Stochastic Modelling and Computational Sciences

A COMPREHENSIVE SURVEY ON EVALUATION METRICS FOR GENERATIVE ADVERSARIAL NETWORKS (GANS) WITH EMPHASIS ON EVALUATION MATRICES

Aakanksha S Choubey[1], Samta Gajbhiye[2] and Rajesh Tiwari[3*]
[1,2]Computer Science and Engineering, Shri Shankarachrya Technical Campus, Bhilai, Chhattisgarh, India 490020
[3] Computer Science and Engineering, CMR Engineering College, Hydrabad, Telangana, India 501401
[*]drrajeshtiwari20@gmail.com

ABSTRACT
This survey offers a thorough introduction to Generative Adversarial Networks (GANs), including information on their architectures, validation metrics, and prominent variations. Introduced in 2014, Generative adversarial networks (GANs) are useful instruments for generating a variety of and lifelike data across different fields, such as computer vision and other applied fields. The generative modelling space has completely changed with the advent of GANs, the combination is a generative network playing a minimax game with a discriminative network. Over time, numerous innovations have been made, leading to a enormous number of GAN variations. The aim of the survey is to offer a broad outline of GANs by summarizing their applications, assessment metrics, and underlying architecture. The survey concludes by addressing current issues and outlining further research directions in the field.

Keywords: GAN, generative network, minimax, assessment metrics.

1. INTRODUCTION
In 2014, Generative Adversarial Networks (GANs) were introduced by author [1], which have had a significant impact on deep learning by providing a strong structure for producing great class and different data. In a GAN setup, a generator creates data, whereas the discriminator assesses its validity. Through this adversarial process, the generator becomes better at producing data that nearly resembles actual data, and the discriminator grows into more adept at distinctive between actual and false data. Since their introduction, GANs have been applied in various domains, resulting in the creation of multiple specialized variations. Generating data based on predetermined conditions is made possible by Conditional GAN (CGAN) [2], CycleGAN [3] bests in unpaired image-to-image conversion, and StyleGAN [4] is renowned for generating images with diverse styles and structures. GANs have also been stretched to text [5], music [6], 3D modelling [7], and more, with applications in super-resolution, irregularity discovery, data augmentation, and image synthesis. GAN evaluation is difficult because of the intricacy of the data that is produced and the subjective nature of visual assessment, with various metrics proposed, among which the Inception Score (IS) is one of the most prominent.

2. LITERATURE SURVEY
Generative Adversarial Networks (GANs) [1] are a deep learning outline with the potential to create artificial data that closely look like real-world data. Goal of early studies in this field was to produce realistic images. In 2015, Radford et al. presented the Deep Convolutional GAN (DCGAN) [8], which produced high-quality images by utilizing batch normalization and convolutional layers and a particular loss function, marking significant advancements in image production. The Progressive Growing GAN (ProGAN), introduced by Karras et al. in 2017 [5], generates images with better resolution and quality than conventional GANs. ProGAN gradually increases the resolution of the generated images by using a stepwise training approach for multiple generators and discriminators. Their findings showed that ProGAN could produce images from a variety of datasets, including the CelebA dataset, that closely resembled actual photographs [9].

The corpus of literature on GANs currently in existence spans a wide range of analytical tasks and multiple surveys, frequently focusing on particular viewpoints within limited areas like computer vision (CV) and natural language processing (NLP). Jabbar et al., for example [10], explore GAN apply in areas like CV, NLP, music, and medicine, highlighting academic publications, real-world instances, and training challenges. Xia et al [11]. focus on GAN inversion techniques, discussing both reconstruction and optimization based methods, their advantages, and the difficulties involved. Durgadevi et al [12] provide an summary of the many proposed GAN variants up to 2020, detailing their applications across different domains. Alom et al [13], cover different facets of deep learning, including GANs, while Nandhini et al [14], investigate deep CNNs and GANs for analysing images computationally. Kulkarni et al [15]. present strategies for music generation based GAN, and Sampath et al [16], summarize advances GANs for CV tasks involving imbalanced datasets. Authour review time series applications by GAN models, and Xun et al. [17]. Review medical image segmentation based on GAN models. Ji et al. [18] focus on GAN constructions for representational music generation, and Wang et al. [19] review GAN frameworks addressing CV challenges. Author give a thorough analysis of task-oriented GAN applications, and Iglesias et al. [20] summarize recent GAN variants and their applications.

Back in 2014, a group led by Ian Goodfellow came up with a set of machine learning systems recognized as Generative Adversarial Networks, GANs [1]. GANs made up of neural networks named as generator and discriminator that employ in a competitive zero-sum game. The discriminator is responsible for evaluating the authenticity of the data, while the objective of generator is to create lifelike data. The objective is to reach where the generator generates data that is more and more convincing, and the discriminator develops extra skillful at distinguishing actual data from false data. The accompanying figure illustrates the general structure of GANs, depicting the roles and interactions of the generator and discriminator. Figure 1 shows general structure of Generative Adversarial Network.

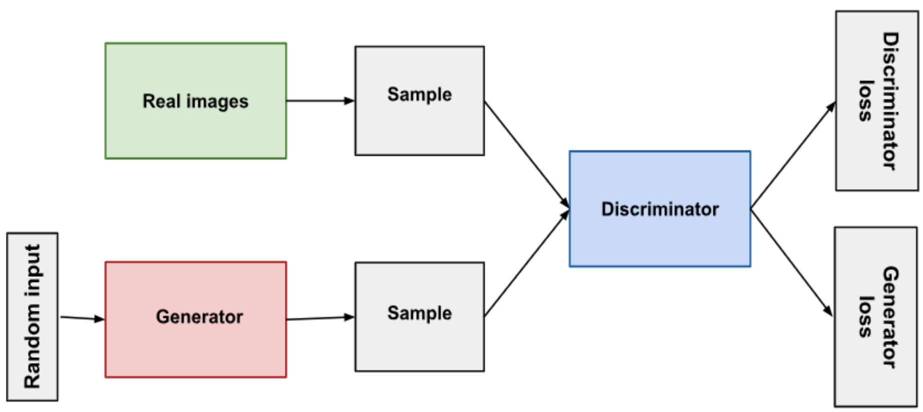

Figure1: General structure of Generative Adversarial Network

The way to express this minimax loss function as:

$$\min_G \max_D L = \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))], \quad (1)$$

The adversarial process drives two networks generator and discriminator to enhance their performance, resultant in the creation of high-quality data that closely mirrors real data. Equation 1 is the minimax loss function.

2.1 Generator and Discriminator Networks

2.1.1 Generator Network: The generator function is to generate data that looked alike to actual data. This means that the generated samples should be realistic enough that the discriminator not able to differentiate them from genuine data. The generator's success is measured by its ability to "fool" the discriminator into classifying its outputs as real. The generator typically operates as follows:

2.1.1.1 Generator Process: It starts with a random vector, often sampled from a standard usual distribution or even distribution. This vector serves as a latent representation of the data. The generator, which is usually a deep neural network with multiple layers, transforms this random vector into a structured data sample. The transformation involves a series of operations, such as fully connected layers, convolutional layers, and activation functions, which progressively refine the random noise into a coherent and meaningful data instance. The final production of the generator is a data sample that resides in the same space as the real data. For instance, if the GAN is trained on images, the generator's output will be an image with the same dimensions and characteristics as those in the training dataset.

