*applied sciences*

MDPI

*Article*

# ICSO: A Novel Hybrid Evolutionary Approach with Crisscross and Perturbation Mechanisms for Optimizing Generative Adversarial Network Latent Space

**Zhihui Chen [1,2], Ting Lan [1,*], Zhanchuan Cai [1], Zonglin Liu [3] and Renzhang Chen [4]**

[1] School of Computer Science and Engineering, Macau University of Science and Technology, Macau 999078, China; 2202853gia30002@student.must.edu.mo (Z.C.); zccai@must.edu.mo (Z.C.)
[2] School of Information and Intelligent Engineering, Guangzhou Xinhua University, Guangzhou 523133, China
[3] LANGO Tech Co., Ltd., Guangzhou 510000, China; zonglin.liu@lango-tech.cn
[4] The Modern Educational Technology Center, Jinan University, Zhuhai 519070, China; jnulion@jnu.edu.cn
* Correspondence: tlan@must.edu.mo or lantingleo@gmail.com

**Abstract:** Hybrid evolutionary approaches have gained significant attention for solving complex optimization problems, but their potential for optimizing the low-dimensional latent space of generative adversarial networks (GANs) remains underexplored. This paper proposes a novel improved crisscross optimization (ICSO) algorithm, a hybrid evolutionary approach that integrates crisscross optimization and perturbation mechanisms to find the suitable latent vector. The ICSO algorithm treats the quality and diversity as separate objectives, balancing them through a normalization strategy, while a gradient regularization term (i.e., GP) is introduced into the discriminator's objective function to stabilize training and mitigate gradient-related issues. By combining the global and local search capabilities of particle swarm optimization (PSO) with the rapid convergence of crisscross optimization, ICSO efficiently explores and exploits the latent space. The extensive experiments demonstrate that ICSO outperforms state-of-the-art algorithms in optimizing the latent space of various classical GANs across multiple datasets. Furthermore, the practical applicability of ICSO is validated through its integration with StyleGAN3 for generating unmanned aerial vehicle (UAV) images, showcasing its effectiveness in real-world engineering applications. This work not only advances the field of GAN optimization but also provides a robust framework for applying hybrid evolutionary algorithms to complex generative modeling tasks.

**Keywords:** crisscross optimization; evolutionary optimization; generative adversarial network; latent space; particle swarm optimization

check for updates

Academic Editor: Rui Araújo

Received: 8 April 2025
Revised: 4 May 2025
Accepted: 6 May 2025
Published: 8 May 2025

**Citation:** Chen, Z.; Lan, T.; Cai, Z.; Liu, Z.; Chen, R. ICSO: A Novel Hybrid Evolutionary Approach with Crisscross and Perturbation Mechanisms for Optimizing Generative Adversarial Network Latent Space. *Appl. Sci.* **2025**, *15*, 5228. https://doi.org/10.3390/app15105228

**Copyright:** © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Hybrid evolutionary algorithms, by integrating the advantages of multiple search strategies, have proven to be robust and efficient in solving complex optimization problems. When applied to a generative adversarial network (GAN), they exhibit significant potential, achieving notable success in fields such as advanced prototyping [1], and enhancing robotic functionalities [2], optimizing logistics operations [3], and making improvements in predictive maintenance [4]. In recent years, with increasing attention on sustainable optimization studies, researchers have begun to explore how to incorporate sustainability principles into the integration of GANs and evolutionary algorithms, aiming to reduce computational resource consumption, improve model efficiency, and balance environmental responsibility and social impact while advancing technological progress. Despite their

promising prospects, practical applications of GANs based on evolutionary algorithms face challenges, including deformation and distortion during generation, and a lack of diversity in outcomes, which can appear monotonous. To overcome these issues, researchers have developed improved models like DCGAN [5], Wasserstein GAN [6], and StyleGAN [7], significantly enhancing the quality and diversity of outputs and thus increasing the realism and innovativeness of results.

GAN is one of the most important technologies which has made significant progress in the industry in recent years. At present, many researchers have studied GAN from different aspects: (1) objective function-based ones, (2) model structure-based ones, and (3) latent space-based ones. Objective function-based ones can improve the stability of adversarial training, reduce the problems of mode collapse and gradient vanishing during training, enhance the quality of generated images, and promote the convergence of models, e.g., the recurrent stacked GAN (RSGAN), the spectral normalized GAN (SNGAN), and the least squares GAN (LSGAN). However, the design of objective functions is often problem dependent, requiring different optimization strategies for different tasks and datasets. For model structure-based ones, since the model structure plays a decisive role in the performance of GAN, the expression ability of the generator and discriminator can be significantly enhanced by using complex model structure. Because the generator learns the real data distribution more effectively, it thus improves the quality of generated images. In addition, an excellent model structure also helps to alleviate problems such as gradient vanishing and mode collapse during training. However, the complex model structure causes the training process to become more computationally expensive. The optimization of the latent space through interpolation, transformation, and traversal can improve the robustness of model input and the controllability of generated sample. This helps to build more expressive latent space, which improves the quality and diversity of generated images.

Compared with the objective function-based ones and the model structure-based ones, the optimization of the latent space has the following advantages: By adjusting the latent vector with different dimensions or directions, a finer control over the generated samples can be achieved. At the same time, the latent space of GAN is completely independent of other components, such as objective function and model structure. The latent space of GAN have emerged as a pivotal research frontier, and optimizing this space through evolutionary algorithms presents notable advantages. By simulating the process of biological evolution, evolutionary algorithms systematically explore the latent space, transcending the constraints of random initialization. In contrast to gradient descent, which can often converge to local optima, yielding samples that may not be globally optimal, evolutionary algorithms delve deeply into the search space, identifying latent vectors that are closer to global optimality.

This paper focuses on the optimization of the latent space using evolutionary algorithms and then proposes a novel hybrid evolutionary algorithm, called the improved crisscross optimization (ICSO) algorithm, to optimize random latent vectors so as to enhance the quality and diversity of generated images. On the one hand, the original quality fitness and diversity fitness are balanced by normalization. Simultaneously, since the discriminator gradient has a certain influence on the diversity of generated images, a gradient penalty is introduced to constrain the discriminator gradient. On the other hand, based on the traditional crisscross optimization algorithm, ICSO is proposed by incorporating the local optimal solution and the global optimal solution from PSO. The contributions of this paper are summarized as follows:

1. Based on the original crisscross optimization, this paper introduces a fusion strategy of local and global optimal solutions from the particle swarm optimization, and proposes an improved evolutionary algorithm—improved crisscross optimization. This algorithm is specifically designed to optimize the latent space of generative adversarial networks, thereby enhancing the quality and diversity of generated images.
2. This study proposes the normalization to balance the quality and diversity of generated images, ensuring that both are effectively considered in the fitness. Meanwhile, it introduces a gradient penalty mechanism for the discriminator, which constrains the discriminator's gradients to enhance the stability and performance of the model during adversarial training.

The rest of this paper is organized as follows: Section 2 introduces the related work. Section 3 introduces ICSO in detail, including objective function and improved crossover optimization. In Section 4, a comparison of ICSO versus its competitors in terms of generated sample quality using mainstream architectures on the different datasets is made. Moreover, ICSO is compared with classical GANs to verify the diversity of ICSO. And ICSO is also applied to StyleGAN3, a mainstream network model, to validate its applicability in industry. The conclusion of this paper is presented in Section 5.

