algorithms

Article

GAGAN: Enhancing Image Generation Through Hybrid Optimization of Genetic Algorithms and Deep Convolutional Generative Adversarial Networks

Despoina Konstantopoulou [1,2], Paraskevi Zacharia [1,*], Michail Papoutsidakis [1], Helen C. Leligou [1] and Charalampos Patrikakis [2]

[1] Department of Industrial Design and Production Engineering, University of West Attica, Egaleo, 122 41 Athens, Greece; depi.konst@gmail.com (D.K.); mipapou@uniwa.gr (M.P.); e.leligkou@uniwa.gr (H.C.L.)
[2] Department of Electrical and Electronics Engineering, University of West Attica, Egaleo, 122 41 Athens, Greece; bpatr@uniwa.gr
* Correspondence: p.zacharia@uniwa.gr; Tel.: +30-210-538-1055

Abstract: Generative Adversarial Networks (GANs) are highly effective for generating realistic images, yet their training can be unstable due to challenges such as mode collapse and oscillatory convergence. In this paper, we propose a novel hybrid optimization method that integrates Genetic Algorithms (GAs) to improve the training process of Deep Convolutional GANs (DCGANs). Specifically, GAs are used to evolve the discriminator's weights, complementing the gradient-based learning typically employed in GANs. The proposed GAGAN model is trained on the CelebA dataset, using 2000 images, to generate 128 $\times$ 128 images, with the generator learning to produce realistic faces from random latent vectors. The discriminator, which classifies images as real or fake, is optimized not only through standard backpropagation, but also through a GA framework that evolves its weights via crossover, mutation, and selection processes. This hybrid method aims to enhance convergence stability and boost image quality by balancing local search from gradient-based methods with the global search capabilities of GAs. Experiments show that the proposed approach reduces generator loss and improves image fidelity, demonstrating that evolutionary algorithms can effectively complement deep learning techniques. This work opens new avenues for optimizing GAN training and enhancing performance in generative models.

Keywords: image generation; generative AI; genetic algorithms; deep convolutional generative adversarial networks; discriminator weight optimization

check for updates

Citation: Konstantopoulou, D.; Zacharia, P.; Papoutsidakis, M.; Leligou, H.C.; Patrikakis, C. GAGAN: Enhancing Image Generation Through Hybrid Optimization of Genetic Algorithms and Deep Convolutional Generative Adversarial Networks. *Algorithms* **2024**, *17*, 584. https://doi.org/10.3390/a17120584

Academic Editor: Arslan Munir

Received: 14 November 2024
Revised: 15 December 2024
Accepted: 18 December 2024
Published: 19 December 2024

1. Introduction

In recent years, advancements in machine learning and artificial intelligence (AI) have led to significant progress in generative models, which are used to create realistic synthetic data across various domains. Among these models, Generative Adversarial Networks (GANs) [1,2] have emerged as a cornerstone technology, particularly in applications requiring high-quality image synthesis [3], such as medical imaging [4], autonomous driving [5], and virtual media generation [6,7]. GANs consist of two competing neural networks—a generator that produces images and a discriminator that evaluates them. This adversarial setup has proven effective at refining the model's ability to create realistic outputs. However, despite the wide adoption of GANs, their training remains challenging. Issues such as mode collapse, unstable convergence, and sensitivity to hyperparameter selection complicate the generation of diverse, high-fidelity images.

One approach to address these limitations has been the development of Deep Convolutional GANs (DCGANs) [8], which incorporate Convolutional Neural Networks (CNNs) to enhance image quality by capturing complex spatial hierarchies in visual data. DCGANs have demonstrated success in tasks like facial recognition [9–11] and image synthesis [12–14], due to their capacity to model high-dimensional image data. Nonetheless, their reliance on gradient-based learning techniques leaves DCGANs vulnerable to issues like overfitting and local minima. Researchers have thus explored methods beyond traditional optimization, with evolutionary algorithms, particularly Genetic Algorithms (GAs), showing promise for enhancing GAN performance. Genetic Algorithms utilize evolutionary principles—such as selection, crossover, and mutation—to iteratively improve solutions, offering a global search capability that complements the local search tendencies of gradient-based methods. This evolutionary approach has demonstrated its potential for improving model robustness, optimizing network hyperparameters, and introducing greater variability in generated outputs.

Several approaches have been explored to address the inherent challenges in GAN training. The authors in [15] introduced Spectral Normalization to stabilize GAN training by constraining the Lipschitz constant of the discriminator, thereby reducing gradient issues that can lead to oscillatory or divergent training. Another approach, Wasserstein GAN (WGAN), introduced in [16], replaces the traditional GAN loss with the Wasserstein distance metric, offering a more stable objective function and improving convergence. While effective, these approaches do not completely resolve the problems of GAN training instability and mode collapse.

Evolutionary computation, inspired by biological processes, offers another promising avenue. Genetic Algorithms (GAs) [17–19] are optimization techniques that use principles of natural selection to refine solutions over successive generations. This approach, while traditionally applied to optimization problems, has been increasingly recognized for its potential in enhancing neural network training. GAs operate through selection, crossover, and mutation of "individuals" within a population, which in the GAN context can be applied to optimize model weights. The application of GAs to neural networks has shown potential for improving model robustness and convergence, particularly by augmenting standard gradient-based optimization techniques with global search capabilities that prevent premature convergence to local minima [20].

More recently, hybrid models integrating GAs with GANs have been proposed. The work in [21] has demonstrated that applying evolutionary algorithms can improve the exploration of solution spaces in GAN training. These approaches enhance model diversity and stability, addressing challenges such as mode collapse and unstable convergence. By fostering diversity among model candidates, evolutionary algorithms offer a broader search strategy that supports the development of more robust and adaptable GAN models. In [22], the authors proposed a structure tuning method for DCGANs using the Non-dominated Sorting Genetic Algorithm II (NSGA-II). Their approach optimizes both the network's architecture and hyperparameters, aiming to improve GAN performance and robustness. Experiments demonstrated that this approach enhanced the quality and accuracy of generated images.

In [23], the authors employed Genetic Algorithms to optimize GAN hyperparameters dynamically during training, achieving improved model stability and convergence. The results indicate that hybrid approaches, combining GAs and GANs, can improve training efficiency and robustness. This finding suggests that such integrated models effectively exploit the strengths of both GAs and GANs, offering the potential to develop more efficient generative models. The authors in [24] proposed the Evolution-GAN, which uses multiple generators trained with an evolutionary strategy to address the mode collapse problem common in standard GANs. By incorporating mutations and an additional classifier network, Evolution-GAN promotes generator diversity, leading to improved image quality and variety in synthetic outputs. Their experiments demonstrated superior performance over traditional GANs and DCGANs.

The authors in [25] propose a novel method for embedding hidden data into images while simultaneously generating adversarial samples. Using a combination of Genetic Algorithms and Least Significant Bit (LSB) encoding, the approach ensures high image quality and effective concealment of information. This work highlights the potential of combining generative modeling with data hiding techniques to enhance image security and privacy, illustrating the versatility of GAN-based frameworks in multimedia applications.

The work in [26] introduces a novel bi-level evolutionary framework aimed at optimizing both the generator and discriminator in GANs simultaneously, to achieve better training stability. By applying evolutionary strategies at two distinct levels, the method enhances model convergence and robustness. The main objective of this work is to address common GAN issues, such as mode collapse and unstable training dynamics, through the integration of evolutionary optimization. This aligns with the broader exploration of hybrid models, showcasing the potential of evolutionary algorithms to improve GAN performance in generating high-quality synthetic data.

The work in [27] proposes a method for reconstructing facial images from feature-based inputs by employing a GAN generator as a distribution constraint to ensure realistic outputs. By integrating Genetic Algorithms (GAs) to optimize the reconstruction process, the study demonstrates how evolutionary strategies can enhance the quality and realism of generated images. The experimental results demonstrate the effectiveness of this approach in producing high-quality facial reconstructions that closely align with the original input features.

The authors in [28] introduced a Multi-Scale Evolutionary GAN (MEvo-GAN) designed specifically for enhancing underwater imagery by integrating multi-scale convolutional layers with evolutionary algorithms. This approach effectively reduces common underwater image issues, such as low contrast and color distortion, while stabilizing GAN training through genetic selection. Despite these innovations, MEvo-GAN's design is specifically adapted to underwater conditions, highlighting the need for more generalized frameworks for broader applications in image enhancement and generation.

Recent advances in deep learning have demonstrated the versatility of neural networks in a wide range of applications, including medical imaging and data security [29,30]. While these studies focus on distinct domains, they share a common theme with GAN-based frameworks: optimizing learning through innovative techniques. Similarly, GAGAN combines Genetic Algorithms with GANs to enhance image generation by refining discriminator training, contributing to the exploration of hybrid models in deep learning. Notably, these advancements necessitates considerable computational resources, as demonstrated in real-world applications in medical image analysis and data augmentation. The use of advanced hardware is essential not only for the successful training of GAGAN but also for ensuring efficient, high-quality image generation in practical applications where performance and accuracy are critical.

Expanding on this research direction, the present study proposes a novel hybrid approach, termed GAGAN, which combines GAs with DCGANs to address key GAN training challenges. Specifically, the GAGAN framework utilizes GAs to optimize the discriminator's weights, enhancing its effectiveness at distinguishing real from generated images, thereby improving the generator's performance. By combining the strengths of evolutionary computation with deep learning, GAGAN aims to stabilize convergence, reduce generator loss, and ultimately produce higher-quality images.

