A Facial Authentication-based Deepfake Detection Machine Learning Model

Joseph Mwanza[a], Aaron Zimba[b] and Muwanei Sinyinda[c]

a. Department of Computer Science and Information Technology, Mulungushi University Kabwe, Zambia
josephmwanza69@gmail.com

b. Department of Computer Science, ZCAS University, Lusaka Zambia, email: aaron.zimba@zcasu.edu.zm

c. Department of Computer Science and Information Technology, Mulungushi University Kabwe, Zambia
msiniyinda@mu.edu.zm

Abstract— **In an era dominated by digital media, the escalating menace of media distortion, particularly propelled by the advancement of deepfake technology, has emerged as a critical concern spanning the realms of virtual landscapes and reality. The rise of deepfake technology has posed significant challenges to the authenticity of visual content in today's digital world. This study proposes a novel approach to deepfake detection using pixel analysis. By closely examining the pixel characteristics and patterns within manipulated images and videos, we developed an algorithm that can distinguish between real and fake content with high accuracy. Our algorithm combines two state-of-the-art deep learning models, Resnext and Long-Short Term Memory (LSTM), in a supervised machine learning framework. To enhance the performance of our algorithm, we applied standard pixel normalization during the preprocessing phase. Our proposed method achieved an impressive accuracy score of 95.6% on a public dataset of deepfake images and videos. This result demonstrates the efficacy of pixel analysis in detecting deepfakes. This research contributes significantly to countering the increasing threat of deepfake media manipulation, safeguarding the authenticity of visual content in today's digital world.**

Keywords— Deepfake, Authentication, Video, LSTM, Machine Learning

INTRODUCTION

Deepfake is a modern video editing technique driven by AI. It blends, replaces, and overlays images and videos to create believable fake videos [1]. Deepfakes have become popular online recently. They can change an actor's face in a video with another actor's face, given enough pictures of both. These videos are known as 'Deepfakes.' They got noticed when they were used in inappropriate ways, inserting faces of known people into adult videos on sites like Reddit. [2].

Deepfake is a growing type of online scam, becoming more common as we rely more on technology due to the recent pandemic [3]. There are three types: Photo, Audio, and Video deepfakes. Photo deepfake alters faces and bodies by swapping or merging them with others, making the person in the photo look completely different. Audio deepfake has two kinds: voice swapping, which changes the speaker's voice, and text-to-speech, converting text to various accents and voices. Face-Swapping switches the face in a video with someone else's [4].

Face detection is a computer technology that detects human faces in digital images. Face recognition goes further, identifying people and counting faces [5]. Face Authentication goes even further, confirming if the person is who they say they are in a picture. Unlike recognition, it needs prior info, like passwords, to confirm them [6]. Biometric systems using unique traits to confirm identity are popular. Deepfakes use AI and lots of data to copy faces, voices, and actions [7]. They learn from videos with two people. Basically, deepfakes use AI and face mapping to switch faces in a video [8].

Deepfakes use neural networks to imitate a person's face, voice, and actions by learning from lots of data [7]. A computer learns to switch faces by training on a video with two people. Basically, deepfakes use AI and facial mapping to replace someone's face in a video with another's [8].

Deepfake algorithms fall into two types: face swapping and face reenactment, based on their goals. They use Generative Adversarial Networks (GANs), where two neural networks work together to make realistic content. These networks, the discriminator and generator, learn from the same dataset of images, videos, or sounds. GANs can take many images of someone and create a new picture that's similar but not the same. Soon, GANs might change heads, bodies, and voices with less data. Even though deepfakes usually need many images, researchers found a way to create a fake video using just one shot, like a selfie [9].

The rise of deepfake technology threatens the authenticity of digital visuals content. Detecting deepfakes is challenging because they're increasingly sophisticated [10]. Deepfake creators are using advanced techniques, making detection harder. Deepfakes can be created in various ways. [11]. So, one method might not find all types of deepfakes effectively.

This study proposes a novel approach to deepfake detection using pixel analysis. By closely examining the pixel characteristics and patterns within manipulated images and videos, we developed an algorithm that can distinguish between real and fake content with high accuracy.

The rest of the paper is organized as follows: Section 2 looks at Related Works while Section 3 presents the proposed method. Experiments, results and discussions follow in Section 4 and the conclusion is drawn in Section 5.