## 2. Related Work

### 2.1. Crisscross Optimization

CSO [8] is an evolutionary algorithm, which consists of horizontal crossover (HC) and vertical crossover (VC). HC refers to the selection of two particles from a population at random to perform the crossover operation. VC refers to the crossover operation between the dimensions of a particle, which is designed to help an individual break free from local optima.

Although CSO was proposed in 2014, compared with traditional evolutionary algorithms such as PSO, its fast convergence characteristics and global search capability have attracted many scholars to conduct continuous research. To address the challenge of accurate prediction in newly constructed wind farms due to a lack of historical data, Meng et al. [9] proposed a novel multi-gradient evolutionary deep learning neural network prediction model, incorporating time-series GAN and multivariate variational mode decomposition to capture dynamic time series correlations and generate high-quality samples. To enhance wind power prediction accuracy for newly built wind farms lacking sufficient historical data, Meng et al. [10] proposed a novel model based on secondary evolutionary generative adversarial networks and a dual-dimension attention mechanism-assisted bidirectional gate recurrent unit, aiming to generate high-quality realistic data and improve sensitivity to key input information.

### 2.2. Latent Space in GAN

The latent space refers to the space of low-dimensional representations that generators use, and points in this space are typically sampled from a random Gaussian distribution. By learning to approximate the real data distribution, the generator is able to transform the latent space into the data space, thus generating realistic samples.

The latent space is an important part of GAN. Liu et al. [11] interpolated the latent space and analyzed its advantages in spatial disentanglement, which can precisely manipulate single feature maps and solve the problem of "spatially entangled modification" caused by simple linear interpolation. Wang et al. [12] successfully solved the instability problem by introducing additional interference in the original Langevin Stein variational gradient descent (LSVGD). This method has the ability of an implicit regularization, which significantly improves the diversity of particles. Zhang et al. [13] determined whether

normality is abnormal in the latent space, implemented abnormal detection in videos, and effectively improved the performance of video anomaly detection and recognition. Wang et al. [14] built perturbations by manipulating single or multiple combinations of latent vectors, thereby purposefully manipulating the features of generated images.

## 3. Method

The paper proposes the ICSO algorithm, which is an enhanced version of the original CSO, primarily including population initialization, improved horizontal crossover, and vertical crossover.

### 3.1. Overall Framework of ICSO

The ICSO algorithm develops based on the original CSO, which consists of HC and VC. HC primarily manages information exchange among different individuals, enhancing their exploration capability through such external exchanges, aiding the population in finding better solutions. In contrast, VC focuses on interactions between different dimensions within an individual, leading to escaping local optima, thus achieving a refined and optimized performance. The original CSO faces challenges in search efficiency and achieving global optimality due to the randomness in its search direction. To address these problems, ICSO incorporates concepts of local and global optima from PSO, using information from optimal solutions as perturbation factors to guide the search process. This enhancement enables ICSO to maintain the original CSO's strong global search ability while more effectively exploring the search space by heuristically guiding the population towards known optimal positions.

The pseudocode and architecture of ICSO are shown in Algorithm 1 and Figure 1, respectively, which resembles traditional evolutionary algorithms in its process. Initially, the population is initialized with random Gaussian noise (Line 1 in Algorithm 1 and "Random Gaussian Noise" in Figure 1). Subsequently, the iterative evolution core revolves around "Improved HC", "VC", and "Selection based on *fitness*". In each iteration, the improved horizontal crossover is employed to compute the new noise for individuals, leveraging crossover operations between distinct individuals within the population to broaden the exploration of the latent space (Line 9 in Algorithm 1 and "Improved HC" in Figure 1). Concurrently, the vertical crossover performs a crossover operation across dimensions within the same individual, fostering integration and innovation among diverse features (Line 17 in Algorithm 1 and "VC" in Figure 1). Notably, regardless of the crossover operation applied, individuals undergo a renewed fitness evaluation, which subsequently updates both local and global optimal solutions, ensuring the population's continuous progression towards optimality (Line 3, 10, and 18 in Algorithm 1 and "Selection based on *fitness*" in Figure 1). Ultimately, the algorithm terminates autonomously upon reaching a predefined iteration limit or satisfying specific convergence criteria, outputting the meticulously selected optimal Gaussian noise (Line 27 in Algorithm 1 and "Optimal Gaussian Noise" in Figure 1).

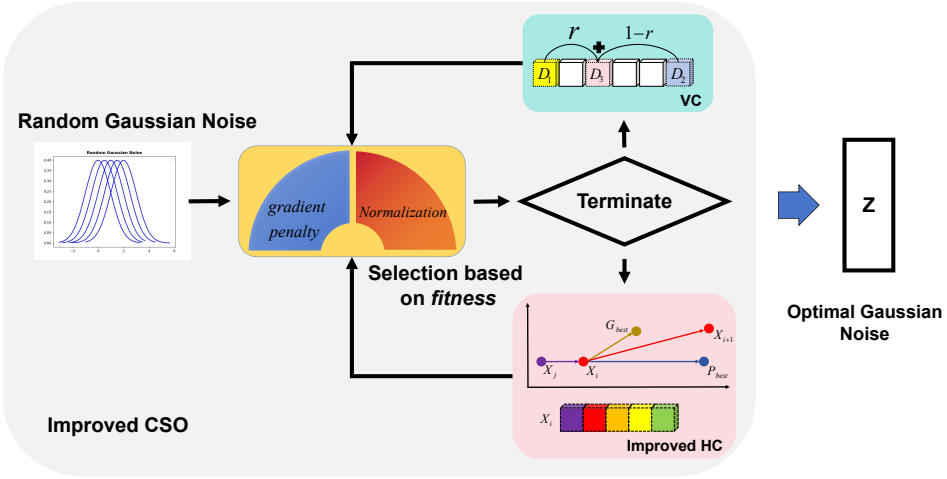



**Figure 1.** Architecture of ICSO. In the context of "Improved HC", $X_i$ and $X_j$ denote individuals that are formulated using Gaussian noise, whereas $G_{best}$ and $P_{best}$ represent the global and local optimal solutions, respectively, emulating principles from PSO. Within the framework of "VC", $D_1$, $D_2$, and $D_3$ indicate the dimensions of individuals across various dimensions. Concerning the "Selection based on *fitness*", the fitness is calculated through two components: the gradient penalty and normalization.

**Algorithm 1** The Overall Framework of ICSO

**Input:** The number of individuals $n$
**Output:** The optimal Gaussian noise **gbest**
**Procedure:**

Initialize the position $\mathbf{X}_i$ for each individual ($i = 1, 2, ..., n$) using Gaussian noise.
**for** $i = 1$ to $n$ **do**
    Calculate the fitness $\mathbf{F}_i$ which formulates as Equation (13)
    Set the personal best position $\mathbf{pbest}_i = \mathbf{X}_i$
**end for**
Set the global best position **gbest**
**for** $t = 1$ to $iter_{max}$ **do**
    **for** $i = 1$ to $n$ **do**
        $\mathbf{X}_{i+1}$ = Improved Horizontal Crossover($\mathbf{X}_i$) which formulates as Equation (14)
        $\mathbf{F}_{i+1}$ = Calculate Fitness($\mathbf{X}_{i+1}$) which formulates as Equation (13)
        **if** $\mathbf{F}_{i+1} < Fitness(\mathbf{pbest}_i)$ **then**
            Update personal best position $\mathbf{pbest}_{i+1} = \mathbf{X}_{i+1}$
        **end if**
        **if** $\mathbf{F}_{i+1} < Fitness(\mathbf{gbest})$ **then**
            Update global best position $\mathbf{gbest} = \mathbf{X}_{i+1}$
        **end if**
        $\mathbf{X}_{i+1}$ = Vertical Crossover($\mathbf{X}_i$) which formulates as Equation (15)
        $\mathbf{F}_{i+1}$ = Calculate Fitness($\mathbf{X}_{i+1}$) which formulates as Equation (13)
        **if** $\mathbf{F}_{i+1} < Fitness(\mathbf{pbest}_i)$ **then**
            Update personal best position $\mathbf{pbest}_{i+1} = \mathbf{X}_{i+1}$
        **end if**
        **if** $\mathbf{F}_{i+1} < Fitness(\mathbf{gbest})$ **then**
            Update global best position $\mathbf{gbest} = \mathbf{X}_{i+1}$
        **end if**
    **end for**
**end for**
Set the optimal Gaussian noise **gbest**

### 3.2. Problem Definition

The objective function of the GAN can be expressed as follows:

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

where $p_{\text{data}}(x)$ represents the distribution of real data; $p_z(z)$ denotes the prior distribution of the latent variable $z$. The Discriminator $D$ is responsible for distinguishing between real data and data generated by the generator. The Generator $G$ is designed to create data that closely mimic the distribution of real data.

