Journal of Information Systems Engineering and Management
2025, 10(55s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

MSFF-Net: A Deepfake Detection Network Based on Multi-Space Feature Fusion Technique

Raveena[1*], Rita Chhikara[2], Pooja Punyani[3]

[1]*Ph.D candidate, Department of Applied Sciences, The NorthCap University, Gurugram, India; raveena22asd002@ncuindia.edu*
[2]*Professor, Department of Computer Science and Engineering, The NorthCap University, Gurugram, India; ritachhikara@ncuindia.edu*
[3]*Assistant Professor, Indira Gandhi National Open University (IGNOU), New Delhi, India; Poojapunyani.pp@gmail.com*
[*] *Corresponding Author: raveena22asd002@ncuindia.edu*

ARTICLE INFO

Received: 30 Dec 2024

Revised: 05 Feb 2025

Accepted: 25 Feb 2025

ABSTRACT

The history of digital manipulation has entered a new chapter with the development of deepfake technology, opening the door to realistic yet misleading visual content creation. Deepfake technology's rapid rise has raised worries about potential misuse because it may produce fake or misleading information with incredible realism. Because deepfakes made it possible for actors' faces to be perfectly altered in movies, the entertainment industry was the first to adopt them. Since then, however, they have evolved and been utilized in a variety of settings, including social networking sites, political issues, and journalism. Feature extraction from digital photographs had a major impact in the field of detecting deepfakes, where separating actual information from manipulated media is crucial. Many techniques for detecting deepfakes based on fabricated features have been developed recently. Textural features are a frequently used type of forged feature. However, most of the current detection methods relies on single space features, ignoring other potentially informative features. To overcome this drawback, this study presents a novel deepfake detection network named MSFF-Net (Multi-Space Feature Fusion Network). This network integrates Histogram of Oriented Gradients features with the deep features extracted using ResNet50 model. HOG descriptors analyze local intensity gradients, which helps in understanding the texture and shape of the image, while the deep features extracted by the ResNet50 model capture high-level semantic information. By combining these diverse feature sets, one can create a richer representation of the data that encompasses both low and high-level characteristics, thereby enhancing the robustness and effectiveness of deepfake detection. The experiments are performed on the publicly available datasets 140K real and fake faces dataset, DFDC, and Celeb-df. MSFF-Net showed better performance than the other SOTA models, thereby improving the ability to detect deepfakes.

Keywords: Deepfake; feature fusion; Histogram of Oriented Gradients, Convolutional Neural Network, Resnet50, Multi-space features.

INTRODUCTION

The term "deepfake," is a combination of the words "deep learning" and "fake," pertains to the method by which the original subject of a target image is modified. With the use of technology, the impersonator appears to be doing or saying things that they have never done or said. GANs [1] and variational-autoencoders (VAEs) [2] are used to construct deepfakes. The performance of recent generative adversarial network face creation networks, like StyleGAN [3], StarGAN [4], and Interface-GAN [5], is quite good. The creation of deepfake videos is easy with these networks. The most popular method for producing deepfakes involves training two VAE models to generate human faces; the only thing that is changed to produce an image is the decoder portion. The two models share the encoder portion, while the decoder portion is trained independently. Face-swapping GANs create Deepfake films by utilizing a GAN (such as StyleGAN) with the neural talking-heads technique. The latest DFDC dataset [6] was employed by a Deepfake autoencoder, which utilized a mask face swap technique to modify the facial landmarks of each frame The EfficientNetwork-b7 [7] model performs better than other models when trained on the DFDC dataset [6] released by Facebook AI.

Visual manipulation is deemed detrimental or falsified when the content is distorted for the purpose of deception, as fabricated material can easily sway the decisions and thoughts of others. Verifying visual data is therefore essential for the preservation and stability of the community. Three distinct categories comprise digital visual manipulations: image, video and facial features manipulation (FCM). Within the initial classification, image manipulation, any component or entity within the picture is altered, including copy-move and splicing [8, 9, 10, 11]. The copy-move technique involves copying and relocating a section of an image within the same image. A segment of one picture is transferred and pasted into another during the splicing process. Vector machines (VMs) are utilized to alter video in the spatiotemporal, or spatial-temporal domains [12, 13, 14]. Examples of VMs include intra-frame forgery and inter-frame forgery. FCM involves the manipulation of a single human visage, while the remaining content remains intact [15, 16, 17]. In recent times, the designation "DeepFake" has been applied to all FCM [18], including Face2Face, NeuralTextures, FaceSwap and others.

FCM is categorized as high-risk due to the significant role that the human visage assumes in establishing interactions that validate or communicate a particular message; any form of manipulation at this stage would result in the establishment of an erroneous message. Incredibly, over the past three years, as deep learning has advanced at an accelerated rate, FCM detection and generation have garnered considerable interest from researchers around the globe. Deep learning and variants of GANs [19] that can generate high-fidelity images have enabled anyone to generate a face that appears real. Knowledgeable individuals refer to the latter as the Deepfake. The problem is that models for generating Deepfake with high performance are readily available. When utilized with malice, deepfakes present an imminent and critical threat to the credibility of the news that we are exposed to.

The significant rise in false face photos and videos has been driven by the quick development of Deepfake algorithms, which are producing increasingly realistic-looking fake photographs and movies. Videos and photos with false content bring up a number of unsettling issues on widely used social media platforms, including fraud and for spreading fake news. As a result, the need for Deepfake detection techniques to mitigate its effects has increased significantly [20–22]. Deepfake detection is actually a difficult classification problem. Finding the distinctions between actual and false photos is the most crucial part of Deepfake detection. These problems might be solved by developing a machine learning system that uses techniques and algorithms to extract features from photos and distinguish between actual and fraudulent ones. This system would be able to deliver correct findings. Despite considerable efforts to identify these manipulations, the efficacy of detection remains inadequate, which parallels the progress made in generation methods.

The primary objective of this study is to expose Deepfake through the development of an innovative method that integrates multi-space features to achieve high detection performance. The primary contributions of our study is:

- A novel method Multi-space feature fusion-based detection system using a fusion of Histogram of Oriented Gradients [23], ResNet50 [24] is proposed, achieving better results compared to the SOTA models.
- Then feature fusion is applied to create a feature vector of each image.
- After feature fusion, Principal Component Analysis is applied for feature selection, focusing on selecting features crucial for detecting deepfakes.
- SVM is used as classifier to improve the performance of the MSFF-Net.
- When evaluating MSFF-Net, it is compared with different SOTA models on publicly available datasets 140K real and fake faces dataset [25], Celeb-df [26], and DFDC [27].

The remaining sections are organized as follows: the literature review is presented in Section 2. Section 3 contains the motivation. In Section 4, the methodology employed for this paper is explicated. The experimental outcomes and a comparison with other SOTA methods are detailed in Section 5. The conclusion is presented in Section 7, following the discussion in Section 6.

