UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

Comparison Quality and Performance of Reenactment Deepfakes Models

Gisell Anahis Villarreal Pereira

Ingeniería en Ciencias de la Computación

Trabajo de fin de carrera presentado como requisito
para la obtención del título de
Ingeníera en Ciencias de la Computación

Quito, 23 de diciembre de 2022

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

HOJA DE CALIFICACIÓN
DE TRABAJO DE FIN DE CARRERA

Comparison Quality and Performance of Reenactment Deepfakes Models

Gisell Anahis Villarreal Pereira

Nombre del profesor, Título académico Daniel Riofrío, PhD

Quito, 23 de diciembre de 2022

© DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído todas las Políticas y Manuales de la Universidad San Francisco de Quito USFQ, incluyendo la Política de Propiedad Intelectual USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en la Ley Orgánica de Educación Superior del Ecuador.

Nombres y apellidos: Gisell Anahis Villarreal Pereira

Código: 00139416

Cédula de identidad: 1719812057

Lugar y fecha: Quito, 23 de diciembre de 2022

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en http://bit.ly/COPETheses.

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on http://bit.ly/COPETheses.

ABSTRACT

Deep learning algorithms have advanced due to rapid technological breakthroughs and the dramatic increase in the large-scale storage of public databases. Deepfake content has evolved by improving the techniques used in computer processing vision, natural language, and image detection. Many deepfakes tamper with data and generate fake content, influencing and impacting society positively and negatively. The algorithms behind deepfakes enable a generation of counterfeit images and videos that are hard to distinguish from authentic ones. Such is the case of the reenactment deepfakes, which are used to manipulate facial expressions and poses from source to target. Many consider deepfakes as an underlying cause behind current social issues such as identity theft, threatening democracy and national security through fake news, and financial scams caused by attackers.

This paper is a survey of reenactment deepfake algorithms that are used to create deepfakes with Generative Adversarial Networks (GANs), autoencoders, and transformers. We chose StarGAN and DaGAN models of reenactment deepfakes and compared them in terms of training time, computational complexity, and overall quality and performance.

Key words: Deep learning, Deepfake, Autoencoders, GAN, Complexity Measurements, Facial Expressions, Reenactment.

TABLA DE CONTENIDO

Introducción 10

CATEGORIES OF DEEPFAKES 12

Primary 12

One to one (identity [target] to identity [source]). 12

Many to one (multiple identities [target] to a single identity [source]). 12

Many to many (multiple ids to multiple ids). 12

Secondary 12

Reenactment. 12

Replacement. 13

Editing and synthesis. 13

SOCIAL IMPACTS 13

Negative Impacts 13

Fake News. 14

Pornography. 14

Security. 14

Positive Impacts 14

Education. 14

Entertainment. 14

REENACTMENT DEEPFAKE MODELS 15

Diverse Image Synthesis for Multiple Domains (StarGAN V2) 15

Network architecture. 15

Generator. 15

Mapping network. 16

Style encode. 16

Discriminator. 16

Losses weights. 17

Style reconstruction. 17

Style diversification. 17

Preserving source characteristics. 17

Depth-aware generative adversarial network for talking head video Generation (DaGAN) 19

Network architecture. 20

Face depth network. 20

Keypoint estimator. 20

Occlusion estimator. 20

Feature encoder. 21

Depth encoder. 21

Discriminator. 21

Losses weights. 22

Perceptual Loss. 22

GAN loss. 22

Equivariance loss. 22

Keypoints distance loss. 22

COMPLEXITY IN DEEP LEARNING MODELS 25
Computational complexity 25

EXPERIMENTATION 25
Database 25
Deep learning models 26
Proposed method 26
Experimental setup 26
Data processing. 27
StarGAN V2 27
DaGAN. 27
Training and test sets. 27
StarGAN V2 27
DaGAN. 28
Model configuration. 28
StarGAN V2 28
DaGAN. 29
Assessment metrics. 29

RESULTS AND DISCUSSION 29
Result visualization 30
Complexity evaluation 31

CONCLUSIONS AND FUTURE WORK 32

ACKNOWLEDGMENT 33

Referencias bibliográficas 34

ÍNDICE DE TABLAS

Table 1. Generator Architecture StarGAN V2 17

Table 2. Mapping network architecture StarGAN V2 18

Table 3. Style encoder and discriminator architectures of StarGAN V2.................................18

Figure 2. Second test of CelebA-HQ data set with different ages of StarGAN V2.................18

Table 4. Representation of DaGAN model with the Net and Subnets...................................19

Table 5. Depth and Feature encoder architecture of DaGAN..22

Table 7. Keypoint estimator Architecture of DaGAN 23

Table 8. Occlusion Estimator Architecture of DaGAN 23

Table 9. Face depth network decoder Architecture of DaGAN...23

Table 11. Training hyperparameters of StarGAN V2 model withCelebA-HQ dataset........28

Table 12. Training hyperparameters of StarGAN V2 model withFacial expression recognition dataset..28

Table 13. Trainig hyperparameters of DaGAN model withVoxCeleb1 29

Table 10. Results presentation complexity measurements of FLOPS in model training. 29

ÍNDICE DE FIGURAS

Figure 1. First test: different hairstyle of StarGAN V2 (CelebA-HQ) 18
Figure 2. Second test different ages of StarGAN V2 (CelebA-HQ) 18
Figure 4. Test which translate facial emotion expressions of StarGAN V2. 19
Figure 3. Test which translate style atributes and expressions of StarGAN V2. 19
Figure 8. An example of DaGAN with test of a male source and female driving video. 24
Figure 7. An example of DaGAN with test of a female source and female driving video. 24
Figure 6. An example of DaGAN with test of a female source and male driving video. 24
Figure 5. An example of DaGAN with test of a male source and male driving video. 24
Figure 9. Depth maps with a gray scale of driving image (video) to tamper image source. ...25

INTRODUCCIÓN

The recent phenomenon of deepfakes implicates the manipulation of multimedia content which can now provide a very advanced level of realism. In particular, the advancement in machine learning and the availability of powerful and easy-to-use deep learning tools have increased interest from the research community. Due to the realism of tampered content, deepfakes have become popular among a wide range of developers with varying computer skills, from senior-level to entry-level professionals.

Artificial intelligence applications like deepfakes have both positive and negative impacts on society. Some benefits are in the creative or productive sway in photography, video games, virtual reality, movie production, entertainment, historical education through the reanimation of historical figures, and allowing individuals to virtually try on clothes while shopping, among others. Contrastingly, deepfakes have added a new layer of complexity to the misinformation effect submitted on the internet as part of fake news (civil and political), non-consensual pornography, and overall insecurity (economic, private, and social).

