International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

An Intelligent Ensemble Techniques for Enhancing Deepfake Video Detection Performance

Shraddha Suratkar, Ankit Jaiswal, Atharva Patil, Anupum Laddha, Aryan Khurana, Rahul Ingle, Rita Patil

Submitted:03/11/2024 **Revised**: 14/12/2024 **Accepted:** 25/12/2024

Abstract: In this research paper exploits the Ensembling approach for the detection of Deepfake videos. Fewer experiments employing Ensmebling techniques were conducted in the Deepfake domain. By training our model on many datasets and then ensembling their feature layers, we aimed to tackle the problem of over-fitting on a single dataset. Our study yielded superior outcomes compared to current techniques, leveraging the Face Forensics++ (FF++) dataset. This dataset comprises 1000 original video sequences subject to various manipulations. We have also employed effective pre-processing approaches to increase the accuracy of our present work. This includes Katna framework for key-frame extraction and MTCNN library for extracting facial features from the key-frames. MTCNN's accuracy in face detection is one of its key features. It features a cascaded architecture composed of three neural networks that enhance the face identification results gradually. This method aids in reducing false positives and improving the precision of face detection. After training these weak learners we will use them along with a CNN model for final prediction. So the final model will contain these weak learners and a Convolutional Neural Network (CNN) model.

Index Terms— *Deepfakes, Video Deepfake Detection, Convolutional Neural Networks(CNN), XceptionNet, EfficientNetb0, Ensemble.*

1.Introduction

Significant advancements in artificial neural network(ANN)-based innovations are critical in controlling multimedia material. FakeApp and FaceApp are two software applications powered by artificial intelligence that have gained notoriety for their ability to seamlessly swap faces in both images and videos, creating highly realistic results. Anyone may alter their outward look by changing their haircut, age, gender, and other distinguishing qualities. The proliferation of these fake movies has raised various concerns and has become well-known behind the scenes.

"Deepfake" originates from the merging of "Deep Learning" and "Fake," denoting the creation of highly realistic videos or images with the aid of deep learning technology. It gained prominence when an unidentified Reddit user utilized deep learning algorithms towards replacing faces in adult films with different ones, resulting in convincing counterfeit videos, circa late 2017.

Fake videos are produced through a process involving two distinct neural networks: a generative network and a discriminative network, which utilize a FaceSwap technique. The generative network employs an encoder and decoder to fabricate synthetic visuals, while the discriminative network assesses the authenticity of newly generated images. This approach was conceptualized by Ian Goodfellow, who introduced Generative Adversarial Networks (GANs) as the

integration of these two networks.

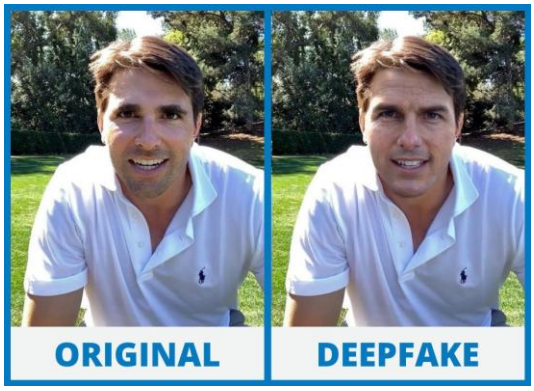

Fig. 1. Real image Vs. Fake image[1]

Deepfake technology gets its tag from DL, a form of AI. DL algorithms, capable of independently tackling complex problems using vast datasets, are harnessed in Deepfake AI to seamlessly swap faces in videos, images, and other digital content, creating convincing but falsified visuals.

Deepfake technology employs a dual-process involving a generator and a discriminator algorithm. The generator crafts synthetic content, prompting the discriminator to distinguish between real and artificial material. Feedback from the discriminator about the authenticity of the content informs the generator for refinement in subsequent iterations of the Deepfake creation process.

Deepfake technology poses numerous risks, ranging from the creation of revenge porn using synthetic faces of victims to producing convincing videos of public figures making false statements. Additionally, there's the potential for CEOs to manipulate stock markets with fabricated company performance videos, as well as online predators using fake identities in video chats.

2. Literature survey

A. Ensemble Methods

In paper [2], they have proposed an ensemble technique that is better than the SOTA for the used dataset by 0.5%. They have used a bagging method. In the proposed model they combine all the features of the weak learners(overfitted) into one convolution layer rather than combining outputs, thus increasing the output of the final model.

B. Deepfake Detection

This paper [3], involves pure ML models which areutilized to observer interpretability, which is a significant disadvantage of DL-based models. It also includes Ex- tracting features from existing datasets using techniques like Haralic Texture Feature (HTF), Histogram of Oriented Gradients (HOG), Color Histogram (CH), Hu Moments (HM) and building a feature set out of it.

In paper [4], they demonstrated a brand-new process that can find frequencies particular to GANs (generative adversarial networks), which serve as a distinctive finger- print for various generative designs.

C. Using CNN and RNN

In their study [5], the authors employed a CNN model coupled with Error Level Analysis (ELA) for image preprocessing. Their dataset consisted of 24,000 images, evenly divided between authentic and Deepfake images, for both training and testing purposes. Through their efforts, they attained an impressive accuracy rate of 99%.

In the work presented in reference [6], the authors introduced a methodology designed to identify Deepfake videos by integrating temporal awareness. Their approach involved employing a CNN to capture features at the frame level. Subsequently, these features were utilized to train a

recurrent neural network (RNN). They have used HOHA dataset (300 Videos) and 300 more videos found on the internet. Theygot a test accuracy of 97.1%.

The paper [7] starts with implications of Deepfake and briefly summarizes it. Then it talks about various attributes or features that can be used for deep fake detection like smoothness of skin, color of skin, eye blinking rate, face warping artifacts. We use DenseNet169 with face warping artifacts in this research paper.

In a this study [8], ResNet50 demonstrates superior performance compared to alternative models, supported by experimental results. Commonly utilized methods for deep fake detection include CNNs for extracting frame features, LSTM for analyzing temporal sequences, and RNNs for identifying temporal irregularities across frames.

In paper [9], the authors suggested a single Gabor function that could generate circular, elliptical, and linear Gabor filters. The suggested function may be used to pictures with a variety of forms. A back-propagation learning architecture was used to enable the suggestedCNN function's flexibility.

In their paper [10], the authors introduced a model comprising a CNN combined with a classifier network. They evaluated three distinct CNN architectures: ResNet50, XceptionNet, and InceptionV3, conducting a comparative analysis. After careful consideration, XceptionNet was selected and integrated with the proposed classifier for classification tasks. The FaceForensics++ dataset was employed for experimentation.

