

Dataset Generation and Study of Deepfake Techniques

Sergio A. Falc3n-L3pez¹, Antonio Robles-G3mez¹[0000-0002-5181-0199], Llanos Tobarra¹[0000-0003-2779-4042], and Rafael Pastor-Vargas¹[0000-0002-4089-9538]

Universidad Nacional de Educaci3n a Distancia
ETSI Inform3tica; C/Juan del Rosal, 16, 28040 Madrid, Spain
arobles@scc.uned.es

Abstract. The consumption of multimedia content on the Internet has nowadays been expanded exponentially. These trends have contributed to fake news can become a very high influence in the current society. The latest techniques to influence the spread of digital false information are based on methods of generating images and videos, known as *Deepfakes*. This way, our research work analyzes the most widely used Deepfake content generation methods, as well as explore different conventional and advanced tools for Deepfake detection. A specific dataset has also been built that includes both fake and real multimedia contents. This dataset will allow us to verify whether the used image and video forgery detection techniques can detect manipulated multimedia content.

Keywords: Deepfake · Dataset Generation · Detection Techniques · Multimedia Manipulation.

1 Introduction

Any news can nowadays travel around the world in a few minutes. If the news that is transmitted also contains false information, the consequences could reach the point of manipulating the opinion of society [6]. For instance, society could watch a fake video in which an election candidate is performing immoral acts or even a photo expressing symbols or ideals against a campaign. Not only the detection of this information is necessary, but it also has to be effective in time, since the generation of false evidence can even violate the integrity of people.

In addition to this, the consumption of information on the Internet is almost unlimited, mainly due to the increment of mobile devices usage around the world. These devices make possible to create a large amount of daily multimedia content at an incredible speed of propagation. Message applications, social networks, and news websites allow false content to spread easily. For this reason, it is necessary to continue investigating to detect false information as quickly and rigorously as possible.

In the world of cybersecurity, criminals never rest, and they are analyzing new technologies with which to satisfy their interests regardless of the consummation of some crimes in the process. Current applications are sometimes not capable of

detecting these cutting-edge manipulations, since criminals also rely on forensic techniques. Therefore, it is important to continue with continuous research that allows, through the use of new techniques and technologies, to detect and contain the achievement of crimes.

Therefore, the main objective of this work is to study of Deepfake techniques for image and video manipulation. The use of tools capable of generating Deepfakes will be studied, as well as Deepfake-oriented detection techniques are also tested. A specialized Deepfake dataset has also been generated, including both fake and real multimedia content. This dataset will allow us to study deeply the manipulation of multimedia content.

This paper is organized as follows. In Section 2, the principal Deepfake generation and detection techniques are studied. After that, Section 3 details the repository created and the used tools for its generation. Finally, Section 4 ends with some conclusions and possible future work.

2 Deepfake Studies

2.1 Generation Techniques

The main techniques known for the generation of Deepfakes are based on solutions implemented in a basic way through GANs (Generative Adversarial Networks), and several variations. The research community especially distinguishes four different methods for the generation of fakes in humans, based on the main types of existing facial manipulations. According to [21], the following methods are identified: *Entire Face Synthesis*, *Identity Swap/Faceswap*, *Attribute Manipulation*, and *Expression Swap*.

The *Entire Face Synthesis* technique consists of creating non-existent complete faces. An approach can be found in [13]. There is another work [14], where faces of people who do not exist [10] are generated. The *Identity Swap/Faceswap* consists of replacing the face of a person in a video with the face of another person. The *Attribute Manipulation* technique is also known as face editing or facial retouching. An image in which a face appears is used as a source, and the manipulation consists of modifying some of the attributes of the face, such as the color of the hair and skin, generating an appearance of the face that is more aged or even more rejuvenated, and adding elements to the face such as glasses or other accessories. One example can be found in [12].

Finally, the *Expression Swap* also known as *face reenactment*, consists of modifying the person’s facial expression. The most popular approaches are based on Face2Face [20], which uses computer graphics to get the first few frames of a video. From these frames, a temporary facial identity is obtained, and from there, all subsequent facial expressions are tracked. Another of the most prominent approaches, such as Neural Textures [19], is based on neural textures, carrying out a rendering approach that uses the original video data to learn a neural texture of the target person, including a rendering network.