LITERATURE REVIEW

Korshunov P. et al. (2018) [28], assessed how well detection methods could discriminate between authentic video clips and DeepFake video clips, describing DeepFake detection as a problem of binary classification. This method focused on measuring the quality of the image, and an SVM classifier was used to classify videos of high quality with an error rate of 8.97%. Sadly, the subjectivity of such technique limits it. The system also needs evaluation to look into the sensitivity of human experimentation for Deepfakes, as well as a more reliable detection method. Masood M. et al. (2021) [29], uses a CNN models Shuffle Net and Alex Net for classification. The procedure starts with normalising the images before performing an error level analysis. Afterwards, the precise information from the CNN models are extracted using SVM and KNN techniques. The proposed method offers a cutting-edge method to discern between authentic and false images. The images were pre-processed by downsizing them to 225 x 225 before comparing the compression ratio between original and fake shots. For feature extraction they have used the Shuffle Net and Alex Net models. Finally, KNN and SVM classifiers were used to categorize the deep features. This work makes use of “Real and Fake Face detection" dataset. Using KNN and SVM, respectively, Shuffle Net's accuracy was 88.2% and 87.9%, whereas Alex Net's accuracy was 86.8% and 86.1%.

Ismail A. et al. (2021) [30], describes a novel deepfake detection method that makes use of a convolutional neural network, an aggressive gradient boost, and a single look (YOLO-CNN-XGBoost). Using the CelebDF-FF++ (c23) merged dataset, the methodology achieves AUC- score of 90.62%, 85.39% sensitivity, 93.53% specificity, 86.36% F1-measure 85.39% recall, 90.73% accuracy, 87.36% precision. Taeb M. et al. (2022) [32] examines the most well-known face-detection classification models, including VGG19, Customized Convolutional neural network, and DenseNet121, using a large genuine and synthetic image detection database. The last layer's result vectors were extracted and used to represent the pictures in the examined architectures. For the Customized CNN architecture, DenseNet architecture, and VGG-19 architecture, the vector sizes were 512, 1024, and 2048, respectively. The top 50 principal components and the vector points of the dominating variable were kept using principal component analysis (PCA). The output vectors from principal component analysis were divided into two class using a SVM, legitimate and fraudulent. VGG19 performs better than other examined models and obtains highest accuracy of 95%.

Masood M. et al. (2021) [31], presents a pipeline for person face identification and detection from input visual samples. The deep features are computed from the retrieved faces using a variety of deep learning algorithms. In order to classify the data as real or altered, an SVM classifier is lastly trained over these features. They used the OpenFace2 package for facial detection in our implementation. OpenFace can estimate head posture, track eye-gaze, and identify facial activity units in addition to detecting faces using 2D and 3D facial landmarks. Ten different cutting-edge feature extraction models have been taken into account. VGG-16, ResNet101, XceptionNet, InceptionResV2, VGG-19, MobileNetv2, EfficientNet, Inception V3, NASNetMobile, DenseNet-169 are the pre-trained CNNs models employed. With a accuracy of 98%, the DenseNet-169 model had the highest accuracy of all the ones used. It was closely followed by the XceptionNet with a accuracy of 97.2%. The VGG16 exhibits the lowest accuracy, 89%.

Deng L. et al. (2022) [33], a new EfficientNet-V2 network is suggested for use in determining the authenticity of images and videos. This study uses FF++ data and achieved an accuracy of 97.90%. Rafique et al. (2023) [34], proposed a method, which combines Error Level Analysis and deep learning techniques. The researchers investigate the efficacy of ELA in uncovering inconsistencies introduced by image compression, while simultaneously harnessing the capabilities of Deep CNNs to capture essential features. The research explores three significant CNN architectures—GoogLeNet, ResNet18, and SqueezeNet—as solutions for accurate deepfake detection. Additionally, classifiers such as SVM and KNN are introduced to refine the precision of classification. ResNet18 coupled with KNN achieves a remarkable accuracy of 89.5%.

Ismail A. et al. (2022) [35], uses a face detector called You Only Look Once (YOLO) to identify people in video frames. The first feature extraction method, a suggested CNN model, is built on the HOG approach. An improved version of CNN Xception is the second one. The two recovered sets of information are combined and fed into a sequence of GRUs for extractacting the temporal as well as spatial data and determine the veracity of movies (GRUs). The proposed method achieved an AUROC score of 95.53%, accuracy of 95.56%, precision of 97.06%, recall of 96.21%, F-score of 96.63%, sensitivity of 96.21%, and specificity of 94.29%.

To identify fake videos, Bacanin N. et al. (2022) [36] suggests using the YOLO-Local Binary Pattern Histogram (YOLO-LBPH). YOLO is used to identify facial region in pictures or video frames. Using the EfficientNet-B5 technique, the spatial characteristics are retrieved from the facial image. Then to extract temporal features these are given as input to LBPH, Multi-Task Cascaded CNN are used to segregate the frames (MTCNN). This work uses EfficientNet-B5 to extract facial features. The precision scores for the CASIAWebFace, DFFD, and CelebDFFaceForensics++ (c23) datasets are, respectively, 86.88%, 88.9%, and 91.35%. Recall rates for the CASIA-Web Face dataset are 94.35%, the DFFD dataset is 93.7%, and the Celeb-DF-Face Forensics ++ (c23) dataset is 92.4%.

Raveena et al. (2023) [37] presents a comparative study of several ML algorithms. And performance is evaluated by calculating measures like Precision value, F1-score, AUC score, recall value, and accuracy score.

MOTIVATION

Visual content, specifically that which depicts human presence and behavior, is frequently regarded as evidence that the events in question transpired. Recent technological advancements have facilitated and expanded the accessibility of altering such content, thereby contributing to the transformation of thinking. Social injustice is compromised when individuals who lack the means to verify its veracity believe false news that is disseminated via social media. Therefore, the ability to identify false videos is growing in importance.

However, the majority of current models relied on RGB features for detection purposes, resulting in a restricted dataset that complicates the identification of images or videos by distinct information domains. To rectify the aforementioned deficiencies, we incorporate HOG characteristics into the deep features derived from the ResNet50 model. HOG descriptors are capable of examining local intensity gradients, thereby offering valuable insights into the texture and shape attributes present in images. In recent years, CNN networks have shown remarkable success in presenting and learning graph structure data. This success has translated into exceptional performance in several computer vision tasks. Motivated by these developments, ResNet50 model is implemented to detect manipulation abnormalities and discrepancies in synthetic images or movies, leading to improved performance.

METHODOLOGY

The prime objective of MSFF-Net is to precisely detect and differentiate deepfake photos from genuine ones. **Figure 1** shows the block diagram of MSFF-Net, Preprocessing the data is the first step, after which feature extraction is carried out on the dataset using feature extraction methods. Following the process of feature extraction, Principal Component Analysis, was utilized for selecting features, and machine learning model is then used for categorization.

