IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 5, Issue 12, April 2025

Impact Factor: 7.67

Detection of Deepfake Images Using Deep Learning Models

Nikita Mohod, Amar Sable, Om Mozrikar, Dhanvi Killol, Ashish Ulhe, Vanshika Bule, and Pratham Batamwar

Department of Computer Science and Engineering

SIPNA College of Engineering and Technology, Amravati, Maharashtra.

Abstract: *The emergence of deep learning has led to remarkable progress in generating hyper-realistic synthetic images, widely known as deepfakes. These AI-generated visuals pose significant challenges for detection, as well as serious risks such as misinformation, public manipulation, and reputational harm. Identifying and mitigating the spread of such content is an ongoing concern in the field of digital media forensics. This study explores a deepfake detection approach based on a deep learning architecture — the Xception model. Leveraging the DeepFake Detection Challenge (DFDC) dataset, which includes 124,647 labeled images split evenly between authentic and manipulated content, the model is trained to classify deepfake imagery. The proposed system achieves a training accuracy of 98.5% and a validation accuracy of 94.10%, indicating that the Xception model is a promising tool for effectively detecting deepfake images in practical applications.*

Keywords: Deep Fake, XceptionNet, fake image, real image classification, neural network

I. INTRODUCTION

ARTIFICIAL intelligence (AI) has made big developments in areas like computer vision and image processing.Similarly, deep learning approaches have changed visual processing, one new technology called “deep fakes” has appeared, deep fake means that is pretending to be someone else which is typically use to spread a fake news, deep fake images can be described as, people create fake images by copying someone’s face,and movements and placing them in different places to misguide or to spread misinformation, deep fakes have been created for a variety of purposes, including entertainment, art, and education. However, they can also be used to spread false information, manipulate politics, or conduct fraud. While they provide creative opportunities, misuse can have adverse impacts. Deep Fakes are created using AI models that learn from real images, making it hard to tell what’s real and what’s fake. Because of this, there’s a need to find ways to detect deep fakes and stop their harmful use. Deep Fakes are becoming more frequently used to generate synthetic data about politicians, communities, actors, and media due to increased worldwide interdependence and trust on social media platforms. This leads to the spread of fraude news on social media.The wide use of imagesharing facilities such as Telegram, Instagram, Reddit, WhatsApp, and Wikipedia can make it difficult to tell the difference between correct and altered photos [1].

A. Drawbacks Of Deep Fake Image

Deepfakes created risk because they produce realistic looking but fake images which used to spread false information to misguide or to create misinformation/disinformation that may change public opinion and influence elections. They enable identity theft and fraud into believing they are communicating with reliable individuals. Deepfake can also be used to damage someone’s reputation by creating embarrassing circumstances which could also result in bullying and harassment. This technology also loses trust between digital media, making it difficult to differentiate between real and fake. In addition, producing and disseminating deepfake content without permission presents significant ethical and legal issues, perhaps infringing on privacy rights and resulting in legal repercussions. Deepfakes have become popular and tend to generate Deepfakes have become popular and

DOI: 10.48175/IJARSCT-25947

.jpg

Fig. 1: Your caption here

tend to generate headlines, harming privacy and reputation. In Figure 3, Indian actress Rashmika Mandanna was caught in a deep fake film that copied her facial expressions and look, making it hard to differentiate between the two. A deep fake image can be difficult to distinguish due to its accurate facial expressions and appearance.

II. LITERATURE REVIEW

This section presents the literature review, divided into two parts: one focusing on Convolutional Neural Networks (CNN) and the other on Generative Adversarial Networks (GAN).

A. Convolution Neural Network

Y. Li. et. al. introduce a unique approach that leverages the natural physiological cue of eye blinking as a key indicator for identifying deepfakes [2].Synthetic videos often fail to replicate natural blinking patterns, leading to infrequent or unnatural eye movements. To detect these anomalies, the study uses a combination of CNN, long-term recurrent convolutional networks (LRCNs), and recurrent neural networks (RNNs) to analyze blinking behaviors. With around 90% accuracy, this method highlights the effectiveness of proposed techniques. A. Khodabakhsh et. al. highlight the crucial role of both spatial and temporal features in identifying deepfakes [3]. By integrating hybrid models like CNN, RNN, and 3D CNN, this approach captures not only frameby-frame inconsistencies but also sequential anomalies across time. This comprehensive feature analysis improves the model’s ability to generalize across different deepfake techniques, making it more robust against various types of manipulation.

