What AI Has Broken Can Be Fixed By AI As Well

Madhu Vemana
mv2283
mv2283@nyu.edu

Mujahid Ali Quidwai
maq4265
maq4265@nyu.edu

Sidharth Shyamsukha
ss14885
ss14885@nyu.edu

Abstract

Deepfake technology is impacting the people and society with devastating consequences likely to be noticed if not controlled timely. Actually, it is kind of serious exploitation of AI and machine learning technology used to perform a face swap to create the illusion that someone either said something that they didn't say or are someone they're not and such changes are not noticeable to human eye, compelling people to believe easily. Detecting such deepfake technology is difficult for machines or AI-backed systems, especially if they are not trained with the right quality and amount of data sets.The manual deepfake detection can be a little time consuming and need a lot of man hours,Our novel solution provides a solution to detect deep fake with human level accuracy and with minimal resolution utilization.Thus, this paper follows a deep learning approach and presents a network, with a low number of layers so they can deployed on edge devices while focusing on the mesoscopic properties of images. We evaluate these fast networks both on an existing dataset and a dataset we have created by randomly combining images from various benchmark deepfake detection dataset.Further we have improved on binary classification of images (Fake vs Real) by changing the classifier to SVM(support vector machines) a lite tool that can be easily applied to the relatively small-scale tasks.

1 Introduction

Deepfake is one of the powerful tools that can be used to generate hyper-realistic videos with swapped faces. Suppose the goal is to swap faces of person A in a video with faces of person B, or vice versa. The first step is to collect a dataset with A's face and another dataset with B's. Then we can build two autoencoders, which contain an encoder and a decoder, one for each dataset. One small tweak is that the two autoencoders share the same encoder. By doing so, the encoding pattern is forced to be the same and can be understood by both decoders. Now with the datasets and neural networks, we can train each autoencoder separately.Figure 1 depicts how we implemented a GAN on our own faces to generate a faceswap. Our proposed architecture is a extended version

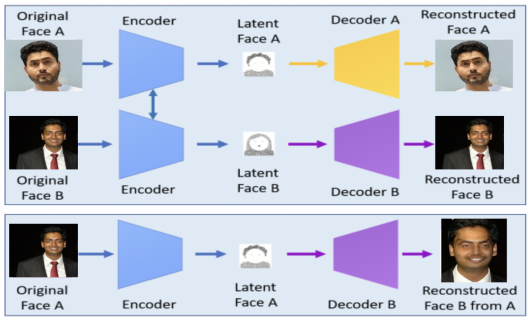


Figure 1: Shows a Deepfake Creation Process

of Mesoinception4 architecture.The tests we ran on this architecture demonstrate a very successful detection rate with more than 96 % for Deepfake[11] and 93 % for Faceforensic++[8]and 96 % for CelebDF[7] and an aggregate performance of 66.7 % on the combined dataset.Our architecture is more robust and simple to be easy deployable on edge devices within the industry settings.

2 Related Works

The first paper that has worked to detect deep fake was by Afchar[1] called Mesonet.The Mesonet is one of the first models dedicated to detection of the Deepfake video falsification technique. The prediction are made for each face image extracted from videos using Viola-Jones detector[12] on frame-by-frame basis,the same as video creation process.The mesonet takes mesopic approach to manipulation detection.As Discussed by Afchar[1] in his paper on Mesonet, a microscopic analysis of image could not be implemented, due to degradation which happend with frames of videos while they undergo compression for transfer over they internet. There are two popular varient of Mesonet:

- **Mesonet** : In Mesonet[1] there are 4 convolutional layers.The convolution layer uses kernels of size 3X3,while the following convolutional layers use kernel of size 5X5.Each convolutional layer is followed by a batch normalization layer and a pooling layer.The convolution layer is connected to a fully connected layer and a pooling layer.The last convolution layer is connected to a fully-connected layer and a pooling layerwith 16 neurons and then the output layer.
- **MesoInception-4**[1] : A fancier variant of the MesoNet is the MesoInception-4. Here, the convolutional layers are switched to inception modules inspired by the inception module in GoogleNet. Inside each inception module, there are four parallel branches of (dilated) convolutional layers[4], each with a different reception field. However, instead of increasing the filter size, the inception module here increases the reception field by using the dilated convolutional layers with different factors. The feature maps from each branch are then concatenated to form the output of the module.[2][5][9]

