symmetry

MDPI

Article

Spotting Deepfakes and Face Manipulations by Fusing Features from Multi-Stream CNNs Models

Semih Yavuzkilic [1], Abdulkadir Sengur [1], Zahid Akhtar [2] and Kamran Siddique [3,*]

[1] Department of Electrical and Electronics Engineering, Fırat University, Elazig 23000, Turkey; semihyavuzkilic@gmail.com (S.Y.); ksengur@gmail.com (A.S.)
[2] Department of Network and Computer Security, State University of New York Polytechnic Institute, Utica, NY 13502, USA; akhtarz@sunypoly.edu
[3] Department of Information and Communication Technology, Xiamen University Malaysia, Sepang 43900, Malaysia
* Correspondence: kamran.siddique@xmu.edu.my

Abstract: Deepfake is one of the applications that is deemed harmful. Deepfakes are a sort of image or video manipulation in which a person's image is changed or swapped with that of another person's face using artificial neural networks. Deepfake manipulations may be done with a variety of techniques and applications. A quintessential countermeasure against deepfake or face manipulation is deepfake detection method. Most of the existing detection methods perform well under symmetric data distributions, but are still not robust to asymmetric datasets variations and novel deepfake/manipulation types. In this paper, for the identification of fake faces in videos, a new multi-stream deep learning algorithm is developed, where three streams are merged at the feature level using the fusion layer. After the fusion layer, the fully connected, Softmax, and classification layers are used to classify the data. The pre-trained VGG16 model is adopted for transferred CNN1stream. In transfer learning, the weights of the pre-trained CNN model are further used for training the new classification problem. In the second stream (transferred CNN2), the pre-trained VGG19 model is used. Whereas, in the third stream, the pre-trained ResNet18 model is considered. In this paper, a new large-scale dataset (i.e., World Politicians Deepfake Dataset (WPDD)) is introduced to improve deepfake detection systems. The dataset was created by downloading videos of 20 different politicians from YouTube. Over 320,000 frames were retrieved after dividing the downloaded movie into little sections and extracting the frames. Finally, various manipulations were performed to these frames, resulting in seven separate manipulation classes for men and women. In the experiments, three fake face detection scenarios are investigated. First, fake and real face discrimination is studied. Second, seven face manipulations are performed, including age, beard, face swap, glasses, hair color, hairstyle, smiling, and genuine face discrimination. Third, performance of deepfake detection system under novel type of face manipulation is analyzed. The proposed strategy outperforms the prior existing methods. The calculated performance metrics are over 99%.

Keywords: deepfake; fake face detection; face manipulations; multi-stream CNNs

Citation: Yavuzkilic, S.; Sengur, A.; Aktar, Z.; Siddique, K. Spotting Deepfakes and Face Manipulations by Fusing Features from Multi-Stream CNNs Models. *Symmetry* **2021**, *13*, 1352. https://doi.org/10.3390/sym13081352

Academic Editor: Leyi Wei

Received: 1 July 2021
Accepted: 21 July 2021
Published: 26 July 2021

1. Introduction

Artificial intelligence (AI) technologies have become one of the most popular applications in recent years. Today, these technologies have a wide variety of applications. However, some of these practices are now described as a threat or danger. One of the applications considered dangerous is Deepfake. Deepfakes are a type of manipulation in which a person's image is altered or swapped by another person's face via artificial neural networks in an image or video [1]. The manipulation of Deepfakes can be performed using different algorithms or applications. Generative adversarial networks (GANs) [2] algorithms are at the top of these methods. These methods achieve surprising results and give examples of faces that are almost indistinguishable from the real ones. In addition,

many public applications such as DeepFake, Face2Face, FaceApp, and Face Swap Live have emerged with the rapid increase of social media use. The aforementioned applications and software can be used to change a person's facial age, gender, hair color, or facial expression; to swap two faces with each other; or to create synthetic facial specimens of a person not in the real world.

Deepfakes are attributed to manipulated videos or other digital representations produced by advanced AI and Deep Learning (DL) that generate fake images and sounds that perform to be real. Although this technology may seem amusing and harmless, it can lead to significant risks and dangers through the use of applications and software by malicious people. For example, it can be used to produce audio and video imitations for theft, fraud, or revenge porn. The cyber wars and digital disasters in today's world, deepfake can be used to trigger social events and for perception management. Similarly, trial processes can be manipulated by using false evidence in courts; people are thereby accused of crimes that they did not commit. Therefore, innocent people can go to jail unfairly. Besides, malicious actors can use Deepfake algorithms as a tool to produce and spread fake news about a company being involved in consumer fraud, bribery, sexual harassment, or such crimes. They can show you or someone from your company as the person that committed these crimes. This kind of unrealistic news can damage your company's reputation and make it difficult for you to prove otherwise.

The conventional countermeasure against deepfake is deepfake detection methods. Several datasets were published to overcome deepfake detection, such as UADFV [3], DeepFake-TIMIT [4], and Celeb-DF [5]. In addition, some companies released large-scale datasets to support researchers against this phenomenon. Google created a large-scale dataset, which is called DeepFakeDetection [6], to promote researches in developing new methods for deepfake detection. Furthermore, Facebook and Microsoft presented a new dataset that has been included in the Deepfake Detection Challenge (DFDC) [7].

Many studies have been conducted on deepfake detection. For example, Afchar et al. [8] presented a convolutional neural network (CNN)-based deepfake detection method. The method in [8] concentrated on the mesoscopic properties of images via a deep neural network (DNN) with few layers. It was trained on the datasets that were collected by authors and were composed of hyper-realistic forged videos. The authors obtained 98.4% and 95.3% accuracy for deepfake and Face2Face datasets, respectively. In [9], the authors integrated a new DL-based approach for detecting deepfake videos. The method detected the artifacts by matching the synthesized facial areas and their neighborhood regions with a CNN model. Exclusively, they used a residual network with a 50-layer (ResNet-50) [10] model to detect the deepfake videos by highlighting the facial warping artifacts introduced by rescaling and interpolation processing in fundamental deepfake generation algorithms. Researches were implemented on UADFV and DeepfakeTIMIT low quality (LQ) and DeepfakeTIMIT high quality (HQ) datasets and they achieved 97.4%, 99.9%, and 93.2% accuracy rate, respectively. Zhou et al. [11] utilized two-stream CNNs with the underlying GoogleNet model [12] for face manipulation detection. While the first stream detects interfering artifacts on a face, the second one is a trained patch-based triple net. Experiments were performed on the unpublished dataset, which was composed by authors helping of SwapMe and FaceSwap applications. Their method attained a 99.9% detection accuracy rate.