Therefore, society is involved in an environment that blurs the boundary between what is real and what is not. As a result, online users are wrapped in a tide of distrust and uncertainty regarding online information through the multiple discrepancies and disorienting messages that malicious actors introduce into digital media. Consequently, it may yield harmful psychological effects leading to social uncertainty. Hence, many countries and regions have actively carried out the refinement of relevant digital laws and regulations.

In a technical definition, deepfakes are multimedia content generated by an artificial neural network, a branch of machine learning (*The Creation and Detection of Deepfakes: A Survey: ACM Computing Surveys: Vol 54, No 1*, 2022). They use these algorithms to digitally overlap one person’s face and voice onto other people’s videos (Kugler and Pace, 2021). They are composed of target and source identities (we use t to denote the target and s to denote the source). We also represent xs and xt for each image of these identities of videos and xg as the deepfake image generated from s and t. Its first internet appearance took place in 2017 on Reddit forums through computer-generated pornographic videos y that swapped faces of public figures with people in the pornography industry (Kugler and Pace, 2021). Meanwhile, another user developed an app DeepNude, which allows anyone to generate fake nude images of women and other applications used in different contexts like FakeApp, FaceSwap, and ZAO (Masood et al., 2021).

Other open-source projects on GitHub also use publicly available Autoencoders and Generative Adversarial Networks. These allow the user to examine a person's facial expressions and movements to synthesize on faces of other people (Nguyen et al., 2022). Furthermore, popular applications such as Synthesizing Obama allow for synchronized speech with audio recording or text-based editing with a technique called lip-sync (*The Creation and Detection of Deepfakes: A Survey: ACM Computing Surveys: Vol 54, No 1*, 2022) and (Tolosana et al., 2020). Finally, current investigations are looking into creating full-body deepfakes (*The Creation and Detection of Deepfakes: A Survey: ACM Computing Surveys: Vol 54, No 1*, 2022), (Masood et al., 2021) and the generation from a single image (Verdoliva, 2020) or a lower image amount. Thereby, deepfakes have a broad ecosystem categorized into sections and subsections identified in (*The Creation and Detection of Deepfakes: A Survey: ACM Computing Surveys: Vol 54, No 1*, 2022) and (Tolosana et al., 2020).

CATEGORIES OF DEEPFAKES

Mirsky and Lee in (2022) describe that Deepfakes fall into two principal categories: primary and secondary, which are explained as follows.

Primary

This category describes the quantity of source and targets to manage in the deepfake generation.

One to one (identity [target] to identity [source]).

A model that uses a specific identity to drive a specific identity.

Many to one (multiple identities [target] to a single identity [source]).

A model that uses a specific identity to drive a specific identity.

Many to many (multiple ids to multiple ids).

A model that uses any identity to drive any identity.

Secondary

This category is based on a context of human taxonomies to tamper with facial or body representations to obtain fake content.

Reenactment.

It refers to an expression, mouth, gaze, poses, or body of x_t managed by x_s. Hence, facial reenactment represents a pattern of gesture tampering, adjusting the target subject eyes, mouth, nose, forehead, and jaw in video output to reflect those of the source subject.

- Expression

- Mouth
- Pose
- Body

Replacement.

This consists of replacing the face of person with another in a video. Hence, the forgery of x_t by its replacement with the identity of x_s, preserving the s identity.

- Transfer
- Swap

Editing and synthesis.

This works on the attributes of x_t , where they can be added, altered, or removed in subject. For example, this includes changing a target’s clothes, facial attributes, age, weight, beauty, and ethnicity.

- Entire Face Synthesis
- Attribute Manipulation
- Lip-sync

SOCIAL IMPACTS

Negative Impacts

In academic debates, deepfakes trigger two types of harm in society. First, it affects each subject's identity depicted in images or videos and his/her privacy. The second one impacts in mass psychology within a community with misleading information. The following subsections provide a brief explanation and examples of current and potential negative and positive social impacts of the use of deepfakes.

Fake News.

Deepfakes affect individuals' perceptions of truth and a sense of distrust to the information they convey. Finally, this reduces people's trust in news on social media (Vaccari & Chadwick, 2020).

Pornography.

The non-consensual pornography laws has aligned with the deepfakes because it cause harm to the target's reputation and his/her standing in a community. In consequence, the subject suffers a breakdown in emotional well-being due to a damaged perception self-identity identity in society (Harris, 2019).

Security.

In this AI era, national security can be affected by fake satellite images of Earth that contain objects that do not exist to mislead military analysts. For example, creating a fake bridge across a river although there is no such a bridge in reality (Tolosana et al., 2020).

Positive Impacts

Education.

The digital reanimation may aid in film restoration, helping to preserve cultural artistic heritage and historical works. Hence, it let new generations learn about historical characters, or acquire more accurate knowledge of a period of time in history (Tolosana et al., 2020).

Entertainment.

In the multimedia industries, the ease of editing and synthetically replacing actor's dialog or other aspects of their performance would mean cost reductions and time saved in the editing and post-production spheres. Also, foreign films can have realistic video dubbing (Tolosana et al., 2020).

REENACTMENT DEEPFAKE MODELS

The previous classification helps to understand the specific aims at the creation of deepfakes. In particular, each class uses a specific model which characterizes its architecture or mechanisms of learning. For this research, we selected the deepfake reenactment which is distinguished with manipulation of facial expression, mouth, gaze, pose, or body according to aim (Nirkin et al., 2020) In this case, we focused on studying tampering facial expression and pose with deep learning models such as StarGAN V2 and DaGAN.

Diverse Image Synthesis for Multiple Domains (StarGAN V2)

The StarGAN V2 model was designed with four modules that generate diverse images across multiple domains. Here, a domain is a set of images grouped in terms of distinctive features category. Also, each image has a style that renders a unique appearance. Therefore, this model learns the mappings between all available domains using a single generator. The generator also uses domain label as an additional input, for the model learns to transform an image into the corresponding domain (Choi et al., 2019).

Network architecture.

Generator.

The task of the generator is to translate an input image x into an output image $G(x, s)$, which reflects the style code of a specific domain. In fact, this style code supply by a style encoder E, or mapping network F. The style encoder extracts the style code from a given reference image, while the mapping network acquires transforming random Gaussian noise into a style code. Also, generator uses AdaIN (Adaptive Instance Normalization) layers to inject s into G, through learned affine transformations provide scaling and shifting vectors (Choi et al., 2019).