Different tools capable of generating Deepfakes have been used in this work to cover the four generation techniques mentioned above: Entire Face Synthesis

through the Dall-E2 [2] project; Identity Swap/FaceSwap via DeepFaceLab [3]; Attribute Manipulation with the FaceApp [4] mobile application; and Expression Swap using Avatarify [1].

2.2 Detection Techniques

As a second part of the review, a set of Deepfake detection tools are studied. First, we have explored the tools that use conventional anomaly detection techniques in images and videos. Specifically, the application *Forensically* [5] will allow us to apply different techniques based on blind methods since, when analyzing Deepfakes, we do not have data collections related to the image or video to analyze. Results of the set of techniques applied with this tool must be evaluated by a human agent, so they are left to the criteria and level of expertise of the forensic analyst. For the analysis of images and videos, the same process must be carried out by taking into consideration that, with the videos we have analyzed individually the frames that compose them. The process consists of using the different utilities offered by the tool for each content.

According to this, the *MantraNet* application is capable of detecting several manipulation techniques and even unfamiliar ones. As a result, this tool returns a map of manipulation probabilities for each pixel in an image to be checked. The code of the notebook that is provided in [7] repository has been adapted to our purposes [18]. It is not available for videos.

The *Image Forgery Detection with CNNs* tool, which also uses Deep Learning, detects tampering based on the use of copy-move, delete and splice techniques. The script provided by the tool has been adapted to read all images found in the directory at the same time. This script returns 1 if the image has been manipulated and 0 if the image is not considered as manipulated. The adapted script has been run with the images from our repository, using for each run a different model, available in the application repository. The details about the parameters are described in [16]. Since this application is not designed for video detection, the model with the best statistics was selected and applied to all video frames in the repository.

Two plug-ins designed for the forensic analysis tool, *Autopsy*, have also been used. One of the plugins is used to detect manipulated images and another one is used to detect videos. The plugin can be downloaded and installed by following the instructions in the repository [17]. Once the application is installed and the plugins are placed in the corresponding directory, Autopsy has been loaded to launch the plugins against our repository. To use the image plugin, we activated the module called “Detect photo manipulations”. To launch the module to detect Deepfake in videos, we activate the module “Detect Deepfake videos”. Once the fingerprinting is completed, this tool returns the probability of tampering for images. For Videos, the tool returns whether they are Deepfake or not. As an example, Fig. 1 shows a set of images with their probability of being manipulated, among other options.

The *MesoNet* tool is based on the technique shown in [11]. The Python code located in [8] has been used for our purposes. Two already trained models can

The screenshot shows the Autopsy 4.15.0 interface. The main window displays a table titled 'Manipulation scores' with the following columns: Source File, S, C, Probability of being manipulated, and Data Source. The table lists various image files (e.g., D00001.jpg, D00002.jpg, etc.) and their corresponding probabilities. A red arrow points to the 'Manipulated source' column header in the table.

Source File	S	C	Probability of being manipulated	Data Source
D00001.jpg			0.02663748	LogicAI/NetS1
D00002.jpg			0.05667362	LogicAI/NetS1
D00003.jpg			0.1373051	LogicAI/NetS1
D00004.jpg			0.02795251	LogicAI/NetS1
D00005.jpg			0.00949403	LogicAI/NetS1
D00006.jpg			0.0766831	LogicAI/NetS1
D00007.jpg			0.0346174	LogicAI/NetS1
D00008.jpg			0.17488215	LogicAI/NetS1
D00009.jpg			0.73364695	LogicAI/NetS1
D00010.jpg			0.13801898	LogicAI/NetS1
FAR0001.jpg			0.11234033	LogicAI/NetS1
FAR0002.jpg			0.58753279	LogicAI/NetS1
FAR0003.jpg			0.19290316	LogicAI/NetS1
FAR0004.jpg			0.11478171	LogicAI/NetS1
FAR0005.jpg			0.95041595	LogicAI/NetS1
FAR0006.jpg			0.84954231	LogicAI/NetS1
FAR0007.jpg			0.69965046	LogicAI/NetS1
FAR0008.jpg			0.88148594	LogicAI/NetS1
FAR0009.jpg			0.69471872	LogicAI/NetS1
FAR0010.jpg			0.60016385	LogicAI/NetS1
FAR0011.jpg			0.98870758	LogicAI/NetS1
FAR0012.jpg			0.10111111	LogicAI/NetS1

