applied sciences

MDPI

Article

High Performance DeepFake Video Detection on CNN-Based with Attention Target-Specific Regions and Manual Distillation Extraction

Van-Nhan Tran [1], Suk-Hwan Lee [2], Hoanh-Su Le [3] and Ki-Ryong Kwon [1,*]

[1] Department of Artificial Intelligence Convergence, Pukyong National University, Busan 48513, Korea; tvnhanpk@pukyong.ac.kr
[2] Department of Computer Engineering, Dong-A University, Busan 49315, Korea; skylee@dau.ac.kr
[3] Faculty of Information Systems, University of Economics and Law, Vietnam National University Ho Chi Minh City, Ho Chi Minh 700000, Vietnam; sulh@uel.edu.vn
* Correspondence: krkwon@pknu.ac.kr; Tel.: +82-10-5720-3838

Abstract: The rapid development of deep learning models that can produce and synthesize hyper-realistic videos are known as DeepFakes. Moreover, the growth of forgery data has prompted concerns about malevolent intent usage. Detecting forgery videos are a crucial subject in the field of digital media. Nowadays, most models are based on deep learning neural networks and vision transformer, SOTA model with EfficientNetB7 backbone. However, due to the usage of excessively large backbones, these models have the intrinsic drawback of being too heavy. In our research, a high performance DeepFake detection model for manipulated video is proposed, ensuring accuracy of the model while keeping an appropriate weight. We inherited content from previous research projects related to distillation methodology but our proposal approached in a different way with manual distillation extraction, target-specific regions extraction, data augmentation, frame and multi-region ensemble, along with suggesting a CNN-based model as well as flexible classification with a dynamic threshold. Our proposal can reduce the overfitting problem, a common and particularly important problem affecting the quality of many models. So as to analyze the quality of our model, we performed tests on two datasets. DeepFake Detection Dataset (DFDC) with our model obtains 0.958 of AUC and 0.9243 of F1-score, compared with the SOTA model which obtains 0.972 of AUC and 0.906 of F1-score, and the smaller dataset Celeb-DF v2 with 0.978 of AUC and 0.9628 of F1-score.

Keywords: DeepFake detection; computer vision and pattern recognition; artificial intelligence

Citation: Tran, V.-N.; Lee, S.-H.; Le, H.-S; Kwon, K.-R. High Performance DeepFake Video Detection on CNN-Based with Attention Target-Specific Regions and Manual Distillation Extraction. *Appl. Sci.* **2021**, *11*, 7678. https://doi.org/10.3390/app11167678

Academic Editor: Byung-Gyu Kim

Received: 3 August 2021
Accepted: 17 August 2021
Published: 20 August 2021

1. Introduction

DeepFake is a type of artificial intelligence which is used to produce convincing pictures, audio, and video forgeries. The main methods used to construct DeepFakes are based on deep learning and correlate training Generative Neural Networks (GANs) [1] architectures. Generative Adversarial Networks (GAN) are deep learning techniques for training generative models, which are most commonly used for the generation of synthetic images. The GAN model architecture involves two sub-models: a generator model for generating new examples and a discriminator model for classifying whether the generated examples are real or fake. The growth of GAN lead to the development of a series of applications and sophisticated techniques, such as face swapping, face manipulation, and face synthesis, resulting in a rapidly increasing number of fake videos with accurate quality and more complexity. The results of the DeepFake generation have become increasingly realistic in recent years, making it harder to isolate the real from the fake for the normal eyes. Multimedia content that has been tampered with is increasingly being

utilized in a variety of cybercrime operations (also mentioned in Ferreira et al. research [2]). Fake news, disinformation, digital kidnapping, and ransomware-related crimes are only a few of the most common crimes perpetrated and disseminated using altered digital pictures and videos.

DeepFake detection solutions usually use multimodal detection approaches to evaluate whether target material has been altered or created synthetically. Existing detection approaches often focus on developing AI-based algorithms in algorithmic detection methods such as Vision Transformer [3,4], two-stream neural network [5], MesoNet (which is proposed by Afchar et al. [6]), etc. However, less attention is paid to manual image processing to focus on highlighting the important regions of an image. This often results in the model having to process all of the videos, making the model heavier. In order to improve the DeepFake detection approach, we used both a manual processing and an AI-based algorithm in this research. The most important information, regions, and features will be carefully focused and processed before being put into deep neural networks. Concentrating on the most relevant elements to learn not only reduces the needless learning burden on these networks, but also improves the overall model's accuracy.

The major concept of our article will be to take advantage of few most popular classification models to identify fake videos and show how to transform the DeepFake detection into a simpler classification problem. As the existing classification models are designed for high accuracy, the reasonable selection of these models will also increase the ability to solve the problem of DeepFake detection. We also propose processing methods to convert the input sequences of DeepFake detection into the inputs of a basic classification model with two classes (class "1" for real video and class "0" for fake video). Our proposal also accommodates the processing steps to avoid losing essential features and supports synthesizing afterwards.

2. Related Work

2.1. Face Forgery Generation

Face forgery generation is one of the fields of image synthesis. The objective is to create new faces using generative adversarial networks (GANs) [1]. The most popular approach is StyleGAN [7], which makes it possible to control the image synthesis via scale-specific modifications to the styles. Even with the growth of StyleGAN2 [8], which is based on data-driven unconditional generative image modeling, non-existent lifelike faces can be made with near-real sophistication and are often indistinguishable fake people who don't actually exist in real life. An application to create non-existent lifelike face of peoples using StyleGAN2 is mentioned in this tool [9]. The algorithms underpinning the AI are trained on publicly available pictures before being asked to generate fresh variants that satisfy the requisite level of realism. In addition, many synthetic programs [10] are available as open source and may be used by anyone.