In [13], the authors investigated the GAN pipeline for the detection of varied artifacts among real and fake images. They suggested a detection technique dependent on color characteristics and a linear support vector machine (SVM) for classification. The method was applied on the NIST MFC2018 [14] dataset. They reached 70.0% of the area under the curve (AUC). Nataraj et al. [15] presented a detection method based on natural image statistics and steganalysis. The suggested technique has especially relied on a compound of pixel co-occurrence matrices and CNNs. The method was first tested on a dataset of various objects composed using the CycleGAN [16] structure. Moreover, the authors performed an interesting analysis to notice the validity of the proposed method to fake images generated

through two GAN structures (CycleGAN and StarGAN [17]) with good generalization outcomes. Next, in [18], the authors performed this detection method on the 100K-Faces dataset and gained an EER result of 7.2%.

Nguyen et al. [19] utilized capsule structures [20] that rely upon a VGG19 [19] network to detect crafted facial pictures and videos. The method obtained 97.05% accuracy rate on the FaceForensics++ [21] dataset. The work in [22] presented a detection technique that tried to capture visual artifacts in the eyes, teeth, and face circumferences of the crafted faces. Two different processes were performed, i.e., multilayer feedforward neural network and logistic regression (LogReg) model classifiers. The suggested method was obtained a 0.866 AUC value on an unpublished dataset with deepfake videos from YouTube, and frames were cropped from CelebA dataset [23]. In [24], the authors presented a fake detection approach that relied on the analysis of the convolutional traces. They used Expectation-Maximization (EM) algorithm [25] to extract features. Classifiers such as k-Nearest Neighbors (k-NN), Support Vector Machine (SVM), and Linear Discriminant Analysis (LDA) were used in the study. The introduced approach was tested using forged images created through AttGAN [26], GDWCT [27], StarGAN, StyleGAN, and StyleGAN2 [28], obtaining a 99.81% accuracy rate for the best performance. In [29], the authors presented DeepFakesONPhys, which is the backdemanded physiological measurement for detection of Deepfake. The suggested method particularly relied on a Convolutional Attention Network (CAN) that is trained for heart rate estimation utilizing remote photoplethysmography (rPPG). The offered CAN structure comprises two parallel CNN networks that extract and share temporal and spatial information from video frames. The proposed method was performed on Celeb-DF and DFDC datasets and obtained 98.7%, and 94.4% accuracy values, respectively. A summary of Deepfake manipulation detection methods is presented in Table 1. Despite remarkable progress in deepfake and face manipulation detection, the majority of the existing techniques are effective only for symmetric data distributions. However, their performance degrades under asymmetric or novel datasets and deepfake/manipulation types not used in the training process.

Table 1. A representative list of deepfake detection methods.

Methods	Techniques	Dataset	Year
Afchar et al. [8]	Designed CNN	Private dataset	2018
Li et al. [9]	Face Warping Features	UADFV and DeepfakeTIMIT	2019
Zhou et al. [11]	GoogleNet model	Private dataset	2017
McCloskey and Albright [13]	GAN-Pipeline Features	NIST MFC2018	2018
Nataraj et al. [15]	Steganalysis Features	100K-Faces	2019
Nguyen et al. [19]	Capsule Network	FaceForensics++	2019
Matern et al. [22]	CNN and Logistic Regression Model	Private dataset	2019
Guarnera et al. [24]	GAN-Pipeline Features	Private dataset	2020
Hernandez-Ortega et al. [29]	Convolutional Attention Network (CAN)	Celeb-DF and DFDC	2021

This paper presents a new large-scale dataset (i.e., World Politicians Deepfake Dataset (WPDD)) to improve deepfake detection algorithms. Videos of 20 different politicians were downloaded from YouTube to create the dataset. These downloaded videos were divided into small parts and over 320,000 frames were obtained. Finally, these frames were grouped and various manipulations were applied as seven different manipulation classes for men and women. In addition, a novel multi-stream deep learning method is proposed for fake face detection in a given video. The proposed architecture consists of three streams, which are combined in the fusion layer. After the fusion layer, the classification is carried out through the fully connected, Softmax, and the classification layers, respectively. The original image is used in the first stream to learn broad features like head shape and hair color. The blurred picture is utilized as input in the second stream, which focuses on the skin color. The last stream is concerned with the sharpened input image's local characteristics of the face. Face detection, face blurring, and face sharpening techniques

are used in the preprocessing step. The detected face regions are initially cropped and then are resized to 224 × 224. Sharpness is described as the contrast between two or more colors. The pre-trained VGG16 model is adopted for transferred CNN1stream. In transfer learning, the weights of the pre-trained CNN model are further used for training the new classification problem. In the second stream (transferred CNN2), the pre-trained VGG19 model is used and in the third one, the pre-trained ResNet18 model is considered.

The remainder of the paper is as follows. In the next section, the methodology and the related theory together with CNN, transfer learning, and dataset are introduced. In Section 3, the experimental works and results are examined. The paper is concluded in Section 4.

2. Proposed Mutli-Stream Deepfake Detection Methodology

The proposed multi-stream deep learning method is depicted in Figure 1. As seen in Figure 1, the proposed architecture consists of three streams, which are combined in the fusion layer at feature level. The first stream uses the original image to learn general characteristics such as head shape and hair color. The second stream is used to focus on the skin color, where the blurred image is used as input. The last stream deals with the local features of the face that are provided by the sharpened input image. In the preprocessing stage, face detection, face blurring, and face sharpening operations are applied. The detected face regions are initially cropped and then are resized to 224 × 224. Sharpness is described as the contrast between two or more colors. A smooth transition from black to white is appealing. The transformation from black to gray to white is vague. When images are sharpened, the contrast around the edges where different colors intersect is increased. For blurring of the detected face images, the guided filter is considered. The radius of the square window of the guided filter is chosen as 12.