The other models which are currently the industrial benchmark for detection of deep fake as well as generation of deepfake Datasets are:

- **Face2Face++ :** Reenactment methods, are designed to transfer image facial expression from a source to a target person. Face2Face, introduced by Rossler[8]., is its most advanced form. It performs a photorealistic and markerless facial reenactment in real-time from a simple RGB-camera. The program first requires few minutes of prerecorded videos of the target person for a training sequence to reconstruct its facial model. Then, at runtime, the program tracks both the expressions of the source and target actors video. The final image synthesis is rendered by overlaying the target face with a morphed facial blendshape to fit the source facial expression.
- **Deepfake** Deepfake is a technique which aims to replace the face of a targeted person by the face of someone else in a video. It first appeared in autumn 2017 as a script used to generate face-swapped adult contents.It was using autoencoder-decoder pairing structure. In that method, the autoencoder extracts latent features of face images and the decoder is used to reconstruct the face images. To swap faces between source images and target images, there is a need of two encoder-decoder pairs where each pair is used to train on an image set, and the encoder's parameters are shared between two network pairs[5].

3 Proposed Method

We propose to detect images that have been tampered with using the mesoponic properties of the image. We are implementing the Mesoinception-4 net architecture that uses the inception module introduced by Szegedy[10]. The network is very simple and has a significantly smaller number of parameters as compared to other papers in the area. We have applied data augmentation techniques to make the network more robust. We have used a more advanced classification method i.e. SVM(RBF) for better classification at the end of the fully connected layers. This is our novel suggestion.