Face swapping is the most common type of face modification currently. The DeepFake face swap method is built on two Auto-encoders [11], where one common encoder is used in training and rebuilding sources and another one to target face training pictures, respectively. The aim of face swapping is to generate a new fake image or video of a person after swapping its face. Presently, a variety of approaches have been proposed. Some prominent methods can be mentioned such as RSGAN [12], FSGAN [13], DCGAN [14], PGGAN [15], FSNet [16], High Fidelity Identity Swapping [17], and StarGAN v2 [18]. A lot of datasets were created based on face swaps, of which the standout is the DeepFake Detection Challenge [19] (DFDC) dataset. In addition, face manipulation was also used in face forgery generation such as MulGAN [20], MaskGAN [21], PuppetGAN [22], and HistoGAN [23].

2.2. Face Forgery Detection

Various techniques have been developed to identify fake synthetic pictures. In the research on detection of face manipulation [24], the authors used attention layers on top of feature maps to extract the manipulated regions of the face. In the research of Liu et al. [25], they presented a novel spatial-phase shallow learning (SPSL) method, which combines spatial image and phase spectrum to capture face forgeries upsampling artifacts and enhance transferability. Zhao et al. [26] formulated DeepFake detection as a fine-grained classification problem and proposed a new multi-attentional DeepFake detection network that can be more efficient for collecting local important features. Mazaheri et al. [27] suggested a system that uses image manipulation approaches and a mix of facial expression recognition to identify modification and localization in facial expression. Chen et al. [28] proposed a light weight DeepFake detection network by using the successive subspace learning (SSL) principle from various parts of face images to automatically extract important features. Dynamic face augmentation was utilized to improve the DeepFake detection model's performance and prevent model overfitting owing to the large number of data generated from a limited number of distinct objects by Das et al. [29]. Kim et al. [30] presented a detection approach that combined color distribution analysis of the vertical edge region in a modified picture with pixel correction techniques to reduce the discomfort of pixel value discrepancies. In addition, a lot of proposals related to vision transformer were published, which can be mentioned by Wodajo et al. [3]. They proposed a convolution vision transformer, which is a combined convolutional neural network (CNN) and vision transformer (ViT) to learn features and categorization by using an attention mechanism. Heo et al. [4] proposed a model with EfficientNetB7 [31] as a basic backbone.

All of these methods were mentioned previously based on deep learning. There are variety of approaches relied on non-deep learning-based methods too. Yang et al. [32] used SVM classifier to detect splicing generated facial region. Matern et al. [33] applied multi-layer perceptron (MLP) classifiers and extracted the landmarks to detect fake videos. Besides, some researches were proposed a face friend-safe adversarial for recognition face system as well as protect system by Kwon et al. [34,35].

3. Proposed Methodology

In this section, we describe the architecture for DeepFake detection based on the classifier network with manual attention target-specific regions to create distillation set, which not only can improve the accuracy of classification using neural networks, but also allows the use of a lighter backbone. We introduce some steps in image processing to manually create a set of important data which is called manually distillation set in this paper and focus on special regions in Section 3.1. In addition, we also provide a model structure that we have used as a normal image classification model, which can generate features for each domain, to facilitate synthesis in the next step. In Section 3.3, we discuss on how to merge several frames and multi-regions in images before deciding on the final result.

3.1. Image Preprocessing

The objective of this section is to preprocess the picture before it is fed into the next stages. This part is critical because it influences the quality of the entire process moving ahead. It also improves the quality of the entire process through data processing.

3.1.1. Face Extraction

The first step in this process is person detection [36], which is implemented by the open library "OpenCV" [37]. Person detection is required because it helps identifying the real person's face. It also helps avoiding situations when faces are from a non-real person or any object (something identical to a person such as statue). This phase aims to reduce

face detection mistakes as much as possible too. After identifying the main people in video, we need to extract their faces from these people that have been detected. Multitask cascaded convolutional networks [38] (MTCNN) are then used to extract the face. These steps will significantly reduce the amount of misleading data in the dataset that is used to fool the model. An example for the inclusion of these fictitious photographs is shown in Figure 1. Typically, some approaches just concentrate on face extraction such as Wodajo et al**.** [3], Heo et al. [4], Selim et al. [39], etc. These models sometimes might identify the face of a non-real person and that could significantly adversely affect the quality of the model.

Figure 1. The example images are taken from a video with label is "REAL" of DFDC [19] dataset. Including real person along with many small pictures contain fake faces that are not the main person. Two images were captured at different time in the same video.

Person detection is currently very popular with several models that are far superior to the "OpenCV" model, such as high-level semantic feature detection [40] and person detection for far field [41]. However, the reason that a library that is not very effective at person detection was picked since it strictly constrains the input data in detecting person in order to get good extracted faces in future output. On account of that, the confidence level for recognizing those faces is likewise set to high while utilizing the MTCNN [38] face extraction library.

3.1.2. Face Augmentation

The overfitting problem is always considered carefully in DFDC datasets. When a model learns the information and noise in the training data to the point then it degrades the model's performance on new data, this is known as overfitting. This means that the model learns too well with the training data for the DFDC dataset, but the outcomes with the test data are not as good as expected. One method for resolving this issue is to use augmentation [42]. In terms of increasing this quality, previous research has also found that data augmentation can help to mitigate this negative effect [29]. This is a crucial approach for generating more usable data and improving the model's quality during training. The methods of augmentation used in this proposal are mostly focused on information dropping [43,44], illustrated in Figure 2. It mainly focuses on the meaningful regions of the face to distinguish the real from the fake, such as the eyes, nose and mouth. These important regions are illustrated in Figure 3. Important regions are dropped randomly to increase data diversity.

Figure 2. (**a**) The original face without any augmentation. (**b–d**) Three types of augmentation: cut-out of mouth, eyes and nose.