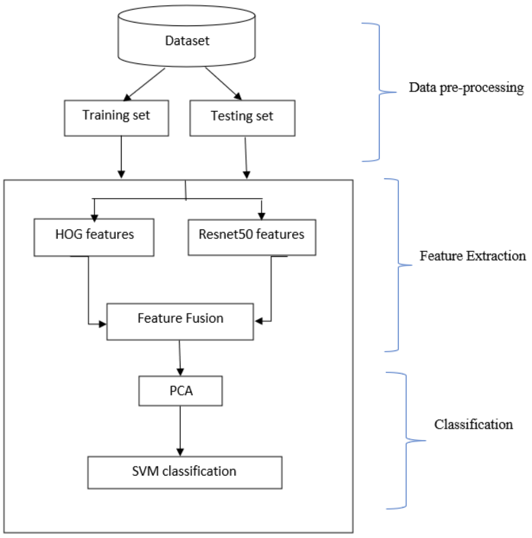


Figure 1. General block diagram of the MSFF-Net

Data Pre-Processing

To prepare each input image for feature extraction, the pre-processing phase is applied. The data is partitioned into two separate subsets: train, test. This split allows to train the models on a good amount of data, validate their performance during training, and evaluate their accuracy and robustness on unseen data. To ensure consistency in the input data and enable efficient training, all images are resized to 224x224 pixels. **Figure 2** and **Figure 3** presents some examples of real and fake images.

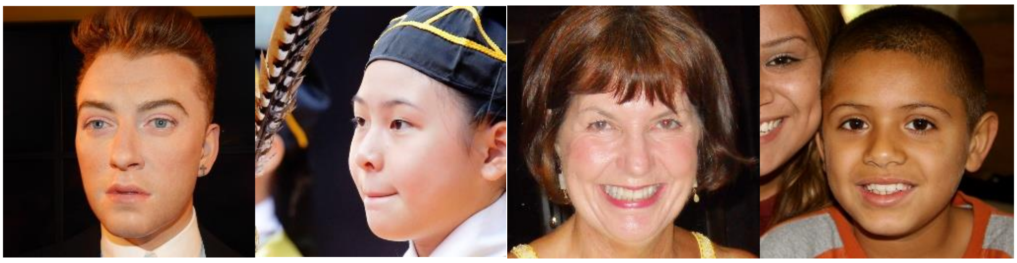


Figure 2. Real face images

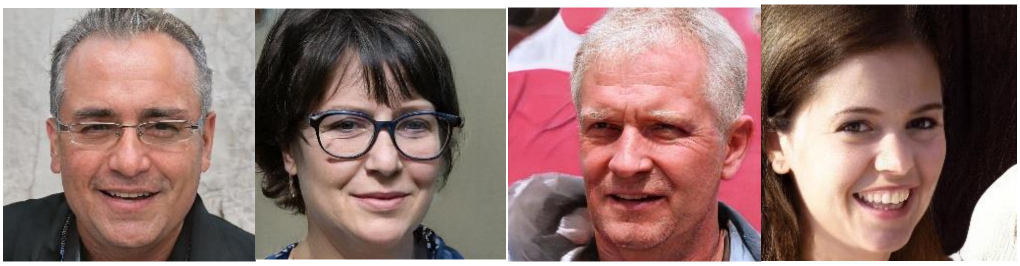


Figure 3. Fake face images

Feature Extraction

In addition to texture and shape features, a CNN based deep features are used for computing the final feature vector of each face image.

Deep-learning based feature extraction

The feature extraction approach employs a pre-trained Resnet50. In this process, the convolutional base of the model employs it to extract deep features, while discarding the remaining network components. **Figure 4** illustrates the architecture of ResNet50.

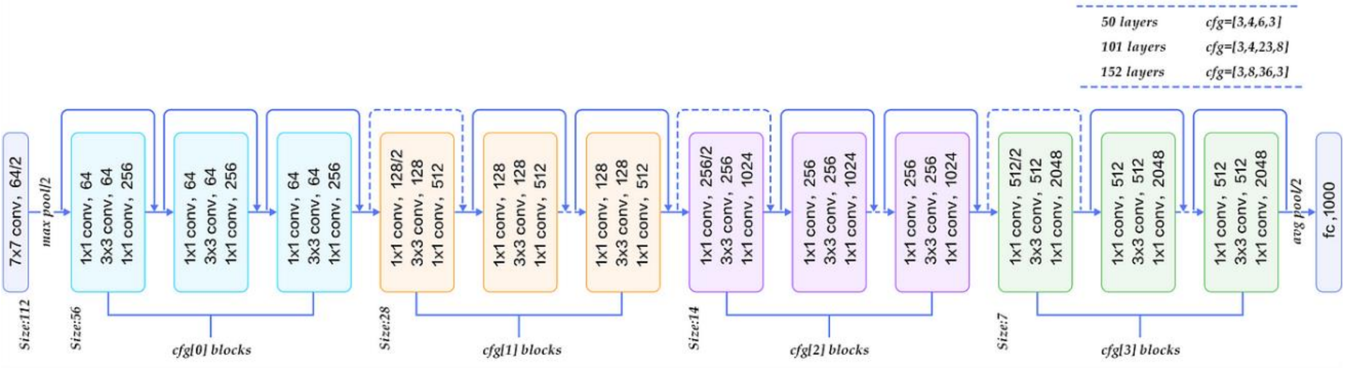


Figure 4. Architecture of ResNet50

To improve detection performance and accommodate specialized features, this ResNet50 model is fine-tuned using the 140k real and face datasets [25], Celeb-df [26], and DFDC [27]. As input for the model, every face image is scaled to 224×224×3. The final convolutional block (conv5_block3_out) is used to derive the feature vector. The size of the feature vector is 2048.

HOG feature extraction

According to Dalal et al. [38], HOG is a highly resilient and resistant to environmental changes representation that finds extensive utility in recognition and detection tasks. Figure 5 illustrates the flowchart illustrating the process of calculating HOG feature descriptors. The algorithm comprises the subsequent comprehensive steps: Prior to beginning the human detection algorithm, the image is preprocessed and scaled to the desired dimensions, such as 128 by 64 pixels. Then, for a particular image, a detection window is established.

Following this, the gradient is calculated, which comprises the magnitude and orientation of every pixel in the detecting window. The gradient can be computed using a variety of operators; for example, it can be expressed as follows [38]:

$$G_x(x,y) = P(x+1,y) - P(x-1,y), \tag{3}$$

$$G_y(x,y) = P(x,y+1) - P(x,y-1),$$

The variables Gx(x, y), Gy(x, y), and P(x, y) denote the horizontal and vertical gradients, the value of the pixel situated in (x, y) respectively.

$$G(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \tag{4}$$

$$\theta(x,y) = arctan\frac{G_x(x,y)}{G_y(x,y)}$$

where θ(x, y) and G(x, y) denote the magnitude and orientation of the pixel at the point (x, y), respectively. The derivation process of each pixel may not only extract profile information but also helps in reducing the impact of illumination. Additionally, in the third phase, the image inside the detection window is separated into many cells that are all the same size. As an adjustable parameter, the cell size is proportional to the dimensions of the image or object. For instance, an 8*8 cell may be utilized to identify pedestrians within a 128*64 detecting window. In the fourth step, the direction gradient histogram is calculated for every single cell based on the gradient magnitude and orientation of every pixel. The gradient's direction is partitioned into bins along the n-axis, and the histogram for each bin is tallied in accordance with the orientation of every pixel within the cell. Consequently, each pixel's weight in the histogram is represented by the gradient's magnitude. Encoding the local area of the image while preserving the importance of each area is the goal of this stage. Fifth, because to variations in lighting and backdrop contrast, the gradient density will shift across a wide range. To mitigate the issue or reduce the impact of light sensitivity, adjacent cells are conbined into blocks. This results in the histogram of a block being formed by concatenating the histograms of each cell in a specific order.