3.1 Architecture

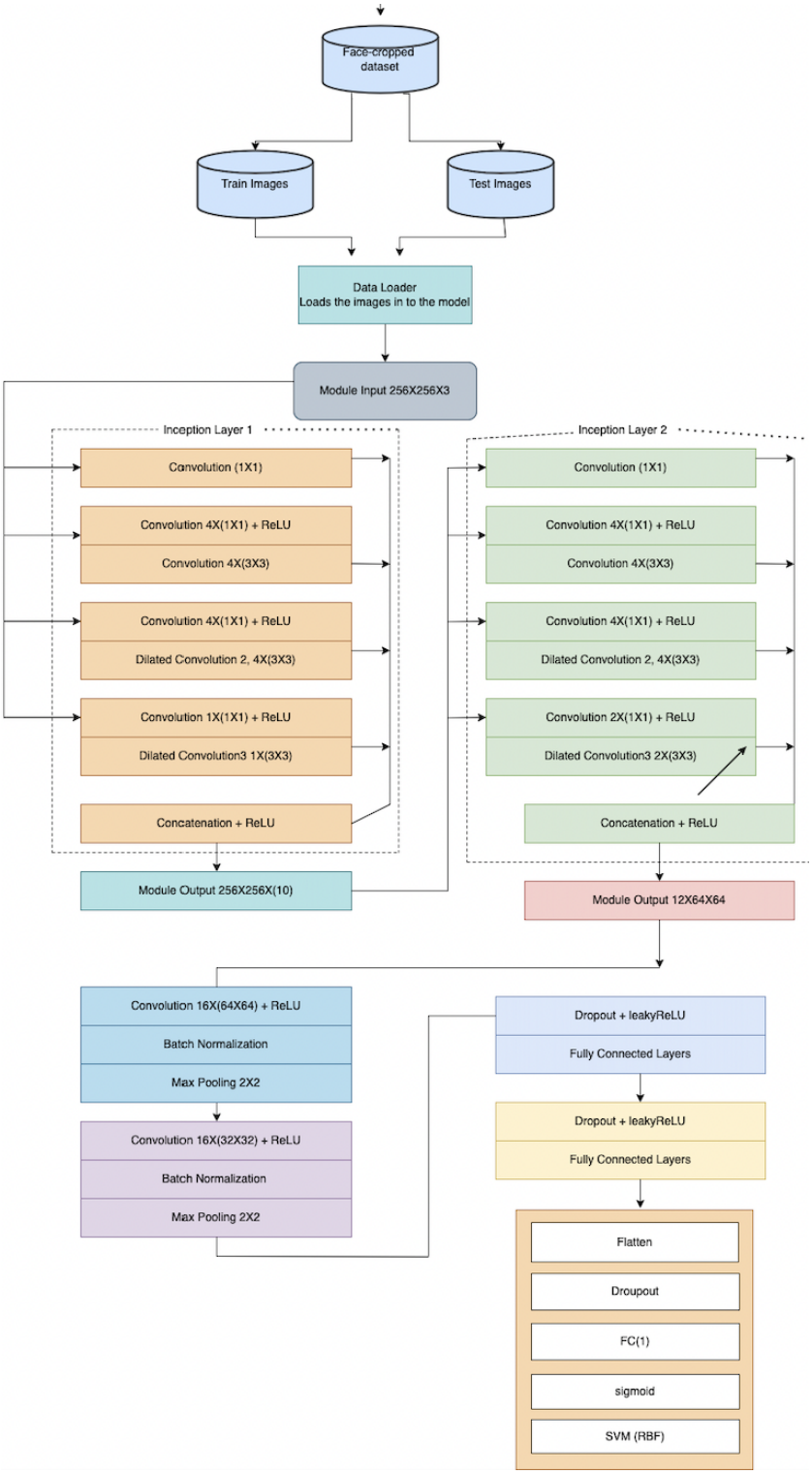


Figure 2: Architecture of the Inception Modules used in our model of MesoInception-4+SVM(RBF)(Classifier).

4 Experiments

In this section we expose the results of the implementation of the our proposed architectures to detect deep fake. In order to extend our work and compare with existing industrial benchmark for detection of deep fake we have implemented ROC and confusion matrix for mesoinception-4[1] and our architecture and to cross verify them.

4.1 Dataset

4.1.1 Datatset Used

The availability of large-scale datasets of DeepFake videos is an enabling factor in the development of DeepFake detection method. The experiments were conducted on 3 benchmark datasets: Celeb-DF (v2) (CD2)[6], DeepFake Detection Challenge Preview (DFDC)[3] , FaceForensics++ (FF++)[8]. The distribution of images is shown in following table:

Set	Size of the forged image class	Size of real image class	Total
Training	5111	7250	12361
Test	2889	4259	7148
	8000	11509	19509

Table 1: Data Distribution

In order to validate our model based on this dataset we did a train: validation:test split of 70:20:10.The split of dataset is as follows (Table 2 and Table 3):

Set	Size of the forged image class	Size of real image class	Total
Training	7175	10337	17512
Test	773	1172	1945
	7948	11509	19457

Table 2: Train Test Split

Set	Size of the forged image class	Size of real image class	Total
Training	5740	8270	14020
Validation	1435	2067	3502
	7175	10337	17512

Table 3: Train Validation Set Split

4.1.2 Data Pre-Processing

We take in 3 benchmark dataset and merge it into a single combined dataset to fetch into the pre processing pipeline of our model in order to train the model on a more robust dataset.

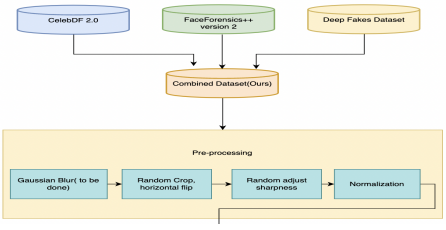


Figure 3: Data Preprocessing Pipeline

4.1.3 Data Augmentation

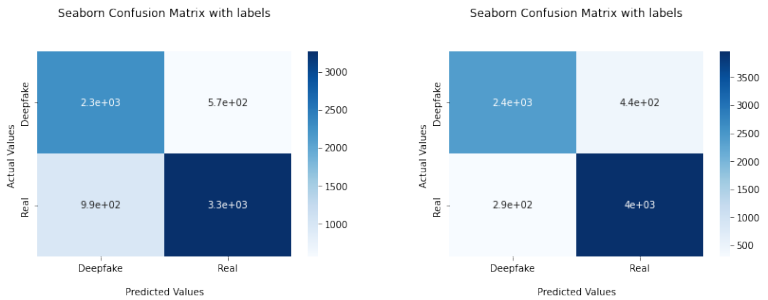
Data augmentation is the process of preprocessing the data before feeding it to the model. It increases the diversity of the data which allows the model to generalize well and results in higher accuracy. We have implemented the following kinds of transforms.