Experimental evaluations on the CelebA dataset demonstrate that the GAGAN model outperforms traditional DCGANs in both image fidelity and stability of training. The findings suggest that incorporating evolutionary techniques into GAN training introduces a valuable complement to deep learning, paving the way for more efficient and resilient generative models. This study, therefore, contributes to the evolving field of generative AI by proposing a method that balances local and global search mechanisms, offering new insights into GAN optimization and the potential of hybrid approaches in machine learning. The innovation of this work is summarized around the following key points:

1. Hybrid Approach: This study integrates Genetic Algorithms (GAs) with Deep Convolutional GANs (DCGANs), establishing a novel hybrid model that utilizes the global search capabilities of GAs alongside traditional gradient-based optimization.

2. Discriminator Optimization: The primary innovation lies in using GAs to optimize the weights of the discriminator, enhancing its ability to differentiate between real and synthetic images, which improves the generator's feedback loop.
3. Enhanced Training Stability: By optimizing the discriminator's weights using evolutionary processes, the proposed model improves training stability compared to traditional GAN methods, reducing issues such as mode collapse and oscillatory convergence common in GAN training.
4. Improved Image Fidelity: Through experiments on the CelebA dataset, the GAGAN framework demonstrates enhanced image quality and realism compared to traditional DCGAN models, producing outputs with better texture, clarity, and variety.

The paper is organized as follows: Section 2 provides a concise overview of the kinematics of GA and GAN architectures, essential for understanding the subsequent development of the proposed approach. Section 3 introduces the integration of GAs with GANs, crucial for establishing a hybrid model. Section 4 presents the experimental setup and the results after evaluating the effectiveness of GAGAN in image generation based on a dataset including 2000 images. Finally, conclusions and directions for future advancements are presented in Section 5.

2. An Overview of GA and GAN Architectures

2.1. Genetic Algorithms

Genetic Algorithms (GAs) are adaptive heuristic search methods inspired by the principles of genetics and natural selection, specifically designed to address complex optimization and search problems. By simulating evolutionary processes, GAs iteratively refine a population of candidate solutions, referred to as individuals or chromosomes, toward an optimal or near-optimal solution over successive generations.

The core mechanism of GAs lies in their ability to mimic the process of natural evolution, progressively improving solution quality through a structured sequence of selection, crossover, and mutation operations. At each generation, a population of potential solutions, representing a diverse sample of the solution space, is evaluated using a fitness function—a critical component that quantifies each solution's effectiveness in solving the target problem. High-fitness solutions are more likely to be selected as parents, ensuring that favorable traits are propagated into the next generation.

The crossover process recombines genetic material from two parent solutions to create offspring, promoting diversity and exploration within the search space. This diversity is essential for thorough exploration, and minimizes the likelihood of the algorithm becoming trapped in local optima. Additionally, mutation introduces small random alterations to individual solutions, preserving genetic diversity and preventing premature convergence toward suboptimal solutions.

The iterative process of selection, crossover, and mutation continues until a predefined termination condition is met, such as reaching a specified number of generations or achieving an optimal solution, as shown in Figure 1. GAs are particularly advantageous in complex, multi-dimensional optimization problems where traditional approaches may face challenges. Their inherent flexibility, ability to perform a global search, and capacity to handle noisy or discontinuous objective functions make GAs an effective tool for solving challenging optimization problems across various domains.

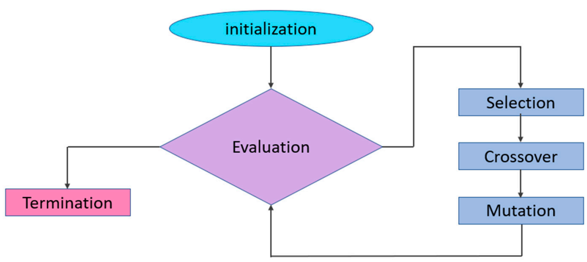

Figure 1. Genetic Algorithm phases.

2.2. Generative Adversarial Networks and Deep Convolutional GANs

Generative Adversarial Networks (GANs) [31] are a prominent class of machine learning models designed to generate realistic synthetic data. GANs operate within an adversarial framework in which two neural networks, the generator and the discriminator, participate in a continuous competition. This adversarial process, structured as a zero-sum game, encourages both networks to iteratively refine their capabilities: the generator learns to create data that increasingly resemble true samples, while the discriminator enhances its ability to distinguish synthetic data from genuine examples. This dynamic interaction is fundamental to GANs, facilitating significant advancements in data realism and fidelity with each training iteration.

At the core of a GAN is its dual-network architecture. The generator synthesizes new data samples from random noise, aiming to produce outputs indistinguishable from the real dataset. Concurrently, the discriminator evaluates these samples, assigning a likelihood score based on how closely they resemble actual data. As training progresses, the generator's outputs improve in authenticity, while the discriminator adapts to increasingly challenging inputs. This interplay between generation and evaluation is fundamental to GANs' effectiveness across a variety of applications, including image synthesis, style transfer, and data augmentation. The GAN architecture is illustrated in Figure 2.

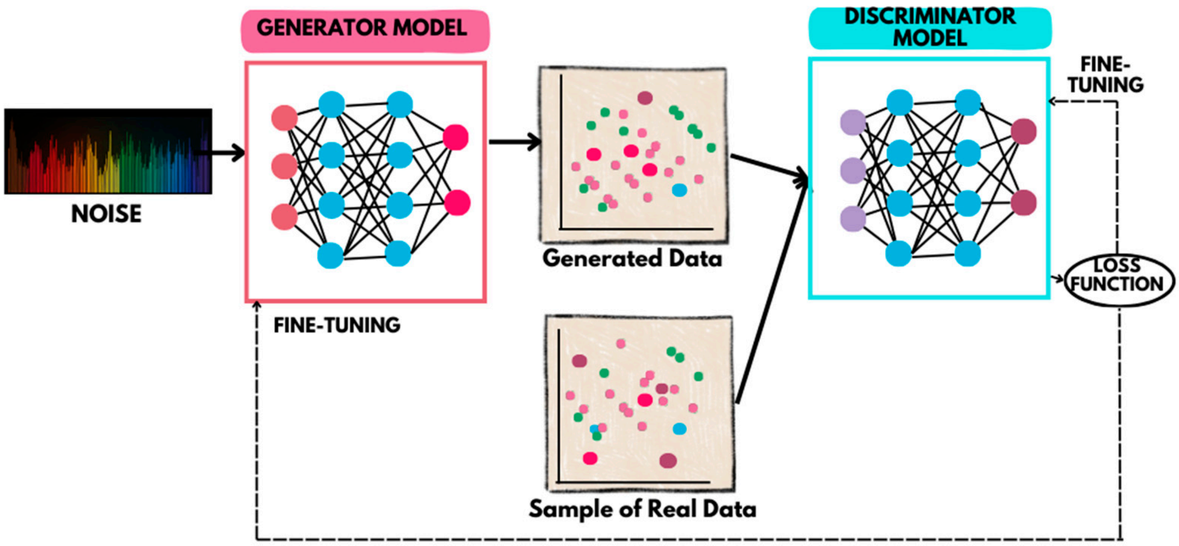

Figure 2. General overview of the GAN architecture workflow.

A significant advancement in the evolution of GANs was the introduction of Convolutional Neural Networks (CNNs) into the framework, as demonstrated by the development of Deep Convolutional GANs (DCGANs). While the original GAN framework introduced in [1] was not inherently convolutional, the adaptation of CNNs, particularly in DCGANs [14], has proven especially beneficial for image-based applications. CNNs are specialized for processing grid-like data structures, such as images, using convolutional layers to automatically learn hierarchical features. This makes them particularly well-suited for capturing the spatial hierarchies essential to high-quality image generation.

By replacing the fully connected layers in traditional GAN architectures with convolutional layers in both the generator and discriminator, DCGANs significantly enhance performance in visual data processing. This architectural shift allows DCGANs to capture detailed features such as edges, textures, and shapes with greater precision, thereby improving the fidelity and resolution of generated images. To illustrate the DCGAN workflow, Figure 3 presents the structure of the generator, while Figure 4 depicts the discriminator.

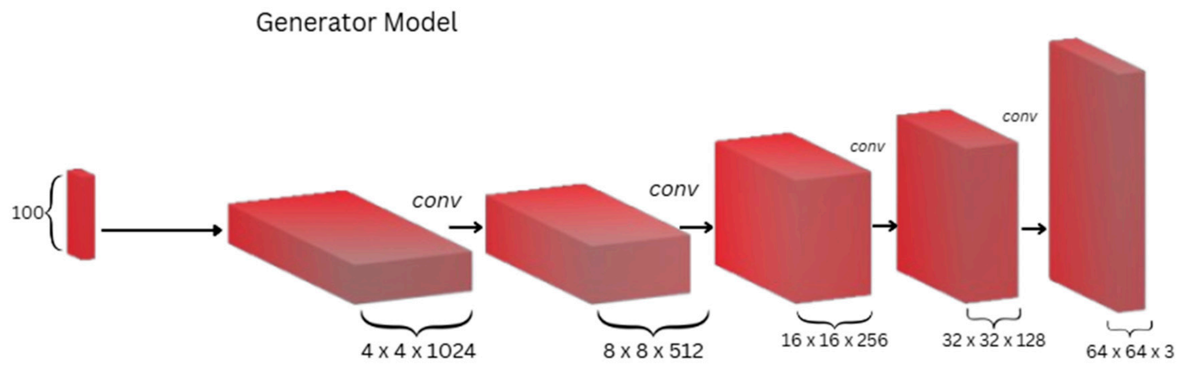

Figure 3. DCGAN generator architecture.