The pre-trained VGG16 model is adopted for transferred CNN1stream. In transfer learning, the weights of the pre-trained CNN model are further used for training the new classification problem. In the second stream (transferred CNN2), the pre-trained VGG19 model is used, and in the third one, the pre-trained ResNet18 model is considered.

Simonyan et al. [30] developed the VGG16 model, which is a deep CNN model. The VGG16 model has a total of 41 layers, 16 of which have learnable weights. The learnable layers are made up of 13 convolutional layers and three completely linked layers. Both convolutional layers in the VGG16 model use tiny 3 × 3 kernels, and max-pooling layers come after convolutional layers. VGG19 was developed as a deeper version of the VGG16 model [30]. It contains 47 layers where there are 16 convolutional layers, 18 ReLu layers, 5 max-pooling layers, 2 dropout layers, 3 fully connected layers, softmax, and classification output layer. The pretrained ResNet18 is another deep CNN model with 71 layers. The ResNet18 model is composed of 20 convolutional layers, 17 ReLu layers, 20 Batch Normalization layers, 8 addition layers, 1 Max pooling layer, 1 global pooling layer, softmax and classification output layer. The outputs of fully connected layers from each stream are then concatenated using the fusion layer. Average pooling is employed to the output of the fusion layer. The results are used as input to the last fully connected layer that has either eight or two output neurons. To classify, the softmax layer is used.

2.1. CNN and Transfer Learning

Convolutional Neural Networks (*CNNs*) are a type of neural network that are becoming increasingly common due to their high success in several image-based object recognition applications [31,32]. With its end-to-end learning architecture, CNN can extract features and classify them. Convolution, pooling, normalization, and fully connected layers are among the layers that make up a general CNN architecture. To construct the network, convolution, pooling, and normalization layers are embedded in sequential order. Convolution and pooling operations are used in succession to create high-level features on which classification is done. The classification is done in the CNN architecture's fully connected layer. During CNN preparation, a large number of parameters must be modified

in the CNN architecture. The traditional back-propagation algorithm is commonly used to train CNNs.

Transfer Learning (*TL*) is classified as the process of transferring information from one domain to another for classification and feature extraction [33]. TL is performed using a deep CNN model that was previously trained on a large dataset in the deep learning perspective. The pre-trained CNN model is fine-tuned using a new dataset with a lower number of training images than the previously trained datasets. As fine-tuning a pre-trained CNN model is normally much faster and easier than training a CNN model with randomly initialized weights from scratch, TL has recently been used in many deep learning applications.

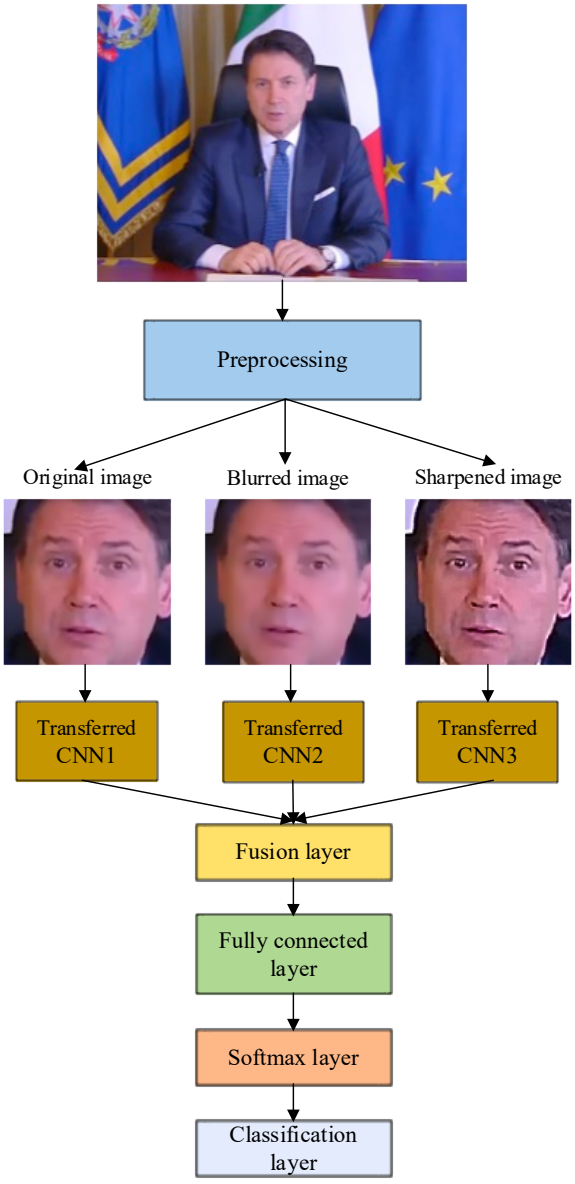


Figure 1. The illustration of the proposed multi-stream CNN for deepfake classification.

In general, the initial layers of CNN models learn features like edges, curves, corners, and color blobs, and the final layers of CNN models reflect abstract and unique features [31]. In functional implementations, the fully attached layer, softmax layer, and classification output layer of the pre-trained CNN model are discharged, and the remaining layers are passed to the current classification mission.

2.2. Dataset

To create the related (i.e., World Politicians Deepfake Dataset (WPDD)) dataset, the videos of twenty world-famous politicians, nine of them women and eleven of them men, were downloaded from YouTube. These videos, the actual lengths between the range of 5 to 16 min, were divided into 946 videos with a length of 10 to 20 s. Then, the frames of these videos were extracted and a total of 320,499 images (frames) were obtained. These images were grouped and various manipulations were applied, including 7 different classes which are makeup (for women only), hairstyle, hair color, age, smile, glasses, beard (for men only), and face swap. For each manipulation class, different types of manipulations were applied to belong to the related class. For instance, the different hairstyle types such as curly, straight and wavy were used for the hairstyle manipulation class. Figure 2 shows a sample of the dataset.