In this study [11], researchers explored the effectiveness of eight distinct machine learning techniques for discerning between tampered and untampered images. These methods comprised three traditional machine learning algorithms, namely Support Vector Machine, Random Forest, and Decision Tree, alongside five deep learning models: DenseNet121, DenseNet201, ResNet50, ResNet101, and VGG19.

In a recent study [12], researchers introduced an effective Deepfake detection technique known as HCiT. Unlike conventional approaches that directly utilize a pure-ViT model on image patch sequences, this method involves feeding feature maps into ViT. These feature maps are derived from fine-tuning Xception on the Deepfake dataset.

In a recent study [13], researchers employed advanced deepfake detection techniques, specifically leveraging models such as Xception and MobileNet, to discern between authentic and deepfake videos. Their findings revealed impressive accuracy rates ranging from 91% to 98% across different datasets, depending on the specific deepfake technology utilized. Additionally, the researchers introduced a novel approach by incorporating a voting mechanism that aggregates the results from four distinct methods, enhancing the overall detection capability beyond reliance on any single technique.

The study [14] presents an experimental application of a straightforward data augmentation technique known as Face-Cutout. This approach involves dynamically removing portions of an image based on facial landmark data, enabling the model to focus solely on relevant areas of the input.

D. Using Ensemble Methods

In their research [15], the authors introduced a technique for identifying deepfake facial images. Initially, they extracted diverse features such as gray gradient features, spectrum features, and texture features from both genuine and counterfeit face images. These features were amalgamated into an ensemble feature vector through a flattening process. Subsequently, this feature vector was inputted into a back-propagation neural network for training a classification model designed as the Deepfake detector. Their experimentation employed the CelebA dataset, achieving an accuracy rate of 97.04%.

In a recent paper [16], a novel detection approach has been introduced to address the growing issue of deep forgery. The method integrates the swift performance of the Xception model with the superior accuracy of the EfficientNetB4 model. Remarkably, the

results demonstrate real-fake accuracy rates ranging from 92.10% to 97.80% in the DFDC dataset, 98.27% to 87.47% in the CELEB-DF dataset, and 78.71% to 79.32% in the FaceForensics dataset.

This paper [17] uses an ensemble of XCeption models. Deep Fake methods involve elements that are invisible to the human eye. It discusses a Kaggle tournament and the several datasets that were utilised in it. By adding together the average of their forecasts, three Xception models wereintegrated.

In this paper [18], DeepfakeStack was employed to assess various cutting-edge deep learning models. DeepfakeStack operates by training a meta-learner atop pre-trained base-learners. It provides a framework to fit the meta-learner using the predictions from the base-learners, demonstrating the effectiveness of ensemble techniques in classification tasks.

This paper [19] talks about hierarchical ensembles of weakly supervised models. The model was improved by integrating a method for weakly supervised deep attention data augmentation into the feature map processing of the classifier model. They used DFDC and CelebDF dataset. The model achieved an accuracy of 92.20%.

In a particular study (reference [20]), researchers employed a method for extracting key video frames to minimize computational requirements in the detection of Deepfake videos. They utilized two datasets: FaceForensics++ and the Deepfake Detection Challenge datasets. Their results showed an accuracy of 98.5% when using the FaceForensics++ dataset alone and 92.33% accuracy when combining both the FaceForensics++ and Deepfake Detection Challenge datasets.

The study [21] introduces an attention mechanism aimed at producing interpretable model inferences, thereby enhancing the network's learning capacity. Additionally, the paper employs a triplet siamese training approach to extract intricate features from the data, resulting in improved classification accuracy.

E. Using other features

In a recent study, researchers employed a hybrid approach for detecting Deepfakes, combining temporal and deep learning models. Specifically, they utilized a blend of ResNext and LSTM architectures for the temporal-based model, while employing a triplet model architecture for the deep learning-based detection. Results indicated that the temporal model achieved the highest testing accuracy of 92.42%, while the triplet model attained an accuracy of 91.88%. Integrating these models into a final pipeline yielded an improved testing accuracy of 94.31%.

In paper [23], they have tried to use eye blinking to identify deep fake videos. CNNs have been applied for classifying eye states, while LSTM networks have been used for sequence learning. The eye aspect ratio was additionally used to measure the dimensions of open and closed eyes and to detect instances of blinking. Motivated by the continuous popularity of digital face alterations, particularly Deep-Fakes.

In study [24], a novel approach is presented, utilizing a dual-path pipeline integrating a Neural Ordinary Differential Equations (NODE)-based neural network alongside a transformer biased towards facial features. This setup is designed to handle visual content from various perspectives effectively. The transformer component facilitates the linking of landmarks over longer distances. Additionally, to enhance robustness, the system employs an attention-guided augmentation-based self-ensemble technique.

In paper [25], a novel approach was introduced for the detection of three distinct Deepfake methods: face swap, puppet-master, and attribute change. This method involved the utilization of three prevalent traces commonly associated with the Deepfake process: residual noise, warping artifacts, and blur effects. These identifiable markers were leveraged within their proposed network architecture

specifically designed for Deepfake detection.

In paper [26], the authors introduced a novel approach for examining notable alterations in eye blinking. This method, termed Deep-Vision, was devised to authenticate Deepfakes by leveraging machine learning alongside a combination of algorithms and a heuristic technique to detect these alterations.

In paper [27], presents a method that incorporates temporal information to automatically detect Deepfake videos. Utilizing a basic convolutional LSTM architecture, the approach can effectively discern whether a video has undergone manipulation with just 40 frames of video data.

In reference [28], researchers introduce a novel multi-branch detection network incorporating a dual attention mechanism, which encompasses both channel and spatial attention. This architecture enables comprehensive learning of contextual semantic information from both local and global artifacts. The efficacy of this approach is assessed using a deep fake dataset, with experimental outcomes indicating a notable achievement in test accuracy, reaching 96.45%.

In paper [29], the authors have demonstrated that XceptionNet which has been purposefully retrained out- performs this network. Alternative methods make use of LSTM analysis to capture the temporal progress of video frames. In this instance, combine a set of frame-based characteristics that wereoriginally extracted with a recurrent process.

In paper [30], an innovative aggregation module called DF-VLAD is introduced. This module revolutionizes the process of aggregating multiple frames by shifting the aggregation from the output layer to the feature layer. This adjustment not only enhances the flexibility of aggregation but also leverages the objective function of forgery detection to directly influence the learning of depth representation at the frame level.

In paper [31], researchers have employed a novel prior-attention mechanism. This mechanism utilizes pre-existing textural cues like edges and noise to construct attention maps directly. By leveraging these inherent attention maps, the model effectively amplifies discriminative features without the need for extra supervision. Additionally, the paper introduces a Feature Abstraction Block (FAB) to enhance feature representation.

The paper [32] presents two widely used Deepfake detection models utilizing Xception and EfficientNet architectures. It also incorporates five diverse databases, sourced from Google and Jigsaw, FaceForensics++, DeeperForensics, Celeb-DF, and an in-house large dataset DF-Mobio. The training process involves various augmentation methods, including a novel technique termed 'data farming.'