3.1.3. Patch Extraction

Several essential characteristics of the face, notably the eyes, nose, and mouth, which are difficult to represent, are used to distinguish between fake and real faces. This content is also emphasized in the research content [45]. So as to solve this problem in the preprocessing section it will be handled tightly, facial landmarks are extracted from each extracted face by using "OpenFace2" [46] an open-source toolbox. As the input to our networks, we harvest regions of interest (RoI). Our goal is to reveal the artifacts that exist between the false face and the surrounding. The rectangular regions that encompass both the face and surrounding areas are chosen as the RoIs. Specifically, we use facial landmarks to determine the RoIs. The rectangular regions box can be defined as equation (1).

$$[x_0 - \hat{x}_0, y_0 - \hat{y}_0, x_1 + \hat{x}_1, y_1 + \hat{y}_1] \quad (1)$$

where $x_0, y_0, x_1, and\ y_1$ indicate the smallest bounding box capable of covering all facial landmarks. The variables $\hat{x}_0,\ \hat{y}_0, \hat{x}_1, \hat{y}_1$ are random values between [0, $\frac{w}{6}$] and [0, $\frac{h}{6}$], where h and w are the height and width of the extracted face box, respectively.

After extracting facial landmarks, patches of face are cropped from different parts of face, including the left eye, the right eye, nose and mouth. The RoIs are resized to 32 × 32.

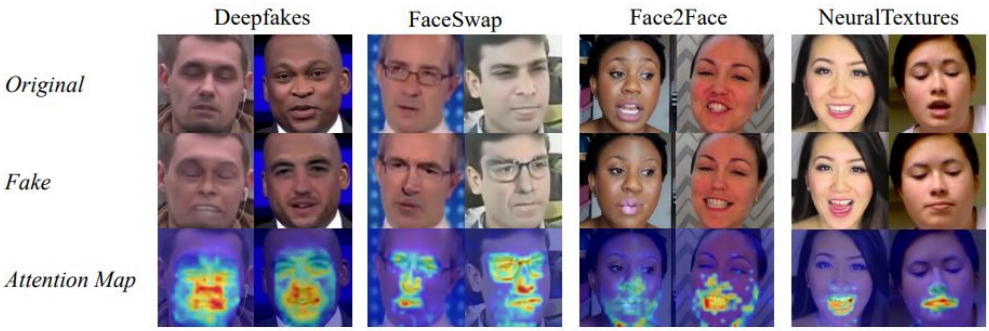

Figure 3. The figure from research of Miao et al. [45] show the important information allocated in the specific regions of face. The original row contains the original face image. Second row is a set of manipulated face samples picked at random from the FF++ [47] testing set. And last row shows that the attention maps of manipulated face samples. DeepFakes, FaceSwap, Face2Face, and NeuralTextures columns are the four manipulation methods.

3.1.4. Distillation Set

The idea of distillation through attention method [48] by Facebook AI, they introduced a teacher-student strategy specific transformers. It was based on a distillation token that ensures the student learns from the teacher through attention. It automatically generates a distillation token, which functions similarly to a class token in that it interacts

with other embeddings via self-attention. It allows their model to learn from the output of the teacher through the self-attention layers. Instead of using model self-attention throughout the training process to instruct the model focusing on important regions we used the idea to develop our model which is able to learn locally during preprocessing steps before the actual training process. Distillation set including a lot of patches which are grouped into sets to designate which parts of the face will be trained, as well as which pieces of the face will be learnt at which classifier network input. This arrangement into distillation set increases training accuracy and makes it convenient for region ensemble.

3.2. Classifier Network Architecture

We obtain special features and distillation sets containing a lot of useful information for training and inference through previous steps. These materials which are used to differentiate the real from the fake in a frame concentrated mostly on the face and the important regions in it, and thus training the entire frame will be unnecessary. If we use the all information in an image, it will dilute the essential information that the model needs to focus on. This may lead to an adverse effect on the quality of the model or cause the size of model to grow. At the input of classifier network are distillation sets, and each distillation set will contain extracted faces, face augmentation, multi-region of face in patches along with "Real" or "Fake" label. In this step, the issues essentially become two-class classification problems.

The design is depicted in the Figure 4. In this section, two main backbones, InceptionV3 [49] and MobileNet [50] are utilized without last fully connected layers. The InceptionV3 network will be used to train the entire face as well as the outcomes of the face augmentation, while the MobileNet component will train face patches. The output of these two components is concatenated and fed into global average pooling. This was suggested by Min Lin [51]. Now we have feature sets, which contain information to discriminate between actual and fake faces, as well as significant areas.

If we analyze and evaluate the output of the classifier network (feature sets) to make decision as "Real" or "Fake", we will get DeepFake detection for images. However, because the frames and regions are not combined, the results are often lower than the model with combination and classification module.