Figure 2. Sample frames from our WPDD dataset. The first row shows real images of different politicians. The following rows depict fake images of manipulation classes, i.e., makeup (only for women)/beard (only for men), hairstyle, hair color, age, smiling, glasses, and face swap, respectively.

3. Experiments

In this section, we provide the experimental evaluations of the proposed deepfake and face manipulation detection framework. Furthermore, a comparison of the proposed method with prior techniques, performance under novel face manipulation types, and performance on existing datasets are presented.

3.1. Experimental Protocol

The experimental works were performed on MATLAB with a workstation that contains the NVIDIA Quadro M4000 GPU. As it was mentioned earlier, the detected face regions were resized to 224 × 224 for being compatible with the input of the pre-trained deep models. The well-known Viola–Jones detection algorithm was used for face region detection [34]. Randomly selected 70% of datasets were used in training of the proposed method, and the rest 30% of datasets were used for testing of the proposed method. In the fine-tuning of the pre-trained CNN models, the number of epochs was set to 5. The dropout probability was set to 0.2. The MiniBatch size, Gradient Threshold, and the initial learning rate were chosen as 32, 2, and 0.001, respectively.

In the experimental works, two different scenarios were considered. In the first one, the discrimination between the real and fake faces was carried out, and in the second scenario, the discrimination between the seven fake faces and real faces (eight-class classification problem) was considered. The classification performance was measured by using the accuracy, sensitivity and specificity criteria.

3.2. Experimental Results

3.2.1. Scenario 1

Figure 3 shows the training and the loss progress of the multi-stream CNN model for Scenario 1. As seen in Figure 3a, the multi-stream CNN for the first scenario was trained for 250 iterations, where the elapsed time was 54 min. The initial performance of the CNN model was ~50%, and it dropped to 40% at the beginning of the training. It increased towards 90% at the 50th iteration, and after the 100th iteration, the training accuracy was 100%. The loss value (Figure 3b) was around five at the beginning of the training, and it increased to 10 after a few iterations. After the 50th iteration, it dropped around 0.1 and after the 100th iteration, the loss value was quite close to 0. Figure 4 shows the obtained confusion matrix for Scenario 1. While the columns of the confusion matrix show the predicted classes, the rows show the true classes. From Figure 4, it is seen that 120 fake faces were classified as real and 160 real faces were classified as fake faces.

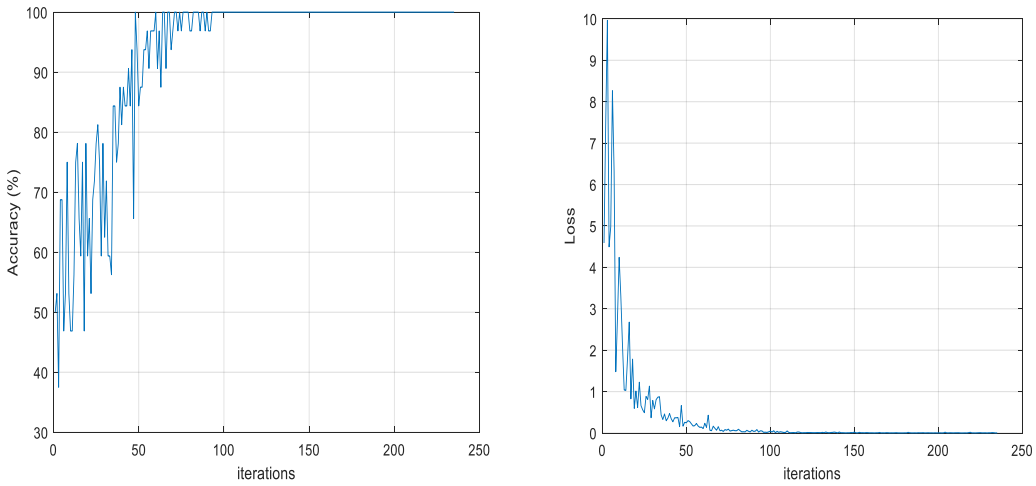


Figure 3. The training accuracy and the loss curves for the training of Scenario 1.

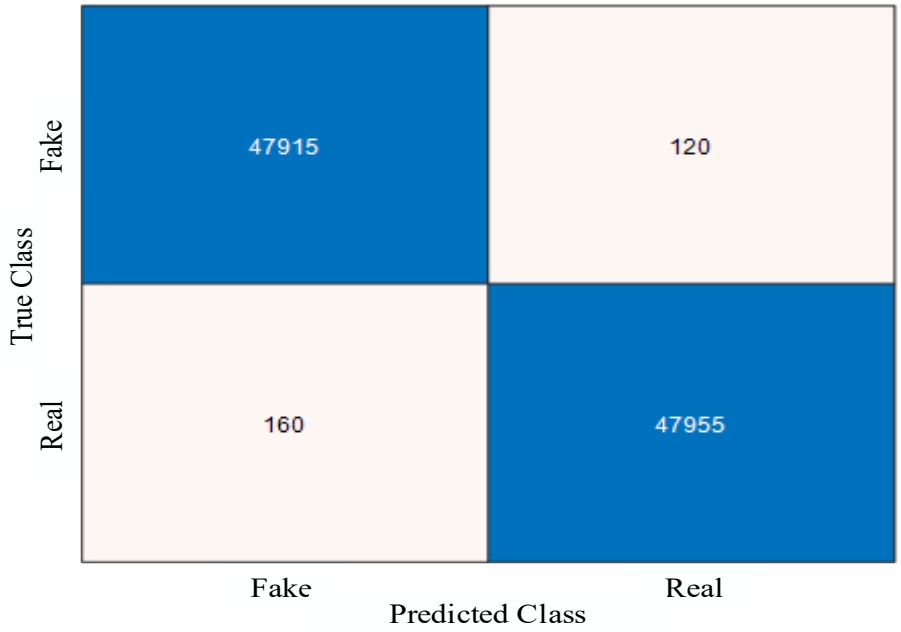


Figure 4. The obtained confusion matrix for Scenario 1.