2.1 Literature gap

For Deepfake identification, a variety of algorithms havealready been developed using ML and DL. These methods have, however, been constrained by the length of time needed to train the model and the test accuracy. According to several studies, models tendto overfit a single dataset. Thus, the model's accuracy decreases when new data is introduced. There aren't many studies on the techniques for ensembling several pre-trained models and deep learning models to increase accuracy. The researchers continue to focus their studyand interest heavily on this area. To give a concurrentclassifier with superior performance, we are attempting to take advantage of each model's performance on a dataset and combine these models to create a composite ensemble model.

3. Proposed Methodology

This section outlines our suggested technique for detecting video face manipulation, or determining if a face in a video frame is real or false. The idea of ensembling forms the foundation of the suggested approach. It is a well-known fact that improved prediction performance can result from model ensembling. As a result, we concentrate on determining if and how various CNN-based

classifiers may be trained to capture various high-level semantic information that enhances the ensemble's performance for this particular challenge. To accomplish this, we begin by taking a look at the EfficientNet model, which is put forth as a cutting-edge method for CNN automated scaling. In comparison to other cutting-edge CNNs, this collection of architectures delivers higher accuracy and efficiency, and it turns out to be highly helpful in meeting the hardware and time limits established by DFDC. Moreover, two approaches, given an EfficientNet design, to make the model advantageous for the ensembling. On the one hand, we suggest integrating an attention mechanism, which gives the analyst a way to determine which segment of the examined video is most instructive for the categorization procedure. Conversely, we explore the integration of Siamese training methodologies into the learning process to extract more information from the input.

3.1 Dataset Description

Our project utilized the FaceForensics++ (FF++) dataset, which consists of 1000 original video sequences manipulated using four automated face manipulation methods: Face2Face, FaceSwap, Deepfakes, and Neural Textures. FaceSwap and Face2Face are graphics-based methods for facial manipulation, while Deepfakes and Neural Textures are learning-based approaches. The dataset aims to mitigate the threat of DeepFake videos by providing a large collection of manipulated videos. It comprises 977 YouTube images with mostly frontal faces and no occlusions, facilitating the creation of realistic forgeries using automated tampering methods. This dataset can be utilized for various tasks such as image and video classification, as well as segmentation.

Here, the Dataset are split and each of the manipulation techniques in the ratio of 80:20 for training andtesting. The train data is used to train the weak learnersand this is further splitted in the ratio of 80:20 for training and validation. This Training data is used to train the final ensemble model and the validation dataset is usedto check for overfitting.

The final statistics for each set are displayed in Table I.

TABLE I DATASET DISTRIBUTION FOR EACH MODEL

Model	Train	Validation	Test
Individual Weak Learner	1600	0	400
Final Model	5120	1280	1600

3.2 Ensemble methods

In order to develop a strong model for improving model performance, the ensemble model built utilizing the EfficientNet-B0 model is constructed by carefully integrating base models. In ensemble learning (EL), when several ML models are integrated to create a prediction, max-voting is a frequently employed strategy. When EfficientNet-B0, a prediction is made for a given input by each and every model in the ensemble; the forecast that receives the most votes is regarded as the final prediction. The key benefit of using max-voting in EL is that it can lower mistake rates and raise prediction accuracy overall. The ensemble may identify a greater variety of patterns and features in the data by mixing different models, which results in predictions that are more reliable and accurate. By properly integrating base models, the ensemble model constructed with the max-voting strategy is strong and can improve model performance. In EL, when several ML models are integrated to create a prediction, EfficientNet-B0 is a frequently employed strategy. When EfficientNet-B0, a prediction is made for a given input by each model in the ensemble; the forecast that receives the most votes is regarded as the final prediction. The key benefit of using max-voting in EL is that it can lower mistake rates and raise prediction accuracy overall. The ensemble may identify a greater variety of patterns and features in the data by mixing different models, which results in predictions that are more

reliable and accurate.

3.2.1 Networks

- **EfficientNet-B0**

Train several EfficientNet-B0 instances on various data subsets or with varying initializations. Then the model initialization or training data to guarantee that every EfficientNet-B0 instance learns a slightly different pattern. Add together each EfficientNet-B0 model's predictions. Averaging forecasts or employing more sophisticated strategies like stacking, boosting, or bagging are common approaches. Next, using regularization strategies to stop overfitting, as every model in the ensemble can have over fitted to a different feature of the data. To attain the best results, adjust the hyperparameters for each individual model and the ensemble. Assess the ensemble's performance using cross-validation or on a validation set.

- **Multi-task Cascaded Convolutional Networks**

A well-liked face detection framework that can identify faces and facial landmarks in a picture is called MTCNN. Bounding box regression, landmark localization, and face detection are MTCNN's three primary functions. To identify faces in an image, use the MTCNN model. Bounding boxes are usually provided by MTCNN around faces that are detected. Candidate face areas are suggested in the first stage of MTCNN, and further stages improve detection and get rid of false positives. For every recognized face, the MTCNN model can also predict facial landmarks like the mouth, nose, and eyes. This step is up to the individual use case and is optional. To extract the face regions from the source image, use the bounding boxes that were acquired during the face detection stage. Additional uses for these extracted faces include facial analysis and recognition.

- **Convolution neural network**

Assemble a labelled dataset made up of pictures with facial location annotations. To enhance model generalization, resize photographs to a consistent size, normalize pixel values, and expand the dataset. Add bounding boxes to the faces in the dataset. Every bounding box in an image should indicate where a face is located. Divide the dataset into sets for validation and training. Utilize the annotated dataset to teach the CNN model how to recognize faces. If a big dataset is available, fine-tune a pre-trained model using transfer learning, for example. Use non-maximum suppression to get rid of low-confidence or redundant detections. To maintain high-confidence face detections, set a threshold for the confidence scores. Extraction of the matching region from the original image is required for each bounding box that represents an identified face.

3.2.2 Preprocessing

Key Frame Extraction:

We have used key frame extraction as we only get theframes with the most relevance and it reduces the computation significantly as the number of framesis reduced. For each video, we have extracted the top 12 frames which are further sent for face extraction. Katna is a freely available tool designed to streamline the process of extracting key frames from videos. These frames serve as concise and precise summaries of the video's content. Developed in Python 3, Katna's video module has undergone thorough testing, proving its efficacy across popular video formats such as .mp4, .mov, and .avi. The kernel function is executed for processing the data. The number of inputs, input width and height are initialized in the kernel function. The function of convolution kernel is expressed by eqn. (1),

$$y(t) = a(t) * b(t) = \sum_{\tau=-\infty}^{+\infty} a(\tau) b(t-\tau) \quad (1)$$

Where a is the input data frame and b is the convolution kernel functions, τ is the time delay and t is the time, respectively.