The latent variable $z$ is randomly generated during the training process of GAN, which may lead to issues such as poor quality of generated samples and mode collapse. The main objective of optimizing the latent space is to find the optimal latent variable $z$, thereby improving the quality and diversity of the samples generated by the generator. When considering the use of evolutionary algorithms to optimize the latent variable $z$, we can define the fitness to guide the optimization of the randomly generated latent variable $z$, which can be expressed as follows:

$$z^* = \arg\min_{z\in\mathbb{R}^d} F(F_q, F_d, GP) \tag{2}$$

where $F_q$ and $F_d$ are evaluation functions for assessing the quality of generation and the diversity of generation, respectively. $GP$ is gradient-based regularization.

### 3.3. Population Initialization

Population initialization is a crucial initial step in ICSO, where it generates $n$ individuals to form the initial population $\mathcal{P}_0$:

$$\mathcal{P}_0 = \{X_1, X_2, \ldots, X_n\} \tag{3}$$

where each individual $X_i$ is a vector with a fixed dimensionality. A high-quality initialization not only increases the probability of the algorithm finding the global optimal solution but also effectively reduces the risk of being trapped in local optima. The latent space of GAN is often generated using the Gaussian distribution. When employing evolutionary algorithms to optimize this latent space, the formula involved in initializing individuals is as follows:

$$X = \mu + \sigma \cdot \eta \tag{4}$$

where $\mu$ represents the mean, which indicates the average value of a dataset, and its formula is as follows:

$$\mu = \int_{-\infty}^{\infty} x f(x) dx \tag{5}$$

where $f(x)$ denotes the probability density function. In Equation (4), $\sigma$ represents the standard deviation, which measures the dispersion of data points around the mean, and its formula is as follows:

$$\sigma = \sqrt{\int_{-\infty}^{\infty} (x-\mu)^2 f(x) dx} \tag{6}$$

and $\eta$ represents a random value drawn from the standard normal distribution $\mathcal{N}(0,1)$, and its probability density function is as follows:

$$f(\eta) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\eta^2} \tag{7}$$

Evaluation is a key operation in the evolutionary algorithm. By measuring the fitness of individuals, evaluation provides an effective evolutionary direction for a population. This paper approaches the problem from the perspective of multi-objective optimization, with a focus on two competing attributes of the generated samples: quality and diversity. On one hand, the generated samples need to exhibit high quality, meaning that they should be realistic enough to trick the discriminator. On the other hand, diversity must be ensured so that the distribution of generated samples is broad enough to prevent mode collapse. These two attributes often conflict in practice: an excessive focus on quality may cause the model to generate samples that concentrate on only a few modes, lacking diversity, while an excessive emphasis on diversity may compromise the overall quality of the generated samples.

The process of observing the average output value from the discriminator, when fed with images generated by the generator, can be considered a form of quality fitness function in the context of IE-GAN [15], as follows:

$$F_q = \mathbb{E}_{z \sim p_z}[D(G(z))] \tag{8}$$

where $D$ is the discriminator, $G$ is the generator, and $z$ is random Gaussian noise.

IE-GAN indirectly evaluates the distributional distance of the samples produced by the generator, so it is achieved by estimating the diversity through the mean absolute error (MAE) between samples. The formal definition of diversity fitness function is as follows:

$$F_d = \frac{1}{n_e} \sum_{i=1}^{n_e} \mathbb{E}_{z_1, z_i \sim p_z}[\|G(z_1) - G(z_i)\|_1] \tag{9}$$

where $n_e$ refers to the number of times that each sample is compared with other samples.

Generally, it is not appropriate to balance different objective functions only using Weighted Sum Approaches. $F_q$ and $F_d$ are two independent objective functions that sometimes differ by a factor of 10 or 100. Therefore, $F_q$ and $F_d$ of all individuals in the population are normalized, thus unifying the different fitnesses to an order of magnitude, wherein the min–max normalization is calculated by the following:

$$F'_q = \frac{F_q - \min(F_q)}{\max(F_q) - \min(F_q)}, \tag{10}$$

$$F'_d = \frac{F_d - \min(F_d)}{\max(F_d) - \min(F_d)} \tag{11}$$

where $\min(x)$ and $\max(x)$ are used to obtain the minimum and maximum values of the data, respectively.

Gradient-based regularization can stabilize the adversarial training of GAN and suppress the mode collapse of models [16]. The gradient penalty is a technology that applies constraints to the gradients of both the generator and the discriminator during training in order to make the training process more stable and smooth. This prevents gradient exploding or gradient vanishing between the generator and the discriminator, thereby improving the quality and diversity of the generated samples. Moreover, when the generator generates realistic samples, the discriminator does not reject the generated sample confidently (i.e., the discriminator updates with small gradient), and when the generator collapses to a small region, the discriminator subsequently labels collapsed points as fake with an obvious countermeasure (i.e., the discriminator updates with a big gradient).

By constraining the gradient of discriminator, the generated samples tend to be dispersed enough to avoid mode collapse, and the formula is as follows:

$$GP = \mathop{\mathbb{E}}_{\hat{\boldsymbol{x}} \sim \mathbb{P}_{\hat{\boldsymbol{x}}}} \left[ \left( \|\nabla_{\hat{\boldsymbol{x}}} D(\hat{\boldsymbol{x}})\|_2 - 1 \right)^2 \right] \tag{12}$$

where $\hat{x}$ is a sample drawn from the distribution $\mathbb{P}_{\hat{x}}$; $\left( \|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1 \right)^2$ calculates the squared difference between the L2 norm of the gradient of the discriminator $D$ with respect to $\hat{x}$ and 1. The goal is to keep the gradient norm close to 1, which helps in satisfying the Lipschitz condition.

A fitness function that integrates both the quality and diversity of generated images is designed and combined with GP to construct a multi-objective function, aiming to balance these two conflicting attributes. Normalization is applied to ensure fairness and consistency between the quality and diversity metrics, and the objective function is formulated in a minimization form:

$$\min_{z \sim p_z} F(z) = \min_{z \sim p_z} \Big( \lambda \cdot F'_q(z) + (1 - \lambda) \cdot F'_d(z) + \mu \cdot GP(z) \Big) \tag{13}$$

where $\lambda$ is a scalar parameter used to balance the contributions of $F'_q(z)$ and $F'_d(z)$. When $\lambda$ is close to 0, the model focuses more on $F'_d(z)$, while when $\lambda$ is close to 1, the model focuses more on $F'_q(z)$. $\mu$ is another scalar parameter used to adjust the weight of the gradient penalty term $GP(z)$. A smaller $F(z)$ indicates that the evaluated individual has a better generative performance.