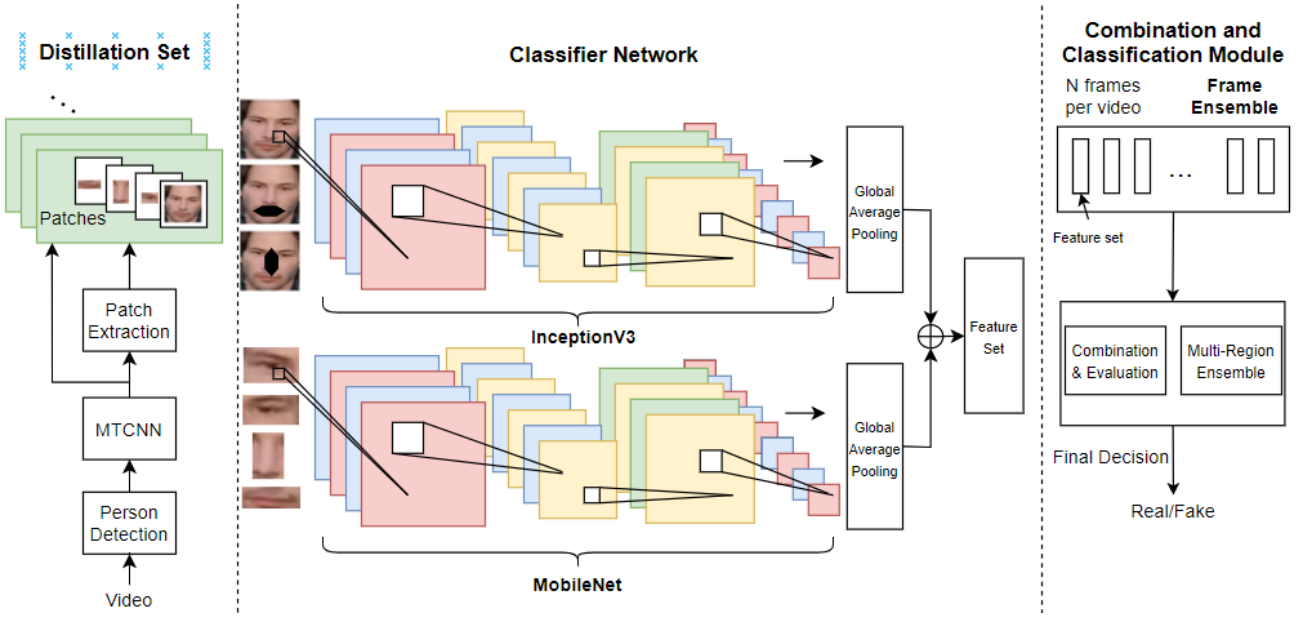

Figure 4. An overview of the proposed DeepFake detection method.

3.3. Combination and Classification Module

This part will choose and synthesize feature sets of frames in a video, synthesize regions in faces and change virtual threshold levels to make final decision for videos. As depicted in the Figure 4.

3.3.1. Frame Ensemble

The model gathered feature sets of N frames per video. By using only a subset of the frames, we will reduce the size of input. Due to a large variation in video lengths, extracting N frames per second would also result in a large variation in input lengths. Although this could be solved with padding, we would end up with some input mostly composed of zeros, which in turn could lead to poor training performance. Therefore, N frames per video, which did not depend on the video length was selected. An example for comparison of two video sampling methods is shown in Figure 5.

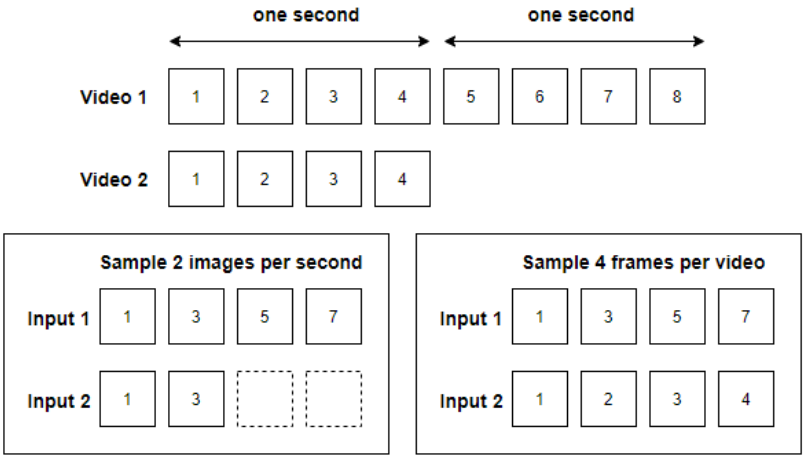

Figure 5. Comparison of two video sampling methods. Dotted rectangles indicate zero padding.

3.3.2. Multi-Region Ensemble

N frames after being synthesized will be put into multi-region ensemble for synthesis and evaluation, region information is also collected from feature sets and then pooled to derive video classification results while evaluating if the video is real or fake.

The majority of prior research has relied on a threshold parameter, which represents the probability of fake video to synthesize information between frames in a video in order to make a final decision about whether the video is real or fake. This threshold parameter is generally set to a fixed value, which might make the model more sensitive to fake or real video, lowering the overall model quality. In our proposal, this threshold for distinguishing real and fake will be dynamically changed to match the trained parameters to solve this problem, enhancing the capacity to properly predict the outcome.

In this section we use a simple logistic regression model with the input and where the input is portrayed by a value for each video along with ground truth label of video. It can be represented as $X = [x_1, x_2, \dots, x_M] \in \mathbb{R}^{d\,x\,M} = \left[\frac{a_1}{N}, \frac{a_2}{N}, \dots, \frac{a_M}{N}\right]$ and $Y = [y_1, y_2, \dots, y_M]$, where M is number of samples; N is selected frames per video for combination and classification module, a_i is representation of the probability of fake video value.

The output of logistic regression is written in the following format (2):

$$f(x) = \theta(w^T X) \quad (2)$$

where θ is called logistic function, w is weight value.

We assume that the probability that a data x falls into class 1 is $f(w^T x)$, x falls into class 0 is $1 - f(w^T x)$, we can represent it as follows (3):

$$P(y_i|x_i; w) = z_i^{y_i}(1 - z_i)^{1-y_i} \quad (3)$$

$P(y_i|x_i; w) = z_i^{y_i}(1 - z_i)^{1-y_i}$where $z_i = f(w^T x_i)$.

In this case, we use the Stochastic Gradient Descent (SGD). Loss function with (x_i, y_i) is (4):

$$J(w; x_i, y_i) = -(y_i log z_i + (1 - y_i)\log(1 - z_i)) \quad (4)$$

3.3.3. Limitation

The accuracy of this part is highly reliant on the quality of the classifier network in the previous section. If the classifier network's accuracy is high, the whole model's quality will improve considerably; nevertheless, if this part's quality is low, it will have a resonance effect, causing the results deteriorate.

4. Experiments

This section describes the dataset as well as the parameters in depth. We also compare to the SOTA model and some current models. In Section 4.1, we will discuss the datasets. In addition, we discuss the parameter setup and configuration environment necessary in the training process in Section 4.2, and in Section 4.3, the outcomes of the experiment will be analyzed, and the proposals will be compared to existing researches.

4.1. Dataset

We evaluate the performance of our model on two DeepFake video datasets: the smaller dataset Celeb-DF v2 [52], and the bigger dataset DFDC [19]. The goal of testing on both types of datasets is to check that the model works effectively on both small and large datasets, as well as to compare the differences in output between the two datasets.

Celeb-DF v2 [52] dataset contains real and DeepFake synthesized videos. This dataset is greatly extended from previous Celeb-DF v1. Currently, Celeb-DF v2 includes 590 original videos collected from YouTube with subjects of different ages, ethnic group, and genders, and 5639 corresponding DeepFake videos.