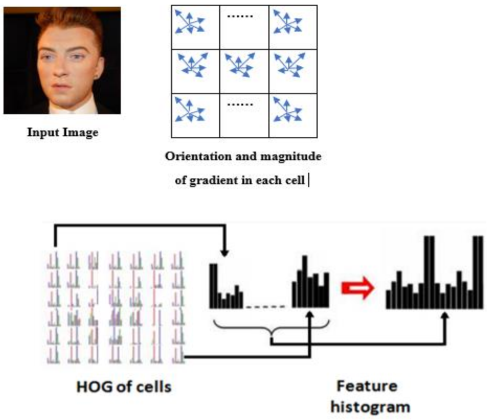
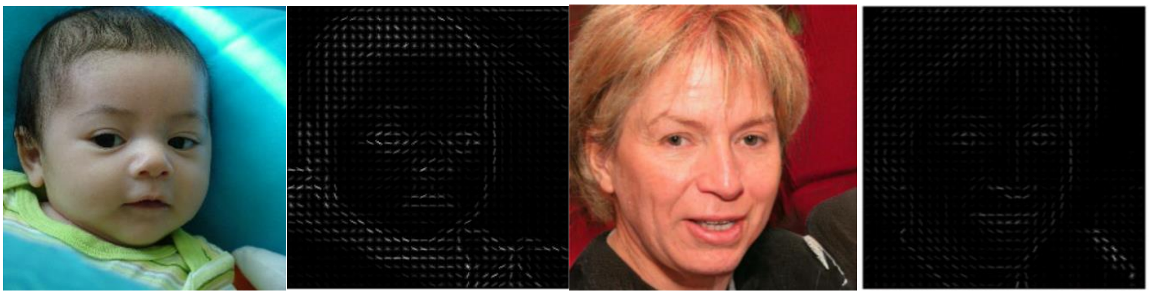


Figure 5. The processes of calculating the HOG feature descriptor

To further lessen the impact of light and shadow, each block will undergo the normalization process. Some cells belong to distinct blocks, allowing the same cell to possess multiple normalization properties. Ultimately, the target image's hog descriptor is created by concatenating the histogram characteristics from various blocks. **Figure 6** HOG images real and fake images.



(a) Real Face (b) HOG of real face (c) Fake Face (d) HOG of Fake face

Figure 6. Example of HOG face images

Feature Fusion

After applying feature extraction modules, we have two features values: HOG feature vector of size 26244, ResNet50 feature vector of size 2048. Finally, the feature vectors are concatenated, where the final feature vector length is 28292. By combining these diverse feature sets, one can create a richer representation of the data that encompasses both low-level and high-level characteristics. After concatenating, PCA has been used to minimize the feature vector because the number of features was relatively high. Any irrelevant features that don't considerably improve the classification algorithms' accuracy are eliminated using PCA. The chosen features are supplied individually to the ML classifiers for classification.

Classification

SVM is used to tackle problems related to text recognition, face identification, handwriting analysis, classification, and more. High-dimensional data sets are utilized in this classifier. In contrast to other machine learning techniques, support vector machines maximize the distance between the nearest data points across all classes through their implementation of the decision boundary. It operates in a very straightforward and easy way: all you have to do is locate the hyperplane that divides the data points into several target classes [39]. The decision boundary produced by SVM is known as the hyperplane. Mathematically, SVM is defined as

$$(x) = sign(w.x + b) \tag{7}$$

Here, sign () is defined as 1 for positive numbers and −1 for negative numbers. Additionally present are input data (x), weight (w), and hyperplane bias (b). SVMs are utilized for binary classification by employing a Gaussian radial basis kernel function.

$$\mathrm{G}(y_i, y_j) = y_i' y_j \tag{8}$$

RESULTS

Implementation Details

The MSFF-Net has been implemented in Python using TensorFlow, and Keras and it was trained in a Kaggle notebook using 140k real and fake faces dataset [25], Celeb-df [26] and DFDC [27] datasets. The dataset is divied into 80-20 ratio for traing and testing. **Figure 7** presents the overall framework of the MSFF-Net. To enhance the detection performance, Resnet50 [24] is used to obtain deep-learned features from face images. Following this, designate a particular model layer to characterize the features. The ResNet pre-trained network was fine-tuned via transfer learning using the Adam optimizer, with a lr rate of 0.001 and trained for 20 epochs. These hyperparameters were determined empirically in an effort to attain the highest level of performance.

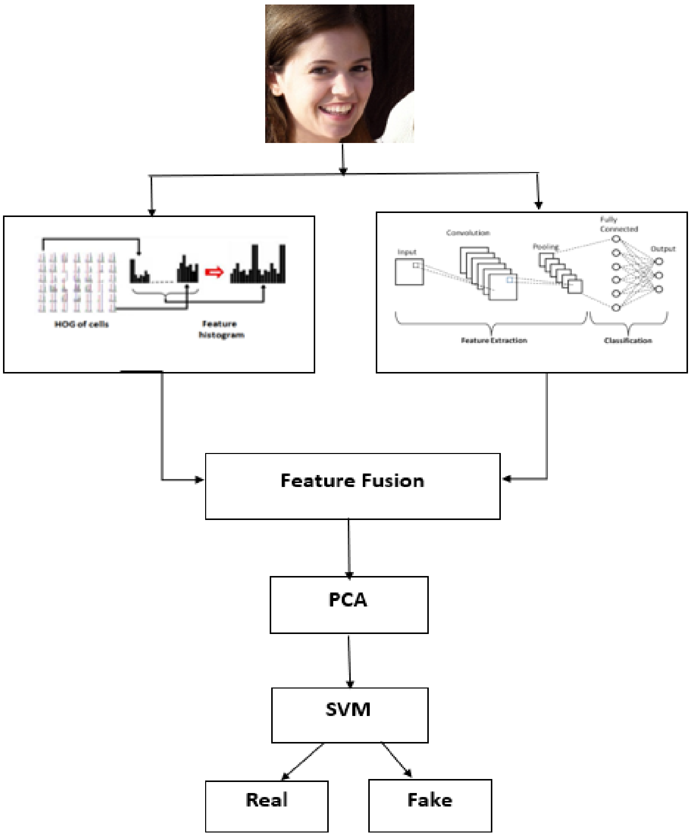


Figure 7. The overall framework of MSFF-Net.

Datasets

The experiments are performed using the 140k Real and Fake faces [25], DFDC [27], and Celeb-df [26] datasets. The 140k Real and Fake faces [25] dataset is downloaded from Kagglke website, it consists of 140,000 images. 590 authentic YouTube videos were used to generate 5,639 DeepFake videos for 59 celebrities of various ages, genders, and ethnic backgrounds [26]. DFDC [27] comprises 5,000 Snapchat-generated videos, both authentic and fabricated. As per the MSFF-Net proposal, every final face image is resized to a dimension of 224×224.

Performance metrics

For evaluating the effectiveness of MSFF-Net various metrics, like the confusion matrix, accuracy, F1-measure, false positive rate, sensitivity, specificity, false negative rate, positive and negative predictive values were used.