### *3.4. Improved Crisscross Optimization*

ICSO, an integration of PSO and CSO, primarily consists of improved HC and VC. The original CSO effectively speeds up the convergence ability of ICSO by running the HC and VC operations alternately. In addition, ICSO introduces the local optimal search mechanism of PSO, which helps ICSO to quickly search local optimal solution in its own neighborhood. Similarly, ICSO incorporates the global optimal search mechanism of PSO, guiding toward the direction of population optimization through information interaction between particles. This approach can accelerate the convergence of the population and avoid falling into local optimal solutions [17].

HC randomly selects two particles from the population to perform crossover operations between the same dimensions, aiming to promote diversity and enhance search capability among particles through information exchange. Unlike the original CSO, this operator incorporates principles from the PSO, introducing the local optimal solutions of individuals and the global optimal solution from population into the original CSO (as shown in Equation (14)). The perturbation, which consists of the local and global optimal solutions, can make the particle heuristic search and help the operator search towards the global optimal solution. By integrating the rapid convergence ability of the original CSO with the heuristic search strategy of PSO, this method not only accelerates the convergence speed of the population but also effectively avoids being trapped in local optimal solutions, thereby enhancing the efficiency of global search. The formula for HC is as follows:

$$\begin{aligned} X_{i+1} = & X_i + r \cdot \left( X_j - X_i \right) \\ & + c_1 \cdot \left( pbest_i - X_i \right) + c_2 \cdot \left( gbest - X_i \right) \end{aligned} \tag{14}$$

where $r$ is the random number between $[0, 1]$; $c_1$ is the individual learning factor that controls the step size of the individual moving towards its $pbest$; $c_2$ is the social learning factor that controls the step size of the individual moving towards the $gbest$. In the standard implementation of PSO, both $c_1$ and $c_2$ are commonly set to 2; $X_i$ is the Gaussian noise of

generation $i$; $pbest_i$ is the local optimal solution and $gbest$ is the global optimal solution. The particles produced by the improved HC must be compared with their parent particles, and only the particles with better fitness can be retained in the VC operation.

Figure 2 provides the schematic of a particle's trajectory of evolution in the procedure of improved HC. The operation ① borrows from the original CSO. The main part of the search space is constructed with parent particles as the diagonal vertices of a hypercube. In particular, it is clear that to reduce unsearchable blind spots, the original CSO also searches the periphery of the hypercube with a lower probability. In this paper, however, we utilize the local optimal solutions and global optimal solutions from PSO as perturbations, replacing the aforementioned part of the original CSO. This search mechanism effectively leverages the individual experience of particles and the optimal experience of the population, heuristically guiding the population to search for blind spots on the periphery of the hypercube, thereby enhancing the global search capability of the operations (② and ③).

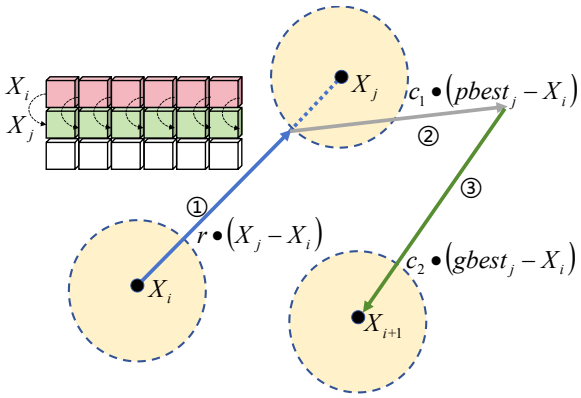



**Figure 2.** Procedure of improved HC. In the figure, $X_i$ and $X_j$ are two individuals generated through Gaussian noise. The diagram illustrates the process of generating offspring based on these two individuals, corresponding to the flowchart of Equation (14).

VC, which is an operation from the original CSO, is a crossover operation between two different dimensions of a particle, aimed at escaping stagnant dimensions to prevent particles from falling into local optimal solutions. The formula for VC is as follows:

$$D_{d3} = r \cdot D_{d1} + (1 - r) \cdot D_{d2} \tag{15}$$

where $r$ is the random number between $[0, 1]$ and $D_{di}$ is the $i$ dimension of the particle. As with HC, only the better particles are retained to the next iteration.

Since the number of iterations is directly proportional to the number of stagnant particles, the more iterations there are, the higher the value of $P$, which in turn determines the operating probability of the VC. The formula is as follows:

$$P = item / \max(item) \tag{16}$$

where $item$ is the current number of iterations and $\max(item)$ is the total number of iterations.

As shown in Figure 3, the process of VC is used to remove stagnant dimensions by performing crossover operation among different dimensions of an individual. Once a certain stagnant dimension of an individual escapes from a stuck value, it spreads rapidly through the whole population via the HC operation. Moreover, this also helps other stagnant dimensions to quickly escape local minima through the VC. It is the crisscross operation on both horizontal and vertical directions that makes ICSO have a unique global search ability for addressing multi-modal problems with many local minima.

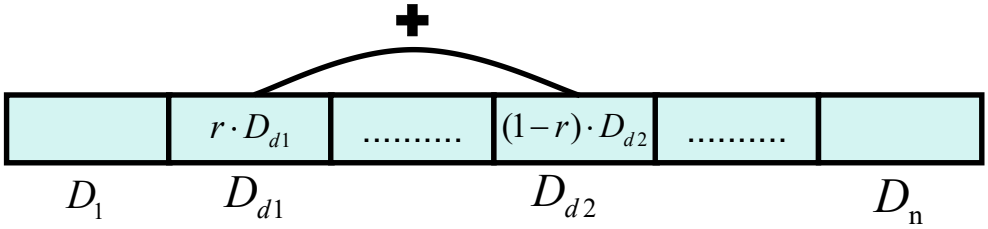



**Figure 3.** Procedure of VC. In the figure, $D_i$ represents the dimension of an individual. The diagram illustrates the process of generating new dimensions for an individual across two different dimensions according to Equation (15).

Each iteration requires checking if the current individual's new position $X_{i+1}$ is better than its historically best position $pbest_i$. If the objective function at the current position is smaller, then it is updated as follows:

$$pbest_{i+1} = \begin{cases} X_{i+1}, & \text{if } F(X_{i+1}) < F(pbest_i) \\ pbest_i, & \text{otherwise} \end{cases} \tag{17}$$

Similarly, the global best *gbest* is the best individual extremum among the entire population, representing the globally optimal solution within the current search space. During each iteration, it is necessary to find the optimal one among all individual' *gbest*; that is

$$gbest = \arg \min_{i=1,\ldots,n} F(pbest_i) \tag{18}$$

## 4. Experimental Results and Analysis

In this section, ICSO is compared with some state-of-the-art algorithms to validate the quality and diversity of the generated images. In addition, we have integrated ICSO into StyleGAN3 and applied it in real industrial scenarios.