RELATED WORKS

Guera [12] suggested detecting deepfakes recurrent neural networks. They proposed a system that looks at frames in a row using a convolutional Long Short-Term Memory (LSTM). This convolutional LSTM has two parts:

1. A part that gets important stuff from each frame.
2. 2. Another part that understands the order of frames over time.

They used a model called InceptionV3 to quickly understand each frame. Then, they took the important information from this model and gave it to the LSTM, which understands the order of frames. To figure out if a bunch of frames is fake or not, they used some more layers in the network. This helped them decide if the sequence of frames is likely fake or real.

Their dataset consisted 600 videos – 300 deepfake videos sourced from various websites and 300 randomly chosen from the HOHA dataset [14], focusing on human behavior scenes from popular movies. Cozzolino and Verdoliva [15] introduced the noise print camera fingerprint technique. This method employs a CNN to enhance and remove parts of scenes. They trained a Siamese network using pictures taken by different cameras to recognize camera-specific traits. This network, initially designed for noise patterns in photos, is a specialized type [16]. They paired images from the same and different cameras for training. Post-training, they employed CNN in the Siamese Network's approach to identify related noise patterns and display camera-related aspects from a better camera model using the provided image.

S. Bani-Ahmad [17] proposed a Hybrid Deep Learning Model Based on Visual and Audio Features the proposed novel hybrid deep learning model for deepfake video detection, identifies two main approaches for deepfake detection: (1) image-based and (2) video-based. The image-based method used only the visual content of the image to detect deepfakes, while the video-based method considered the temporal changes in the video to identify deepfakes.

Kim, Tariq, and Woo [18] proposed a technique to detect deepfakes in a sophisticated way. They used something called Knowledge Distillation and Representation Learning (ReL). They called their method FReTAL. First, they took frames (like pictures) from real and fake videos. They used FFmeg library on these frames to prepare them. Then, they found face features with the help of MTCNN library. They made sure all faces were the same size. The images were made smaller, like 128 by 128 by 3 pixels.

Table 1 summarizes the contrast between our proposed model and existing models.

Table 1 Comparison of reviewed models

MODEL	**Pixel Signatures**	**Temporal Features**	**Facial Extraction**	**I.N***
[12]	X	✓	X	X
[13]	X	✓	X	X
[14]	X	X	X	X
[15]	✓	X	X	X

[17]	X	✓	X	✓
[18]	X	X	X	X
PROPOSED	✓	✓	✓	✓

*IN: Image Normalisation

PROPOSED METHOD

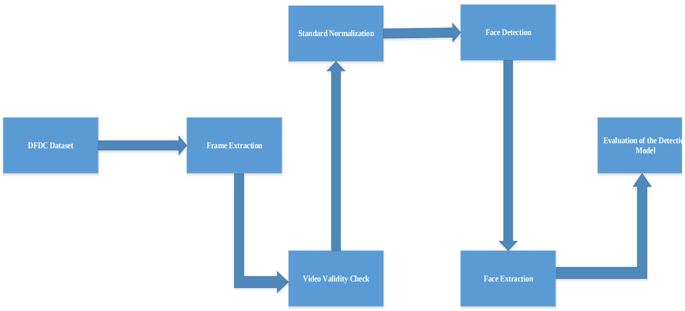
In our research, we proposed a new and effective way to identify deepfake videos, using advanced techniques for high accuracy and trustworthiness. Our method has three main parts:

First, we use a deep learning setup, specifically a convolutional neural network (CNN), to pull out important details from video frames. This CNN has been trained on a large dataset to learn deepfake-related traits.

Second, we identify pixel traits for examination, aiding deepfake detection.

Lastly, we use spatial-temporal analysis to study how frames connect and move in videos. This helps find inconsistencies caused by face manipulation.

Our approach aims to create a strong solution for deepfake detection, outperforming old methods in dealing with new deepfake tech challenges.



We've focused on finding important pixel traits for pixel analysis. By studying the pixel-level properties of real and altered videos, we can uncover patterns and inconsistencies in the deepfake production process. This detailed pixel analysis is crucial to accurately spot deepfakes. Looking closely at facial features, emotions, noise, and how things move helps us detect even subtle differences that human eyes might miss. It's vital to explore these pixel properties to build a powerful deepfake detection model, especially as AI-generated fake content becomes more concerning. This safeguards digital media's authenticity and reliability.