Frame extraction and selection criteria:

- Detect frames with significant deviations from preceding ones by assessing absolute differences in LUV color space.
- Apply brightness scoring to filter extracted frames.
- Utilize entropy and contrast scores for further filtering of extracted frames.
- Conduct K-Means Clustering on frames using their image histograms.
- Choose the optimal frame from clusters based on Laplacian variance, aiding in detecting image blur.

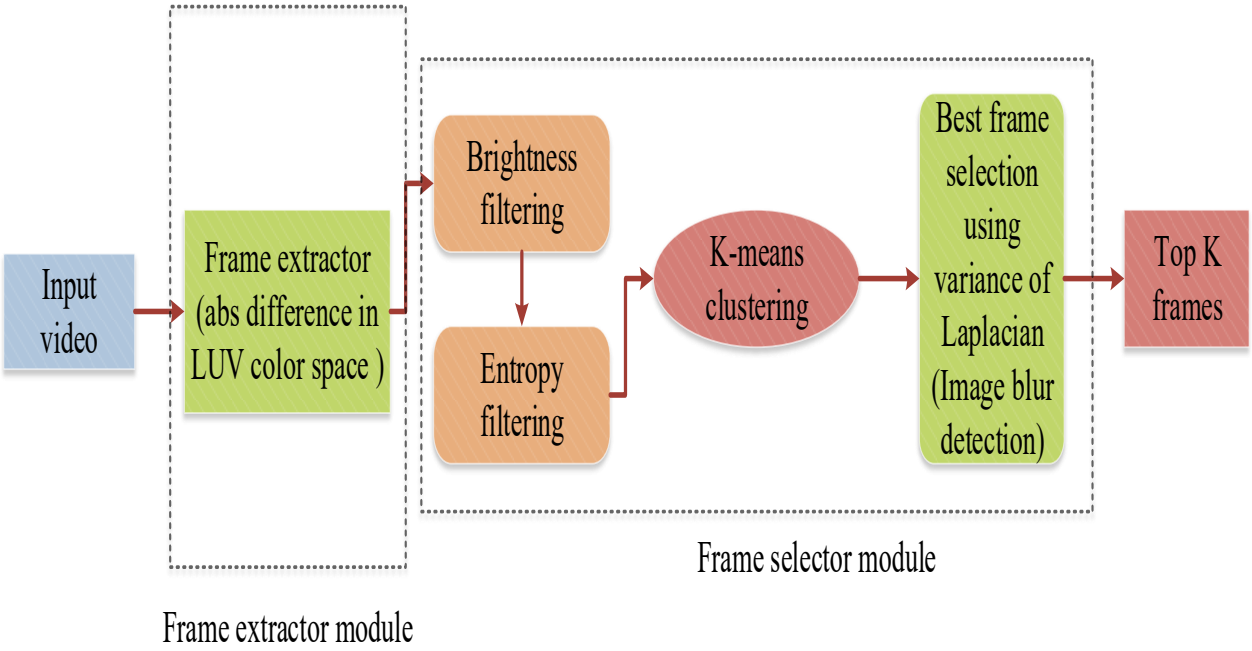

Fig. 2. High-level architecture of Katna[33]

Figure 2 represents the High-level architecture of the Katna Model. After passing the Deepfake video through the frame extraction pipeline, 12 key-framesare extracted.

Face Extraction (MTCNN):

MTCNN (Multi-task Cascaded Convolutional Networks) is a DL-based face detection and extraction system. It has various benefits over other processes, making it a popular choice in a wide range of applications. MTCNN's accuracy in face detection is one of itskey features. It features a cascaded architecture composed of three neural networks that enhance the face identification results gradually. This method aids in reducing false positives and improving the precision of face detection. MTCNN is built for real-time performance, making it ideal for image processing in real-time applications. It achieves quick inference times by using the parallelism inherent in convolutional neural networks (CNNs) and optimized network topologies. Another major feature of MTCNN is its resistance to changes in position and size. It can recognize and extract faces in a variety of orientations, tilts, and distances from the camera. This resilience is provided by the network's numerous phases, which handle various elements of the face identification, such as initial bounding box estimate, facial landmark localization, and final bounding box re- fining. MTCNN enables facial landmark localization in addition to face detection. It can calculate the coordinates of important facial features like the nose, eyes, and mouth. This data is beneficial for applications such as face alignment, emotion recognition, and face attribute analysis. MTCNN can recognize numerous faces in a picture at the sametime. It can handle numerous persons or faces in a scene and offers to bounding box coordinates and facial landmarks for each recognized face.

3.2.3 Feature Extraction from Weak Learners

In our work, we have used multiple weak learnersto train on individual datasets i.e. Deepfake generation methods. We've employed transfer learning to educate our less proficient learners. This not only reduces the time taken to train the model but also guarantees the efficiency of weak learners. To train the weak learners we are using EfficientNet-b0 model. The FF++ dataset is divided into 4 datasets for each weak learner. These 4 datasets contains each of the

Deepfake generation method (FaceSwap, Face2Face, Deepfake, and NeuralTextures). Each of the individual weak learners are trained for 100 epochs to overfit them on the training data. This is done so that the weak learner can easily identify if a new video is given to the model, which is generated using the same generation method. The videos are first preprocessed using preprocessing techniques such as key frame extraction and face extraction (MTCNN), and the preprocessed images are then passed through lossless transformations before beingpassed to the final model for training via the Data Loader library. After the convolution layer processes frame features, the data proceeds to the max-pooling layer for further feature extraction. Here, invariant features are extracted by partitioning the features into different segments. Moreover, the max operation is applied in all partitions of the system. Thus, the extracted features by the max-pooling phase is executed using eqn. (6), [55]

$$m_j^{y,k} = max_{v=1}\left(m_{(j-1)\times q+v}^{y-1,k}\right) \quad (6)$$

where q is the section range and v is the width window.

