Hindawi
Advances in Multimedia
Volume 2023, Article ID 3775614, 11 pages
https://doi.org/10.1155/2023/3775614

Review Article

Algorithm Comparison and Evaluation of GAN Models Based on Image Transferring from Desert to Green Field

Zhenyu Liu and Hongjun Li

College of Science, Beijing Forestry University, Beijing 100083, China

Correspondence should be addressed to Hongjun Li; lihongjun69@bjfu.edu.cn

Received 7 November 2022; Revised 30 June 2023; Accepted 5 July 2023; Published 20 July 2023

Academic Editor: Yu-Chen Hu

Some time-consuming and labor-intensive techniques, like manual drawing or interactive modeling with an image editing system, are often used to show how a desert area might look after being transformed into a green field (oasis) in an image way. In order to improve the rendering efficiency of image style transformation and increase the variety of renderings, we can build an algorithm for automatically generating style images based on machine learning. In this paper, after comparing seven generative adversarial network (GAN) models in the way of theory analysis, we propose a method for generating green fields using desert images as input data, and a comprehensive comparison is presented on how GANs are currently applied to solve the desert-to-oasis problem. Experimental results show that two GAN models, geometrically consistent GAN and cyclic consistent GAN, have the best transfer effect of a desert image to oasis one in the view of quantitative indicators, Fréchet inception distance, and learned perceptual image patch similarity.

1. Introduction

In recent years, image style transfer has been a hot research topic [1, 2], especially using generative adversarial networks (GANs) because GAN models increasingly support unsupervised learning for image-to-image (I2I) translation [3, 4]. With the in-depth study of GANs, GAN models are increasingly applied in various fields [5–10], including agricultural production, environmental preservation, urban architecture, etc. We mainly focus on the topic of image style transfer in the context of environmental preservation.

In the environmental preservation problem, Nazki et al. [7] used adversarial networks for improved plant disease recognition; Xu et al. [10] employed geometric transformation methods and GAN to enhance the environmental micro organizations image. However, few people pay attention to the problem of desert-to-oasis. As one of the image style transfer problems, the transfer from a desert image to a green field (oasis) one not only has significant research meaning in terms of environmental preservation and governance, for example, showing the greening effect of a desert, but also is a technical challenge because there is a big difference in styles between the input and output images. If one wants to visualize what a desert region will look like when it becomes an oasis, one has to use manual drawing technology or some specific software to generate a picture. This is not only less efficient but also more costly in terms of human and financial resources.

Recently, deep learning methods have been increasingly applied to visualization applications in various fields [11–13]. The fundamentals, technologies, and applications have been comprehensively summarized [14]. Although there are few published methods for transferring the desert image to a green field (oasis), we can treat it as a general image translation or style transfer problem. At first, we consider all desert images as one domain and all oasis images as another. In order to get the appearance of the specified desert area after it becomes an oasis, it needs to learn a mapping between the two domains, which is the problem of I2I translation. More specifically, the desert images are used as the input content image, and the oasis images are used as the input style images. By this way, we can get the new images with the oasis style and the desert content, which is also the problem of style transfer. Currently, GAN models are the most popular models used in this field, and we now apply GAN models to the problem of desert-to-oasis.

To provide a comprehensive state-of-the-art review of how GANs are currently applied to solve challenging tasks in the desert-to-oasis problem, we select seven representative GAN models and design appropriate experiments to find which one of GAN models is suitable for image transfer from a desert image into an oasis image. Since the desert dataset and the oasis dataset cannot be obtained directly, we created the original image datasets of deserts and oases and used transfer learning technology to solve the problem.

The main contributions of the paper are as follows:

(1) An algorithm of transfer learning was carefully built. Given that there are no pair images of the desert image and oasis image, the pretraining strategy of transfer learning from winter to summer is proposed.
(2) The experimental scheme was carefully designed for testing the algorithm of desert-to-oasis with seven GAN models. By experimental comparison, it is found that two GAN models are suitable for image transfer from a desert image into an oasis image.
(3) A small dataset including both desert images and green field images is also established for model training.

2. Related Work

2.1. GANs. The original GANs were initially proposed by Goodfellow et al. [15]. The basic idea of GANs is to learn the probability distribution of data from a training set of real data. It is a powerful framework with a generator and a discriminator that usually make use of deep neural networks [16, 17]. Since GANs have multiple challenges in their training, such as convergence, stability problems, or pattern collapse, a series of improvements have been presented in the following years [18–20]. Additionally, GANs have been used in research on various artificial intelligence subfields, such as speech and language processing [21, 22], and malware detection [23].

GANs have also been successfully applied to different computer vision tasks, such as text-to-image synthesis [24], image colorization [25], super-resolution [26], and style transfer [4, 27]. Although those GAN models have been successful in many specific applications, such as stylization and artistry, it is still worth testing their effect on the task of transferring desert image to oasis image.

2.2. Unsupervised I2I (UI2I) Translation. GANs have been used in a variety of image applications, especially for I2I translation. The idea of I2I translation can be traced back at least to the image analogies of Hertzmann et al. [28], who used a nonparametric texture model [29] on a single input–output training image pair. Typically, I2I translations can be divided into two groups: supervised settings (paired) and unsupervised settings (unpaired). Due to the unavailability or difficulty of collecting paired data, unpaired I2I translation has received a lot of attention, and the GAN model selected in this paper also complies with I2I translation in unsupervised settings.

In the unsupervised learning setting, I2I uses two larger but unpaired training image sets to convert images from one representation to another, which makes the task more practical but more difficult. Some attempts have been made to incorporate various constraint assumptions in subsequent studies. Several representative's methods are mainly based on two constraints: the cyclic consistency constraint [30–35] and the beyond-cyclic consistency constraint [36–38]. We, in this paper, choose cyclic consistent GAN (CycleGAN) [35] as one representative method among those methods with the cyclic consistency constraint and choose geometrically consistent GAN (GcGAN) [36] as one representative method among those methods with the beyond-cyclic consistency constraint.