Fig. 1. A set of images with their probability of being manipulated in the Autopsy tool.

be found in the repository, both of which we are to use for testing: Meso-4 and MesoInception4. For image detection in MesoNet, its code has been modified so that it only analyzes images, and the displayed result has been customized. The required adaptations have also been made to analyze only the videos in the repository.

Finally, the *Deepware Scanner* application is an online tool that is based on various Deepfake video detection projects. This application implements various Deepfake detection methods for videos. All we have to do is go to the Deepware Scanner section and, then, upload the video we desire to analyze. Next, a report will indicate if it is Deepfake and the probability of detection of the different models used. An example is shown in Fig. 2.

3 Deepfake Dataset


3.1 Discovery Tools

A Deepfake consists of and its generation, the discovery of techniques capable of detecting anomalies in both images and video begins, which are often used in the forensic analysis environment. The first open tool that has been found and used for the generation of videos or images with Deep Learning has been DeepFaceLab [3].

One of the applications that has been used to generate images has been the FaceApp application on an Android phone [4]. The images that come in the demo have been used to generate fake images, since it is not allowed to test own images for free of charge.

New advances in the creation of images from descriptive texts have been explored, as occurs in the Dall-E2 project. This solution is based on an online tool, which can currently be accessed from [2]. An example of this tool can be observed in Fig. 3.

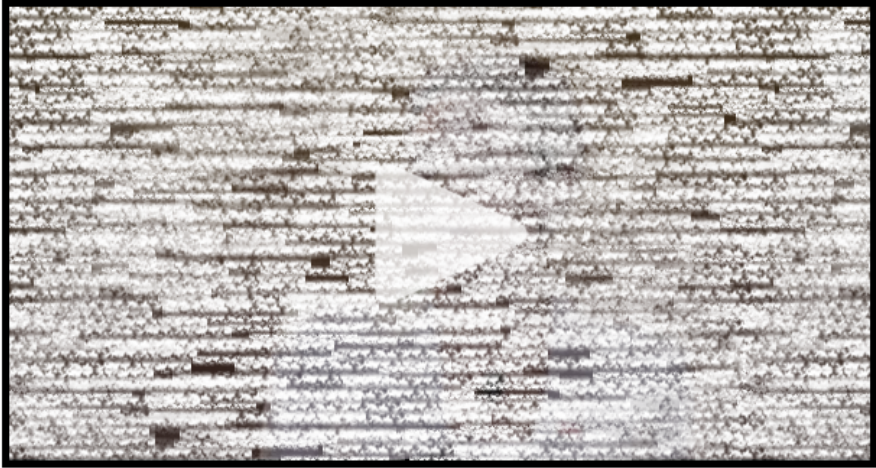
! **DEEFAKE DETECTED**
New Scan



Name:	DFL0001.mp4	User	2022-09-12 13:49:42 UTC
Size:	13.1 MB	Source	3 day(s) ago

DETAILS

Deepware aims to give an opinion about the scanned video and is not responsible for the result. As Deepware Scanner is still in beta, the results should not be treated as an absolute truth or evidence.



Model Results

Avatarify: SUSPICIOUS(58%)

Deepware: DEEFAKE DETECTED(93%)

Seferbekov: DEEFAKE DETECTED(99%)

Ensemble: DEEFAKE DETECTED(98%)

Video

Duration: 13 sec

Resolution: 952 x 500

Frame Rate: 30 fps

Codec: h264

Audio

Duration: -

Channel: -

Sample Rate: -

Codec: -

Fig. 2. Use of Deepware Scanner for an image to detect the Deepfake probability.

