MANTECH Publications
Journal of Artificial Intelligence, Machine Learning and Soft Computing
Volume 10, Issue 2, July-December 2025

Generative Models: An Overview of GANs and Diffusion Models

Umesh Thakur[1], Nilesh Patankar[2], Farhan Qureshi[3]
Associate Professor[1], Assistant Professor[2]
Department of Computer Applications
Mar Ivanios College, Thiruvananthapuram, India
Email: *Umesh91thakur@yahoo.com[1], patankarnn14@gmail.com[2], Qureshi_farhan0j@rediffmail.com[3]*

ABSTRACT

Generative models have emerged as a powerful class of machine learning techniques capable of creating realistic data samples such as images, audio, video, and text. Among the most influential generative approaches are Generative Adversarial Networks (GANs) and Diffusion Models. GANs introduced a competitive learning framework between generator and discriminator networks, leading to high-quality image synthesis. However, training instability and mode collapse have limited their usability. Recently, diffusion models have gained attention due to their stable training process and impressive generative quality by modeling data through gradual noise addition and removal. This paper presents a comprehensive review of GANs and diffusion models, discussing their architectures, working principles, improvements, applications, advantages, and limitations. A comparative analysis is also provided to understand their differences and practical usability in real-world scenarios.

KEYWORDS: *Generative Models, GANs, Diffusion Models, Deep Learning, Image Synthesis, Adversarial Learning, Probabilistic Models*

INTRODUCTION

Generative modeling is a branch of machine learning where models learn to generate new data samples similar to the training data. Unlike discriminative models that classify or predict labels, generative models attempt to understand the underlying distribution of data. This capability has wide applications in computer vision, natural language processing, drug discovery, gaming,

and media creation.

The rise of deep learning has significantly enhanced generative modeling. Two major breakthroughs in this area are Generative Adversarial Networks (GANs) and Diffusion Models. GANs, introduced by Goodfellow et al., changed the landscape of image synthesis by using adversarial training. On the other hand, diffusion models, inspired by thermodynamics and stochastic processes, have shown remarkable stability and performance in image generation tasks.

This paper reviews both approaches in detail and highlights the evolution of generative techniques.

2. Background of Generative Modeling (Elaborated)

Generative modeling focuses on learning the underlying probability distribution of data, commonly denoted as P(x)P(x)P(x). Instead of predicting labels or outputs from inputs (as done in discriminative models), generative models try to **understand how the data itself is formed**. Once this distribution is learned, the model can **generate new samples** that resemble the training data.

Early generative models were grounded in probability theory and statistics. These models worked well for low-dimensional, structured data but faced serious challenges when applied to complex, high-dimensional inputs such as images, audio signals, and natural language. Nevertheless, they formed the conceptual base upon which modern deep generative models like GANs and diffusion models were later built.

The most important traditional generative approaches include Gaussian Mixture Models (GMM), Hidden Markov Models (HMM), Variational Autoencoders (VAE), and Autoregressive models.

2.1 Gaussian Mixture Models (GMM)

Gaussian Mixture Models assume that the data distribution can be represented as a mixture of several Gaussian (normal) distributions. Each Gaussian component represents a cluster or mode in the data.

Mathematically,

$P(x)=\sum_{k=1}^{K} \pi_k \ \mathcal{N}(x \mid \mu_k, \Sigma_k)P(x)=k=1\sum K\pi k\ N(x|\mu k,\Sigma k)P(x) = \sum_{k=1}^{K} \pi_k \ \mathcal{N}(x \mid \mu_k, \Sigma_k)P(x)=k=1∑Kπk N(x|μk,Σk)$

where:

- $\pi k\pi_k\pi k$ is the mixing coefficient,
- $\mu k\mu_k\mu k$ is the mean,
- $\Sigma k\Sigma_k\Sigma k$ is the covariance matrix.

GMMs are trained using the **Expectation-Maximization (EM)** algorithm. They were widely used for clustering, density estimation, and speech recognition.

Limitations for high-dimensional data:

- Covariance matrices become very large and difficult to estimate.
- Assumption of Gaussian distribution is too simple for image data.
- Computationally expensive as dimensions increase.

2.2 Hidden Markov Models (HMM)

Hidden Markov Models are probabilistic models designed for **sequential data**. They assume that the system being modeled is a Markov process with hidden states.