The following works [39–42] further implemented multimode and multidomain synthesis to bring diversity in the translation output. What is more, studies have started to propose methods for a few-shot in I2I [43–45], which are still under further investigation.

2.3. Style Transfer and Domain Mapping. As one of the well-known computer vision tasks, style transfer [46] is an alternative method for performing I2I translation. It usually receives a style image and a content image as input and creates a new image with the first style and the second content [35, 47]. Obviously, we are more concerned with the mapping between two domains than the mapping between two specific images. Although it is different between the problem of image translation and domain mapping because when mapping between domains, it is not limited to the change of style, and some content will also be replaced; however, many examples of mapping across domains in the literature can be considered almost as style transfer [47].

3. GANs and Variants of GANs

Training supervised image translations are not practical due to the difficulty and high cost of acquiring these large pairs of training data in many tasks. For example, the actual photos of desert and oasis in the same land. It is nearly impossible to gather a sizable number of labeled paintings that match the input landscape in the case of photo-to-painting translation. Therefore, unsupervised methods are gradually gaining more attention. The I2I approaches use two large but unpaired sets of training images to transform images between representations in unsupervised learning. Unsupervised image transformation models use unpaired data, which does not require strict correspondence between the source and target domains and is easier to obtain compared to paired data. As a result, the field of unsupervised image transformation has given rise to diverse transformation models.

In this section, we use CycleGAN [35] and GcGAN [36] as representatives of cyclic consistency constraint and beyond-cyclic consistency constraint. To reflect the timeliness of the article, we select five methods from the past 2 years, CUT [48] and FastCUT [48], DCLGAN and SimDCL [49], and F-LSeSim [50], for comparison. All of them are shown in Table 1, and the characteristics and reviews of these methods are presented in the following subsections.

Table 1: List of selected methods, including model name, publication year, and the type of training data, whether multimodal or not and corresponding insights.

Method	Publication	Data	Multimodal	Insights
CycleGAN	2017	Unpaired	No	Cyclic loss
GcGAN	2019	Unpaired	No	One-sided UI2I; geometric transformation preservation
CUT	2020	Unpaired	No	One-sided UI2I; contrastive learning
FastCUT	2020	Unpaired	No	One-sided UI2I; contrastive learning
DCLGAN	2021	Unpaired	No	Two-sided UI2I; contrastive learning
SimDCL	2021	Unpaired	No	Two-sided UI2I; contrastive learning
LSeSim	2021	Unpaired	No	One-sided UI2I; self-similarity

3.1. GANs. The basic structure of the GAN model is shown in Figure 1(a) [4]. In the training process of GANs, the generator and the discriminator are similar to a two-player zero-sum game, and the optimization goal is to achieve Nash equilibrium [51] through sufficient training to make the generated data of the generative network as real as possible. This two-player model can be summarized as a min–max problem between the generator G and the discriminator D, i.e.,

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

where $V(G,D)$ is the value function, $p_{\text{data}}(x)$ represents the distribution of real data, and $p_z(z)$ denotes the model distribution of the random noise input z.

Thus, we obtain a generator that can generate real samples, which is the reason why GANs are widely used for image generation. Note that the adversarial discrimination process in training can be considered as a special loss called adversarial loss, which is one of the highlights of GANs.

Adversarial loss has been employed in numerous existing methods for image transformation tasks, including style transfer, picture super-resolution, and other restoration tasks [4]. The general model of these tasks is shown in Figure 1(b) [4]. For the input image y, the output of the generator network and the actual image are fed to the discriminator to compute the adversarial loss. Unfortunately, these methods can only learn styles from a single image, which can lead to rigid results. With recent advances in GANs, the results of style transformation tend to be more realistic, where styles are not learned from individual images but from a set of images with the target style. This type of style transfer can also be referred to as I2I translation.

3.2. CycleGAN. The pix2pix architecture proposed by Isola et al. [52] can be used for these I2I translation tasks when paired training data are available. However, it cannot be used for unpaired data. The CycleGANs proposed by Zhu et al. [35] in 2017 address this problem well and CycleGAN is also considered an important advancement in the research of image translation for unpaired data. DualGAN [34] and DiscoGAN [31] were proposed almost simultaneously with CycleGAN, and their basic ideas are roughly the same, so we only use CycleGAN as a representative of translation using a cycle-consistency constraint.

The framework structure of CycleGAN is a ring structure consisting of two symmetric GANs, or, to be more precise, there are two generative networks and two discriminative networks. Its generators use a residual network structure to transform images to another domain through intermediate representations, i.e., to convert one class of images to another class of images. Instead of random noise, the input to the generative network is the source domain image dataset X, and the generated images have the characteristics of the target domain image dataset Y. In the network model of CycleGAN, the actual training goal is to learn the mapping from the source domain X to the target domain Y. Let this mapping be G, which corresponds to the generative network in GANs, and G can convert the image x in the source domain X into the image $G(x)$ in the target domain Y. The adversarial loss of this process is as follows:

$$\mathcal{L}_{\text{GAN}}(G,D,X,Y) = \mathbb{E}_{y\sim p_{\text{data}}(y)}[\log D(y)] + \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log(1-D(G(x)))], \tag{2}$$

where D is the discriminator of G, which aims to distinguish the generated sample $G(x)$ from the real sample Y.