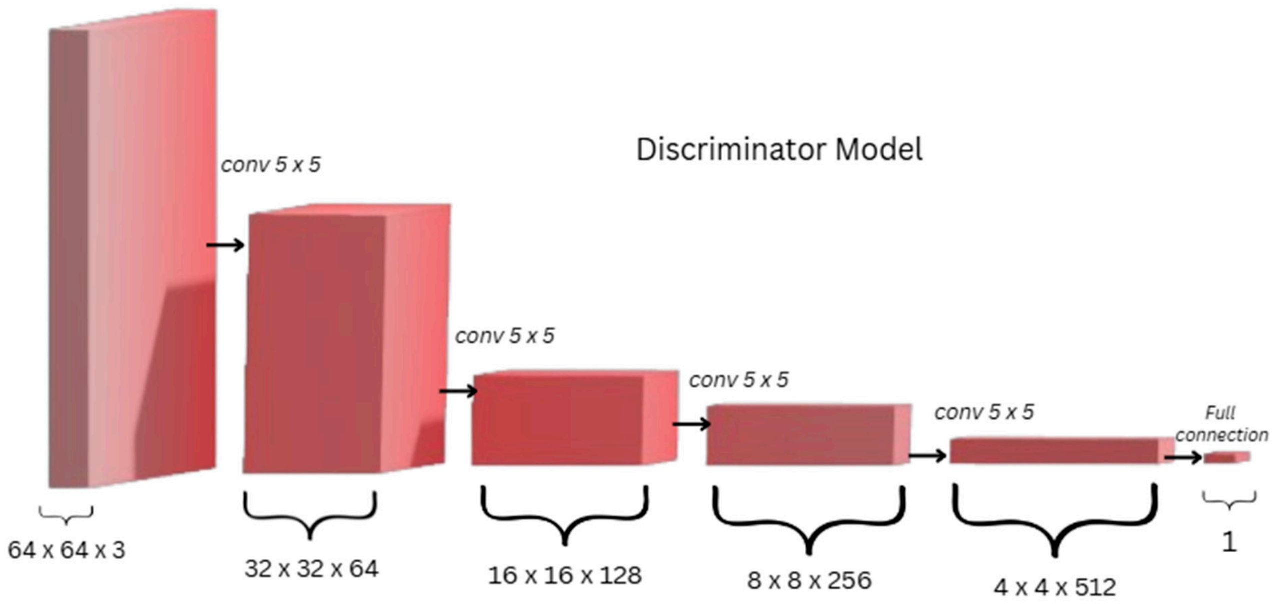

Figure 4. DCGAN discriminator architecture.

More specifically, the DCGAN generator architecture in Figure 3 starts with a latent vector of size 100, which is progressively transformed through several layers to produce a final 64 × 64 image with 3 channels. Initially, the input is reshaped to 4 × 4 × 1024, followed by convolutional layers that unsampled the tensor to 8 × 8 × 512, 16 × 16 × 256, 32 × 32 × 128, and finally to 64 × 64 × 3. On the other hand, the DCGAN discriminator architecture in Figure 4 takes a 64 × 64 image as input and applies a series of convolutional layers to downscale it. The image size is reduced through layers of 32 × 32 × 64, 16 × 16 × 128, and 8 × 8 × 256, before a fully connected layer outputs a single classification score, determining whether the image is real or fake.

The use of CNNs in DCGANs results in more stable training and enables the generation of higher-quality images, making DCGANs particularly effective for tasks like image synthesis, facial recognition, and other applications requiring high-resolution, photorealistic outputs. The convolutional layers in DCGANs also contribute to improved image generation and recognition, enabling the model to learn complex visual patterns with enhanced fidelity and stability compared to traditional GANs. The importance of convolutional architectures within GAN-based image generation thus paves the way for further innovations in this area.

Transitioning from DCGANs to GAGANs, the foundational structure of DCGANs serves as a critical basis for further advancements in image generation. By building on DCGAN's capability to generate realistic images, GAGAN introduces Genetic Algorithms to the framework, enhancing the model's performance through evolutionary optimization. Specifically, GAGAN optimizes the discriminator's ability to distinguish between real and generated images, combining the strengths of both deep learning and evolutionary algorithms. This combination allows GAGAN to push the boundaries of generative modeling, combining DCGAN's convolutional capabilities with the adaptive search strategies of Genetic Algorithms.

3. The GAGAN Framework: Evolutionary Optimization in GAN Architectures

3.1. Insights into GAGAN

The integration of GAs with GANs represents a significant advancement in enhancing the performance and robustness of generative models, particularly in addressing challenges such as mode collapse and training instability. GANs, composed of two adversarial neural networks—the generator and the discriminator—have demonstrated impressive capabilities in producing realistic synthetic data, especially for applications in image generation. However, conventional GAN training often suffers from issues related to unstable convergence and limited diversity in generated outputs, which can hinder model effectiveness.

The GAGAN framework is developed to overcome these limitations by utilizing GAs to optimize the discriminator's weights. Given the discriminator's crucial role in distinguishing real from synthetic data, optimizing its performance directly impacts overall model quality. In this hybrid framework, the discriminator's weights are treated as individuals (chromosomes) within a genetic population, allowing GAs to apply selection, crossover, and mutation operations. This process promotes diversity and introduces a global search mechanism, preventing the model from converging prematurely to suboptimal solutions and encouraging the generator to capture a wider range of features in its outputs. As a result, the model gains the ability to escape local minima and avoids generating repetitive patterns, enhancing the discriminator's accuracy and adaptability in differentiating between real and generated images.

This GAGAN integration yields multiple benefits: first, it stabilizes convergence by guiding the discriminator towards more optimal solutions, thereby providing the generator with reliable, consistent feedback throughout training. Second, the evolutionary diversity fostered by GAs effectively reduces mode collapse, promoting both variability and realism in generated images. Through this dynamic synergy, the GAGAN framework successfully combines the global optimization capabilities of Genetic Algorithms with the robust image generation structure of DCGANs.

In this sophisticated hybrid approach, the GA is responsible for evolving and optimizing the discriminator's weights, while the DCGAN architecture provides a reliable base for generating high-quality images. By optimizing the discriminator through evolutionary processes, GAGAN improves feedback to the generator, enabling it to produce more realistic and diverse images while avoiding issues like mode collapse. This synergy ensures that the model benefits from both the fine-grained feature extraction of DCGANs and the adaptive, broad-search capabilities of GAs, establishing a powerful and resilient model for high-quality synthetic data generation.

Figure 5 illustrates the flow diagram for the parallel processes within GAN and GA components, highlighting their interplay during training. The process begins with the GA component, where a population of discriminator weights is initialized and iteratively refined through evolutionary operations. Each set of weights undergoes fitness evaluation, followed by selection, crossover, and mutation to create new populations of increasingly effective weight configurations. Simultaneously, the GAN process operates with its traditional components: the generator creates synthetic images from random noise, while the discriminator evaluates both real and generated images, providing feedback to improve the generator's performance. The integration of these two systems occurs when the best-performing discriminator weights from the GA process are used to update the GAN's discriminator. This creates a continuous feedback loop, where the evolutionarily optimized discriminator enhances the overall GAN performance, leading to higher quality image generation while maintaining training stability.

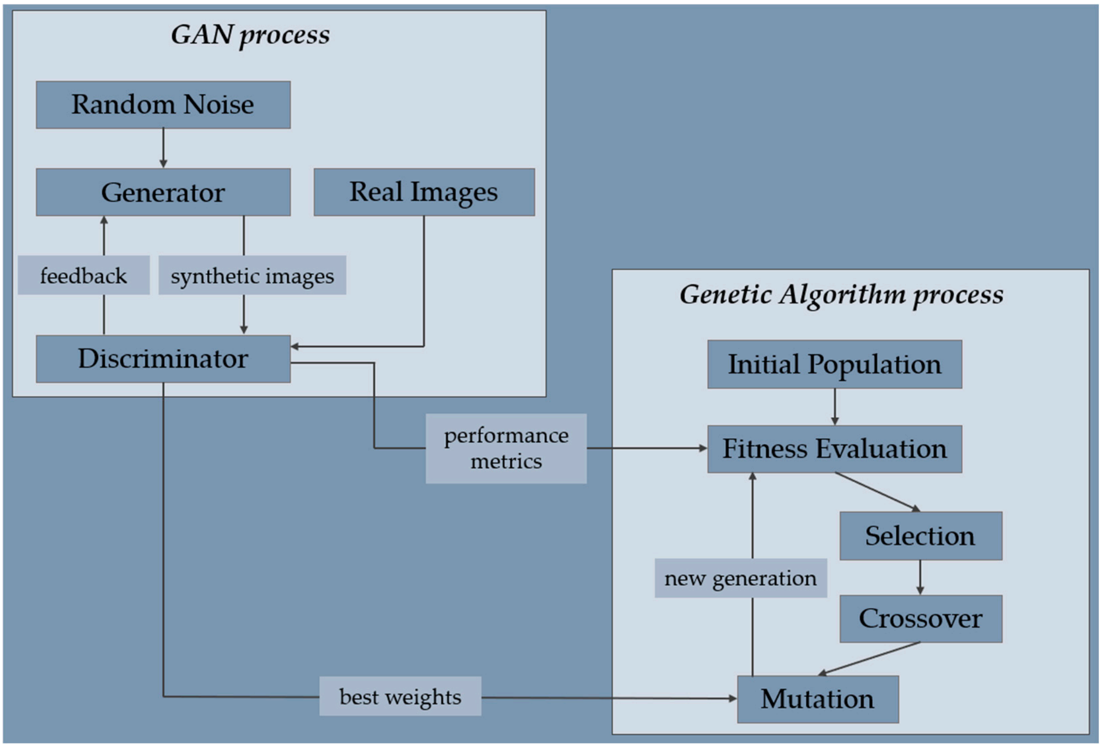

Figure 5. The flow diagram for the GAGAN training process.