The Style domain represents as s that helps G only deal with synthesizing images of all domains. Table 1 contains information of network architecture.

Mapping network.

The mapping networking F comprised of MLP (Multilayer Perceptron) with multiple output branches to provide style codes for all available domains. So, k represents a number domains and output in a model. In such a way, a number of domains k share four utterly connected layers, and each domain has connected four fully layers. Table 2 contains the architecture details of this network. Also, F produces the diverse style codes by sampling the latent vector z in Z and the domain y in Y at random. In fact, latent code has to sample from the standard Gaussian distribution. The dimensions of the latent code, the hidden layer, and the style code define with 16, 512, and 64, respectively.

Style encode.

The style encoder has a multi-task learning setup as well as F. It consists of CNN with k domains and output branches. It works with an extraction of style code $s = E_y(x)$ of, and $E_y(\cdot)$ is the output of E with the respective domain. Likewise, the E processes different reference images to generate diverse style codes that help G to synthesize style reflecting of reference image x. In definitively, all domains share six pre-activation residual blocks, and each domain has one specific fully connected layer and more information present in Table 3 of network architecture.

Discriminator.

A multi-task discriminator contains multiple linear output branches, and each branch D_y has to train a binary classification. This classification distinguishes a real image of domain

y or a fake image $G(x, s)$ generated by G, whereon the output dimension is one set for real/fake classification. It has the network architecture showed in Table 3.

Losses weights.

Style reconstruction.

The style reconstruction manages with style reconstruction loss that aims to train in a single encoder E to develop diverse output for multiple domains. The encoder E allows generator G training a single encoder E to convert the input image into a reflection style of the reference image.

Style diversification.

The generator has to generate diverse images, so G is regularized with the diversity sensitive loss to find significant features for a generation.

Preserving source characteristics.

The cycle consistency loss helps to maintain the domain's original features in input image x like pose. So, the input image keeps its features while it gets a style.

Layer	Resample	Norm	Output Shape
Image x	-	-	256 × 256 × 3
Conv1×1	-	-	256 × 256 × 64
ResBlk	AvgPool	IN	128 × 128 × 128
ResBlk	AvgPool	IN	64 × 64 × 256
ResBlk	AvgPool	IN	32 × 32 × 512
ResBlk	AvgPool	IN	16 × 16 × 512
ResBlk	-	IN	16 × 16 × 512
ResBlk	-	IN	16 × 16 × 512
ResBlk	-	AdaIN	16 × 16 × 512
ResBlk	-	AdaIN	16 × 16 × 512
ResBlk	Upsample	AdaIN	32 × 32 × 512
ResBlk	Upsample	AdaIN	64 × 64 × 256
ResBlk	Upsample	AdaIN	128 × 128 × 128
ResBlk	Upsample	AdaIN	256 × 256 × 64
Conv1×1	-	-	256 × 256 × 3

Table 1. Generator Architecture StarGAN V2

Type	Layer	Activation	Output Shape
Shared	Latent z	-	16
Shared	Linear	Relu	512
Shared	Linear	Relu	512
Shared	Linear	Relu	512
Shared	Linear	Relu	512
Unshared	Linear	Relu	512
Unshared	Linear	Relu	512
Unshared	Linear	-	512
Unshared	-	-	64

Table 2. Mapping network architecture StarGAN V2

Layer	Resample	Norm	Output Shape
Image x	-	-	256×256×3
Conv1×1	-	-	256 ×256×64
ResBlk	AvgPool	-	128×128×128
ResBlk	AvgPool	-	64×64×256
ResBlk	AvgPool	-	32×32×512
ResBlk	AvgPool	-	16×16×512
ResBlk	AvgPool	-	8×8×512
ResBlk	AvgPool	-	4×4×512
LReLU	-	-	4×4×512
ResBlk	-	-	1×1×512
LReLU	-	-	1×1×512
Reshape	-	-	512
Linear * k	-	-	D*k

Table 3. Style encoder and discriminator architectures of StarGAN V2

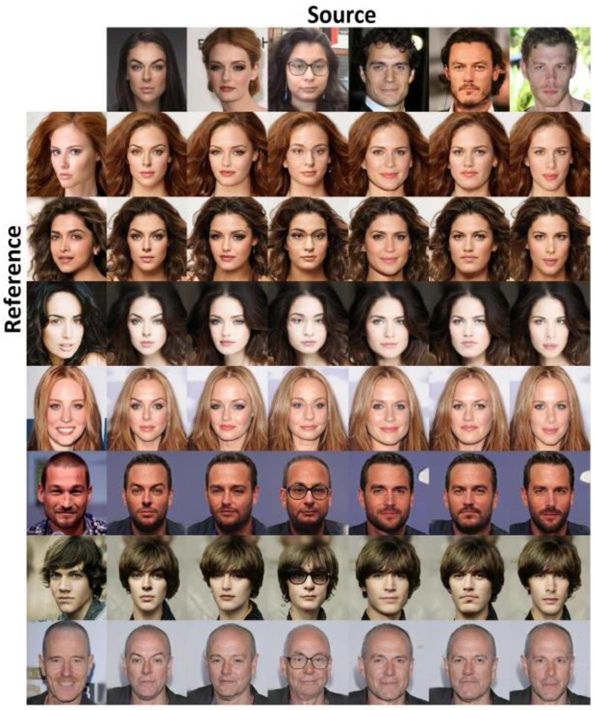

Figure 1. First test: different hairstyle of StarGAN V2 (CelebA-HQ)

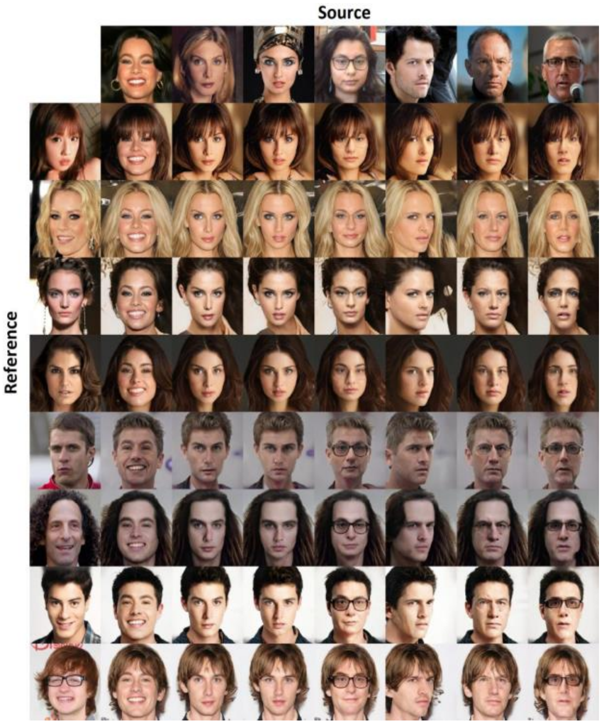

Figure 2. Second test different ages of StarGAN V2 (CelebA-HQ) .