Besides, 47,915 fake and 47,955 real faces were correctly classified by the proposed multi-stream CNN approach. In Table 2, the performance evaluation measures were given for Scenario 1. As seen in Table 2, the calculated accuracy, sensitivity, and specificity scores were 99.71%, 99.67%, and 99.75%, respectively.

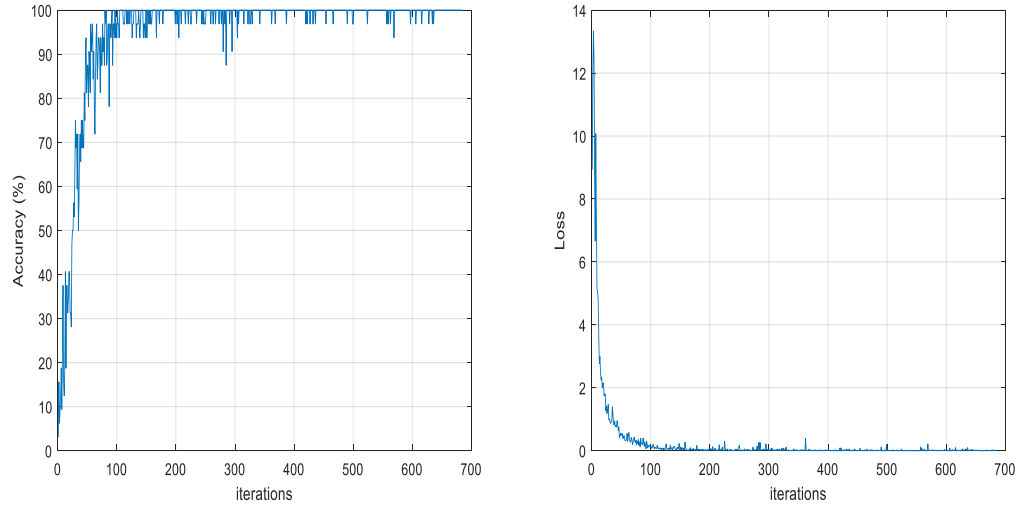
Table 2. The calculated performance measures for Scenario 1.

	Accuracy	Sensitivity	Specificity
Scenario 1	99.71%	99.67%	99.75%

3.2.2. Scenario 2

Figure 5 shows the training and the loss progress of the multi-stream CNN model for Scenario 2. While Figure 5a shows the training accuracy curve, i.e., the loss curve through the training. As seen in Figure 5a, the multi-stream CNN for the first scenario was trained for 700 iterations, where the elapsed time was 81 min. The initial performance of the CNN model was around 10% and it increased toward 90% through the 100th iteration. In addition, after the 100th iteration, the training accuracy was approximately 100%. The loss value was around 13 at the beginning of the training and it dropped to around zero at the 100th iteration. The loss value was quite close to zero after the 100th iteration.

Figure 6 shows the obtained confusion matrix for Scenario 2. The columns of the confusion matrix show the predicted classes, while the rows show the true classes. From Figure 6, 9212, 11,494, 20,083, 10,304, 8505, 9191, 5130, 11,781, and 11,739 faces from Age, Beard, Face swap, Glasses, Hair color, Hairstyle, Makeup, Smiling, and Real classes were correctly classified by the proposed multi-stream CNN approach. As seen in Figure 6, 14 samples from Face swap, Smiling, and Real classes were predicted as Age class. In addition, 63, 7, 7, and 49 samples from Hair Color, Hairstyle, Smiling, and Real were misclassified.



(**a**) (**b**)

Figure 5. The training of Scenario 2's (**a**) the training accuracy curve and (**b**) the loss progress curve.

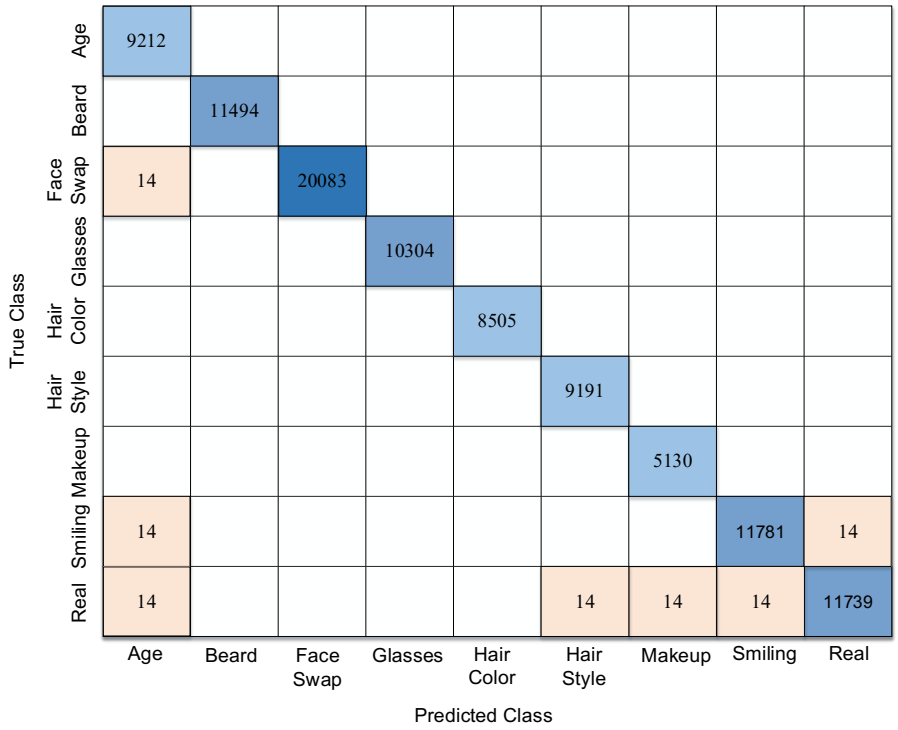


Figure 6. The obtained confusion matrix for Scenario 2.