Confusion matrix

A table that provides a comprehensive evaluation of the performance by presenting the true and predicted category of data elements. It aids in the visualization and quantification of the model's faults in differentiating between various classes [37].

Hinge Loss

Also known as max-margin loss, is a popular loss function used in binary and multiclass classification tasks, particularly in SVM and related models. It is designed to optimize the margin between classes, aiming to ensure a clear separation boundary. In classification tasks, the hinge loss function is crucial for training models to make accurate predictions while maximizing the margin between different classes. By penalizing incorrect predictions, hinge loss encourages models to learn robust decision boundaries, leading to better generalization and performance on unseen data. The hinge loss is defined as:

$$L\big(y, f(x)\big) = \max\,(0, 1 - y.f(x)) \tag{9}$$

where y is true class label, *f(x)* is model output and *L(y,f(x))* is Hinge loss.

Accuracy (Acc)

The ratio of instances that were accurately classified by the classifier to the total instances.

$$\text{Acc} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{10}$$

False Negative Rate (FNR)

The ratio of real images that are mispredicted to be fake.

$$FNR = 1 - TPR \tag{11}$$

True Positive Rate (TPR) or Sensitivity

Also known as Recall, measures the proportion of actual real images that are correctly classified.

$$\text{TPR} = \frac{\text{TP}}{\text{TP+FN}} \tag{12}$$

Specificity (True Negative Rate, TNR)

The ratio of fake images that were correctly classified to be fake.

$$\text{TNR} = \frac{\text{TN}}{\text{FP+TN}} \tag{13}$$

False Positive Rate (FPR)

The ratio of fake images that are mistakenly classified as real.

$$FPR = 1 - TNR \tag{14}$$

F1-measure

The harmonic average of recall and precision.

$$F - measure = 2\frac{PPV * TPR}{PPV + TPR} \tag{15}$$

Negative Predictive Value (NPV)

The ratio of correctly classified fake images to the total images classified as fake.

$$NPV = \frac{TN}{FN + TN} \tag{16}$$

Precision

Also referred to as Positive Predictive Value (PPV is the ratio of correctly classified real images to the total images classified as real by the model. It's calculated as:

$$PPV = \frac{TP}{FP + TP} \tag{17}$$

Equal Error Rate (EER)

It illustrates the model's error rate. The model performs better when the value is lower.

Receiver Operating Characteristic (ROC) curve and Area under the ROC Curve (AUC)

The probability curve is shown by the ROC curve, and the quantity or amount of separateness is shown by the AUC value. The TPR and FPR fluctuate at various threshold values, as seen by the ROC curve.

AUC values close to one suggest that a well-designed model has strong separability. AUC score close to 0 denote the least separable model, which is also the poorest. Furthermore, an AUC score of 0.5 indicates that the algorithm is totally unable to distinguish between groups.

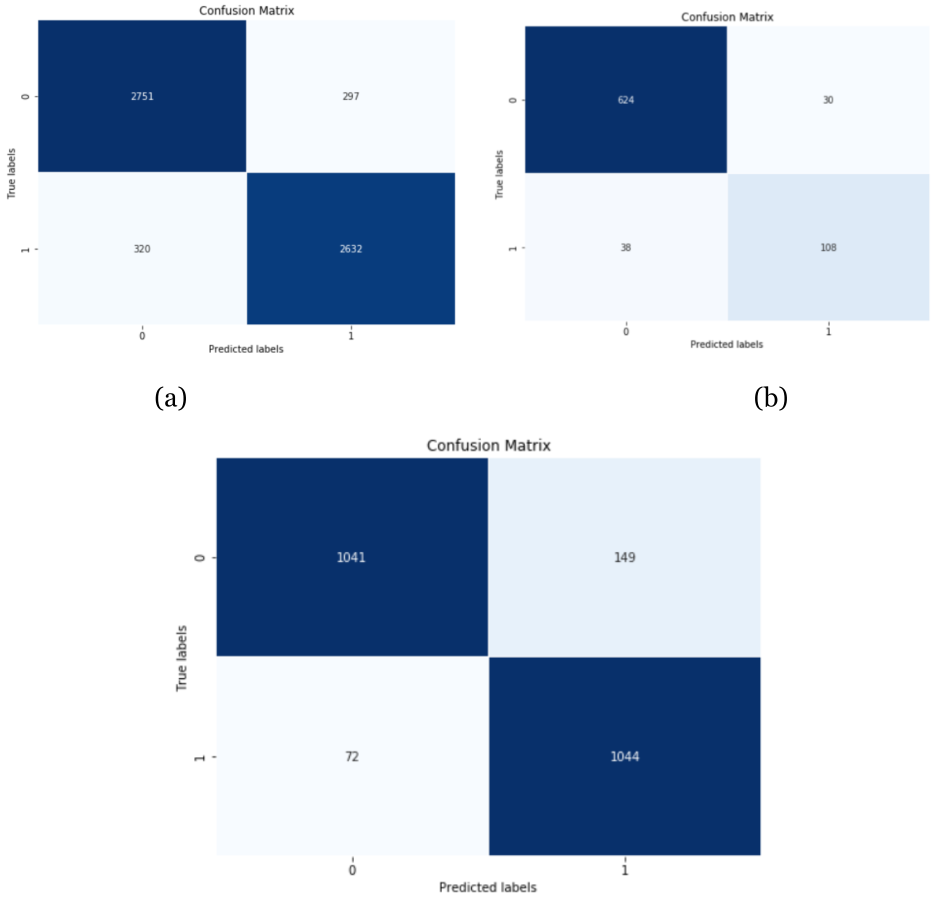
Experimental results analysis

The performance of MSFF-Net on three distinct datasets is presented in **Table 1**. The proposed MSFF-Net's performance on 140k Real and Fake Faces [25] Dataset demonstrates a solid overall accuracy of 89.72%, indicating its competence in distinguishing between genuine and fabricated facial images. The PPV and NPV of 89.86% and 89.58%, respectively, offer insights into the reliability of the model's performance. Although the values are relatively high, suggesting a generally correct classification, there is still potential for improvement, especially in reducing false classifications, which could enhance the MSFF-Net's effectiveness in real-world applications. However, its performance varies across different metrics. It shows a commendable true positive rate (TPR) of 89.16%, suggesting a strong ability to correctly identify fake faces. The proposed MSFF-Net's performance on Celeb-df [26] Dataset exhibits a commendable accuracy of 90.42%, indicating its proficiency in discerning between real and manipulated facial images. Upon closer examination, the dataset's metrics unveil both strengths and areas for improvement. The high true positive rate (TPR) of 93.55% suggests the model's robust capability in correctly identifying fake faces within the dataset. However, the slightly lower true negative rate (TNR) of 87.48% indicates a comparatively lower accuracy in recognizing genuine faces. **Figure 8** depicts the confusion matrix of MSFF-Net on three different datasets. The proposed MSFF-Net achieves an accuracy of 91.50% on DFDC [27] dataset. However, a deeper analysis of its metrics reveals certain strengths and weaknesses. The relatively lower true positive rate (TPR) of 73.97% suggests that the model occasionally struggles to correctly identify fake faces within the dataset.

