National College of Ireland

Comparative study of state of the art deepfake detection models

Research Project

Msc Data analytics

Ajay Kommalapati

Student ID: 20168829

School of Computing
National College of Ireland

Supervisor: Arghir-Nicolae Moldovan

National College of Ireland
Project Submission
Sheet School of Computing

Student Name:	Ajay Kommalapati
Student ID:	20168829
Programme:	MSc in Data analytics
Year:	2021
Module:	Research Project
Supervisor:	Arghir-Nicolae Moldovan
Submission Due Date:	31/1/2022
Project Title:	Comparative study of state of the art deepfake detection models
Word Count:	9187
Page Count:	21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Ajay Kommalapati
Date:	31st January 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	Q
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	Q
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	Q

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Comparative study of state of the art deepfake detection models

Ajay Kommalapti

20168829

Abstract

Data analytics through object recognition and human-level control have all been effectively solved using deep learning. It's not all good news for privacy, democracy, and national security, though, because of developments in deep learning. Deepfake is a recent example of a deep learning-powered application. Fake photos and movies created by deepfake algorithms might be difficult for people to tell apart from real ones. Digital visual media must consequently be able to automatically identify and analyse the integrity of its content. This paper studies the comparison between state of art with two online scanners such as WeVerify and Deepware. Using an EfficientNet backbone trained on ImageNet, these online scanners pretrain with the various datasets and employ an ensemble of five models. For the celeb-df dataset, DefakeHop achieved an AUC of 94.95 percent, while the online scanner Weverify and deepware achieved an accuracy rate of 97 and 75 respectively.

Keyword: AUC, Deepfake, EfficientNet, ImageNet, DefakeHop, Deepware.

1 Introduction

Deepfakes are media such as movies, as well as formats such as photos or audios, that are not authentic. When a computer is given data, it generates a new face depending on the information provided by the user. In September 2019, the AI company Deeptrace discovered 15,000 spoofing videos on the internet, almost double the number discovered in the previous nine months. Donald Trump's rant against Belgium, which he subsequently apologized for, serves as an example of a video that was manipulated. Jeff Bezos and Elon Musk were also featured in a similar film. The rise of deepfakes in recent years has created serious concerns about the veracity of digital content provided by media and other internet streaming services. A fantastic method to explore the creative possibilities of deep learning systems is via the usage of generative designs. While these approaches have been exploited for malicious purposes to modify governments, celebrities, and other public figures' images, deepfakes have developed as a result.

Many high-profile politicians and celebrities have died because of the spread of fake news. There are two new ways for manipulating forensic film for criminal purposes: faceswap and faceswap-GAN.

What kind of technology is being used?

It is referred to as Generative Adversarial Networks, or GAN for short. It makes use of two artificial intelligence algorithms, one of which creates bogus material and the other which assesses the efforts

with the goal of making the system better. Every day, GAN creates fresh human portraits, which are also accessible on the website 'www.thispersondoesntexist.com.'However, it is practically impossible to tell the difference between a genuine picture and a phony one.
Human faces are used in a variety of applications to produce complicated and intriguing images by changing the age, gender, etc. In exchange, clients hand up their personal information, which might be exploited, to these companies.

To enable uploading and downloading, the quality of the changed videos is reduced when they are shared on social media networks. It is possible to see some fuzziness surrounding the face deformation in high-quality film. Consumers can't determine whether the films are genuine or not in low-quality movies, and they are widely disseminated. Everything from politics to entertainment may be affected by manipulating people's perceptions. An example of this is the damage done to politicians' reputations by phony recordings of them performing public events or supporting public services. similarly, fraudulent pornographic videos starring actors have been extensively shared.
Methods that use convolutional neural networks (CNNs) and methods that integrate CNNs as well as recurrent neural networks (RNNs) make up the majority of current state-of-the-art Deepfake detection techniques. These techniques can be broadly divided into two categories:
(1) those that use DL techniques
(2) those that do not.

When compared to the former, the later takes into consideration both geographical and temporal aspects. The drawbacks of DL-based solutions are many. As a first step, these models are often quite big, encompassing many thousands, or even millions, of parameters. It's also time-consuming to train them. Handcrafted characteristics may also be recovered and fed into classifiers in non-DL-based Deepfake detection algorithms Xin Yang et al. (2019) Non-DL-based approaches tend to perform less well than DL-based ones. (defakehop)

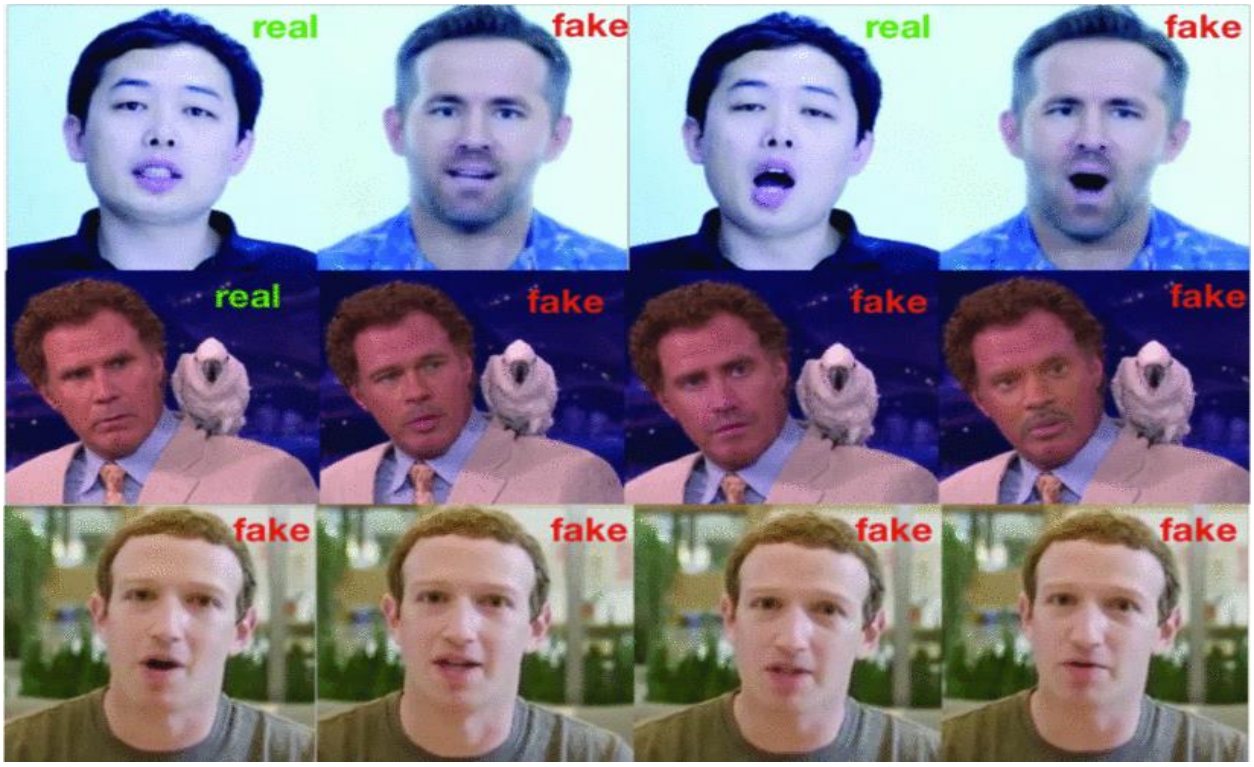

Fig1 : (Top)Head puppetry, (middle)face shifting and (bottom) lip syncing are all examples of DeepFake's work[59].