Following this, Section 3.2 explores the specific mechanisms within the GAGAN framework, detailing how GAs and GANs are effectively combined to achieve superior generative performance.

3.2. Genetic Operators in GAGAN

In the GAGAN framework, genetic operators are vital for optimizing the performance of Generative Adversarial Networks (GANs). The primary operators—selection, crossover, and mutation—work together to create a diverse population of solutions.

3.2.1. Selection Process

The selection process determines which individuals from the current population will reproduce. Tournament selection is employed, where a subset of individuals is randomly chosen, and the one with the best fitness is selected to produce offspring. This method enhances robustness and promotes genetic diversity, crucial for effective search space exploration.

3.2.2. Crossover Process

Crossover is a genetic operator that combines the genetic information from two parent individuals to generate offspring potentially inheriting advantageous traits from both parents. In the GAGAN framework, a two-point crossover is employed, where two points along the weight vectors are randomly selected, and the segments between these points are swapped between the parents. This allows for the mixing of beneficial features from each parent, potentially creating offspring that outperform both parents individually.

The crossover rate determines the probability of two offspring undergoing this two-point crossover, and is set at 0.5. This indicates that 50% of parent pairs will undergo the crossover operation. The two-point crossover strategy facilitates the exploration of new solutions by integrating successful attributes from different regions, thereby increasing the diversity of potential solutions. This enhanced exploration improves the likelihood of generating more effective discriminator weights, which in turn contributes to the improved overall performance of the GAN and the generation of high-quality synthetic images.

In summary, the crossover rate controls the frequency of genetic exchange between parents to create new offspring, while the two-point crossover technique helps explore diverse solutions, enhancing GAGAN's ability to generate high-quality outputs.

3.2.3. Mutation Process

In the GAGAN framework, mutation introduces random changes to the weight vectors of the discriminator network, promoting genetic diversity and preventing premature convergence to local optima. The mutation operator is implemented using a Gaussian mutation strategy, where Gaussian noise is added to each weight in the network. This process alters the weights slightly, enabling the algorithm to explore different regions of the solution space, fostering the discovery of improved solutions.

The mutation rate in GAGAN indicates the probability of mutating each individual weight, and is set at 0.2. This rate strikes a balance between exploration (introducing new variations) and exploitation (fine-tuning existing solutions). The Gaussian mutation strategy ensures diversity within the population, thereby addressing the risk of the algorithm becoming trapped in suboptimal solutions and enhancing the overall optimization process.

In summary, the genetic operators—selection, crossover, and mutation—work together in GAGAN to optimize the weights of the discriminator. By employing tournament selection, two-point crossover, and Gaussian mutation, GAGAN efficiently navigates the solution space, improving the GAN's performance and enabling the generation of high-quality images.

3.2.4. Evaluation and Fitness Function

In the GAGAN framework, once selection, crossover, and mutation have been applied, an individual in the population undergoes an evaluation based on its effectiveness in distinguishing real images from generated ones. This evaluation is essential, guiding the genetic algorithm's optimization by providing a measure of performance for an individual.

The fitness function in GAGAN combines key performance metrics to evaluate the quality of each discriminator configuration. A primary component for this assessment is the discriminator loss, which indicates the model's proficiency in distinguishing real from generated images. Additionally, the fitness function incorporates the generator's loss, enabling the algorithm to gauge how effectively the generator deceives the discriminator. By balancing these elements, the fitness function prioritizes configurations that improve the discriminator's capacity to identify subtle differences between real and synthetic images, ultimately promoting higher-quality image generation.

After calculating the fitness scores, the GA identifies the highest-performing individuals, selecting them to proceed to the next generation. This continuous cycle of evaluation and selection sharpens the population, allowing GAGAN to explore new configurations and optimize the discriminator's performance over time, ultimately improving the GAN's overall quality and stability.

4. Experimental Evaluation of the GAGAN Model

The practical experiments within the GAGAN framework were designed to evaluate the effectiveness of integrating GAs with GANs, specifically to improve the discriminator's performance. The primary aim of this work is to address common challenges in GAN training, such as instability and mode collapse, by using GAs to optimize the discriminator's weights. This approach involves treating the discriminator's weights as individuals within a genetic population, allowing selection, crossover, and mutation to iteratively fine-tune the discriminator. By enhancing the discriminator's ability to distinguish between real and generated images, the GAGAN framework provides more accurate feedback to the generator, facilitating the production of higher-quality synthetic data.

4.1. Experiment Setup and Tests

The dataset employed in the experiments is the CelebFaces Attributes Dataset (CelebA), which contains over 200,000 celebrity images, each annotated with 40 attributes. It features diverse pose variations and background clutter, encompassing 10,177 unique identities, 202,599 face images, and 5 landmark locations. CelebA is suitable for tasks such as face attribute recognition, face detection, landmark localization, and face editing. Only the images were considered for GAGAN training. Specifically, 2000 random images were selected from the available 202,599, at a resolution of 128×128 pixels. This choice facilitated adequate visualization and manageable training time, significantly reducing the computational time compared to using the entire dataset.

Both models DCGAN and GAGAN were trained on a subset of 2000 images from the CelebA dataset for 50 epochs, using the same batch size, optimizer, and hyperparameters. This consistent setup, including identical epochs, datasets, and architecture, was designed to ensure that any differences in performance could be linked exclusively to the design and training strategies of the two models.

The proposed GAGAN model was developed and trained in Python with initial training conducted on Google Colab, utilizing a T4 GPU to evaluate its capacity for handling the model's requirements. The primary focus of the experimental setup was to optimize the discriminator's weights, as it plays a key role in distinguishing real images from generated ones. By fine-tuning the discriminator, we aimed to enhance the feedback loop to the generator, thus improving the quality and diversity of generated images, while addressing issues such as mode collapse.

However, this initial setup encountered challenges, including limited memory capacity and long training times that extended over several hours, even with a reduced dataset. These constraints highlighted the need for a more robust computational environment to support efficient model training and parameter optimization within the GAGAN framework.

To address these issues, we adopted an iterative optimization approach, adjusting various parameters such as dataset size, the number of epochs, image resolution, latent dimension, and learning rates for both the generator and discriminator. Despite these adjustments, the training process remained time-consuming, frequently taking between 3 and 7 h. To resolve this, we transitioned to an A100 GPU, enabling a more efficient training process and faster parameter adjustments. Ultimately, the final configuration was achieved by using a reduced dataset of 2000 images at a 128×128 resolution. With a smaller population size and fewer generations, we reduced training time to approximately one hour, achieving the target results without the complications observed with larger datasets.

4.2. Assessment of GAGAN Stages

Diversity metrics are crucial for evaluating and monitoring the variation within a population of individuals (each representing a potential solution or set of weights) at different stages of the genetic algorithm process. In this study, we measured two key diversity metrics throughout various phases of the genetic algorithm cycle. The Initial Population Diversity quantifies the variety among individuals in the starting population, helping ensure a broad base of solutions to increase the likelihood of optimization. The Final Population Diversity measures how diverse the population is at the end of a generation cycle, after genetic operations. It helps assess whether the population is prematurely converging on similar solutions or continuing to explore diverse options effectively. Table 1 provides the numerical values of these metrics across different training runs.

More specifically, Population Diversity is quantified using the Euclidean distance as the primary metric, which calculates the average pairwise distances between weight vectors in the population. This approach effectively captures the variation among individual solutions. The Initial Population Diversity measures the extent of variation at the start of the algorithm, ensuring a diverse range of potential solutions before genetic operations like crossover and mutation. The Final Population Diversity evaluates the variation remaining after these operations, indicating whether the algorithm maintains diversity or prematurely converges.

Both metrics critically reflect the genetic algorithm's ability to balance exploration and exploitation, which is essential for optimizing the GAN's performance.

Table 1. Diversity assessment through different trainings.