In Table 3, the performance evaluation measures were given for Scenario 2. As can be seen in Table 3, the calculated accuracy, sensitivity, and specificity scores were 99.82%, 99.83%, and 99.97%, respectively.

Table 3. The calculated performance measures for Scenario 2.

	Accuracy	Sensitivity	Specificity
Scenario 2	99.82%	99.83%	99.97%

3.2.3. Performance under Novel Type of Face Manipulation

We performed further experiments to evaluate the generalization capacity of the proposed method across different manipulation types. The accuracy of the proposed deep stream method was evaluated considering only one manipulation type as the training set, while the dataset obtained by the other manipulation type was used as the testing set, e.g., samples only with age manipulation were used as train set and samples with beard manipulation as test set. In Table 4, the obtained results were reported. As seen in Table 4, age, face swap, glasses, hair color, hairstyle, and smiling manipulation types were used in training and tested with the other manipulations. When the age manipulation type was used in training, and tested on the other manipulation types, an 84.84% average detection accuracy score was the obtained.

For the Face swap manipulation type, an 86.47% average detection accuracy score was the obtained. The Beard manipulation type was detected with the highest accuracy score where a 97.81% accuracy score was obtained. The average accuracy score for Glasses manipulation type was 77.01%. For Glasses manipulation type, the best accuracy score 86.30% was obtained for the Age manipulation type. For Hair color manipulation type, the obtained average detection accuracy score was 73.57%. An 88.37% accuracy score was produced for Hairstyle manipulation type when the Hair color manipulation type was used in training. Similarly, for Hairstyle manipulation type, the highest accuracy score 83.26% was obtained for the Hair color manipulation type. The average testing accuracy score for Hairstyle manipulation type was 75.37%. Last, for Smiling manipulation type, a 77.12% average accuracy score was obtained. The results in Table 4 show that the proposed method is capable of obtaining notable generalization capability, even being a simple straightforward method inspired by ensemble learning. The proposed method can capture diverse and more distinctive features for manipulation types not present in the training. Namely, the combination of different streams led to better manipulation-invariant representations. Moreover, the proposed framework can achieve higher generalization accuracy rates on manipulated video samples than the deep models adopted in [35] for manipulated image samples.

3.2.4. Comparison with the Existing Deepfake Detection Methods

We compared our accuracy with some of the existing methods. The comparisons can be seen in Table 5. Different methods were considered for comparisons. For instance, Yang et al. [3] suggested a methodology to detect deepfake videos using inconsistencies in the head poses of the synthesized videos based on an SVM structure on estimated 3D head orientations from each video. The UADFV dataset is used to train the SVM model in this method. The best areas under receiver operating characteristic curve (AUROC) relied on videos and frames were obtained 0.974% and 0.89%, respectively. An attribution network structure was presented by the authors of [36] to map an input image to its associated fingerprint image. As a result, they found out a model fingerprint for each source (each GAN instance plus the actual world), and the correlation index between one picture fingerprint and each model fingerprint is used as the softmax logit to classify. Their suggested method was examined on real faces from the CelebA dataset as well as synthetic faces made using some GAN techniques (ProGAN [36], SNGAN [37], CramerGAN [38], and MMDGAN [39]).

Table 4. Proposed deepfakes detection method's accuracy under novel manipulation type testing. The average row is average value of that particular cross manipulation performances.

Trained on Manipulation Type	Tested on Manipulation Type	Performance (%)
Age	Beard	88.20
	Face Swap	86.25
	Glasses	89.60
	Hair Color	76.21
	Hairstyle	75.23
	Makeup	86.30
	Smiling	92.10
	Average	84.84
Face Swap	Age	95.26
	Beard	97.81
	Glasses	94.30
	Hair Color	75.46
	Hairstyle	77.28
	Makeup	69.80
	Smiling	95.40
	Average	86.47
Glasses	Age	86.30
	Beard	82.14
	Face Swap	80.63
	Hair Color	75.26
	Hairstyle	72.44
	Makeup	71.50
	Smiling	70.78
	Average	77.01
Hair Color	Age	84.65
	Beard	66.36
	Face Swap	69.82
	Glasses	75.35
	Hairstyle	88.37
	Makeup	67.20
	Smiling	63.27
	Average	73.57
Hairstyle	Age	82.36
	Beard	73.26
	Face Swap	78.20
	Glasses	75.60
	Hair Color	83.26
	Makeup	65.46
	Smiling	69.45
	Average	75.37
Smiling	Age	77.85
	Beard	79.80
	Face Swap	82.99
	Glasses	73.21
	Hair Color	68.47
	Hairstyle	75.80
	Makeup	81.70
	Average	77.12

Table 5. Accuracy scores (%) of various methods on WPDD dataset. Bold face corresponds to the top performance.

Method	Accuracy
Yang et al. [3]	98.62%
Afchar et al. [8]	99.21%
Li et al. [9]	99.36%
Yu et al. [36]	99.63%
Proposed Method	99.80%

As seen in Table 5, the proposed method outperformed other methods in terms of accuracy. Yang et al. [36] used the 3D head poses to detect deepfake, where the central face region was particularly considered. Though the work in [36] was good for face swap detection, other types of manipulations were not considered. The authors of [8] proposed two deep networks for face tampering detection. The authors of [8] particularly focused on the mesoscopic properties of images with a low number of layers with only two class classification, but multiclass deepfake detection was not considered. Li et al. [9] presented a deep CNN model for deepfake detection. The authors of [9] exclusively concentrated on the resolution of the video to develop their methodology. They did not use any challenging datasets. Yu et al. [36] used GAN's fingerprints to detect manipulated face samples. However, the work in [36] did not study multiclass face manipulation detection. Regarding performance, the method of Yu et al. [36] produced the second-best accuracy score, where a 99.63% accuracy score was obtained. Besides, a 99.21% accuracy score was obtained for the work in [8]. The methods by Li et al. [9] and Yang et al. [3] produced 99.36% and 98.62% accuracy scores, respectively.