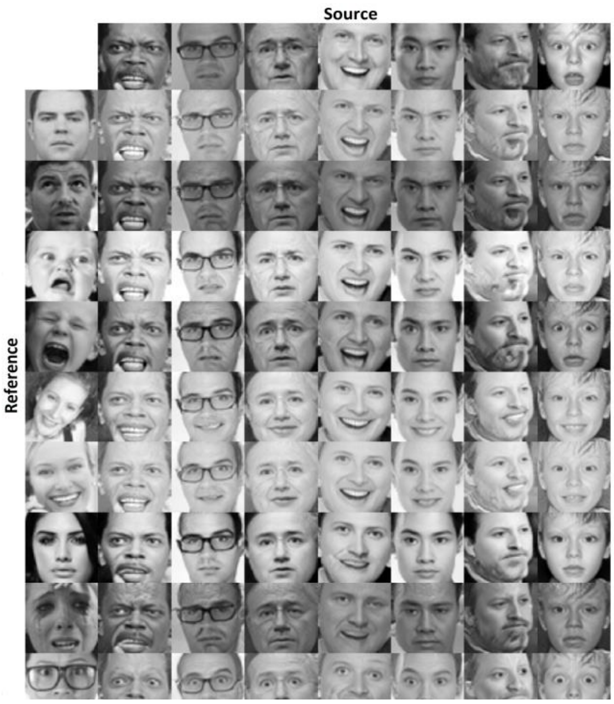

Figure 3. Test which translate style atributes and expressions of StarGAN V2.

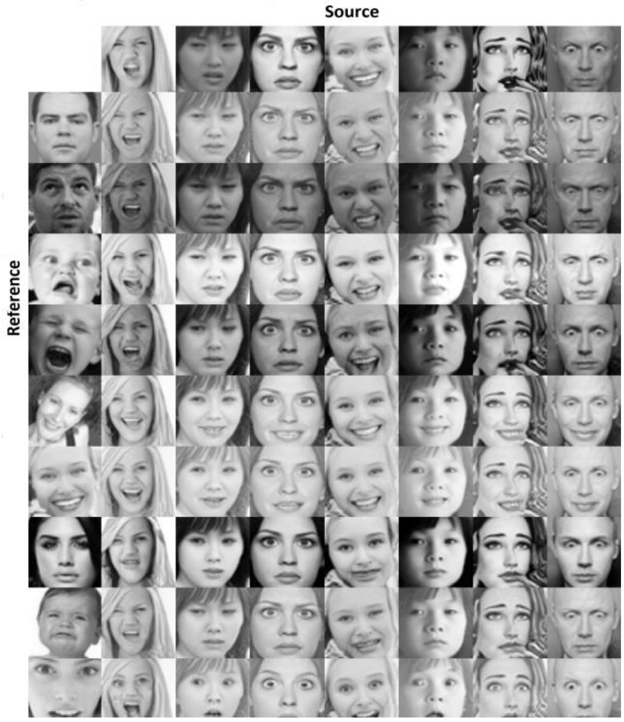

Figure 4. Test which translate facial emotion expressions of StarGAN V2.

Depth-aware generative adversarial network for talking head video Generation (DaGAN)

DaGAN consists of a generator and a discriminator to reach talking head video generation. This model is described in Table 4 which presents a set of neural network modules to produce a synthetic human face video. This is done based on a source image and a driving video that contains the identity and pose information respectively.

Generator		**Discriminator**
* ***Self-Supervised Depth Learning***		***Classification of Real or Fake Images***
Face Depth Network	Encoder	
	Decoder	
* ***Depth-guide Facial Keypoints Detection***		
Face Depth Network	Encoder	
	Decoder	
Keypoints Estimator		
****Cross Modal Attention Mechanism***		
Depth Encoder		
Featured Encoder	Occlusion estimator	

Table 4. Representation of DaGAN model with the Net and Subnets.

Hong et. al. in (Hong et al., 2022) present an approach proposed to leverage DaGAN with:

1. Learning pixel-wise face depth maps in a self-supervised mode to recoup the dense 3D facial geometry from the training face videos.
2. A depth-guided facial keypoints detection to combine both the geometry representations from depth maps with the appearance representations from the images to predict more accurate facial keypoints.
3. Learning of dense depth-aware attention map employing depth maps to constrict the motion field. Thus, the generation of fine-grained details of facial structure and movements are acquired accurately.

Network architecture.

Face depth network.

Training of depth estimation uses twice successive frames from a face video of VoxCeleb1 in a self-supervised manner. So, it allows us to estimate the depth maps of entry face images that are employed in the model's mechanisms for talking head generation. This module comprises an encoder and a decoder. The architecture of this module is detailed in Table 9.

Keypoint estimator.

It does a concatenation of the RGB image and its corresponding depth map issued by ε_d. Thereby, the concatenated appearance and geometry information are used as inputs, toward getting the accurate prediction of sparse key points set in the human face. The architecture of this module is in detailed in Table 7.

Occlusion estimator.

Taking input from the initial warped feature map to predict a motion flow mask M_m and an occlusion map Mo (Hong et al., 2022). The motion flow mask M_m is an outcome of the masked motion field with diverse confidence values of dense 2D motion field estimated. Meanwhile, occlusion map Mo hides regions the feature map of driving video that can not be painted in source image movements. This architecture is presented in Table 8.

Feature encoder.

ε_i extracts the appearance feature map learned from the source image to preserve the highest identity of the source image while maintaining the head motion information between two faces for wrapping. Also, it contains two DownBlocks that preserve the low-level texture image. The architecture of this module is described in Table 5.

Depth encoder.

Taking a source depth map D_s as input to encode a depth feature map F_d that will generate dense guidance for the human face generation.In fact, an output of the dense depth-aware attention map contains 3D guidance geometric, which allows obtain better detail facial structures and micro-movements representations. The architecture of this module is presented in Table 5.

