting spatio-temporal features and enhancing the residual signs of generative models by mapping them onto a frequency domain. Detailed experiments are performed, and the key findings are:

**AQ1:** Experimental results indicate that current models are not effective in accurately distinguishing between genuine and fake faces in complex deepfake phylogeny settings. However, DeePhyNet exhibits superior detection capabilities, achieving detection accuracy of over 93%.

**AQ2:** The experiments demonstrate that the trained models can successfully identify the unique signature of each generative algorithm, with DeePhyNet achieving state of the art accuracy of around 97%.

**AQ3:** Addressing the most challenging research question, DeePhyNet manages to predict the sequence of deepfake creation with about 90% accuracy, highlighting potential areas for further enhancement.

The utility of our research extends to its proficiency in tracing the lineage of an image, achieved through the integration of model attribution and deepfake detection. Pre-trained networks find widespread application across a variety of fields, including the creation of professional artistic content and entertainment. It is imperative for creators to track the usage of their intellectual output to ensure they are duly credited, thereby preventing intellectual property infringement and the illicit use of deepfake technology.

## REFERENCES

- [1] T.-N. Le, H. H. Nguyen, J. Yamagishi, and I. Echizen, "Openforensics: Large-scale challenging dataset for multi-face forgery detection and segmentation in-the-wild," in *International Conference on Computer Vision*, 2021.
- [2] K. Narayan, H. Agarwal, K. Thakral, S. Mittal, M. Vatsa, and R. Singh, "Df-platter: Multi-face heterogeneous deepfake dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2023, pp. 9739–9748.
- [3] 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*, 2019, pp. 7184–7193.
- [4] I. Korshunova, W. Shi, J. Dambre, and L. Theis, "Fast face-swap using convolutional neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3677–3685.
- [5] L. Li, J. Bao, H. Yang, D. Chen, and F. Wen, "Advancing high fidelity identity swapping for forgery detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5074–5083.
- [6] "Ai influencers," <https://tinyurl.com/w7wbcayf>, 2024, accessed: 2024-03-18.
- [7] "Virtual influencers," <https://tinyurl.com/mvmjs4us>, 2024, accessed: 2024-03-18.
- [8] K. Narayan, H. Agarwal, K. Thakral, S. Mittal, M. Vatsa, and R. Singh, "Deepfy: On deepfake phylogeny," in *IEEE International Joint Conference on Biometrics*, 2022, pp. 1–10.
- [9] "ROOP," <https://github.com/s0md3v/roop>, 2023, [Accessed: 04-October-2023].
- [10] X. Yang, Y. Li, and S. Lyu, "Exposing deep fakes using inconsistent head poses," *CoRR*, vol. abs/1811.00661, 2018. [Online]. Available: <http://arxiv.org/abs/1811.00661>
- [11] P. Korshunov and S. Marcel, "Deepfakes: a new threat to face recognition? assessment and detection," *arXiv preprint arXiv:1812.08685*, 2018.
- [12] 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*, 2019, pp. 1–11.
- [13] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The deepfake detection challenge (dfdc) dataset," *arXiv preprint arXiv:2006.07397*, 2020.
- [14] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-df: A large-scale challenging dataset for deepfake forensics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3207–3216.
- [15] B. Zi, M. Chang, J. Chen, X. Ma, and Y.-G. Jiang, "Wilddeepfake: A challenging real-world dataset for deepfake detection," *Proceedings of the 28th ACM International Conference on Multimedia*, 2020.
- [16] P. Kwon, J. You, G. Nam, S. Park, and G. Chae, "Kodf: A large-scale korean deepfake detection dataset," in *IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10744–10753.
- [17] 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," *CoRR*, vol. abs/1803.09179, 2018. [Online]. Available: <http://arxiv.org/abs/1803.09179>
- [18] 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*, pp. 1–1, 2020.
- [19] V. Blanz, K. Scherbaum, T. Vetter, and H.-P. Seidel, "Exchanging faces in images," in *Computer Graphics Forum*, vol. 23, no. 3. Wiley Online Library, 2004, pp. 669–676.
- [20] S. Mosaddegh, L. Simon, and F. Jurie, "Photorealistic face de-identification by aggregating donors' face components," in *12th Asian Conference on Computer Vision*. Springer, 2015, pp. 159–174.

