Multimedia Tools and Applications
https://doi.org/10.1007/s11042-025-20865-4

Detecting face tampering in videos using deepfake forensics

Mohammad Qawasmi[1] · **Omar Al-Kadi**[1]

Received: 14 August 2023 / Revised: 11 April 2025 / Accepted: 19 April 2025
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2025

Abstract
Deepfake technology, enabled by generative adversarial networks, creates fake video content with significant societal impacts, such as electoral bias and celebrity defamation. This research aims to develop an automatic and effective facial tampering detection model for videos, regardless of their quality. The proposed two-fold approach enhances facial frame features using image aggregation and detects feature anomalies through a neural network with convolutions and fully connected layers. The paper introduces image frame quality enhancement and different activation functions to improve performance, achieving up to 98% accuracy with the hybrid model. Additionally, the work explores the detection of deep fakes in low-quality frames by reducing frame quality by 20% and 50%. The hybrid model achieves an accuracy of up to 96% in the first suboptimal condition and up to 93.7% in the second quality reduction condition. This research presents a promising method for deepfake detection, with the potential to mitigate its negative social impact.

Keywords Deepfake · Mesonet · Frame aggregation · Facial expression · Video tampering

1 Introduction

In contemporary times, photographs and videos can be captured and transmitted rapidly across the globe. Consequently, people have become increasingly dependent on these media to determine the authenticity of an event. In particular, "live" accounts captured on video have become newsworthy in both criminal and civil cases. Unfortunately, such videos have the potential to go viral on social media, especially when they are fake. This overreliance on visual evidence can have devastating consequences, as viewers often accept what they see and hear at face value, placing the lives of those involved in precarious situations. Hence, it is crucial to exercise caution when assessing visual evidence to ensure accuracy and impartiality.

The increasing prevalence of deepfakes and social media platforms has created a need for the public to discern between real and fake videos and images. However, identifying deepfakes is a difficult task that requires significant effort. This poses a significant threat to personal identity, national security, and reputation, among other things. Although deepfake technology has been around since the 1990s, it has only become a major concern since 2017, when facial manipulation of celebrities was used for blackmail [2]. Deepfake content

✉ Omar Al-Kadi
o.alkadi@ju.edu.jo

[1] King Abdullah II School of Information Technology, University of Jordan, Amman 11942, Jordan

Published online: 26 April 2025

creators have the power to manipulate images and videos for malicious purposes, posing a threat to individuals' reputation and national security. The ethical implications of deepfakes are vast, including concerns about political polarization, national security, economic impact, pornography, and social media platforms.

Political polarization and deepfakes have created a damaging divide in society, increasing the risk of mistrust in the government and its leaders. Disinformation can also manipulate voters during elections, leading to political unrest and a loss of trust in government officials. Economic problems can also arise from false facts, such as the potential impact on global markets due to misleading information. Social networks play a significant role in the spread of deepfake technology. This poses a threat to the integrity of the media and important personalities around the world. Solutions are needed to prevent deepfakes from continuing to harm public knowledge and create irreparable situations. The evolution of technology in video and photo editing has undergone a remarkable evolution, transitioning from manual interventions to the emergence of sophisticated algorithms such as Generative Adversarial Networks [1]. These advanced algorithms have enabled modifications that are nearly imperceptible to the human eye. As these capabilities advance, the need for robust counterfeit detection algorithms capable of identifying deepfake vulnerabilities becomes increasingly pronounced. This paper directs its attention toward the technical intricacies of both deepfake generation and detection, focusing explicitly on the extraction of a sequence of facial frames at varying resolutions. This process facilitates the creation of diverse facial expressions, enhancing the overall video quality. To achieve this, frame aggregation [53] is combined with a Mesoscale Network [44], adapted to identify subtle manipulated facial expressions. This hybrid methodology allows for the capture of mid-to-low-frequency patterns, the realm where the most delicate manipulative artifacts tend to manifest. The efficacy of the proposed hybrid model is evaluated using the Forensics ++ face data set [11], a widely acknowledged benchmark in the field. The model's performance is further benchmarked against recent comparable works. In order to evaluate the influence of activation functions on network performance, a variety of functions are employed. Moreover, the model's competence in handling low-quality frames is evaluated by deliberately reducing frame quality.

The paper is structured as follows: In the second Section, we examine prominent face manipulation and deepfake techniques. In Section 3, we present our proposed deface detection approach. In Section 4, we discuss our experimental findings and provide analysis. Finally, we conclude the paper in Section 5.

2 Related work

Facial manipulations can be classified into four primary categories based on the extent of the manipulation. This section presents a review of the relevant literature in order of prevalence, from the most common to the least common.

2.1 Identity swap

The first type of facial manipulation is identity swap, in which one person's face is replaced with another in a video. Several approaches have been proposed to detect deepfakes using different features and techniques. Korshunov and Marcel [21] used MelFrequency Cepstral Coefficients and distances between mouth landmarks to detect deepfakes. They reduced the dimensions of feature blocks using Principal Component Analysis and used an RNN based on

the LSTM model for classification. Matern et al. [19] suggested a deepfake detection system based on visual aspects, such as the color of the eyes, reflections, and missing details in the eyes and teeth. They used a logistic regression model and a Multilayer Perceptron (MLP) [26] and achieved 85.1% AUC for the MLP system. Xin Yang et al. [22] used facial expressions and head movements to detect deepfakes. They distinguished fake from real videos based on differences in head positions and used SVM for classification. Their approach achieved 89.0% AUC using the UADFV database, but performed less well on other databases such as the FF++ database (78.0%) and the DFDC database (66.2%). Agarwal and Farid [23] used the OpenFace2 toolkit for feature extraction and examined 18 different units of facial parts related to facial muscle movement, as well as four features related to head movement. They used Pearson's correlation to measure the linearity between features and SVM for classification. They created their own database based on YouTube videos for interested people talking in an official context and achieved an AUC of 96.3% for fake video detection. However, the model encountered some difficulties in detecting fake videos in which the target person did not look directly at the camera, so the authors suggested including linguistic analysis to capture correlations between what is said and how it is said.