- o Adding Gaussian Blur with different kernel sizes (3x3,9x9, 15x15): the noise added to the images could destroy the pixels correlation created by Deepfake architectures and remove the CT.
- o Rotating images by 45, 90, 180 degrees: rotations could lead to interpolation transformation with modification on CTs similar to the Gaussian blur attack.
- o Scaling images $(+50\%, -50\%)$: due to the interpolation operations carried out, information will be added or removed. CT extracted from images with high details (such as those of STYLEGAN and STYLEGAN2) would be more robust to this type of operation. An example can be observed in Figure 2.

Figure 4: Data Augmentation Example

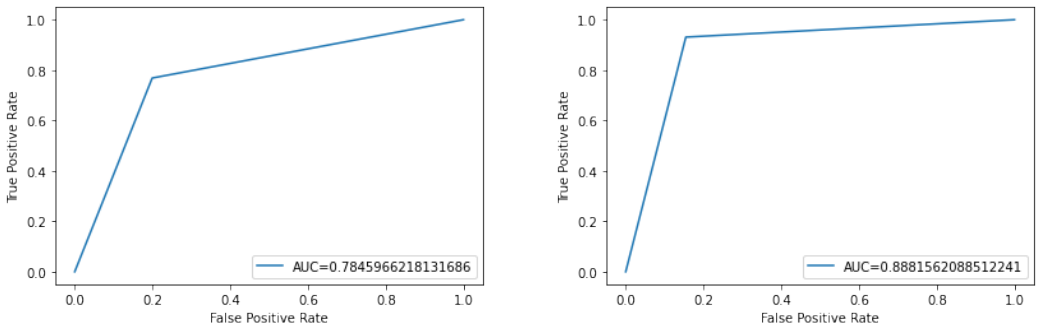
4.2 Classification Setup

We denote $\mathcal{X}$ the input set and $\mathcal{Y}$ the output set, the random variable pair (X, Y) taking values in $\mathcal{X} \times \mathcal{Y}$, and f the prediction function of the chosen classifier that takes. values in $\mathcal{X}$ to the action set $\mathcal{A}$. The chosen classification task is to minimize the error $\mathcal{E}(f) = E[l(f(X), Y)]$, with $l(a, y) = \frac{1}{2}(a - y)^2$..This is achieved via our novel SVM(RBF) architecture on the existing Mesoinception-4[1] output. Both Mesoinception-4 and Mesoinception-4+SVM have been implemented using Pytorch and Tensorflow. Weights optimization of the network is achieved with successive batches of 32 images of size $256 \times 256 \times 3$ using ADAM [13] with default parameters ($\beta_1 = 0.9$ and $\beta_2 = 0.999$). The initial learning rate of 10^{-3} is divided by 10 every 1000 iterations down to 10^{-6}. As both network have a relatively small amount of parameters, few hours of optimization on a standard consumer grade computer were enough to obtain good scores.We ran both the architecture on the three data sets on an epoch of size 10 and obtained following accuracy and AUC values as shown in Fiqure 5 and Fiqure 6.



(a) Mesonet+SVM our architecture (b) Mesoinceotion-4 Architecture

Figure 5: Confusion Matrix



(a) Mesoinception-4+SVM our Architecture

(b) Mesoinception-4 architecture

Figure 6: ROC Curve

4.3 Image classification results

This section summarizes results from the two pre-trained models provided in trained models. Here, ”best” is in terms of accuracy Classification scores of both trained network are shown in Table 3 for the Deepfake dataset. Both networks have reached fairly similar score around 90% considering each frame independently. We do not expect a higher score since the dataset contains some facial images extracted with a very low resolution[1].

Training was meant to be carried out for 30 epochs with a batch size of 32. In fact, the model was trained for only 10 epochs since the results were already satisfactory.

For this particular model, when using a learning rate schedule, the number of steps after which one step of decay should be applied is calculated dynamically based on a decay limit (the lowest learning rate), decay steps and the number of epochs. The reason behind this is that using a fixed number made the decay either too slow or too fast. This makes it more gradual. This feature wasn't implemented during training of the second model. On the test set, the model reported an accuracy of 96.25%.

