Enhancing the concept of Generative Art

Bhushan Mahajan
Department of Computer Engineering
AISSMS College Of Engineering
Pune, India
bhushanlmahajan@gmail.com

Deven Waykar
Department of Computer Engineering
AISSMS College Of Engineering
Pune, India
devenwaykar@gmail.com

Aditya Kadam
Department of Computer Engineering
AISSMS College Of Engineering
Pune, India
adityakadam181@gmail.com

Vaishnavi Bhagde
Department of Computer Engineering
AISSMS College Of Engineering
Pune, India
bhagde.vaishnavi01@gmail.com

S.R Nalamwar
Prof. at Department of Computer Engineering
AISSMS College Of Engineering
Pune, India
srnalamwar@aissmscoe.com

Abstract- In this research, we have developed a Graphical User Interface (GUI) that lists various applications based on the concept of generative art. Our GUI includes applications like random password generation, abstract artwork generation, map generation, and wallpaper generation.

The password generator application generates random passwords derived from images created using image generation algorithms and scripts such as EPWT and Python's in-built random function that utilizes algorithms like mersenne twister, a pseudo-random number generator. The map generation application provides a layout of an area with requested coordinates and different constraints. The generator employs a model that generates maps using a computational approach. Lastly, the wallpaper generation application generates aesthetically pleasing wallpapers with adjustable features. The application employs a model that utilizes generative art techniques to create wallpapers. In conclusion, our research presents a GUI with different applications based on generative art. These applications provide unique and innovative solutions for different domains such as cybersecurity, art, design, and mapping.

Keywords—generative art, abstract art, python, generative adversarial network, hashing algorithms

I. INTRODUCTION

Generative Art (GA) pertains to artwork that has been produced, either entirely or partially, through the utilization of an automated system. Such a system is generally non-human and capable of independently making decisions about the characteristics of an artwork that would otherwise require the artist's direct input. While algorithmic art, which is art generated by computer algorithms, is often synonymous with "generative art," it can also be created using chemical, biological, mechanical, or robotic systems, as well as through the use of smart materials, randomization, mathematics, data mapping, symmetry, tiling, and other means. In some cases, the human artist may assert that the generative system embodies their own artistic concept, while in others, the system may take on the role of the creator.

Graphical user interfaces and computer code have developed into separate art forms for some designers. Generative art (GA) involves generating new ideas, forms, shapes, colors, or patterns through a set of algorithmic procedures. The first step is to establish a set of rules that define the creative boundaries for the generation process. Subsequently, a computer executes those rules to generate new works on behalf of the artist.

Generative artists utilize the capabilities of modern computing power to create new aesthetics. They direct programs to operate within a set of artistic constraints and guide the process towards a desired outcome. By doing so, they leverage the potential of generative art to produce innovative and diverse artworks.

Generating abstract artwork which is our second model involves generating artwork through a set of algorithmic procedures. This method allows artists to create artwork that is both diverse and complex, while also providing a new way of thinking about art and creativity. This research paper explores the potential of generative art for creating abstract artwork, utilizing the capabilities of modern computing power and machine learning techniques to produce innovative and visually striking pieces of art.

In the realm of generative art, one application of algorithmic generation is password generation. As traditional password-hashing algorithms are becoming increasingly vulnerable to attacks, there is a need for more secure methods of password generation. To address this, generative art algorithms can be employed to create strong and unpredictable passwords. By using image generation algorithms and scripts such as EPWT and Python's built-in random function, random and complex images can be generated to derive passwords that are difficult to guess or brute-force. The research paper highlights the practical application of generative art algorithms in password generation and demonstrates the potential of this approach to enhance the security and reliability of password-hashing algorithms.

II. RELATED WORK/ LITERATURE REVIEW

A. What is generative art? by Margret a. Boden, Ernest A. Edmonds[9]: The authors contend that generative art is different from other art forms because it prioritizes the process over the final product and allows for unexpected outcomes. The authors also note that generative art relies on algorithms, chance, and the artist's manipulation of generative processes. Through their examination, Boden and Edmonds emphasize the novel and creative potential of generative art as an innovative form of artistic expression.