2.2 Entire face synthesis

The entire face synthesis involves creating non-existent face images using powerful GANs. To detect the difference between real and fake images, some researchers have suggested analyzing the internal GAN. For example, Wang et al. monitored the behavior of neurons in the GAN to develop a tool for detecting fake faces [8]. By examining the patterns of cell activation layer by layer, they were able to identify key features that are important for the detection of facial manipulation. Their proposed approach, called FakeSpotter, extracts neuro cell cover behaviors for real and fake faces from deep fake recognition systems such as VGG-Face [9], OpenFace [30], and FaceNet [31], and trains an SVM for final classification. Testing real faces from CelebA-HQ databases [32] and FFHQ [33], and fake faces created using InterFaceGAN [34] and StyleGAN [33], they achieved a fake detection accuracy of 84.7% using the FaceNet model. However, FakeSpotter has limitations as it does not perform as well in detecting deep fake videos that involve both facial manipulation and voice swapping.

Other researchers have proposed alternative approaches to detect fake faces. For example, McCloskey and Albright developed a color-based detection system based on the observation that real camera images and fake synthesis images differ in color [28]. Their system transforms a multichannel feature map into a 3-channel color image and uses a linear SVM for the final classification. Using the MFC2018 dataset [29], they achieved an AUC of 70.0%. However, this approach requires a large training dataset and training time. Furthermore, Guarnera et al. suggested a fake detection system based on the analysis of convolutional effects [35]. They extracted features using the Expectation-Maximization algorithm [36] and used common classifiers such as SVM, Linear Discriminant Analysis (LDA) and k-Nearest Neighbors (k-NN) for final detection. Their proposed approach achieved a final precision of 99. 81% when tested on fake images created using AttGAN [37], GDWCT [38], StarGAN [39], StyleGAN and StyleGAN2 [40]. However, this model is limited to detecting fake images generated from specific types of GANs. However, its limitations include the need for costly thermal cameras, rare thermal data in videos, environmental sensitivity, and integration challenges with existing detection systems.

2.3 Attribute manipulation

The process of attribute manipulation involves adjusting certain facial features, including hair color, skin, sex, age, and glasses, among others. Researchers have modified the internal pipeline of GAN to distinguish between real and manipulated images. By monitoring the behavior of neurons, Wang et al. proposed that patterns of cell activation layer after layer could help detect counterfeit faces, capturing more accurate and important features of the facial manipulation detection system. Their proposed approach, called FakeSpotter, extracts neural cell cover behaviors for real and fake faces from deep facial recognition systems such as VGG-Face, OpenFace, and FaceNet. The model is then trained using an SVM for the final classification. The authors achieved the best performance using the FaceNet model, with a detection accuracy of 84.7% for manipulation. The authors trained their proposed approach using real faces from CelebA-HQ databases and FFHQ databases, as well as fake faces generated by InterFaceGAN and StyleGAN.

However, the FakeSpotter approach focuses only on facial images without any regard for sound, whereas the DeepFake Detection Challenge includes two different fake domains, face exchange, and voice swapping. Many studies have also focused on purely deep learning methods to improve accuracy. For instance, Bharati et al. proposed a deep learning approach based on a Boltzmann restricted machine to discover the digital revision of facial images. Their approach achieved an accuracy of 87.1% for manipulation detection of celebrity and ND-IIITD retouching databases. However, this approach cannot determine whether the image is real or fake if there is facial makeup on the target person.

More recently, Jain et al. proposed a similar approach based on nonoverlapping face patches. Their model consists of a CNN feature extractor of 6 convolutional layers and 2 fully connected hidden layers, with the remaining links inspired by the ResNet structure. They achieved almost 100% detection accuracy training their model on the modified ND-IIITD database provided in a previous study and the fake images created using the StarGAN approach and using an SVM for the final classification. However, like Bharati et al.'s approach, this model cannot distinguish between real and fake images with facial makeup on the target person.

2.4 Expression swap

Facial reenactment, also known as expression swap, involves modifying a person's facial expressions. Nguyen et al. [12] proposed a multitask learning approach for the detection of expression swaps, evaluated on the FaceForensics++ database [11]. Their method achieved a 7.1% equal error rate (EER) for Face2Face and a slightly higher 7.8% EER for neuralTexture. The authors used an autoencoder to estimate the final classification and incorporated the attention mechanisms of [3] to improve the training process. To test their proposed discovery policy, they used the DFFD database, which is based solely on data from the FaceForensics++ database. Their approach obtained an area under the curve (AUC) of 99.4% and an EER of 3.4%. However, their model has limitations in detecting high-quality images and audio attacks. Sabir et al. [14] proposed an approach based on recurrent convolutional networks that considers both image and time information. They achieved 94.3% AUC for Face2Face technology in the FaceForensics++ database, but only analyzed low-quality videos. Another proposed approach, which uses optical flow areas to exploit the potential differences between genuine and fake videos, uses the PWC-Net approach [20]. The use of optical flow is motivated by the abnormal visual flow present in fake videos due to the unusual movement of

lips, eyes, etc. This approach achieved preliminary accuracy results of 81. 6% using both the VGG16 and ResNet50 networks for tamper detection, but it is also weak against high-quality images.

Modern deep learning methods have shown good results in detection systems for both Face2Face and neuralTextures. Rossler et al. [11] proposed an XceptionNet-based detection system that achieved 100% near-RAW quality for both manipulations. The evaluation of detection systems also considered different video quality levels to simulate video processing on social networks. In this real scenario, the accuracy of all detection systems is reduced with the video quality, similar to what occurs in identity swap manipulation. The four different categories of facial manipulation and the main associated detection characteristics are illustrated in Fig. 1.

This research paper aims to investigate face-swapping in low-resolution frame image quality and explore improvements using different activation functions. Face-swapping is currently one of the most prevalent challenges on social media and presents an easy avenue to spread fake news about prominent individuals.