- [21] "Faceswap," <https://github.com/MarekKowalski/FaceSwap/>, [Accessed: 04-October-2023].
- [22] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2387–2395.
- [23] J. Thies, M. Zollhöfer, and M. Nießner, "Deferred neural rendering: Image synthesis using neural textures," *ACM Transactions on Graphics*, vol. 38, no. 4, pp. 1–12, 2019.
- [24] P. Garrido, L. Valgaerts, Ö. Rehmsen, T. Thormahlen, P. Perez, and C. Theobalt, "Automatic face reenactment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2014, pp. 4217–4224.
- [25] DeepFaceLab, "github," <https://github.com/iperov/DeepFaceLab>, 2017.
- [26] "Deepfakes," <https://github.com/deepfakes/faceswap>, 2017, [Accessed: 04-October-2023].
- [27] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410.
- [28] Y. Wang, X. Chen, J. Zhu, W. Chu, Y. Tai, C. Wang, J. Li, Y. Wu, F. Huang, and R. Ji, "Hififace: 3d shape and semantic prior guided high fidelity face swapping," *arXiv preprint arXiv:2106.09965*, 2021.
- [29] A. Agarwal, R. Singh, M. Vatsa, and A. Noore, "Magnet: Detecting digital presentation attacks on face recognition," *Frontiers in Artificial Intelligence*, vol. 4, p. 643424, 2021.
- [30] K. Kim, Y. Kim, S. Cho, J. Seo, J. Nam, K. Lee, S. Kim, and K. Lee, "Diffface: Diffusion-based face swapping with facial guidance," *arXiv preprint arXiv:2212.13344*, 2022.
- [31] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: a compact facial video forgery detection network," in *IEEE International Workshop on Information Forensics and Security*, 2018, pp. 1–7.
- [32] H. Dang, F. Liu, J. Stehouwer, X. Liu, and A. K. Jain, "On the detection of digital face manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5781–5790.
- [33] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Capsule-forensics: Using capsule networks to detect forged images and videos," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2019, pp. 2307–2311.
- [34] Y. Li, M.-C. Chang, and S. Lyu, "In ictu oculi: Exposing ai created fake videos by detecting eye blinking," in *2018 IEEE International Workshop on Information Forensics and Security*, 2018, pp. 1–7.
- [35] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1251–1258.
- [36] Y. Li and S. Lyu, "Exposing deepfake videos by detecting face warping artifacts," *arXiv preprint arXiv:1811.00656*, 2018.
- [37] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo, "Face x-ray for more general face forgery detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5001–5010.
- [38] H. Liu, X. Li, W. Zhou, Y. Chen, Y. He, H. Xue, W. Zhang, and N. Yu, "Spatial-phase shallow learning: rethinking face forgery detection in frequency domain," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 772–781.
- [39] 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.
- [40] K. Sun, T. Yao, S. Chen, S. Ding, J. Li, and R. Ji, "Dual contrastive learning for general face forgery detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 2, 2022, pp. 2316–2324.
- [41] Y. Qian, G. Yin, L. Sheng, Z. Chen, and J. Shao, "Thinking in frequency: Face forgery detection by mining frequency-aware clues," in *European Conference on Computer Vision*. Springer, 2020, pp. 86–103.
- [42] J. Li, H. Xie, J. Li, Z. Wang, and Y. Zhang, "Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6458–6467.
- [43] Z. Wang, J. Bao, W. Zhou, W. Wang, and H. Li, "Altfreezing for more general video face forgery detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4129–4138.
- [44] K. Thakral, S. Mittal, M. Vatsa, and R. Singh, "Phygitalnet: Unified face presentation attack detection via one-class isolation learning," in *IEEE 17th International Conference on Automatic Face and Gesture Recognition*, 2023, pp. 1–6.
- [45] S. Chhabra, K. Thakral, S. Mittal, M. Vatsa, and R. Singh, "Low quality deepfake detection via unseen artifacts," *IEEE Transactions on Artificial Intelligence*, 2023.
- [46] B. M. Le and S. S. Woo, "Quality-agnostic deepfake detection with intra-model collaborative learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, October 2023, pp. 22 378–22 389.
- [47] D. Cozzolino and L. Verdoliva, "Noiseprint: A cnn-based camera model fingerprint," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 144–159, 2019.
- [48] N. Yu, L. S. Davis, and M. Fritz, "Attributing fake images to gans: Learning and analyzing gan fingerprints," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7556–7566.
- [49] U. A. Ciftci, I. Demir, and L. Yin, "How do the hearts of deep fakes beat? deep fake source detection via interpreting residuals with biological signals," in *IEEE International Joint Conference on Biometrics*. IEEE, 2020, pp. 1–10.
- [50] F. Marra, D. Gragnaniello, D. Cozzolino, and L. Verdoliva, "Detection of gan-generated fake images over social networks," in *IEEE Conference on Multimedia Information Processing and Retrieval*, 2018, pp. 384–389.
- [51] A. Jain, P. Majumdar, R. Singh, and M. Vatsa, "Detecting gans and retouching based digital alterations via dad-hcnn," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 672–673.
- [52] B. Zhang, J. P. Zhou, I. Shumailov, and N. Papernot, "On attribution of deepfakes," *arXiv preprint arXiv:2008.09194*, 2020.
- [53] T. Yang, Z. Huang, J. Cao, L. Li, and X. Li, "Deepfake network architecture attribution," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 4, 2022, pp. 4662–4670.
- [54] Z. Dias, A. Rocha, and S. Goldenstein, "First steps toward image phylogeny," in *IEEE International Workshop on Information Forensics and Security*, 2010, pp. 1–6.
- [55] Z. Dias, S. Goldenstein, and A. Rocha, "Large-scale image phylogeny: Tracing image ancestral relationships," *IEEE MultiMedia*, vol. 20, no. 3, pp. 58–70, 2013.
- [56] M. Nucci, M. Tagliasacchi, and S. Tubaro, "A phylogenetic analysis of near-duplicate audio tracks," in *IEEE 15th International Workshop on Multimedia Signal Processing*, 2013, pp. 099–104.
- [57] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on image processing*, vol. 21, no. 12, pp. 4695–4708, 2012.
- [58] J. Ricker, S. Damm, T. Holz, and A. Fischer, "Towards the detection of diffusion model deepfakes," *arXiv preprint arXiv:2210.14571*, 2022.
- [59] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I. S. Kweon, and S. Xie, "Convnext v2: Co-designing and scaling convnets with masked autoencoders," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 16 133–16 142.
- [60] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.
- [61] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *2018 IEEE spoken language technology workshop (SLT)*. IEEE, 2018, pp. 1021–1028.
- [62] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
- [63] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Use of a capsule network to detect fake images and videos," *arXiv preprint arXiv:1910.12467*, 2019.
- [64] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," *Advances in Neural Information Processing Systems*, vol. 30, 2017.
- [65] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [66] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly

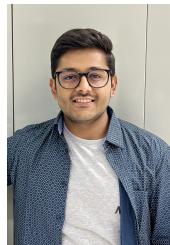
- et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
- [67] S. Mehta and M. Rastegari, "Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer," *arXiv preprint arXiv:2110.02178*, 2021.
- [68] Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie, Y. Wei, J. Ning, Y. Cao, Z. Zhang, L. Dong *et al.*, "Swin transformer v2: Scaling up capacity and resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 12 009–12 019.
- [69] J. Li, Y. Wang, C. Wang, Y. Tai, J. Qian, J. Yang, C. Wang, J. Li, and F. Huang, "Dsfd: dual shot face detector," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5060–5069.
- [70] Z. Gu, T. Yao, Y. Chen, S. Ding, and L. Ma, "Hierarchical contrastive inconsistency learning for deepfake video detection," in *European Conference on Computer Vision*. Springer, 2022, pp. 596–613.
- [71] Y. Zheng, J. Bao, D. Chen, M. Zeng, and F. Wen, "Exploring temporal coherence for more general video face forgery detection," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 15 044–15 054.
- [72] A. Haliassos, R. Mira, S. Petridis, and M. Pantic, "Leveraging real talking faces via self-supervision for robust forgery detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 950–14 962.
- [73] X. Dong, J. Bao, D. Chen, T. Zhang, W. Zhang, N. Yu, D. Chen, F. Wen, and B. Guo, "Protecting celebrities from deepfake with identity consistency transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9468–9478.
- [74] K. Shiohara and T. Yamasaki, "Detecting deepfakes with self-blended images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 720–18 729.
- [75] C. Shuai, J. Zhong, S. Wu, F. Lin, Z. Wang, Z. Ba, Z. Liu, L. Cavallaro, and K. Ren, "Locate and verify: A two-stream network for improved deepfake detection," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 7131–7142.
- [76] Z. Yan, Y. Luo, S. Lyu, Q. Liu, and B. Wu, "Transcending forgery specificity with latent space augmentation for generalizable deepfake detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 8984–8994.
- [77] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, "Cnn-generated images are surprisingly easy to spot... for now," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8695–8704.