When accessing this website [10], the image of the face of a person who “does not exist” will directly appear, and that image has been generated through the implementation of the generation technique described in [14], using a StyleGan. This implementation has been used to download Deepfake images, generated entirely synthetically. For this, we have only accessed the web and downloaded the generated image.

Different fake video-generating apps generally use face swapping. So, it was decided to use Avatarify Desktop, which implements the expression transfer technique by obtaining facial expressions captured through a camera and transferring

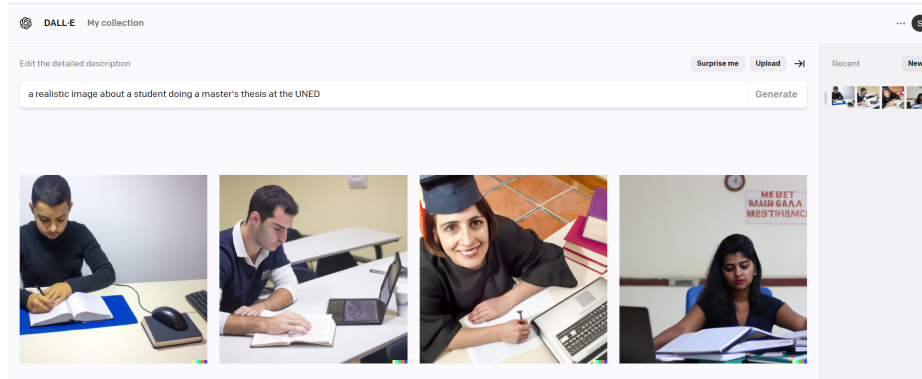


Fig. 3. Interface of the Dall-E2 tool by including some examples.

them to a face photo in real time. We can download it from the following project: Avatarify Desktop [1].

As there are several techniques for generating false multimedia content, the applications mentioned above have been used to create images and videos with different methods so that the techniques for detecting anomalies in images and videos can then be tested. For this reason, it has been decided to create a repository [18], in which there are images and videos generated with DeepFaceLab, images created with FaceApp, Dall-E2 and Thispersondoesnotexist.com, and finally videos generated with Avatarify. In order to discriminate whether the detection tools are capable of distinguishing between real images or videos, real multimedia content has been added from repositories used in other research on Deepfakes.

3.2 The Generated Dataset

In order to analyze the different Deepfake detection tools, a dataset has been created that includes images and videos generated by different applications in order to test the detection capacity of these tools [18].

For the generation of images, the applications *FaceApp*, *Dall-E2* and the images generated from project *thispersondoesnotexist* [10] and *DeepFaceLab* have been used. It has been decided to generate the content with several applications because each one generates a deepfake with different methods and produces images with different results and patterns. For example, DeepFaceLab performs Faceswap, and what it does is exchange the face generated in the original content, which suggests that the splice detection tools (slicing) could have a high probability of detecting the fake, while the images generated by Dall-E2 are completely synthetic, and it could happen that these detection techniques do not work for these cases. With the *FaceApp* application, they have been used as input images tagged as originals from the Celeb-DF repository,[15]. With *Thispersondoesnotexist* and *Dall-E2* completely synthetic images have been obtained. Lastly, we

have used the frames generated by the application *DeepFaceLab*, which it uses to generate deepfake videos.

For the generation of videos, the application *DeepFaceLab* has been used, whose source and destination videos, labeled as real, have been selected from the dataset Celeb-DF, [15], Finally, *Avatarify* has been used to generate deepfake videos through the *expression swap* method. A selection of those images labeled as real from the dataset, Celeb-DF [15], have been used as source images. Next, using a video camera, the expression of a real face has been transferred to the image, and the process has been recorded to generate a final video captured with OBS Studio [9].

Table 1. Dataset features for generated Deepfake images. The repository is located in [18].

Prefix	Tool	Description	Example
FA	FaceApp	4 images from the demo images of the application itself.	FA0001.jpg
FARC	FaceApp	11 images modified from real images of the Celeb-DF dataset [15].	FARC0001.jpg
D	Dall-E2	A total of 20 images: 10 images generated by Dall-E2 (in PNG format); and 10 images converted to JPG format.	D0001.jpg
TPNE	Thispersondoesnotexist	9 web-generated images [10], which implements [14].	TPNE0001.jpg
DFL	DeepFaceLab	A total of 16 images: 8 frames taken from Deepfake videos generated with face swapping (in PNG format); and 8 converted to JPG format.	DFL00006.png

Table 2. Dataset features for generated Deepfake videos. The repository is located in [18].