figure 1 illustrates some of the many types of DeepFake films. DFaker [3], FakeApp [2],faceswap-GAN [4], DeepFaceLab and faceswap [5] are some of the first instances of DeepFakes to commercialize and mainstream face swapping. DeepFake movies may also be generated on demand using an internet service (https://deepfakesweb.com), but there are several online forums discussing the topic.
Using DeepFake techniques based on deep learning, Reddit user "DeepFakes" created pornographic films with celebrities whose faces were changed and posted. In addition to damaging the individuals involved's right to privacy and reputation, this incident has also had a negative impact on internet security. These include hoaxes and financial scams, as well as phony pornographics and fabricated news. If politicians' faces are replaced with deceptive talking films, it might lead to international crises. " As a result, it is critical to create techniques for detecting DeepFake in videos.

Research Questions:
To see whether there was a statistically significant difference between state-of-the-art and web - based deepfake detection scanners.
a. Evaluate the defakehop probability of finding the deepfakes in testing videos.
b. Evaluating the two online scanners probability of finding deepfakes in video.

In this comparative study I have observed few things like limitations over sources i.e., system configurations. It took long time to pre-process the videos such as while extracting faces from videos and cropping the landmarks. And one thing is observed that while running the *Defakehop* state of art tested with the few selected videos to attain the high rate of accuracy. The selection of testing videos is 100 out of total 1203 which 8.3%. Along with defakehop performed with some random splits which gives less accuracy.
The sections of this document are listed as follow. Section 2 provides literature review. Section 3 is having methodology. Section 4 provides implementation and evaluation. In Section 5, the conclusion and feature work. Section 6. acknowledgment is made clear.

2 Related Work

Image forgery has long been a significant problem in the field of forensic science, with matching techniques for visual artifact detection using pixel and frequency analysis. Throughout adding to early generative models, these technologies were able to produce some inauthentic material. Humans, on the other hand, were able to distinguish between findings that were clearly phony and those that were clearly genuine. N. B. A. Warif et al (2016).
models that learn to distinguish between samples first from model distribution and those from other datasets are called discriminative models Generic and discriminative models are akin to criminals and law enforcement officers, respectively, who are attempting to make counterfeit currencies and utilize them without being caught. In this game, competition forces both sides to develop their procedures till the knockoffs are distinct from the originals. Model and optimization training methodologies that are exclusive to this framework may not be found anywhere else. There are two layers of perceptrons in the generating model and one layer of perceptrons in the classification model. The technical name for this is adversarial networks. For choosing from the prediction model, we only have backpropagation and dropout. Neither Markov chains nor approximation inference are necessary. I. Goodfellow et al (2014).

To transform one image into another without considering the context of the original Another technique

is the connection of 2 variables (X and Y) that may be utilized in a variety of ways. It has been proposed to use a Bayesian framework, which considers both the prior and posterior from the source picture as well as probabilities derived from various style photographs but same source image itself. Modern models like CoupledGANs like scene networks Y. Aytar et al (2016). Use weight sharing to accomplish this aim while acquiring a common symbol across several domains. Variational machine like generative adversarial networks is increasingly being utilized in combination with our technique. A separate line of parallel work A. Shrivastava et al (2016). Forces the output signals to have certain "content" characteristics despite their differences. These networks may also include additional concepts like pre - classified space K. Bousmalis et al (2016). Picture pixel and feature spaces to push the output to be near to the input, which is why they are called adversarial networks. However, they do not believe that two data points are embedded in almost the same lesser embeddings in our formulation of similarity functions among input and output. Thus, our technique is applicable to a broad variety of vision and visual challenges.

GANs can create pictures that have been so identical to the true thing that it is practically impossible to distinguish between them. There are a variety of computer graphics programs available that are designed to improve the appearance of photographs. By an opponent loss, images within target domain are unable to be discriminated from those in the translation domain. J. Wu et al (2016).

This intermediate description known as the code vector is developed by autoencoders using a coding and decryption process that estimate the identity mapping. When it comes to obtaining physiologically realistic visual qualities, they've been used. G. E. Hinton et al (2006). And J. Masci et al (2011). For unsupervised designs, there is no need for labeled data. It's a great perk to have. Encoding and decoding approaches, such as the inversion of the generative model, and creating visuals from code vectors A. Zhmoginov and M. Sandler et al (2016). Have a lot of overlap. When it comes to solving challenges in biometrics, auto - encoder have shown to be useful. In addition to recognition - based, actual face aligning, guided face recognition, and learning face representations using stacked autoencoders, they have been utilized to develop stacked progressive autoencoders. Graphics codes that can be recreated in a variety of conditions may be learned using the Convolutional Network Inverse Graphics Network. (e.g., posture and lighting). Using mini batches, where just a single parameter of a scene may be changed, this can be done. Code parameters, like shape as well as scene alterations, have been described in E. Grant et al (2016). When compared to existing methods, our proposed technique takes into consideration all important factors and does not require to categorize images according to specified variants.

To 3D face reconstruction, this paper proposes a unique model deep convolutional autoencoder that incorporates generative and CNN-based regression algorithms. In our network architecture, a CNN encoder and only a CNN decoder are connected through a reduced dimensionality code-layer. Our convolutional autoencoder has a decoder, unlike previous CNN-based decoders. 3D parametric face model is implemented in this layer. Previous fully CNN-based autoencoders could not ensure the semantic relevance of code technical. Our new network makes certain that the decoder's input code vector has the correct semantic significance. Additionally, our decoder is tiny and does not need a large set of CNN weights. A. Tewari et al (2017).
The discipline of computer vision has long considered face pictures and videos to be a vital application area. Some examples of GAN's various usage include face enhancement, facial attribute modification, and frontal view construction. It is also possible to employ GAN for face reconstruction, identity

retention and expression modification W. Shen and R. Liu et al (2018). The phrase "deep fake" was developed to characterize the situation because of developments in VAEs and GANs with face reconstruction and video synthesis. Autoencoders trained on the source and target movies are believed to be employed in the process: The video's source is Face-specific information may be encoded and decoded using that encoder weights. A video source is used to build a target face, which is then warped in actual to use the original picture's blend shapes. The evolutionary algorithm networks of Deep Video Portraits and Vid2Vid are used instead of mixing forms (GANs). This may be caused by light blockage, compression, and rapid movements Y. Choi et al (2018).

