Advances in Consumer Research
https://acr-journal.com/

Volume-2 | Issue-4 | Oct 2025

Original Researcher Article

Enhanced Image Synthesis with GANs: A Hybrid AE–VAE Based Generator Approach

Amit Kumar

Department of Computer Science & Engineering, CCSIT, Teerthanker Mahaveer University, Moradabad (UP), India
Email: amitpanwar889@gmail.com

Received:02/08/2025
Revised: 18/09/2025
Accepted:01/10/2025
Published:06/10/2025

ABSTRACT
Generative Adversarial Networks (GANs) have recently gained remarkable attention in the deep learning community for their ability to generate high-resolution images. They have been successfully applied across diverse domains, including computer vision, semantic segmentation, and medical imaging. A GAN typically consists of two core components: a generator that synthesizes images and a discriminator that evaluates their authenticity.In this work, we propose a generator framework that integrates both Autoencoder (AE) and Variational Autoencoder (VAE), thereby exploiting the non-probabilistic and probabilistic characteristics of data. The generator produces images under specific conditions, while the discriminator ensures the separation of real and generated samples. The novelty of this study lies in the introduction of a hybrid generator design that combines AE and VAE models within the GAN architecture. By leveraging deterministic and probabilistic data representations simultaneously, the proposed approach aims to improve image quality and achieve superior high-resolution generation compared to conventional GAN models.

Keywords: Generative model; GAN; Autoencoder (AE); Variational autoencoder(VAE).

INTRODUCTION

In recent years, deep learning–based generative models have attracted growing interest due to their remarkable advancements in the field. By leveraging large-scale datasets, carefully designed network architectures, and efficient training strategies, these models have demonstrated impressive capabilities in producing highly realistic content such as images, text, and audio. Among them, two major families stand out: Generative Adversarial Networks (GANs) [1] and Variational Autoencoders (VAEs) [2], both of which belong to the broader class of generative models. In a generative model, the network is trained to create objects (e.g., images) from their given descriptions or representations. An encoder processes the input data (x) by mapping it into a latent space representation (z). In the case of an autoencoder (AE), this mapping is deterministic (non-probabilistic), whereas in a variational autoencoder (VAE), it is probabilistic (non-deterministic). The decoder then reconstructs the original data ($\hat{x}$) from the latent representation. On the other hand, GANs employ an adversarial framework comprising two networks: a generator and a discriminator. The generator creates new data samples using existing data, while the discriminator attempts to distinguish between real and generated data. Through iterative training, both networks improve simultaneously, competing in an adversarial setup. Training reaches a stable state, known as Nash equilibrium, when the discriminator can no longer differentiate between real and generated samples. Unlike VAEs, GANs do not explicitly estimate probability distributions but instead model them implicitly. Both networks are trained together using backpropagation.

This paper presents a comparative analysis of GANs and VAEs, highlighting their respective advantages and limitations, with a primary focus on their ability to generate synthetic images.

Different Approach of Generating Images

Autoencoder (AE)

An autoencoder is an unsupervised learning algorithm designed for representation learning. It works by learning a compressed form (code or embedding) of the input data and reconstructing it at the network's output. The goal is that this compressed representation captures the underlying structure of the data—i.e., the intrinsic relationships between its variables—thereby supporting more effective downstream analysis [3]. Autoencoders have been successfully applied to a variety of tasks, including dimensionality reduction, data denoising, compression, and data generation. Typically, an autoencoder is composed of two main components as shown in fig1.

- Encoder: Maps high-dimensional input data into a lower-dimensional latent representation.
- Decoder: Attempts to reconstruct the input data from this latent representation.

The central idea of an autoencoder is to train both encoder and decoder networks jointly in order to learn the optimal encoding–decoding process. During training, data is passed through the encoder–decoder pipeline, the reconstructed output is compared with the original input, and the reconstruction error is minimized through backpropagation, updating the network weights iteratively

Fig 1: Architecture of Autoencoder and its loss function

Two important considerations must be kept in mind when using autoencoders. First, achieving strong dimensionality reduction without reconstruction loss often comes at the cost of producing latent spaces that lack interpretability and regularity. Second, the main objective of dimensionality reduction is not simply to decrease the number of dimensions, but to do so while preserving the essential structural information of the data in the reduced representation. For these reasons, both the dimensionality of the latent space and the depth of the autoencoder (which define the degree and quality of compression) need to be carefully adjusted according to the intended application of dimensionality reduction.Over time, several variants of autoencoders have been introduced to address these shortcomings, aiming for improved generalization, better disentanglement, and adaptation to sequence-based input models. Some notable examples include the Denoising Autoencoder (DAE) [4], the Sparse Autoencoder (SAE) [5], and the more recent Variational Autoencoder (VAE). Traditional autoencoders also face challenges in reconstruction: they tend to form distinct clusters for each class, making it difficult for the decoder to reconstruct inputs since a different code is required for each image. Furthermore, because the latent space in standard autoencoders is often discontinuous, the decoder is unable to reconstruct data from unexplored points in the space [6].