Algorithm 1:

Input: D – DFDC-Dataset (Deepfake video dataset)

```
Initialize empty lists: realVideos, fakeVideos
foreach video in D:
  if video is real:
    Add video to realVideos list
  else:
    Add video to fakeVideos list
Initialize empty list: validVideos
foreach fakeVideo in fakeVideos:
  augmentedVideo = ApplyFrameSlip(fakeVideo)
  if IsValidVideo(augmentedVideo):
    Add augmentedVideo to validVideos list
Initialize face detection model: faceDetector
Initialize deepfake detection model: deepfakeClassifier
foreach video in validVideos:
  frames = ExtractFrames(video)
  for frame in frames:
    detectedFaces = DetectFaces(frame, faceDetector)
    if detectedFaces:
      largestFace = SelectLargestFace(detectedFaces)
      faceImage = ExtractFaceImage(frame, largestFace)
    prediction = PredictDeepfake(deepfakeClassifier, faceImage)
    DisplayPrediction(prediction)
Evaluate model performance metrics
End
```

We break videos into frames, detect and crop each frame's face. Then, these cropped frames create a new video. We choose a threshold from the average frames per video to keep things even. This also considers computer limits. Handling all 300 frames at once is tough in experiments. A 10-second, 30 frames per second video makes 300 frames, taxing computation.

A pre-trained ResNext model is used for feature extraction instead of building from scratch. ResNext is a kind of Residual CNN network that does well with deep neural networks [19]. We're using the "resnext50 32x4d" model for the experiment. It has 50 layers and 32 x 4 dimensions. The model was adjusted by adding necessary layers and picking a good learning rate for the gradient descent to work well. Sequential LSTM takes 2048-dimensional feature vectors from ResNext's last pooling layers. We're using a pre-trained Residual Convolution Neural Network model.

The LSTM layer analyze sequences to figure out timing between frames, with 2048-dimensional vectors as input. One LSTM layer with 2048 latent dimensions, 2048 hidden layers, and 0.4 dropout is used. This compares frames step by step. Leaky ReLU is used. A linear layer with 2048 inputs and 2 outputs teaches average correlation. Adaptive average pooling creates a target-sized picture (H x W). Sequential Layer processes frames one by one. Training is in batches of four. SoftMax layer gauges prediction confidence.

Deepfake detection requires understanding the connections between frames over time. ResNext and LSTM were chosen for their strengths in handling complex visual patterns and sequential video data, respectively – both are critical for precise deepfake detection. This complements the spatial analysis of models like ResNext, enhancing the overall accuracy of the deepfake detection model.

Precision [20] is a measure of accuracy that represents the percentage of correctly classified positive samples among all classified positive samples.

$$Prec = \frac{TP}{TP+FP} \qquad (1)$$

Recall [21] measures how well the model finds actual positives. A high recall means it rarely misclassifies them, showing effective positive identification.

$$REC = \frac{TP}{TP + FN} \qquad (2)$$

The F1-score [22] combines precision and recall for overall model performance. It considers false positives and false negatives, providing a balanced measure. The score goes from 0 to 1: 1 is great, 0 is poor.

$$F1\_score = \frac{2 * Prec * Rec}{Prec + Rec} \qquad (3)$$

The Matthews Correlation Coefficient (MCC) [23] checks binary models. It uses true positives, true negatives, false positives, and false negatives from a confusion matrix. MCC goes from -1 to 1: 1 is great, 0 is random, and -1 is bad.

$$MCC = \frac{(TP * TN - FP * FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \qquad (4)$$

EXPERIMENT RESULTS AND DISCUSSION

A. Dataset Description

The Deepfake detection challenge dataset (DFDC) [24] is utilized for our model training and testing. The DFDC dataset consists of 5000 videos. The dataset has 4 features which are filename, split, original, and label. This dataset was chosen because of the realistic deepfake content, the dataset is large and diverse in that the dataset not only features one ethnic group, which makes it more reliable.

B. Data Preprocessing

The video data goes through several steps before analysis: importing, cleaning, filtering, and converting. Validation of video files and handling errors during validation are vital. Videos are split into frames, and each frame's face is found and cut out. Frames are then put together into new videos using an average of 150 frames, considering computer limits. This eases the computer load. The first 150 frames show how LSTM works, with 30 frames per second and a 112 × 112 resolution for creating new videos.