To avoid the problem that mapping G maps all x into the same image in the target domain Y, i.e., generating a single sample with a collapsed pattern that does not truly reflect the target domain features [53], CycleGAN introduces a cycle-consistent constraint for image reconstruction. The cycle-consistent constraint can be reduced from the target domain Y to the source domain X. Given another generative mapping F, the image y in the target domain image dataset Y is transformed into the image $F(y)$ in the source domain X. In a strict sense, G and F are identical, and the adversarial loss of this process is similar to G. Also, to further reduce the space of possible mapping functions, Zhu et al. [35] argued that the learned mapping functions should be cyclically consistent. For each image x from domain X, the image transformation loop should be able to bring x back to the original image. This is referred to as forwarding loop consistency. Similarly, for each image y from domain Y, the reverse cyclic consistency should be satisfied as well, from which the loop consistency loss can be obtained as follows:

FIGURE 1: Experimental results of image transferring from desert to green scene using seven GANs models.

$$\mathscr{L}_{\text{cyc}}(G,F) = \mathbb{E}_{x\sim p_{\text{data}}(x)}[\|G(F(x)-x)\|_1] + \mathbb{E}_{y\sim p_{\text{data}}(y)}[\|F(G(y)-y)\|_1]. \quad (3)$$

The goal of the training is to make the content of the forward and backward loops as consistent as possible, i.e., to make the two approximately equal signs converge as closely as possible to an equal sign.

The contribution of CycleGAN is that its proposed cyclic consistency constraint can transform images from the source domain X to the target domain Y without requiring paired datasets as training sets, which greatly improves the flexibility of GANs model application scenarios. Meanwhile, CycleGAN has become one of the typical representatives of unsupervised learning models for GANs.

3.3. GcGAN. Although CycleGAN successfully employs GANs in unsupervised learning, where the cyclic consistency constraint removes the dependence on supervised pairwise data, it tends to force the model to generate translated images containing all the information of the input image to reconstruct the input image. It is important to deal with more variations and extreme transformations, especially geometric variations [35]. When these two domains require substantial clutter and heterogeneity rather than small and simple changes in low-level shapes and backgrounds, the use of cyclic loss methods is usually unsuccessful [54].

Many constraints and assumptions have been proposed to improve cycle consistency to solve the above problems. We focus on methods to eliminate the cycle consistency constraint by designing the model as a one-sided translation process. These methods usually consider some geometric distance as the content loss between the original source image and the translated result. For example, DistanceGAN achieves one-sided translation by maintaining the distance between images in the domain [47]. GcGAN preserves a

given geometric transformation between the input images before and after translation [36]. Here, we use GcGAN as a representative.

Fu et al. [36] argued that although CycleGAN and DistanceGAN successfully constrain the solution space, they ignore the special property of images that simple geometric transformations do not change the semantic structure of the image. Based on this special property, the GcGAN was born. GcGAN contains two generators G_{XY} and $G_{\tilde{X}\tilde{Y}}$. Given a predefined common geometric transformation function (e.g., vertical flip and 90° clockwise rotation), GcGAN feeds the original image x and the corresponding image transformed by the predefined geometric transformation into the two generators correspondingly and generates two images in the new domain in which the two images are generated and combined with the corresponding geometric consistency constraints, i.e.,

$$\begin{aligned}\mathscr{L}_{\text{geo}}(G_{XY}, G_{\tilde{X}\tilde{Y}}, X, Y) = \\ \mathbb{E}_{x\sim p_X}[\|G_{XY}(x) - f^{-1}(G_{\tilde{X}\tilde{Y}}(f(x)))\|_1] \\ +\mathbb{E}_{x\sim p_X}[\|G_{\tilde{X}\tilde{Y}}(f(x)) - f(G_{XY}(x))\|_1].\end{aligned} \tag{4}$$

This geometric consistency loss can be considered a reconstruction loss that depends on a predefined geometric transformation function $f(\cdot)$. The geometric consistency constraint reduces the space of possible solutions while keeping the correct solution in the search space.

3.4. CUT and FastCUT. Although improved methods such as DistanceGAN and GcGAN work well, they rely on the relationship between the whole image or usually on a predefined distance function. In fact, in an I2I transformation, each patch in the output should reflect the content of the corresponding patch in the input, independent of the domain. For example, for the generated zebra forehead, one should know that it comes from the horse's forehead and not from other parts of the horse or other parts of the horse. Park et al. [48] proposed a CUT that maximizes the mutual information between input–output pairs by patch-based contrast learning without relying on prespecified distances or operating on the whole image, thus replacing cycle consistency.

The CUT needs to learn mappings in only one direction and avoids using reverse auxiliary generators and discriminators. This can greatly simplify the training procedure and reduce the training time. While using the traditional adversarial loss (see Equation (1)), CUT uses a noisy contrast estimation framework [55] to maximize the mutual information between input and output. The basic idea of contrast learning is to associate two signals, a "query" and its "positive" example in a dataset, and form a contrast with other points which are considered "negatives." The query, positive, and N negatives are mapped to the K-dimensional vectors v, $v^+ \in \mathbb{R}^K$ and $v^- \in \mathbb{R}^{N\times K}$, $v_n^- \in \mathbb{R}^K$ denoting the n^{th} negative, and then they are normalized, and an $(N+1)$-way classification problem is created. The probability of a positive example being selected over negatives is expressed by calculating the cross-entropy loss as follows:

$$\ell(v, v^+, v^-) = -\log\left[\frac{\exp(v\cdot v^+/\tau)}{\exp(v\cdot v^+/\tau) + \sum_{n=1}^{N}\exp(v\cdot v_n^-/\tau)}\right], \tag{5}$$

where τ represents a temperature parameter that scales the distance between the query and other examples, and the default is 0.07. The goal of CUT is to correlate the input and output data, where the query refers to an output, and the positive and negatives are the corresponding input and non-corresponding input.

In unsupervised learning, corresponding patches between the input and output photos are just as important as the entire image sharing the same content. Therefore, CUT uses a multilayer, patch-based learning objective. CUT decomposes the generating function G into two components, an encoder (G_{enc}), then a decoder (G_{dec}), and applies them sequentially to get the generated image $\hat{y} = G(z) = G_{\text{dec}}(G_{\text{enc}}(x))$.