Variational Autoencoder (VAE)

Variational autoencoder is a type of generative model define in 2013.Variational autoencoder is a very popular generative model. It is a combination of two neural network. First network is an encoder network, which takes a data as an input and learn some hidden latent representation of a data. Encoder network convert input data into an encoding vector. Each dimension of an encoding vector represent some feature about a data and is a single value. For example, an encoder network of a generative model that generates an image of human face, will learn features of a human face like smile, hair colour, skin tone etc and represent it in a form of encoding vector with some single value for each feature. Second network is a decoder network, which will take input an encoding vector and learn to generate an original data as an output. Variational autoencoder differs from a traditional autoencoder by instead of learning fix latent representation, it will learn probabilistic distribution for each latent feature of a data [7].

The goal of Encoder network is to compute posterior probability which is P (Z|X

$$P(Z|X) = (P(X \mid Z)\ P(Z))/(P(X)) \quad \text{(From Bayes Rule)}$$

VAE: Network Realization

Just as a standard autoencoder, a variational autoencoder is an architecture composed of both an encoder and a decoder and that is trained to minimize the reconstruction error between the encoded-decoded data and the initial data. However, in order to introduce some regularization of the latent space, we proceed to a slight modification of the encoding-decoding

process: instead of encoding an input as a single point, we encode it as a distribution over the latent space. The model is then trained as follows: first, the input is encoded as distribution over the latent space second, a point from the latent space is sampled from that distribution third, the sampled point is decoded and the reconstruction error can be computed finally, the reconstruction error is backpropagated through the network

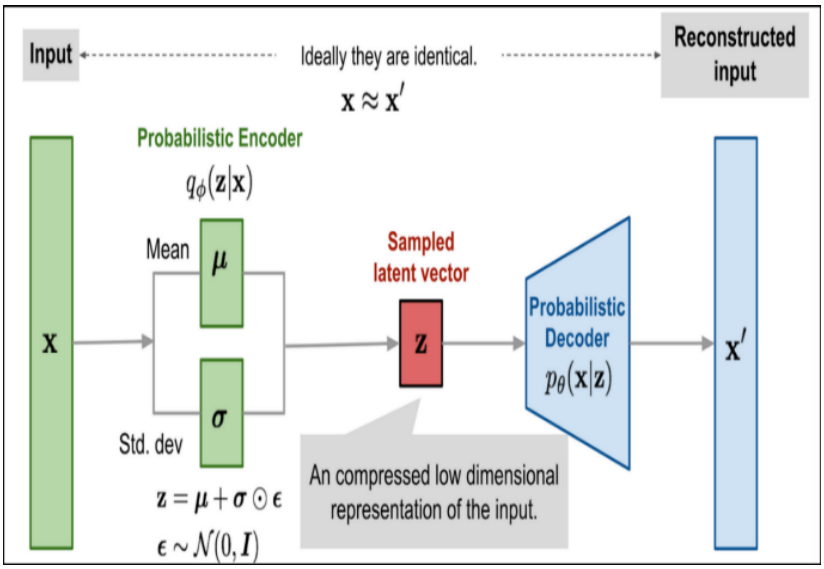

Fig 3: Archicture of the Variational Autoencoder. [9]

An experiment is carried out to evaluate performance of this model. Variational autoencoder is trained on MNIST dataset. Encoder is implemented as a convolutional neural network. Encoder contains one input layer, four hidden layers which performs convolution operations and two fully connected layers. Decoder contains two fully connected layers, four hidden layers which try to reconstruct output image and one output layer. Adam optimizer is used to train the networks and learning rate is set to 0.01. Batch normalization technique is used to speed up the learning process [10]. The result of different epochs are shown below.