### *4.1. Benchmark Dataset*

The CIFAR-10 dataset contains 10 classes with a total of 60,000 32 × 32 RGB images. The dataset contains the following classes: bird, cat, plane, boat, car, truck, dog, deer, horse, and frog, and each class contains 5000 training images and 1000 test images. The STL-10 dataset contains 10 classes with a total of 113,000 96 × 96 RGB images. The training set has 5000 images, the test set has 8000, and the remaining 100,000 are unlabeled images. The STL-10 dataset is unique because it is suitable for semi-supervised learning and self-supervised learning research. The LSUN is a large-scale image dataset, comprising 10 scene categories, such as bedrooms, living rooms, and classrooms, and 20 object categories, totaling approximately 1 million labeled images with 512 × 512 resolution. LSUN-64 refers to a subset with a resolution of 64 × 64 in the LSUN dataset. The CelebA is a large dataset of face attributes containing more than 200,000 64 × 64 RGB images. The CelebA dataset has been leveraged for a range of face-related computer vision tasks, including facial attribute recognition, facial expression analysis, and face generation. The CelebA-64 is a variant of CelebA in which the face of each image is centrally located and labeled with 40 attributes. The ImageNet is a large image dataset containing more than 14 million images, covering more than 20,000 categories. This dataset is usually used for classification, localization, and detection. It is extensively employed in research in the field of computer vision and related industrial applications.

### 4.2. Parameter Setting

The hyperparameters involved in ICSO can be divided into search strategy and GAN: For search strategies, the population size is 10, the maximum number of iterations is 10, and the dimension of the individual (i.e., the Gaussian noise) is 100. The horizontal probability is 1. For GAN, since the latent space focused on in this paper is completely independent from the other components of GAN, the model and parameter setting of GAN remain consistent with [18]. The testing environment is described as follows: PyTorch (version 1.13.1) and Nvidia GeForce GTX 4090 (Nvidia, Santa Clara, CA, USA).

### 4.3. Evaluation Metrics

In this study, we evaluate the performance of the generative models using the following metrics: Inception Scores (IS) [19], Fréchet Inception Distance (FID) [20], Density [21], and Coverage [21]. The IS measures the quality and diversity of generated images, with higher values indicating better performance. The FID quantifies the similarity between the distributions of real and generated images, where lower values correspond to higher generation quality. Density evaluates how well the model covers the data manifold without generating unrealistic samples, while Coverage assesses the proportion of the test data manifold that is captured by the generated samples. Additionally, we report the computational efficiency in terms of GPU training days required to reach convergence, expressed in GPU/days.

### 4.4. Generative Performance

As shown in Table 1, the experiments systematically evaluate the outstanding performance of ICSO in optimizing the latent space through comprehensive comparisons with random Gaussian noise (Baseline) and AdvLatGAN [18] (iterative fast gradient sign method: I-FGSM), revealing the significant advantages of ICSO across various classic GAN architectures, including DCGAN [22], WGAN [23], WGAN-GP [16], SNGAN [24], LSGAN [25], WGAN-div [26], and ACGAN [27]. The experiments cover multiple benchmark datasets ranging from CIFAR-10 and STL-10 to LSUN-64/128, CelebA-64/128, and ImageNet, demonstrating the consistent superiority of ICSO on datasets of different scales and complexities. Specifically, on the CIFAR-10 and STL-10 datasets, ICSO achieves higher IS across multiple GANs, demonstrating the remarkable quality and diversity of the generated images. Similarly, lower FID further validates the authenticity and detail fidelity of the images generated by ICSO. When applied to more complex datasets such as LSUN-64/128, CelebA-64/128, and ImageNet, ICSO leverages its unique evolutionary strategies to deeply explore global optima and generate high-quality samples. Whether in a relatively simple generative adversarial network like DCGAN or in other classic GANs that employ more complex training mechanisms to enhance stability and the quality of generated images, such as WGAN-GP and WGAN-div, ICSO demonstrates outstanding optimization capabilities. However, when applied to SNGAN across CIFAR-10, STL-10, and CelebA-64 datasets, ICSO does not perform as well as AdvLatGAN. This is attributed to SNGAN's use of spectral normalization techniques to stabilize the training process, which provide more stable gradients, highly beneficial for gradient descent-based methods like the I-FGSM used by AdvLatGAN. This consistent outstanding performance across datasets and model architectures demonstrates the high scalability of ICSO, indicating that it is not only applicable to small-scale datasets but also to large-scale datasets, and can be effectively applied to different foundational model architectures. This provides a new approach for improving latent space optimization in GAN.

**Table 1.** Results of ICSO and state-of-the-art algorithms with different datasets on multiple mainstream architectures.

| Dataset | Method | Baseline | | AdvLatGAN | | ICSO | |
|---|---|---|---|---|---|---|---|
| | | IS | FID | IS | FID | IS | FID |
| CIFAR-10 | DCGAN | 5.92 | 46.4 | 6.21 | 41.7 | **6.39** | **38.60** |
| | WGAN | 6.63 | 32.8 | 7.21 | 27.3 | **7.26** | **26.36** |
| | WGAN-GP | 7.47 | 24.7 | 7.60 | 22.6 | **7.81** | **20.32** |
| | SNGAN | 7.29 | 25.5 | **7.58** | **22.3** | 7.40 | 22.46 |
| | LSGAN | 5.87 | 49.3 | 6.13 | 42.8 | **6.66** | **39.10** |
| | WGAN-div | 7.43 | 23.8 | 7.81 | 20.6 | **7.87** | **19.93** |
| | ACGAN | 6.02 | 59.5 | 6.06 | 53.7 | **6.46** | **39.71** |
| STL-10 | DCGAN | 7.18 | 61.2 | 7.33 | 56.3 | **7.62** | **50.94** |
| | WGAN | 6.51 | 73.0 | 7.62 | 51.0 | **7.69** | **49.69** |
| | WGAN-GP | 8.86 | 37.4 | 8.90 | 34.2 | **9.04** | **32.70** |
| | SNGAN | 8.49 | 36.8 | **8.63** | **34.5** | 8.50 | 34.59 |
| | LSGAN | 7.08 | 62.9 | **7.16** | **58.5** | 6.97 | 61.56 |
| | WGAN-div | 8.82 | 37.7 | 9.00 | 32.0 | **9.09** | **31.91** |
| LSUN-64 | WGAN-GP | - | 14.0 | - | 12.4 | **2.63** | **10.39** |
| | SNGAN | - | 11.9 | - | 9.9 | **2.44** | **9.72** |
| LSUN-128 | WGAN-GP | - | 14.1 | - | 12.5 | **2.81** | **12.29** |
| | SNGAN | - | 16.0 | - | 14.2 | **3.22** | **14.11** |
| CelebA-64 | WGAN-GP | - | 19.8 | - | 18.9 | 3.06 | **17.96** |
| | SNGAN | - | 19.5 | - | **18.2** | 2.73 | 18.38 |
| CelebA-128 | WGAN-GP | - | 25.2 | - | 22.3 | **7.89** | **22.30** |
| | SNGAN | - | 22.9 | - | 22.4 | **7.91** | **22.21** |
| ImageNet | WGAN-GP | - | 78.1 | - | 73.6 | **9.81** | **71.50** |
| | SNGAN | - | 98.3 | - | 79.0 | **8.81** | **78.40** |

### *4.5. Mode Collapse*

When evaluating the diversity of GANs, the 2D Gaussian mixture distribution, typically consisting of 8-Gaussian-mixture distribution and 25-Gaussian-mixture distribution, can visually demonstrate the model's mode collapse. Typically, this can be directly observed using kernel density estimation (KDE) plots, with side plots reflecting the probability distribution. To verify the effectiveness of ICSO, ICSO-GAN is compared with five classical GANs, i.e., GAN [28], NS-GAN [28], LS-GAN, E-GAN [29], and IE-GAN. For fairness, various GANs are trained using the same three-layer MLP network architecture. In particular, all images are from [15], except ICSO-GAN. As depicted in Figure 4, on two synthetic datasets, all classical GANs tend to generate some missing modes. Taking the 8-Gaussian-mixture distribution as an example, GAN produces seven modes, NS-GAN produces six modes, and LS-GAN and E-GAN produce very few modes. These results indicate that they are affected by mode collapse. In contrast, ICSO-GAN successfully learns the Gaussian mixture distribution for all modes, although some modes are weakly covered. Similar experimental results also occur in 25-Gaussian-mixture distribution. Compared with most classical GANs, ICSO-GAN has obvious advantages. This suggests that our evolutionary strategy can effectively suppress mode collapse. This is corroborated by the side plots, which show that the probability distribution of ICSO-GAN is closest to that of the target dataset.