B. Generative Art - Tricks and Traps by Grzegorz Kępisty[7]: The field of generative art is growing quickly, utilizing algorithms and computational processes to create visual works. Although it is becoming more popular among artists and designers, there are several potential challenges to creating this kind of art. One such issue is becoming too reliant on the algorithm, resulting in formulaic artwork. Another difficulty is balancing control and unpredictability, as generative systems can produce unexpected results. Additionally, the reliance on technology may prevent some from accessing this field. Despite these challenges, generative art presents unique opportunities for experimentation and exploration

C. A Framework for understanding generative art by Alan Dorin, Gordon Monro, Jon McCormack[8]: To better understand and compare generative artworks, a framework is required that can describe, analyze, and compare them. Existing frameworks, such as those used in kinetic art, are insufficient. Therefore, a new framework is proposed, consisting of four major components: the entities involved in the artwork, the processes used to generate it and their interactions with the environment, and the outcomes experienced by the audience. This framework is flexible enough to accommodate various types of generative systems, including computational, physical, kinetic, and virtual systems, enabling meaningful comparisons across time and space. However, one limitation of this approach is its focus on the technical features of the generative process, rather than the artistic motivations behind the work.

D. A Region Based Easy Path Wavelet Transform for Sparse Image Representation by Renato Budinich[10]: This approach allowed us to achieve high compression ratios while preserving the overall structure and important features of the image. The use of image segmentation helps to identify and preserve important regions of the image while discarding unnecessary details, which can be represented by smaller coefficients in the EPWT-like transform. The deterministic path-finding procedures ensure that the segmentation is consistent and reproducible, which is important for maintaining the integrity of the compressed image. Overall, our method offers a promising approach for efficient compression of natural images while minimizing the loss of important information.

E. Diverse Image Generation via Self-Conditioned GANs by Steven Liu, Tongzhou Wang, David Bau, Jun-Yan Zhu, Antonio Torralba[11]:
When a conditional GAN is trained with clustering labels derived from discriminator features, it is effective at reducing mode collapse, outperforming several previous approaches. We observe that the method continues to perform well when the number of synthesized labels exceeds the number of modes in the data. Furthermore, our method scales well to large-scale datasets, improving Fréchet Inception Distance and Inception Score measures on ImageNet and Places365 generation, and generating images that are qualitatively more diverse than an unconditional GAN.

F. Generative Image Inpainting Based on Wavelet Transform Attention Model by Chen Wang, Jin Wang[12]: Proposed an algorithm that uses wavelet transforms and attention mechanisms to fill in missing regions of an image. The proposed algorithm outperforms several state-of-the-art inpainting methods in terms of both visual quality and quantitative metrics.

G. Tile Art Image Generation Using Conditional Generative Adversarial Networks By Naoki Matsumura, Hiroki Tokura[1]: Proposed a method for generating tile art images using conditional generative adversarial networks. The proposed method generates high-quality tile art images that closely resemble the desired patterns and is robust to variations in the conditioning vector.

H. AI Illustrator[14]: Art Illustration Generation Based on Generative Adversarial

Network" proposes a system called AI Illustrator that generates new, original art illustrations that are similar in style to a given input illustration. The proposed system uses a modified GAN architecture that incorporates a self-attention mechanism and a multi-scale discriminator network to improve the quality of the generated illustrations.

I. "DaRt: Generative Art using Dimensionality Reduction Algorithms"[4]: Proposed a system that generates art using dimensionality reduction algorithms. The system, called DaRt, uses techniques such as principal component analysis (PCA) and t-distributed stochastic neighbor embedding (t-SNE) to reduce the dimensionality of image datasets and generate new, original art pieces. The authors demonstrate the effectiveness of DaRt on several datasets and show that it can generate diverse and visually appealing art pieces. Overall, DaRt provides a novel approach to generative art that leverages dimensionality reduction techniques to generate new and creative visual content.