2.1.1.2 Training: The training process of the generator is tightly coupled with the discriminator's performance and involves the steps: **Adversarial Loss**: The generator is trained using an adversarial loss function. This loss is derived from the discriminator's feedback. Specifically, the generator aims to maximize the probability that the discriminator misclassifies its outputs as real. Mathematically, this can be expressed as minimizing the following loss represented by equation 2 as:

$$L_G = -E_{z\sim(z)}[\log(D(G(z)))] \qquad (2)$$

Here, z is input noise vector, G(z) is the generated data, and D(G(z)) shows probability assigned by the discriminator that G(z) is real. **Backpropagation**: The gradients of the adversarial loss with respect to the generator's factors are calculated using backpropagation. These gradients indicate how the parameters of generator should be adjusted to improve its performance. Optimization: A stochastic gradient descent (SGD) or Adam optimization algorithm is used to update the generator's constraints. The objective is to iteratively minimize the adversarial loss, which corresponds to enhancing the class and practicality of generated data. **Iterative Improvement:** Iterative training is the method used. Every iteration, the generator creates a bunch of false data, which is then evaluated by the discriminator. The generator adjusts its parameters based on the discriminator's feedback. This adversarial loop continues, with the generator constantly enhancing its capability to create accurate data. **Feedback Loop**: The generator relies on the discriminator's evolving capability to differentiate actual data from false data. The discriminator gets better at recognizing fake one, the generator is pushed to generate even more accurate samples. This feedback loop drives both networks to improve over time.

2.1.1.3 Architectural Considerations

Architectural consideration consists some of the following issues in generator architecture. Network Depth and Complexity: The generator network's architecture can significantly influence the quality of the generated data. Deeper networks with more layers and parameters can capture more complex data distributions but require more computational resources and careful tuning to avoid over fitting. Activation Functions: Common activation functions used in the generator include ReLU (Rectified Linear Unit) in the hidden layers and Tanh in the output layer (for generating image data). These choices help in stabilizing the training process and producing high-quality outputs. Upsampling Techniques: In image generation tasks, the generator often employs upsampling techniques such as transposed convolutional layers (also known as deconvolutional layers) to rise the altitudinal resolution of the data progressively. This helps in generating high-resolution images from low-dimensional noise vectors. Batch Normalization: Batch normalization layers are commonly used in the generator to stabilize training and accelerate convergence. By normalizing the inputs to each layer, batch normalization helps mitigate issues related to internal covariate shift and ensures more consistent gradients.

2.1.2 Discriminator (D)

The primary aim of the discriminator is to accurately separate between actual data samples (from the actual data distribution) and false data samples (generated by the generator). Essentially, the discriminator acts as a binary classifier that outputs a probability score indicating whether a given input data sample is real or generated. The process of the discriminator involves several key steps: Input: The discriminator takes data samples as input. These samples can be either real data from the training dataset or synthetic data generated by the generator. The discriminator employs a series of neural network layers to extract relevant features from the input data. For image data, these layers usually consist of convolutional and pooling layers, along with activation functions. For additional data kinds, such text or audio, appropriate feature extraction techniques are used. After extracting features, the discriminator processes these features through fully connected layers to make a final classification. The output is a single scalar number, reflecting the likelihood that the supplied data is real, in the range of 0 to 1. A probability score is the discriminator's ultimate output. The data is categorized as real if the score is near to 1, and as fake if it is close to 0. The data is classified as fake. This probability score guides the generator in improving the realism of its generated samples.

2.1.2.1 Training: The following actions are involved in the discriminator's training process, which is crucial to the general effectiveness of GANs: Adversarial Loss: For training of discriminator loss function used is a binary cross-entropy. The goal of the loss function's design is to rise the likelihood that produced and real data will be classified properly. The loss to the discriminator represented by equation 3 as follows:

$$L_D = -E_{x\sim pdata(x)}[\log D(x)] - E_{z\sim pz(z)}[\log(1-D(G(z)))] \qquad (3)$$

Here, x represent the real data sample, z is noise vector, and the generated data is represented by G(z), and D is the discriminator. The grades of the loss regarding the discriminator's factors are calculated using backpropagation. These grades indicate how the discriminator's parameters should be adjusted to improve its classification accuracy. Stochastic gradient descent (SGD) or Adam are examples of optimization algorithms that are used to update the parameters of discriminator. The goal is to reduce the discriminator's loss, thereby improving its capacity to discriminate between generated and actual data. Training is done through iterations. Discriminator is first taught on a bunch of produced data and a batch of actual data in each iteration. The generator is then trained to enhance its capability to create realistic data that can deceive discriminator. This adversarial loop continues until both networks reach a point at which they are unable to get better their performance significantly. It is crucial to balance the preparation of discriminator and generator. If discriminator becomes too strong, the generator will struggle to improve. Conversely, if the discriminator is too weak, it will not deliver useful response to generator. Techniques such as alternating training steps and adjusting learning rates are used to maintain this balance. Figure 2 shows back propagation in discriminator.

2.1.2.2 Architectural Considerations: The architectural consideration of discriminator network consists, Network Depth and Complexity: The discriminator network typically has multiple layers to capture complex features of the data. However, it should not be too deep to avoid over fitting and ensure faster convergence.

Activation Functions: Common activation functions in discriminator include Leaky ReLU in hidden layers to prevent dead neurons and Sigmoid in the output layer to produce a probability score. Down sampling Techniques: For image data, the discriminator often uses convolutional layers followed by pooling layers or stride convolutions for down sampling. This aids in focusing on key features and lowering the spatial dimensions of the data. Regularization: To avoid overfitting and stabilize training, methods like failure, weight decay, and batch normalization are used. Batch normalization is particularly useful in maintaining consistent gradient flows and speeding up convergence. Figure 2 shows the back propagation in discriminator.

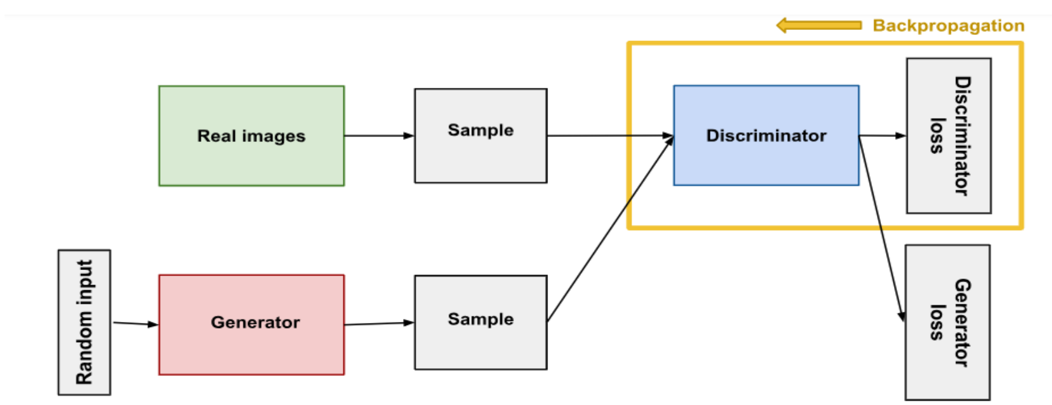

Figure2: Back propagation in discriminator.