Ipek Ganiyusufoglu et. al. For deepfake detection, video-level detectors have not been explored as extensively as image-level detectors [4], which do not exploit temporal data. In this paper, we empirically show that existing approaches on image and sequence classifiers generalize poorly to new manipulation techniques.The author proposes spatio-temporal features, modeled by 3D CNNs, to extend the generalization capabilities to detect new sorts of deepfake videos. Amritpal Singh et. al. addresses the challenge of detecting high-quality forged videos generated using Deepfake technology [5], which poses significant risks to digital information reliability. Unlike prior efforts focused on still images, the authors leverage spatio-temporal features by analyzing sequences of video frames. Their proposed architecture combines lower-level features around regions of interest with inconsistencies across frames.

Alexandros Haliassos et al. developed LipForensics, a deepfake detection technique that is excellent at detecting invisible changes and fending off distortions like compression [6]. LipForensics addresses high-level semantic inconsistencies in mouth motions, which are prevalent in false videos, in contrast to techniques that rely on brittle low-level clues. It learns natural mouth motion patterns by using a spatiotemporal network that has been pretrained on lipreading. It is then fine-tuned using both actual and false data. Numerous tests demonstrate that it outperforms state-of-the-art techniques in terms of generalization and robustness. A deepfake detection model named DFT-MF is proposed by Mousa Tayseer Jafar et al. that focuses on examining and validating lip/mouth movements in videos [7].To identify modified content, the model separates and analyzes mouth features using a deep learning technique. Tests on datasets with both authentic and fraudulent videos show how well the model performs in categorization. According to the data, DFT-MF performs better than other approaches already in use in the field, demonstrating how well it can detect deepfake videos using lip movement analysis.

Edoardo Daniele et. al. proposed a paper that tackles the crucial challenge of identifying facial alteration in video sequences, focusing on sophisticated methods such as FaceSwap and deepfakes [8]. Even if these tools are helpful, they can be dangerous if used improperly, as they can facilitate fake news, cyberbullying, and fake revenge porn. CNN based on EfficientNetB4 is used, which incorporates attention layers and Siamese training. On two publicly available datasets containing more than 119,000 movies, this approach performs well, providing a dependable means of thwarting facial alteration and reducing its detrimental effects on society.

DOI: 10.48175/IJARSCT-25947

B. Geneartive Adveserial Network

R Durall et. al. [9] investigates the limitations of generative convolutional deep neural networks, such as popular GAN architectures, particularly focusing on their reliance on convolution-based up-sampling methods like up-convolution or transposed convolution. The authors propose incorporating a novel spectral regularization term into the training optimization process, which not only ensures spectral consistency and reduces highfrequency errors but also improves the training stability and output quality of the generative networks.

Xiaoyi Dong et al. proposed a novel method for identifying faked photos and films [10]. To ascertain whether the target identity matches, their approach compares a suspect image or video to a reference image or video of the target identity. The authors present the Vox-DeepFake dataset, which connects questionable information to several reference photos, to bolster this strategy. Their Outer Face algorithm outperforms current approaches in terms of accuracy and generalization over a wide range of manipulation techniques. Hao Dang et al. address the growing need for detecting manipulated facial images in digital media forensics [11]. The authors introduce an attention mechanism to enhance feature maps for classification. This method highlights informative regions, improving binary classification accuracy (genuine vs. fake face) and enabling better localization of manipulated areas. Their large-scale dataset and analysis demonstrate the effectiveness of attention mechanisms in face forgery detection and localization.

To identify altered videos, Md. Shohel Rana et. al. suggest a deep ensemble learning method called DeepfakeStack [12]. This technique improves the composite classifier by combining many cutting-edge deep learning classification models. DeepfakeStack surpasses other classifiers by utilizing an ensemble approach, attaining an astounding 99.65% accuracy rate and an AUROC score of 1.0. This method's efficacy shows that it can serve as a strong basis for creating real-time multimedia content detection systems, providing a reliable way to deal with the problems caused by more complex fraudulent content in digital media. Iacopo Masi et al. used a two-branch network structure to identify altered faces in video streams [13]. This method suppresses highlevel face content while enhancing artifacts to isolate modified faces. While the other branch uses a Laplacian of Gaussian (LoG) as a bottleneck layer to increase multi-band frequencies, the first branch processes the original data. In order to differentiate unrealistic samples in the feature space and compress the variability of natural faces, a novel cost function is implemented. Zehao Chen et al. address the difficulty of detecting modified faces by offering a unique method based on multilevel facial semantic segmentation and a cascade attention mechanism [14].They demonstrate the usefulness of segmenting images into semantic chunks, as flaws and distortions are closely associated with these regions. Tests conducted on four datasets show how well their strategy performs and how well it generalizes when compared to other approaches.