An HMM consists of:

- Hidden states
- Transition probabilities between states
- Emission probabilities that produce observations

HMMs were highly successful in speech recognition, handwriting recognition, and bioinformatics.

Why HMM struggled with images:

- Designed mainly for 1D sequential data, not 2D spatial data.
- Cannot effectively model complex spatial dependencies in images.
- Limited representation power for high-dimensional observations.

2.3 Variational Autoencoders (VAE)

Variational Autoencoders introduced deep learning into generative modeling. VAEs are

composed of two networks:

- **Encoder:** Maps input xxx to a latent representation zzz
- **Decoder:** Reconstructs xxx from zzz

Instead of learning a direct mapping, VAE learns a probability distribution over the latent space by optimizing the **Evidence Lower Bound (ELBO)**:

L=Eq(z|x)[log⁡p(x|z)]−DKL(q(z|x)‖p(z))\mathcal{L} = \mathbb{E}_{q(z|x)}[\log p(x|z)] - D_{KL}(q(z|x) \| p(z))L=Eq(z|x)[logp(x|z)]−DKL(q(z|x)‖p(z))

This allows sampling from the latent space to generate new data.

Advantages:

- Stable training
- Probabilistic foundation
- Continuous latent space

Limitations:

- Generated images often blurry
- Trade-off between reconstruction and regularization
- Limited sharpness compared to GANs

2.4 Autoregressive Models

Autoregressive models generate data one element at a time, where each element depends on the previously generated ones:

P(x)=∏i=1nP(xi|x1,x2,...,xi−1)P(x) = \prod_{i=1}^{n} P(x_i \mid x_1, x_2, ..., x_{i-1})P(x)=i=1∏nP(xi|x1,x2,...,xi−1)

Examples include:

- PixelRNN / PixelCNN for images
- GPT-like models for text
- WaveNet for audio

These models explicitly factorize the joint distribution into conditional probabilities.

Strengths:

- Exact likelihood computation

- High-quality sample generation
- Strong theoretical basis

Weaknesses:

- Very slow sampling (sequential process)
- Computationally expensive for large images
- Hard to scale to very high resolutions

2.5 Challenges with High-Dimensional Data

Images, videos, and audio signals are high-dimensional and contain complex spatial or temporal relationships. Traditional generative models struggled because:

- Exponential growth of parameters with dimensions
- Difficulty in capturing non-linear patterns
- High computational cost
- Poor scalability

These challenges demanded models that could automatically learn hierarchical representations from data.

2.6 Transition to Deep Generative Models

The introduction of deep neural networks allowed models to learn **complex non-linear mappings** and hierarchical features directly from raw data. This shift led to:

- GANs using adversarial training without explicit likelihood
- Diffusion models using stochastic noise processes
- Flow-based models using invertible transformations

Deep architectures overcame the representation limitations of classical methods and enabled realistic image, audio, and text generation.

Thus, traditional generative models provided the theoretical base, while deep learning enabled practical, high-quality generative systems capable of handling real-world high-dimensional data.

3. GENERATIVE ADVERSARIAL NETWORKS (GANS) — ELABORATED

Generative Adversarial Networks (GANs), introduced by Ian Goodfellow in 2014, brought a

revolutionary change in generative modeling. Unlike earlier probabilistic models that explicitly estimate the data distribution, GANs learn to generate data **implicitly** through an adversarial game between two neural networks. This adversarial learning enables GANs to produce highly realistic synthetic samples, especially in image generation tasks.

GANs are considered one of the most creative frameworks in deep learning because they mimic a game-like scenario where two models continuously improve by competing with each other.

3.1 Basic Architecture

A GAN consists of two main components:

1. Generator (G)

The generator creates fake data samples from a random noise vector zzz, usually drawn from a simple distribution such as uniform or Gaussian noise.

G(z)→xfakeG(z) \rightarrow x_{fake}G(z)→xfake

Its objective is to learn how to transform noise into data that looks similar to real data.

2. Discriminator (D)

The discriminator acts as a binary classifier. It receives both real data x∼pdatax \sim p_{data}x∼pdata and fake data G(z)G(z)G(z), and predicts whether the input is real or fake.

D(x)→[0,1]D(x) \rightarrow [0,1]D(x)→[0,1]

A value close to 1 means real, and close to 0 means fake.