Since extracting features through the encoder G_{enc} yields a feature stake, L layers are selected, and the feature map is passed through a two-layer MLP H_l, which encodes an input image into a stack of features as follows:

$$\{z_l\}_L = \{H_l(G_{\text{enc}}^l(x))\}_L, \tag{6}$$

where G_{enc}^l denotes the output of the selected l^{th} layer. Similarly, an output image $\tilde{y}$ is encoded as follows:

$$\{\hat{z}_l\}_L = \{H_l(G_{\text{enc}}^l(G(x)))\}_L. \tag{7}$$

Also, each layer and spatial location in the feature stack represents a patch of the input image. Denote the spatial locations in each selected layer as $s \in \{1, \ldots, S_l\}$, where S_l is the number of spatial locations in each layer. Each time a query is selected from the output, the corresponding feature("positive") is called $z_l^s \in \mathbb{R}^{C_l}$, and the other features ("negatives") are called $z_l^{S\setminus s} \in \mathbb{R}^{(S_l-1)\times C_l}$, where C_l is the number of channels in each layer.

The goal of the CUT is to match the corresponding input–output patches at a specific location. Other patches in the input image can be used as negatives; thus, the PatchNCE losses can be obtained as follows:

$$\mathscr{L}_{\text{PatchNCE}}(G, H, X) = \mathbb{E}_{x\sim X}\sum_{l=1}^{L}\sum_{s=1}^{S_l}\ell\left(\hat{z}_l^s, z_l^s, z_l^{S\setminus s}\right). \tag{8}$$

In addition, it is mentioned in the literature that PatchNCE loss can be used $\mathscr{L}_{\text{PatchNCE}}(G, H, Y)$ for images from domain Y as well to prevent unnecessary changes to the generator. Thus the overall objective function is as follows:

$$\mathscr{L}_{\text{total}}(G,D,X,Y)=\mathscr{L}_{GAN}(G,D,X,Y)+\lambda_x\mathscr{L}_{\text{PatchNCE}}(G,H,X)+\lambda_Y\mathscr{L}_{\text{PatchNCE}}(G,H,Y),\tag{9}$$

where $\mathscr{L}_{\text{GAN}}(G,D,X,Y)$ is as Equation (2). And when set $\lambda_X=\lambda_Y=1$ to perform joint training, it is called CUT; when set $\lambda_X=10,\lambda_Y=0$ instead, it is called FastCUT and can be thought of as a faster, lighter version of CycleGAN.

3.5. DCLGAN and SimDCL. Although CUT demonstrates the efficiency of contrast learning, certain design choices limit its performance. For example, one embedding is used for two different domains, which may not capture domain gaps efficiently. To further exploit contrast learning and avoid the drawback of cycle consistency [35], Han et al. [49] proposed a dual contrastive learning approach, called DCLGAN.

DCLGAN aims to maximize mutual information by using two separate embeddings to learn the correspondence between input and output image blocks. DCLGAN has two generators, G and F, and two discriminators, D_X and D_Y, similar to CycleGAN. The first half of the generators are defined as encoders, denoted as G_{enc} and F_{enc}, respectively, while the second half is the decoder, i.e., G_{dnc} and F_{dnc}. For each mapping, DCLGAN extracts the features of the image from the four-layer encoder and sends them to the two-layer MLP mapping head (H_X and H_Y). This mapping head learns to project the features extracted from the encoder onto a stake of features. At this point, G_{enc} and H_X are used as embeddings of domain X, F_{enc}, and H_Y are used as embeddings of domain Y. In addition to bilateral adversarial loss, DCLGAN, similar to CUT, also utilizes patch-based multilayer contrast learning with a cross-entropy loss as Equation (5). Additionally, it introduces similarity index sim as follows:

$$\text{sim}(u,v)=\frac{u^Tv}{\|u\|\|v\|},\tag{10}$$

which denotes the cosine similarity between u and v.

However, differently from CUT, for the generated fake image $G(x)$ belonging to the domain Y, DCLGAN takes the advantage of double learning by using a different embedding of domain Y, i.e., $\{\hat{z}_l\}_L=H_Y\left(F^l_{\text{enc}}(G(x))\right)\}_L$, and the PatchNCE loss of the mapping can be obtained as follows:

$$\mathscr{L}_{\text{PatchNCE}_X}(G,H_X,H_Y,X)=\mathbb{E}_{x\sim X}\sum_{l=1}^{L}\sum_{s=1}^{S_l}\ell\left(\hat{z}_l^s,z_l^s,z_l^{S\backslash s}\right),\tag{11}$$

where $\{z_l\}_L=H_X\left(G^l_{\text{enc}}(x)\right)\}_L$.

A similar loss is introduced for the reverse mapping $F:Y\longrightarrow X$:

$$\mathscr{L}_{\text{PatchNCE}_Y}(G,H_X,H_Y,Y)=\mathbb{E}_{y\sim Y}\sum_{l=1}^{L}\sum_{s=1}^{S_l}\ell\left(\hat{z}_l^s,z_l^s,z_l^{S\backslash s}\right).\tag{12}$$

To prevent unnecessary changes to the generator, DCLGAN adds an identity loss:

$$\mathscr{L}_{\text{identity}}(G,F)=\mathbb{E}_{x\sim X}[\|F(x)-x\|_1]+\mathbb{E}_{y\sim Y}[\|G(y)-y\|_1].\tag{13}$$

Additionally, DCLGAN introduces similarity loss by taking advantage of the fact that images from the same domain have a common style despite their semantic differences. The similarity loss is calculated by projecting the real images and the generated fake images belonging to the same domain into a 64-dimensional vector by four lightweight networks (H_{xr}, H_{xf}, H_{yr}, H_{yf}):

$$\begin{aligned}\mathscr{L}_{\text{sim}}&\left(G,F,H_X,H_Y,H_{xr},H_{xf},H_{yr},H_{yf}\right)\\&=\left[\left\|H_{xr}(H_X(G_{\text{enc}}(x)))-H_{xf}(H_X(G_{\text{enc}}(F(y))))\right\|_1^{\text{sum}}\right]\\&\quad+\left[\left\|H_{yr}(H_Y(F_{\text{enc}}(y)))-H_{yf}(H_Y(F_{\text{enc}}(G(x))))\right\|_1^{\text{sum}}\right],\end{aligned}\tag{14}$$

where x, y, r, and f refer to the true and false images in the domain X, the true and false images in the domain Y, respectively, and sum means that they are added together. This DCLGAN that adds similarity loss to the objective function is called SimDCL, which can be a satisfactory solution to the pattern collapse problem.