After Splitting videos, we created new videos from existing ones, we checked if the videos had actual frames through validation. This helped remove videos with audio alterations or issues affecting model training and detection. We started with 5000 videos, but after checking, 257 were identified as bad. Leaving us with 4743 good videos for ensuring better model performance.

C. Experiment Setup

70% of the data in the optimized dataset will be utilized for training, while the remaining 30% will be used for testing. Every sample has the same opportunity to take part in the trials as a training or testing sample. The same performance indicators (accuracy, precision, and recall) are considered during each cycle (training and testing). The Model was developed using Python in Google colaboratory a hosted Jupyter notebook service.

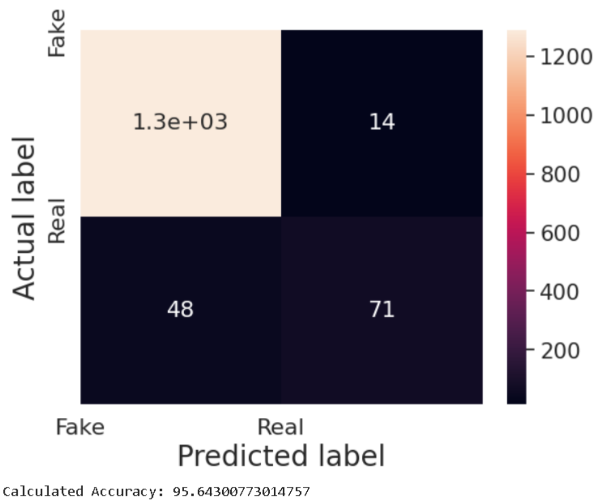
D. Results and Discussion

We used the DFDC dataset for training, testing, and validation. It includes 5000 videos and aims to support research in facial manipulation detection. The Deepfake Detection Challenge (DFDC), launched in September 2019 [25], involves collaboration between industry, academia, and civil society organizations. This challenge provides a dataset with human face recordings and comments indicating whether they were altered. In the preview dataset, 74% are female, 26% are male, and 68% are Caucasian. The dataset is supervised, meaning it has labels for training.

Table2 DFDC Dataset description

Filename	**Label**	**Split**	**Original**
etychryvty.mp4	FAKE	train	uqtqhiqymz.mp4
fonrexmbzz.mp4	FAKE	train	fufcmupzen.mp4
ddtbarpcgo.mp4	REAL	train	ddtbarpcgo.mp4

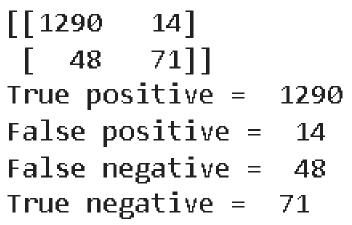
The initial stage of the experiment concentrates on enhancing the dataset and developing the machine learning model. The execution's complexity (during training or testing) is measured in seconds, and accuracy is measured in percentage. It is decided on metrics including accuracy, recall, f1-score, ROC, MCC, and AUC.



```
Calculated Accuracy: 95.64300773014757
```

Figure 1 Accuracy & Confusion Matrix

Figure 1, shows a confusion matrix, also known as an error matrix, which is a unique type of table that may be used to display an algorithm's performance in the context of machine learning, notably the statistical classification issue. The classifiers' accuracy score of 96.3%, which we used to train our model, is shown in Figure 1 above.



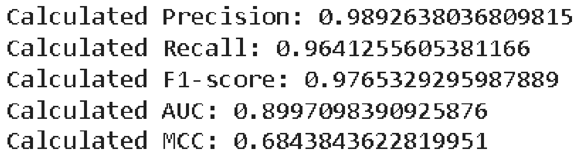
```
[[1290   14]
 [  48   71]]
True positive =  1290
False positive =  14
False negative =  48
True negative =  71
```

Figure 2 Detailed Confusion Matrix

Figure 2 above shows a detailed confusion matrix [26], that depicts the exact classification results produced by the model. Out of the training set is depicted therein.

In Figure 3 below, the model's precision is indicated as 0.98.