3 Methodology

This section outlines effective ways to tackle the problem of Deepfake or Face2Face. It has been found that a single network is not sufficient to efficiently solve these problems. However, due to the similar nature of the counterfeiting techniques used, using identical network structures for both can lead to good results.

Our proposed method involves detecting fake face videos by analyzing them at medium- and low-quality levels. We utilize an image aggregation algorithm to improve image quality, as detecting fake images based on image noise in videos can be challenging. Similarly, at a higher semantic level, it can be difficult for the human eye to distinguish fake images, especially when the image depicts a human face [42, 43]. To overcome this, we suggest adopting a hybrid approach using a deep neural network with a few layers.

Our approach aims to achieve the best performance with a low representation level and a low number of parameters. We use well-established image classification networks [16, 24],

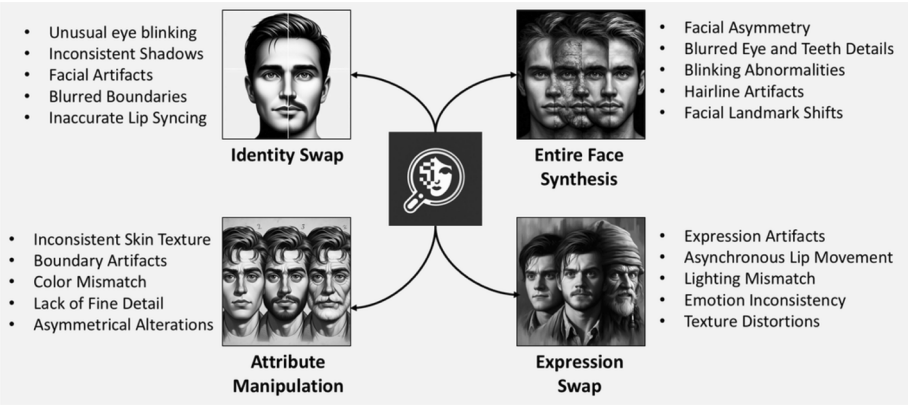

Fig. 1 Deepfake facial tampering categories and main facial characteristics and visual cues

which alternate between convolutional and pooling layers to extract features, and a dense classification network.

3.1 Image aggregation

Due to the collaborative nature of information processing, collaborative filtering strategies are employed for frame denoising [43]. Block-matching and 3D collaborative filtering (BM3D) denoising algorithms are used for this purpose. BM3D combines 2D fragments to create 3D data lines called GROUPS. By filtering GROUPS, a three-dimensional estimate is obtained that includes many pixel estimates. After that, data and information are collected to form the final denoised frame. The quality of denoised frames is improved by adopting local structure treatments. It is also proposed that a BM3D-oriented algorithm can use non-local filtering models of different types, in both edge and smooth forms.

This research employs a denoising frame approach, the combined regression alternative (COBRA) algorithm [43]. The approach combines different classical denoising methods, and its effectiveness may vary depending on the noise type and frame pixels. Figure 2 illustrates the general model of COBRA. The COBRA algorithm is used because it employs many different classical noise reduction methods to obtain numerous pixel predictions for noise reduction. These values are then combined to produce a new noise reduction in the best possible way. Since each classical method has its own advantages and disadvantages, and its efficiency varies depending on the noise type or frame structure, the COBRA approach aims to leverage the strengths of each method to improve denoising performance.

The COBRA algorithm was utilized to reduce the noise in frames. For each pixel in the blurred frame X, multiple estimators were called and then aggregated using a weighted average, as shown in (1):

$$f(p) = \frac{\sum_{q \in X} w(p,q)x(q)}{\sum_{q \in X} w(p,q)} \tag{1}$$

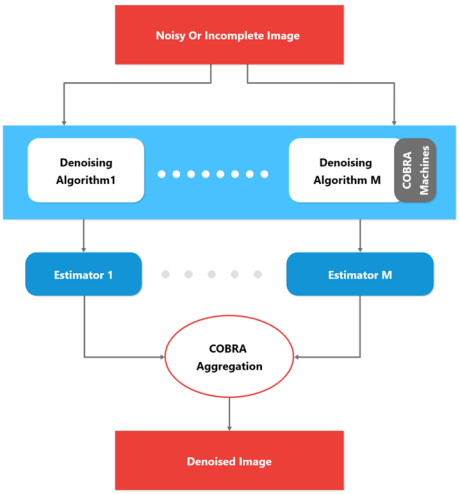

Fig. 2 General model of combined regression alternative approach (COBRA)

The weights are defined in (2) as

$$w(p,q) = \sum_{k=1}^{m} [|f_k(p) - f_k(q)| \leq \epsilon] \geq M\alpha \tag{2}$$

Here, ϵ and $\alpha \in (0, 1)$ were the confidence and proportion parameters, respectively. These weights ensured that to reduce the noise in the p pixels, the COBRA algorithm classified the intensity of each pixel as at least the ratio α, from the initial capabilities of $f_1 \ldots f_m$, with the same value in p and q, up to a confidence level of ϵ. The intensity of pixels based on these weights, including the consensus scale, is calculated in (2). The predicted intensity for each pixel p of the frame was $f(p)$, and the COBRA-denoised frame was the collection of pixels $f(p)$, $p \in X$.

Algorithm 1 follows the general scheme shown in Fig. 2, allowing users to control the number of features used in the frame patch. For each pixel p to denoise, frame correction was considered, centered on p, of size $(2 \times \text{patch size} + 1)^2$. In the trials section, a patch size of 1 was typically a satisfactory value. Thus, for each pixel, a vector was built up of nine features, each of which reduced the noise of the frames differently.

Algorithm 1 Image aggregation algorithm.