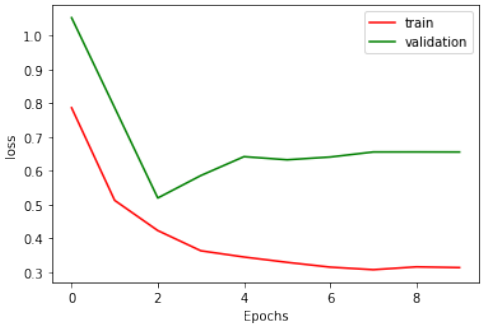


Figure 7: Train and Validation Accuracy for Mesoinception-4 Architecture

	precision	recall	f1-score	support
0	0.89	0.85	0.87	2845
1	0.9	0.93	0.92	4259
Accuracy			0.9	7104
Macro avg	0.9	0.89	0.89	7104
weighted avg	0.9	0.9	0.9	7104

Table 4: ROC Report for Normal Mesoinception-4 Architecture

4.3.1 Next Best Model

Training was meant to be carried out for 18 epochs with a batch size of 64. A learning rate schedule with an initial learning rate of 0.001, decay rate of 0.10, decayed every 5 epochs. The model was trained for 10 epochs.

While training loss and accuracy were not recorded for later reference the validation accuracy was approximately 89%. On the test set, the model reported an accuracy of 90.79

The ROC report:

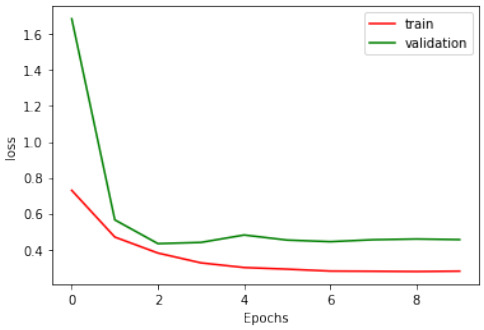


Figure 8: Train and Validation Accuracy for Mesonet+SVM (Proposed Architecture)

	precision	recall	f1-score	support
0				
Accuracy	0.7	0.8	0.75	2845
Macro avg	0.85	0.77	0.81	4259
weighted avg			0.9	7104

Table 5: ROC Report for Mesonception+SVM(Proposed Architecture)

4.4 Intuition behind the work

We have tried to understand how those networks solve the classification problem. This can be done by interpreting weights of the different convolutional kernel and neurons as image descriptors. For instance, a sequence of a positive weight, a negative one, then a positive one again, can be interpreted as a discrete second order derivation. However, this is only relevant for the first layer and does not tell much in the case of faces.

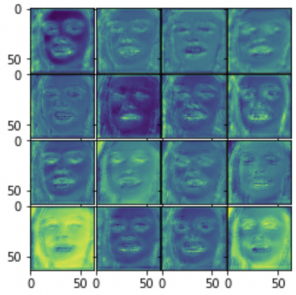


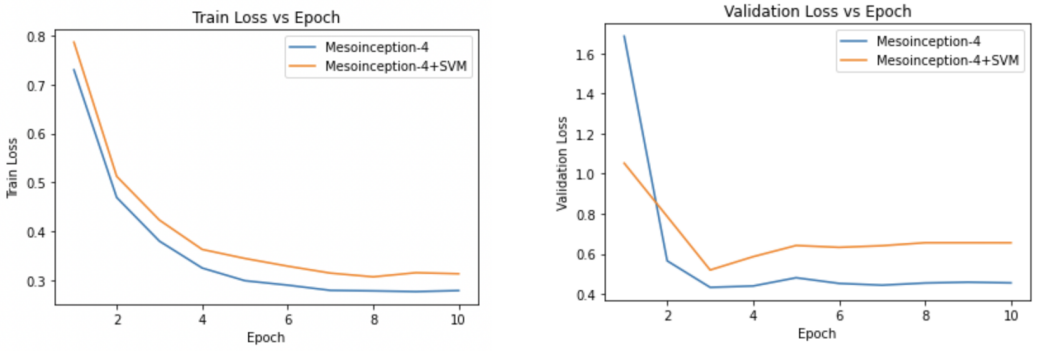
Figure 9: Maximum activation of some neurons of the hidden layer of Mesoinception-4. The optimization was done with λ = 10 and p = 6.

Another way is to generate an input image that maximizes the activation of a specific filter [6] to see what kind of signal it is reacting to. Concisely, if we note f_{ij} the output of filter j of layer i

and x a random image, and we add a regularization on the input to reduce noise, that idea boils down to the maximization of $E(x) = f_{ij}(x) - \lambda\|x\|_p$. Figure 7 shows such maximum activation for several neurons of the last hidden layer of Meso4[1]. We can separate those neurons according to the sign of the weight applied to their output for the final classification decision, thus accounting for whether their activation pushes toward a negative score, which corresponds to the forged class, or a positive one for the real class. Strikingly, positive-weighted neurons activation display images with highly detailed eyes, nose and mouth areas while negative-weighted ones display strong details on the background part, leaving a smooth face area. That's understandable as Deepfake-generated faces tend to be blurry, or at least to lack details, compared to the rest of the image that was left untouched.