J. A Style-Based Generator Architecture for Generative Adversarial Networks by Nvidia authors T. Karras, S. Laine, and T. Aila[13]: The StyleGAN paper introduces a new type of generative model for image synthesis called StyleGAN, which is based on a novel architecture that generates high-quality, diverse, and realistic images with fine details, realistic textures, and high resolution. The architecture is composed of a generator network and a mapping network, which together allow the generation of images with control over the style and appearance of the output. The paper also introduces several additional techniques such as adaptive instance normalization, progressive growing, and style mixing, which contribute to the quality and diversity of the generated images. StyleGAN has achieved state-of-the-art results on several benchmark datasets and has inspired a wide range of follow-up work and applications in the field of generative modeling.

III. METHODOLOGY

A. Understanding Generative Art

Generative art is an art form that utilizes an algorithm or a programmed set of rules to create art. The process of creating generative art is driven by creating a system that can independently produce artwork, where the artist plays a role as a facilitator or collaborator, rather than the only creator. To comprehend generative art, a framework has been provided below that can be used as a reference.

Algorithm refers to a set of rules or instructions that a computer program follows to perform a specific task. In generative art, an algorithm is used to create rules that define how the artwork is generated. The artist must decide on the rules and how they will be implemented.

Data input is a necessary component in generative art, as the algorithm requires some form of data input to generate artwork. This input can take various forms such as random numbers, images, text, or sound. The data input serves as the raw material that the algorithm uses to generate the artwork.

The generative process is the core of generative art, where the algorithm uses the data input to create a set of rules that dictate how the artwork is created. The generative process can either be deterministic, meaning that the rules always produce the same output given the same input, or non-deterministic, meaning that the output can vary even with the same input.

Feedback loops are often used in generative art to modify the algorithm or data input to achieve the desired result. The feedback loop allows the artist to refine the artwork or create entirely new variations.

The output of generative art can take many forms, such as static images, interactive installations, or sound pieces. The output is generated by the algorithm and data input and can be unpredictable or unexpected, depending on the nature of the generative process. [2]

The final output of generative art is open to interpretation by the viewer. While the artist may have some intention behind the artwork, the generative process often produces unexpected results that can be interpreted in many different ways. The viewer brings their own experiences and perspectives to the artwork, creating a unique interpretation of the generative art piece. [2]

B. Image training using GAN:

Generative Adversarial Networks (GANs) are a class of machine learning models that are designed to generate new, synthetic data that is similar to the training data. The GAN model consists of two main components: a generator and a discriminator. The generator is responsible for generating new data samples, while the discriminator is responsible for distinguishing between real and synthetic data samples. During the training process, the generator tries to generate data that is similar to the training data, while the discriminator tries to distinguish between the real and synthetic data. This process continues until the generator produces data that is indistinguishable from the training data, at which point the training is considered to be successful. [3] The GAN model works by training a generator to produce synthetic data that is similar to the training data, and a discriminator to distinguish between real and synthetic data. The training process involves optimizing both the generator and discriminator models simultaneously, using a loss function that measures the quality of the generated data. The generator generates data samples by drawing random noise from a latent space, and then mapping that noise to the output space using a neural network. The discriminator then receives both real and synthetic data samples, and tries to distinguish between them. As the training process continues, the generator learns to produce more realistic data samples, while the discriminator learns to better distinguish between real and synthetic data. The ultimate goal of the training process is to generate synthetic data that is indistinguishable from the training data, thereby producing a model that can generate new, high-quality data samples.

One particular application of GANs is in handling imbalanced datasets. In such datasets, one or more classes may be underrepresented, which can lead to machine learning models producing biased or inaccurate results. By training a GAN on the imbalanced dataset, the generator can learn to produce synthetic data for the underrepresented class, which can then be combined with the original dataset to create a more balanced dataset. This approach can help improve the accuracy and generalizability of machine learning models trained on imbalanced datasets.