DFDC [19] dataset is currently the largest DeepFake dataset, which is available on the Internet, with over 100,000 total. These clips were generated by face swap dependent on GAN. Inference for the DFDC dataset, we evaluated on the 5000 test dataset.

4.2. Training Evaluation

4.2.1. Image Preprocessing

As explained in Section 3, we use MTCNN with strict conditionals, extracting only if confidence is greater than 90 percent and cropping and resizing extracted faces to 128 × 128. Then 68 facial landmarks are detected. Based on these facial landmarks, the left eye, the right eye, nose, and mouth are cropped with dimensions of 32 × 32 for each one.

4.2.2. Training Parameters

We train classifier Network with batch size of 64 and 30 epochs. We use Adam optimizer with learning rate is 0.001, the exponential decay rate for the 1st moment estimates beta_1 is 0.9, and for 2nd moment estimates is 0.999, without epsilon arguments. In the Combination and Classification Module, we choose 40 frames per video. We didn't use all the frames from the video as at 25 or 30 frames per second, most of the frames look alike. By using only a subset of the frames, we will reduce the size of our input and therefore speed up the training process. Furthermore, in Celeb-DF v2 and DFDC dataset, most of the videos are pretty short (usually a few seconds), and therefore we suggest a 40 frames per video which is appropriate for the length of the video while also ensuring sufficient

training time. Regarding hardware parameters, we trained and evaluated on NVIDIA GeForce RTX 2080 Ti Graphics Card.

4.2.3. Evaluation Metrics

It is extremely useful for measuring Recall, Precision, F1-score, and most importantly AUC-ROC curves. It can be calculated through the confusion matrix shown in Table 1.

Table 1. Confusion matrix.

		Predicted	
		Negative	Positive
Actual	Negative	True Negative (TN)	False Positive (FP)
	Positive	False Negative (FN)	True Positive (TP)

Each column in the confusion matrix represents instances in a predicted class, while each row represents instances in an actual class. The positive class refers to the original and unmanipulated images, while the negative class represents the manipulated ones. True positive (TP) represents the outcome where the model correctly predicts the positive class. Similarly, true negative (TN) represents the outcome where the model correctly predicts the negative class. False positive (FP) represents the outcome where the models incorrectly predict the positive class. And false negative (FN) represents the model incorrectly predicts the negative class.

Precision measures the percentage of positive identifications was actually correct. Precision is defined as (5):

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

Recall measures the percentage of actual positives was identified correctly. Recall is defined as (6):

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

F1-score is the harmonic mean of precision and recall. It is difficult to compare two models with low precision and high recall or vice versa. F1-score helps to measure Recall and Precision at the same time. F1-score is defined as (7):

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{7}$$

True positive rate (TPR) is a synonym for recall and is defined as (8):

$$TPR = \frac{TP}{TP + FN} \tag{8}$$

False positive rate (FPR) is defined as (9):

$$FPR = \frac{FP}{FP + TN} \tag{9}$$

Accuracy measures the percentage of number of correct predictions. Accuracy is defined as (10):

$$Accuracy = \frac{Number\ of\ correct\ predictitons}{Total\ number\ of\ predictions} = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

An ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model. This curve plots with two parameters: true positive rate (TPR) and false positive rate (FPR). AUC (area under the ROC curve) measures the

two-dimensional area under the ROC curve. AUC provides an aggregate measure of performance across all possible classifications. With AUC, we can measure how well predictions are ranked, rather than their absolute values since AUC is scale-invariant.

Training results of the classifier network are shown in Figures 6 and 7. This step basically becomes a simple case of classification model with two classes ("Real" and "Fake"), therefore it doesn't take a long time in this section. Most of the time will be spent on the distillation set extraction. The training outputs of the Celeb-DF v2 dataset is shown in Figure 6 and DFDC dataset is shown in Figure 7. The performance of the training set is usually better than the validation set, for example, the accuracy of the training set is approximately equal to "1". The accuracy of the validation set is not too much lower than the training set, detailed values are listed in Table 2. This demonstrates that the model has partially solved the overfitting problem mentioned in Section 3.1.

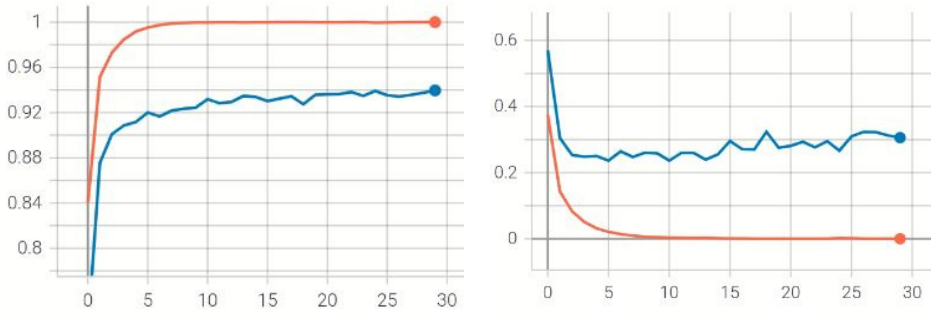

Figure 6. The training output of classifier network on the Celeb-DF v2 dataset. The accuracy of the model is on the left, while the loss of the model is on the right. The red represents the training outcomes, while the blue represents the validation outcomes.

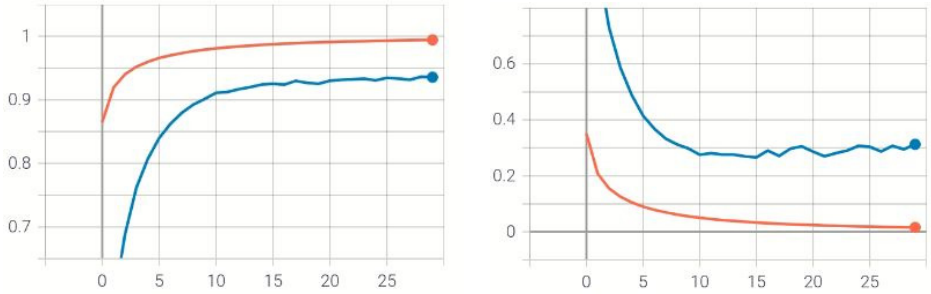

Figure 7. The training output of classifier network on the DFDC dataset. The accuracy of the model is on the left, while the loss of the model is on the right. The red represents the training outcomes, while the blue represents the validation outcomes.