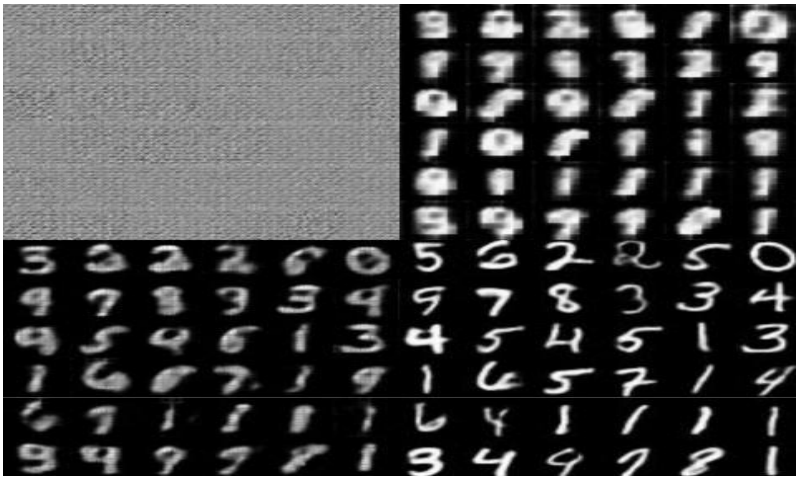

Fig 4: Reconstructed images via variational autoencoder

Fig 4 shows images are sampled at four different epoch (i) First epoch(ii) second epoch (iii)fifth epoch (iv) hundred epoch through variational autoencoder. As shown in the above figure, image quality is improve as number of epochs increases. If model is exposed to only KL divergence loss then information of the data or the class identity will be lost and because we have loss the class identity or the structural information of the data, the decoder does not know what to generate because every latent vector coming from all different category, they are same to the decoder. So, we have to use both KL divergence loss as well as reconstruction loss. KL divergence will try to make the distribution compact and reconstruction loss or data loss will try to maintain the class belongingness. It is very important to optimize both the loss in a model so that decoder can able to decode a sample perfectly and learn smooth data distribution. Disadvantage of the variational autoencoder is that images generated by VAE will be blur. Following fig shows the difference between autoencoder and variational autoencoder.

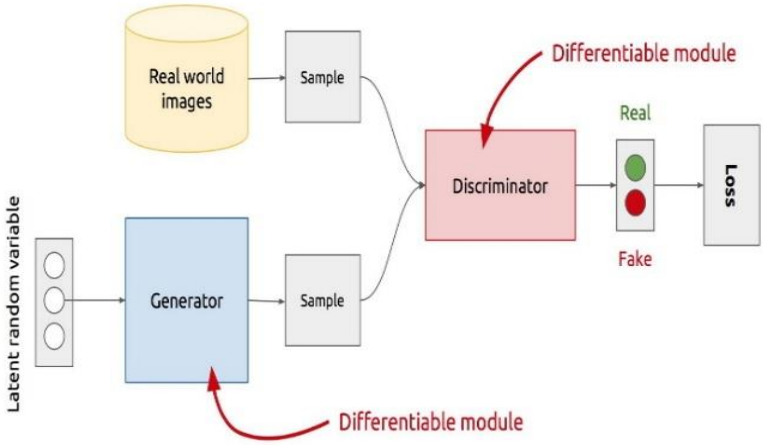

Fig 5: Architecture of Generative Adversarial Network.

Generative Adversarial Network (GAN)

Generative Adversarial Network (GAN) is another form of generative model for the reconstruction of the data from the latent code. Generative Adversarial Network are deep neural network architecture consist of two neural networks competing one against the other. Generative Adversarial Network consists of classes of network.one is generative network and other is discriminative network. Generative network generates some probability distribution which become close to the original data that we want to approximate. Discriminative Network tries to discriminate between fake sample (from generator) and real samples. Both of them learn distribution and this is implicit not the explicit unlike in case of VAE [11, 12].