Data: Noise image
Result: y ← Denoise image
$Xtrain \leftarrow$ training images with artificial noise
$ytrain \leftarrow$ original training images (ground truth)
$Xtest \leftarrow$ feature extraction from Data noise in a vector of size $(nb\_pixels, (2 \times psize + 1)^2)$
$ytest \leftarrow s$ prediction of Xtest by COBRA
$Psize \leftarrow$ the pixel patch size to consider M = the number of COBRA machines to use
$Estemator \leftarrow [\,]$
while $Data \leq Psize$ **do**

$$f(p) \leftarrow \frac{\sum_{q \in x} w(p,q) \times (q)}{\sum_{q \in x} w(p,q)} \tag{3}$$

$$w(p,q) = \left(\sum_{k=1}^{m} |f_k(p) - f_k(q)| \leq \epsilon \right) \geq M\alpha \tag{4}$$

```
; /* F(p) to aggregate these estimators by doing weighted average
on the intensities & w([p,q]) to define the weights */
```

end while
← Assemble all the estimators that, in turn, modified all poor pixels.

3.2 MesoNet architecture

MesoNet is a deep learning architecture for detecting facial manipulation in images, specifically focusing on detecting manipulation at a mesoscopic (medium-scale) level, such as the modification of facial features [44]. It involves a series of convolutional and pooling layers to extract features from the input image, followed by several fully connected layers for classification.

In the networks' process, the video is first segmented into frames, and then faces are detected and cropped before being saved. To improve the features in the frames, the image

aggregation algorithm is used by passing the three dimensions of the frame data (255, 255, and 3), as shown in Fig. 3. The frames are then fed into the network, which identifies fake frames from real ones through the following steps: In the first step, the model parameters are set for training. The relevant parameters include fixed input frames of size (256, 256, 3), a batch normalization size of 64, 200 training epochs, a max pooling layer, a learning rate of 0.00001, and a decay rate of 0.00005. The loss function used is cross-entropy, and Stochastic Gradient Descent is employed for optimization. During training, the shuffle variable is set to True, and the experiment's performance is evaluated using ACC. In the second step, feature extraction is performed using four convolutional neural network blocks, each consisting of three methods: a convolutional layer, a batch normalization layer, and a max pooling layer. The convolution layer sets the size and number of filters used for convolution, with each filter representing a different feature of the frame. ReLU functions are used to prevent the exponential growth of computations required to run the neural network, and to avoid the vanishing gradient problem. Batch normalization improves the speed, performance, and stability of the neural network, while max pooling reduces the dimensionality of the data.

In the third step, Dropouts are used to regularize and improve the robustness of the fully connected layers. In the fourth step, a sigmoid function is applied, which can directly handle outputs ranging from 0 to 1. In the fifth step, the prediction is evaluated. MesoNet's prediction is accurate when the actual label corresponds to MesoNet's predicted output after rounding to 0 or 1.

Finally, in the sixth step, a deepfake video program is developed using an online executable IDE. For more information, see Fig. 3.

3.3 Detecting tampering

To effectively solve the classification problems caused by deepfake images using deep learning networks, an approach involves interpreting the weights of the convolutional kernel and neurons as descriptions of frames. However, this method provides limited information in the case of facial images and applies only to the first layer. Instead, generating an input frame that selectively activates a specific filter and observing the resulting signal interactions can be

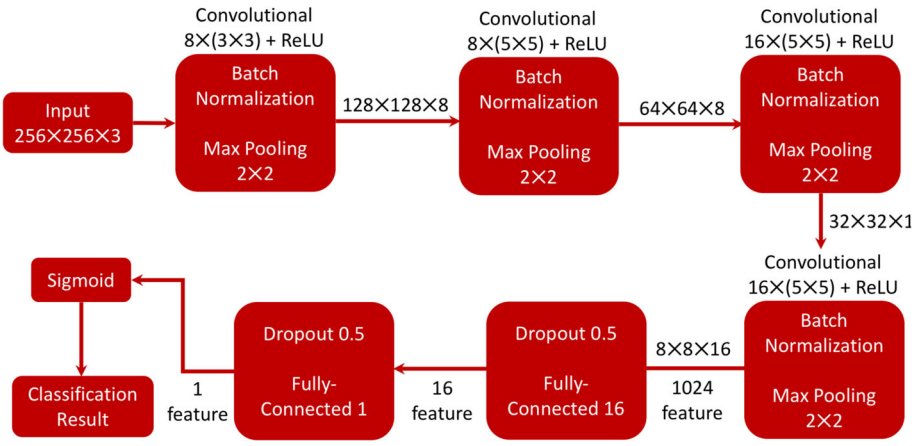

Fig. 3 The network architecture of Meso-4. Layers and parameters are displayed in the boxes, output sizes next to the arrows

employed. To do this, we use a maximization approach expressed by $E(x) = f_{ij}(x) - \lambda ||x||p$, where f_{ij} represents the output filter for each layer j, and incorporate input regulation to reduce noise.

Figure 4 shows the maximum activation of several neurons in the four convolution layers of MesoNet. These neurons are classified according to the weight mark applied to their product, which facilitates classification decisions. By calculating whether the activation pushes towards a negative score (i.e. counterfeit category) or a positive score (i.e., real class), we can determine the authenticity of the image. Interestingly, positive weighting neurons tend to display intricate details of the eyes, nose, and mouth, while passive weight areas exhibit