Table 1. Performance of MSFF-Net on three different datasets (140Kreal and fake faces dataset, Celeb-df, and DFDC)

Metrics	140k real and fake faces	DFDC	Celeb-df
Accuracy	0.8972	0.9150	0.9042
TPR	0.8916	0.7397	0.9355
FPR	0.0974	0.0459	0.1252
TNR	0.9026	0.9541	0.8748
F1-measure	0.8951	0.7606	0.9043
FNR	0.1084	0.2603	0.0645
PPV	0.8986	0.7826	0.8751
NPV	0.8958	0.9426	0.9353
Hinge loss	0.3374	0.1966	0.2712
AUC score	0.9617	0.9671	0.9669
EER	0.1072	0.2143	0.1180

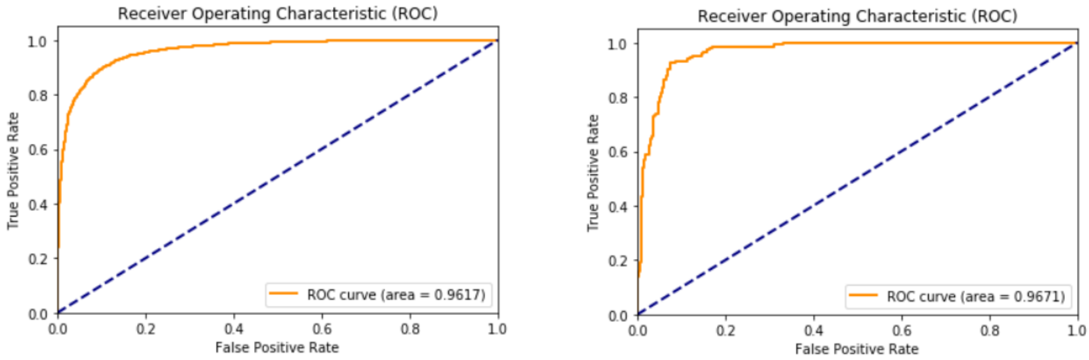
Figure 9 presents the AUC-ROC graphs of proposed MSFF-Net on 140k Real and Fake Faces [25], Celeb-df [26] and DFDC [27] datasets. These scores serve as indicators of the models' ability to distinguish between authentic and forged facial images. These scores suggest strong discriminatory power across all three datasets, indicating their effectiveness in correctly ranking positive and negative samples. A higher AUC score signifies superior performance in binary classification tasks, where distinguishing between genuine and forged faces is crucial. Thus, these results underscore the models' proficiency in accurately identifying manipulated facial images, thereby contributing to advancements in deepfake detection technology. On the 140k Real and Fake Faces [25] model achieves a slightly lower AUC score of 0.9617. The proposed MSFF-Net achieves the highest AUC value of 0.9671 on DFDC [27] data, followed closely by the Celeb-df [26] data with a score of 0.9669.



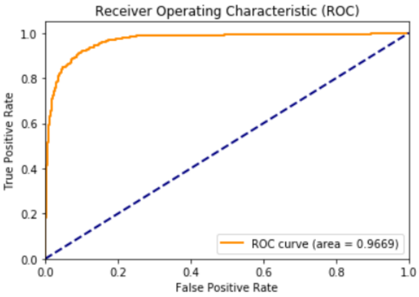
(a) (b)

(c)

Figure 8. Confusion Matrix, (a) for 140k real and fake faces dataset, (b) for DFDC dataset, (c) for celeb-df dataset



(a) (b)



(c)

Figure 9. ROC curve, (a) for 140K real and fake faces dataset, (b) DFDC dataset, (c) for Celeb-df dataset

Comparison with the SOTA models

Table 2 presents a comprehensive performance of MSFF-Net against several SOTA models on the Celeb-df [26] and DFDC [27] data. Across both datasets, MSFF-Net outperforms or closely rivals existing approaches on the basis of accuracy (ACC), AUC value, and Equal Error Rate (EER), indicating its effectiveness in detecting deepfake content. MSFF-Net demonstrates exceptional performance on the DFDC [27] dataset, attaining an accuracy of 91.5%, an AUC of 0.967, and an EER of 0.214. This highlights its superiority over the majority of other approaches. Similarly, on the Celeb-df [26] data, MSFF-Net demonstrates competitive performance with an accuracy score of 90.4%, an AUC score of 0.966, and an EER of 0.118. These outcomes highlight the efficacy of the proposed MSFF-Net in accurately discerning between genuine and forged facial images, highlighting its potential to make a substantial contribution to the development of deepfake detecting technologies.

Table 2. Comparison of MSFF-Net and the SOTA models on DFDC and celeb-df datasets

Models	**DFDC**			**CELEB-DF**		
	ACC	**AUC**	**EER**	**ACC**	**AUC**	**EER**
Xception(baseline) [40]	0.589	0.655	0.405	0.654	0.675	0.384
ForensicTransfer [41]	0.540	-	0.464	0.620	-	0.404
Multi-task [42]	0.511	-	0.494	0.584	-	0.511
MLDG [43]	0.607	0.682	0.370	0.595	0.609	0.418
LTW [44]	0.631	0.690	0.368	0.634	0.641	0.397
RECCE [45]	0.640	0.701	0.355	0.673	0.695	0.336
Fengkai Dong [46]	0.635	0.733	0.342	0.726	0.823	0.264
Proposed method (MSFF-Net)	0.915	0.967	0.214	0.904	0.966	0.118

Table 3 presents a performance of MSFF-Net against several SOTA models, with the evaluation criterion being accuracy. In the DFDC [27] dataset, the proposed MSFF-Net achieves an accuracy score of 0.915, while existing models such as Matern [47], Wu [48], Khalid [49], and Fung [50] achieve accuracy of 0.795, 0.904, 0.862, and 0.890, respectively. On the Celeb-DF [26] dataset, the proposed MSFF-Net attains an accuracy score of 0.904, while the other models achieve scores of 0.834, 0.940, 0.879, and 0.900. On celeb-df [26] dataset, Wu [48] achieves higher accuracy then the proposed MSFF-Net. These findings highlight the performance of the proposed MSFF-Net against existing approaches, highlighting its strengths and areas for improvement in accurately classifying facial images as authentic or manipulated.

Table 3. Comparison of the MSFF-Net and the SOTA models. The evaluation indicator is the Accuracy.

Models	**DFDC**	**CELEB-DF**
Matern [47]	0.795	0.834
Wu [48]	0.904	0.940
Khalid [49]	0.862	0.879

Fung [50]	0.890	0.900
Proposed method (MSFF-Net)	0.915	0.904

Table 4 illustrates a comparative analysis of the MSFF-Net alongside various SOTA models, focusing on the evaluation metric of the AUC score. For the DFDC [27] dataset, MSFF-Net achieves an AUC value of 96.71%. In contrast, existing models such as Xception [51], ProtoPNet [52], DPNet [53] score 91.27%, 84.46%, 92.44% respectively. However, some models have unspecified values. For the Celeb-DF [26] dataset, MSFF-Net achieves an AUC score of 96.69%. In comparison, other models like F3-net [54], Multi-attentional Detection [56] score 65.17%, 67.44% respectively. These findings highlight the comparative performance of the proposed MSFF-Net against existing approaches, emphasizing its effectiveness in discriminating between genuine and forged facial images across different data.