Discriminator has the output node that distinguish real image from fake image. We feed noise vector in generator which produce fake image. We feed this fake image and real image into the discriminator to distinguish between fake image (image generated from generator) and real image (coming from real dataset) 13, 14]

Proposed Hybrid AE–VAE Based Generator Approach

We propose a theoretical framework that combines the strengths of Autoencoders (AEs), Variational Autoencoders (VAEs), and Generative Adversarial Networks (GANs) while mitigating their individual limitations (see Fig. 6). In this approach, the standard generator network (G) in a typical GAN is replaced with a hybrid generator incorporating both AE and VAE, whereas the discriminator (D) remains unchanged. The goal of this system is to generate image data by learning the underlying distribution of a given dataset, enabling the creation of new samples that reflect the learned distribution. In the proposed generator, both non-probabilistic and probabilistic latent codes are learned: the autoencoder captures the discontinuous, non-probabilistic latent representation, while the variational autoencoder models the continuous, probabilistic latent representation. After generating images with both AE and VAE, the Structural Similarity Index (SSIM) is calculated. If AE achieves higher SSIM with fewer epochs, it is selected for the generator; alternatively, if VAE produces higher SSIM in fewer epochs and with lower computational cost, VAE is used.

This hybrid generative model has potential applications in semantic image editing [11], data augmentation [12], and style transfer [13]. Additionally, the representations learned by the model can be used for tasks such as classification [14] and image retrieval [15]. As with conventional generative models, all network parameters are optimized via backpropagation. Similar efforts to combine VAEs and GANs have been proposed in the Adversarial Variational Bayes (AVB) framework. While AVB is conceptually similar, it replaces the Kullback–Leibler (KL) divergence with adversarial training in its objective function. In contrast, our model explicitly leverages both AE and VAE within the generator, capturing both non-probabilistic and probabilistic characteristics of the data.

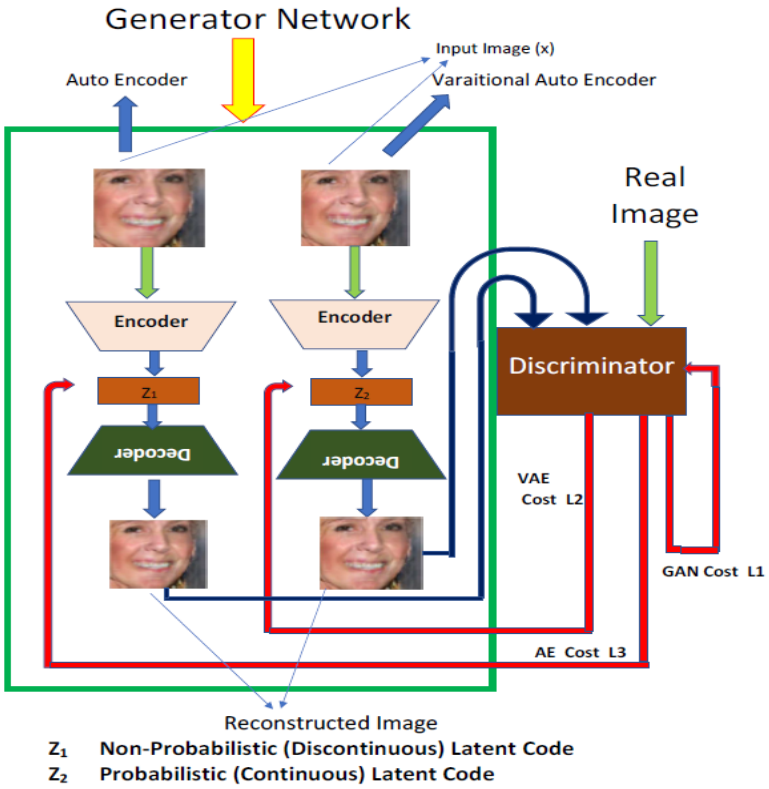

Fig 6: Architecture of proposed model.

AE–VAE Based Generator

Since our proposed theoretical model uses a variational Auto-encoder and autoencoder as the generator. From a broad perspective, variational Auto-encoder and auto-encoders consist of two primary parts, the Encoder and the Decoder. Each of these entities can be multidimensional, data sampled from a distribution x is fed into the encoder as input. The layers of an encoder decrease in dimensionality with each passing layer, essentially putting the input data through a funnel that deterministically maps it to a latent space z. The Decoder is the exact opposite, where the dimensionality increases with each layer, the decoder maps the data from the latent space back to the initial input as $\hat{x}$. These two mappings result in a reconstruction of the original input and the two networks are trained with the objective of producing reconstructed image that is as close as possible to the original. Training of both Autoencoder and Variational Autoencoder can be attained using backpropagation that can learn the parameters (weights and bias) of the encoder and decoder network that does map which results in minimum difference between the original and reconstructed data. We require variational inference to have a complete VAE. The primary objective in order to introduce variational inference to an Auto-encoder would be to derive a lower bound estimator. Implicit models often run into distributions that are often intractable and datasets that are too large and computationally expensive. We can consider that z is from some prior distribution p(z) and the variable x is from a distribution P(x|z).