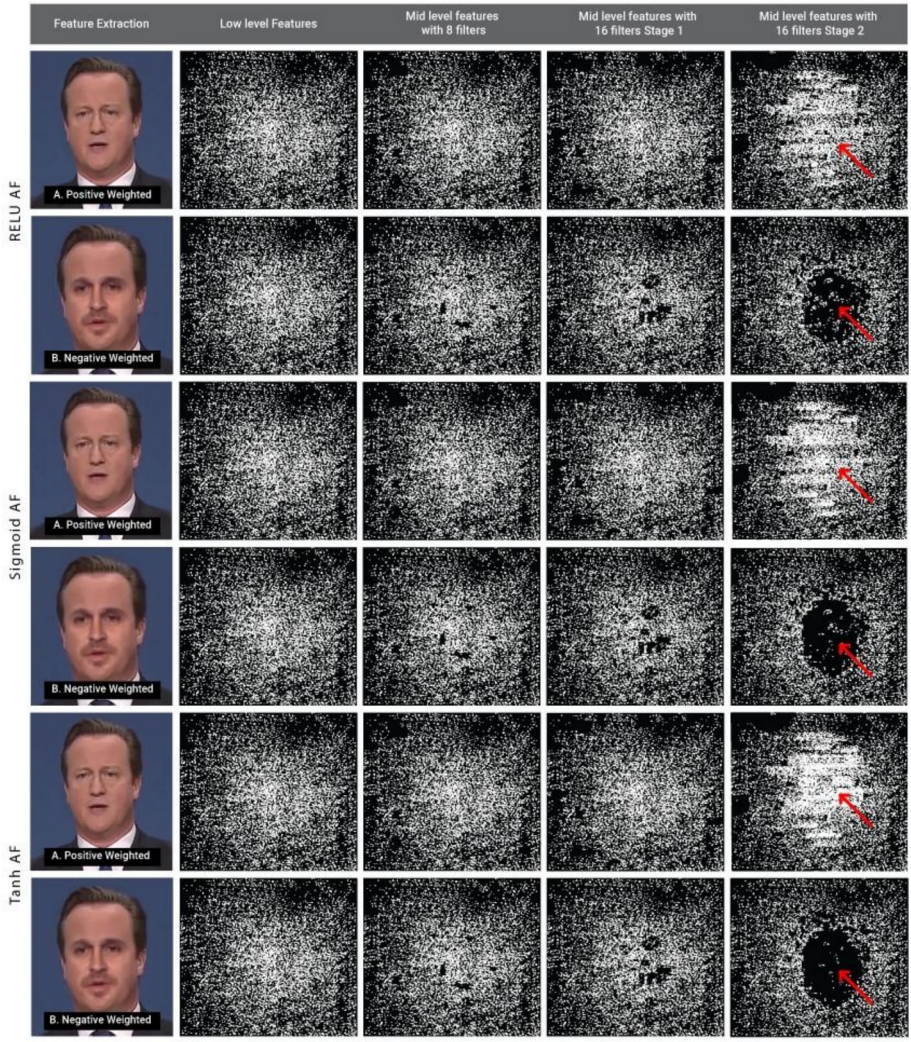

Fig. 4 Variation filters and image features at different levels. The first row shows original and fake images for RELU, sigmoid, and Tanh activation functions. The second row displays MesoNet's low and mid-level features, with the second column representing a low-level feature with an aggregation function of $3 \times 3 \times 8$, and the third and fourth columns representing mid-level features with aggregation functions of $5 \times 5 \times 16$

strong details on the back, resulting in a softer facial area. This can be attributed to the fact that deepfake images tend to be blurry or lack detail, especially compared to the rest of the intact frame.

4 Experiments and discussion

This section describes the results of experiments conducted in three main categories: improving frame quality and detecting deep falsification in videos. The experiments involved using an image aggregation algorithm to extract and improve frame features, enhancing the MesoNet fake frame detection technique with different activation functions, and developing a hybrid model that combines the image aggregation algorithm with MesoNet to detect fake frames in low-quality frames.

4.1 Deepfake dataset

The FaceForensics ++ dataset was selected due to its more than a thousand manipulated videos and their original counterparts created through the Face2Face approach. The dataset is already divided into validation, training, and testing sets, and provides lossless compressed videos, allowing the evaluation of the model's robustness across various compression levels. A subset of 19,509 images (as shown in Table 1) was randomly selected from the dataset using the Viola-Jones detector [46] and a trained neural network to detect facial features [45], with approximately 50 faces extracted per scene. Real facial images were also included, and the data set was manually reviewed to ensure accuracy and maintain a balanced distribution of high- and low-resolution images in both the training and testing sets.

4.2 Image aggregation results

To ensure statistical accuracy, we repeated all experiments using the COBRA algorithm five times. The ensemble learning method designed for regression tasks integrates several well-known noise removal techniques, such as Gaussian, salt-and-pepper, speckle, random patch suppression, and Poisson. We found that aggregating the results of multiple approaches produced superior image quality compared to using a single approach.

To assess the quality of the image improvements, we used the PSNR and RMSE scales. The PSNR scale measures the quality of image reconstruction, with typical values ranging from 20 to 80 dB [48]. Table 2 displays the results of the five runs of the COBRA algorithm, with PSNR and RMSE measures for each run. In general, all the results were favorable. The best result, with a PSNR value of 78.51 and an RMSE value of 0.028, was achieved in run5.

Table 1 Deepfake dataset

Set	Forged	Real	Total
Training	5111	7250	12361
Testing	2889	4259	7148

Table 2 Image aggregation results using COBRA algorithm

Run	PSNR	RMSE
1	69.04	0.090
2	77.73	0.033
3	75.18	0.044
4	73.04	0.067
5	78.51	0.028
Avg	74.70	0.052

4.3 MesoNet results

We refer to X as the input group and Y as the output group, and the random variable pair (X, Y) takes values in $X \times Y$ and f, as a prediction function of the selected item that takes values in X to the action group A. The selected classification task is to reduce the error action group A. The selected classification task is to reduce the error $\varepsilon(f) = E[l(f(X), y)]$, with $l(a, y) = \frac{1}{2}(a - y)^2$.

MesoNet network was implemented with Python 3.5 using the Keras 2.1.5 module [54]. The weights of the network are improved by successive batches of 75 images of $256 \times 256 \times 3$ using ADAM [55] with default parameters ($\beta_1 = 0.9$ and $\beta_2 = 0.999$). The initial learning rate is 10^{-3} divided by 10 per 1000 repetitions, down to 10^{-6}. To enhance the generalization and durability of our approach, we subjected input batches to various minor random perturbations, including zooming, rotation, horizontal shifting, brightness adjustment, and color transformation. Despite the relatively small number of parameters in the MesoNet network, we achieved impressive results with just a few hours of improvement on a standard consumer-class PC.

The ratings for the MesoNet trained network on the Face2Face forgery dataset can be found in Table 3. The MesoNet network achieved an average score of 94% across all individual frames in the dataset.