Table 4. Comparison of the MSFF-Net and the SOTA models. The evaluation indicator is the AUC score.

Models	DFDC	CELEB-DF
Xception [51]	91.27%	65.50%
ProtoPNet [52]	84.46%	69.33%
DPNet [53]	92.44%	68.20%
SPSL [54]	-	76.88%
F3 – net [55]	-	65.17%
Multi-attentional Detection [56]	-	67.44%
Proposed method (MSFF-Net)	96.71%	96.69%

DISCUSSION

Results shown in Table 2 demonstrate that MSFF-Net performs significantly better than SOTA models. Hybrid-learning feature extractors and HOG's respective benefits are merged in the proposed MSFF-Net. Classification accuracy is enhanced through the combination of deep learning feature analysis and texture analysis. Additional results are presented in Table 1 for the performance of MSFF-Net on various datasets. The main advantages of MSFF-Net are:

- By combining texture and deep features, MSFF-Net enhanced the performance.
- MSFF-Net is easily implementable and outperforms existing SOTA models.
- Across various datasets, MSFF-Net shows impressive results.

The MSFF-Net can be implemented in practical scenarios as a highly competitive approach, based on its prior benefits. The reduction in manipulation artifacts may, nevertheless, compromise the accuracy of detection due to the stunning and continuous improvements in deepfake generation techniques. Furthermore, it's important to mention that our proposed method doesn't rely on temporal features, which may also affect the detection accuracy.

CONCLUSION

Recent intrusions by DeepFake software and tools into social media and mobile devices caused mistrust of visual content. It became effortless to fabricate news to misrepresent politicians, actors, or anyone else. This results in instability, which requires an immediate investigation into the visual data. In this paper, a fusion method is used to fuse learned texture features, with deep features to increase performance. The texture and shape features of the images have been extracted using the Histogram of Oriented Gradients. And for deep features Resnet50 model is used. Then concatenate the features extracted from two different spaces. After concatenating the features PCA is employed to identify and choose the significant features and to minimize the feature vector because the number of features were relatively high.

The experimental results on various feature extraction approaches demonstrate that the proposed fusion network outperforms other SOTA techniques. In the future, the goal is to include both spatial and temporal content, as well as audio content, which can enhance the performace of detection models.

Data availability

The datasets are available in the Kaggle repositories: https://www.kaggle.com/competitions/ deepfake-detection-challenge and https://www.kaggle.com/datasets/xhlulu/140k-real-and-fake-faces.

CONFLICT OF INTEREST

The authors declare there is no conflict of interest.

REFERENCES

[1] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. Advances in neural information processing systems, vol. 27.

[2] Kingma DP, Welling M (2014) Stochastic gradient vb and the variational auto-encoder. In: Second international conference on learning representations, ICLR, vol 19, p 121

[3] Karras T, Laine S, Aila T (2019) A style-based generator architecture for generative adversarial networks. In: proceedings of the IEEE/CVF Conference on computer vision and pattern recognition, pp 4401–4410

[4] Choi Y, Choi M, Kim M, Ha J-W, Kim S, Choo J (2018) Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In: proceedings of the IEEE conference on computer vision and pattern recognition, pp 8789–8797 https://doi.org/10.1109/CVPR.2018.00916.

[5] Shen Y, Yang C, Tang X, Zhou B (2022) Interfacegan: Interpreting the disentangled face representation learned by gans, IEEE Transactions on Pattern Analysis and Machine Intelligence. vol. 44.

[6] Dolhansky B, Bitton J, Pflaum B, Lu J, Howes R, Wang M, Ferrer CC (2020) The deepfake detection challenge dataset, arXivpreprint arXiv arXiv:2006.07397

[7] Seferbekov S (2020) https://github.com/selimsef/dfdc deepfake challenge. Accessed 24 Jan 2022

[8] Elaskily MA, Elnemr HA, Dessouky MM, Faragallah OS (2019) Two stages object recognition based copy-move forgery detection algorithm. Multimedia Tools and Applications, vol. 78(11), pp.15353–15373. https://doi.org/10. 1007/s11042-018-6891-7

[9] Fadl SM, Semary NA (2017) Robust copy-move forgery revealing in digital image using polar coordinate system. Neuro computing vol. 265, pp.57–65. https://doi.org/10.1016/j.neucom.2016.11.091

[10] Pun C-M, Liu B, Yuan X-C (2016) Multi-scale noise estimation for image splicing forgery detection. journal of Visual Communication and Image Representation vol. 38, pp.195–206. https://doi.org/10.1016/j.jvcir.2016.03.005

[11] Zhang Q, Lu W, Weng J (2016) Joint image splicing detection in dct and contourlet transform domain. Journal of Visual Communication and Image Representation vol. 40, pp.449–458. https://doi.org/10.1016/j.jvcir.2016/07.013

[12] Bakas J, Naskar R, Dixit R (2019) Detection and localization of inter-frame video forgeries based on inconsistency in correlation distribution between haralick coded frames. Multimedia Tools and Application vol. 78(4), pp.4905–4935. https://doi.org/10.1007/s11042-018-6570-8

[13] K Sitara, Mehtre BM (2018) Detection of inter-frame forgeries in digital videos. Forensic Science International vol. 289, pp.186 206. https://doi.org/10.1016/j.forsciint.2018.04.056

[14] Fadl S, Han Q, Qiong L (2020) Exposing video inter-frame forgery via histogram of oriented gradients and motion energy image. Multidimensional Syststem and Signal Processing, pp.1–20

[15] Kumar P, Vatsa M, Singh R (2020) Detecting face2face facial reenactment in videos. In: The IEEE Winter Conference on Applications of Computer Vision (WACV). DOI:10.1109/WACV45572.2020.9093435

[16] Rossler A, Cozzolino D, Verdoliva L, Riess C, Thies J, Niesner M (2019) Faceforensics++: Learning to detect manipulated facial images. In: Proceedings of the IEEE International Conference on Computer Vision, p. 1–11. **DOI:** 10.1109/ICCV.2019.00009

[17] Zhou P, Han X, Morariu VI, Davis LS (2017) Two-stream neural networks for tampered face detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), p. 1831 1839. IEEE. **DOI:** 10.1109/CVPRW.2017.229

[18] Juefei-Xu F, Wang R, Huang Y, Guo Q, Ma L, Liu Y (2021) Countering malicious deepfakes: Survey, battleground, and horizon. arXiv:2103.00218.

[19] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2020) Generative adversarial networks. Communications of the ACM vol. 63(11), pp.139–144, https://doi.org/10.1145/3422622

[20] Fernando T, Fookes C, Denman S, Sridharan S(2020) Detection of fake and fraudulent faces via neural memory networks. IEEE Transactions on Information Forensics and Security. vol.16, pp.1973–1988, https://doi.org/10.1109/TIFS.2020.3047768

[21] Kong C, Chen B, Li H, Wang S, Rocha A, Kwong S (2022) Detect and locate: exposing face manipulation by semantic-and noise-level telltales. IEEE Transactions on Information Forensics and Security. vol. 17, pp.1741–1756, https://doi.org/10.1109/TIFS.2022.3169921

[22] Yang J, Li A, Xiao S, Lu W, Gao X(2021) MTD-Net: learning to detect deepfakes images by multi-scale texture difference. IEEE Transactions on Information Forensics and Security. vol. 16, pp.4234–4245, **DOI:** 10.1109/TIFS.2021.3102487

[23] Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol 1, pp 886–893. IEEE. https://doi.org/10.1109/CVPR.2005.177

[24] He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, https://doi.org/10.1109/CVPR.2016.90.