3.6. F-LSeSim. To alleviate the problem of scene structure discrepancies, previous approaches have attempted to achieve this by using pixel-level image reconstruction losses [52, 56, 57], cyclic consistency losses, or feature-level perceptual losses [58, 59] and PatchNCE losses, but the domain-specific nature of these losses hinders the transformation across large domain gaps. Zheng et al. [50] proposed F-LSeSim, which designs a domain-invariant representation to accurately represent the scene structure instead of using raw pixels or features of coupled appearance and structure.

First, a simple network, for example, VGG16 [60], is used to perform feature extraction on the image x in domain X and the translated image $\hat{y}$ to obtain f_x and f_y, respectively. The self-similarity is calculated using the spatial correlation mapping as follows:

$$S_{x_i}=\left(f_{x_i}\right)^T(f_{x*}),\tag{15}$$

where $\left(f_{x_i}\right)^T\in\mathbb{R}^{1\times C}$ is the feature of the query point x_i with C channels, $f_{x*}\in\mathbb{R}^{C\times N_p}$ is the corresponding feature contained in the patch of N_p points, and $S_{x_i}\in\mathbb{R}^{1\times N_p}$ captures the feature spatial correlation between the query point and other points in the patch.

Next, the structure of the whole image is represented as a collection of multiple spatially correlated mappings $S_x = [S_{x_1}, \ldots, S_{x_s}] \in \mathbb{R}^{N_s \times N_p}$, where N_s is the number of sampled patches. Then, the multiple structural similarity mappings between the input x and the translated image $\hat{y}$ are compared as follows:

$$\mathscr{L}_s = d\left(S_x, S_{\hat{y}}\right), \tag{16}$$

where $S_{\hat{y}}$ is the corresponding spatially correlated mapping in the target domain, $d(\cdot)$ can be considered in the form of L_1 distance and cosine distance, resulting in fixed self-similarity (FSeSim).

Due to the limited generality of FSeSim, Zheng et al. [50] proposed learned self-similarity (LSeSim) to obtain a more general spatially correlated mapping. LSeSim is represented by using a form of contrast loss, similar to CUT and DCLGAN. The difference is that LSeSim creates enhanced images x_{aug} by applying structure-preserving transformations to generate similar-face slice feature pairs for self-supervised learning. Specifically, $v = S_{x_i} \in \mathbb{R}^{1 \times N_p}$ represents the spatially correlative map of the "query" patch. $v^+ = S_{\hat{x}_i} \in \mathbb{R}^{1 \times N_p}$ and $v^- \in \mathbb{R}^{K \times N_p}$ are "positive" and "negative" patch samples, respectively. The query patch is positively paired with a patch at the same location i in the enhanced image x_{aug} and negatively paired with patches sampled from other locations in the image or with patches in other images y. As a result, the contrast loss can be obtained as follows:

$$\ell_c = -\log\left[\frac{\exp(\text{sim}(v, v^+)/\tau)}{\exp(\text{sim}(v, v^+)/\tau) + \sum_{k=1}^{K}\exp(\text{sim}(v, v_n^-)/\tau)}\right], \tag{17}$$

where K denotes the number of negative patches and defaults to 255. It is worth noting that this contrast loss is only used to optimize the structural representation of the network. The spatially relevant loss of the generator is always the loss in Equation (12).

In this paper, we directly use LSeSim to conduct experiments, so we will refer to the abbreviation of the method as "LSeSim" afterward.

4. Experiments and Results

We used an RTX A4000 graphics card from the AutoDL cloud platform for our experiment. The experimental scheme consists of three steps. First, considering the lack of paired actual photos of desert and oasis of the same land, we employ the above methods to pretrain the models on the winter ⟶ summer dataset [35]. Second, the desert-to-oasis application is taken on a small sample desert-to-oasis dataset using the transfer learning method. Finally, the effectiveness of each method is evaluated by qualitative and quantitative analysis of the generated results of each method.

4.1. Datasets and Implementation Details. Numerous datasets are unaligned, including the well-known horse to zebra, apple to orange, winter to summer, etc. The difficulty of implementing GAN models on small-scale datasets is complicated since there is no publicly available dataset from desert to oasis, and it is challenging to acquire enough data.

Then, we first pretrain each GAN model using the dataset winter to summer [35] and then utilize the generated models for transfer training and testing using the dataset desert to oasis. The models are selected to be pretrained on the winter-to-summer dataset due to the similar semantic information properties that summer and oasis share.

Winter ⟶ summer: the training set in this dataset contains 1,231 summer images and 962 winter images, and the test set contains 309 summer images and 238 winter images, all collected from ImageNet [61].

Desert ⟶ oasis: we downloaded 200 pictures about desert and oasis respectively from the Internet, named desert2oasis dataset, where the training set contains 100 pictures of desert and 100 pictures of oasis, and the test set also contains 100 pictures of deserts and 100 pictures of oasis. Because these images are randomly downloaded desert images from the Internet, the dataset we constructed has a certain degree of randomness, diversity, and generalizability.

We perform a simple preprocessing on each desert image before these images are input into the GAN model. The preprocessing includes two steps: each image is cropped to a square size and then scaled uniformly to the size of 256×256. The advantages of using fixed-size images include improving the robustness of the model, avoiding the overfitting of the model, and making the model training time controllable.

4.2. Metrics. We focus on qualitative and quantitative analysis to evaluate the quality and authenticity of the generated green space images. On the one hand, we use two indexes, Fréchet inception distance (FID) and learned perceptual image patch similarity (LPIPS), to evaluate the generation effect of GAN models. On the other hand, we conducted a qualitative human evaluation, which can be found in the first paragraph of Section 4.3.