Using GAN-based visual watermarking, Aakash Varma et al. suggest a proactive defense against facial alterations [15]. By adding a reconstructive regularization to the GAN's loss function, their method embeds a unique watermark into created images, in contrast to conventional passive detection techniques that only detect fakes after they have been generated. The efficiency of the watermarking technique is confirmed by experiments conducted on various datasets.

III. METHODOLOGY

A. Dataset

We have chosen to utilize the DeepFake Detection Challenge (DFDC) dataset, which has data on over 20.7k real images and 73.2k fake images. These datasets frequently contain a variety of frames, to identify fake images. The DeepFake Detection Challenge dataset has great value for research in both deepfake removal and general artificial face detection. The frames offer valuable information over time to identify problems that are common in deepfakes, within fake image dataset also contain other information which differentiates between fake and real images, such as facial landmarks and pixel alterations.

IV. PROPOSED METHODOLOGY

This section describes the CNN architecture proposed in this work. As illustrated in Fig. 2, convolutional layers are used to identify critical features in an image, such as edges or textures, while pooling layers help reduce the dimensions of these feature maps without discarding essential information.

Each input image is resized and normalized before being fed into the model using this equation :

$$X \in R^{h\times w\times c}$$

where h = 299, w = 299, and c = 3 (for RGB images), and X represents the preprocessed input image.

As the feature maps are processed, they are flattened into a one-dimensional array and passed to the fully connected layer, which combines the extracted features to form predictions. Finally, the input is categorized into appropriate classes by the output layer.

Deepfake image detection divided into five phases:

1) Collection of the DFDC dataset.
2) Resizing and preprocessing the input images.
3) Training the Xception model to extract features from real and fake images.
4) Evaluating the trained model using various evaluation metrics on a separate validation dataset.
5) Predicting whether a given image is real or fake.

The proposed convolutional neural network is based entirely on depthwise separable convolution layers, which reduce computational complexity compared to standard convolution. The separable convolution operation is defined as:

SeparableConv(X) = PointwiseConv(DepthwiseConv(X))

In depthwise convolution, each filter is applied to only one input channel:

$$Y_d^{(k)} = X^{(k)} * F^{(k)}, \quad \forall k \in [1, C]$$

In pointwise convolution (1x1 conv), outputs are mixed across channels:

$$Y = \sum_{k=1}^{C} Y_d^{(k)} * P^{(k)}$$

The Xception model consists of three main flows: Entry flow, Middle flow, and Exit flow. In the entry flow, basic image patterns such as edges and colors are captured. The middle flow, which is repeated eight times, extracts more complex patterns, textures, and abstract features. Each layer includes a residual connection:

$$H(X) = F\ (X) + X$$

The exit flow consolidates high-level feature representations for classification. Feature extraction is performed by 36 convolutional layers structured into 14 modules, with residual connections around all modules except the first and last.

After the feature extraction phase, additional layers are added. A Global Average Pooling (GAP) layer is used to reduce the dimensionality and avoid overfitting by replacing the fully connected layer:

$$G_i = \frac{1}{H \times W} \sum_{h=1}^{H} \sum_{w=1}^{W} X_{i,h,w}$$

During training, dropout regularization is applied:

$$\hat{G}_i = G_i \cdot r_i \qquad r_i \sim \mathrm{Bernoulli}(p)$$

The dense layer with sigmoid activation outputs a probability for the image being fake:

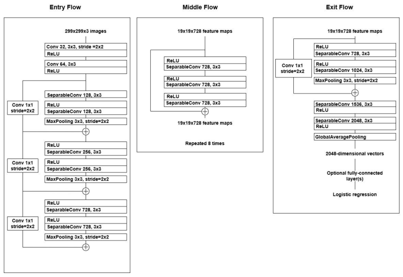
$$z = \sum_{i=1}^{C} w_i \hat{G}_i + b$$

$$y = \sigma(z) = \frac{1}{1 + e^{-z}}$$

The final classification decision is made based on:

$$\text{Label} = \begin{matrix} 1 \ \text{ if } \hat{y} \geq 0.5 \ \ (\text{Fake}) \\ 0 \ \text{ if } \hat{y} < 0.5 \ \ (\text{Real}) \end{matrix}$$

These components together enhance the model's ability to generalize and accurately classify real and fake im



A. Deep Fake Detection System (DFDS) Algorithm

Step1: DFDS DP (Data Preparation): This step involves collection of data.