4.4 Different activation functions

Activation functions can be mathematically expressed in the neural network node. However, only a few of these functions are commonly used and are well-known for neural network analysis. The reverse propagation technique [49, 50] strikes the derivatives of the activation function. Therefore, the selected activation function must be differential [51]. Furthermore, the function should be provided smoothly for the updates of the reverse propagation weight to avoid zigzag, for example, in the sigmoid function [52]. Lastly, the activation function should easily calculate the power of backup computing, an important feature in very large neural networks consisting of millions of nodes. Below is an analysis of some functions with individual pros and cons, including real-world models, that will lead to an important comparison between them.

Table 3 Classification scores of MesoNet network on face2face dataset

Class	Forged	Real	Total
Score	92.01	96.27	94.14

4.4.1 Sigmoid function

A mathematical function with sigmoid curve features is called the sigmoid function. X-function groups are not linear because this type of activation function is nonlinear. Thus, it makes sense to stack layers. This also applies to nonbinary activation. It also has a smooth graded value. This makes them suitable for neural networks with binary classification. The sigmoid function is defined as

$$f(x) = \frac{1}{1 + e^{-x}}. \tag{5}$$

4.4.2 Tanh function

Neural network stops may occur at edge value if we only use the sine activation function. Thus, we need to apply the overload function as an alternative, which is also known as the Tanh function. The output values in the Tanh function range from -1 to 1, which is just an extension of the sigmoid function curve. Thus, negative inputs for deterministic functions will be assigned to a negative output. In addition, for input values that are close to zero, they will also be set to output values that are close to zero. Therefore, the network is not interrupted due to the above features during training. Another reason for the preference for Tanh over sigmoid is that Tanh derivatives are larger than sigmoid derivatives near zero. The Tanh function can be defined as

$$f\ (x) = \frac{(e^{x} - e^{-x})}{(e^{x} + e^{-x})}. \tag{6}$$

4.4.3 Basic rectified linear unit (ReLU)

This type of activation function is responsible for converting weighted inputs that are summarized from node to strict output or relative total. These functions are multi-definition linear functions, and positive inputs usually come out directly; otherwise the output is zero. RELU is an activation that assigns zero to the same values and a value above zero described in (7). It is noticeable that for minor regression problems with neural networks, Tanh can be better than RELU. However, any function approaching the RELU activation function is always multi-definition linear.

$$f(x) = \max(0, x). \tag{7}$$

By comparing the different functions and the characteristics of the activation functions, Table 4 shows the results of correctly classified image frames using different AFs. It is evident that the Tanh is still reliable and able to work well in binary classification models. This could be interpreted as the output range of [-1, 1] of the Tanh activation function helps center the data around zero and improves learning by reducing bias shift. Its steeper gradient compared to that of the sigmoid aids in mitigating vanishing gradient problems, leading to better training and faster convergence. Additionally, Tanh's zero-centered output can enhance the network's ability to capture subtle anomalies introduced by deepfakes, while avoiding issues like dead neurons seen with ReLU.

Table 4 Classification accuracy with different activation functions

Activation function	Sigmoid	Tanh	RELU
Accuracy	94.1%	95.8%	93.6%

4.5 Classification with aggregation

One of the drawbacks of video analysis, particularly for online videos, is the pressure it exerts on the quality of information, resulting in a significant loss of accuracy. However, the advent of facial feature enhancement methods offers the possibility of improving experiment results and obtaining more precise overall scores for videos. In this paper, we focus on seven traditional noise removal methods, including the Gaussian filter, median filter, bilateral filter, Chambolle's method [47], non-local means [16, 17], Richardson-Lucy deconvolution [13, 14], and the in-painting method [18]. Our aim was to compare the performance of these filters and to take advantage of the strengths of each to achieve better results. For example, Gaussian filters are known to blur edges, while median filters are effective against salt and pepper noise. The bilateral filter is reputed to preserve edges and retain finer details of the image, which is also a strength of non-local means. The COBRA assembly scheme was employed to aggregate the results of these filters, leading to a significant improvement in detection rates, with a peak of 98% obtained on the deepfake dataset with MesoNet.

The results are presented in Table 5, which shows the number of correctly classified images after aggregation with various activation functions. Our findings indicate that Tanh was the most effective in extracting features from images and distinguishing real from fake features compared to other activation functions. This underscores the reliability of Tanh in binary classification models. However, the classification time of the model increased due to the image aggregation process, which enhanced the quality of the existing features.

4.6 Classification with different resolutions

We investigated the effectiveness of the MesoNet model in detecting deepfakes across different resolution reduction levels. Specifically, we evaluated the model performance under mid-quality (20% image resolution reduction) and low-quality (50% image resolution reduction) settings. It is worth noting that further reducing the image resolution could result in videos with indiscernible details, which is a rare occurrence in real-world scenarios. Our experiments revealed that the hybrid model performed exceptionally well even under varying frame compression conditions. In particular, the model achieved a resolution score of 93. 75% when the quality of the frame was reduced by half and 96. 73% when the quality of the frame was reduced to 20%. These results are presented in Table 6.

Table 5 Video classification scores on Face2Face dataset before and after image aggregation

Activation Function	Accuracy	Time (hrs)
Sigmoid	94.10%	22.06
Tanh	95.83%	19.56
RELU	93.63%	16.50
Sigmoid Aggregation	97.90%	23.54
Tanh Aggregation	98.05%	21.01
RELU Aggregation	96.23%	18.43

Table 6 Video classification scores on Face2Face dataset with different resolutions

Activation function	Accuracy	20% quality	50% quality
Sigmoid	97.90%	96.03%	92.81%
Tanh	98.05%	96.73%	93.75%
Relu	96.23%	94.82%	90.61%

4.7 Performance evaluation

Assessing the robustness in detecting deepfake facial tampering across different levels of resolution and video quality degradation is significant for several reasons. *a) Real-world applicability:* ensures that the model is effective even when video quality is compromised, making it more practical for diverse applications, especially since deepfakes are sometimes generated at low resolution to complicate detection [57]. *b) Generalization:* testing helps determine how well the model performs beyond its training conditions, revealing its ability to handle various distortions and artifacts. *c) Performance metrics:* provide insights into how performance changes with video quality degradation, helping identify thresholds where accuracy might start to decline. *d) Improvement opportunities:* are identified by analyzing performance, offering chances to refine and enhance the model to better handle such variations.