FID [62] is a metric for evaluating the quality of the generated images and is based on inception scores [63]. The main idea of FID is that since the pre-trained network model can extract sample feature information, then extract the real sample and the generated sample feature information, respectively, assume that the features conform to a multivariate Gaussian distribution, and then calculate the Fréchet distance between the distributions. A lower FID means a lower Fréchet distance between the real image and the generated image, the higher the quality and diversity of the generated image.

LPIPS [64] constructs a perceptual similarity dataset and uses this dataset to train a perceptual network to calculate the difference between the generated image and the real image from different levels of features, respectively. The perceptual similarity dataset contains both real and distorted images, so LPIPS is more robust in evaluating the generated images with

Table 2: Values of FID, LPIPS, and runtime of experimental results of different GAN models.

Method	FID	LPIPS	Runtime (each epoch) (s)
CycleGAN	135.3	**0.7051**	18
GcGAN	**132.2**	0.7078	20
CUT	153.2	0.7056	18
FastCUT	179.4	0.7221	**14**
DCLGAN	174.0	0.7333	25
SimDCL	174.2	0.7520	23
LSeSim	244.7	0.7101	22

LPIPS, learned perceptual image patch similarity. Bold numbers represent the optimal values for the corresponding indicators.

different degrees of realism. LPIPS measures the image quality in terms of the similarity between features, and a smaller value indicates a more realistic generated image.

4.3. Result. Some experimental results obtained by the different methods on the desert2oasis dataset are shown in Figure 1. From the figure, we can find that although the final styles of green space are very different, each method can effectively realize image transfer learning from desert images to oases; the images produced by the same model for various desert images have diverse esthetics. Among the seven methods, only LSeSim shows the incomplete greening effect. Because of the "Mode collapse" problem, where the diversity of the generated image samples gets low, and the style is homogeneous, LSeSim has the output image style consistent for diverse input images.

From a quantitative point of view, the values of FID metrics and LPIPS metrics using those seven methods are listed in Table 2. From the table, we found that the FID values of the green field images generated by GcGAN and CycleGAN are the smallest among seven methods; according to the index, LPIPS, these three methods, CycleGAN, CUT, and GcGAN, perform well. Their LPIPS values are all less than 0.708. In terms of runtime, FastCUT/CUT, CycleGAN, and GcGAN all have relatively short runtimes, allowing for faster training based on good results.

In general, GcGAN and CycleGAN have achieved better experimental results than the other five methods in practicing the transfer learning from desert to oasis. The generated green space image represents a possible future state of this place and can be at least applied to digital media, digital entertainment, and virtual reality.

Compared to previous research in the field of image style transformation and GAN model evaluation, the focus of this paper is to pretrain the GAN model using existing datasets with limited datasets and then train it on a small-scale desert oasis dataset, thus using transfer learning to turn desert images into oasis images. The significant improvement or advancement in this paper lies in the establishment of the algorithm for transfer learning in the application scenario of a desert to an oasis. Given that desert images and oasis images are not paired, a pretraining strategy for transfer learning from winter to summer is proposed.

4.4. Discussion. Obviously, the experimental results of transferring deserts into oases using those seven methods show some shortcomings. For example, among our experimental results, three original desert images and their generated oasis images under different GAN models are shown in Figure 2. In the first row of Figure 2, Each GAN model generates a different kind of oasis image, maintains the semantic details of the original images, such as the shape and location of the sky and desert, and generates the desired oasis style. However, in the second row, only the oasis images generated by the FastCUT model keep the shape and location of the sky and desert, while the other GAN models change the shape of the sky, and some even generate the color of the desert. And it can be found that this phenomenon is not isolated, but many oasis images generated from the original desert images have this problem.

Another noteworthy issue in the third row of Figure 2 is that since our dataset is trained from the original desert to the oasis, the dataset contains almost no images of people or objects in the desert, so it leads to the situation that images of people and objects in the desert are trained by the GAN model to turn persons or objects into an oasis as well, which is caused by the distribution characteristics of the training dataset. When the dataset is too large, dimensionality reduction techniques [65] or random sampling techniques could be taken into account.

In addition, the effect of the generated oasis images is not as excellent as the expected effect since the FID metric value of each GAN model is not less than 130, while the metric value of LPIPS is not less than 0.7. To overcome some of the aforementioned problems, one research direction worth trying is to analyze image features [66] and identify objects in desert scenes with adaptive histogram technique [67], as well as global optimization [68].

Although the visual effect is worth celebrating, our method still has its limitations. The one is that there are no pair-wise datasets, and the dataset used in our model is not large enough, which will reduce transfer learning effectiveness and efficiency. The other is that it is not possible to get the specific desired style because the images generated by the model of GAN are random. When the used desert images do not conform to the norm, the generated oasis may not meet the expectations.

5. Conclusion and Future Work

In order to show the effect of turning a desert image into an oasis image using transfer learning, seven GAN models are selected and experimented. Experimental results show that both GcGAN and CycleGAN are suitable for achieving this goal according to the quantitative indexes, FID and LPIPS. In addition, The strategy of transfer learning was carefully designed; that is, the pretraining strategy of transfer learning from winter to spring was adopted; a small data set, including both desert images and green field images, was also established for model training.

The application scenario of the desert-to-oasis in this paper belongs to the small-scale dataset. GAN models work mainly by learning real sample distributions and requiring enough real samples for training to perform well [45], so

Figure 2: Several examples with large differences in experimental results using different methods.

the research about using small-scale datasets to get better results is challenging and significant. Although there has been some research on GAN models for small dataset problems, such as transferring GAN (TGAN) [45], OST [43], TuiGAN [44], etc., research on implementing unsupervised learning with extremely limited data is yet to be further developed. In this paper, we choose the TGAN-like idea to pretrain the GAN models using an existing dataset and then train it on a small-scale desert oasis dataset. Therefore it can be worth discussing and investigating what kind of existing dataset is selected for pretraining as a way to improve the generation effect.