AE–VAE Based Discriminator

The discriminator for our proposed model will be a typical image classifier, since our AE and VAE generator network will be trained to generate images. More than the discriminator itself, the key focus in this section will the adversarial approach we will take to train the VAE generator. Hence, we must begin with understanding how Generative Adversarial Networks work and what advantages or disadvantages do they bear. GANs work very well when implemented with deep neural networks. The whole system can be thought of as a pair of neural networks that is competing with each other. The discriminator network D(x), can be thought of a function that maps from image data (x) to a probability distribution which defines whether the image is from the real data distribution or the generator distribution either from AE or VAE. On the other hand, the generator distribution over data (x) is learnt with the generator network G(z).In this proposed architecture the generator network consists both autoencoder and variational autoencoder and generator network learn both non-probabilistic features and probabilistic features of the input data. The generator learns over data x, through a prior input noise variable p (z), where (z) is random noise from latent space. Here, the generator G is differentiable and its parameters can be trained with backpropagation. Similarly, discriminator D is trained with the objective of properly distinguishing between real and fake image, thus maximizing the probability of correctly labelling them. G is also trained concurrently with D, where G tries to minimize log (1-D(G(z))).

CONCLUSION

This study provides a comparative analysis of three popular models autoencoder, variational autoencoder

and generative adversarial network with the propose hybrid AE-VAE based generator in GAN.on basis of their objective, performance and architecture. Three models have their own pros and cons. Our architecture is still being developed. We are currently focusing on hyper-parameter tuning and improving the quality of the output of our architecture. It is proposed to develop advanced approach in which both non-probabilistic and probabilistic nature of data worked simultaneously in the generator network in order to generate high resolution images. Nonprobabilistic approach is utilized through autoencoder (AE) whereas, probabilistic approach is applied through variational autoencoder (VAE) and both of these approaches are applied in generator network.

REFERENCES

1. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial nets. In: Advances in Neural Information Processing Systems. 2014;2672–80.
2. Kingma DP, Welling M. Auto-encoding variational Bayes. Found Trends Mach Learn. 2019;12:307–92.
3. Belkin M, Niyogi P. Laplacian Eigenmaps for dimensionality reduction and data representation. Neural Comput. 2003;15:1373–96.
4. Vincent P, Larochelle H, Bengio Y, Manzagol PA. Extracting and composing robust features with denoising autoencoder. In: Proceedings of the 25th International Conference on Machine Learning; 2008 Jul; Helsinki, Finland.
5. Makhzani A, Frey B. K-Sparse Autoencoder. In: International Conference on Learning Representations (ICLR); 2014.
6. Baldi P. Autoencoder, unsupervised learning and deep architecture. JMLR: Workshop and Conference Proceedings. 2012;27:37–50.
7. Doersch C. Tutorial on variational autoencoder. arXiv:1606.05908 [stat.ML]. 2016 Aug 16.
8. Shiffman M. Under the hood of the variational autoencoder (in prose and code) [Internet]. 2016 Aug 22 [cited 2025 Oct 6]. Available from: http://blog.fastforwardlabs.com/2016/08/22/under-the-hood-of-the-variational-autoencoder-in.html
9. Weng L. From Autoencoder to Beta-VAE [Internet]. 2018 Aug 12 [cited 2025 Oct 6]. Available from: https://lilianweng.github.io/lil-log/2018/08/12/from-autoencoder-to-beta-vae.html
10. Chauhan JT. Comparative study of GAN and VAE. Int J Comput Appl. 2018;182.
11. Wang K, Gou C, Duan Y, Lin Y, Zheng X. Generative adversarial network: Introduction and outlook. IEEE/CAA J Autom Sin. 2017;4:588–98.
12. Goodfellow I. Generative adversarial networks. NIPS 2016 Tutorial.
13. Wang Z, Ward TE. Generative adversarial networks in computer vision. arXiv:1906.01529v3 [cs.LG]. 2020 Feb 3.
14. Xlangli Y, Deng Y, Loy CC. Real or not real, that is the question. arXiv:2002.05512v1 [cs.LG]. 2020 Feb 12.
15. Yue H, Sun X, Yang J, Wu F. Landmark image super-resolution by retrieving web images. IEEE Trans Image Process. 2013;22(12):4865–78.