Additionally, several methods for detecting videos with face alterations have been given. Because some of these algorithms concentrate on recognizing movies containing solely DeepFake modifications, others are meant to be indifferent to the methodology used to execute face manipulation. It is possible to identify fake videos using a combination of such a Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN). In the current version of DeepFake, video frames are spliced together using a face-synthesizer. When calculating the 3D head posture, artifacts left over from the splicing process may be observed. Leverage this fact to your advantage and then use the difference in head posture computed using complete facial landmarks and just a selection of them to distinguish between genuine and fake films D. Afchar et al (2018). On the UADFV database, this approach was able to provide competitive results. Face warping anomalies may be used to identify fake videos, according to the same researchers. The authors of X. Yang et al (2019). use a deep neural network based on mesoscopic properties taken from video frames to recognize edited films created by DeepFake and Face2Face algorithms., Face2Face, DeepFake, NeuralTexture, and FaceSwap were shown to be better at identifying fake videos than shallow network models in a supervised context, according to the findings published in.

As computing power increased and deepfakes became a real concern, numerous approaches for classifying fake films emerged. There are several modern algorithms that may be used to identify manipulations based on the discrepancy among visual artifacts, head posture variation, and segmentation masks. Two-stream CNN uses a face categorization networks and a patched triplet network to identify the modified faces. LeNet architecture is used to train the classification network. Forcing pictures to be embedded closer together, Patch Triplet network aids in this task. These traits are learned in MesoNet using shallow designs that include an inception module. In contrast, recent studies have shown that deep structures outperform short systems by a substantial margin. To detect manipulation, HeadPose measures the distance between a synthetic image head posture and indeed the input images head pose L. Verdoliva et al (2020). The landmarks of altered faces are displaced first from source faces when the 3D head posture is estimated from 2D coordinates. Artifacts introduced by warping phases are spotted by Li and Lyu. CNNs are taught to identify inconsistencies between the final picture and its original state after undergoing different transformations, such as scaling, rotation, and scaling. Visual artifacts were examined. to identify edited photos.

They demonstrated that the difference between being an actual footage and an edited one can be readily detected using facial features. These types of manipulations (DeepFakes and Face2Face) were the primary focus of their evaluations. To improve classification accuracy, classification, reconstruction, and segmentation were combined in the Multi-task learning technique A. Rossler et al (2019). For classification, the final activation of the encoder-decoder method is employed. Capsule networks are aimed to avoid the requirement to train thousands of variables for deep neural networks by using a

smaller amount of parameters.Because the face has indeed been altered, a dynamic routing method is used to build an activation map. We can show that these methods don't work well with fresh and more difficult datasets P. S. H. Q. Yuezun Li et al (2019). When the video's quality is good, we may observe a variety of ways function effectively. A considerable loss in performance occurs when the footage was compressed to a medium and high level. Neural Textures' accuracy decreases from 95 percent to 50 percent, which suggests that the system has been unable to acquire any characteristics at such low resolution and just annotates the movies as false or genuine.

The lack of visual information is a major drawback when movies are purposely constructed. This is due to the absence of training images that contain images of the subject while their bare eyes. In order to get past this detection, you may introduce photographs of individuals with their closed eyes into training sessions Another way to spot tampered digital data is to keep an eye out for unusual head postures. The color-space features of GAN-generated and actual images are compared, as well as the difference is utilized to classify the images S. Iizuka et al (2017)., O. Langner et al (2010). Reverse engineering a GAN architecture's last computational layer is used in this work to determine whether the fundamental evidence of a putative Deepfake may be retrieved. Using this approach, the predictions of Deepfakes can be explained, which is very useful for forensic investigations since it can not only categorize an image as fake but also forecast the most likely technology used for production, comparable to camera model identification in image forensics analysis. Based on this method, we can identify the key periodic components (e.g., the transposed computational layer) in the created pictures. Popescu et al. A. C Popescu and H. Farid et al (2005). pioneered a similar method in a landmark article dedicated to highlighting digital frauds in CFA interpolated photos some time ago. Furthermore, compared to current methods, the suggested strategy has shown its ability to obtain excellent outcomes in a near-natural setting using photos created by five distinct processes and varying image sizes.

These approaches include pattern allows detection, eye blinking detection and generic video detection in the forensic community. Facial forgery datasets and anomaly detection are two of the most often used approaches to identify Deepfakes. A useful collection of face forgeries, FaceForensics++ A. Rossler et al (2019)., makes it possible for build deep learning methods based on forgeries. A training CNN (Convolution Neural Network) technique to detecting false video has been studied Kim DaeHee et al (2018) as well [48]. These approaches offer reliable outcomes, but they need a large amount of data and ongoing refinement. Rather of wasting time collecting data, we opted to do research Li, Y., Chang et al (2018). that may have a wider impact. Many Deepfakes faces do not blink, according to a study. To get around these detection methods, several novel examples have surfaced in which the discriminator has been tweaked to validate blinking.
For example, advancements in technology such as DeepVision Li, Y., Chang et al (2018). need increasingly complex integrity verification systems. Using the outcomes of medical, biological, and brain engineering research, and deep learning and methods founded on technical and statistics expertise, DeepVision verifies the integrity of deepfakes by detecting major changes in the eye blinks of the fakes. This technique will help us overcome the limits of integrity verification based just on pixels.

On the other hand, invite 100 paid performers and gather high-resolution source material (1920 * 1080) with a variety of postures, emotions, and illuminations. Using 3DMM blendshapes, several over-the-top expressions may be augmented. All actors have given us permission to use and manipulate their faces. Face swapping (i.e., DF-VAE) is a novel end-to-end face swapping approach (i.e., seven kinds of perturbations applied to the false movies at five intensity levels). With the addition of distortions, our

dataset better reflects real-world situations.
FaceForensics Benchmark Andreas Rossler et al (2019). Has recently been offered as a key benchmark for face modification detection. It contains six image-based face forgery detection benchmarks. Although FaceForensics Test introduces aberrations towards the videos through turning them into various compression rates, a thorough examination of other perturbation kinds and their combination is lacking. Celebr-DF P. S. H. Q. Yuezun Li et al (2019). Presents a facial forgery detection benchmark with seven techniques trained and evaluated on diverse datasets. Similarities between training and test sets may be seen in several of the benchmarks cited above. When these approaches are used in a real-world situation with a wide range of phony films, they are rendered useless because of the inherent biases they impose on the system.

In the case of false videos, we feel that most of these PPG deviations provide useful information. In addition, genuine movies have a greater level of PPG signal interoperability from multiple parts of a face than synthetic videos. This helps us ensure that environmental impacts (such as light and occlusion) are consistent. In order to combat compression artifacts, we employ a mix of G channel-based PPG (also known as G-PPG or G) and chrominance-based PPG (also known as C-PPG or C) to extract the PPG signal exclusively from the green channel of an RGB picture.