Minimax Objective Function

GAN training is formulated as a two-player minimax game:

min⁡Gmax⁡DV(D,G)=Ex∼pdata[log⁡D(x)]+Ez∼pz[log⁡(1−D(G(z)))]\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}}[\log D(x)] + \mathbb{E}_{z \sim p_z}[\log(1 - D(G(z)))]GminDmaxV(D,G)=Ex∼pdata[logD(x)]+Ez∼pz[log(1−D(G(z)))]

- The **discriminator** tries to **maximize** this objective by correctly classifying real and fake samples.
- The **generator** tries to **minimize** this objective by producing samples that fool the discriminator.

This competitive process pushes the generator to learn the true data distribution.

3.2 Working Principle

The training process of GANs occurs in iterative steps:

1. Sample real data from the dataset.
2. Sample random noise and generate fake data using the generator.
3. Train the discriminator on both real and fake samples.
4. Update the generator based on how well it fooled the discriminator.
5. Repeat the process until the generator produces highly realistic samples.

Over time:

- The discriminator becomes better at detecting fake data.
- The generator becomes better at producing realistic data.

At equilibrium, the discriminator cannot distinguish between real and fake samples, meaning:
$D(x)=0.5D(x) = 0.5D(x)=0.5$
This indicates that the generator has successfully learned the data distribution.

Training Challenges

Despite their success, GANs are notoriously difficult to train due to:

- **Mode collapse:** Generator produces limited variety of outputs.
- **Vanishing gradients:** Discriminator becomes too strong.
- **Non-convergence:** Oscillatory training behavior.
- **Sensitive hyperparameters**

These issues motivated the development of many improved GAN variants.

3.3 Popular GAN Variants

Over the years, several variants of GANs were developed to address stability, quality, and control issues.

Variant	Key Idea	Application
DCGAN	Convolutional layers for images	Image synthesis
Conditional GAN	Adds label condition	Controlled generation
CycleGAN	Image-to-image translation	Style transfer
WGAN	Wasserstein distance for stability	Stable training

Variant	Key Idea	Application
StyleGAN	Style-based architecture	High-resolution faces

3.4 Advantages of GANs (Elaborated)

Generative Adversarial Networks became extremely popular mainly because of the practical advantages they offered over earlier generative approaches like VAEs and autoregressive models. Their adversarial learning mechanism allows them to generate highly realistic outputs with efficient inference.

1. High-Quality Sharp Images

One of the biggest strengths of GANs is their ability to produce **visually sharp and detailed images**.

- GANs do not optimize pixel-wise reconstruction loss (like mean squared error used in VAEs), which usually leads to blurry outputs.
- Instead, the generator learns through feedback from the discriminator, which judges whether the image looks *real* rather than whether it matches pixel values.
- This adversarial loss encourages the generator to capture fine textures, edges, and high-frequency details.
- Models like **StyleGAN** and **BigGAN** can generate faces and objects that are almost indistinguishable from real photographs.

This makes GANs highly suitable for applications such as face synthesis, art generation, super-resolution, and gaming graphics.

2. Fast Sampling

After training, GANs can generate samples **very quickly**.

- Generation requires only a **single forward pass** through the generator network.
- There is no iterative denoising (as in diffusion models) or sequential pixel generation (as in autoregressive models).
- This makes GANs suitable for **real-time applications** such as video generation, interactive design tools, and augmented reality.

Because of this speed, GANs are often preferred when latency is critical.

3. Flexible Architecture

GANs are highly flexible and can be adapted for various tasks by modifying the architecture or training setup.

- Conditional GANs allow control over the output using labels or text.
- CycleGAN and Pix2Pix perform image-to-image translation.
- SRGAN performs image super-resolution.
- 3D-GANs and Video-GANs extend the idea to 3D objects and video frames.

This architectural flexibility allows GANs to be used in diverse domains beyond simple image generation.

3.5 Limitations of GANs (Elaborated)

Despite their impressive results, GANs are known to be difficult to train and maintain. Several practical issues limit their robustness and usability.

1. Mode Collapse

Mode collapse occurs when the generator learns to produce **limited varieties** of outputs.

- Instead of learning the full data distribution, the generator finds a few samples that successfully fool the discriminator and keeps generating similar outputs.
- For example, a GAN trained on faces may generate faces with very similar features repeatedly.
- This reduces diversity and defeats the purpose of generative modeling.

Many GAN variants such as WGAN and minibatch discrimination were proposed to reduce this **problem, but it still remains a challenge.**

2. Training Instability

GAN training is inherently unstable because it involves two networks competing with each other.