Prefix	Tool	Description	Example
DFL	DeepFaceLab	A total of 4 videos: 3 original videos from the Celeb-DF repository [15]; and a video generated from DeepFaceLab demo videos.	DFL0001.mp4
AV	Avatarify	4 videos based on real images from the Celeb-DF repository [15].	AV0001.mp4

The generated multimedia content has been uploaded to the dataset and has been structured in two directories:

Table 3. Dataset features for real images. The repository is located in [18].

Prefix	Tool	Description	Example
RC	Celeb-DF	11 images from the “img_align_celeba.zip” in the Celeb-DF repository [15].	RC0001.jpg
RVC	Celeb-DF	6 images from videos of the Celeb-DF repository [15]; Celeb-real directory in “Celeb-DF.zip”.	RCV00025.jpg
RDFL	DeepFaceLab	A total of 4: 2 images extracted from DeepFaceLa (in PNG format); and 2 images converted to JPG format.	RDFL00009.jpg

Table 4. Dataset features for real videos. The repository is located in [18].

Prefix	Tool	Description	Example
C	Celeb-DF	3 videos of the Celeb-DF repository [15]; Celeb-real directory in “Celeb-DF.zip”.	C0001.mp4

- */images*: containing two sub-directories, “fake” containing the Deepfakes images generated, and “real”, where the real images are located and pulled from the repository Celeb-DF [15].
- */videos*: Within this directory, there are two sub-directories called “fake” in which the generated Deepfakes videos are found, and “real”, in which the real videos downloaded from the repository Celeb-DF [15] have been added.

Tables 1 and 2 specify the details of the generated fake files contained in the repository. The first one represents the generated images and the second one the generated videos. Tables 3 and 4 show the real images and videos included in our dataset. In each of the different tables, the nomenclature and content of the repository are described in order to easily identify the origin of the images and videos.

All the tables have the same structure. The first column describes the prefix used to name each image or video file. The second column indicates which tool has been used to generate them or, in the case of real images and videos, the repository where they come from. The third column indicates the number of images generated and some comments about their generation. Finally, there is an example column of the nomenclature used to name each image or video.

4 Conclusions and Further Works

Without the use of computational resources, it is now very difficult or almost impossible to detect the manipulation of multimedia content. Internet has also generated a new trend in which Deepfake contents are already part of our daily

digital lives. This fact has led to the appearance of multiple mobile applications and tools for Deepfake generation, which are available to any consumer of technology. An additional problem is the possibility of immediate sharing and propagation of the fake multimedia content created. The integrity of human beings, the security of a country, or media manipulation to change the intention to vote in elections, among others could be violated.

The detection of these manipulations is absolutely necessary to be able to give a quick response to society and to deny or mitigate those attacks. Therefore, this work has first studied different generation techniques of Deepfakes, in order to raise awareness about the relevance of this research topic. Several conventional and advanced tools for Deepfake detection have been also explored that could be employed to mitigate the Deepfake problem on the Internet. A specific Deepfake dataset has also been created for further validation purposes with these detection techniques.

As future work, it is proposed to generate a more robust dataset with enough content for each type of existing generation technique and, additionally, study of conventional techniques and specific anomaly detection techniques.

Acknowledgements Authors would like to acknowledge the support of the 2022-2023 SUMA-CITeL research project (096-043077), the 2023-2024 LearnIoTOn-Cloud research project (2023-PUNED-0018), the CiberGID UNED innovation group with the CiberScratch 2.0 project, as well as the I4Labs UNED research group with the 2022-2024 In4Labs research project (TED2021-131535B-I00), officially recognized by the Ministry of Science and Innovation. The authors also acknowledge the support of the E-Madrid-CM Network of Excellence (S2018/TCS-4307) from the Madrid Regional Government; and the SNOLA Network of Excellence (RED2018-102725-T) from the Spanish Ministry of Science, Innovation and Universities.