GAN models also have high requirements for the quality of datasets; for example, in the desert-to-oasis scenario, a high-quality dataset means that the pictures of the desert and oasis should preferably have only desert and oasis rather than other factors such as people or other objects. However, high-quality datasets are often difficult to obtain. Thus it will be a future research direction to investigate which factors affect the model generation effect and how to circumvent the poorer generated images from low-quality sample data to reduce the high requirement for dataset quality [69]. In the previous discussion, it was found that the sky, as well as people or objects in the desert images, are factors that cause poor generation results, so it can be considered to use the image semantic information segmentation method to process the images first and then train them, which may achieve better results.

The style of the images generated by the GAN model is not controllable. For example, some of the generated oasis images have a lake, while others do not. Furthermore, in real-scenario applications, there is a vast demand to control image generation with specific attributes or features according to the actual needs and combined with the user intention [54]. For instance, in the desert-to-oasis scenario, the lake in the oasis is generated at random since it may be discovered that some generated images contain the lake while others do not, depending on the outcomes. Alternately, create the desired style of oasis, including the variety of flora, whether the surrounding greenery is made up of grassland or shrubs, etc. Consequently, there are two potential future research directions for this study; the first is to investigate what factors affect the model generation effect and how to circumvent the poorer images generated from low-quality sample data to reduce the high demand for dataset quality; the second is to learn how to guide GAN methods to generate image samples with specific forms, effects, and effects.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under grant no. 61571046.

References

[1] X. Su, J. Song, C. Meng, and S. Ermon, "Dual diffusion implicit bridges for image-to-image translation," in *The Eleventh International Conference on Learning Representations (ICLR)*, pp. 1–18, OpenReview.net, 2023.

[2] W. Zhang, C. Cao, S. Chen, J. Liu, and X. Tang, "Style transfer via image component analysis," *IEEE Transactions on Multimedia*, vol. 15, no. 7, pp. 1594–1601, 2013.

[3] X. Chen and R. C. Jia, "An overview of image-to-image translation using generative adversarial networks," in *Pattern Recognition. ICPR International Workshops and Challenges*, A. Del Bimbo, R. Cucchiara, S. Sclaroff et al., Eds., vol. 12666 of *Lecture Notes in Computer Science*, pp. 366–380, Springer, 2021.

[4] Y. Chen, Y. Zhao, W. Jia, L. Cao, and X. Liu, "Adversarial-learning-based image-to-image transformation: a survey," *Neurocomputing*, vol. 411, pp. 468–486, 2020.

[5] B. Espejo-Garcia, N. Mylonas, L. Athanasakos, E. Vali, and S. Fountas, "Combining generative adversarial networks and

agricultural transfer learning for weeds identification," *Biosystems Engineering*, vol. 204, pp. 79–89, 2021.

[6] C. Huang, G. Zhang, J. Yao et al., "Accelerated environmental performance-driven urban design with generative adversarial network," *Building and Environment*, vol. 224, Article ID 109575, 2022.

[7] H. Nazki, S. Yoon, A. Fuentes, and D. S. Park, "Unsupervised image translation using adversarial networks for improved plant disease recognition," *Computers and Electronics in Agriculture*, vol. 168, Article ID 105117, 2020.

[8] C. Sun, Y. Zhou, and Y. Han, "Automatic generation of architecture facade for historical urban renovation using generative adversarial network," *Building and Environment*, vol. 212, Article ID 108781, 2022.

[9] A. N. Wu, R. Stouffs, and F. Biljecki, "Generative adversarial networks in the built environment: a comprehensive review of the application of GANs across data types and scales," *Building and Environment*, vol. 223, Article ID 109477, 2022.

[10] H. Xu, C. Li, M. M. Rahaman et al., "An enhanced framework of generative adversarial networks (EF-GANs) for environmental microorganism image augmentation with limited rotation-invariant training data," *IEEE Access*, vol. 8, pp. 187455–187469, 2020.

[11] M. Brahimi, K. Boukhalfa, and A. Moussaoui, "Deep learning for tomato diseases: classification and symptoms visualization," *Applied Artificial Intelligence*, vol. 31, no. 4, pp. 299–315, 2017.

[12] A. Jiang, M. A. Nacenta, and J. Ye, "VisuaLizations as intermediate representations (VLAIR): an approach for applying deep learning-based computer vision to non-image-based data," *Visual Informatics*, vol. 6, no. 3, pp. 35–50, 2022.

[13] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer et al., "Deep learning with convolutional neural networks for EEG decoding and visualization," *Human Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, 2017.

[14] C. L. Chowdhary, M. Alazab, A. Chaudhary, S. Hakak, and T. R. Gadekallu, *Computer Vision and Recognition Systems Using Machine and Deep Learning Approaches: Fundamentals, Technologies and Applications*, Institution of Engineering and Technology, 2021.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, vol. 27, pp. 2672–2680, 2014.

[16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[17] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," ICLR, 2016.

[18] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 214–223, PMLR, 2017.

[19] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein GANs," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 5769–5779, Curran Associates Inc., 2017.

[20] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2794–2802, IEEE, 2017.

[21] J. Li, W. Monroe, T. Shi, S. Jean, A. Ritter, and D. Jurafsky, "Adversarial learning for neural dialogue generation," 2017.

[22] L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: sequence generative adversarial nets with policy gradient," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 2852–2858, AAAI Press, 2017.

[23] W. Hu and Y. Tan, "Generating adversarial malware examples for black-box attacks based on GAN," 2017.

[24] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pp. 1060–1069, PMLR, 2016.

[25] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European Conference on Computer Vision (ECCV)*, pp. 649–666, Springer, 2016.

[26] C. Ledig, L. Theis, F. Huszár et al., "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4681–4690, IEEE, 2017.

[27] D. Chen, L. Yuan, J. Liao, N. Yu, and G. Hua, "Stereoscopic neural style transfer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6654–6663, IEEE, 2018.