The recall is particularly important in scenarios where the cost of false negatives is high, such as in medical diagnoses or fraud detection. In the context of our model, a recall score of 0.989% is shown below in Figure 3.



```
Calculated Precision: 0.9892638036809815
Calculated Recall: 0.9641255605381166
Calculated F1-score: 0.9765329295987889
Calculated AUC: 0.8997098390925876
Calculated MCC: 0.6843843622819951
```

Figure 3 Evaluation Matrices

F1-score is great when balancing precision and recall matters, useful for imbalanced datasets. Our model's F1-score is 0.97% as in Figure 3.

An MCC score of 0.68% (in Figure 3) indicates a fair correlation between predicted and real labels, but not perfect.

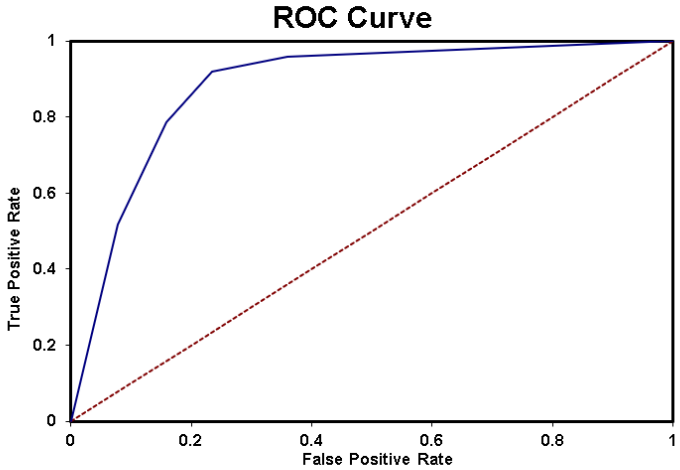


Figure 4 ROC Curve

In binary classification, the AUC (Area Under the Curve) [27] score is important, especially with ROC curves [28]. It shows how well a model separates positive and negative examples. From Figure 4, the AUC score of 0.89% means our model is likely to rate positive instances higher than negative ones. It's good at assigning higher probabilities to positives. This helps when identifying important case matters.

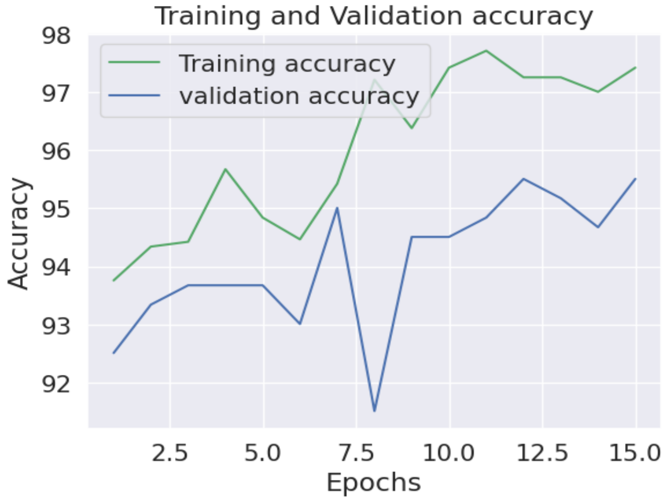


Figure 5 Training and Validation accuracy

Training and validation accuracy evaluate the model's performance in training. Training accuracy assesses the model's predictions on familiar training data [29]. High training accuracy reflects effective learning from this data. Validation accuracy measures the model's performance on distinct validation data, indicating its potential on new data. Figure 5 depicts training and validation accuracy throughout training and testing.

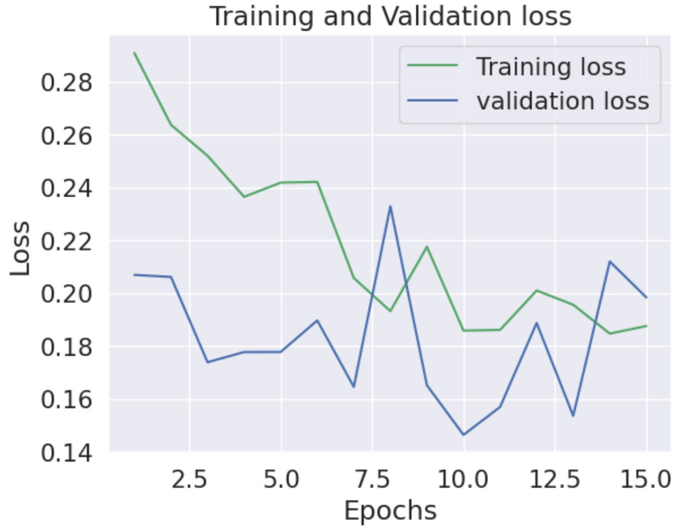


Figure 6 Training and Validation loss

Training and validation loss evaluate the model's performance during training. Training loss checks how well the model fits the training data [30], measuring the gap between predictions and actual labels. Reducing training loss aids learning. Validation loss, from a distinct dataset, assesses the model's new data performance fairly. Figure 6 displays training and validation loss during training and testing.