2.2 Adversarial training process: The fundamental process that propels the enhancement of discriminator (D) and generator (G) of GAN, is adversarial training. In this process, generator and discriminator play game of zero-sum in which generator's goal is making fool the discriminator by making ever-more-accurate data, whereas the goal of discriminator is to recover at telling real data from fake. Important Phases in the Process of Adversarial Training: Set the generator and discriminator networks' initial parameters. For both networks, define the loss functions. Install optimizers (like Adam or SGD) to update the network's parameters. As input data real data samples were used from training dataset. From a predefined distribution (such as a uniform or normal distribution), create random noise vectors z. Discriminator Training: Forward Pass (Real Data): Pass the actual data x through the discriminator D to obtain the probability (x) that x is real. Forward Pass (Fake Data): Generate fake data G(z) using the generator. Pass G(z) through the discriminator to obtain the probability D(G(z)) that G(z) is real. Compute Discriminator Loss: Calculate the discriminator's loss using the binary cross-entropy expressed by equation 4 loss function:

$$L_D = -E_{x\sim pdata(x)}[\log D(x)] - E_{z\sim pz(z)}[\log(1-D(G(z)))] \qquad (4)$$

Backpropagation: Compute the gradients of L_D with respect to the discriminator's parameters. Parameter Update: Update the discriminator's parameters using the optimizer.

Generator Training:

Forward Pass (Fake Data): Generate fake data G(z) using the generator. Pass G(z) through the discriminator to obtain the probability D(G(z)) that G(z) is real. Compute Generator Loss: Calculate the generator's loss by the binary cross-entropy loss function represented by equation (5) as:

$$L_G = -E_{z\sim pz(z)}[\log D(G(z))] \qquad (5)$$

Backpropagation: Calculate the gradients of L_G with respect to the generator's constraints.

Parameter Update: Update the generator's parameters using the optimizer.

3. GAN VARIANTS

This segment will offer overall outline of different GAN prototypes, taking into account their distinct features and real-world uses. We will also go over these GAN variants' mathematical formulations using conventional notations and present their comparative study. The variants of GANs are:

3.1. Vanilla GAN: Generally speaking, the aim of Vanilla GAN is to produce new images by learning from the original image data. The generator, which creates images from noise, and the discriminator, which attempts to discern between z and the true images, x, are the two main components of the model. A minimax game is being played by the Vanilla GAN between a discriminator D and a generator G. Whereas D aims to separate them, G attempts to produce samples that are as near to the real data as feasible. The loss objective is therefore formulated as equation 6:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))] \qquad (6)$$

where $x \sim px(x)$ denotes the actual data distribution and $z \sim pz(z)$ is a random noise (often a standard Gaussian). Pictures produced by GANs are quite good even without the use of conventional feature engineering.

3.2. DCGAN: DCGAN, hosted by Radford et al. in 2015, represents an important innovation in generative AI, mainly in image generation. As a particular types of GAN construction, DCGANs integrate CNN and GAN methods to produce high-quality, photorealistic images with elaborate details. They can independently study and make images without extra control, highlighting their utility in unsupervised learning states. DCGANs are notable for their comparatively adaptable training procedure, facilitated by refined architectural workings such as stride complexities, batch normalization, and permeable corrected linear unit activation functions. Experimentally, DCGANs have demonstrated admirable outcomes on extensive image datasets like CIFAR-10 and ImageNet. However, they come with important computational demands, compassion to hyper parameters, and experiments such as limited variety of produced images and way ruin. Although these restrictions, DCGANs have been successfully applied in areas like image synthesis, style transfer, and image super-resolution. Their significant effect on generative modelling remains to motivate improvements and modernization in the field.

3.3. CGAN: Conditional GAN (CGAN) is a widespread kind of GAN that produce data by appending outside inputs such as tags or classes. Convolutional neural networks (CGANs) have been broadly used in applications such image synthesis of computer vision, picture-to-image conversion, and text-to-image synthesis since their debut by author [2]. Unlike classic GANs, the CGAN architecture directs the generator to produce data that satisfies the identified situations by providing conditional information (y) to both discriminator (D) and generator (G). The CGAN framework's loss function is provided by:

$$L_D = -E_{x\sim pdata(x)}[\log D(x)] - E_{z\sim pz(z)}[\log(1-D(G(z)))] \quad (7)$$

According to discussions in the literature [21], CGANs produce data that is tailored to a particular input. For example, based on the input, Images of a particular animal can be produced by a CGAN that has been trained on animal photos. Because CGANs receive additional inputs, they produce synthetic data that is higher quality than vanilla GANs and has better coherence, organization, and visual similarity to authentic samples. Because they employ outside input to direct the process of generating data, they also exhibit better resistance to noise than other artificial neural networks. Though CGANs are flexible, they also have a number of drawbacks. With high-dimensional datasets, they become computationally complicated; with sparse or noisy input data, they are prone to overfitting; they require clear tags or classes in the participation dataset; and they are susceptible to adversarial attacks [22,23]. Despite these shortcomings, CGANs show to be an effective method for producing data constructed on outside input. Nonetheless, it is crucial to take these restrictions into account when using CGANs to solve particular issues. Alternative conditioning techniques, such as the application of descriptions in natural language or a range of situations, can be studied in future studies [24].

3.4. Wasserstein GAN: Author introduced the Wasserstein GAN (WGAN), a version of GAN that maximizes the loss function to improve training stability and mitigate mode collapse problems. In order to enhance the creation of realistic samples and guarantee significant gradients during training, WGAN makes use of the Wasserstein distance. Weight clipping and the addition of a critic network allow WGAN to achieve improved stability during training. CGANs are flexible, but they have a few drawbacks. With high-dimensional datasets, they become computationally complex; with sparse or noisy input data, they are prone to overfitting; they need clear tags or classes in the input dataset; and they are susceptible to adversarial attacks [22, 23]. Despite these limitations, CGANs show to be a practical method for producing data from outside sources. Nevertheless, these limitations must be taken into account when using CGANs to solve particular problems. Alternative conditioning techniques, such as employing a range of scenarios or natural language descriptions, can be studied in future studies [24]. The Wasserstein distance in mathematics that converts the distribution P to the distribution Q is represented by equation 8 as follows:

$$W(\mathbb{P},\mathbb{Q}) = \inf_{\theta\in\pi(\mathbb{P},\mathbb{Q})} \mathbb{E}_{(\tilde{X},\tilde{Y})\sim\theta}\left[||\tilde{X}-\tilde{Y}||\right] \quad (8)$$

The Wasserstein GAN (WGAN), hosted by Arjovsky and Bottou in 2017, employs discriminator function D designed as a critic network. Instead of assigning probability values as in traditional GANs, The Wasserstein distance between the created and actual data supplies is estimated by the critic. The likeness or unlikeness between the input samples and the actual data circulation is shown by these scores. WGAN training requires optimizing the critic's parameters to maximize the alteration in detractor values between generated and real data. WGAN modifies its discriminator loss function to uphold the Lipschitz continuity condition while preserving the essential structure of the loss functions by clipping the discriminator weights. All things considered, WGANs have outperformed traditional GANs in terms of training stability. They are more resilient to mode collapse and less susceptible to hyperparameter changes. The Wasserstein distance is used to enable better gradient flow and smoother optimization, which results in training more quickly and with better samples. Though, computing the

Wasserstein distance can be computationally demanding. Even though WGANs provide improved stability, careful hyperparameter and network design tuning is still necessary to achieve acceptable outcomes. Moreover, WGANs may only be partially applicable to other forms of data because they are primarily designed for producing images. To sum up, WGANs are a promising development in GAN technology that solves a number of its drawbacks and offers insightful information about distribution distances. However, these difficulties must be carefully considered in order to apply them practically to real-world issues.