Numerous applications and investigations of GANs have evolved for face completeness, facial attribute manipulation, frontal view synthesis, facial reenactment, identity preservation, and expression altering. Because of Viola Jones, face images and videos have been viewed as an essential application area for computer vision. Due to advances in the generative capacity, realistic representation, but also effectiveness of Photographs was taken and Generative adversarial for facial recreation and video synthesizing the "deep fake" idea has emerged, which replaces one person's face with another as seamlessly as possible in one video. Autoencoders trained on the source and the target videos are believed to be used in the deep-fake generator. It's important to keep encoder weights the same so that generic features may be contained in the decoder and face-specific characteristics can be put into the decoder. Additionally, Face2Face S. Mandelli et al (2018). Uses a video reconstruction method to distort a target face in real-time based on a mix of the source video's blend shapes. Vid2vid and Deep Video Faces use GANs rather than blending shapes to get this effect. Due to lighting, occlusion, compression, and rapid movements, there are still missing frames and misaligned faces.
Image forensics traditionally uses physical or statistical artifacts to verify an image's content. In latest survey or publications, an overview of these approaches may be obtained. Inconsistencies in lighting and reflections are examples of physics-based techniques. A variety of statistical algorithms have been developed to identify manipulations, confirm noise presumptions from metadata, or learn about modification trails, such as recolouring or recompression T. Baltrušaitis et al (2016).

Several studies have investigated towards identifying deepfake video modifications since they emerged in early 2018. Handcrafted aspects Y. Li, M.-C. Chang et al (2018). ncluding blinking irregularities, biological signs, and artificial details were addressed. This manual detection features exploits known flaws in creation procedures. The cycle continues again as deepfake production techniques adjust to avoid detection. Recent detection approaches use deepfake datasets to identify individual forgeries in actual footage.

This method relies on a shallow neural network to identify forgeries at a microscale (or intermediate) level of detail, avoiding tiny details lost in video compression. They also improved their model by

replacing conventional convolution modules by using MesoInception blocks. Suggest using capsule networks Y. Li, M.-C. Chang et al (2011). To identify replay assaults and computer-generated pictures and movies. They claim that dynamic routing between capsules increases the probability of identifying high-quality counterfeit S. Sabour, N. Frosst and G. E. Hinton et al (2017). An autoencoder-based architecture was used for transfer learning by Cozzolino et al. A decoder that also learns to produce a mask of the altered area was added. XceptionNet is a promising deep neural network for extracting features when trained with ImageNet J. Deng et al (2009). eights. So, we employ XceptionNet rather than the networks used in previous research. A modified version of insight separable convolution layer is used to reduce the number of parameters while improving performance.

S. Agarwal et al (2019). Approached the challenge as one of detecting aberrant activity.They extract face landmarks and temporal activities from real-time films and utilize these to train just one SVM. Although this approach cannot be utilized on unexpected faces, it is a good alternative for future deepfake generating advancements. It would be better to control a detector to identify all government officials, business executives, or individuals of interest.

Table 1: Evaluation of benchmarking techniques' detection performance using the Accuracy value just at frame level as that of the assessment measure. The best and second-best performances are shown by boldface and underbar, respectively. The italics indicate that it does not provide AUC at the frame or video level. The AUC values for DefakeHop are presented at both the frame and video levels. The AUC values for benchmarking techniques were derived from Ruben Tolosana et al (2020), Yuezun Li et at (2020). A technique of deep learning, b method of superficial learning.

Table 1: AUC score of different techniques

	Method	Celeb-DF v1	Number of parameters
Zhou *et al.*.(2017)[12]	InceptionV3[a]	55.7%	24M
Afchar *et al.*.(2018)[13]	Meso4[a]	53.6%	27.9K
Li *et al.*.(2018) [14]	FWA[a] (ResNet-50)	53.8%	23.8M
Yang *et al.*.(2019 [15]	HeadPose[b] (SVM)	54.8%	-
Matern *et al.*.(2019)[16]	VA-MLP[b]	48.8%	-
Rossler *et al.*.(2019)[17]	Xception-raw[a]	38.7%	22.8M
Nguyen *et al.*.(2019 [18]	Multi-task[a]	36.5%	-
Tolosana *et al.*.(2020) [19]	*Xception*[a]	83.6%	22.8M
	DefakeHop (Frame)	<u>93.12%</u>	42.8K
	DefakeHop (Video)	**94.95%**	42.8K

3 Methodology

The proposed DefakeHop approach includes three main modules

1. PixelHop++

2. Feature Distillation Module

3. ensemble classification

The DefakeHop technique begins with face image preprocessing. Face pictures from video frames are cut out, aligned, and normalized before being sent to following meanings in the pipeline to guarantee accurate and consistent inputs. When DefakeHop preprocesses, the software can handle a wide range of video formats. Here, can find a few more specifics. Initially, video frames are used to create a picture. Then, using an open-source toolkit called "OpenFace2," 68 facial markers are extracted from each frame. To ensure that all facial landmarks are extracted from a single frame, the faces are reduced in size to 128 x 128 pixels and rotated in specified coordinates. For PixelHop++'s input data, 32 x 32-pixel patches are cut from various sections of a head (e.g., the mouth, right eye, and left eye).

Openface is facial expression behaviour analysis and interpretation has grown in popularity recently. Sentiment analysis community and those interested in designing interactive apps based on face behaviour analysis may find OpenFace useful. For the first time, OpenFace T. Baltrušaitis et al (2016). Can recognize face landmarks, head poses, facial action units, and eye-gaze estimates. The basic OpenFace computer vision algorithms achieve cutting-edge outcomes in all the following activities. Our program also works in real-time and can be operated via a camera without any gear. OpenFace's lightweight messaging system enables for simple interaction with other apps and devices.

PixelHop++ collects deep and discriminating local features. As depicted in Figure. 2, its input is a 32 x 32 color picture of a face image. Users may choose block size and stride. This method may be repeated to create a bigger receptive field. The proposed DefakeHop system uses 3 PixelHop++ components in succession, each having a block size of 3 x 3 and a stride of 1. The first pixel block contains a flattened vector of 3 x 3 x K_0 variables, where K_0 = 3 again for RGB source. Filtration and dimension reducing the c/w Saab signal is transformed towards the compressed vector of size K_1 = 9 K_0 = 27 to create a visual features of dimensions K_{11}. Max-pooling. There is spatial redundancy between neighboring pixels because their blocks overlap. It uses the (2x2)-to-1x1 maximum pooling unit to further lower output spatial resolution.
The 3 c/w Saab transformation used to describe this technique. The color image of 32 x 32 x 3 is the node in the tree shown in Figure 2. The input vector for the first hop is 3 x 3 x 3, which equals 27. A regional average is followed by 26 resonance frequencies. Low-frequency (blue), mid-frequency (green), and high-frequency (red) are the three classifications that it uses (in gray). There are nodes in the tree for each of the channels. Low-frequency channel responses may be transmitted to the next stage for a further c/w Saab transform since high-frequency channel responses may be discarded due to insufficient spatial correlations. However, it has a more limited field of view. Increasing the depth of field reduces the amount of spatial information available, but the overall image is more comprehensive. Saab's channel-wise (c/w) transform is designed to reduce model size without sacrificing performance by using channel separability.

Feature Distillation Module:
After using PixelHop++ to extract features from a face, we are left with a modest yet effective collection of features. Despite this, PixelHop++'s output dimension is still too long to also be input it into a classifier. To put it another way, the first hop's output dimension is $15 \times 15 \times K_{11} = 225K_{11}$. To gain a concise description of whether a face is phony or genuine, it uses two ways.