Training Run No.	Initial Population Diversity	Final Population Diversity	Diversity of the Population with the Best Individual
1	0.69109	0.5348	0.5359
2	0.67069	0.6109	0.5904
3	0.65262	1.3420	1.2684
4	0.67069	0.6109	0.5904
5	0.67539	0.8560	0.7979
6	0.68734	0.9153	0.8700

To illustrate the crossover operation during a specific training phase, Figure 6 presents the two-point crossover process, where two parent weight vectors (Parent #1 and Parent #2) exchange segments at randomly selected points. This exchange produces two offspring solutions, Offspring #1 and Offspring #2, which inherit segments from the respective opposite parent. Offspring #1 retains a section from Parent #2, while Offspring #2 inherits a portion from Parent #1, as highlighted in the subplots.

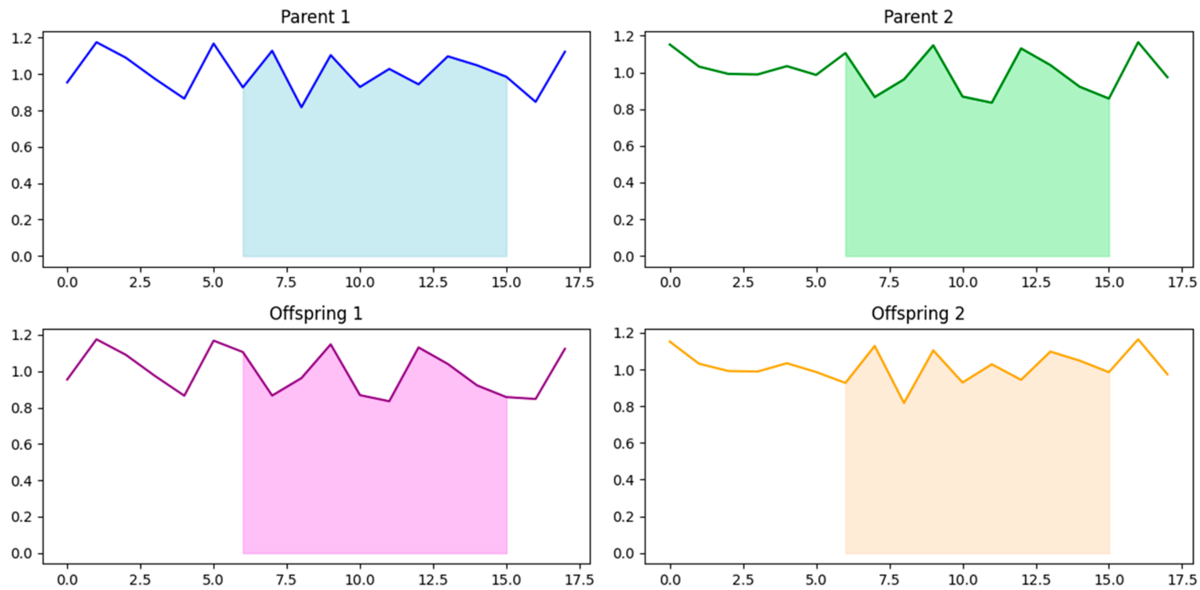

Figure 6. Two-Point crossover in GAGAN.

The upward and downward fluctuations in the lines represent variations in weight values across the vectors, with overlapping sections indicating the regions preserved from each parent. This process facilitates genetic exploration by introducing new combinations of features, promoting diversity within the population and ensuring broader exploration of the solution space.

Diversity metrics play a critical role in preventing premature convergence by maintaining sufficient variation in the population. The graphs appear immediately after evaluating diversity, indicating that the crossover operation occurs right after the initial population assessment. By actively enhancing the search space with varied offspring solutions, the two-point crossover process strengthens the model's robustness, helping GAGAN avoid local optima and achieve more effective training outcomes.

4.3. Numerical Analysis

In this section, we present a detailed analysis for numerical performance metrics of the DCGAN and GAGAN models. By examining the final losses for multiple iterations of both models, we aim to provide insights into their relative effectiveness in generating high-quality synthetic images.

In GAGAN, the discriminator loss measures how well the discriminator distinguishes real data from generated data, while the generator loss reflects the generator's ability to create realistic samples. These losses are competitive, with each model improving against the other in an adversarial cycle that enhances the realism of generated data and the performance of the GAN. Balancing these losses is key to optimizing the GAN's realism and guiding the GA in evolving effective discriminator parameters. The discriminator loss quantifies the discriminator's ability to classify real samples as "real" and generated samples as "fake", typically using binary cross-entropy to penalize misclassifications. The generator loss, also often based on binary cross-entropy, measures how well the generator fools the discriminator into misclassifying generated samples as real. These losses are competitive: as one model improves, it forces the other to adapt in an adversarial cycle, enhancing the generator's realism and discriminator's accuracy. Balancing these losses is critical for maximizing GAN realism and guiding the GA's optimization of effective discriminator parameters.

Figure 7 illustrates the generator loss (g_loss) and discriminator loss (d_loss) values of the GAGAN model across different training iterations. The plot highlights the convergence dynamics of both the generator and discriminator during training. The results indicate relatively stable d_loss values, suggesting consistent discriminator performance in differentiating real from generated images, while the g_loss values fluctuate more prominently, reflecting the generator's ongoing adjustments to improve the realism of synthetic outputs. This comparison provides insights into the model's training stability and the adversarial balance achieved by the GAGAN framework.

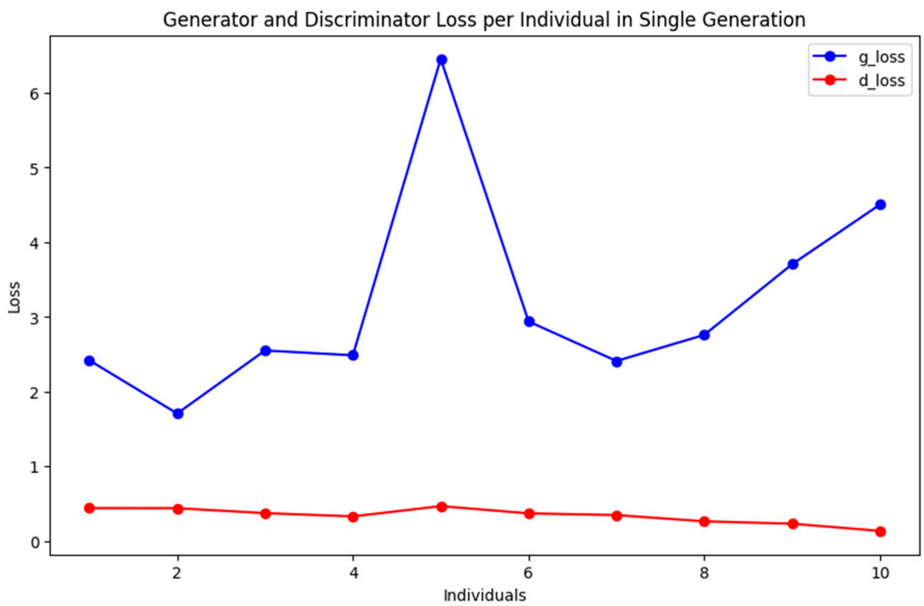

Figure 7. Comparison of generator and discriminator losses for GAGAN across individuals.

In the zero-sum game between the generator and discriminator in GANs, achieving equilibrium between their respective losses is crucial for optimal training. As illustrated in Figure 7, the d_loss in GAGAN remains remarkably stable throughout the training process, indicating its effectiveness in consistently distinguishing real from generated images with minimal fluctuation. In contrast, the g_loss fluctuates more prominently, reflecting the generator's continuous efforts to enhance the realism of its outputs. The notable difference in magnitude between the two losses—where d_loss is less than half of g_loss—demonstrates the enhanced stability achieved by GAGAN. While the generator continues refining its outputs, it is challenged by a more stable and well-optimized discriminator. This suggests that GAGAN maintains a balanced adversarial dynamic, with the discriminator providing stable feedback that guides the generator's learning without the instability often seen in traditional GANs. The stability of the discriminator loss highlights the improved training dynamics enabled by the integration of genetic algorithms.

To evaluate the performance of both DCGAN and GAGAN models, we present the recorded losses of the final results. These loss values serve as critical indicators of how well each model has optimized the generator and discriminator components. Table 2 displays the final recorded generator (g_loss) and discriminator (d_loss) losses of the DCGAN and GAGAN models at the last epoch, following 50 epochs of training. Each row corresponds to a specific run of the algorithm for each model (e.g., DCGAN/1, GAGAN/2), indicated in the Model/Training run No. column. The g_loss column reflects the generator's ability to produce realistic images that successfully deceive the discriminator, while the d_loss column shows the discriminator's effectiveness in distinguishing real images from generated ones. Lower values in these columns suggest improved model performance, with balanced losses indicating stable adversarial training.

Table 2. Loss comparison between DCGAN and GAGAN models.