3.5. AttGAN: A variant of the GAN framework called AttGAN, or Attribute GAN, is intended to produce images with programmable features such as expression, gender, and age. AttGAN, which was first presented by He et al. in 2019 in their paper "AttGAN: Facial Attribute Editing by Only Changing What You Want," allows users to alter particular facial features without affecting the face's overall identity or appearance. Two sub networks—one is encoder (G_{Enc}) and other decoder (G_{Dec})—replace the traditional generator (G) of a typical GAN in the AttGAN architecture. Furthermore, the discriminator network is integrated with an attribute classifier (C). G_{Enc} encodes an input image $x^{\tilde{a}}$ into a latent vector representation $s = G_{Enc}(x^{\tilde{a}})$ during training, given a set of n-dimensional binary attributes $\tilde{a}$.

Simultaneously, Author used to manipulate the characteristics of $x^{\tilde{a}}$ to alternative set of n-dimensional attributes $\tilde{b}$, resulting in an edited image $x^{\hat{\tilde{b}}}$ defined as $x^{\hat{\tilde{b}}} = G_{Dec}(s, \tilde{b})$. In order to complete this unsupervised learning task, C works with the pair of encoders and decoders to make sure that $x^{\tilde{b}}$ displays the desired attribute changes. A reconstruction loss guarantees the preservation of attributes, ensuring that the collaboration between the latent vector s and attribute $\tilde{b}$ consistently produces $x^{\tilde{b}}$ and that the collaboration between s and attribute $\tilde{a}$ always produces $x^{\tilde{a}}$, resembling the input image $x^{\tilde{a}}$. The adversarial loss in the training process ensures the generation of convincing images. Consequently, the encoder-decoder-based generator of AttGAN's overall loss function can be written as equation 9 and equation 10 as follows:

$$L_{\text{Enc, Dec}} = \lambda_{\text{Rec}} \mathbb{E}_{x^{\tilde{a}}} \left[\| x^{\hat{\tilde{a}}} - x^{\tilde{a}} \|_1 \right] + \lambda_{\text{Cls}_G} \mathbb{E}_{x^{\tilde{a}}, \tilde{b}} \left[\mathrm{H}\left(\tilde{b}, C\left(x^{\hat{\tilde{b}}} \right) \right) \right] - \mathbb{E}_{x^{\tilde{a}}, \tilde{b}} \left[D\left(x^{\hat{\tilde{b}}} \right) \right] \quad (9)$$

and the following is how the classifier and discriminator's losses are calculated:

$$L_{\text{D, Cls}} = \lambda_{\text{Cls}_D} \mathbb{E}_{x^{\tilde{a}}} \left[\mathrm{H}\left(\tilde{a}, C\left(x^{\tilde{a}} \right) \right) \right] - \mathbb{E}_{x^{\tilde{a}}} \left[D\left(x^{\tilde{a}} \right) \right] + \mathbb{E}_{x^{\tilde{a}}, \tilde{b}} \left[D\left(x^{\hat{\tilde{b}}} \right) \right], \quad (10)$$

Here H is the cross-entropy loss and the hyperparameters are used to balance the losses. AttGAN provides a number of advantages in the field of picture creation, including exact governor over the features of produced images, permitting operators to change the expression, age, gender, and other features. Its adaptability to various domains and activities enables image synthesis applications to be customized and flexible. With the original image's visual qualities preserved, the model creates realistic images that roughly match the required features. Due to AttGAN's computational complexity, deploying it in production environments or on devices with limited resources may be difficult. Moreover, the success of AttGAN depends on labelled data with element explanations, which is not constantly readily accessible. The number and quality of attribute annotations can affect the model's performance and generalizability [25]. However, when employing AttGAN or comparable models, ethical issues pertaining to representation, identity, and privacy need to be taken into account [26]. The circulation and variety of preparation data can also affect the model's concert and capability to handle anomalous or out-of-distribution structures [27]. In summary, AttGAN offers accurate characteristic management, adaptability, and accurate image generation skills; however, when applying the model in practical settings, care should be taken with regard to resource needs, data dependencies, and ethical issues.

3.6. Least Squares GAN.

A discriminator modelled as a classifier with the sigmoid cross entropy loss function is commonly used in traditional GAN models. Nevertheless, if this choice of loss function resulted in vanishing gradients during training, the deep representations would struggle to learn. In response, Mao et al. unveiled least squares GAN (LSGAN), a cutting-edge method, in 2017, which substitutes the discriminator's value with the least square's loss function [28]. The following mathematical expressions correspond to the generator loss function (LG) and discriminator loss function (LD) of the LSGAN model as in equation 11:

$$L_{\mathrm{G}} = \frac{1}{2}\mathbb{E}_{z\sim p_z}\left[(D(G(z)) - c)^2\right],$$
$$L_{\mathrm{D}} = \frac{1}{2}\mathbb{E}_{x\sim p_{\mathrm{data}}}(D(x) - b)^2 + \frac{1}{2}\mathbb{E}_{z\sim p_z}(D(G(z)) - a)^2, \quad (11)$$

where the standards that G wants D to trust for false data are represented by the c encrypting scheme, and the labels for false data and actual data for D are represented by the a, b encoding scheme. Comparing the LSGAN framework to traditional GANs, there is a noticeable improvement, producing synthetic data of a higher caliber and providing better stability and convergence during training.

According to inception score (IS), it has produced more realistic images than standard GANs on a variety of datasets, including CIFAR-10 [28]. However, because the objective function of LSGANs uses squared loss, they frequently result in fuzzy images. Because the loss function punishes large differences between the false and actual images but ignores minor differences, the created images typically deficiency sharpness and sufficient details. To improve the sharpness of synthetic images, researchers have made adjustments to the loss function in later studies [29, 30]. Although LSGANs have the potential to produce high-quality images, more work is necessary to resolve these problems and produce precise and thorough results.