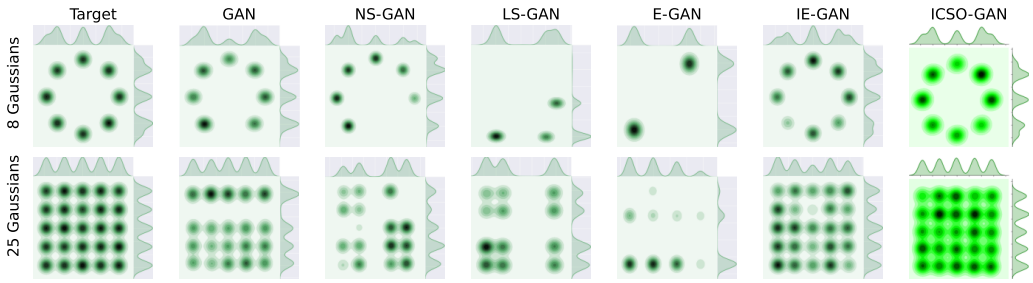



**Figure 4.** KDE plots of the target data and the generated data of various GANs trained on the synthetic dataset.

### *4.6. Ablation Study and Analysis*

**Comparison with some popular evolutionary algorithms:** In order to clearly demonstrate the advantages of ICSO, the experiment selected several well-recognized and widely applied algorithms in the field of optimization, including PSO [30], DE [31], CSO, BOA [32], and WOA [33], and conducted a systematic comparison between these algorithms and the proposed ICSO based on the WGAN-GP framework and CIFAR-10 dataset. In addition to the IS and FID, the experiment introduces density and coverage, which evaluate the quality and diversity of generated images from two perspectives: the degree of overlap between generated samples and real data in the feature space, and the coverage of categories. Through a comparison in Table 2, it can be clearly observed that ICSO exhibits significant superiority over other evolutionary algorithms in multiple key metrics. Further analysis of the reasons reveals that both DE and CSO rely on information exchange strategies among individuals to generate new offspring. However, relying solely on the intra-population evolution mechanism often fails to maintain population diversity, making the algorithm prone to falling into local optima. Both BOA and WOA utilize a two-stage mechanism involving global exploration followed by local exploitation to generate new candidate solutions. Nevertheless, due to the absence of mutation operators, these algorithms may suffer from limited population diversity, leading to premature convergence. In contrast, ICSO ingeniously incorporates the evolutionary mechanism of PSO, enabling the more precise and efficient tracking of the global optimal solution during the evolutionary iteration process, rather than simply performing crossover operations among individuals. From an overall efficiency perspective, ICSO demonstrates excellent performance. Although its two-stage operational mechanism leads to a longer runtime compared to PSO and DE, the performance improvement brought by this design far outweighs the additional computational cost. Compared with CSO, BOA, and WOA, ICSO also exhibits higher runtime efficiency. Unlike CSO, which relies solely on cross-dimensional crossover operations to escape local optima, and BOA and WOA, which employ a two-stage strategy combining global and local search, the strength of ICSO lies in its effective integration of a global best mechanism during the evolutionary process. This enables the algorithm to more accurately guide the population toward the global optimal solution, thereby enhancing both convergence speed and search quality.

**Comparison with different objective functions:** By comparing multiple perspectives of the objective function in ICSO, the evolutionary process can be guided more precisely, thereby significantly improving the performance of GAN. We conduct similar experiments above, utilizing WGAN-GP on CIFAR-10, and apply the evaluation metrics of IS, FID, Density, and Coverage. It can be seen from Table 3 that ICSO has a competitive advantage in various metrics. Specifically, ICSO not only achieves higher scores on the IS, indicating that the images it generates are both realistic and diverse, but also achieves lower values on the FID, further demonstrating the high consistency between its generated samples and the real data distribution. In addition, ICSO also performs exceptionally well in

terms of Density and Coverage, as it is able to generate dense sample points within the real data distribution while effectively covering a broader range of distribution areas, avoiding the problem of mode collapse. In summary, in the process of optimizing GAN, this algorithm comprehensively surpasses other comparative algorithms by enhancing the quality, diversity, and coverage breadth of the generated samples.

**Table 2.** Results of some classical evolutionary algorithms optimizing the latent space.

| Algorithm | IS (↑) | FID (↓) | Density (↑) | Coverage (↑) | GPU/Days |
|---|---|---|---|---|---|
| PSO | 7.577 | 23.036 | 0.478 | 0.726 | 2.0 |
| DE | 7.583 | 22.164 | 0.474 | 0.729 | 2.2 |
| CSO | 7.688 | 21.801 | 0.492 | 0.719 | 2.99 |
| BOA | 7.655 | 21.912 | 0.485 | 0.735 | 3.02 |
| WOA | 7.712 | 22.012 | 0.488 | 0.725 | 3.12 |
| ICSO | **7.816** | **20.328** | **0.510** | **0.760** | 2.98 |

Note: ↑ indicates that a higher value is better, while ↓ indicates that a lower value is better.

**Table 3.** Results of different objective functions of ICSO.

| Evaluation | IS (↑) | FID (↓) | Density (↑) | Coverage (↑) |
|---|---|---|---|---|
| E-GAN | 7.652 | 21.624 | 0.492 | 0.747 |
| IE-GAN | 7.693 | 21.574 | 0.506 | 0.750 |
| Non-GP ICSO | 7.719 | 22.531 | 0.488 | 0.740 |
| Unnormalized ICSO | 7.762 | 21.437 | 0.521 | 0.757 |
| ICSO | **7.816** | **20.328** | **0.510** | **0.760** |

Note: ↑ indicates that a higher value is better, while ↓ indicates that a lower value is better.

### *4.7. Industrial Application of StyleGAN3 with ICSO*

To evaluate the performance and applicability of ICSO in industrial applications, we integrate it into the StyleGAN3 [34] network architecture, which possesses outstanding industrial characteristics, and apply it to the inspection of traditional electric poles, addressing challenges brought by grid expansion, such as increasing demand for inspection personnel and misjudgments or omissions due to extended inspection cycles. By applying ICSO to power line inspection, a typical industrial scenario, we not only validate its effectiveness in enhancing infrastructure resilience, but also further demonstrate its potential in automating and intelligentizing inspection tasks. At the same time, the generative AI capabilities embodied in this method provide traditional industrial systems with novel means of data generation and enhancement, offering technological support for advancing industrial digital transformation. Especially in the current context emphasizing green development, low carbon emissions, and efficient resource utilization, ICSO has the capability to generate large-scale, high-quality image data at relatively low real-world acquisition costs. This provides a feasible pathway toward building data reuse mechanisms, alleviating data acquisition bottlenecks, and achieving sustainable operations [35,36]. By employing ICSO-GAN to generate images of electric poles taken by an unmanned aerial vehicle (UAV), the aim is to construct a dataset specifically for the automated inspection of electric poles. The experiment bases on the stylegan3-r-ffhqu-256x256 model as a pre-trained model with an additional 5000 kimgs of training. The original dataset contains a total of 1000 images of a single category of electric poles. To assess the performance of the proposed algorithm, it is compared with random Gaussian noise and I-FGSM optimized latent space in StyleGAN3. Evaluation metrics include fid50k_full (FID against the full dataset), kid50k_full (Kernel

inception distance [37] against the full dataset), pr50k3_full (Precision and recall [38] against the full dataset, i.e., pr50k3_full_precision and pr50k3_full_recall), ppl2_wend (Perceptual path length [39] in latent space, endpoints, full image), eqt50k_int (Equivariance [34] w.r.t. integer translation: EQ-T), eqt50k_frac (Equivariance [34] w.r.t. fractional translation: EQ-Tfrac), and eqr50k (Equivariance [34] w.r.t. rotation: EQ-R).