- If the discriminator becomes too strong, the generator receives almost no gradient (vanishing gradient problem).
- If the generator becomes too strong, the discriminator fails to learn.
- The training may oscillate and fail to converge to equilibrium.
- Small changes in learning rates or architecture can cause failure.

This instability makes GANs harder to train compared to VAEs or diffusion models.

3. Difficult Hyperparameter Tuning

GAN performance is very sensitive to hyperparameters such as:

- Learning rate
- Batch size
- Network architecture
- Optimizer choice
- Loss function variants

Finding the right combination often requires significant experimentation and experience. Unlike more stable models, GANs do not have a straightforward training recipe that works universally.

4. Diffusion Models

4.1 Concept

Diffusion models work by gradually adding noise to data (forward process) and learning to reverse this process (denoising) to generate new data.

4.2 Forward and Reverse Process

Forward:

q(xt|xt−1)=N(xt;1−βtxt−1,βtI)q(x_t | x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I)q(xt|xt−1)=N(xt;1−βtxt−1,βtI)

Reverse:

Learn pθ(xt−1|xt)p_\theta(x_{t-1} | x_t)pθ(xt−1|xt) to remove noise step by step.

4.3 Denoising Diffusion Probabilistic Models (DDPM)

DDPM uses neural networks to predict noise at each timestep and reconstruct images from pure noise.

4.4 Diffusion Variants

Model	Feature	Use Case
DDPM	Basic diffusion	Image generation
DDIM	Faster sampling	Efficient generation

Model	Feature	Use Case
Latent Diffusion	Operates in latent space	Text-to-image
Stable Diffusion	Open-source model	Creative media
Imagen	Large-scale diffusion	Photorealistic images

4.5 Advantages of Diffusion Models

- Stable training
- High diversity
- Less mode collapse

4.6 Limitations

- Slow sampling
- High computational cost

5. COMPARATIVE ANALYSIS OF GANS AND DIFFUSION MODELS

Aspect	GANs	Diffusion Models
Training Stability	Low	High
Sample Quality	Very sharp	Very realistic
Mode Collapse	Common	Rare
Sampling Speed	Fast	Slow
Complexity	Moderate	High
Applications	Faces, art, video	Text-to-image, photorealism

6. APPLICATIONS

6.1 Image Generation

GANs and diffusion models generate photorealistic images used in media, games, and advertising.

6.2 Medical Imaging

Used to generate synthetic MRI/CT scans for training.

6.3 Text-to-Image Systems

Diffusion models power systems like Stable Diffusion and DALL·E.

6.4 Video Generation

GAN-based video synthesis and emerging diffusion video models.

6.5 Drug Discovery

Molecule generation using generative approaches.

RECENT RESEARCH TRENDS

- Hybrid GAN-Diffusion models
- Faster diffusion sampling techniques
- Text-guided generation
- 3D generative modeling
- Ethical and copyright concerns

CHALLENGES AND FUTURE DIRECTIONS

- Reducing computational cost
- Improving controllability
- Avoiding bias in generated content
- Real-time generation
- Responsible AI usage

CONCLUSION

Generative models have revolutionized AI-driven content creation. GANs introduced adversarial learning and produced high-quality results but faced stability issues. Diffusion models provided a more stable and powerful alternative, now leading in text-to-image and photorealistic generation. Both approaches continue to evolve, and hybrid methods may define the future of generative AI.

REFERENCES

1. Goodfellow, I., et al. “Generative Adversarial Nets.” NIPS, 2014.
2. Radford, A., Metz, L., Chintala, S. “DCGAN.” ICLR, 2016.
3. Arjovsky, M., et al. “Wasserstein GAN.” ICML, 2017.
4. Karras, T., et al. “StyleGAN.” CVPR, 2019.
5. Ho, J., Jain, A., Abbeel, P. “Denoising Diffusion Probabilistic Models.” NeurIPS, 2020.
6. Song, J., et al. “DDIM.” ICLR, 2021.

7. Rombach, R., et al. “Latent Diffusion Models.” CVPR, 2022.
8. Saharia, C., et al. “Imagen.” ICML, 2022.
9. Nichol, A., Dhariwal, P. “Improved DDPM.” ICML, 2021.
10. Brock, A., et al. “Large Scale GAN Training.” ICLR, 2019.