3.7. InfoGAN: a GAN modification, make the most of the common information between a subsection of the generator's input and output in order to learn disentangled representations of data. In 2016, Chen et al. presented it. This is how InfoGAN formulates the generator's loss function as given in equation 12:

$$L = \mathbb{E}_{x\sim p_{\mathrm{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))] - \lambda\mathcal{I}(c; G(z)), \quad (12)$$

where λ is a hyperparameter that controls the transaction between the adversarial loss and the common information term, and I(c;G(z)) is the mutual information between the learned latent code c and the generator's output G(z). The InfoGAN framework's information-theoretic methodology improves its capacity to study illustrations that make tasks involving data consideration, explanation, and manipulation easier. Regarding unsupervised learning assignments such as creating images and enhancing data, InfoGAN is a flexible and scalable alternative to supervised methods because it doesn't require explicit supervision or labeling However, for high-dimensional composite datasets, The InfoGAN architecture may discover It is difficult to study meaningful and interpretable illustrations, and its advantages might not permanently outweigh the added complication and computational cost. Overall, InfoGAN demonstrates encouraging performance in learning separated demonstrations, though specific goals, data characteristics, and resource availability all affect how effective it is [29]. Future developments and research may be able to solve these issues and enhance this strategy even more.

3.8. CycleGAN: Unlike traditional GANs, CycleGAN is an unsupervised image-to-image conversion framework that was presented by author and does not require paired training data. It is based on cycle consistency, which allows two generators and two discriminators to convert images between two domains while sustaining coherence. First generator (G_{XY}) converts images from the source domain X to the objective domain Y, while another generator (G_{YX}) does the opposite. Stated differently, the G_{YX} function is such that $G_{YX}(G_{XY}(x)) = x$. On the other hand, the discriminators differentiate between the genuine and converted images that the generators produce.

Training this construction necessitates CycleGAN's cycle regularity loss, which preserves the supposed advancing and regressive stability of the creative and round-trip translated images. This guarantees that translators translate intelligibly across domains, preserving important features and content. The cycle consistency loss function has the following mathematical expression equation 13:

$$\mathcal{L}_{\text{cycle}}(G_{XY}, G_{YX}) = \mathbb{E}_{x \sim p_{\text{data}}}\left[\|G_{YX}(G_{XY}(x)) - x\|_1\right] + \mathbb{E}_{y \sim p_{\text{data}}}\left[\|G_{XY}(G_{YX}(y)) - y\|_1\right]. \tag{13}$$

CycleGAN's primary benefit is its exceptional visual fidelity in image production. It performs well on a collection of image-to-image conversion jobs, such as object transformation, colorization, and style transfer. Large-scale dataset training is also made feasible by its computational efficiency. However, CycleGAN often suffers from MC, and its efficiency declines with increasing parameter count [31]. Nevertheless these drawbacks, CycleGAN is still a useful tool for translating images, and these problems should be the focus of any future work on data translation tasks [32]. For instance, it exhibits encouraging results in the adaptation for the medical imaging domain [33].

3.9. Style GAN: Initially, a random vector is fed into StyleGAN. This vector is mapped into a style vector that depicts various facets of the appearance and style of the picture. The style vector is then used by the generator network to create an image. The generated image's authenticity is assessed by the discriminator network. The generator and discriminator work together during StyleGAN training to try to outsmart one another. The discriminator attempts to identify genuine and correctly generated images, whereas the generator attempts to create images that can fool it.

3.10. ProGAN: The ProGAN was introduced by author in response to the drawbacks of conventional GANs, including low-resolution output and training instability. Through the use of advanced development technique, ProGAN progressively increases the generator and discriminator networks' size and complexity during training. The model can learn uneven features and upgrade them with this incremental approach, resulting in high-resolution images. ProGAN achieves training stability and produces exceptionally high-quality, visually realistic images by beginning with low-resolution image production and gradually adding layers and specifics at each stage. This method has proven effective in a number of fields, such as super-resolution, image synthesis, and style transfer. The generated images' resolution is gradually increased during training, going from a low resolution (like 4×4) to a high resolution (like 1024×1024). The generator and discriminator are modernized by a mix of loss functions at every resolution level. Progressive updates at increasing resolutions, in contrast to the traditional GAN framework, throughout training, ensure top-notch image creation with exquisite features and grains. Better scalability is provided by ProGAN, allowing images to be generated at any resolution. During training, it shows increased stability, resolving problems like MC. ProGAN's adaptability makes it appropriate for a range of image synthesis applications, such as medical imaging, video processing, and satellite imaging [34]. ProGAN training, however, may be computationally costly, particularly for big datasets or sophisticated models. As with other GANs, interpretability may present difficulties, making it challenging to distinguish the learned representations. Furthermore, ProGAN may not be as generalizable to novel or untested data, necessitating additional training on new datasets or fine-tuning [35].

3.11. BigGAN: BigGAN is a novel approach for large-scale GAN training that enables high-quality synthesis of natural imagesIn 2018, Brock et al. made the initial presentation of it [36]. It attempts to resolve the problem of producing high-resolution images, which is difficult for conventional GANs to accomplish [37]. BigGAN uses a large-scale construction and a special truncation method to produce high-fidelity images with fine details and textures. It also has a distinctive appearance. The model has been qualified on a large dataset of images and can generate images at different resolutions, up to 512×512 pixels. Similar to traditional GANs, the generator (G) and discriminator (D) parameters are updated using gradient descent techniques during the BigGAN model's training. Whereas the generator seeks to decrease the objective, the discriminator seeks to make the most of it. BigGAN incorporates self-attention mechanisms and class-conditional GANs (class-CGANs) as architectural modifications to improve image quality and variety. The generator's output is stabilized and controlled through the use of regularization techniques such as truncation tricks and orthogonal regularization. Progressive resizing and interpolation are two examples of data augmentation techniques that are used to manage high-resolution images. Regular GANs are not able to produce images with the same level of detail and texture as the BigGAN architecture due to its modified training approach. With improvements in scalability, mode collapse mitigation, and wide range of applications in video processing, satellite imaging, and medical imaging, this improved model is highly recommended. It is computationally taxing, though, particularly when working with big datasets or intricate models [37]. Furthermore, the framework's ability to generalize to new, hidden data is limited, frequently necessitating additional training on new datasets or fine-tuning [38].

3.12. Super Resolution GAN (SRGAN)

A GAN-based outline for super-resolution imaging was hosted by Ledig et al. in 2017 [39]. It uses discriminator networks and a generator to produce high-resolution images with an upscaling factor of 4 from low-resolution inputs. SRGAN combines adversarial and content losses with a perceptual loss function to achieve super-resolution. The perceptual loss is represented mathematically as equation 14 as:

$$l^{SR} = l_x^{SR} + 10^{-3} l_{Gen}^{SR}, \quad (14)$$

where the adversarial loss is denoted by l SR Gen and the content loss by l SR x. A pre-trained VGG-19 model is employed in the SRGAN framework's content loss, which gives the network information almost the generated image's quality and content. Conversely, the adversarial loss is in charge of guaranteeing that the generator network produces realistic images. SRGANs can produce high-quality images with improved textures and specifics, which raises the quality of the image as a whole. Research on perceptual quality has revealed that they are excellent at creating realistic and aesthetically pleasing images [40]. Because of their noise resistance, SRGANs can process noisy or low-quality input images and still provide high-quality results. Additionally, this model exhibits adaptability.