Dimensional reduction**:** Due to the similarity of the input pictures, there are substantial correlations between both the spatially responses of 15 for just a given channel. So, we use PCA to reduce spatial dimension. To achieve a high compression ratio of N1 K11 dimension, we preserve the highest N1 PCA components.

Channel-wise Soft Classification**:** After reducing spatial and spectral redundancy, each hop yields K11 channels with N1 spatial dimension. It build a fuzzy classifier model for each channel. The soft choice indicates the likelihood of a channel including a fraudulent video. Various classifiers might be used here. The extended gradient boosting classifier (XGBoost) was chosen for our model since it has a small model size and is easy to train. To avoid overfitting, all soft classifiers use XGBoost with max-depth one.

It combine the probability of any and all channels to describe a face patch. The output dimension is K11, which is much smaller than PixelHop++'s output. In the end, a classifier will decide if this face patch is real or not.

Ensemble Classification Module:

A video clip's authenticity may be determined using soft selections from all facial regions (face patches) and selected frames. This is a regional grouping. [1] Because each facial region may respond independently, we aggregated their likelihood. Three facial regions mouth, right eye and left eye are examined.

It also concatenates the current frame and its six nearby frames for every picture to provide more information about the time. To conclude, we calculate the video clip's falsehood probability by averaging all its frames' probabilities. In the end, the decision may be made using different probabilities at the aggregate frame-level.

DeFakeHop was light wight and best state of art to identify the deepfake detection. The way it designed it gives the best results compared to other state of arts and it was easy to reproduce and modify as well.

DeFakehop performed on the different landmarks produced after the preprocessing. In this study landmark are chosen left eye , right eye and mouth.

4 Design Specification

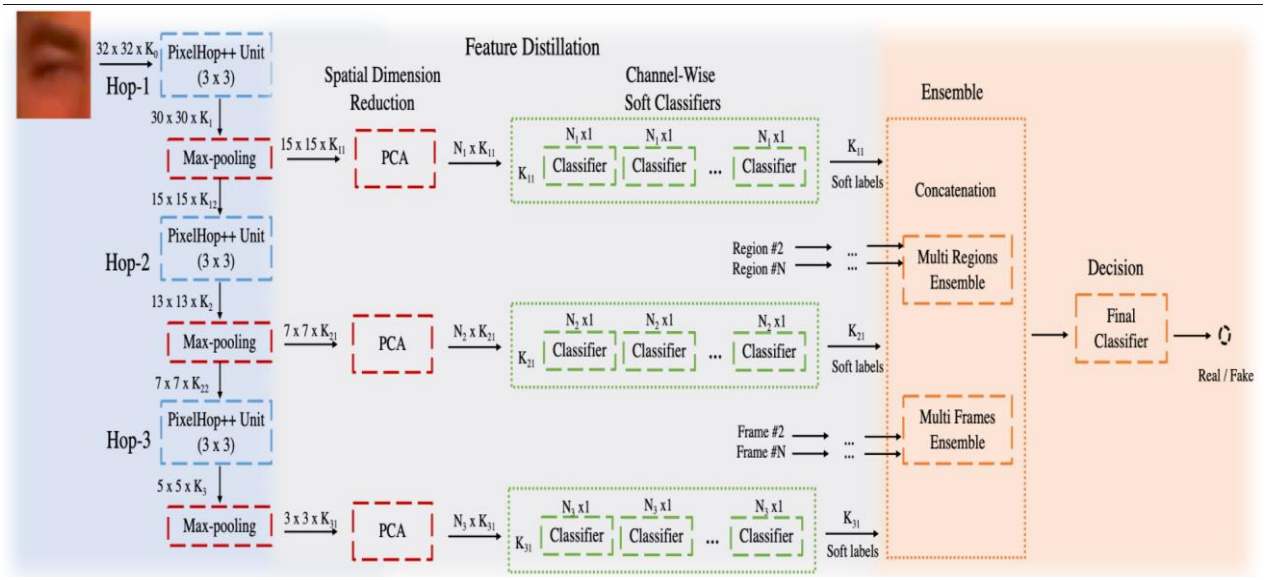

Fig. 2. The DefakeHop technique is described in detail.

The source is indeed a color picture of a human face with a resolution of 32 by 32 pixels. Hyperparameters such as block and duration may be set by the user. Multiple phases of this technique may be used to expand the receptive field. This program has three PixelHop++ subunits in cascade, each with a block size of 3 x 3 and

the duration identical to one, without padding, in the proposed DefakeHop implementation As a compressed vector, a chunk of a pixel in the first hop has 3x3xK0 = 9K0 variables. The root is a 32x32x3 color picture. The first hop's local input vector has a size of 333 = 27. We may get a local average and 26 center frequency as a result. Low-frequency, mid-frequency, and high-frequency channels (blue, green, and yellow respectively) are grouped into three categories (in gray). Tree-like representations of channels may be used to visualize the structure of the network. Due to poor spatial correlations, answers of high-frequency channels may be deleted, response of half channels are maintained, and responses of reduced channels are fed into the next step for a c/w Saab transform.

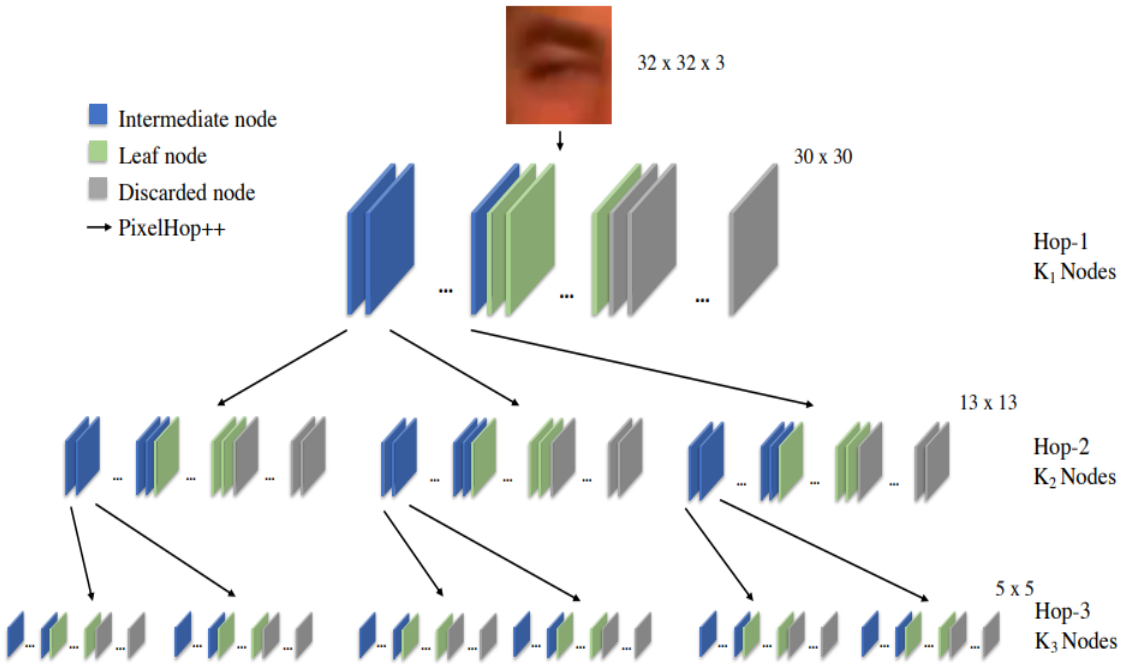

Fig. 3, a three-stage c/w Saab transformation is shown to describe this method.