We can also measure frame-by-frame quality using feature sets at the output of the classifier network. However, we are primarily interested in DeepFake detection for video, so the results of classifier network will be used in the combination and classification module.

Table 2. Training results of the classifier network on validation set of Celeb-DF v2 and DFDC.

	Accuracy of Validation Set	Loss of Validation Set
Celeb-DF v2	94.2%	0.3
DFDC	93.75%	0.295

4.3. Performance Evaluation

Figure 8 illustrates the ROC curve related to the processing of the validation set, the good performance of the classifier can also be seen, in which the curve is pushed to the upper left corner of the graphic. This means that the area under the ROC is also higher.

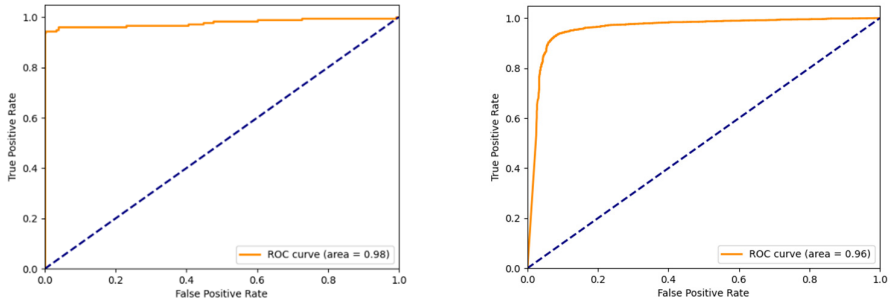

Figure 8. ROC-AUC curve. The left side is the curve of Celeb-DF v2 validation set, the right side is the curve of DFDC validation set.

Figure 9 illustrates the confusion matrix of our model with the Celeb-DF v2 dataset. Our model is compared to the current state-of-the-art model [39]. The model's performance is equivalent to SOTA. We used the train dataset to train our model and the validation set to test it. In Figure 10, the confusion matrix of our model is compared to that of the SOTA model [39]. Our model has a relatively comparable 0.958 of AUC and 0.9243 of F1-score to the SOTA model, which had an AUC of 0.972 and 0.906. However, since SOTA is built on Efficient Network, which is often very heavy, our proposed method is better in terms of a lighter architecture while maintaining the same level of quality.

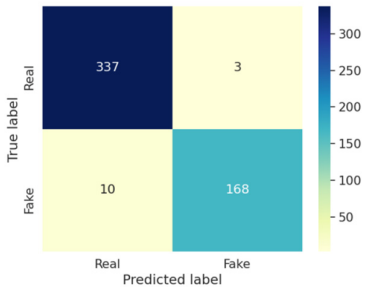

Figure 9. Confusion Matrix of our model with the Celeb-DF v2 dataset [52]. Top-right, top-left, bottom-right, bottom-left means false-positive, true-negative, true-positive, false-negative in order.

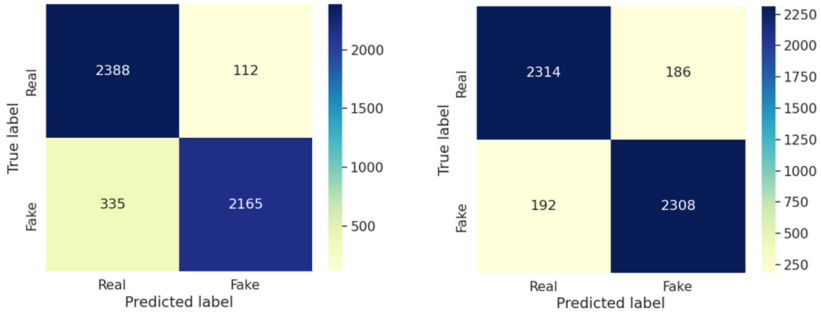

Figure 10. Confusion Matrix with DFDC dataset [19]; the left side is the SOTA model's prediction which is described in Heo et al. [4] results and the right side is our results. Top-right, top-left, bottom-right, bottom-left means false-positive, true-negative, true-positive, false-negative in order.

The comparison of AUC values is shown in Table 3. For Celeb-DF v2 dataset, the majority of researches in [5,6,47,53,54] may not have a high AUC. Outstanding research can be mentioned is Chen et al. [28], which achieved a pretty good result of 90.56% AUC with this dataset. Not only that, but their outcomes are also superior in terms of a number of parameters and their research goal is to create a lightweight model that can operate on systems with minimal hardware requirements. In this section, our method achieved an AUC of 97.8%, which is somewhat slightly better than theirs. Our fundamental model is based on a deep neural network for classification; thus, the number of parameters is frequently larger. For DFDC dataset, the best results can be mentioned in the research of Heo et-al [4] with 97.8% AUC. On the other hand, the idea of Young-Jin research is similar to SOTA model [39], they are also based on the Vision Transformer model, EfficientNet with distillation methodology. Thus, their results must have a large number of parameters although they are not mentioned in their research contents. In this section our results are a bit lower than theirs, but we will undoubtedly have the advantage in terms of the number of parameters.

Table 3. Comparison of the detection performance of benchmarking methods with the AUC value. The AUC results of benchmarking methods are taken from [4,28,52].