3.**13. StarGAN**A specialized GAN model known as StarGAN was initially held by Choi et al. [39] and is meant for multi-domain image-to-image conversions. StarGAN provides the ability to translate images across a multitude of areas using a generator and a discriminator, in contrast to CycleGAN, which focuses on translating images between two specialized domains. Through this model, the generator network G is trained to map an input image (x) to an crop image (y), conditionally on the casually created objective domain tag (c) (G(x,c) −→ y). An additional classifier, x −→ {Dsrc(x),Dcls(x)}, is used to generate the probability distribution for the source and domain labels D in the case of the discriminator network D. This system uses in order to ensure a multi-domain image translation by equation 15 is efficient.

$$\begin{aligned} L_G &= \mathbb{E}_x[\log D_{src}(x)] + \mathbb{E}_{x,c}[\log(1 - D_{src}(G(x,c)))] - \lambda_1 \mathbb{E}_{x,c}[-\log D_{cls}(c \mid G(x,c))] \\ &\quad + \lambda_2 \mathbb{E}_{x,c,c'}[\|x - G(G(x,c),c')\|_1] \text{ and} \\ L_D &= -\mathbb{E}_x[\log D_{src}(x)] - \mathbb{E}_{x,c}[\log(1 - D_{src}(G(x,c)))] - \mathbb{E}_{x,c'}[\log D_{cls}(c' \mid x)], \end{aligned} \quad (15)$$

where the hyper-parameters λ1 and λ2 regulate the effect of the reconstruction and the loss of domain classification. Loss in the corresponding StarGAN model. During the training phase, the elements of the loss functions are iteratively optimized to provide multi-domain image-to-image translations that are of excellent quality. When it comes to multi-domain image translation problems, the StarGAN framework has several benefits. To reduce computational complexity, for all domains, it employs a single generator-discriminator network. With little or no paired data, StarGAN can efficiently learn domain mappings while maintaining the character of input images inside the similar target domain. Nevertheless, it has a number of shortcomings, such as a complicated loss function that necessitates a lengthy training procedure [41]. It can be difficult to control image superiority and manage conversions between composite domains that have noticeable changes in appearance or structure in StarGAN [42]. Furthermore, there may be ethical issues with this model's ability to significantly alter images [43].

3.14. Mobile Image Enhancement GAN (MIEGAN): Pan et al.'s 2021 introduction of MIEGAN, a revolutionary technique in the field of GAN-based constructions, has as its main goal improving the visual quality of photos captured using mobile devices. This project entails multiple alterations to the traditional GAN architecture. An autoencoder and a feature converter are combined into a multi-module cascade generative network, which is used by the MIEGAN model. This improved generator's encoder consists of two streams, the second of which is in charge of boosting the areas that have poor brightness, which is a common problem in mobile photography that results in diminished clarity. A dual network structure is used in the feature transformational module to further capture the image's local and global information. Moreover, the MIEGAN model uses an adaptive multi-scale discriminator rather than a conventional single discriminator to improve the generative network's capacity to generate images with higher visual quality. On both a global and local scale, this multi-scale discriminator helps to distinguish between true and fraudulent photos. The discriminator uses an adaptive weight allocation technique to balance the assessments from the local and global discriminators.

4. EVALUATION METRICS FOR GAN-BASED MODELS

In contrast to traditional deep learning architectures, generative models such as GANs work a minimax loss function that is iteratively learned to balance the generator and discriminator networks. It is exciting to evaluate training progress or model performance using loss measurements for GAN training since the model lacks an objective loss function. Both qualitative and quantitative GAN evaluation techniques have been created in order to address this [44]. These techniques evaluate the produced data's potential uses as well as its quality and diversity [45]. Above the previous ten years, a number of metrics have arisen with different strengths and specialized applications due to the lack of a universal meter for assessing deep generative models. An overview of the common evaluation metrics used in various applications will be given in this section.

4.1. IS: The Inception Score (IS) is a frequently used statistic for evaluating the diversity and caliber of data generated by GANs [46]. It uses the Inception v3[47] neural network classifier, which was pretrained on the ImageNet [48] dataset, to classify generated samples across 1000 classes. Higher-quality samples are anticipated to have low entropy and good classification into particular classes; the IS gauges sample quality based on classification probabilities. Better performance in terms of diversity and quality is indicated by higher scores, which normally range from 1 to the number of sessions in the classifier.

But IS is not without its constraints. It has trouble with Mode Collapse (MC), where GANs produce extremely identical samples, inflating IS values unnecessarily and misrepresenting actual diversity. Furthermore, IS is dependent on Inception v3's performance, which isn't always consistent with how well images are seen by humans. Modified versions of IS have been presented as a solution to these problems. Modified IS, for example, measures diversity among images in the same category [49], while other variations, such as mode score, take into account the label's prior data distribution to evaluate both quality and diversity [50].

4.2. Fréchet Inception Distance

One well-known evaluation metric for determining the ability and variety of images produced by GANs is the Fréchet inception distance (FID). Similar to the Wasserstein-2 distance, the Fréchet distance is used to quantify the likenesses and variances between the circulations of real and artificial images. FID computes the distance between the produced and real picture distributions as well as their mean and covariance. Mathematically, FID is expressed as in equation 16:

$$\mathrm{FID} = |\mu - \mu_w|^2 + \mathrm{tr}\left(\Sigma + \Sigma_w - 2\left(\Sigma\Sigma_w\right)^{1/2}\right) \tag{16}$$

where the mean and covariance pairings for produced and real pictures, respectively, are represented by (μ, Σ) and (μw, Σw). The strength of FID is its capacity to take into account many types of contamination, such as blur, Gaussian noise, and other artifacts, which helps to provide a solid assessment of images generated by GANs. FID, which is widely used and approved, offers a standardized method for comparing outcomes between various GAN topologies, encouraging uniform evaluation of image quality [51].

4.3. The Classic Structural Similarity Index (SSIM): Using multi-scale structural similarity (MS-SSIM), one can assess how well GAN-generated images are, is an extension of the traditional structural similarity index (SSIM) [52]. It evaluates image properties at many scales, including contrast and luminance, providing a thorough comparison of synthetic and actual datasets structural and geometric similarities. MS-SSIM's capacity to take into account substantial dependencies among highly related pixels increases its sensitivity to perceptual quality.

4.4. Classifier Two-Sample Test (C2ST): A classification-based technique called the Classifier Two-Sample Test (C2ST) is used to appraise GANs' generalization ability in jobs containing the creation of synthetic data [53]. It uses a classifier to differentiate between produced and actual samples, such as one-nearest neighbor [52]. The quality of generated data is considered using the classifier's performance as a metric. Since C2ST is not restricted to any one data format, it is useful in a variety of domains. In order to complement other measures that address the distributional and perceptual elements of created data, it emphases on the discriminative part of data quality.

4.5. The subjective nature of musical perception makes it difficult to evaluate music made by GANs. For musical complexity, conventional quantitative criteria that are employed for visuals are inadequate. The coherence and quality of music produced by GANs are assessed using objective metrics that assess musical attributes, structure, style, distinctiveness, and tonality. Subjective listening provides information about musical quality by considering melody, harmony, rhythm, and emotional resonance, is still the most trustworthy assessment technique.