Model/Training Run No.	g_loss	d_loss
DCGAN/1	3.0347	0.4647
DCGAN/2	2.5461	0.2854
DCGAN/3	2.1455	0.3585
DCGAN/4	2.8245	0.3433
DCGAN/5	2.1596	0.5967
GAGAN/1	6.4968	0.0676
GAGAN/2	8.0332	0.0786
GAGAN/3	7.6608	0.0635
GAGAN/4	7.6245	0.0693
GAGAN/5	5.8762	0.0665
GAGAN/6	8.6764	0.0273
GAGAN/7	5.6197	0.0852
GAGAN/8	8.1790	0.0704
GAGAN/9	4.7299	0.0855
GAGAN/10	8.1988	0.0761

To further assess model performance, the means and variances shown in Table 3 are derived from the data provided in Table 2. The computed statistic metrics for the DCGAN and GAGAN models provide a detailed evaluation of their performance across multiple training runs. For DCGAN, the discriminator loss has a mean value of 0.410, indicating its average performance, and a variance of 0.0121, reflecting relatively consistent behavior. The generator loss for DCGAN has a mean of 2.5420, with a variance of 0.1252, showing slightly greater variability in how well the generator adapts during training. In contrast, GAGAN's discriminator loss exhibits a significantly lower mean of 0.0690, coupled with a minimal variance of 0.00025, underscoring its stability and robust performance across runs. Meanwhile, the generator loss for GAGAN has a higher mean of 7.110 and a variance of 1.5986, indicating greater fluctuation in its adaptation process but demonstrating its overall effectiveness in generating diverse outputs.

These metrics are further illustrated in Figure 8, which displays the mean and variance of the generator and discriminator losses for both GAGAN and DCGAN, highlighting the stability of the discriminator in GAGAN and the more pronounced fluctuations in its generator loss across multiple training runs.

Table 3. Mean and variance of losses for DCGAN and GAGAN models.

Model	Metric	g_loss	d_loss
DCGAN	Mean	2.5420	0.4100
	Variance	0.1252	0.0121
GAGAN	Mean	7.1100	0.0690
	Variance	1.5986	0.0003

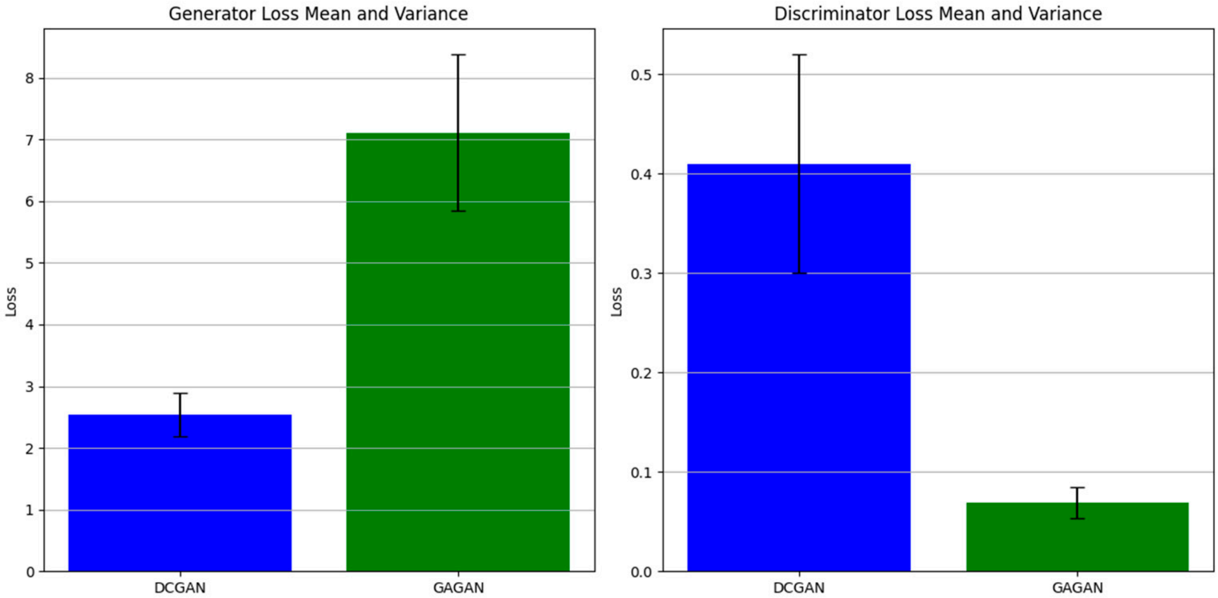

Figure 8. Comparison of generator and discriminator loss for DCGAN and GAGAN.

The comparative analysis reveals that GAGAN models consistently outperform DCGAN models in terms of discriminator loss (d_loss), indicating a more effective training of the discriminator. For instance, GAGAN/1 achieved a d_loss of 0.0676, which is significantly lower than DCGAN/1's d_loss of 0.4647. Similarly, GAGAN/6 and GAGAN/3 displayed strong performance, with d_loss values of 0.0273 and 0.0635, respectively, highlighting their superior ability to distinguish real from generated images. The near-zero d_loss values observed in GAGAN/6 and GAGAN/3 suggest a highly proficient discriminator, capable of accurately identifying real data. In contrast, DCGAN models exhibited consistently higher d_loss values, with DCGAN/5 reaching a peak of 0.5967, indicating a less effective discriminator in discerning between authentic and synthetic images.

In terms of generator loss (g_loss), GAGAN models showed varied performance, with some models excelling and others encountering challenges. For example, GAGAN/5 and GAGAN/9 achieved lower g_loss values of 5.8762 and 4.7299, respectively, indicating that these generators were more successful in producing images that deceived the discriminator effectively. However, higher g_loss values were observed in GAGAN/6 (8.6764) and GAGAN/8 (8.1790), reflecting occasional difficulties in generating convincing images. Interestingly, despite these higher g_loss values, the corresponding low d_loss values suggest that the discriminators in these cases were well-trained and presented a greater challenge for the generators to overcome.

On the other hand, DCGAN models exhibited relatively lower g_loss values across the board, with DCGAN/3 and DCGAN/2 achieving g_loss values of 2.1455 and 2.5461, respectively, indicating more effective generators within the DCGAN framework. However, the consistently higher d_loss values for DCGAN models imply a weaker overall training process for the discriminator. For instance, DCGAN/5's d_loss of 0.5967 points to a less efficient discriminator, unable to effectively differentiate between real and synthetic images.

Table 4 presents a detailed comparison between DCGAN and GAGAN, using additional metrics alongside statistical analysis through *p*-values and confidence intervals (CIs). One notable limitation of GAGAN is its higher generator loss compared to DCGAN, with a mean generator loss of 7.110 versus 2.542. This indicates that GAGAN's generator faces greater difficulty in deceiving the discriminator. However, this is a direct consequence of the optimized GAGAN discriminator. The discriminator's enhanced ability to distinguish real from generated images creates a more challenging adversarial environment for the generator, making it harder for the generator to achieve lower loss values.

Table 4. Statistical analysis of generator and discriminator losses for DCGAN and GAGAN.

Model	Metric	Mean	CI (95%)	*p*-Value
DCGAN	g_loss	2.5420	2.051–3.033	0.000
	d_loss	0.4100	0.257–0.562	0.003
GAGAN	g_loss	7.1100	6.156–8.063	0.000
	d_loss	0.0690	0.057–0.081	0.003

Conversely, GAGAN's discriminator demonstrates superior performance, with a significantly lower mean loss of 0.069 compared to 0.410 for DCGAN. This demonstrates the exceptional performance of GAGAN's discriminator in identifying real versus generated images. This improvement is a result of using the GA that optimizes the discriminator's architecture and configurations, allowing it to better detect fine-grained differences between real and generated data.

The statistical analysis supports these findings, with *p*-values for both generator and discriminator loss comparisons below 0.05, confirming that the observed differences are statistically significant. Furthermore, the 95% confidence intervals provide a range of plausible values for the true means, further reinforcing the reliability of these results. These metrics demonstrate both the challenges posed by the GAGAN's enhanced discriminator and the significant advantages of GA-based optimization in improving the model's overall performance.

To further evaluate the image quality of both GAGAN and DCGAN models, two widely adopted metrics were used: Frechet Inception Distance (FID) and Inception Score (IS). FID measures the distance between the distributions of real and generated images, with lower values indicating higher similarity and quality. Table 5 presents the FID, the IS mean and the IS standard deviation (STD) values obtained after 50 epochs of training for both models. The IS mean represents the average score across multiple runs, reflecting the overall quality and diversity, while the IS STD quantifies the variability in the score, indicating the consistency of the model's performance.

Table 5. FID and IS Metrics for GAGAN and DCGAN Models.

Model/Training Run No.	FID	IS Mean	IS STD
DCGAN/1	295.6075	2.5592	0.2140
DCGAN/2	295.3973	2.7832	0.1698
DCGAN/3	293.9751	3.0798	0.2913
DCGAN/4	308.9873	2.6243	0.3303
DCGAN/5	298.0760	3.4491	0.2243
GAGAN/1	120.4281	1.9632	0.1701
GAGAN/2	115.2125	2.032	0.1473
GAGAN/3	112.4268	2.0292	0.1986
GAGAN/4	113.0503	2.0144	0.2139
GAGAN/5	114.7395	2.0652	0.2098

The results indicate that GAGAN significantly outperforms DCGAN in terms of FID and IS metrics. FID, a widely adopted metric that quantifies the similarity between the distributions of real and generated images (with lower values indicating better performance), shows GAGAN achieving scores between 112.4 and 120.4. These values are substantially better compared to DCGAN's range of 293.9 to 308.9, underscoring GAGAN's superior capability in generating realistic images. While DCGAN achieves slightly higher IS means (ranging from 2.56 to 3.44, compared to GAGAN's range of 1.96 to 2.06), this could be attributed to a trade-off between image diversity and fidelity. The consistently lower FID scores for GAGAN suggest that its primary focus on producing highly realistic images may occasionally reduce diversity, as reflected in its IS values. Nevertheless, GAGAN's relatively stable IS performance and significantly better FID scores highlight its robustness and overall effectiveness as an advanced generative model for producing high-quality, realistic images.