Discriminator.

It performs binary classification of real or fake images with a simple architecture used in FOMM (First order motion model) in (Siarohin et al., n.d.). It collects the intermediate feature maps and feeds them into the GAN loss L_G. Also, it is a single-scale discriminator used for training 256 × 256 images.

Losses weights.

Perceptual Loss.

In fact, an output of the dense depth-aware attention map contains 3D guidance geometric, which allows obtain better detail facial structures and micro-movements representations.

GAN loss.

An output of the dense depth-aware attention map contains 3D guidance geometric, which allows for obtaining better detail of facial micro-movements and face structures. Hence, model can minimize featuring matching loss in the discriminator.

Equivariance loss.

The equivariance loss assures consistency of image-specific keypoints to tamper facial expressions.

Keypoints distance loss.

The keypoints distance loss fits the detected facial keypoints around a small neighborhood with a penalization of distance if keypoints fall out a predefined threshold.

Layer	Resample	Activation	Norm	Output Shape
Conv7x7	-	Relu	BN	64x128x128
Conv3x3	AvgPool	Relu	BN	128x256x256
Conv3x3	AvgPool	Relu	BN	256x512x512

Table 5. Depth and Feature encoder architecture of DaGAN

Layer	Resample	Activation	Norm	Output Shape
Conv4x4	AvgPool	Relu	-	8x1x1
Conv4x4	AvgPool	Relu	BN	64x8x8
Conv4x4	AvgPool	Relu	BN	128x64x64
Conv4x4	-	Relu	BN	256x128x128
Conv1x1	-	-	-	512x256x256

Table 6. Discriminator Architecture of DaGAN.

Layer	**Resample**	**Activation**	**Norm**	**Output Shape**
Conv3x3	AvgPool	Relu	BN	8x64x64
Conv3x3	AvgPool	Relu	BN	64x128x128
Conv3x3	AvgPool	Relu	BN	128x256x256
Conv3x3	AvgPool	Relu	BN	256x512x512
Conv3x3	AvgPool	Relu	BN	512x1024x1024
Conv3x3	Interpo	Relu	BN	1024×512×512
Conv3x3	Interpo	Relu	BN	512×256×256
Conv3x3	Interpo	Relu	BN	256×128×128
Conv3x3	Interpo	Relu	BN	128×64×64
Conv3x3	Interpo	Relu	BN	64×32×32
Conv7x7	-	-	-	1×64×64

Table 7. Keypoint estimator Architecture of DaGAN

Layer	**Resample**	**Activation**	**Norm**	**Output Shape**
Conv3x3	AvgPool	Relu	BN	64x128x128
Conv3x3	AvgPool	Relu	BN	128x256x256
Conv3x3	AvgPool	Relu	BN	256x512x512
Conv3x3	AvgPool	Relu	BN	512x1024x1024
Conv3x3	AvgPool	Relu	BN	512x1024x1024
Conv3x3	Interpo	Relu	BN	512x1024x1024
Conv3x3	Interpo	Relu	BN	1024×512×512
Conv3x3	Interpo	Relu	BN	512×256×256
Conv3x3	Interpo	Relu	BN	256×128×128
Conv3x3	Interpo	Relu	BN	128×64×64
Conv7x7	-	-	-	1×64×64
Conv7x7	-	-	-	1×64×64

Table 8. Occlusion Estimator Architecture of DaGAN

Layer	Resample	Activation	Output Shape
Conv3×3	-	Relu	256
Conv3×3	-	Relu	256
Conv3×3	Interpo	Relu	128
Conv3×3	-	Relu	128
Conv3×3	Interpo	Relu	64
Conv3×3	-	Relu	64
Conv3×3	Interpo	Relu	32
Conv3×3	-	Relu	32
Conv3×3	Interpo	Relu	16
Conv3×3	-	Relu	16
Conv3x3	-	-	1

Table 9. Face depth network decoder Architecture of DaGAN

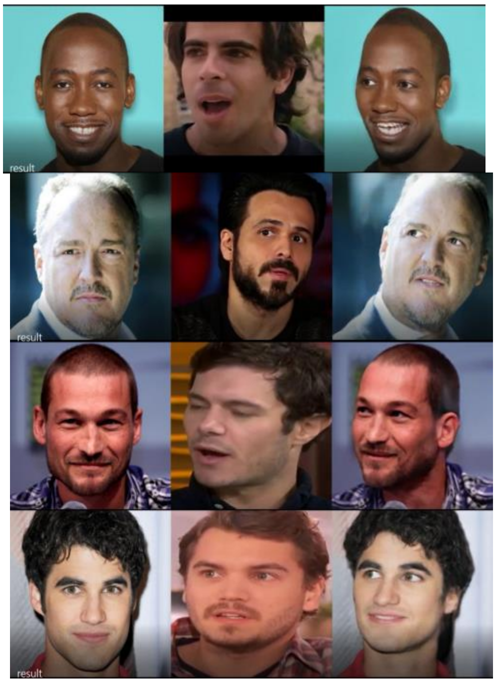

Figure 5. An example of DaGAN with test of a male source and male driving video.

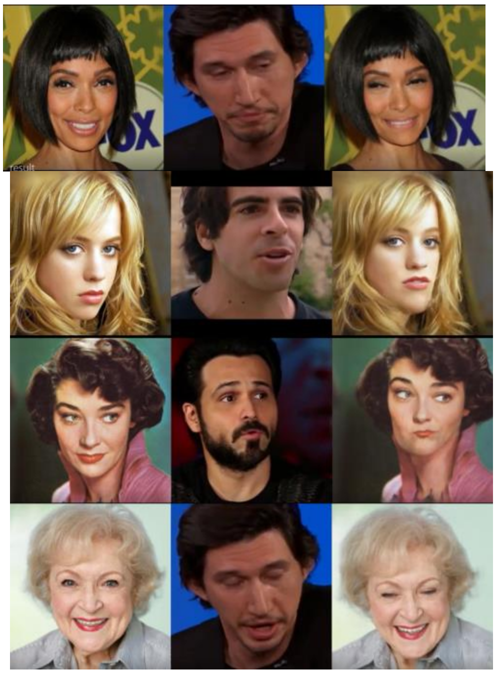

Figure 6. An example of DaGAN with test of a female source and male driving video.