StyleGAN is a type of GAN that extends the traditional GAN model by incorporating a style-based generator architecture. This architecture is designed to control the style of the generated images by separating the latent space into two components: a style vector and a content vector. The style vector controls the high-level features of the image, such as the pose and the lighting, while the content vector controls the low-level features, such as the texture and the shape. By controlling these separate components, the StyleGAN model is able to generate highly realistic images that exhibit both diversity and consistency in style. Additionally, the StyleGAN model has been shown to be highly versatile and can be trained on a wide range of datasets, making it a valuable tool for a variety of image generation tasks. [13]

We have trained a StyleGAN neural network using a carefully curated dataset consisting of approximately 1000 abstract images. The training process involved feeding this dataset into the StyleGAN network, which was then able to learn the underlying patterns and features of the abstract art images. Through this process, the network developed an understanding of how to generate new abstract images that resemble those in the training dataset. After extensive training, we have observed that the StyleGAN network has produced some remarkable results. The generated abstract images are visually stunning and show a remarkable resemblance to the original training dataset. The network has been successful in capturing the unique features and intricate details of each image in the dataset, allowing it to produce new images that are both aesthetically pleasing and artistically meaningful. Overall, our efforts in training the StyleGAN network on this abstract image dataset have yielded some impressive results. We are excited about the potential applications of this technology in the field of art and design, and we believe that it has the potential to revolutionize the way that we create and appreciate abstract art. [5]

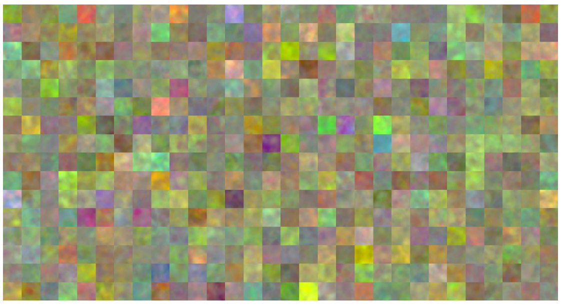


Fig. 1. Images generated at tick 2

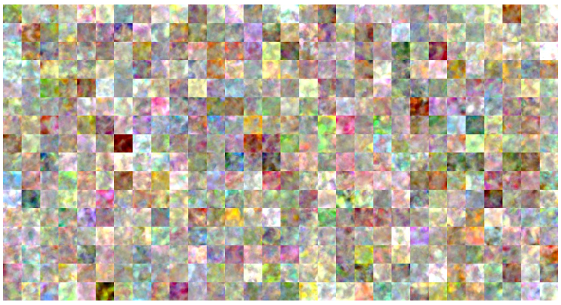


Fig. 2. Images generated at tick 12

Fig. 3. Images generated at tick 23

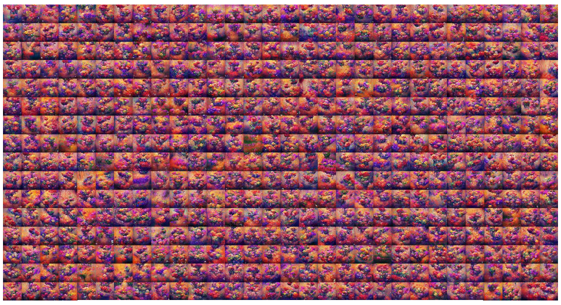


Fig. 4. Images generated at tick 70

C. Algorithms used for password generation:

1) Mersenne Twister

Mersenne Twister is a pseudorandom number generator (PRNG). It is a type of random number generator that is based on a mathematical algorithm. This algorithm is designed to generate a sequence of numbers that appear to be random, but are actually generated by a deterministic process. The Mersenne Twister algorithm was developed by Makoto Matsumoto and Takuji Nishimura in 1997. It is a far more complex algorithm than earlier PRNGs and is capable of producing a much longer period of random numbers. The algorithm works by generating a sequence of unique numbers using a mathematical formula. It starts with an initial seed value, which is then used to generate a sequence of numbers using the formula. The sequence of numbers is then used to generate a random number. The algorithm is often used in computer games and simulations, as it is capable of generating a wide range of random numbers. It is also used in cryptography and other applications. where a secure random number generator is needed. The Mersenne Twister algorithm is one of the most widely used PRNGs and is considered one of the most reliable and secure random number generators available. It is also relatively fast, making it ideal for applications that require a large number of random numbers to be generated quickly.