The variances for FID and IS metrics, calculated from the scores presented in Table 5, are summarized in Table 6. These variances provide insight into the variability of the scores across multiple training runs, highlighting the consistency or instability in the models' performance.

Table 6. FID and IS Variances for DCGAN and GAGAN.

Model	FID Variance	IS Variance
DCGAN	29.7179	0.1079
GAGAN	7.9706	0.0110

Table 6 presents IS and FID for both DCGAN and GAGAN across multiple training runs, along with their corresponding variances. The variances provide insight into the model's performance consistency. For DCGAN, the relatively high IS variance (0.1079) and FID variance (29.7179) indicate significant variability in the quality and diversity of generated images across runs. In contrast, GAGAN shows much lower variances for both IS (0.0011) and FID (7.9706), demonstrating greater stability and consistency in performance across training sessions. These results emphasize GAGAN's ability to produce more reliable outputs, while DCGAN's performance shows greater fluctuation.

To further illustrate this variability, Figure 9 presents the results with error bars representing the standard error of the mean (SEM) for both IS and FID metrics. These error bars, directly related to the variances in Table 6, provide a clear representation of the uncertainty in the mean metric values. This representation underscores the reliability of the reported improvements, and highlights GAGAN's consistent performance across multiple runs.

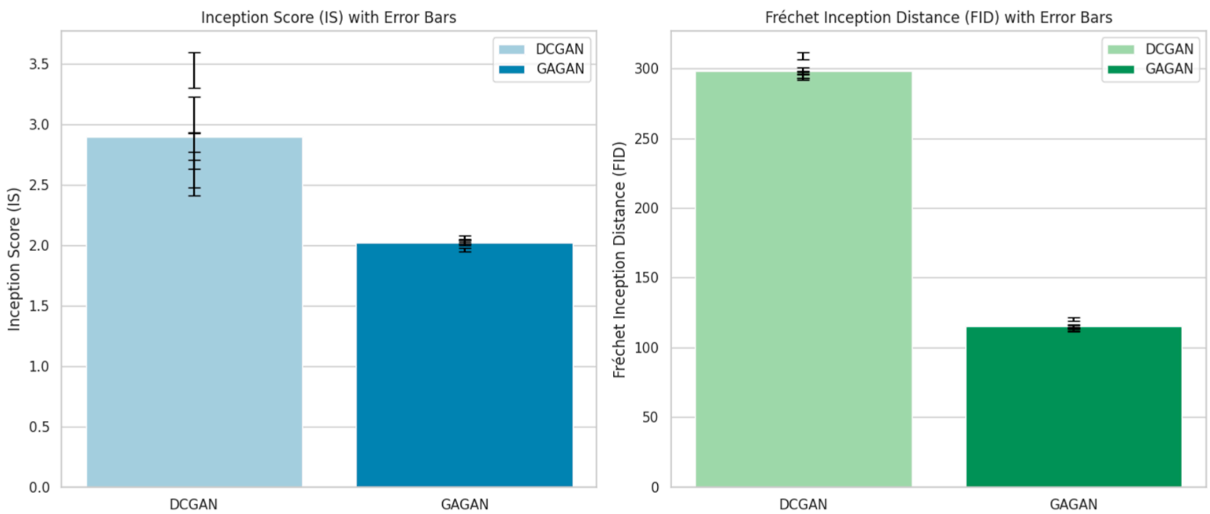

Figure 9. Error bars for the standard error of the mean for IS and FID metrics across training runs.

In summary, while DCGAN models tend to stabilize around lower generator losses, their higher discriminator losses highlight an imbalance in training both components effectively. GAGAN models, however, demonstrate strong discriminator performance, particularly in GAGAN/6 and GAGAN/3, which achieved the lowest d_loss values. Although GAGAN models display a broader range of g_loss values, reflecting variability in generator performance, they excel in training discriminators that are adept at identifying real versus generated data. This indicates GAGAN's overall superiority in achieving a balanced and effective adversarial training dynamic, leading to improved generative modeling capabilities.

Overall, the analysis underscores the effectiveness of the GAGAN framework in enhancing the training dynamics of generative models. The superior performance of GAGAN models in discriminator training, reflected by significantly lower d_loss values, demonstrates the successful integration of Genetic Algorithms to optimize the discriminator's weights. This evolutionary approach allows GAGAN to fine-tune the discriminator more effectively than traditional gradient-based optimization alone, thereby providing more accurate and consistent feedback to the generator. Consequently, even when some GAGAN models exhibit higher g_loss values, the improved discriminator performance helps maintain the adversarial balance, driving the generator to refine its outputs continuously.

In contrast, DCGAN models, despite showing stable g_loss values, tend to suffer from higher and less-stable discriminator losses. This suggests a limitation in their ability to train an effective discriminator, which may hinder the generation of high-quality, realistic images. The consistent gap in discriminator performance between the two approaches highlights the advantage of the GAGAN framework in achieving more robust training. By effectively combining the global search capabilities of Genetic Algorithms with the feature extraction power of DCGANs, GAGAN proves to be a more efficient and balanced method for generating high-quality synthetic data. This improved training dynamic ensures that GAGAN can better handle challenges like mode collapse and maintain a stable adversarial process throughout training, ultimately leading to higher fidelity and diversity in the generated images.

4.4. Visual Comparison of GAGAN and DCGAN Models

Alongside the numerical analysis, we conducted a detailed examination of the visual outputs generated by both DCGAN and GAGAN, presented in Figures 10 and 11, respectively, to further illustrate the qualitative differences in image quality despite comparable training parameters. The images produced by GAGAN consistently demonstrate superior detail, diversity, and realism. Specifically, the GAGAN outputs feature more defined textures and sharper edges, contributing to a more lifelike appearance across generated images. These qualities indicate that the genetic optimization applied to the discriminator in GAGAN enhances the generator's ability to produce higher-quality, more realistic outputs.

In contrast, the images generated by DCGAN tend to be less varied, and often display visible artifacts or inconsistencies, suggesting limitations in capturing complex details and achieving output stability. This is particularly evident when comparing similar image classes between the models, where GAGAN-generated images present a broader range of facial expressions and structural variety, while DCGAN's outputs are more prone to repetitive patterns and lack finer stylistic elements. The reduced artifacts and increased diversity in GAGAN's outputs highlight the success of the genetic algorithm in addressing common GAN challenges, such as mode collapse and artifact formation, by promoting greater diversity within the generator's outputs.

The qualitative differences observed between the two models reinforce the quantitative findings, suggesting that the evolution of discriminator weights through genetic algorithms provides significant advantages for image realism and diversity. This synergy between numerical and visual analyses underscores GAGAN's ability to outperform DCGAN by achieving a balance of detail, diversity, and stability, ultimately resulting in synthetic images that are both visually convincing and varied.

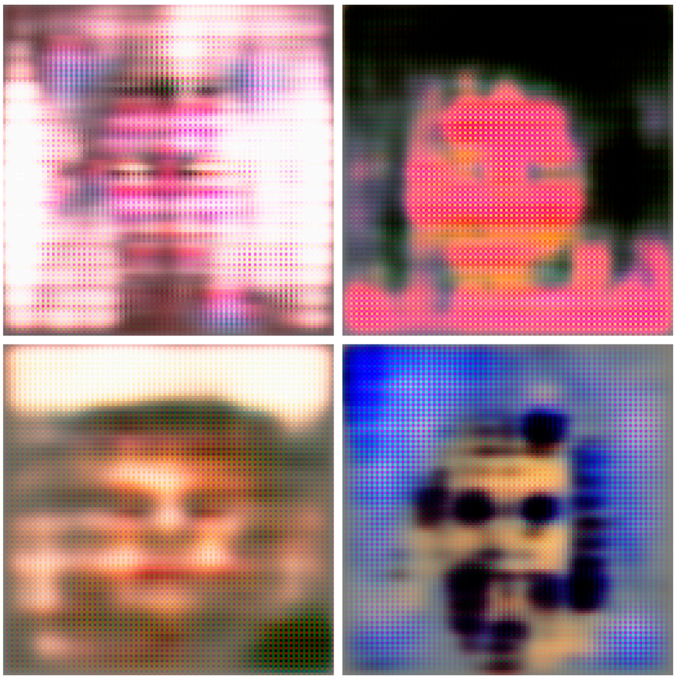

Figure 10. Generated images with DCGAN.

Figure 11. Generated images with GAGAN.

5. Conclusions

This study presented a novel hybrid approach, GAGAN, which integrates Genetic Algorithms (GAs) with Deep Convolutional GANs (DCGANs) to address common challenges in GAN training, such as mode collapse and unstable convergence. The primary innovation of GAGAN lies in the evolutionary optimization of the discriminator's weights, enhancing its ability to distinguish real images from generated ones and providing more effective feedback to the generator. This optimization process combines the local search capabilities of gradient-based methods with the global exploration offered by GAs, resulting in improved training stability and output quality.