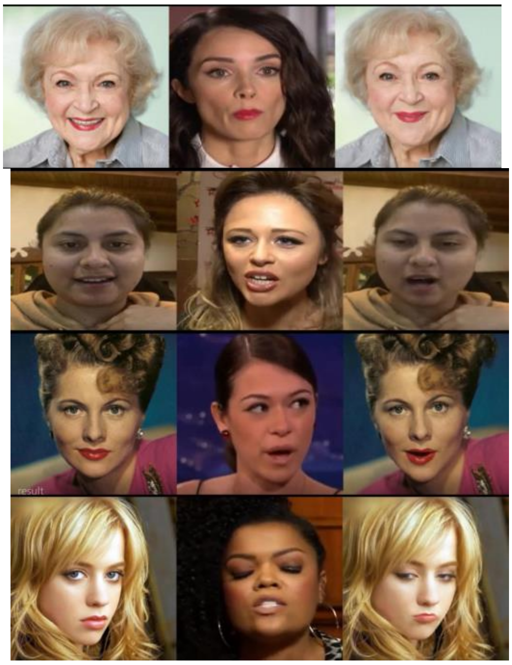

Figure 7. An example of DaGAN with test of a female source and female driving video.

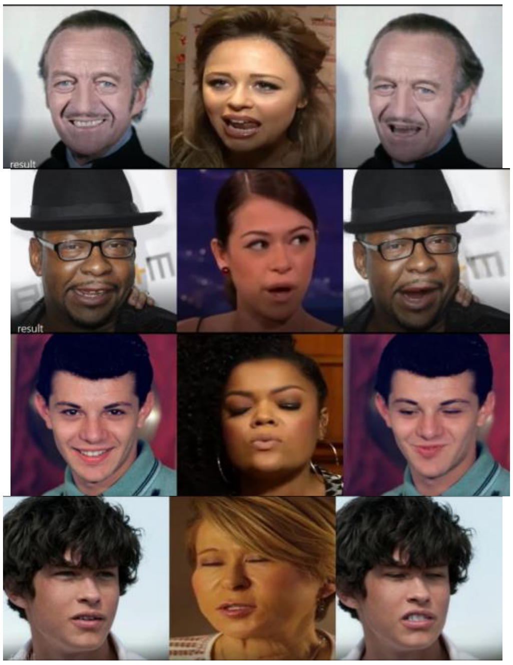

Figure 8. An example of DaGAN with test of a male source and female driving video.

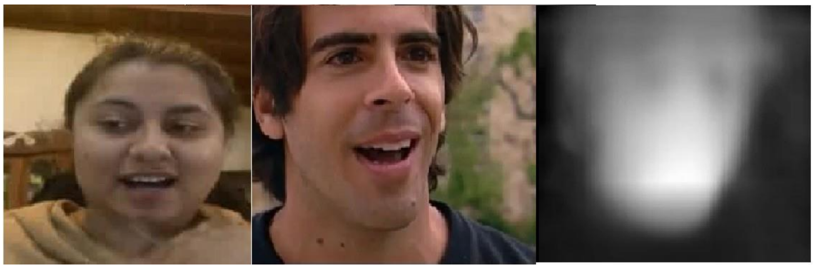

Image src Driving vid. Depth map

Figure 9. Depth maps with a gray scale of driving image (video) to tamper image source.

COMPLEXITY IN DEEP LEARNING MODELS

Computational complexity

It is measured with floating point operations (FLOPs) to determine model performance (Zhang et al, 2021). The counting of FLOPs is relied in measure in terms of the amount of additions, subtractions, and multiplications which allows do a fair comparisons among training models. Also, FLOPs are significant to understand a performance in a chip or devices using the model.

EXPERIMENTATION

Database

The dataset used for StarGAN V2 experimentation is CelebA-HQ. This dataset has a set of images with different resolutions, portrait images, and crowds of several people (Karras et al., 2017). A data processing phase ensures consistent high quality, and that the images are centered on the facial region. The dataset has images of 1024 x1024 with high-quality resolution. This data set is divided into two domains of male and female.

For StarGAN V2, one can use a data set with face expressions (J. Oheix , 2018) with seven domains such as angry, happy, disgust, fear, neutral, sad, and surprise. This dataset has a size of 48 x48 to focus on facial expressions in gray scale.

The DaGAN uses a VoxCeleb1 dataset which has a database of YouTube videos. This database has a great number of expressions of 1,251 celebrities. These videos involve interviews of a wide variety of ethnicities, gender, and ages. Moreover, these interviews evolve in several environments like outdoors, professional expositions, and on the red carpet. These videos are resized to 256 x256 for training.

Deep learning models

We use generative adversarial networks to achieve image-to-image translations and 3D object generation for the experiments. These models are explained in the section Reenactment Deepfake Models in a subsection of Network Architecture.

Proposed method

The proposed experiment has been done with the StarGAN V2 and DaGAN models to learn about their architectures. These architectures are composed by learning mechanisms that allow getting map features for images or sequence images. Indeed, feature maps help outline style and appearance representations of the source, which will generate fake images or videos. From this, we proceeded to train our models to obtain a field with checkpoints that afford to compile some tests, and we could define their result performance. Also, we decided to measure the complexity of both models to determine differences in their training performance.

Hence, we used the library DeepSpeed for PyTorch, which helps us get these measurements into the training workflow. In order to use this library, we imported FlopsProfiler, which shows a model profile after training with functions such as start, stop, and end with steps assigned to the profile.

Experimental setup

The training experiment is implemented on an NVidia DGX workstation with 4 GPU-NVIDIA Tesla V100-DGXS-32GB with 32 GB per GPU (128 GB total) of GPU memory.

Data processing.

StarGAN V2.

Therefore, images stand processed with the removal and artifacts of JPEG and super-high resolution to achieve a high visual quality. Also, the image is extended through mirror padding and Gaussian filtering to produce a visually satisfactory depth-of-field effect. Finally, facial landmark positions help make an ideal crop to find a high-quality resampling with a final image-specific resolution. So, we reproduce this data processing to improve our performance results. In the case of Face expression recognition, a dataset with a gray scale with seven domains use the same preprocessing to obtain a zoom face image. So, after this processing, we resized images to 48 x 48 with only faces to get an expression as styles to tamper in our sources.

DaGAN.

For the preprocessing of the VoxCeleb data set containing 22,496 videos extracted from YouTube, an initial bounding box is obtained from the first frame of the video. The initial bounding box helps us to track the face until it is far enough. Afterward, the video frame cuts down into a minimal crop containing all bounding boxes that repeat until the last sequence. Also, the dataset excludes the lower-resolution videos, and the rest are resized to 256 x 256 (Siarohin et al., n.d.).