References

1. Avatarify. Website: <https://github.com/alievk/avatarify-desktop> (Date of last access: September 13, 2023)
2. Dall-E2. Website: <https://openai.com/dall-e-2/> (Date of last access: September 13, 2023)
3. DeepFaceLab. Website: <https://github.com/iperov/DeepFaceLab> (Date of last access: September 13, 2023)
4. FaceApp. Website: <https://play.google.com/store/apps/details?id=io.faceapp&gl=ES&pli=1> (Date of last access: September 13, 2023)
5. Forensically. Website: <https://29a.ch/photo-forensics> (Date of last access: September 13, 2023)
6. INCIBE: Deepfakes (*In Spanish*). Website: <https://www.incibe.es/aprendeciberseguridad/deepfakes> (Date of last access: September 13, 2023)
7. MantraNet. Website: <https://github.com/ISICV/ManTraNet> (Date of last access: September 13, 2023)

8. MesoNet. Website: <https://github.com/DariusAf/MesoNet> (Date of last access: September 13, 2023)
9. OBS Studio. Website: <https://obsproject.com/> (Date of last access: September 13, 2023)
10. This Person Does Not Exist. Website: <https://this-person-does-not-exist.com/> (Date of last access: September 13, 2023)
11. Afchar, D., Nozick, V., Yamagishi, J., Echizen, I.: MesoNet: a compact facial video forgery detection network. In: 2018 IEEE International Workshop on Information Forensics and Security (WIFS). IEEE (dec 2018). <https://doi.org/10.1109/wifs.2018.8630761>, <https://doi.org/10.1109/2Fwifs.2018.8630761>
12. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: Stargan: Unified generative adversarial networks for multi-domain image-to-image translation (2017). <https://doi.org/10.48550/ARXIV.1711.09020>, <https://arxiv.org/abs/1711.09020>
13. Gonzalez-Sosa, E., Fierrez, J., Vera-Rodriguez, R., Alonso-Fernandez, F.: Facial soft biometrics for recognition in the wild: Recent works, annotation, and COTS evaluation. IEEE Transactions on Information Forensics and Security **13**(8), 2001–2014 (aug 2018). <https://doi.org/10.1109/tifs.2018.2807791>, <https://doi.org/10.1109/2Ftifs.2018.2807791>
14. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan (2019). <https://doi.org/10.48550/ARXIV.1912.04958>, <https://arxiv.org/abs/1912.04958>
15. Li, Y., Yang, X., Sun, P., Qi, H., Lyu, S.: Celeb-df: A large-scale challenging dataset for deepfake forensics (2019). <https://doi.org/10.48550/ARXIV.1909.12962>, <https://arxiv.org/abs/1909.12962>
16. Rao, Y., Ni, J.: A deep learning approach to detection of splicing and copy-move forgeries in images. In: 2016 IEEE International Workshop on Information Forensics and Security (WIFS). pp. 1–6 (2016). <https://doi.org/10.1109/WIFS.2016.7823911>
17. Sara Ferreiras: Photo and video manipulations detector. Website: <https://github.com/saraferreirascf/Photo-and-video-manipulations-detector> (Date of last access: September 13, 2023)
18. Sergio A. Falcón: Deepfake Repository. Website: https://github.com/oigres5/DeepfakeTFM_UNED (Date of last access: September 13, 2023)
19. Thies, J., Zollhöfer, M., Nießner, M.: Deferred neural rendering: Image synthesis using neural textures (2019). <https://doi.org/10.48550/ARXIV.1904.12356>, <https://arxiv.org/abs/1904.12356>
20. Thies, J., Zollhöfer, M., Stamminger, M., Theobalt, C., Nießner, M.: Face2face: Real-time face capture and reenactment of rgb videos (2020). <https://doi.org/10.48550/ARXIV.2007.14808>, <https://arxiv.org/abs/2007.14808>
21. Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., Ortega-Garcia, J.: Deepfakes and beyond: A survey of face manipulation and fake detection (2020). <https://doi.org/10.48550/ARXIV.2001.00179>, <https://arxiv.org/abs/2001.00179>