There are spatial and spectral interpretations in each of the responses in Hop-1, Hop-2, and Hop3. There is greater spatial information in the first few hops, but they have a limited vision. Like this hop progresses, it loses spatial information however gains a wider perspective. Because of the channel separability used by the channel-wise Saab transform (c/w), its model size may be reduced without compromising performance.

5 Implementation

Large-scale datasets are required to create and test DeepFake detection systems. Current DeepFake datasets, on the other hand, have poor graphical fidelity and will not reflect DeepFake films that have gone viral online. This dataset, CelebDF[41], features high-quality DeepFake movies of celebrities made using an enhanced synthesis approach. It's a new and hard DeepFake dataset. DeepFake detection techniques and datasets are thoroughly evaluated to illustrate Celeb-increased DF's degree of difficulty. In celeb-df v1 there are two types of real videos such as celeb-real and YouTube real together 408 videos and there is another fake video folder which is celeb-df synthesis 795 videos. The format of video is MPEG4.

Pre-processing for Defakehop:

Initially in defakehop , from each video the frames are extracted with frame rate of 30 per second, from every

video of real and fake the frames are extracted altogether 29k frames. Experiment carries 30 frames for second for better read of the images to while training. In the landmark extractor

Fig 4: landmark extraction and

The process of open face is 300-W face validation dataset, four sub-datasets were evaluated: Annotation Face inside the Open Helen, LFPW, (AFW), and IBUG. We started with the challenge organizers' boundary boxes. Our suggested hierarchical model was first assessed. 6a shows the outcomes. The hierarchical approach improves facial landmark detection accuracy. In a second study, we compared our method to other online-available facial landmark identification algorithms trained to recognize the same face features (or their subsets). DRMF, tree-based deformable models, enhanced version of CLM , GNDPM (GNDPM) , and SDM (Supervised Descent Method).

OpenFace can work with live webcam feeds, recorded video files, picture sequences, and individual photographs. The processed data may be saved as CSV files for shape parameters, face landmarks, gaze vectors, and Action Units. In addition to HOG characteristics, aligned face pictures may be stored as image sequences or movies. The recorded behaviours may also be loaded using ELAN for convenient viewing. Saved facial behaviours may be used as characteristics in medical condition analysis, emotion prediction and social signal analysis systems. And lastly, OpenFace T. Baltrušaitis et al (2016). May be used to create real-time interactive apps based on face analysis algorithms This is accomplished by utilizing ZeroMQ 5. Anybody desiring the features may get them over a network. A similar technique has been employed in ophthalmology. It also provides examples for Python and C++ of real-time ZeroMQ message listening from OpenFace.

```
stream = os.popen(r"OpenFace\FeatureExtraction -f {input} -out_dir {output}"
        .format(input = file_path, output = output_dir))
```

Fig 5: it depicts code snippet for openface

This is the code snippets of openface. In open face there are executable file which is *FeatureExtraction* which crop the frames from video. In this passing the two variables, input is video file and output file is landmark folder where it stores the frames of each video which passes as video.

Patch extractor:

After the landmark extractor experiment carries the patch extractor which cut the whole image into the different parts such right eye, left eye and mouth. This experiment focusses only on these 3 areas in face. For patch extractor the video is given as input, and code written like the it extracts the regions which provided in the region array.

```
regions = ["left_eye", "right_eye", "mouth"]
```

Fig 6 : it demonstrating the regions in patch_extractors.

And after passing the regions array each frame crop the landmarks with passing the regions. Later it creates each folder for right eye, left eye and mouth for video and it cuts the frame for that region. Here it considering the one frame for every 6 frames in avoid the same repeating of frames. Finally it produce the output of image and .npy file for every image of each frame as input. Another file with test and train the whole patch files is data file. In this file it generates the separate files for training and testing .npy file by combining the all the .npy files for each respective region.

Units of PixelHop++ All three hops feature 3x3 filters. It is possible to have a maximum of 10 PixelHop++ units. There are three channels in the input for Hop-1, resulting in a 27D input image. Channel-wise technique using Hop-2 and Hop-3 results in 9D input vectors since each input has just one channel. For Hop-2 and Hop-3, numerous c/w Saab morphs are available. There are numerous c/w Saab transforms to choose from when selecting the channel. It is possible to acquire channels for both Hop-2 and Hop-3 that are more than 9 channels. Three PixelHop++ units employ a maximum of 27 × 10, 9×10, 9×10 parameters.
PCA-Sp Hop-1 goes between 225 to 45, Hop-2 both over 49 to 30, while Hop-3 from 9 to 5. Spatial PCAs have 22545, 4925, and 9 5 parameters. The chosen number of retained channels is 45, 25, 5.

The tree count is 100. Each tree has branch and leaf nodes. The tree structure determines the anticipated value while the intermediate nodes decide the dimension and border to divide. The settings are 400 and 16,000 for max-depths of 1 and 6. Each XGBoost has a maxdepth of 1, 6, or 12. The overall number of parameters for 30 stream XGBoosts is 12,000, whereas the ensemble XGBoost has 19,000 parameters.
The final model shape, 75235, is an upper bound estimate due to the XGBoost maximum depth and channel number per hop.

6 Evaluation

For the celeb-df dataset, DefakeHop achieved an AUC of 94.6 percent, while the online scanner achieved an accuracy rate of 85.18 percent. DefakeHop was performed with the different data splits, for each split it produces the different AUC scores. Interestingly when this experiment carries some random splits the AUC values are less than the value actual experiment.
This defakehop performed with three times with three different dataset splits:

6.1 Case studt 1: List of testing videos: The dataset includes a list of testing videos, and in this split the films are arranged according to this list

6.2 Case studt 2: Random split: In this split the video are divided based 80% videos are trained and 20% videos are given as the testing

6.3 Case studt 3: Manual split: Defakehop is experimented with the manual split due to while considered the first two split observed that whatever the videos that are training not in even few videos are covering in the split. In the first split where list of videos is already given, in that provided only 100 videos out of the 1203 videos which is just 8% of total. Even in that video observed that fakes are given only few celeb faces. By considering that this manual split taken place such way

that considered the minimum few set of faces are trained and along with at least one set of celeb faces are given for the testing.