5 Conclusion

Mesonet and mesoinception model try to detect deepfakes by catching the discrepencies generated in natural human features like the pixels in eye, image noise, etc. We created a new data set to test our novel approach using the combination of deepfake[2] , face2face[8], CelebDF[6]. The main focus in constructing and training the model was to make it modular and portable. This methodology can be scaled up to any dataset size having the images of predefined size. We ran our model and mesoinception model for 10 epochs on both real and fake images and found validation accuracy of around 66%. This can be further increased if the model is run for around 200+ epochs which we were unable to do due to time and resources constraints[1].
Most deep fakes are usually spread through social media websites which compresses those images and important features are lost, which can be detrimental for our model.But, since large number of deep fakes are released on these social media platforms, they can be collected and added to the training of the model to further increase the detection capability of the model. For the exact computational constraints we observed better performance with SVM as the the classifier as compared to mesoinception4[1].Below we represent both the training loss and the validation loss for both the model along with the accuracy and AUC values.



(a) train accuracy vs number of epochs

(b) validation accuracy vs number of epochs

Figure 10: SVM+Mesonet (Proposed Architecture) Vs current industry standard Mesoinception

Method	ACC	AUC
Mesonet	64.16	82.20
Mesoinception4	67.51	88.82
Mesoinception4+SVM	67.57	78.46

Table 6: Generalization on Deepfake Dataset

The tests demonstrate a very successful detection rate with more than 67% for Deepfake[11] and 62 % for Faceforensic++[8]and 70 % for CelebDF[7] and an aggregate performance of 66.33 % on the combined dataset. In this paper, we introduce what Deepfake is and presented two models for Deepfake detection. Each model showed a promising result. After training with a method, the

Method	ACC	AUC
Mesonet	63.16	80.02
MesoInception4	65.51	85.03
MesoInception4+SVM	62.57	75.06

Table 7: Generalization on Face2Face dataset

Mesoinception-4+SVM can reliably detect fake images generated by the same method with a really simple model that may run on edge devices. However, we can't say we have nailed the Deepfake detection. The models we present today can identify deepfake images easily due to the obvious imperfection(at least to deep neural nets) in the generated images. As the generative networks improve, this game of cat and mouse will continue.

6 Github Repo Link

6.1 `https://github.com/Ali-Maq/Deep_Learning`

References

[1] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Mesonet: a compact facial video forgery detection network. In *2018 IEEE international workshop on information forensics and security (WIFS)*, pages 1–7. IEEE, 2018.

[2] Oscar de Lima, Sean Franklin, Shreshtha Basu, Blake Karwoski, and Annet George. Deepfake detection using spatiotemporal convolutional networks. *arXiv preprint arXiv:2006.14749*, 2020.

[3] Brian Dolhansky, Russ Howes, Ben Pflaum, Nicole Baram, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) preview dataset. *arXiv preprint arXiv:1910.08854*, 2019.

[4] Luca Guarnera, Oliver Giudice, and Sebastiano Battiato. Deepfake detection by analyzing convolutional traces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 666–667, 2020.

[5] David Güera and Edward J Delp. Deepfake video detection using recurrent neural networks. In *2018 15th IEEE international conference on advanced video and signal based surveillance (AVSS)*, pages 1–6. IEEE, 2018.

[6] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df (v2): a new dataset for deepfake forensics. *arXiv preprint arXiv:1909.12962*, 2019.

[7] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3207–3216, 2020.

[8] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1–11, 2019.

[9] Ekraam Sabir, Jiaxin Cheng, Ayush Jaiswal, Wael AbdAlmageed, Iacopo Masi, and Prem Natarajan. Recurrent convolutional strategies for face manipulation detection in videos. *Interfaces (GUI)*, 3(1):80–87, 2019.

[10] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[11] Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia. Deepfakes and beyond: A survey of face manipulation and fake detection. *Information Fusion*, 64:131–148, 2020.

[12] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. Ieee, 2001.