2)Hashing Algorithms:

A hashing algorithm is a type of computer programming algorithm that creates a unique output from a given input. It is used to store and retrieve information from large datasets. Hashing algorithms are used for a variety of applications, such as data encryption, data integrity, data authentication, and digital signatures. Hashing algorithms are designed to be one-way functions, which means that it is difficult to determine the original input from the hash output. This makes it difficult for hackers to access sensitive data. Hashing algorithms also enable efficient lookups of data by creating an index of the data.Hashing algorithms are used in many areas of computer science and cryptography. In cryptography, hashing algorithms are used to create digital signatures and to verify the integrity of files. Hashing algorithms are also used in software development, to ensure that the software is not modified without the user's knowledge. There are many different types of hashing algorithms, each with its own set of advantages and disadvantages. Some of the most popular hashing algorithms include SHA1, SHA2, and MD5. Each of these algorithms is designed to be secure and reliable. SHA1 is the most widely used hashing algorithm. It is used in a variety of applications, including secure web communications and digital signatures. SHA2 is an improved version of SHA1. It has a larger hash length, which makes it more secure. MD5 is another popular hashing algorithm. It is used to check the integrity of files, and it is often used to create digital signatures. No matter which hashing algorithm is used, it is important to remember that they are not infallible. They can be broken by sophisticated hackers and malicious software. It is important to use strong passwords and other security measures to protect sensitive data. Hashing algorithms are essential for secure data storage and retrieval. They are used in a variety of applications, and they offer a reliable way to store and retrieve data. While no hashing algorithm is completely secure, they offer an important layer of security for sensitive data

3) ROT-13:

Rot-13 is a simple letter substitution cipher that replaces a letter with the letter 13 places further down the alphabet. It is a special case of the Caesar cipher which was developed in ancient Rome. Rot-13 is used for obscuring text, not for encrypting it. It is a method of hiding text so that only those who know the cipher can read it. It is often used for hiding jokes, puzzle solutions, offensive material, and spoilers in online conversations. Rot-13 works by shifting each letter of the alphabet by 13 places. For example, the letter A becomes the letter N, the letter B becomes O, and so on. To decrypt a Rot-13 message, simply shift the letters again by 13 places. Rot-13 is a weak cipher because it is easily broken by frequency analysis. This is because each letter is simply shifted by the same amount, leaving the frequencies of the letters unchanged. Rot-13 is also vulnerable to brute force attacks. An attacker can easily try all 26 possible shifts to break the cipher. Despite its weaknesses, Rot-13 remains a popular choice for hiding text online. It is simple to implement and provides a reasonable level of obscurity.

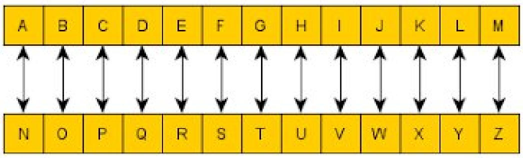


Fig. 5. ROT-13 algorithm

D. Representation of Password Generation Process:

This is a procedure for creating passwords using the aforementioned algorithms.-

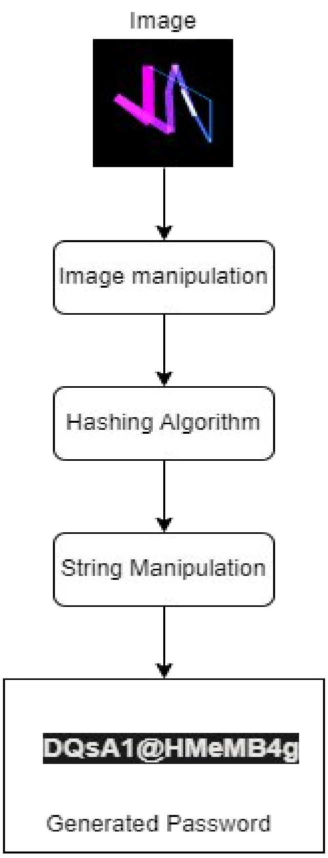


Fig. 6. Password Generation Process

Image manipulation: The first step is to manipulate the image in a way that generates a unique and unpredictable result. This can be done by applying various image processing techniques, such as rotating, scaling, flipping, or distorting the image. The goal is to create a modified image that is difficult for an attacker to guess or reverse-engineer.