Table 2: List of AUC scores with different spllits

	frame	video
List of testing videos	93.9	94.6
Random split	89.6	93.4
Manual split	88.6	92.8

6.4 Online Scanners:

The aim for project is comparing between the state of arts with online scanner and compare the accuracy between them. For that two online sources have been take Weverify [7] and Deepware [8].
Deepware[8] having a westie where user can upload video can find the deepfakes in that video and it gives the where deepfake is found are not or suspicious or no deepfake is found. At the end provided the probability of the deepfake in video. Where Weverify is also having the web portal to access that one need to register and get access for model to use. This study managed to compare the all the testing video probabilities with state of art in excel sheet. This can be compared between the accuracy between the subsets like male vs female, and celeb-real vs YouTube real vs celeb-synthesis

Table 3: female vs male accuracy between the defakehop and online scanners

	female	male
Defakehop	90.5	69.2
Deepware	98.6	92.3
Weverify	79.6	61.5

Table 4: Table 3: youtube real vs celeb-real vs celeb- synthesis accuracy between the defakehop and online scanners

	Youtube real	Celeb real	Celeb synthesis
Defakehop	90	37.5	88.7
Deepware	93.3	87.5	100
WeVerify	63.3	62.5	82.3

6.5 Discussion:

Among these splits can observe that the values are larger with the list of testing videos rest comparing with

the other two splits.

And, in this comparative study the two online scanners have been used. One is WeVerify[7] and another one is Deepware[8]. In deepware it is possible to check one video at a time, after passing the video it produces the results in such way that shows the percentage of deepfake in that video and display whether the deepfake detected or not with speedometer indicated the percentage. And it displays to some of the real video as suspicious when their deepfake percentage is around 40 - 60. The overall accuracy of the deepware is 97%.

Another online scanner which is WeVerify regarding the training setup, employ an ensemble of five models with just an EfficientNet backbone that was pre-trained on ImageNet; four of the models then refined on the DFDC dataset, while the fifth model was easily adjusted just on WildDeepFake *Bojia Zi et al (2020).* dataset. It also provides the interactive way where can upload the video where it also performs the frames cropping etc. After testing whole test videos system achieved an accuracy of 75%, an ROC-AUC of 85.18%, an F1 score of 80.31% and a log-loss of 0.4517.

While scanning with online sources observed that some real videos have tested as fake video and some of fake videos are tested as real videos. Real videos are detected as fake video due to video quality and somewhere tested like 0(zero) percentage which celebs are popularly known as sports persons and film actors.

From the table 3 and 4 can observe that comparatively the defakehop gave the best results. And in among the three arts deepware performed well and gives the best outcome in detecting the videos. Weverify also performed but it pretrained on the WildDeepFake *Bojia Zi et al (2020)* which gave near values to the other two arts.

As mentioned in the research question this study is about the comparative study between state of art DeFakhop and online scanners. DeFakehop was performed on the different splits along with the original split which was mentioned in the paper. And the online scanners were also produced the results on the original split to compare with the state of the art.

7 Conclusion and Future Work

In this research, we examined deepfake detection and one state-of-the-art technique for determining the accuracy of two online sources. Throughout this study, the usage of image separation processes such as openface resulted in higher quality deepfake photos, which seem more authentic than the normally low-quality images seen on the internet. The defakehop is light weight component for easy test and train the deepfake videos. It is performing with more accuracy for few selected videos rather than on different splits. The AUC scores of the defakhop when performed on the list of provided testing video is very accurate which is 93.9 and 94.6 for frames and video respectively. Where the results are changed when the splits are increased from the percentage of approx. 8 to 20 percentage split of testing videos, the values are 89.6 frames and 94.6 video and 88.6 frames and 92.8 video of random and manual split respectively. Along with that there are two other online scanner whose accuracies are 75 and 97 of Weverify and Deepware respectivel, among all the arts defakehop gave the nearer values to the other two online scanners which are Weverify and Deepware.

When it comes to future work regarding this study there are few areas where can focus more like regarding the deepfake performance, where it is taking a lot of times when it is running complete dataset like celeb-df. And in defakehop regions are selected as left eye, right eye and mouth along with that can crop other patches like nose and forehead. And, can find another state of art which gives the accuracy like defakehop, can also perform on different dataset. Also, investigate how to generate deepfakes with fewer defects and a better-

quality picture utilizing image enhancement techniques in order to make them more difficult to identify using deepfake detection methods. It is very much clear when there is creation and detection.

8 Acknowledgment

I'd want to express my gratitude to Arghir-Nicolae Moldovan, my supervisor. He has been tremendously patient with my continual monitoring of my research endeavour and has been helpful as he assures the study's success. I would like to gratitude to Arghir-Nicolae Moldovan for almost all of his assistance in assisting me in remaining on track with my development. Without Arghir-Nicolae Moldovan, this research would not have been feasible. The public has taken notice of his willingness to exert such effort.

References

1.“FakeApp,” https://www.malavida.com/en/soft/ fakeapp/.

2. “DFaker github,” https://github.com/dfaker/df.

3. “faceswap-GAN github,” https://github.com/ shaoanlu/faceswap-GAN.

4. “faceswap github,” https://github.com/ deepfakes/faceswap.

5. “DeepFaceLab github,” https://github.com/ iperov/DeepFaceLab.

6. H. -S. Chen, M. Rouhsedaghat, H. Ghani, S. Hu, S. You and C. . -C. Jay Kuo, "DefakeHop: A Light-Weight High-Performance Deepfake Detector," *2021 IEEE International Conference on Multimedia and Expo (ICME)*, 2021, pp. 1-6, doi: 10.1109/ICME51207.2021.9428361.

7. “Weverify” , https://weverify.eu/tools/deepfake-detector/

8. “Deepware”, https://scanner.deepware.ai/

9. Xin Yang, Yuezun Li, and Siwei Lyu, “Exposing deep fakes using inconsistent head poses,” in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, pp. 8261–8265.

10. Shruti Agarwal, Hany Farid, Yuming Gu, Mingming He, Koki Nagano, and Hao Li, “Protecting world leaders against deep fakes.,” in CVPR Workshops, 2019, pp. 38–45.

11. Falko Matern, Christian Riess, and Marc Stamminger, “Exploiting visual artifacts to expose deepfakes and face manipulations,” in 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW). IEEE, 2019, pp. 83–92.

12.Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis, “Two-stream neural networks for tampered face detection,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, 2017, pp. 1831–1839.

13. 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). IEEE, 2018, pp. 1–7.

14. Xin Yang, Yuezun Li, and Siwei Lyu, “Exposing deep fakes using inconsistent head poses,” in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, pp. 8261–8265 Huy H Nguyen, Fuming Fang, Junichi Yamagishi, and Isao Echizen, “Multi-task learning for detecting and segmenting manipulated facial images and videos,” arXiv preprint arXiv:1906.06876, 2019.

15. Huy H Nguyen, Junichi Yamagishi, and Isao Echizen, “Use of a capsule network to detect fake images and videos,” arXiv preprint arXiv:1910.12467, 2019.

16. Huy H Nguyen, Fuming Fang, Junichi Yamagishi, and Isao Echizen, “Multi-task learning for detecting and segmenting manipulated facial images and videos,” arXiv preprint arXiv:1906.06876, 2019.

17. Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia, “Deepfakes and beyond: A survey of face manipulation and fake detection,” arXiv preprint arXiv:2001.00179, 2020.