Training and test sets.

StarGAN V2.

The StarGAN V2 dataset has 28,000 images of celebrities with 17,943 female images and 10,057 male images for training. The evaluation dataset has 1,000 images in each domain. The test dataset has source and reference folders that contain images of females and males. In the case of Face expression recognition, the gray scale image dataset has 4,000 images for seven domains and validations for 1,000 images each one.

DaGAN.

Over the preprocessing, we acquired 12,331 videos for training, while the test has 44 videos with frame lengths from 64 to 1,024. Further, we create a test set through sampling 2,083 image sets of 100 videos selected at random for testing.

Model configuration.

StarGAN V2.

The configuration of the hyperparameter is found in tables: Table 11 and 12.

Hyperparameters	**StarGAN V2**	
Learning rate	D, E and G	10^{-4}
	F	10^{-6}
Adam Optimizer	β_1	0
	β_2	0.99
Batch size	8	
R1 regularization	1	
Cyclic Consistency Loss	1	
Style Reconstruction Loss	1	
Diversity Sensitive Loss	1	

Table 11. Training hyperparameters of StarGAN V2 model withCelebA-HQ dataset

Hyperparameters	**StarGAN V2**	
Learning rate	D, E and G	10^{-4}
	F	10^{-6}
Adam Optimizer	β_1	0
	β_2	0.99
Batch size	8	
R1 regularization	1	
Cyclic Consistency Loss	3	
Style Reconstruction Loss	3	
Diversity Sensitive Loss	3	

Table 12. Training hyperparameters of StarGAN V2 model withFacial expression recognition dataset

DaGAN.

The hyperparameter configuration is shown in tables: Table 13.

Hyperparameters	**DaGAN**	
Batch size	8	
Learning rate	Generator	2^{-6}
	Discriminator	2^{-4}
	KP Detector	2^{-4}
Loss weights	Generator	1
	Discriminator	1
	Kp distance	10
	Equivariance	10
Transform parameters	σ_{affine}	0.05
	σ_{tps}	0.005

Table 13. Trainig hyperparameters of DaGAN model withVoxCeleb1

Assessment metrics.

In the collection of evaluation metrics in the library DeepSpeed of FlopsProfiler class, we have used the following functions that return a profile model numbers of floating-point operations (FLOPs). These measurements are collected during training. We collect latency, flops, and the number of parameters by model.

Measures	**StarGAN V2**	**DaGAN**
fwd flops per GPU:	2902834.73 G	103490221.33 G
fwd latency	1.95 s	5.89 s
bwd latency	5.24 s	10.02 s
fwd FLOPS per GPU	1488.63 TFLOPS	1757049.59 TFLOPS
bwd FLOPS per GPU	1107.95218 TFLOPS	2065673.08 TFLOPS
fwd+bwd FLOPS per GPU	1211.19668 TFLOPS	1951418.37831 TFLOPS

Table 10. Results presentation complexity measurements of FLOPS in model training.

RESULTS AND DISCUSSION

Throughout this project, we found out different types of deepfakes that are used with diverse purposes. However, some of them are used to inflict damage on others with fake content. We found different forms to create deepfakes, as with some GitHub projects and apps. Thus, deepfakes creation is at reach by any person who has an application, and developer skills who can improve a model to generate better fake content. Also, many datasets of deepfakes are extracted from social media that present as stealing content with a no-consensual owner. Moreover, an attacker can not be found because this fake content is uploaded anonymously. Furthermore, to enrich the discussion of available content for deepfake creation, in this project, we analyzed tow models measured under different metrics during training.

Result visualization

For an analysis of the results of the StarGAN V2 model, we have to emphasize the improvement in the style encoder and mapping network. Because these modules extract a style code for the generator which focuses to use style. So, these styles are rendered in the image sources. Also, figure 1 and figure 2 allow us to check a successful style capture of references to translate into the source. We determine the model can synthesize the identity of the origin in various appearances that reflect the styles of the reference images as hairstyle, makeup, beard, and age. Also, figure 1 shows that the poses and expressions translate in a reasonable manner in the source images from the reference ones. In figure 1 and figure 2 , we have to know that each row is an identity that transfers into different styles, and the relationship of columns is a style that reflects different identities. Afterward, it determines that the styles extracted from the reference images are represented in the original images correctly.

Besides, in the Face expression recognition dataset, we find in figure 3 and figure 4 focuses on facial expressions. Because we try to prove to transfer facial expressions on source images as a style in learning. Therefore, we can determine whether this model can be used to create a reenactment deepfake with image content. Thereby, the model presents rich styles across multiple domains, which generate remarkably outperforming images from the previous methods.

On model training, the 3D dense face geometry recovers as depth maps in figure 9 which reveal a proper generation of talking heads. Indeed, depth maps enable to obtain of facial keypoints estimated to reflect a face structure that produces motion fields for feature warping into source images from driving video. The examples figure 5, figure 6, and figure 7 tested allow understanding with results depth map functionality. Based on a gray scale that relates to the surface distances (closer is lighter or deep is darker). For these reasons, we can look a reflected flow motion fields in warping source images in time by driving video. Therfore, the DaGAN model, we look at an acceptable visual facial expression movements performance in figures 5, 6, 7, and 8. Therefore, the results show us how some driving videos can give proper visual performance through video. However, other videos are harmed with a different form of the head person on video driving, opposite face position, and space of source image to move as Figure 6 in image source 3. We conclude that this model can create reenactment deepfakes, but it has to improve in the keypoints detection. Thereby, keypoints generate adaptive movements for the source image through time.

In the analysis of models, we determined that both employ extraction feature maps to tamper facial expression and pose. Also, both models could be used to do reenactment deepfakes which shows some real results with their training. Moreover, a chosen dataset training allows us to generate specific results. For example, in DaGAN we have a dataset of interviews, hence in terms of the expressions there are few or none samples of sadness, fear, or surprise.

Complexity evaluation

The complexity measurements gained in both training models submit the prime source of complexity that stems from the operations required per sample. For this purpose, our work uses a running complexity that shows results in table 10 of both models. The comparison of both models allows us to determine the model StarGAN V2 has better performance than DaGAN because StarGAN V2 needs a lower amount of FLOPs to train the model. Furthermore, in (Nicosia et. Al, 2021) it is explained that the amount higher of FLOPs indicates lower performance and slower running. Therefore, the results show a StarGAN V2 is a less complex model than DaGAN.