Hashing algorithms: Once the modified image has been generated, the next step is to use a hashing algorithm to generate a cryptographic hash of the image. A hash function takes an input (in this case, the modified image) and produces a fixed-size string of characters, which is unique to that input. Popular hashing algorithms include MD5, SHA-1, and SHA-256.

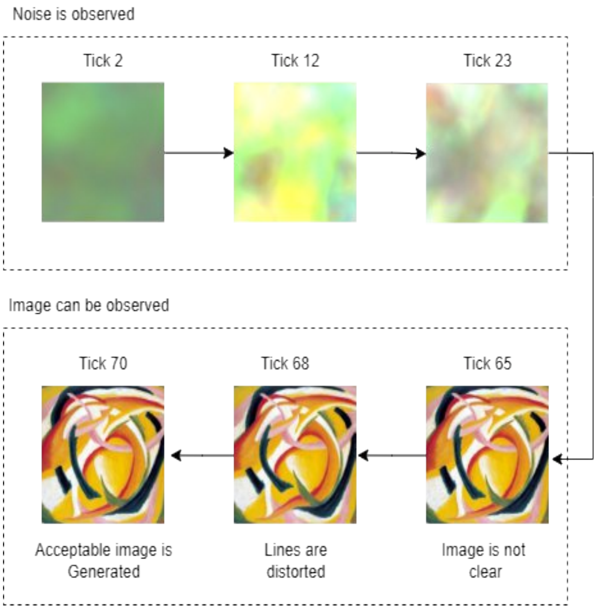
String manipulation: Finally, the hashed image is converted into a password using string manipulation techniques. This involves converting the hash into a sequence of characters, such as numbers, letters, and symbols, that meet the desired password requirements. For example, the password might need to be a certain length, contain a mix of upper and lowercase letters, and include special characters. The string manipulation process might involve truncating or expanding the hash, rearranging the characters, or adding additional characters to meet the password requirements.

Overall, this process generates a password that is unique to the original image, and is difficult for an attacker to guess or brute-force. However, it is important to note that this technique should be used in combination with other password security measures, such as using strong and unique passwords for each account, and enabling two-factor authentication where possible.

IV. RESULTS AND DISCUSSION

A. Image generation using GAN:

During training, the StyleGAN network learns to generate images that resemble a specific set of training data



At the beginning of training, the StyleGAN network generates images from random noise vectors. These images will look like random noise, with no discernible patterns or features.

As the network trains, it begins to refine its output, gradually generating images that more closely resemble the training data. This process typically takes many hours or even days, depending on the complexity of the dataset and the power of the hardware being used. [6]

Over time, the StyleGAN network will produce images that begin to take on recognizable features, such as facial features in the case of portrait training data. These images may still look distorted or strange, with exaggerated or unrealistic features.

As training continues, the network will gradually produce images that more closely resemble the training data, with realistic and proportionate features. The images may still have minor imperfections or artifacts, but they will generally be recognizable and acceptable as valid images.

The final output of a trained StyleGAN network will be high-quality images that are generated from random noise vectors. These images will be visually consistent with the training data.

B. Password generation:

The proposed technique of generating a password from an image using image manipulation, hashing algorithms, and string manipulation offers several benefits for password security. Firstly, it creates a unique password that is specific to the original image, making it difficult for attackers to guess or crack through brute-force methods. Secondly, the use of image manipulation techniques can enhance the unpredictability and complexity of the resulting password, adding an extra layer of security. Thirdly, the hashing of the modified image ensures that even if an attacker gains access to the original image, they cannot reverse-engineer the password. However, it should be noted that this technique should not be solely relied upon for password security. Combining this technique with other measures such as deploying strong and distinct passwords for every account and activating two-factor authentication where feasible, can better protect confidential information and prevent unauthorized access. In conclusion, a multi-layered approach to password security that incorporates the proposed technique can provide stronger protection against cyber threats and increase the overall security of sensitive information.