3.2.4 Classification using CNN model

After training these weak learners we will use them along with a CNN model for final prediction. So the final model will contain these weak learners and a CNN model. The video will first gothrough the preprocessing pipeline i.e. face extraction and key frame extraction. After that the feature extraction class will extract the features of the images from these weak learners and pass them to the CNN as presented in Fig 3. The CNN model will then predict the final output. In our optimization scheme, we utilize the Adam Optimizer along with a cross-entropy loss function. In the convolutional layer of our neural network, input images undergo convolution with multiple filters. Biases are then added, followed by the application of a nonlinear activation function to each layer. For example, the incoming image could be colored and multichannel. The following eqn. (1) describes the application of one convolution filter:

$$c(x',y') = \sum_{c_n}\sum_{\Delta x'}\sum_{\Delta y'} i(c_n, x'+\Delta x', y'+\Delta y')w'(c_n, \Delta x', \Delta y') \quad (1)$$

Where, c is denoted as convolution outcome, c_n is represented as channel number and input image is denoted as i .output image and filter matrix point is denoted as (x', y'), w' respectively. The matrix 'w' used for filtering can be viewed as multichannel since it contains distinct coefficients for each channel of the input image. Following this, a nonlinear activation function φ is applied to the output of the convolution operation and then added to it.

$$c'(x',y') = \phi[c(x',y') + a) \quad (2)$$

Here, First filter function is applied to the convolution layer which is demoted as c' , and is defined as a bias function. Given that neural networks typically comprise numerous filters, each contributing one channel, the output of the convolutional layer can be viewed as multi-channel in nature. Let's examine the computational complexity of a convolutional layer. Let us consider (n×m) is the pixel size of input image and (k×k(is the filter size of channels. Then, the selected classification network can minimize the error using the eqn. (3)

$$e(f) = e[f(x',y')] \quad (3)$$

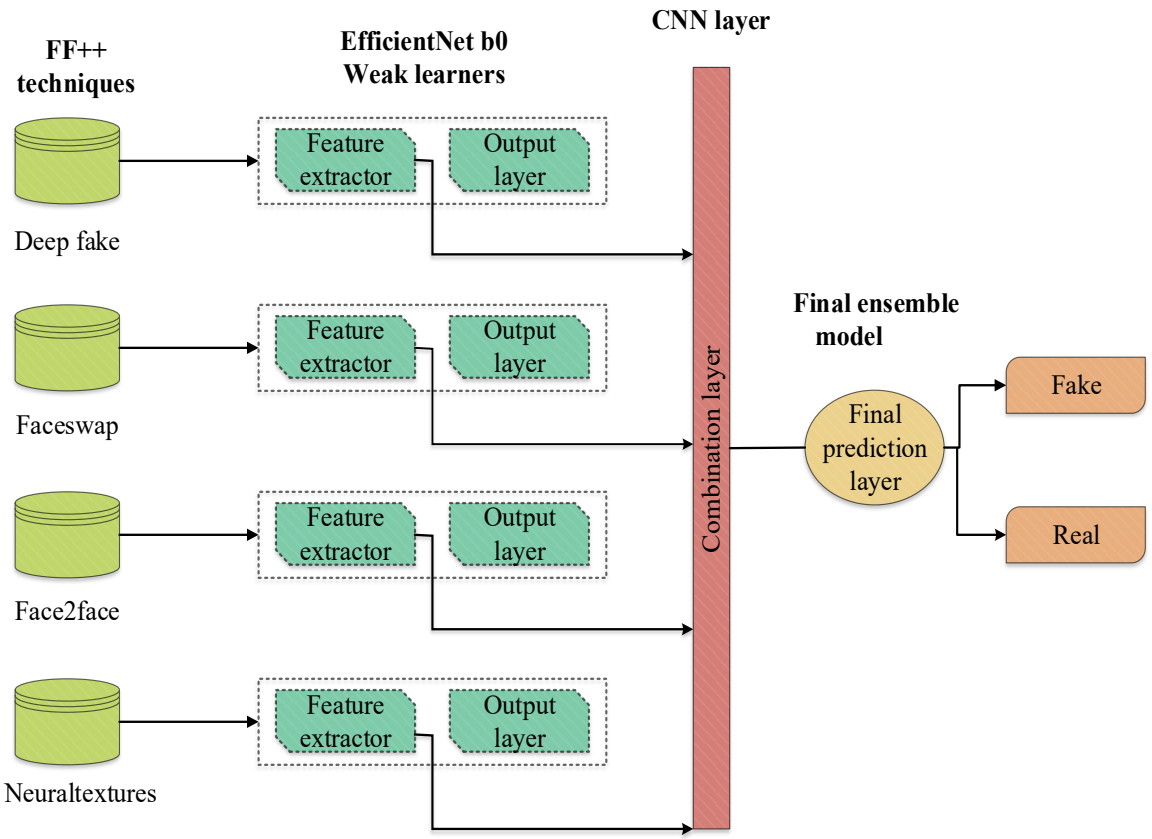

Fig. 3 Model Architecture

To prevent overfitting, we are also using a validation dataset. If the accuracy for the validation dataset reduces for the latest 5 epochs than the best one then it will stopthe training and it will return the model with the bestaccuracy over the validation dataset.

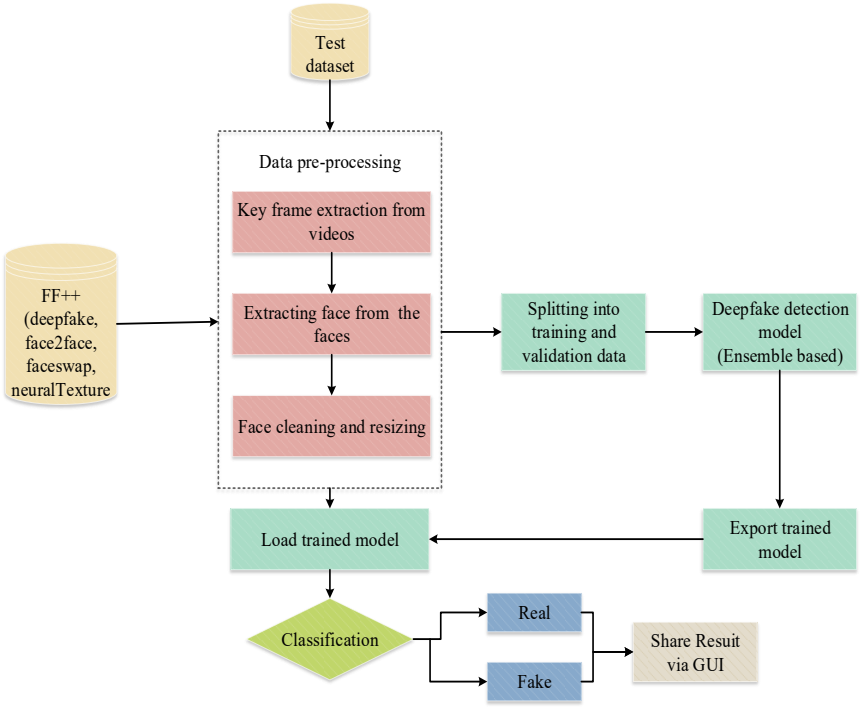

Fig. 4 System Architecture

Figure 4 represents our proposed System Architecture. As we can see, the input test video goes through the pre- processing pipeline, after which, the face extracted fromit will be loaded into our Ensemble CNN binary classifier to classify it as fake or real.