**Kartik Thakral** received the Bachelor of Technology degree in Computer Science from the College of Engineering Roorkee, India, in 2019. He was awarded the Prime Minister Research Fellowship in October 2021. Currently, he is pursuing a Ph.D. degree from the Indian Institute of Technology Jodhpur. His research interests include computer vision, deep learning, and biometrics.



**Harsh Agarwal** received his Bachelor of Technology degree in Electrical Engineering from Indian Institute of Technology Jodhpur, India in 2023. He is currently pursuing a MSc in Computing with specialisation in AI/ML from Imperial College London, United Kingdom. His research interests include Computer Vision, Reinforcement Learning and Efficient Large Language models.



**Kartik Narayan** received his Bachelor of Technology degree in Computer Science from Indian Institute of Technology Jodhpur, India, in 2023. He is currently pursuing a Ph.D. degree in Computer Science from Johns Hopkins University, United States. His research interests include computer vision, deep learning, and face biometrics, with a focus on vision-language and generative models.



**Surbhi Mittal** received her B.Sc.(Hons) in Computer Science, and M.Sc. in Computer Science from the University of Delhi, India in 2017 and 2019, respectively. She is the recipient of the Senior Researcher Fellowship from UGC-NET, India and the IBM Ph.D. Fellowship. Currently, she is pursuing her Ph.D. degree from the Indian Institute of Technology Jodhpur. Her research interests include computer vision and deep learning with applications in biometrics and fairness of AI algorithms.



**Mayank Vatsa** obtained MS and PhD degrees in Computer Science from the West Virginia University, USA. Currently, he is a Professor at IIT Jodhpur, India. He is a Fellow of IEEE and IAPR, the recipient of the prestigious Swarnajayanti Fellowship award from Government of India, A. R. Krishnaswamy Faculty Research Fellowship at the IIT-Delhi, and several Best Paper and Best Poster Awards at international conferences. He has served as Area/Associate Editor of Pattern Recognition and Information Fusion, the General Co-Chair of IJCB2020, and PC Co-Chair of FG2021, AVSS2021, IJCB2014, ICB2013. From 2015 to 2018, he served as the Vice President (Publications) of the IEEE Biometrics Council where he led the efforts to start the IEEE TRANSACTIONS ON BIOMETRICS, BEHAVIOR, AND IDENTITY SCIENCE. He has also participated in several Indian government initiatives including UIDAI (Aadhaar), designing biometrics Standards for e-Gov applications, Responsible AI, DigiYatra, and formation of TIH of Computer Vision and ARVR at IIT Jodhpur.



**Richa Singh** is currently a Professor at IIT Jodhpur, India. She is a Fellow of IEEE and IAPR. She was a recipient of the Kusum and Mohandas Pai Faculty Research Fellowship at the IIT-Delhi, the FAST Award by the Department of Science and Technology, India, and several best paper and best poster awards in international conferences. She has also served as the Program Co-Chair of CVPR2022, ICMI2022, IJCB 2020, FG2019, and BTAS 2016, and a General Co-Chair of FG2021 and ISBA 2017. She was also the Vice President (Publications) of the IEEE Biometrics Council. She is an Associate Editor-in-Chief of Pattern Recognition, and Area/Associate Editor of several journals. She has participated in several initiatives including UIDAI (Aadhaar) and designing biometrics Standards for e-Gov applications.