4.6. A statistical metric called maximum mean discrepancy (MMD): is used to express how different two probability distributions are from one another. By comparing the generated samples with actual data distributions constructed on their mean values in a high-dimensional space, MMD evaluates the quality of the samples in GAN evaluation. A lesser disparity between the data distributions is indicated by a lower MMD score, which implies that the artificial data is more alike to the original data.

4.7. The Temporal dependencies in time series data make it difficult to evaluate GAN models. Sequential data patterns are too complex to be captured by traditional measures designed for static images. As a result, a mix of quantitative and qualitative metrics is applied. The subjective visual judgment used in qualitative assessment is based on human opinion. Numerous quantitative methods, such as the Pearson correlation coefficient, Wasserstein-1 distance, dynamic time warping, and root mean square error, are used to address issue.

4.8. Quantification of uncertainty in GANs for trustworthy model-based inferences, uncertainty quantification (UQ) is critical for evaluating uncertainties in computational and practical applications. In order to produce out-of-distribution (OoD) samples and help classifiers accurately assess uncertainty in picture classification, Oberdiek et al. suggested an approach utilizing GANs. He and colleagues conducted a survey of UQ models in deep neural networks, emphasizing the difficulties in training GAN-based models even if they may incorporate uncertainty in the data. A different survey examined different UQ measures, but it also pointed out that the lack of ambiguous ground facts made validation challenging.

5. RESEARCH FINDINGS AND DISCUSSION

The versatility and adaptability of Generative Adversarial Networks (GANs) have led to a major expansion of their application areas beyond traditional text visualization. A survey of academic papers released has led to the following important discoveries:

- **Integration with Deep Learning:** To expand and improve their applications, GANs are becoming more and more integrated with deep learning methodologies. Strong database parameters are crucial for training GAN generators and discriminators, according to researchers.
- **Model Assessment Methods:** Sturdy model evaluation methods are essential for improving GAN performance and getting rid of flaws. It has been proven that combining different evaluation rules improves the performance of GAN models.
- **Various Uses: Visual Generations:** Conversions between various media types, including image-to-image, text-to-image, and audio-to-image jobs, are performed by GANs in architectural planning, 3D visual model generation, and other activities. Medical Imaging: GANs are getting a lot of attention for use in body scan and X-ray diagnosis, among other medical imaging applications. Signal Prediction: GANs work well for

anticipating and visually interpreting signals, especially in power supply units and medical applications. GANs have shown to be useful instruments in natural language processing (NLP) applications.

- **Addressing Shortfalls:** Despite their promise, GANs have a number of drawbacks, such as: Training Challenges: GAN training can be resource- and complexity-intensive. Large amounts of data and heterogeneous data can lead to model instability. Noise and incorrect forecasts: Inaccurate data might result in noisy outputs and incorrect forecasts. Performance Optimization: To maximize their effectiveness and lessen these difficulties, a thorough assessment of GAN variations is necessary.

6. CONCLUSION

In a variety of domains, GANs have proven to be effective tools for producing diverse and high-quality data. Notwithstanding obstacles like as unstable training and challenging evaluation, GANs have demonstrated impressive performance and have the potential to spur innovation. To further expand their potential, future research should concentrate on strengthening GAN stability, creating better assessment criteria, and investigating hybrid techniques. The aforementioned findings indicate that when new variants and framework enhancements appear, the potential and breadth of GAN applications continue to expand. Even though GANs have limited memory and need a lot of processing power, they are becoming into effective prediction tools for a variety of applications, including media conversions, 3D visual model creation, and architectural design. Precise assessment criteria are essential to demonstrate their value and appropriateness in order to fully realize their potential. Future studies ought to concentrate on a few crucial areas:

- **Task Dependency:** Examine the effectiveness of particular generative models in various settings.
- **Comparative Studies:** Evaluate current measurements through empirical and analytical comparisons.
- **Effective Evaluation Metrics:** Provide and investigate metrics for grading generative models and pinpointing their shortcomings.
- **Training Data Memorization:** Improve techniques for gauging the extent to which generative models can recall training data.
- **Evaluation Parameters:** Examine whether boosting generator samples with evaluation parameters such as IS or FID works well as a loss function.
- **Training Challenges:** Discuss technical issues including mode collapse, non-convergence, and instability that arises during GAN training and offer creative fixes.
- **Generalization and Fairness:** Determine how broadly or equally GAN-generated images can be used.

The aim of these study proposals is to enhance the application, performance, and dependability of GAN models in a variety of contexts**.**

REFERENCES

1. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative Adversarial Networks. arXiv preprint arXiv:1406.2661.
2. Mirza, M., & Osindero, S. (2014). Conditional Generative Adversarial Nets. arXiv preprint arXiv:1411.1784.
3. Zhu, J., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. Proceedings of the IEEE International Conference on Computer Vision (ICCV).
4. Karras, T., Laine, S., & Aila, T. (2019). A Style-Based Generator Architecture for Generative Adversarial Networks. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

5. Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2018). Progressive Growing of GANs for Improved Quality, Stability, and Variation. arXiv preprint arXiv:1710.10196.

6. Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., & Metaxas, D. N. (2017). StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. Proceedings of the IEEE International Conference on Computer Vision (ICCV).

7. Yu, L., Zhang, W., Wang, J., & Yu, Y. (2017). SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. Proceedings of the AAAI Conference on Artificial Intelligence.

8. Radford A, Metz L and Chintala S 2015 Unsupervised representation learning with deep convolutional generative adversarial networks (arXiv:1511.06434).

9. Zhang Y, Yin Z, Li Y, Yin G, Yan J, Shao J and Liu Z 2020 CelebA-Spoof: large-scale face anti-spoofing dataset with rich annotations Computer Vision–ECCV 2020: 16th European Conf. (Glasgow, UK, 23–28 August 2020)(Proc., Part XII vol 16) (Springer) pp 70–85.

10. Jabbar A, Li X and Omar B 2021 A survey on generative adversarial networks: variants, applications and training ACM Comput. Surv. 54, 1–49.

11. Xia W, Zhang Y, Yang Y, Xue J-H, Zhou B and Yang M-H 2022 GAN inversion: a survey IEEE Trans. Pattern Anal. Mach. Intell., 45, 3121–38.

12. Durgadevi M et al 2021, Generative adversarial network (GAN): a general review on different variants of GAN and applications 2021 6th Int. Conf. on Communication and Electronics Systems (ICCES) (IEEE) pp 1–8.

13. Alom M Z, Taha T M, Yakopcic C, Westberg S, Sidike P, Nasrin M S, Hasan M, Van Essen B C, Awwal A A and Asari V K 2019 A state-of-the-art survey on deep learning theory and architectures Electronics 8 292.

14. Nandhini Abirami R, Durai Raj Vincent P, Srinivasan K, Tariq U and Chang C-Y 2021 Deep CNN and deep GAN in computational visual perception-driven image analysis Complexity 2021, 1–30.