[25] NVlabs. NVlabs/ffhq-dataset: Flickr-faces-HQ dataset (FFHQ), (2019). Available at: https://archive.org/details/ffhq-dataset

[26] Li Y, Yang X, Sun P, Qi H, Lyu S (2020) Celeb-df: A large-scale challenging dataset for deepfake forensics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, p. 3207–3216, https://doi.org/10.1109/CVPR.42600.2020.003270

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

[28] Korshunov P, Marcel S, (2018) Deepfakes: A new threat to face recognition? Assessment and detection. Cornell University.

[29] Rafique R, Nawaz M, Kibriya H, Masood M (2021) DeepFake detection using error level analysis and deep learning. In 4th International Conference on Computing and Information Sciences (ICCIS) doi.org/10.1109/ICCIS54243.2021.9676375

[30] Ismail A, Elpeltagy M, Zaki MS, Eldahshan K (2021) A New Deep Learning-Based Methodology for Video Deepfake Detection Using XGBoost, Sensors 21, vol.21, pp.5413. doi.org/10.3390/s21165413

[31] Masood M, Nawaz M, Javed A, Nazir T, Mehmood A, & Mahum R (2021) Classification of deepfake videos using pre-trained convolutional neural networks, International Conference on Digital Futures and Transformative Technologies (ICoDT2) doi.org/10.1109/ICoDT252288.2021.9441519

[32] Taeb M, Chi H, (2022) Comparison of deepfake detection techniques through deep learning, Cybersecurity and Privacy, vol.2, pp.89–106. https://doi.org/10.3390/jcp2010007

[33] Deng L, Suo H, Li D (2022) Deepfake video detection based on EfficientNet-V2 network. Computational Intelligence and Neuroscience, article ID 3441549, 13 pages. https://doi.org/10.1155/2022/3441549

[34] Rafique R, Gantassi R, Amin R, Frnda J, Mustapha A, Alshehri A H (2023) Deep fake detection and classification using error-level analysis and deep learning. Scientific Reports. vol. 13, pp.7422 (2023). https://doi.org/10.1038/s41598-023-34629-3

[35] Ismail A, Elpeltagy M, Zaki M S, Eldahshan K (2022) An integrated spatiotemporal-based methodology for deepfake detection. Neural Computing and Applications. vol.34(24), pp.21777–21791. https://doi.org/10.1007/s00521-022-07633-3

[36] Zhang K, Zhang Z, Li Z, Qiao Y (2016) Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks. IEEE Signal Processing Letters, vol.23(10), pp.1499-1503, DOI: 10.1109/LSP.2016.2603342.

[37] Raveena, Punyani P, Chhikara R (2023) Comparision of different Machine Learning algorithms for Deepfake Detection. International Conference on Communication, Security and Artificial Intelligence (ICCSAI), IEEE publications, Greater Noida, India, pp. 58-63, https://doi.org/10.1109/ICCSAI59793.2023.10421164.

[38] Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol 1, pp 886–893, DOI: 10.1109/CVPR.2005.177

[39] Sandhya N, Charanjeet K R (2016) A review on machine learning techniques, International Journal of Recent Innovation and Trends in Computing and Communication vol. 4.3, pp.451–8.

[40] Rossler A, Cozzolino D, Verdoliva L, Riess C, Thies J, Niessner M (2019) FaceForensics++: Learning to Detect Manipulated Facial Images. In: 2019 IEEE/ CVF International Conference on Computer Vision (ICCV). IEEE, Seoul, Korea (South, pp. 1–11. https://ieeexplore.ieee.org/document/9010912/.)

[41] Cozzolino D, Thies J, Rössler A, Riess C, Nießner M, Verdoliva L (2018) Forensictransfer: Weakly-supervised domain adaptation for forgery detection, arXiv preprint arXiv:1812.02510.

[42] Nguyen H H, Fang F, Yamagishi J, Echizen I (2019) 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), pp. 1–8, iSSN: 2474-9699. https://doi.org/10.1109/BTAS46853.2019.9185974.

[43] Li D, Yang Y, Song Y Z, Hospedales T (2018) Learning to generalize: meta learning for domain generalization. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32, no. 1, https://doi.org/10.1609/aaai.v32i1.11596

[44] Sun K, Liu H, Ye Q, Gao Y, Liu J, Shao L, Ji R (2021) Domain general face forgery detection by learning to weight. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35(3), pp. 2638–2646, https://doi.org/10.1609/aaai.v35i3.16367.

[45] Cao J, Ma C, Yao T, Chen S, Ding S, Yang X (2022) End-to-end reconstruction classification learning for face forgery detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4113–4122.

[46] Dong F, Zou X, Wang J (2023) Contrastive learning-based general Deepfake detection with multi-scale RGB frequency clues. In: Journal of King Saud University– Computer and Information Sciences vol.35, pp.90–99, https://doi.org/10.1016/j.jksuci.2023.03.023

[47] Matern F, Riess C, Stamminger M (2019) Exploiting visual artifacts to expose deepfakes and face manipulations. In: 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW), p.83–92. https://doi.org/10.1109/WACVW.2019.00020

[48] Wu X, Xie Z, Gao Y, Xiao Y (2020) Sstnet: Detecting manipulated faces through spatial, steganalysis and temporal features. In: ICASSP 2020- 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), p. 2952–2956. IEEE, **DOI:** 10.1109/ICASSP40776.2020.9053969

[49] Khalid H, Woo S S (2020) Oc-fakedect: Classifying deepfakes using one-class variational autoencoder. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, p. 656–657

[50] Fung S, Lu X, Zhang C, Li C T (2021) Deepfakeucl: Deepfake detection via unsupervised contrastive learning. In: 2021 International Joint Conference on Neural Networks (IJCNN), p. 1–8. IEEE

[51] Chollet F (2019) Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, pp. 1251–1258.

[52] Chen C, Li O, Tao D, Barnett A, Rudin C, Su J K (2019) This looks like that: deep learning for interpretable image recognition. Advances in Neural Information Processing Systems, vol. 32, pp. 8930–8941.

[53] Trinh L, Tsang M, Rambhatla S, Liu Y (2021) Interpretable and trustworthy deepfake detection via dynamic prototypes. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1973–1983. https://doi.org/10.48550/arXiv.2006.15473

[54] Liu H, Li X, Zhou W, Chen Y, He Y, Xue H, Zhang W, Yu N (2021) Spatial-phase shallow learning: Rethinking face forgery detection in frequency domain. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 772–781.

[55] Qian Y, Yin G, Sheng L, Chen Z, Shao J (2020) Thinking in frequency: Face forgery detection by mining frequency-aware clues. In Proceedings of the European Conference on Computer Vision. Springer, pp. 86–103.4

[56] Zhao H, Zhou W, Chen D, Wei T, Zhang W, Yu N (2021) Multi-attentional deepfake detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2185–2194.