	Celeb-DF v2	DFDC	Number of Parameters
Zhou et al. (2017) [5]	53.8%	61.4%	24 M
Afchar et al. (2018) [6]	54.8%	75.3%	27.9 K
Rossler et al. (2019) [47]	48.2%	49.9%	22.8 M
Nguyen et al. (2019) [53]	57.5%	53.3%	3.9 M
Li et al. (2020) [54]	64.6%	75.5%	-
Chen et al. (2021) [28]	90.56%	-	42.8 K
Heo et-al. (2021) [4]	-	97.8%	-
Ours	**97.8%**	**95.8%**	**26 M**

Table 4 depicts the results obtained with the benchmarking of the Celeb-DF v2 dataset and DFDC dataset. Notable in the table is the F1-score of DFDC, we can see that the proposed model is robust in keeping fair assessment between fake videos and real videos and the F1-score was 0.9243, which was slightly higher than 0.919 of Heo et-al [4] proposal. This might be due to dynamic threshold which is described in Section 3.3.

Table 4. Benchmark videos.

	Precision	Recall	F1-Score	Accuracy
Ours–Celeb-DF v2	0.9825	0.9438	0.9628	0.9749
Ours–DFDC	0.9254	0.9232	0.9243	0.9244

5. Conclusions

In this paper, we have proposed a method for DeepFake detection. A model consisted of CNN network with high-performance and lighter weight model compared to current models such as SOTA [39] and combined vision transformer and EfficientNet [4]. We found an 0.958 of AUC, 0.9243 of F1-score for the DFDC dataset and 0.978 of AUC, 0.9628 of F1-score for Celeb-DF v2 with 26M parameters. By combining a manual technique with an AI-based algorithm, we improved the DeepFake detection method. The most important information, regions, and features were carefully cleaned and processed before being put into deep neural networks. Important preprocessing steps were also recommended to improve the model's quality considerably. In the future work, we will look for ways to enhance the model so that we can continue to propose lighter models while also improving its accuracy.

Author Contributions: Conceptualization, V.-N.T.; funding acquisition, K.-R.K.; investigation, V.-N.T.; methodology, V.-N.T.; project administration, S.-H.L., H.-S.L., and K.-R.K.; software, V.-N.T., S.-H.L., H.-S.L., and K.-R.K.; supervision, S.-H.L., H.-S.L., and K.-R.K.; validation, S.-H.L., H.-S.L., and K.-R.K.; writing—original draft, V.-N.T.; writing—review and editing, V.-N.T. and H.-S.L., and K.-R.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (2020R1I1A306659411, 2020R1F1A1069124), and the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2021-2020-0-01797) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation).

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

References

1. Ian, G.; Jean, P.A.; Mehdi, M.; Bing, X.; David, W.-F.; Sherjil, O.; Aaron, C.; Yoshua, B. Generative adversarial networks. *Commun. ACM* **2020**, *63*, 139–144.
2. Ferreira, S.; Antunes, M.; Correia, M.E. Exposing Manipulated Photos and Videos in Digital Forensics Analysis. *J. Imaging* **2021**, *7*, 102.
3. Wodajo, D.; Atnafu, S. Deepfake Video Detection Using Convolutional Vision Transformer. *arXiv* **2021**, arXiv:2102.11126.
4. Heo, Y.J.; Choi, Y.J.; Lee, Y.W.; Kim, B.G. Deepfake Detection Scheme Based on Vision Transformer and Distillation. *arXiv* **2021**, arXiv:2104.01353.
5. Zhou, P.; Han, X.; Morariu, V.I.; Davis, L.S. (2017, July). Two-stream neural networks for tampered face detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017; pp. 1831–1839.
6. Afchar, D.; Nozick, V.; Yamagishi, J.; Echizen, I. Mesonet: a compact facial video forgery detection network. In Proceedings of the 2018 IEEE International Workshop on Information Forensics and Security (WIFS), Hong-Kong, China, 11–13 December 2018; pp. 1–7.
7. Karras, T.; Laine, S.; Aila, T. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA**,** 18–23 June 2019; pp. 4401–4410.
8. 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 Pattern Recognition, Seattle, WA, USA, 2020; pp. 8110–8119
9. Non-Existent Lifelike Face of Peoples Using StyleGAN2. Available online: https://www.thispersondoesnotexist.com/ (accessed on 11 August 2021).
10. StyleGAN2—Official TensorFlow Implementation. Available online: https://github.com/NVlabs/stylegan2 (accessed on 30 June 2021).
11. Kingma, D.P.; Welling, M. Auto-encoding variational bayes. *arXiv* 2013, arXiv:1312.6114.
12. Natsume, R.; Yatagawa, T.; Morishima, S. Rsgan: Face swapping and editing using face and hair representation in latent spaces. *arXiv* **2018**, arXiv:1804.03447
13. Nirkin, Y.; Keller, Y.; Hassner, T. Fsgan: Subject agnostic face swapping and reenactment. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 7184–7193.
14. Karras, T.; Aila, T.; Laine, S.; Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation. *arXiv* **2017**, arXiv:1710.10196
15. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv* **2015**, arXiv:1511.06434
16. Natsume, R.; Yatagawa, T.; Morishima, S. Fsnet: An identity-aware generative model for image-based face swapping. In *Asian Conference on Computer Vision*; Springer: Cham, Switzerland, 2018; pp. 117–132.
17. Li, L.; Bao, J.; Yang, H.; Chen, D.; Wen, F. Advancing high fidelity identity swapping for forgery detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 5074–5083.
18. Choi, Y.; Uh, Y.; Yoo, J.; Ha, J.W. Stargan v2: Diverse image synthesis for multiple domains. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 8188–8197
19. Dolhansky, B.; Bitton, J.; Pflaum, B.; Lu, J.; Howes, R.; Wang, M.; Canton Ferrer, C. The deepfake detection challenge dataset. *arXiv* **2020**, arXiv: 2006.07397.
20. Guo, J.; Qian, Z.; Zhou, Z.; Liu, Y. Mulgan: Facial attribute editing by exemplar. *arXiv* **2019**, arXiv:1912.12396.
21. Lee, C.H.; Liu, Z.; Wu, L.; Luo, P. Maskgan: Towards diverse and interactive facial image manipulation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 5549–5558.
22. Usman, B.; Dufour, N.; Saenko, K.; Bregler, C. Cross-Domain Image Manipulation by Demonstration. *arXiv* **2019**, arXiv:1901.10024.