Table 3 Model Comparison

MODEL	Pixel Signatures	Temporal Features	Facial Extraction	I.N*	Acc
[18]	X	X	X	X	93.2%
PROPOSED	✓	✓	✓	✓	95.6%

*Acc: Accuracy

Table 3 shows a comparison against, Bani-Ahmad et al. who proposed a hybrid deep learning model for deepfake detection, blending audio and visual features. They employ a RNN to capture audio patterns and a pre-trained CNN to extract visual cues. Our study, in contrast, concentrates on analyzing facial deepfakes through pixel attributes. We utilize a CNN-pre-trained Resnext-50 and LSTM architecture to identify visual distortions, pixel inconsistencies, and artifacts caused by deepfake manipulation.

CONCLUSION

Our study focused on identifying pixel features for analyzing and classifying videos as deepfake or authentic. We developed an effective model using supervised learning, pixel analysis, a pre-trained model, and LSTM. Our method worked well, accurately spotting deepfakes. We discovered that fake videos have distinct traits like pixelation, smooth borders around changes, and time inconsistencies.

Our proposed deepfake detection model offers a reliable way to address the rising issue of fake videos in social media, politics, and finance. With its high accuracy of 95.6%, it can help reduce the security risks posed by deepfake videos in real-world scenarios.

Real-time deepfake detection is essential, but current methods are slow due to high computational needs. Additionally, these methods struggle to tell twins apart due to their similar faces. This flaw could misidentify twins, especially if they were involved in creating the deepfake. Solving these issues is crucial for effective deepfake detection, which should handle twin identification and swiftly spot diverse deepfake types in real time. This study faces challenges: the evolving deepfake landscape, concerns about deceptive attacks, resource and time demands, and the interdisciplinary nature of solutions.

REFERENCES

[1] Maras MH, Alexandrou A. Determining authenticity of video evidence in the age of artificial intelligence and in

the wake of Deepfake videos. The International Journal of Evidence & Proof. 2019 Jul;23(3):255-62.

[2] Kietzmann J, Lee LW, McCarthy IP, Kietzmann TC. Deepfakes: Trick or treat?. Business Horizons. 2020 Mar 1;63(2):135-46.

[3] Vizoso Á, Vaz-Álvarez M, López-García X. Fighting deepfakes: Media and internet giants' converging and diverging strategies against Hi-Tech misinformation. Media and Communication. 2021 Mar 3;9(1):291-300.

[4] LaMonaca JP. A break from reality: Modernizing authentication standards for digital video evidence in the era of deepfakes. Am. UL Rev.. 2019;69:1945.

[5] Darapaneni N, Evoori AK, Vemuri VB, Arichandrapandian T, Karthikeyan G, Paduri AR, Babu D, Madhavan J. Automatic face detection and recognition for attendance maintenance. In2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS) 2020 Nov 26 (pp. 236-241). IEEE.

[6] Prince SJ, Elder JH, Warrell J, Felisberti FM. Tied factor analysis for face recognition across large pose differences. IEEE Transactions on pattern analysis and machine intelligence. 2008 Apr 18;30(6):970-84.

[7] Yu CM, Chang CT, Ti YW. Detecting deepfake-forged contents with separable convolutional neural network and image segmentation. arXiv preprint arXiv:1912.12184. 2019 Dec 21.

[8] [8] Ramachandran S, Nadimpalli AV, Rattani A. An experimental evaluation on deepfake detection using deep face recognition. In2021 International Carnahan Conference on Security Technology (ICCST) 2021 Oct 11 (pp. 1-6). IEEE.

[9] Westerlund M. The emergence of deepfake technology: A review. Technology innovation management review. 2019;9(11).