In this work, we considered and studied a multi-stream approach for both binary and multiclass deepfake detection. Three CNN models were incorporated in the multi-stream structure for improving the accuracy of the proposed method. The obtained results showed that the proposed structure was efficient enough to improve the obtained accuracy. Such improvement of accuracy is quite significant in real world applications, where millions of the samples are subjected to checking authenticity. One of the main reasons for the proposed method attaining higher accuracy is that it is based on ensemble learning mechanism utilizing complementary information/features. As ensemble learning classification mechanisms, which combined several base features/algorithms, have shown better performances compared to traditional methods, especially the ones using only one kind of features/algorithm [40].

3.2.5. Performance of the Proposed Method on Existing Datasets

We also analyzed the proposed method on some of the existing datasets (i.e., DeepFake-TIMIT HQ and Celeb-DF), and the performance in terms of accuracy is reported in Table 6. In [41], the authors presented the FakeCatcher for detection of deepfake. FakeCatcher takes advantage of the difference of biological signals hidden in videos to discriminate forged videos from real videos. They performed the proposed method on Face Forensics [42], Face Forensics++, CelebDF, and on a new deepfakes dataset, resulting with 96%, 94.65%, 91.50%, and 91.07% accuracies, respectively. Jafar et al. [43] introduced a deepfake detection model with mouth features (DFT-MF) that uses the mouth as a biological signal to detect deepfake videos. The proposed method evaluated on DeepfakeTIMIT LQ, DeepfakeTIMIT HQ, and Celeb-DF datasets, achieving 98.7%, 73.1%, and 71.25% accuracy rate, respectively. The robustness of our proposed method has been demonstrated by the very good results achieved, accuracy values of 99.9% and 98.2% for the DeepFake-TIMIT HQ and Celeb-DF datasets, respectively. Our results also outperformed the state-of-the-art fake detectors in [28,41,43] for the DeepFake-TIMIT HQ and Celeb-DF datasets.

As it is seen in the Table 6, the proposed method produced 99.98% and 99.95% accuracy scores for DeepFake-TIMIT HQ and Celeb-DF datasets, respectively. As our proposed method used original, blurred and sharpened versions of the input face images on a multi-

stream CNN structure, the obtained results thus were a bit improved then the compared methods. The kind of manipulations present in the dataset also affect the accuracy of the deepfake detection method, e.g., DeepFake-TIMIT HQ [4] used straightforward GANs-based algorithm to generate the manipulated samples, which are comparatively having lower quality and higher artifacts than Celeb-DF [5] dataset. Therefore, the performance of the proposed method is better for DeepFake-TIMIT HQ dataset.

Table 6. Accuracy scores (%) of proposed deep stream method on various datasets.

Dataset	Accuracy
DeepFake-TIMIT HQ [4]	99.98%
Celeb-DF [5]	99.95%

4. Conclusions

In this paper, a novel approach is proposed for fake face detection in a given video. In the proposed approach, a multi-stream deep CNN approach was developed. In the multi-stream CNN architecture, three pre-trained CNN models were combined by using a fusion layer. The inputs to the pre-trained CNN models are original, blurred, and sharpened images, respectively. Three fake face detection scenarios are considered in the experimental works. First, the fake and real face discrimination was considered. Second, seven face manipulations (i.e., age, beard, face swap, glasses, hair color, hairstyle, smiling, and real face discrimination) were carried out. Third, performance of deepfake detection system under novel type of face manipulation is evaluated. The experimental results show that the proposed method outperformed the prior methods. As a future work, we will investigate the robustness of the proposed system against adversarial attacks and face orientations, and thereby devise more robust frameworks.

Author Contributions: Conceptualization, Z.A. and A.S.; methodology, Z.A., A.S. and S.Y.; software, S.Y.; validation, S.Y. and A.S.; formal analysis, A.S.; investigation, S.Y.; resources, S.Y. and Z.A.; data curation, S.Y.; writing—original draft preparation, S.Y.; writing—review and editing, A.S., Z.A. and K.S.; supervision, Z.A. and A.S.; funding acquisition, K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Xiamen University Malaysia Research Fund (Grant No: XMUMRF/2019-C3/IECE/0006).

Institutional Review Board Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Akhtar, Z.; Dasgupta, D. A Comparative Evaluation of Local Feature Descriptors for DeepFakes Detection. In Proceedings of the IEEE International Symposium on Technologies for Homeland Security (HST), Woburn, MA, USA, 5–6 November 2019; pp. 1–5.
2. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 1–9.
3. Yang, X.; Li, Y.; Lyu, S. Exposing deep fakes using inconsistent head poses. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 1–4.
4. Korshunov, P.; Marcel, S. Deepfakes: A new threat to face recognition? assessment and detection. *arXiv* **2018**, arXiv:1812.08685.
5. Li, Y.; Yang, X.; Sun, P.; Qi, H.; Lyu, S. Celeb-DF: A new dataset for DeepFake forensics. *arXiv* **2019**, arXiv:1909.12962.
6. Deep Fake Detection Dataset. Available online: https://ai.googleblog.com/2019/09/contributing-data-to-deepfake-detection.html (accessed on 25 June 2021).
7. Dolhansky, B.; Howes, R.; Pflaum, B.; Baram, N.; Ferrer, C. The Deepfake Detection Challenge (DFDC) Preview Dataset. *arXiv* **2019**, arXiv:1910.08854.
8. Afchar, D.; Nozick, V.; Yamagishi, J.; Echizen, I. Mesonet: A compact facial video forgery detection network. In Proceedings of the IEEE International Workshop on Information Forensics and Security (WIFS), Hong Kong, China, 11–13 December 2018; pp. 1–7.
9. Li, Y.; Lyu, S. Exposing DeepFake videos by detecting face warping artifacts. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA, 16–17 June 2019; pp. 1–7.

10. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1–12.
11. Zhou, P.; Han, X.; Morariu, V. I.; Davis, L.S. Two-stream neural networks for tampered face detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017; pp. 1–9.
12. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–12.
13. McCloskey, S.; Albright, M. Detecting GAN-generated imagery using color cues. *arXiv* **2018**, arXiv:1812.08247.
14. Guan, H.; Kozak, M.; Robertson, E.; Lee, Y.; Yates, A.; Delgado, A.; Zhou, D.; Kheyrkhah, T.; Smith, J.; Fiscus, J. MFC datasets: Largescale benchmark datasets for media forensic challenge evaluation. In Proceedings of the IEEE Winter Applications of Computer Vision Workshops, Waikoloa Village, HI, USA, 7–11 January 2019; pp. 63–72.
15. Nataraj, L.; Mohammed, T.; Manjunath, B.; Chandrasekaran, S.; Flenner, A.; Bappy, J.; Roy-Chowdhury, A. Detecting GAN generated fake images using co-occurrence matrices. *arXiv* **2019**, arXiv:1903.06836.
16. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv* **2017**, arXiv:1703.10593.
17. Choi, Y.; Choi, M.; Kim, M.; Ha, J.; Kim, S.; Choo, J. StarGAN: Unified generative adversarial networks for multi-domain imageto-image translation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8789–8797.
18. Neves, J.; Tolosana, R.; Vera-Rodriguez, R.; Lopes, V.; Proenca, H. Real or fake? spoofing state-of-the-art face synthesis detection systems. *arXiv* **2019**, arXiv:1911.05351.
19. Nguyen, H.H.; Yamagishi, J.; Echizen, I. Use of a capsule network to detect fake images and videos. *arXiv* **2019**, arXiv:1910.12467.
20. Sabour, S.; Frosst, N.; Hinton, G.E. Dynamic routing between capsules. In Proceedings of the Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 1–11.
21. Rossler, A.; Cozzolino, D.; Verdoliva, L.; Riess, C.; Thies, J.; Nießner, M. Faceforensics++: Learning to detect manipulated facial images. *arXiv* **2019**, arXiv:1901.08971.
22. Matern, F.; Riess, C.; Stamminger, M. Exploiting visual artifacts to expose DeepFakes and face manipulations. In Proceedings of the IEEE Winter Applications of Computer Vision Workshops (WACVW), Waikoloa Village, HI, USA , 7–11 January 2019; pp. 83–92.
23. Liu, Z.; Luo, P.; Wang, X.; Tang, X. Deep learning face attributes in the wild. In Proceedings of the IEEE International Conference on Computer Vision Workshop (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1–11.
24. Guarnera, L.; Giudice, O.; Battiato, S. DeepFake Detection by Analyzing Convolutional Traces. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 1–10.
25. Moon, T.K. The Expectation-Maximization Algorithm. *IEEE Signal Process. Mag.* **1996**, *13*, 47–60. [CrossRef]
26. He, Z.; Zuo, W.; Kan, M.; Shan, S.; Chen, X. AttGAN: Facial Attribute Editing by Only Changing What You Want. *IEEE Trans. Image Process.* **2019**, *28*, 5464–5478. [CrossRef] [PubMed]
27. Cho, W.; Choi, S.; Park, D.K.; Shin, I.; Choo, J. 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, Long Beach, CA, USA, 15–20 June 2019; pp. 1–15.
28. Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; Aila, T. Analyzing and Improving the Image Quality of StyleGAN. In Proceedings of the IEEE/CVF Conference on Computer Vision and Patter Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 1–21.
29. Hernandez-Ortega, J.; Tolosana, R.; Fierrez, J.; Morales, A. Deepfakeson-phys: Deepfakes detection based on heart rate estimation. In Proceedings of the 35th AAAI Conference on Artificial Intelligence Workshops, Online, 17 April 2021; pp. 1–8.
30. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015; pp. 1–14.
31. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
32. Deniz, E.; Şengür, A.; Kadiroğlu, Z.; Guo, Y.; Bajaj, V.; Budak, U. Transfer learning based histopathologic image classification for breast cancer detection. *Health Inf. Sci. Syst.* **2018**, *6*, 1–7. [CrossRef] [PubMed]
33. Orenstein, E.C.; Beijbom, O. Transfer Learning and Deep Feature Extraction for Planktonic Image Data Sets. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, USA, 24–31 March 2017; pp. 1082–1088.
34. Viola, P.A.; Jones, M.J. Rapid Object Detection using a Boosted Cascade of Simple Features. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Kauai, HI, USA, 8–14 December 2001; pp. 1–8.
35. Akhtar, Z.; Mouree, M.R.; Dasgupta, D. Utility of Deep Learning Features for Facial Attributes Manipulation Detection. In Proceedings of the IEEE International Conference on Humanized Computing and Communication with Artificial Intelligence (HCCAI), Irvine, CA, USA, 21–23 September 2020; pp. 55–60.
36. Yu, N.; Davis, L.; Fritz, M. Attributing Fake Images to GANs: Analyzing Fingerprints in Generated Images. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 1–11.

37. Miyato, T.; Kataoka, T.; Koyama, M.; Yoshida, Y. Spectral Normalization for Generative Adversarial Networks. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018; pp. 1–26.
38. Bellemare, M.; Danihelka, I.; Dabney, W.; Mohamed, S.; Lakshminarayanan, B.; Hoyer, S.; Munos, R. The Cramer Distance as a Solution to Biased Wasserstein Gradients. *arXiv* **2017**, arXiv:1705.10743.
39. Binkowski, M.; Sutherland, D.; Arbel, M.; Gretton, A. Demystifying MMD GANs. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018; pp. 1–36.
40. Ganaie, M.A.; Hu, M.; Ensemble deep learning: A review. *arXiv* **2021**, arXiv:2104.02395.
41. Ciftci, U.A.; Demir, I.; Yin, L. FakeCatcher: Detection of Synthetic Portrait Videos Using Biological Signals. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, 1–17. [CrossRef] [PubMed]
42. Rossler, A.; Cozzolino, D.; Verdoliva, L.; Riess, C.; Thies, J.; Nießner, M. FaceForensics: A Large-scale Video Dataset for Forgery Detection in Human Faces. *arXiv* **2018**, arXiv:1803.09179.
43. Jafar, M.T.; Ababneh, M.; Al-Zoube, M.; Elhassan, A. Forensics And Analysis Of Deepfake Videos. In Proceedings of the 11th IEEE International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 7–9 April 2020; pp. 53–58.