**Quantitative comparison:** As can be seen from Figure 5 and Table 4, ICSO demonstrates superior performance across multiple evaluation metrics. Specifically, ICSO achieves the best performance on metrics such as fid50k_full, kid50k_full, pr50k3_full_precision, eqt50k_int, and eqr50k. These metrics cover various aspects of generative models, including image quality (such as FID and KID), precision, and recall. Although ICSO's performance is slightly inferior to other methods on some metrics (such as pr50k3_full_recall, ppl2_wend and eqt50k_frac), overall, ICSO shows competitiveness across most metrics. This indicates the robustness and effectiveness of ICSO in handling complex tasks.

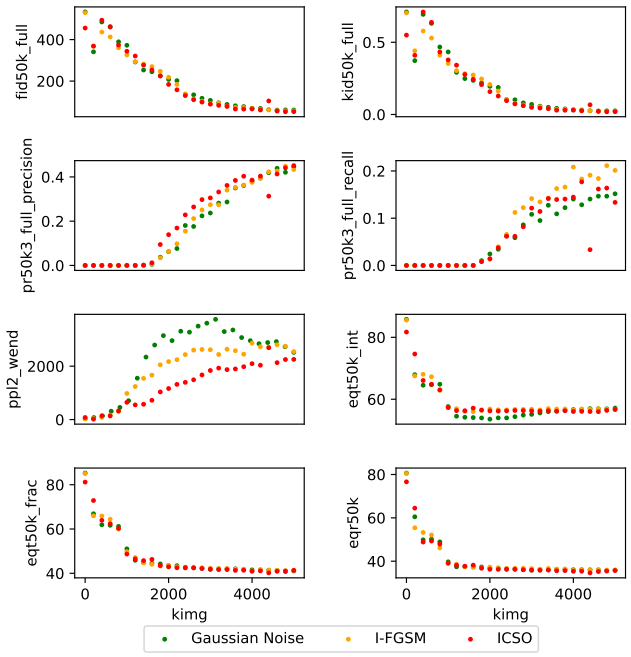



**Figure 5.** Comparison of ICSO with other algorithms across various metrics.

**Table 4.** Minimum values of ICSO and other algorithms on various metrics.

| Metrics | Gaussian Noise | I-FGSM | ICSO |
|---|---|---|---|
| fid50k_full (↓) | 60.591 | 56.371 | **52.639** |
| kid50k_full (↓) | 0.024 | 0.022 | **0.019** |
| pr50k3_full_precision (↑) | 0.445 | 0.433 | **0.451** |
| pr50k3_full_recall (↑) | 0.151 | **0.201** | 0.133 |
| ppl2_wend (↑) | 2517.458 | **2545.676** | 2254.295 |
| eqt50k_int (↓) | 57.132 | 56.699 | **56.689** |
| eqt50k_frac (↓) | **41.085** | 41.324 | 41.317 |
| eqr50k (↓) | 35.850 | 35.955 | **35.707** |

Note: ↑ indicates that a higher value is better, while ↓ indicates that a lower value is better.

**Qualitative results:** StyleGAN3 generates random images from a latent space that has been optimized using various algorithms. As shown in Figure 6, the images generated by other algorithms are distorted when compared with those generated by ICSO. Since ICSO

incorporates the global optimal mechanism and the local optimal mechanism from PSO on the basis of the original CSO algorithm, it can always guide the evolution of the population towards the globally optimal direction, rather than relying solely on heuristic iterations.




**Figure 6.** Images are generated by StyleGAN3, utilizing variously optimized latent spaces for the UAV electric poles dataset.

## 5. Conclusions

This paper proposes the improved crisscross optimization (ICSO) algorithm for optimizing the latent space of GAN. In the algorithm, normalization is employed to equilibrate the order of magnitude between two independent attributes of quality and diversity within the objective function. Then, the gradient of the discriminator is constrained by introducing the gradient regularization term (i.e., GP) into the discriminator's loss function, which helps to avoid problems such as mode collapse and gradient exploding in the training process of GAN. By incorporating the local and global search mechanisms of PSO with the rapid convergence of the original crisscross optimization, ICSO effectively directs the evolution of population towards the optimal individuals. The experiments show that the proposed algorithm has significant advantages in the quality and diversity of the generated images, and it also has the prospect of being applied to industrial applications, specifically in the area of UAV image generation.

In the future, we will study how to leverage attention mechanisms, wavelet analysis, and other technologies to explore the impact of latent space on the generation performance of GAN in greater depth.

**Author Contributions:** Methodology, Z.C. (Zhihui Chen), T.L. and Z.C. (Zhanchuan Cai); Software, Z.C. (Zhihui Chen) and Z.L.; Validation, Z.C. (Zhihui Chen), Z.C. (Zhanchuan Cai), Z.L. and R.C.; Data curation, Z.C. (Zhihui Chen); Writing—original draft, Z.C. (Zhihui Chen); Writing—review & editing, T.L., Z.C. (Zhanchuan Cai), Z.L. and R.C.; Visualization, Z.C. (Zhihui Chen) and T.L.; Supervision, Z.C. (Zhanchuan Cai); Project administration, T.L.; Funding acquisition, T.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Science and Technology Development Fund of Macau under Grant 0037/2023/ITP1, the Macau University of Science and Technology Faculty Research Grants under Grant FRG-24-025-FIE, and the Guangzhou Development District International Cooperation Project under Grant 2023GH01.

**Data Availability Statement:** The data used in the experiments of this article are publicly available online. Specifically, the CIFAR-10 dataset can be accessed at https://www.cs.toronto.edu/~kriz/cifar.html, the STL-10 dataset at https://cs.stanford.edu/~acoates/stl10/, the LSUN dataset at https://www.yf.io/p/lsun, the CelebA dataset at https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html, and the ImageNet dataset at https://www.image-net.org/. The UAV-captured images used in this study will be made available by the authors upon request.