The complexity is determined by its architecture, inputs, and output data by a model in a given device. Moreover, in the same reference (see (Nicosia et. Al, 2021) for more detail), it is explained that convolution layers are one the most computationally intensive layers in Neural Networks. Hence, StarGAN V2 utilizes an image database with a 256 x 256 size, for it trains with four modules learning to transfer styles. The modules have a lower quantity of convolutional layers that require lower floating point operations to generate a fake image. Instead, DaGAN is a model with video data inputs that train as an image frame of video. Hence, processing a video dataset requires a higher complexity during learning.

CONCLUSIONS AND FUTURE WORK

In conclusion, the research on deepfakes allowed us to learn about their types based on the features tampered in a target. Accordingly, deepfake creation depends on the training model and its data set, because both shape the learning of the model to get the target results. Therefore, in the case of StarGAN V2, it contains an image dataset that focuses on characteristics such as hair, beard, and age. This model produced results based on physical features. While the other data set has facial expressions that allow us to transfer facial expressions in source images. Instead, in the DaGAN, we found that a dataset of interview videos prevents reenactment with extreme expressions of astonishment, fear, and sadness. Further, deepfakes also need extraction of intermediate representations that allows for capturing feature representation for their manipulation based on a facial or body taxonomy.

Measuring complexity could be understood as the total time spent to finish a specific set of algorithms in a given device (Zhou et. al, 2022). Hence, the deep learning algorithms analyzed show us the computational quantity required of a model to work with its input features and output per layer. Also, the FLOPS is a measure of computer performance that allows one to obtain a speed unit to measure the instructions per second.

In a review of the study, we found other focal points to enhance the models tested. Hence, we could try new datasets with DaGAN that allows us to make extreme expressions such as anger, sadness, fear, and surprise. While in the case of StarGAN V2, we can also improve its results by training it with a physical pose, and expression features to get better reenactment deepfakes.

ACKNOWLEDGMENT

Authors thank the Applied Signal Processing and Machine Learning Research Group of USFQ for providing the computing infrastructure (NVidia DGX workstation) to implement and execute the developed source code.

REFERENCIAS BIBLIOGRÁFICAS

A. S. Tehrani, H. Cao, S. Afsardoost, T. Eriksson, M. Isaksson, and C. Fager. (2010). *A comparative analysis of the complexity/accuracy tradeoff in power amplifier behavioral models*. IEEE Transactions on Microwave Theory and Techniques, vol. 58, no. 6, pp. 1510–1520.

A. Siarohin, S. Lathuili`ere, S. Tulyakov, E. Ricci, and N. Sebe. (2019). *First order motion model for image animation. Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alch´e-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/31c0b36aef265d9221af80872ceb62f9-Paper.pdf

C. Vaccari and A. Chadwick. (2020). Deepfakes and disinformation: Exploring the impact of synthetic political video on deception, uncertainty, and trust in news.

D. Harris. (2019). Here and the Law Cannot Protect You. Deepfakes: False pornography is here and the law cannot protect you. Duke Law Technology Review, pp. 99–127.

D. Justus, J. Brennan, S. Bonner, and A. S. McGough. (2018). *Predicting the computational cost of deep learning models*. IEEE International Conference on Big Data (Big Data), 2018, pp. 3873–3882.

F.-T. Hong, L. Zhang, L. Shen, and D. Xu. (2022). Depth-aware generative adversarial network for talking head video generation.

G. Nicosia, V. Ojha, E. L. Malfa, G. L. Malfa, G. Jansen, P. M. Pardalos, G. Giuffrida, and R. Umeton. (2021). *Machine Learning, Optimization, and Data Science*. Springer Nature.

J. Oheix. (2018). *Face expression recognition dataset*. [Online]. Available: https://www.kaggle.com/datasets/jonathanoheix/face-expression-recognition-dataset?resource

L. Verdoliva, *Media forensics and deepfakes: an overview*. (2020). [Online]. Available: https://arxiv.org/abs/2001.06564

M. B. Kugler and C. Pace. (2021). *Deepfake privacy: Attitudes and regulation*. SSRN Electronic Journal.

M. Masood, M. Nawaz, K. M. Malik, A. Javed, and A. Irtaza. (2021). *Deepfakes generation and detection: State-of-the-art, open challenges, countermeasures, and way forward*. [Online]. Available: https://arxiv.org/abs/2103.00484

R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega- Garcia. (2020). Deepfakes and beyond: A survey of face manipulation and fake detection. [Online]. Available: https://arxiv.org/abs/2001.00179.

S. Bianco, R. Cadene, L. Celona, and P. Napoletano. (2018). *Benchmark analysis of representative deep neural network architectures.* IEEE Access, vol. 6, pp. 64 270–64 277.

T. T. Nguyen, Q. V. H. Nguyen, D. T. Nguyen, D. T. Nguyen, T. Huynh-The, S. Nahavandi, T. T. Nguyen, Q.-V. Pham, and C. M. Nguyen. (2019). *Deep learning for deepfakes creation and detection: A survey*. [Online]. Available: https://arxiv.org/abs/1909.11573

T. Karras, T. Aila, S. Laine, and J. Lehtinen. (2017). *Progressive growing of gans for improved quality, stability, and variation*. [Online]. Available: https://arxiv.org/abs/1710.10196

X. Zhou, H. Liu, C. Shi, and J. Liu. (2022). Deep Learning on Edge Computing Devices: Design Challenges of Algorithm and Architecture. Elsevier.

Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha. (2020). Stargan v2: Diverse image synthesis for multiple domains.

Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo. (2018). *Stargan: Unified generative adversarial networks for multi-domain image-to- image translation*. [Online]. Available: https://arxiv.org/abs/1711.09020

Y. Mirsky and W. Lee. (2022). *The creation and detection of deepfakes*. ACM Computing Surveys, vol. 54, no. 1, pp. 1–41, [Online]. Available: https://doi.org/10.1145%2F3425780

Y. Nirkin, L. Wolf, Y. Keller, and T. Hassner. (2020). *Deepfake detection based on the discrepancy between the face and its context.* [Online]. Available: https://arxiv.org/abs/2008.12262

Y. Zhang, J. Wang, J. Sun, B. Adebisi, H. Gacanin, G. Gui, and F. Adachi. (2021). *Cv-3dcnn: Complex-valued deep learning for csi prediction in fdd massive mimo systems.* IEEE Wireless Communications Letters, vol. 10, no. 2, pp. 266–270.