[28] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, "Image analogies," in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 327–340, Association for Computing Machinery, 2001.

[29] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Proceedings of the Seventh IEEE International Conference on Computer Vision (ICCV)*, vol. 2, pp. 1033–1038, IEEE, 1999.

[30] J. Kim, M. Kim, H. Kang, and K. Lee, "U-GAT-IT: unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation," in *8th International Conference on Learning Representations*, pp. 1–18, OpenReview.net, 2020.

[31] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, "Learning to discover cross-domain relations with generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, pp. 1857–1865, PMLR, 2017.

[32] M. Li, H. Huang, L. Ma, W. Liu, T. Zhang, and Y. Jiang, "Unsupervised image-to-image translation with stacked cycle-consistent adversarial networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 184–199, Springer, 2018.

[33] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 700–708, Curran Associates Inc., 2017.

[34] Z. Yi, H. Zhang, P. Tan, and M. Gong, "DualGAN: unsupervised dual learning for image-to-image translation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2849–2857, IEEE, 2017.

[35] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2223–2232, IEEE, 2017.

[36] H. Fu, M. Gong, C. Wang, K. Batmanghelich, K. Zhang, and D. Tao, "Geometry-consistent generative adversarial networks for one-sided unsupervised domain mapping," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2427–2436, IEEE, 2019.

[37] W. Wu, K. Cao, C. Li, C. Qian, and C. C. Loy, "Transgaga: geometry-aware unsupervised image-to-image translation," in

Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8012–8021, IEEE, 2019.

[38] Y. Zhao, R. Wu, and H. Dong, "Unpaired image-to-image translation using adversarial consistency loss," in *European Conference on Computer Vision (ECCV)*, pp. 800–815, Springer, 2020.

[39] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "StarGAN v2: diverse image synthesis for multiple domains," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 8188–8197, IEEE, 2020.

[40] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 172–189, IEEE, 2018.

[41] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang, "Diverse image-to-image translation via disentangled representations," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 35–51, Springer, 2018.

[42] H.-Y. Lee, H.-Y. Tseng, Q. Mao et al., "DRIT++: diverse image-to-image translation via disentangled representations," *International Journal of Computer Vision*, vol. 128, pp. 2402–2417, 2020.

[43] S. Benaim and L. Wolf, "One-shot unsupervised cross domain translation," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 2108–2118, Curran Associates Inc., 2018.

[44] J. Lin, Y. Pang, Y. Xia, Z. Chen, and J. Luo, *Tuigan: learning versatile image-to-image translation with two unpaired images*, in *European Conference on Computer Vision–ECCV 2020*, pp. 18–35, Springer, 2020.

[45] Y. Wang, C. Wu, L. Herranz, J. van de Weijer, A. Gonzalez-Garcia, and B. Raducanu, "Transferring GANs: generating images from limited data," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 218–234, Springer, 2018.

[46] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," 2015.

[47] S. Benaim and L. Wolf, "One-sided unsupervised domain mapping," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 752–762, Curran Associates Inc., 2017.

[48] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu, "Contrastive learning for unpaired image-to-image translation," in *Computer Vision – ECCV 2020*, pp. 319–345, Springer, 2020.

[49] J. Han, M. Shoeiby, L. Petersson, and M. A. Armin, "Dual contrastive learning for unsupervised image-to-image translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 746–755, IEEE, 2021.

[50] C. Zheng, T.-J. Cham, and J. Cai, "The spatially-correlative loss for various image translation tasks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16407–16417, IEEE, 2021.

[51] L. J. Ratliff, S. A. Burden, and S. S. Sastry, in *Characterization and computation of local Nash equilibria in continuous games*pp. 917–924, IEEE, 2013.

[52] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1125–1134, IEEE, 2017.

[53] I. Goodfellow, "Nips 2016 tutorial: generative adversarial networks," Nips, 2016.

[54] Y. Pang, J. Lin, T. Qin, and Z. Chen, "Image-to-image translation: methods and applications," IEEE Transactions on Multimedia, 2021.

[55] A. van dan Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018.

[56] Q. Chen and V. Koltun, "Photographic image synthesis with cascaded refinement networks," in *Proceedings of the IEEE International Conference on Computer Vision (ECCV)*, pp. 1511–1520, IEEE, 2017.

[57] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and Russ Webb Apple Inc, "Learning from simulated and unsupervised images through adversarial training," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2107–2116, IEEE, 2017.

[58] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 658–666, Curran Associates Inc., 2016.

[59] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Computer Vision – ECCV 2016*, pp. 694–711, Springer, 2016.

[60] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations (ICLR)*, pp. 1–14, OpenReview.net, 2015.

[61] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: a large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, IEEE, 2009.

[62] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "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*, pp. 6629–6640, Curran Associates Inc., 2017.

[63] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 2234–2242, Curran Associates Inc., 2016.

[64] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595, IEEE, 2018.

[65] G. Thippa Reddy, M. Praveen Kumar Reddy, K. Lakshmanna et al., "Analysis of dimensionality reduction techniques on big data," *IEEE Access*, vol. 8, pp. 54776–54788, 2020.

[66] R. Kaluri and C. H. Pradeep Reddy, "Optimized feature extraction for precise sign gesture recognition using self-improved genetic algorithm," *International Journal of Engineering and Technology Innovation*, vol. 8, no. 1, pp. 25–37, 2018.

[67] R. Kaluri and C. H. Pradeep Reddy, "An enhanced framework for sign gesture recognition using hidden Markov model and adaptive histogram technique," *International Journal of Intelligent Engineering & Systems*, vol. 10, no. 3, pp. 11–19, 2017.

[68] C. Nour and V. Zeidan, "Optimal control of nonconvex sweeping processes with separable endpoints: nonsmooth maximum principle for local minimizers," *Journal of Differential Equations*, vol. 318, pp. 113–168, 2022.

[69] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, "Generative adversarial networks: introduction and outlook," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 588–598, 2017.