[10] Ahmed SR, Sonuç E, Ahmed MR, Duru AD. Analysis survey on deepfake detection and recognition with convolutional neural networks. In2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) 2022 Jun 9 (pp. 1-7). IEEE.

[11] Shahzad HF, Rustam F, Flores ES, Luís Vidal Mazón J, de la Torre Diez I, Ashraf I. A Review of Image Processing Techniques for Deepfakes. Sensors. 2022 Jun 16;22(12):4556..

[12] D. Guera and E. J. Delp, "Deepfake Video Detection Using Recurrent Neural Networks," Proc. AVSS 2018 - 2018 15th IEEE Int. Conf. Adv. Video Signal-Based Surveill., 2019, doi: 10.1109/AVSS.2018.8639163.

[13] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "MesoNet: A compact facial video forgery detection network," 10th IEEE Int. Work. Inf. Forensics Secur. WIFS 2018, no. December, 2019, doi: 10.1109/WIFS.2018.8630761..

[14] O. Fried et al., "Text-based editing of talking-head video," ACM Trans. Graph., vol. 38, pp. 1–14, 2019.

[15] Cozzolino D, Verdoliva L. Noiseprint: A CNN-based camera model fingerprint. IEEE Transactions on Information Forensics and Security. 2019 May 13;15:144-59.

[16] Cui Y, Guo D, Shao Y, Wang Z, Shen C, Zhang L, Chen S. Joint classification and regression for visual tracking with fully convolutional siamese networks. International Journal of Computer Vision. 2022 Jan:1-7.

[17] S. B.-A. et Al, "Hybrid Deep Learning Model for Deepfake Video Detection Based on Visual and Audio Features," pp. 1–11, 2021.

[18] M. Kim, S. Tariq, and S. S. Woo, "FReTAL: Generalizing deepfake detection using knowledge distillation and representation learning," IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work., pp. 1001–1012, 2021, doi: 10.1109/CVPRW53098.2021.00111.

[19] Diba A, Fayyaz M, Sharma V, Arzani MM, Yousefzadeh R, Gall J, Van Gool L. Spatio-temporal channel correlation networks for action classification. InProceedings of the European Conference on Computer Vision (ECCV) 2018 (pp. 284-299).

[20] Davis J, Goadrich M. The relationship between Precision-Recall and ROC curves. InProceedings of the 23rd international conference on Machine learning 2006 Jun 25 (pp. 233-240).

[21] Torgo L, Ribeiro R. Precision and recall for regression. InDiscovery Science: 12th International Conference, DS 2009, Porto, Portugal, October 3-5, 2009 12 2009 (pp. 332-346). Springer Berlin Heidelberg.

[22] Avola D, Cinque L, Foresti GL, Lamacchia F, Marini MR, Perini L, Qorraj K, Telesca G. A shape comparison reinforcement method based on feature extractors and f1-score. In2019 IEEE International Conference on Systems, Man and Cybernetics (SMC) 2019 Oct 6 (pp. 2155-2159). IEEE.

[23] Chicco D, Jurman G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC genomics. 2020 Dec;21(1):1-3.

[24] Dolhansky B, Bitton J, Pflaum B, Lu J, Howes R, Wang M, Ferrer CC. The deepfake detection challenge (dfdc) dataset. arXiv preprint arXiv:2006.07397. 2020 Jun 12.

[25] Dolhansky B, Howes R, Pflaum B, Baram N, Ferrer CC. The deepfake detection challenge (dfdc) preview dataset. arXiv preprint arXiv:1910.08854. 2019 Oct 19.

[26] Liang J. Confusion matrix: Machine learning. POGIL Activity Clearinghouse. 2022 Dec 12;3(4).

[27] Huang J, Ling CX. Using AUC and accuracy in evaluating learning algorithms. IEEE Transactions on knowledge and Data Engineering. 2005 Jan 31;17(3):299-310.

[28] Bradley AP. The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern recognition. 1997 Jul 1;30(7):1145-59.

[29] Sun Y, Wang X, Tang X. Deep learning face representation from predicting 10,000 classes. InProceedings of the IEEE conference on computer vision and pattern recognition 2014 (pp. 1891-1898).

[30] Zhang C, Bengio S, Hardt M, Recht B, Vinyals O. Understanding deep learning (still) requires rethinking generalization. Communications of the ACM. 2021 Feb 22;64(3):107-15