4 Result And Analysis

In this segment, we present findings from experiments employing Deep Learning models to detect video Deepfakes. Our evaluation metrics encompass validation accuracy and ROC score, serving as benchmarks for comparing model performance. Table II shows the results for Video Deepfake Detection after using the Deep Learning models. We used the Adam optimization algorithm as our optimizer and the loss function as the cross-entropy loss function. From the results obtained from Table II, We can conclude that as the Weak Learners were intentionally overfit- ted on their manipulation technique they gave low Accuracy on complete Dataset which consist of the all the four manipulation techniques. Whereas the Final ensemble model gave an Accuracy of 97.73%.

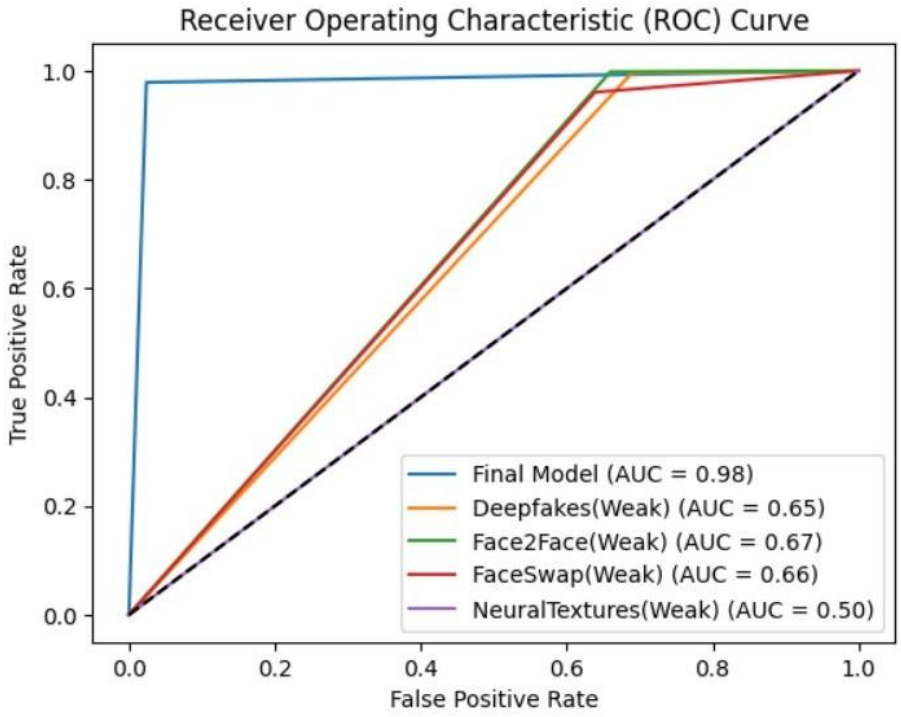

Fig. 5. ROC curve for all weak learners and the final model

The research implemented a method to normalize pixel values, originally ranging from 1 to 255, to a new range of 0 to 1. This normalization was done to facilitate the integration of external datasets into the model beyond the one initially used for training. Consequently, two distinct lists were created to categorize images based on their classification: correctly predicted DeepFakes and misclassified DeepFakes. To maintain the association between video pixels and their respective categories, a list was generated. Additionally, a for loop was employed to efficiently sort the groups into these four categories, as illustrated in figures 6, 7, and 8.