Table 7 presents a comparative analysis of various approaches for detecting deepfakes in face–to–face swaps. The best results achieved for each public database are highlighted in bold. It is worth mentioning that all studies utilized the faceforensics++ database, except for Li et al., who employed a modified version of the original database, namely Deepfake TIMIT.

Although the latest deepfake approaches show visually impressive results, it was shown that they can still be detected by trained forgery detectors. It is particularly encouraging that the difficult state of low-quality video can also be addressed; image aggregation to improve image quality, and the MesoNet to detect deepfake, where humans and handmade features face difficulties. To train detectors using domain knowledge, we offer a hybrid model that may

Table 7 Comparison with performance in literature*

Work	Method	Classifier	AUC	Dataset
Matern et al. [19]	Visual Features	LR	78.0%(HQ), 77.0%(LQ)	FF++
Yang et al. [22]	Head Pose	SVM	89.0%(HQ), 89.0%(LQ)	FF++
Agarwal and Farid [23]	Head Pose and Facial	SVM	96.3%(HQ), –(LQ)	FF++
Huang et al. [58]	Face Swaping	CNN	96.8%(HQ), –(LQ)	FF++
Li et al. [56]	Face Warping	CNN	**99.7%(HQ)**, –(LQ)	Deepfake TIM
Luo et al. [59]	Face Forgery	CNN	95.7%(HQ), –(LQ)	FF++
Rossler et al. [11]	Steg analysis	CNN	94.0%(HQ), –(LQ)	FF++
Sun et al. [60]	Face Forgery	CNN	96.9%(HQ), –(LQ)	FF++
Nguyen et al. [12]	Deep Learning	Capsule Net	96.6%(HQ), –(LQ)	FF++
Our model	Frame aggregation	CNN +AF	98.1%(HQ), **93.75%(LQ)**	FF++

* HQ and LQ stands for high and low image quality respectively, and '–' sign in LQ means research did not investigate low quality images

help develop deep detection algorithms and make them more accurate, with a low-quality dataset for manipulated faces.

In this paper, we focus on the impact of detection of modern manipulation methods, in particular facial swapping, for low-quality videos. All image data, and pre-trained model, used in this work are publicly available and are already used by other researchers. In particular, transfer learning is of great importance in the forensic community. The performance results were very good in cases of different image compression as the model's accuracy to reduce the image quality of the half was 93.75%. The proposed new approach could be promising for future research in the field of digital media forensics, particularly with a focus on facial forgery.

5 Conclusion

This research work aimed to determine the feasibility of developing image aggregation and MesoNet frame aggregation to improve the quality of frames as input and output to determine fake or real frames. The work showed that the introduction of a generic aggregate denoising process can improve the performance of prefilters and fully exploit their capabilities. Furthermore, the proposed deepfake detection network was able to achieve an average face-to-face recognition rate of 98% and 93. 75% for low-quality videos. However, the study also suggests that more tools are needed to develop deeper, more effective and efficient networks in the future. This includes exploring audio techniques for spotting deepfakes, discovering a comprehensive approach capable of detecting all types of deepfakes, and exploring the detection of multiple deepfake faces in the same frame.

Author Contributions **Mohammad Qawasmi:** Writing - original draft; Formal analysis; Validation. **Omar Al-Kadi:** Conceptualization; Methodology; Supervision; Writing - review & editing.

Funding No funding was received for conducting this work.

Data Availability Dataset used in this work is publicly available.

Declarations

Competing Interests The authors declare no competing interests.

References

1. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Bengio Y (2020) Generative adversarial networks. Commun ACM 63(11):139–144
2. Caldelli R, Galteri L, Amerini I, Del Bimbo A (2021) Optical Flow based CNN for detection of unlearnt deepfake manipulations. Pattern Recognit Lett 146:31–37
3. Bharati A, Singh R, Vatsa M, Bowyer KW (2016) Detecting facial retouching using supervised deep learning. IEEE Trans Inf Forensics Secur 11(9):1903–1913
4. Flynn PJ, Bowyer KW, Phillips PJ (2003) Assessment of time dependency in face recognition: an initial study. In: International conference on audio-and video-based biometric person authentication. Springer, Berlin, Heidelberg, pp 44-51
5. Jain A, Singh R, Vatsa M (2018) On detecting gans and retouching based synthetic alterations. In: 2018 IEEE 9th International conference on biometrics theory, applications and systems (BTAS) pp 1–7
6. 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, pp 770-778