18. Huy H Nguyen, Junichi Yamagishi, and Isao Echizen, “Use of a capsule network to detect fake images and videos,” arXiv preprint arXiv:1910.12467, 2019

19. Ruben Tolosana, Sergio Romero-Tapiador, Julian Fierrez, and Ruben Vera-Rodriguez, “Deepfakes evolution: Analysis of facial regions and fake detection performance,” arXiv preprint arXiv:2004.07532, 2020

20. Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia, “Deepfakes and beyond: A survey of face manipulation and fake detection,” arXiv preprint arXiv:2001.00179, 2020.

21 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, 2020, pp. 3207–3216

23. N. B. A. Warif, A. W. A. Wahab, M. Y. I. Idris, R. Ramli, R. Salleh, S. Shamshirband, and K.-K. R. Choo, “Copy-move forgery detection: Survey, challenges and future directions,” Journal of Network and Computer Applications, vol. 75, pp. 259 – 278, 2016

24. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Advances in Neural Information Processing Systems 27. Curran Associates, Inc., 2014, pp. 2672–2680.

25. Y. Aytar, L. Castrejon, C. Vondrick, H. Pirsiavash, and A. Torralba. Cross-modal scene networks. arXiv preprint arXiv:1610.09003, 2016.

26. A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. arXiv preprint arXiv:1612.07828, 2016.

27. K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. arXiv preprint arXiv:1612.05424, 2016

28.] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In NIPS, pages 82–90, 2016.

29.G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. Science, 313(5786):504–507, July 2006

30. J. Masci, U. Meier, D. Cires¸an, and J. Schmidhuber. Stacked convolutional auto-encoders for hierarchical feature extraction. In International Conference on Artificial Neural Networks, 2011

31. A. Zhmoginov and M. Sandler. Inverting face embeddings with convolutional neural networks. arXiv:1606.04189, June 2016.

32.E. Grant, P. Kohli, and M. van Gerven. Deep disentangled representations for volumetric reconstruction. In ECCVW, 2016

33. A. Tewari, M. Zollhofer, H. Kim, P. Garrido, F. Bernard, P. Perez, and C. Theobalt, “Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction,” in The IEEE International Conference on Computer Vision (ICCV), Oct 2017

34. W. Shen and R. Liu, “Learning residual images for face attribute manipulation,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1225–1233, 2017.

[35] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, “Stargan: Unified generative adversarial networks for multi-domain image-toimage translation,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018.

[36] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, “Arbitrary facial attribute editing: Only change what you want,” arXiv:1711.10678, 2017.

37. D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, “Mesonet: a compact facial video forgery detection network,” Proceedings of the IEEE International Workshop on Information Forensics and Security, pp. 1–7, December 2018, Hong Kong

38. X. Yang, Y. Li, and S. Lyu, “Exposing deep fakes using inconsistent head poses,” Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 8261–8265, May 2019, Brighton, United Kingdom.

39. L. Verdoliva, “Media forensics and deepfakes: an overview,” 2020.

40. A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and ¨ M. Nießner, “Faceforensics++: Learning to detect manipulated facial images,” CoRR, vol. abs/1901.08971, 2019. [Online]. Available: http://arxiv.org/abs/1901.08971

[41] P. S. H. Q. Yuezun Li, Xin Yang and S. Lyu, “Celeb-df: A new dataset for deepfake forensics,” arXiv preprint arXiv:1909.12962, 2019.

42. S. Iizuka, E. Simo-Serra, and H. Ishikawa. Globally and locally consistent image completion. ACM Transactions on Graphics (ToG), 36(4):1–14, 2017.

43. O. Langner, R. Dotsch, G. Bijlstra, D. HJ Wigboldus, S. T Hawk, and AD Van Knippenberg. Presentation and validation of the radboud faces database. Cognition and emotion, 24(8):1377–1388, 2010

44. A. C Popescu and H. Farid. Exposing digital forgeries in color filter array interpolated images. IEEE Transactions on Signal Processing, 53(10):3948– 3959, 2005.

45. A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies and M. Niener., "FaceForensics++: Learning to Detect Manipulated Facial Images", Jan. 2019. [Online] Available: https://arxiv.org/abs /1901.08971

46. Kim DaeHee, Choi SeungWan, Kwak SooYeong., "Deep Learning Based Fake Face Detection", Journal of the Korea Industrial Information Systems Research, vol. 23, no. 5, pp. 9-17, Oct. 2018.

47. Li, Y., Chang, M. C., and Lyu, S., "In ictu oculi: Exposing ai generated fake face videos by detecting eye blinking", Jun. 2018. [Online] Available: https://arxiv.org/abs/1806.02877

48. S. Lawrence, C. L. Giles, A. C. Tsoi, A. D. Back, "Face recognition: A convolutional neural-network approach", IEEE Trans. Neural Netw., vol. 8, no. 1, pp. 98-113, Jan. 1997.

49. Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Chris- ¨ tian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. arXiv preprint, arXiv:1901.08971, 2019.

50. J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, ¨ “Face2Face: Real-time Face Capture and Reenactment of RGB Videos,” in Proc. Computer Vision and Pattern Recognition (CVPR), IEEE, 2016.

51. S. Mandelli, N. Bonettini, P. Bestagini, V. Lipari, and S. Tubaro. Multiple JPEG Compression Detection Through Task-Driven Non-Negative Matrix Factorization. In Acoustics, Speech and Signal Processing, 2018 IEEE International Conference on (ICASSP), pages 2106–2110, 2018.

52. T. Baltrušaitis, P. Robinson and L. Morency, "OpenFace: An open source facial behavior analysis toolkit," *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016, pp. 1-10, doi: 10.1109/WACV.2016.7477553.

53. Y. Li, M.-C. Chang, H. Farid, and S. Lyu, “In ictu oculi: Exposing ai generated fake face videos by detecting eye blinking,” 2018, arXiv:1806.02877.

54. G. E. Hinton, A. Krizhevsky, and S. D. Wang, “Transforming autoencoders,” in Int. Conf. Artif. Neural Netw. Springer, 2011, pp. 44–51.

55. S. Sabour, N. Frosst, and G. E. Hinton, “Dynamic routing between capsules,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 3856–3866.

[56] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vision Pattern Recognit., 2009, pp. 248–255.

57 . S. Agarwal, H. Farid, Y. Gu, M. He, K. Nagano, and H. Li, “Protecting world leaders against deep fakes,” in Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops, 2019, pp. 38–45.

58. Bojia Zi, Minghao Chang, Jingjing Chen, Xingjun Ma, and Yu-Gang Jiang. 2020. WildDeepfake: A Challenging Real-World Dataset for Deepfake Detection. In Proceedings of the 28th ACM International Conference on Multimedia (MM '20). Association for Computing Machinery, New York, NY, USA, 2382–2390. DOI:https://doi.org/10.1145/3394171.3413769

59. S. Lyu, "Deepfake Detection: Current Challenges and Next Steps," *2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 2020, pp. 1-6, doi: 10.1109/ICMEW46912.2020.9105991