Step2: DFDS DAP (Data Augmentation and Prepro- cessing): This process involves image data for training. Training images are augmented with transformations like rescaling.

Step3: DFDS MB (Model Building): In this part, load the Xception model (pre-trained on ImageNet) as the base model. Add custom classification layers on top of the base model (GlobalAveragePooling2D, Dropout, Dense). Compile the model with an Adam optimizer, binary cross-entropy loss, and accuracy metric.

Step4: DFDS IT (Initial Training): This step involves callbacks like ReduceLROnPlateau (to adjust learning rate) and EarlyStopping (to prevent overfitting).

```
Algorithm 1 Deep Fake Detection System (DFDS)
Require: Input Image
Ensure: Detection Result (Real or Fake)
1: DFDS DP ← Collect training dataset of real and fake images
2: DFDS DA ← Apply preprocessing (e.g., rescaling and resizing)
3: DFDS MB ← Load Xception model as the base model
4: DFDS IT ← Train the model on the dataset for a defined number of epochs
5: DFDS FT ← Fine-tune the model with additional training for better performance
6: DFDS ME ← Evaluate the model to obtain validation loss and accuracy
7: DFDS P ← Perform predictions on input image and display result (Real or Fake)
```

Step5: DFDS FT (Fine-tuning): In this step, unfreeze the base model layers. Compile the model again with a lower learning rate and further continue training for more epochs.

Step6: DFDS ME (Model Evaluation): Evaluate the model performance on the validation set using model.evaluate. Get validation loss and accuracy.

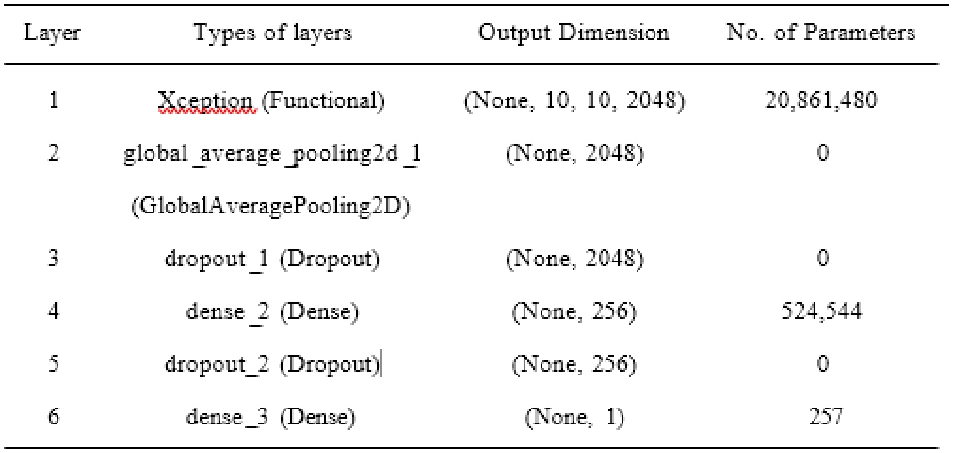
Step7: DFDS P (Prediction): In this part, load the saved model and preprocess an input image. Make predictions using the model and display the results.

V. RESULT AND DISCUSION

In this section, we discuss the effectiveness of the suggested model and the outcomes obtained. Each layer of the model helps in efficient training as shown in Table I. Xception model output dimensions and number of parameters for each layer

DOI: 10.48175/IJARSCT-25947

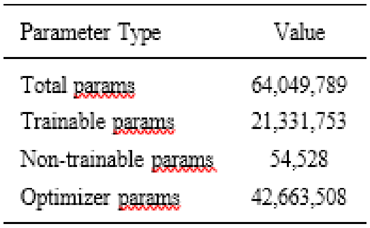
TABLE I: Layer configuration and parameter details of the Xception model



Layer	Types of layers	Output Dimension	No. of Parameters
1	Xception (Functional)	(None, 10, 10, 2048)	20,861,480
2	global_average_pooling2d_1 (GlobalAveragePooling2D)	(None, 2048)	0
3	dropout_1 (Dropout)	(None, 2048)	0
4	dense_2 (Dense)	(None, 256)	524,544
5	dropout_2 (Dropout)	(None, 256)	0
6	dense_3 (Dense)	(None, 1)	257