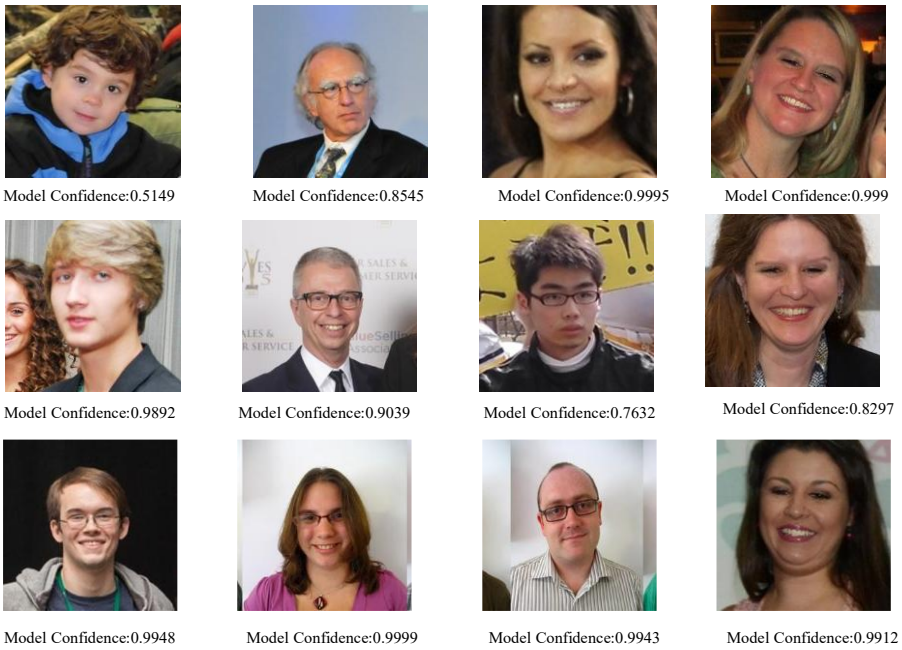

Fig.6 DeepFake prediction

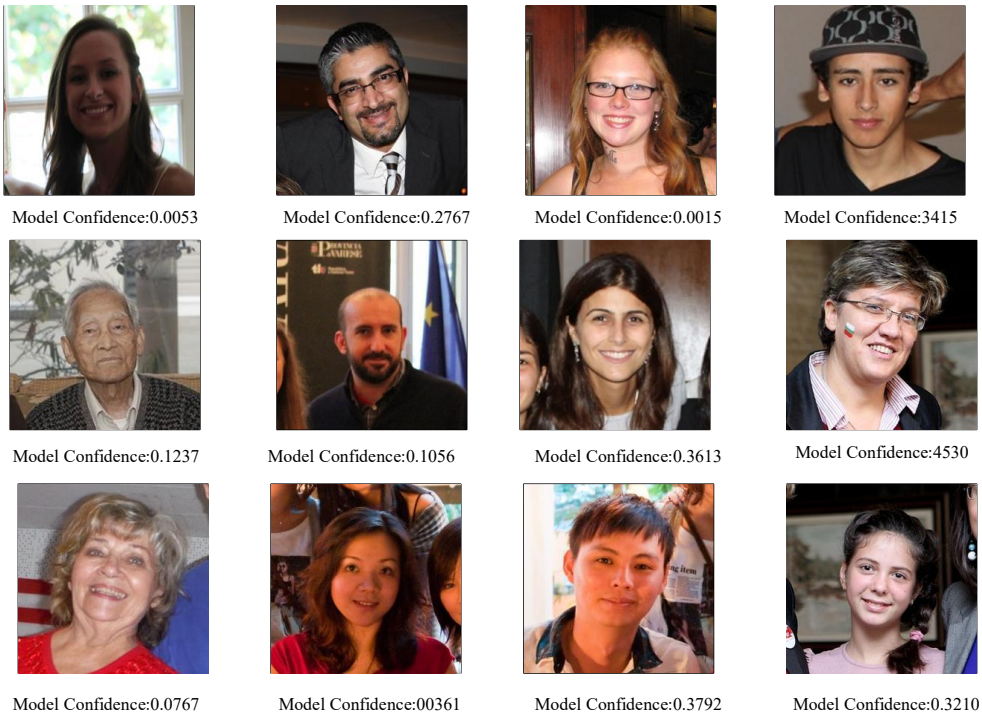

Fig.7 Correct DeepFake prediction

Fig.8 Misclassified DeepFake prediction

4.1 Model selection

In order to choose the optimal deep learning model for our classification problem, we assessed a number of well-known models in this study. We took into consideration the models Xception, EfficientNet-B0, XceptionResnetV3, and ResNet101, they are all typically applied to problems involving image classification. We assessed the methods' precision, effectiveness, and capacity for generalization. We discovered that the EfficientNet-B0 model outperformed the other models in every assessment metric after much testing and comparison. Originally, EfficientNet-B0 emerged as a leading performer in image classification tasks, notably on datasets like ImageNet. This method is a good fit for our goal since it achieves great accuracy at a comparatively low computational cost. Second, to maximize efficiency, CNN special architecture skillfully balances the scaling of depth, width, and resolution. Because of this, it works well with a variety of image sizes and resolutions, which makes it a suitable match for our varied collection of input photographs. When it came to our particular task, CNN consistently performed better than the other models, attaining higher accuracy and lower loss. Ultimately, we ran experiments with all of the models that were being considered. EfficientNet-B0 proved to be the most suitable option for our image classification challenge due to its exceptional performance, computational efficiency, and flexibility in handling a wide range of picture inputs.

4.1 Performance evaluation

The following Equations was used to compute a number of performance metrics, including accuracy, error rate, sensitivity, and specificity, which were used to assess the suggested Ensemble model.

$$A_c'(i) = \frac{T'(p'') + T'(n'')}{T'(p'') + T'(n'') + F'(p'') + F'(n'')} \tag{4}$$

$$P_r'(i) = \frac{T'(p'')}{T'(p'') + F'(p'')} \tag{5}$$

$$R_c'(i) = \frac{T'(n'')}{T'(p'') + F'(n'')} \tag{6}$$

$$S'_p(i) = \frac{T'(n'')}{T'(n'') + F'(p'')} \tag{7}$$

$$E'_r = \frac{F'(p'') + F'(n''))}{T'(p'') + T'(n'') + F'(p'') + F'(n'')} \tag{8}$$

$$F1(s) = \frac{2 \times [P'_r(i) \times R'_c(i)]}{P'_r(i) + R'_c(i)} \tag{9}$$

4.1.1 Area Under the Receiver Operating Characteristics (AUROC)

The AUROC curve, often abbreviated as AUCROC, serves as a metric to evaluate the performance of classification models across different threshold settings. The ROC curve itself is a graphical representation of the true positive rate (TPR) against the false positive rate (FPR), with TPR on the y-axis and FPR on the x-axis. Essentially, AUC quantifies the model's ability to distinguish between different classes. A higher AUC indicates a better ability of the model to correctly predict instances of both classes. In simpler terms, it measures how well the model can identify true positives and true negatives, with higher values indicating better performance in this regard.

Table.2 Overall outcomes

Models	Precision	Recall	F1 Score	Accuracy (%)	AU ROC
Deepfake	0.789	0.653	0.607	65.43	0.65
Face2Face	0.799	0.669	0.629	67.01	0.67
FaceSwap	0.751	0.66	0.627	66.18	0.66
NeuralTextures	0.82	0.717	0.693	71.8	0.50
Final	**0.9774**	**0.9773**	**0.9773**	**97.73**	**0.98**

From graph obtained in 5, We can conclude that the Weak Learners are getting a much lower Area under ROC scores of Deepfakes (AUC=0.65) , Face2Face (AUC=0.67), FaceSwap(AUC=0.66), NeuralTextures(AUC=0.50) whereas the Final Ensemble Model got a AUC of 0.98 this signifies that the High performance is achieved by the final model in differentiating between the positive and negative classes.

4.1.2 Intra test comparison

We present the findings of our tests, which we ran on the FF++ public dataset, in this section. Additionally, this result was contrasted with Celeb-DF, which evaluates the model's capacity to detect forged traces in Deepfake films by testing and training it on the identical dataset. The evaluation metric of choice is accuracy, and based on our findings, we provide an extensive visualization analysis. The findings shown in Table V clearly illustrate the major benefits of ensemble classifiers in Deepfake video detection when using our suggested approach. The EfficientNet technique is used to further enhance the model's performance, making it superior to all of its competitors.

Table.3 Comparison of dataset

Techniques	FF++ dataset	Celeb-DF dataset
Multi task [34]	77	54
Meso4 [35]	84	54
FWA [36]	80.11	56.2
Embedding [37]	96	67
3DCNN [38]	92	89
Capsule [39]	76	54
Proposed	97	92

5. Conclusion And Future Scope

With respect to the Deepfake Video Detection task, we researched and worked on enough research papers by conducting a Literature Survey. The survey resulted in a Literature gap in the current studies regarding Deepfake Detection Methods. We discovered a similar issue in all of the papers: most of the models employed in the implementation were becoming over-fitted to a certain dataset. As a result, we intended to apply the Ensemble approach to several datasets to acquire better results. We trained 4 EfficientNet-b0 weak learners on each of the Deepfake video generation techniques in FF++ Dataset that is Deepfake, face2face, FaceSwap, NeuralTextures and over fitted it. Finally we extracted the features from each of these trained-over fitted models and used it in a combinational CNN layer to combine the features from these weak learner models.

Thus, from our research and experimentation, we conclude that our model provides observable results by over-coming the result of overfitting using ensembling. We added a comparison table, listing the outcomes from our weak learner models and the Ensemble model. However, we could get even better results, if we can utilize all the Deepfake' publicly available datasets. For our research, we plan to use all of Deepfakes' publicly available datasets such as Deepfake Detection Challenge Dataset and the Celeb-DF Dataset and train weak learner models on them and include them in our final model. As a result, we anticipate covering the avail-able Deepfake datasets to increase the overall performance and scope of our architecture. Furthermore, we propose using unique pre-processing approaches to increase our model's input efficiency. This comprises various key-frame extraction tactics as well as face extraction treatments. We also plan to execute real- time Deepfake detection using our qualified model, which will be usable by a variety of apps and services. We also plan to optimize our model and implementation using hyperparameter tuning and other optimizations in order to reduce the prediction time. We may also combine multiple pre-trained models to train them on different datasets.