15. Arjovsky, M., & Bottou, L. (2017). Towards Principled Methods for Training Generative Adversarial Networks. arXiv preprint arXiv:1701.04862.

16. Kulkarni R, Gaikwad R, Sugandhi R, Kulkarni P and Kone S 2019 Survey on deep learning in music using GAN Int. J. Eng. Res. Technol. 8 646–8.

17. Sampath V, Maurtua I, Aguilar Martin J J and Gutierrez A 2021 A survey on generative adversarial networks for imbalance problems in computer vision tasks J. Big Data, 8, 1–59.

18. Brophy E, Wang Z, She Q and Ward T 2023 Generative adversarial networks in time series: a systematic literature review ACM Comput. Surv. ,55, 1–31.

19. Xun S et al 2022 Generative adversarial networks in medical image segmentation: a review Comput. Biol. Med. 140 105063.

20. Ji S, Yang X and Luo J 2023 A survey on deep learning for symbolic music generation: representations, algorithms, evaluations and challenges ACM Comput. Surv., 56, 1–39.

21. Wang Z, She Q and Ward T E 2021 Generative adversarial networks in computer vision: a survey and taxonomy ACM Comput. Surv., 54 , 1–38.

22. Ren W et al 2022 Experimental quantum adversarial learning with programmable superconducting qubits Nat. Comput. Sci. 2 711–7.

23. Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I and Fergus R 2013 Intriguing properties of neural networks (arXiv:1312.6199).
24. Denton E L, Chintala S and Fergus R et al 2015 Deep generative image models using a Laplacian pyramid of adversarial networks Advances in Neural Information Processing Systems p 28.
25. Dai T, Feng Y, Chen B, Lu J and Xia S-T 2022 Deep image prior based defense against adversarial examples Pattern Recognit., 122 ,108249.
26. Higgins I, Matthey L, Pal A, Burgess C, Glorot X, Botvinick M, Mohamed S and Lerchner A, (2017), Beta-VAE: learning basic visual concepts with a constrained variational framework Int. Conf. on Learning Representations.
27. Hou X, Shen L, Sun K and Qiu G 2017 Deep feature consistent variational autoencoder 2017 IEEE Winter Conf. on Applications of Computer Vision (WACV) (IEEE), pp 1133–41.
28. Mao X, Li Q, Xie H, Lau R Y, Wang Z and Paul Smolley S 2017 Least squares generative adversarial networks Proc. IEEE Int. Conf. on Computer Vision pp 2794–802.
29. Gui J, Sun Z, Wen Y, Tao D and Ye J 2021 A review on generative adversarial networks: algorithms, theory, and applications IEEE Trans. Knowl. Data Eng., 35 , 3313–32.
30. Huang Z, Wang X, Huang L, Huang C, Wei Y and Liu W 2019 CCNet: criss-cross attention for semantic segmentation Proc. IEEE/CVF Int. Conf. on Computer Vision pp 603–12.
31. Sergio Y C M W H and Colmenarejo G 2016 Learning to learn for global optimization of black box functions Stat 1050.
32. Gurumurthy S, Kiran Sarvadevabhatla R and Venkatesh Babu R 2017 DeLiGAN: generative adversarial networks for diverse and limited data Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp 166–74.
33. Hashemi S R, Salehi S S M, Erdogmus D, Prabhu S P, Warfield S K and Gholipour A 2018 Asymmetric loss functions and deep densely-connected networks for highly-imbalanced medical image segmentation: application to multiple sclerosis lesion detection IEEE Access, 7 ,1721–35.
34. Zhang R, Isola P, Efros A A, Shechtman E and Wang O 2018 The unreasonable effectiveness of deep features as a perceptual metric Proc. IEEE Conf. on Computer Vision and Pattern Recognition pp 586–95
35. He Z, Zuo W, Kan M, Shan S and Chen X 2019 AttGAN: facial attribute editing by only changing what you want IEEE Trans. Image Process, 28, 5464–78.
36. Cunningham P and Delany S J 2021 k-nearest neighbour classifiers-a tutorial ACM Comput. Surv., 54 ,1–25.
37. Brock A, Donahue J and Simonyan K 2018 Large scale GAN training for high fidelity natural image synthesis (arXiv:1809.11096).
38. Ledig C et al 2017 Photo-realistic single image super-resolution using a generative adversarial network Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp 4681–90.
39. Choi Y, Choi M, Kim M, Ha J-W, Kim S and Choo J 2018 StarGAN: unified generative adversarial networks for multi-domain image-to-image translation Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp 8789–97.
40. Li Y, Fang C, Yang J, Wang Z, Lu X and Yang M-H 2017 Universal style transfer via feature transforms Advances in Neural Information Processing Systems, p 30.

41. Huang X and Belongie S 2017 Arbitrary style transfer in real-time with adaptive instance normalization Proc. IEEE Int. Conf. on Computer Vision, pp 1501–10.
42. Isola P, Zhu J-Y, Zhou T and Efros A A 2017 Image-to-image translation with conditional adversarial networks Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp 1125–34.
43. Thies J, Zollhofer M, Stamminger M, Theobalt C and Nießner M 2016 Face2Face: real-time face capture and reenactment of RGB videos Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp 2387–95.
44. Johnson J, Alahi A and Fei-Fei L 2016 Perceptual losses for real-time style transfer and super-resolution Computer Vision–ECCV 2016: 14th European Conf. (Amsterdam, The Netherlands, 11–14 October 2016) (Proc., Part II vol 14) (Springer), pp 694–711.
45. Wang W, Yang H, Tuo Z, He H, Zhu J, Fu J and Liu J., (2023), VideoFactory: swap attention in spatiotemporal diffusions for text-to-video generation (arXiv:2305.10874).
46. Westerlund M 2019 The emergence of deepfake technology: review Technol. Innov. Manage. Rev., 9 ,40–53.
47. Korshunov P and Marcel S 2019 Vulnerability assessment and detection of deepfake videos 2019 Int. Conf. on Biometrics (ICB) (IEEE), pp 1–6.
48. Dumoulin V, Belghazi I, Poole B, Mastropietro O, Lamb A, Arjovsky M and Courville A 2016 Adversarially learned inference (arXiv:1606.00704).
49. Higgins I, Matthey L, Pal A, Burgess C, Glorot X, Botvinick M, Mohamed S and Lerchner A, (2017), Beta-VAE: learning basic visual concepts with a constrained variational framework Int. Conf. on Learning Representations.
50. Dai T, Feng Y, Chen B, Lu J and Xia S-T 2022 Deep image prior based defense against adversarial examples Pattern Recognit, 122 ,108249.
51. Hou X, Shen L, Sun K and Qiu G 2017 Deep feature consistent variational autoencoder 2017 IEEE Winter Conf. on Applications of Computer Vision (WACV) (IEEE), pp 1133–41.
52. Reed S, Akata Z, Yan X, Logeswaran L, Schiele B and Lee H 2016 Generative adversarial text to image synthesis Int. Conf. on Machine Learning (PMLR), pp 1060–9.
53. Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V and Courville A C 2017 Improved training of Wasserstein GANs Advances in Neural Information Processing Systems, p 30.