A 1x1 Xception(Functional) layer contains 32 filters to initiate the model. The activation function for this layer is ReLU(Rectified Linear Unit) and receives a 299x299 RGB images as input. The second layer is Global Average Pooling layer, it reduce the spatial dimension of the feature map(i.e height and width). It convert the 2D feature map to a 1D feature vector by arrange value across each feature map. This layer use for feature extraction. The third one is Dropout layer. In Xception, the dropout layer used after the fully connected (Dence) layer in the classification. During training, it randomly drops(set of zero) a fraction of the input neurons to prevent co-adaptation of feature. The dropout rate define the fraction of neurons dropped, Usually it specified as a value between 0 and 1. For ex:rate = 0.5 means 50%. The fourth layer is Dense layer, this layer maps the extracted features from the previous layer(global average pooling layer) to the probability distribution over the target class. This layer receives a flattened vector of features from the previous layers, each neuron in this layer has a set of weights and bias term.

TABLE II: Summary of total, trainable, and nontrainable model parameters



Parameter Type	Value
Total params	64,049,789
Trainable params	21,331,753
Non-trainable params	54,528
Optimizer params	42,663,508

This model takes advantage of CNN to extract hierarchical and distinct features from images, which are then utilized for the binary classification task. By including multiple convolutional, pooling and dense layers, the model is able to get complex patterns in the data. The dense layers at the end of the model carry out the final classification by manipulate these learned features. The summary of trainable, and non-trainable model parameters is shown in Table II.

A. Evaluation Metrics

In this model we use the accuracy score to measure the model performance. Accuracy is on of the primary evaluation metric in machine learning,the accuracy is find out by following formula,

Accuracy = Number of correct predictions/Total number of predictions

In a binary categorization problem, this can also be written as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Where: TP= True Positive, TN= true Negative, FP= False Positive, FN=False Negative

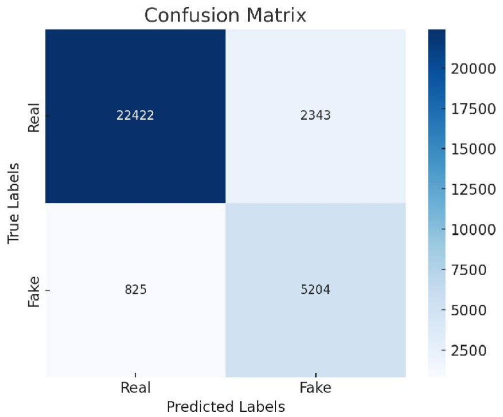


Fig. 3: Evaluation Metrics

B. Analysis of Proposed Model

Accuracy is a Straightforward metric that provides a general measures of model performance across all classes [1]. The accuracy gives us a quick understanding of how well our model is able to correctly recognize the fake and real image. For Xception model, we have used binary cross entropy and Adam optimizer, to increase the model's learning rate. We spanned over 60 epoch and validation batch size of 32 then we were able to achieve 98.5% training accuracy and 94.10% validation accuracy as shown in figure 4. the model loss over time is shown in figure 5.

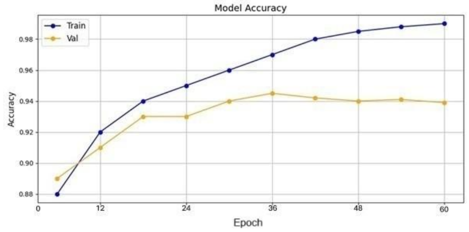


Fig. 4: The performance of model in terms of Accuracy

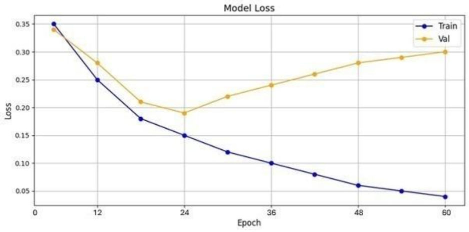


Fig. 5: The performance of model in terms of Loss

TABLE III: Performance Metrics of the Xception Model

Model	Accuracy	Precision	Recall	F1 Score
Xception	89.71%	68.95%	86.32%	76.66%

VI. MODEL PERFORMANCE ANALYSIS

Table III presents the performance metrics of the Xception model used for detecting deepfake images. The model's effectiveness is evaluated using four key classification metrics:

- Accuracy (89.71%) indicates the overall proportion of correctly classified instances among all predictions. It reflects the model's general performance across both classes (real and fake images).