**Conflicts of Interest:** Author Zonglin Liu was employed by the company LANGO Tech Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## References

1. Zhu, B.; Hu, X.G. WGAN-Based Realization Process of Gravel Soil for Hydraulic Property Simulation. *Appl. Sci.* **2024**, *14*, 9873. [CrossRef]
2. Jeong, H.; Lee, H.; Kim, C.; Shin, S. A Survey of Robot Intelligence with Large Language Models. *Appl. Sci.* **2024**, *14*, 8868. [CrossRef]
3. Ruiz-Gándara, A.; Gonzalez-Abril, L. Generative Adversarial Networks in Business and Social Science. *Appl. Sci.* **2024**, *14*, 7438. [CrossRef]
4. Bonci, A.; Fredianelli, L.; Kermenov, R.; Longarini, L.; Longhi, S.; Pompei, G.; Prist, M.; Verdini, C. DeepESN Neural Networks for Industrial Predictive Maintenance through Anomaly Detection from Production Energy Data. *Appl. Sci.* **2024**, *14*, 8686. [CrossRef]
5. Wu, Z.; Wei, C.; Xia, Y.; Ji, Z. SAITI-DCGAN: Self-Attention based Deep Convolutional Generative Adversarial Networks for Data Augmentation of Infrared Thermal Images. *Appl. Sci.* **2024**, *14*, 11391. [CrossRef]
6. Tamayo-Urgilés, D.; Sanchez-Gordon, S.; Valdivieso Caraguay, Á.L.; Hernández-Álvarez, M. GAN-Based Generation of Synthetic Data for Vehicle Driving Events. *Appl. Sci.* **2024**, *14*, 9269. [CrossRef]
7. Wang, Z.; Wang, C. DSpix2pix: A New Dual-Style Controlled Reconstruction Network for Remote Sensing Image Super-Resolution. *Appl. Sci.* **2025**, *15*, 1179. [CrossRef]
8. Meng, A.B.; Chen, Y.C.; Yin, H.; Chen, S.Z. Crisscross Optimization Algorithm and its Appl. *Knowl.-Based Syst.* **2014**, *67*, 218–229. [CrossRef]
9. Meng, A.; Zhang, H.; Yin, H.; Xian, Z.; Chen, S.; Zhu, Z.; Zhang, Z.; Rong, J.; Li, C.; Wang, C.; et al. A Novel Multi-Gradient Evolutionary Deep Learning Approach for Few-Shot Wind Power Prediction Using Time-Series GAN. *Energy* **2023**, *283*, 129139. [CrossRef]
10. Meng, A.; Chen, S.; Ou, Z.; Xiao, J.; Zhang, J.; Chen, S.; Zhang, Z.; Liang, R.; Zhang, Z.; Xian, Z.; et al. A Novel Few-Shot Learning Approach for Wind Power Prediction Applying Secondary Evolutionary Generative Adversarial Network. *Energy* **2022**, *261*, 125276. [CrossRef]
11. Liu, Y.; Li, Q.; Deng, Q.; Sun, Z. Towards Spatially Disentangled Manipulation of Face Images with Pre-Trained StyleGANs. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 1725–1739. [CrossRef]
12. Wang, D.; Qin, X.; Song, F.; Cheng, L. Stabilizing Training of Generative Adversarial Nets via Langevin Stein Variational Gradient Descent. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 2768–2780. [CrossRef] [PubMed]
13. Zhang, Y.; Nie, X.; He, R.; Chen, M.; Yin, Y. Normality Learning in Multispace for Video Anomaly Detection. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *31*, 3694–3706. [CrossRef]
14. Wang, S.; Chen, S.; Chen, T.; Nepal, S.; Rudolph, C.; Grobler, M. Generating Semantic Adversarial Examples via Feature Manipulation in Latent Space. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 17070–17084. [CrossRef]
15. Li, J.; Li, J.; Zhou, W.; Lü, S. IE-GAN: An Improved Evolutionary Generative Adversarial Network Using a New Fitness Function and a Generic Crossover Operator. *arXiv* **2021**, arXiv:2109.11078.
16. Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A.C. Improved Training of Wasserstein GANs. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5769–5779.
17. Steffen, V. Particle Swarm Optimization with a Simplex Strategy to Avoid Getting Stuck on Local Optimum. *AI Comput. Sci. Robot. Technol.* **2022**, 1–40. [CrossRef]
18. Li, Y.; Mo, Y.; Shi, L.; Yan, J. Improving Generative Adversarial Networks via Adversarial Learning in Latent Space. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 8868–8881.
19. Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved Techniques for Training GANs. In Proceedings of the 30th International Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 2234–2242.
20. Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; Hochreiter, S. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 6629–6640.
21. Naeem, M.F.; Oh, S.J.; Uh, Y.; Choi, Y.; Yoo, J. Reliable Fidelity and Diversity Metrics for Generative Models. In Proceedings of the 37th International Conference on Machine Learning, Online, 13–18 July 2020; pp. 7176–7185.
22. Lim, S.; Chan, C.; Faizal, E.; Ewe, K. Progressive Expansion: Cost-Efficient Medical Image Analysis Model with Reversed Once-For-All Network Training Paradigm. *Neurocomputing* **2024**, *581*, 127512. [CrossRef]

23. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein Generative Adversarial Networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 214–223.
24. Miyato, T.; Kataoka, T.; Koyama, M.; Yoshida, Y. Spectral Normalization for Generative Adversarial Networks. In Proceedings of the 6th International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018; pp. 1–26.
25. Mao, X.; Li, Q.; Xie, H.; Lau, R.Y.; Wang, Z.; Paul Smolley, S. Least Squares Generative Adversarial Networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2794–2802.
26. Wu, J.; Huang, Z.; Thoma, J.; Acharya, D.; Van Gool, L. Wasserstein Divergence for GANs. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 653–668.
27. Kang, M.; Shim, W.; Cho, M.; Park, J. Rebooting Acgan: Auxiliary Classifier GANs with Stable Training. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 23505–23518.
28. 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; pp. 2672–2680.
29. Wang, C.; Xu, C.; Yao, X.; Tao, D. Evolutionary Generative Adversarial Networks. *IEEE Trans. Evol. Comput.* **2019**, *23*, 921–934. [CrossRef]
30. Kennedy, J.; Eberhart, R. Particle Swarm Optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
31. Storn, R.; Price, K. Differential Evolution–a Simple and Efficient Heuristic for Global Optimization over Continuous Sspaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
32. Benmamoun, Z.; Khlie, K.; Bektemyssova, G.; Dehghani, M.; Gherabi, Y. Bobcat Optimization Algorithm: An Effective Bio-Inspired Metaheuristic Algorithm for Solving Supply Chain Optimization Problems. *Sci. Rep.* **2024**, *14*, 20099. [CrossRef]
33. Benmamoun, Z.; Khlie, K.; Dehghani, M.; Gherabi, Y. WOA: Wombat Optimization Algorithm for Solving Supply Chain Optimization Problems. *Mathematics* **2024**, *12*, 1059. [CrossRef]
34. Karras, T.; Aittala, M.; Laine, S.; Härkönen, E.; Hellsten, J.; Lehtinen, J.; Aila, T. Alias-Free Generative Adversarial Networks. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 852–863.
35. Liu, Y.; Wu, D.; Zhou, W.; Fan, K.; Zhou, Z. EACP: An Effective Automatic Channel Pruning for Neural Networks. *Neurocomputing* **2023**, *526*, 131–142. [CrossRef]
36. Wu, X.; Feng, Y.; Lou, S.; Zheng, H.; Hu, B.; Hong, Z.; Tan, J. Improving NeuCube Spiking Neural Network for EEG-based Pattern Recognition using Transfer Learning. *Neurocomputing* **2023**, *529*, 222–235. [CrossRef]
37. Bińkowski, M.; Sutherland, D.J.; Arbel, M.; Gretton, A. Demystifying MMD GANs. *arXiv* **2018**, arXiv:1801.01401.
38. Kynkäänniemi, T.; Karras, T.; Laine, S.; Lehtinen, J.; Aila, T. Improved Precision and Recall Metric for Assessing Generative Models. In Proceedings of the 33rd International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 3927–3936.
39. Karras, T.; Laine, S.; Aila, T. A Style-Based Generator Architecture for Generative Adversarial Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4401–4410.


**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.