The experimental evaluation, conducted using the CelebA dataset, demonstrated several advantages of the GAGAN framework over traditional DCGANs. GAGAN produced significantly better results in terms of FID, highlighting its ability to generate images that closely align with the distribution of real data. Although DCGAN achieved slightly higher IS means, GAGAN exhibited greater consistency, with lower variability in its scores, reflecting stable and reliable performance. These metrics confirm GAGAN's ability to generate realistic and high-quality images while maintaining robustness across multiple training runs.

The analysis of generator and discriminator losses further supports GAGAN's effectiveness. GAGAN consistently achieved lower discriminator losses than DCGAN, demonstrating its superior capacity to distinguish real from synthetic images. While some GAGAN configurations showed higher generator losses, reflecting challenges in deceiving the discriminator, others achieved strong performance, highlighting the framework's adaptability. Visual comparisons supported these findings, with GAGAN-generated images displaying enhanced textures, sharper details, and greater variability, while DCGAN outputs often exhibited repetitive patterns and artifacts.

The integration of GAs also enhanced the overall stability of the training process. Despite initial computational constraints, including long training times on Google Colab, the transition to a more robust computational setup using an A100 GPU facilitated faster convergence. By optimizing key parameters such as population size and the number of generations, the GAGAN framework achieved a balance between processing efficiency and image quality. This iterative fine-tuning process reduced training time while maintaining or enhancing the fidelity of the generated images.

These findings established a strong foundation for future research and improvement in GAGAN's design and application. Key areas for advancement include optimizing both the generator and discriminator simultaneously, to further refine their interactions, and developing more sophisticated genetic operators to enhance the evolutionary process. Additionally, incorporating perceptual metrics into the fitness evaluation could provide a more detailed measure of output quality, aligning the discriminator's optimization with human perceptual standards. By pursuing these enhancements, we anticipate that further advancements in both the efficiency and capability of GAGAN models will emerge, expanding the boundaries of generative modeling and setting new standards for quality and diversity in synthetic image generation.

Author Contributions: Conceptualization, D.K., P.Z. and C.P.; methodology, D.K., P.Z. and H.C.L.; validation, H.C.L. and C.P.; investigation, P.Z. and M.P.; data curation, D.K.; writing—original draft preparation, D.K., P.Z. and M.P.; writing—review and editing, D.K., M.P. and H.C.L.; supervision, P.Z. and C.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Original data presented in the study are openly available at [https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html] (accessed on 28 October 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; Volume 2, pp. 2672–2680.
2. Aggarwal, A.; Mittal, M.; Battineni, G. Generative adversarial network: An overview of theory and applications. *Int. J. Inf. Manag. Data Insights* **2021**, *1*, 10000. [CrossRef]
3. Shamsolmoali, P.; Zareapoor, M.; Granger, E.; Zhou, H.; Wang, R.; Emre Celebi, M.; Yang, J. Image synthesis with adversarial networks: A comprehensive survey and case studies. *Inf. Fusion* **2021**, *72*, 126–146. [CrossRef]
4. Skandarani, Y.; Jodoin, P.-M.; Lalande, A. GANs for Medical Image Synthesis: An Empirical Study. *J. Imaging* **2023**, *9*, 69. [CrossRef] [PubMed]
5. Lee, J.; Shiotsuka, D.; Nishimori, T.; Nakao, K.; Kamijo, S. GAN-Based LiDAR Translation between Sunny and Adverse Weather for Autonomous Driving and Driving Simulation. *Sensors* **2022**, *22*, 5287. [CrossRef] [PubMed]
6. Koh, A.J.H.; Tan, S.Y.; Nasrudin, M.F. A systematic literature review of generative adversarial networks (GANs) in 3D avatar reconstruction from 2D images. *Multimed. Tools Appl.* **2024**, *83*, 68813–68853. [CrossRef]
7. Aldausari, N.; Sowmya, A.; Mohammadi, G. *Video Generative Adversarial Networks: A Review*; ACM Computing Surveys: New York, NY, USA, 2022; Volume 55.
8. Radford, A.; Metz, L.; Chintala, S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv* **2015**, arXiv:1511.06434.
9. Wu, C.; Ju, B.; Wu, Y.; Xiong, N.N.; Zhang, S. WGAN-E: A Generative Adversarial Networks for Facial Feature Security. *Electronics* **2020**, *9*, 486. [CrossRef]
10. Toshpulatov, M.; Lee, W.; Lee, S. Generative adversarial networks and their application to 3D face generation: A survey. *Image Vis. Comput.* **2021**, *108*, 104119. [CrossRef]
11. Hwang, R.-H.; Lin, J.-Y.; Hsieh, S.-Y.; Lin, H.-Y.; Lin, C.-L. Adversarial Patch Attacks on Deep-Learning-Based Face Recognition Systems Using Generative Adversarial Networks. *Sensors* **2023**, *23*, 853. [CrossRef]
12. Kang, H.; Park, J.-S.; Cho, K.; Kang, D.-Y. Visual and Quantitative Evaluation of Amyloid Brain PET Image Synthesis with Generative Adversarial Network. *Appl. Sci.* **2020**, *10*, 2628. [CrossRef]
13. Ku, H.; Lee, M. TextControlGAN: Text-to-Image Synthesis with Controllable Generative Adversarial Networks. *Appl. Sci.* **2023**, *13*, 5098. [CrossRef]
14. Zhang, Y.; Han, S.; Zhang, Z.; Wang, J.; Bi, H. CF-GAN: Cross-domain feature fusion generative adversarial network for text-to-image synthesis. *Vis. Comput.* **2023**, *39*, 1283–1293. [CrossRef]
15. Miyato, T.; Kataoka, T.; Koyama, M.; Yoshida, Y. Spectral Normalization for Generative Adversarial Networks. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
16. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein Generative Adversarial Networks. In Proceedings of the 34th International Conference on Machine Learning, PMLR 70, Sydney, Australia, 6–11 August 2017.
17. Goldberg, D. *Genetic Algorithms in Search, Optimization and Machine Learning*; Addison-Wesley: Reading, MA, USA, 1989.
18. Holland, J.H. *Adaption in Natural and Artificial Systems*, 2nd ed.; University of Michigan Press: Ann Arbor, MI, USA; MIT Press: Cambridge, MA, USA, 1992.
19. Michalewitz, Z. *Genetic Algorithms + Data Structures = Evolution Programs*, 3rd ed.; Springer: Berlin/Heidelberg, Germany, 1996.
20. Stanley, K.O.; Miikkulainen, R. Evolving Neural Networks through Augmenting Topologies. *Evol. Comput.* **2002**, *10*, 99–127. [CrossRef] [PubMed]
21. Costa, V.; Lourenço, N.; Correia, J.; Machado, P. Exploring the Evolution of GANs through Quality Diversity. In Proceedings of the GECCO'20: Genetic and Evolutionary Computation Conference, Cancún, Mexico, 8–12 July 2020.
22. Du, L.; Cui, Z.; Wang, L.; Ma, J. Structure tuning method on deep convolutional generative adversarial network with nondominated sorting genetic algorithm II. *Concurr. Comput. Pract. Exp.* **2020**, *32*, e5688. [CrossRef]
23. Alarsan, F.I.; Younes, M. Best Selection of Generative Adversarial Networks Hyper-Parameters Using Genetic Algorithm. *SN Comput. Sci.* **2021**, *2*, 283. [CrossRef]
24. Liang, Y.; Han, Z.; Nie, X.; Ohkura, K. Improving generative adversarial network with multiple generators by evolutionary algorithms. *Artif. Life Robot.* **2022**, *27*, 761–769. [CrossRef]
25. Wang, D.; Li, M.; Zhang, Y. Adversarial Data Hiding in Digital Images. *Entropy* **2022**, *24*, 749. [CrossRef] [PubMed]
26. Liu, F.; Wang, H.; Zhang, J.; Fu, Z.; Zhou, A.; Qi, J.; Li, Z. EvoGAN: An evolutionary computation assisted GAN. *Neurocomputing* **2022**, *469*, 81–90. [CrossRef]
27. Dong, X.; Miao, Z.; Ma, L.; Shen, J.; Jin, Z.; Guo, Z.; Teoh, A.B.J. Reconstruct face from features based on genetic algorithm using GAN generator as a distribution constraint. *Comput. Secur.* **2023**, *125*, 103026. [CrossRef]
28. Fu, F.; Liu, P.; Shao, Z.; Xu, J.; Fang, M. MEvo-GAN: A Multi-Scale Evolutionary Generative Adversarial Network for Underwater Image Enhancement. *J. Mar. Sci. Eng.* **2024**, *12*, 1210. [CrossRef]
29. Marullo, G.; Ulrich, L.; Antonaci, F.G.; Audisio, A.; Aprato, A.; Massè, A.; Vezzetti, E. Classification of AO/OTA 31A/B femur fractures in X-ray images using YOLOv8 and advanced data augmentation techniques. *Bone Rep.* **2024**, *22*, 101801. [CrossRef] [PubMed]

30. Dhar, M.K.; Zhang, T.; Patel, Y.; Gopalakrishnan, S.; Yu, Z. FUSegNet: A deep convolutional neural network for foot ulcer segmentation. *Biomed. Signal Process. Control* **2024**, *92*, 106057. [CrossRef]
31. Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; Bharath, A.A. Generative Adversarial Networks: An Overview. *IEEE Signal Process. Mag.* **2018**, *35*, 53–65. [CrossRef]