- Precision (68.95%) measures the proportion of correctly predicted positive instances (deepfakes) out of all instances predicted as positive. A moderate precision suggests that the model occasionally misclassifies real images as deepfakes (i.e., false positives).
- Recall (86.32%) reflects the proportion of actual deepfakes that were correctly identified by the model. A high recall indicates the model is effective at capturing most deepfake instances, which is critical in security-related applications.
- F1 Score (76.66%) provides a balance between precision and recall. It is particularly useful when the dataset is imbalanced or when both false positives and false negatives carry significant costs. The F1 score demonstrates that the model maintains a good trade-off between detecting deepfakes and minimizing false alarms.

Overall, the Xception model shows strong performance in terms of accuracy and recall, making it a promising approach for deepfake image detection. However, its lower precision suggests there is room for improvement in reducing false positives.

VII. CONCLUSION

Detecting deepfake content is a complex challenge due to its high level of realism and subtle visual indications. Developments in AI techniques make it difficult to differentiate between deep fakes and real ones. In conclusion, this work demonstrates that the Xception model, an efficient CNN architecture, has become effective in detecting deep fake images, with a valuable 90% detection accuracy in the DFDC data set. The model's robustness has been further improved by using the techniques of data enhancement and dropout regularization, underscoring the significance of sophisticated deep learning methods and sizable, varied datasets in the fight against media manipulation. This research opens the door for further advancements in deepfake identification and presents an effective approach to mitigate the dangers caused by the increasing amount of fake information or to reduce the harm to someone's reputation. This work covers the way for continued innovation in deepfake detection, offering a promising solution to mitigate the risks posed by the spread of fake content.

REFERENCES

[1] D. Samal, P. Agrawal, and V. Madaan, "Deepfake image detection & classification using conv2d neural networks," 2024.

[2] 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 (WIFS), pp. 1–7, Ieee, 2018.

[3] A. Khodabakhsh and C. Busch, "A generalizable deepfake detector based on neural conditional distribution modelling," in 2020 international conference of the biometrics special interest group (BIOSIG), pp. 1–5, IEEE, 2020.

[4] I. Ganiyusufoglu, L. M. Ngoˆ, N. Savov, S. Karaoglu, and T. Gevers, "Spatio-temporal features for generalized detection of deepfake videos," arXiv preprint arXiv:2010.11844, 2020.

[5] A. Singh, A. S. Saimbhi, N. Singh, and M. Mittal, "Deepfake video detection: a time-distributed approach," SN Computer Science, vol. 1, no. 4, p. 212, 2020.

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

[7] M. T. Jafar, M. Ababneh, M. Al-Zoube, and A. Elhassan, "Forensics and analysis of deepfake videos," in 2020 11th international conference on information and communication systems (ICICS), pp. 053–058, IEEE, 2020.

[8] N. Bonettini, E. D. Cannas, S. Mandelli, L. Bondi, P. Bestagini, and S. Tubaro, "Video face manipulation detection through ensemble of cnns," in 2020 25th international conference on pattern recognition (ICPR), pp. 5012–5019, IEEE, 2021.

[9] R. Durall, M. Keuper, and J. Keuper, "Watch your upconvolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 7890–7899, 2020.

DOI: 10.48175/IJARSCT-25947

[10] X. Dong, J. Bao, D. Chen, W. Zhang, N. Yu, D. Chen, F. Wen, and B. Guo, "Identity-driven deepfake detection," arXiv preprint arXiv:2012.03930, 2020.

[11] 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, pp. 5781–5790, 2020.

[12] M. S. Rana and A. H. Sung, "Deepfakestack: A deep ensemblebased learning technique for deepfake detection," in 2020 7th IEEE international conference on cyber security and cloud computing (CSCloud)/2020 6th IEEE international conference on edge computing and scalable cloud (EdgeCom), pp. 70–75, IEEE, 2020.

[13] I. Masi, A. Killekar, R. M. Mascarenhas, S. P. Gurudatt, and W. AbdAlmageed, "Two-branch recurrent network for isolating deepfakes in videos," in Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16, pp. 667–684, Springer, 2020.

[14] Z. Chen and H. Yang, "Attentive semantic exploring for manipulated face detection," in ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1985–1989, IEEE, 2021.

[15] A. V. Nadimpalli and A. Rattani, "Proactive deepfake detection using gan-based visible watermarking," ACM Transactions on Multimedia Computing, Communications and Applications, vol. 20, no. 11, pp. 1–27, 2024.

[16] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258, 2017.

DOI: 10.48175/IJARSCT-25947