References

[1] "Video personalization using deep fake technology," Maverick, 2023.

[2] M. M. Antonio Bruno, Davide Moroni, "Efficient adaptive ensem- bling for image classification," ResearchGate, 2022.

[3] A. H. S. Md. Shohel Rana, Beddhu Murali, "Deepfake detection using machine learning algorithms," in *10th International Congress on Advanced Applied Informatics (IIAI-AAI*

2021), IEEE, 2021.

[4] F. L. Irene Amerini, Gianmarco Baldini, "Image and video forensics," ResearchGate, 2021.

[5] J. C. D. K. N. A. V. A. B. S. C. G. C. R. Kartheek, "Detecting Deepfakes using deep learning," in *2021 International Conference on Recent Trends on Electronics, Information, Communication Technology (RTEICT)*, IEEE, 2021.

[6] D. G. 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.

[7] A. A. L. Artem A Maksutov, Viacheslav O Morozov, "Methods of Deepfake detection based on machine learning," in *2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, IEEE, 2020.

[8] W. M. Wubet, "The Deepfake challenges and Deepfake video de- tection," in *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, IEEE, 2021.

[9] A. A. Ahmed Khalifa, Nawal Zaher, "Convolutional neural network based on diverse gabor filters for Deepfake recognition," IEEE, 2022.

[10] P. C. Alakananda Mitra, Saraju, "A novel machine learning based method for Deepfake video detection in social media," in *2020 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS)*, Research Gate, 2020.

[11] Y. W. Siddharth Solaiyappana, "Machine learning based medical image Deepfake detection: A comparative study," Elsevier, 2022.

[12] B. K. S. A. F. W. H. Z. A. A. Hadid, "Hcit: Deepfake video detection using a hybrid model of cnn features and vision transformer," in *2021 International Conference on Visual Communications and Image Processing (VCIP)*, IEEE, 2021.

[13] R. W. Deng Pan, Lixian Sun, "Deepfake detection through deep learning," in *2020 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT)*, IEEE, 2020.

[14] A. D. Sowmen Das, Selim Seferbekov, "Towards solving the deep- fake problem: An analysis on improving Deepfake detection using dynamic face augmentation," in *IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, 2021, pp. 3776-3785*, IEEE, 2021.

[15] J. Z. K. C. G. S. X. Lin, "A heterogeneous feature ensemble learning based Deepfake detection method," in *ICC 2022 - IEEE International Conference on Communications*, IEEE, 2022.

[16] M. K. Serhat AtaŞ, Ismail Ilhan, "An efficient Deepfake video de-

[17] tection approach with combination of efficientnet and xception models using deep learning," in *2022 26th International Conference on Information Technology (IT)*, IEEE, 2022.

[18] T. C. Joanna Baciak, Magdalena Zurawska, "Deepfake video detec- tion using the ensemble of neural networks," ResearchGate, 2020.

[19] A. H. S. Md. Shohel Rana, "Deepfakestack: A deep ensemble- based learning technique for Deepfake detection," in *2020 7th IEEE International Conference*, 2020.

[20] A. M. V. Samuel Henrique Silva, Mazal Bethany, "Deepfake forensics analysis: An explainable hierarchical ensemble of weakly supervised models," Elsevier, 2022.

[21] P. C. . E. K. Alakananda Mitra, Saraju P. Mohanty, "A machine learning based approach for Deepfake detection in social media through key video frame extraction," Springer, 2021.

[22] S. M. Nicolò Bonettini, Edoardo Daniele Cannas, "Video face manipulation detection through ensemble of cnns," in *2020 25th International Conference on Pattern Recognition (ICPR)*, 2020.

[23] J. John(B) and B. V. Sherif, "Multi-model Deepfake detection using deep and temporal features," in *ICIPCN 2022: Third*

International Conference on Image Processing and Capsule Networks, 2022.

[24] J. F. Ruben Tolosana, Ruben Vera-Rodriguez, "Deepfakes and be- yond: A survey of face manipulation and fake detection," in *Infor- mation Fusion, 2020*, arXiv, 2020.

[25] Z. S. Zhengbo Luo, Sei-ichiro Kamata, "Transformer and node- compressed dnn based dual-path system for manipulated face de- tection," in *2021 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2021.

[26] J. K. S.-K. J. S. L. D. J. J.-U. Hou, "Detection enhancement for various Deepfake types based on residual noise and manipulation traces," IEEE, 2022.

[27] K. K. TackHyun Jung, Sangwon Kim, "Deepvision: Deepfakes detec- tion using human eye blinking pattern," ResearchGate, 2020.

[28] A. K. A. P. A. N. M. G. S. Baudha, "Exploiting spatiotemporal inconsistencies to detect Deepfake videos in the wild," in *2022 10th International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-22)*, IEEE, 2022.

[29] D. D. H. C. G. C. H. Shi, "Multi branch Deepfake detection based on double attention mechanism," in *2021 International Conference on Electronic Information Engineering and Computer Science (EIECS)*, IEEE, 2021.

[30] J. F. Ruben Tolosana, Ruben Vera-Rodriguez, "Video face manipu- lation detection through ensemble of cnns," ResearchGate, 2021.

[31] Y. H. Z. L. M. Z. W. L. S. Li, "Df-vlad: Deepfake video detection based on feature aggregation," in *2021 11th International Conference on Information Technology in Medicine and Education (ITME)*, IEEE, 2021.

[32] J. D. Cai Yu, Peng Chen, "Focus by prior: Deepfake detection based on prior-attention," in *2022 IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, 2022.

[33] P. Korshunov and S. Marcel, "Improving generalization of Deepfake detection with data farming and few-shot learning," in *IEEE Trans- actions on Biometrics, Behavior, and Identity Science*, IEEE, 2022.

[34] Alok, "Video key frame extraction with katna," Medium, 2019.

[35] H. H. Nguyen, F. Fang, J. Yamagishi, and I. Echizen, "Multi-task learning for detecting and segmenting manipulated facial images and videos," in 2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS). IEEE, 2019, pp. 1–8.

[36] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: a compact facial video forgery detection network," in 2018 IEEE International Workshop on Information Forensics and Security (WIFS). IEEE, 2018, pp. 1–7

[37] J. Straub, "Using subject face brightness assessment to detect 'deep fakes'(conference presentation)," in Real-Time Image Processing and Deep Learning 2019, vol. 10996. SPIE, 2019, p. 109960H

[38] Y. Nirkin, L. Wolf, Y. Keller, and T. Hassner, "Deepfake detection based on discrepancies between faces and their context," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 10, pp. 6111–6121, 2021

[39] Y. Wang and A. Dantcheva, "A video is worth more than 1000 lies. comparing 3dcnn approaches for detecting deepfakes," in 2020 15Th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020). IEEE, 2020, pp. 515–519

[40] 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