7. Chu LC, Anandkumar A, Shin HC, Fishman EK (2020) The potential dangers of artificial intelligence for radiology and radiologists. J Am College Radiol 17(10):1309–1311
8. Wang R, Juefei-Xu F, Ma L, Xie X, Huang Y, Wang J, Liu Y (2019) FakeSpotter: a simple yet robust baseline for spotting AI-synthesized fake faces. arXiv:1909.06122
9. Parkhi OM, Vedaldi A, Zisserman A (2015) Deep face recognition
10. Bitesize BBC (2019) Deepfakes: what are they and why would i make one?
11. Rossler A, Cozzolino D, Verdoliva L, Riess C., Thies J, Nießner M (2019) Faceforensics++: learning to detect manipulated facial images. In: Proceedings of the IEEE international conference on computer vision, pp 1-11
12. Nguyen HH, 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
13. Anthropics Technology Ltd (n.d.-b) PortraitPro - easy photo editing software. PortraitPro
14. Sabir E, Cheng J, Jaiswal A, AbdAlmageed W, Masi I, Natarajan P (2019) Recurrent convolutional strategies for face manipulation detection in videos. Interfaces (GUI) 3(1)
15. Schroepfer M (2019) Creating a data set and a challenge for deepfakes. Facebook artificial intelligence
16. Buades A, Coll B, Morel JM (2005) A non-local algorithm for image denoising. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol 2 pp 60-65
17. Buades A, Coll B, Morel JM (2011) Non-local means denoising. Image Process Line 1:208–212
18. Chui CK, Mhaskar HN (2010) MRA contextual-recovery extension of smooth functions on manifolds. Appl Comput Harmon Anal 28(1):104–113
19. 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), pp 83-92
20. Sun D, Yang X, Liu MY, Kautz J (2018) Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8934–8943
21. Korshunov P, Marcel S (2018) Deepfakes: a new threat to face recognition? assessment and detection. arXiv:1812.08685
22. Yang X, Li Y, Lyu S (2019) Exposing deep fakes using inconsistent head poses. In: ICASSP 2019-2019 IEEE international conference on acoustics, speech and signal processing (ICASSP), pp 8261–8265
23. Agarwal S, Farid H, Gu Y, He M, Nagano K, Li H (2019) Protecting world leaders against deep fakes. In: CVPR Workshops, pp 38–45
24. Baltrusaitis T, Zadeh A, Lim YC, Morency LP (2018) Openface 2.0: facial behavior analysis toolkit. In: 2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018), pp 59–66
25. Galbally Herrero J, Marcel S, Fiérrez J (2014) Image quality assessment for fake biometric detection: application to Iris, fingerprint, and face recognition. IEEE Trans Image Process
26. Goodfellow I, Bengio Y, Courville A, Bengio Y (2016) Deep learning. Cambridge: MIT press, vol 1, no 2
27. King DE (2009) Dlib-ml: a machine learning toolkit. J Mach Learn Res 10:1755–1758
28. McCloskey S, Albright M (2018) Detecting gan-generated imagery using color cues. arXiv:1812.08247
29. Guan H, Kozak M, Robertson E, Lee Y, Yates AN, Delgado A, Fiscus J (2019) MFC datasets: large-scale benchmark datasets for media forensic challenge evaluation. In: 2019 IEEE winter applications of computer vision workshops (WACVW), pp 63–72
30. Amos B, Ludwiczuk B, Satyanarayanan M (2016) Openface: a general-purpose face recognition library with mobile applications. CMU School Comput Sci 6(2)
31. Schroff F, Kalenichenko D, Philbin J (2015) Facenet: a unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 815–823
32. Karras T, Aila T, Laine S, Lehtinen J (2017) Progressive growing of gans for improved quality, stability, and variation. arXiv:1710.10196
33. 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
34. Alaluf Y, Patashnik O, Cohen-Or D (2021) Only a matter of style: age transformation using a style-based regression model. arXiv:2102.02754
35. Guarnera L, Giudice O, Battiato S (2020) Deepfake detection by analyzing convolutional traces. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, pp 666–667
36. Moon TK (1996) The expectation-maximization algorithm. IEEE Signal Process Mag 13(6):47–60
37. He Z, Zuo W, Kan M, Shan S, Chen X (2019) Attgan: facial attribute editing by only changing what you want. IEEE Trans Image Process 28(11):5464–5478

38. Cho W, Choi S, Park DK, Shin I, Choo J (2019) Image-to-image translation via group-wise deep whitening-and-coloring transformation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 10639–10647
39. Choi Y, Choi M, Kim M, Ha JW, 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
40. Karras T, Laine S, Aittala M, Hellsten J, Lehtinen J, Aila T (2020) Analyzing and improving the image quality of stylegan. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 8110–8119
41. Dang H, Liu F, Stehouwer J, Liu X, Jain AK (2020) On the detection of digital face manipulation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 5781–5790
42. Biau G, Fischer A, Guedj B, Malley JD (2016) COBRA: a combined regression strategy. J Multivar Anal 146:18–28
43. Guedj B, Desikan BS (2018) Pycobra: a python toolbox for ensemble learning and visualisation. J Mach Learn Res 18(190):1–5
44. Afchar D, Nozick V, Yamagishi J, Echizen I (2018) Mesonet: a compact facial video forgery detection network. In 2018 IEEE international workshop on information forensics and security (WIFS), pp 1–7
45. King DE (2009) Dlib-ml: a machine learning toolkit. J Mach Learn Res 10:1755–1758
46. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, vol 1, pp I–I
47. Chambolle A (2005) Total variation minimization and a class of binary MRF models. In: International Workshop on energy minimization methods in computer vision and pattern recognition, pp 136–152
48. Okumura H (2018) Is it really impossible to divide by zero? J Appl Math 27(2):191–198
49. Robbins H, Monro S (1951) A stochastic approximation method. Ann Math Stat 400–407
50. Nielsen MA (2015) Neural networks and deep learning, vol 25. Determination press, San Francisco, CA, USA
51. Hecht-Nielsen R (1992) Theory of the backpropagation neural network. In: Neural networks for perception, pp 65–93
52. LeCun Y, Bottou L, Orr GB, Müller KR (1998) Neural networks: tricks of the trade. Springer Lect Notes Comput Sci 1524(5–50):6
53. Guedj B, Rengot J (2020) Non-linear aggregation of filters to improve image denoising. In: Science and information conference, pp 314–327
54. Chollet F (2015) Keras: deep learning library for theano and tensorflow
55. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv:1412.6980
56. Li Y, Chang MC, Lyu S (2018) 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
57. Li Y, Bian S, Wang C, Polat K, Alhudhaif A, Alenezi F (2023) Exposing low-quality deepfake videos of social network service using spatial restored detection framework. Expert Syst Appl 231:120646
58. Huang B, Wang Z, Yang J, Ai J, Zou Q, Wang Q, Ye D (2023) Implicit identity driven deepfake face swapping detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 4490–4499
59. Luo Y, Zhang Y, Yan J, Liu W (2021) Generalizing face forgery detection with high-frequency features. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 16317–16326
60. Sun K, Liu H, Yao T, Sun X, Chen S, Ding S, Ji R (2022) An information theoretic approach for attention-driven face forgery detection. In: European conference on computer vision. Springer, pp 111–127

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.