23. Afifi, M.; Brubaker, M.A.; Brown, M.S. Histogan: Controlling colors of gan-generated and real images via color histograms. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Montreal, QC, Canada, 11–17 October 2021; pp. 7941–7950.
24. Dang, H.; Liu, F.; Stehouwer, J.; Liu, X.; Jain, A.K. On the detection of digital face manipulation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 5781–5790.
25. Liu, H.; Li, X.; Zhou, W.; Chen, Y.; He, Y.; Xue, H.; Yu, N. Spatial-phase shallow learning: rethinking face forgery detection in frequency domain. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Montreal, QC, Canada, 11–17 October 2021; pp. 772–781.
26. Zhao, H.; Zhou, W.; Chen, D.; Wei, T.; Zhang, W.; Yu, N. Multi-attentional deepfake detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Montreal, QC, Canada, 11–17 October 2021; pp. 2185–2194
27. Mazaheri, G.; Roy-Chowdhury, A.K. Detection and Localization of Facial Expression Manipulations. *arXiv* **2021**, arXiv:2103.08134.
28. Chen, H.S.; Rouhsedaghat, M.; Ghani, H.; Hu, S.; You, S.; Kuo CC, J. DefakeHop: A Light-Weight High-Performance Deepfake Detector. In 2021 IEEE International Conference on Multimedia and Expo (ICME), Shenzhen, China, 5–9 July 2021; pp. 1–6.
29. Das, S.; Datta, A.; Islam, M.; Amin, M. Improving DeepFake Detection Using Dynamic Face Augmentation. *arXiv* **2021**, arXiv:2102.09603.
30. Kim, D.K.; Kim, D.; Kim, K. Facial Manipulation Detection Based on the Color Distribution Analysis in Edge Region. *arXiv* **2021**, arXiv:2102.01381.
31. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
32. Yang, X.; Li, Y.; Lyu, S. Exposing deep fakes using inconsistent head poses. In *ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: New York, NY, USA, 2019; pp. 8261–8265.
33. Matern, F.; Riess, C.; Stamminger, M. Exploiting visual artifacts to expose deepfakes and face manipulations. In 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW), Waikoloa Village, HI, USA, 7–11 June 2019; pp. 83–92.
34. Kwon, H.; Kwon, O.; Yoon, H.; Park, K.W. Face Friend-Safe Adversarial Example on Face Recognition System. In Proceedings of the 2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN), Split, Croatia, 2–5 July 2019; pp. 547–551.
35. Kwon, H.; Yoon, H.; Park, K.W. Multi-targeted backdoor: Indentifying backdoor attack for multiple deep neural networks. *IEICE Trans. Inf. Syst.* **2020**, *103*, 883–887.
36. Wu, J.; Geyer, C.; Rehg, J.M. Real-time human detection using contour cues. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai China, 9–13 May 2011; pp. 860–867.
37. Goyal, K.; Agarwal, K.; Kumar, R. Face detection and tracking: Using OpenCV. In Proceedings of the 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 20–22 April 2017; Volume 1, pp. 474–478.
38. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503.
39. Selim Seferbekov. Available online: https://github.com/selimsef/dfdc_deepfake_challenge (accessed on 8 May 2021).
40. Liu, W.; Liao, S.; Ren, W.; Hu, W.; Yu, Y. High-level semantic feature detection: A new perspective for pedestrian detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA**,** 18–23 June 2019; pp. 5187-5196.
41. Wei, H.; Laszewski, M.; Kehtarnavaz, N. Deep learning-based person detection and classification for far field video surveillance. In Proceedings of the 2018 IEEE 13th Dallas Circuits and Systems Conference (DCAS), Dallas, TX, USA, 12 November 2018; pp. 1–4.
42. Chen, P.; Liu, S.; Zhao, H.; Jia, J. Gridmask data augmentation. *arXiv* **2020**, arXiv:2001.04086.
43. DeVries, T.; Taylor, G.W. Improved regularization of convolutional neural networks with cutout. *arXiv* **2017**, arXiv:1708.04552.
44. Singh, K.K.; Yu, H.; Sarmasi, A.; Pradeep, G.; Lee, Y.J. Hide-and-seek: A data augmentation technique for weakly-supervised localization and beyond. *arXiv* **2018**, arXiv:1811.02545.
45. Miao, C.; Chu, Q.; Li, W.; Gong, T.; Zhuang, W.; Yu, N. Towards Generalizable and Robust Face Manipulation Detection via Bag-of-local-feature. *arXiv* **2021**, arXiv:2103.07915.
46. Baltrusaitis, T.; Zadeh, A.; Lim, Y.C.; Morency, L.P. Openface 2.0: Facial behavior analysis toolkit. In Proceedings of the 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), Xi'an, China, 15–19 May 2018; pp. 59–66.
47. Rossler, A.; Cozzolino, D.; Verdoliva, L.; Riess, C.; Thies, J.; Nießner, M. Faceforensics++: Learning to detect manipulated facial images. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 1–11.
48. Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; Jégou, H. Training data-efficient image transformers & distillation through attention. *Int. Conf. Mach. Learn.* **2021**, *139*, 10347–10357.

49. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–2 July 2016; pp. 2818–2826.
50. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.;Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
51. Lin, M.; Chen, Q.; Yan, S. Network in network. *arXiv* **2013**, arXiv:1312.4400.
52. Li, Y.; Yang, X.; Sun, P.; Qi, H.; Lyu, S.C.D. A large-scale challenging dataset for DeepFake forensics. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13-19 June 2020; pp. 14–19.
53. Nguyen, H.H.; Yamagishi, J.; Echizen, I. Use of a capsule network to detect fake images and videos. *arXiv* **2019**, arXiv:1910.12467.
54. Li, Y.; Lyu, S. Exposing deepfake videos by detecting face warping artifacts. *arXiv* **2018**, arXiv:1811.00656.