VI. CONCLUSION AND FUTURE SCOPE

In conclusion, this research explored the potential benefits of generative art and image manipulation techniques for enhancing the security of password-based authentication systems. Specifically, we discussed the StyleGAN algorithm, which can generate high-quality images that are challenging to distinguish from real images, and the proposed technique of generating a password from an image using hashing algorithms and string manipulation.

The results suggest that this technique can provide an additional layer of security for password-based authentication systems by creating a unique and complex password that is challenging for attackers to guess or crack. However, we also emphasized the importance of using this technique in combination with other password security measures, such as strong and distinct passwords for every account and enabling two-factor authentication.

Overall, the findings of this research highlight the potential benefits of incorporating generative art and image manipulation techniques into password security systems to enhance security and protect against unauthorized access. Future research can continue to explore and innovate in these areas to improve the effectiveness of password security measures and better protect sensitive information from cyber threats.

REFERENCES

[1] N. Matsumura, H. Tokura, Y. Kuroda, Y. Ito and K. Nakano, "Tile Art Image Generation Using Conditional Generative Adversarial Networks," 2018 Sixth International Symposium on Computing and Networking Workshops (CANDARW), Takayama, Japan, 2018, pp. 209-215, doi: 10.1109/CANDARW.2018.00047.

[2] N. Correia and A. Prata, "Multimedia application with interactive and generative digital animation for performance arts," Proceedings EC-VIP-MC 2003. 4th EURASIP Conference focused on Video/Image Processing and Multimedia Communications (IEEE Cat. No.03EX667), Zagreb, Croatia, 2003, pp. 771-776 vol.2, doi: 10.1109/VIPMC.2003.1220557.

[3] P. Morerio, R. Volpi, R. Ragonesi and V. Murino, "Generative Pseudo-label Refinement for Unsupervised Domain Adaptation," 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), Snowmass, CO, USA, 2020, pp. 3119-3128, doi: 10.1109/WACV45572.2020.9093579.

[4] R. Cutura, K. Angerbauer, F. Heyen, N. Hube and M. Sedlmair, "DaRt: Generative Art using Dimensionality Reduction Algorithms," 2021 IEEE VIS Arts Program (VISAP), New Orleans, LA, USA, 2021, pp. 59-72, doi: 10.1109/VISAP52981.2021.00013.

[5] A. Brock, J. Donahue and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis", International Conference on Learning Representations, 2019

[6] I. J. Goodfellow, "NIPS 2016 tutorial: Generative adversarial networks", CoRR, 2017.

[7] Grzegorz Kępisty,"Generative Art:Tricks and Traps",February 2022

[8] Alan Dorin, Gordon Monro, Jon McCormack,"A Framework for Understanding Generative Art", December 2012Digital Creativity 23(3-4) DOI:10.1080/14626268.2012.709940

[9] Margaret A. Boden & Ernest A. Edmonds (2009) What is generative art?, Digital Creativity, 20:1-2, 21-46, DOI: 10.1080/14626260902867915

[10] Renato Budinich,"A region-based easy-path wavelet transform for sparse image representation", International Journal of Wavelets, Multiresolution and Information ProcessingVol. 15, No. 05, 1750045 (2017)

[11] Steven Liu, Tongzhou Wang, David Bau, Jun-Yan Zhu, Antonio Torralba, "Diverse image generation via self-conditioned gans." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.

[12] Chen Wang, Jin Wang,"Generative image inpainting based on wavelet transform attention model." 2020 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, 2020.

[13] T. Karras, S. Laine, and T. Aila,"A style-based generator architecture for generative adversarial networks." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

[14] Z. Chen, L. Chen, Z. Zhao and Y. Wang, "AI Illustrator: Art Illustration Generation Based on Generative Adversarial Network," 2020 IEEE 5th International Conference on Image, Vision and Computing (ICIVC), Beijing, China, 2020, pp. 155-159, doi: 10.1109/ICIVC50857.2020.9177494.