TG-TSGNet: A Text-Guided Arbitrary-Resolution Terrain Scene Generation Network

Yifan Zhu, Yan Wang, Xinghui Dong, *Member, IEEE*

***Abstract*—With the increasing demand for terrain visualization in many fields, such as augmented reality, virtual reality and geographic mapping, traditional terrain scene modeling methods encounter great challenges in processing efficiency, content realism and semantic consistency. To address these challenges, we propose a Text-Guided Arbitrary-Resolution Terrain Scene Generation Network (TG-TSGNet), which contains a ConvMamba-VQGAN, a Text Guidance Sub-network and an Arbitrary-Resolution Image Super-Resolution Module (ARSRM). The ConvMamba-VQGAN is built on top of the Conv-Based Local Representation Block (CLRB) and the Mamba-Based Global Representation Block (MGRB) that we design, to utilize local and global features. Furthermore, the Text Guidance Sub-network comprises a text encoder and a Text-Image Alignment Module (TIAM) for the sake of incorporating textual semantics into image representation. In addition, the ARSRM can be trained together with the ConvMamba-VQGAN, to perform the task of image super-resolution. To fulfill the text-guided terrain scene generation task, we derive a set of textual descriptions for the 36,672 images across the 38 categories of the Natural Terrain Scene Data Set (NTSD). These descriptions can be used to train and test the TG-TSGNet[1]. Experimental results show that the TG-TSGNet outperforms, or at least performs comparably to, the baseline methods in image realism and semantic consistency with proper efficiency. We believe that the promising performance should be due to the ability of the TG-TSGNet not only to capture both the local and global characteristics and the semantics of terrain scenes, but also to reduce the computational cost of image generation.**

***Index Terms*—Text-to-image generation, terrain scenes, super-resolution, Mamba, autoregressive models.**

I. Introduction

TERRAIN scene images have a wide range of applications in the fields of augmented reality [1], virtual reality [2] and geographic mapping [3]. As one of the visually impactful elements in natural scenes, terrains not only visualize environmental characteristics but also play an important role in the immersion and experience of observers. However, the acquisition of natural terrain scene images faces many challenges, particularly in complex geographic environments. Due to the harsh geographic conditions and high equipment costs, not only the acquisition of natural terrain scene images is expensive, time-consuming, labor-intensive and even dangerous, but also the coverage and resolution are often difficult to satisfy the actual requirements, which makes collection of a large number of high-quality terrain scene images very difficult. Therefore, terrain scene image generation techniques have become an alternative solution.

Fig. 1. Examples of the terrain scene images generated by the TG-TSGNet at various resolutions, such as 256×512, 300×400, 512×1024, etc.

Traditional terrain image generation methods normally were developed on top of fractal models [4, 5, 6] and physical erosion models [7, 8]. Although fractal models can efficiently generate self-similar terrains using mathematical algorithms, the images lack physical realism and diversity. In contrast, physical erosion models are able to generate more realistic terrain images by simulating natural erosion processes. However, the computational overhead of these models is more expensive. In addition, the above methods suffer from limitations in flexibility and cannot precisely control the specific characteristics of the terrain in terms of the requirements of users. As a result, the customized and controllable generation demanded in practical applications is difficult to realize.

With the rapid advancement of deep learning techniques, terrain images can be synthesized using deep image generation methods [9, 10, 11, 12]. Although these methods can generate more realistic and diverse terrain images by learning from a large number of terrain images, they still encounter challenges, such as high computational cost and fixed resolution. For example, the Vector Quantized Generative Adversarial Network (VQGAN) [12] severely relies on the moving-window sampling and decoding operation in order to generate a high-resolution image, and is thus computationally inefficient.

Wang et al. [10] introduced a Lightweight VQGAN, i.e.,

This study was supported by the National Natural Science Foundation of China (NSFC) (No. 42176196) (Corresponding author: Xinghui Dong).

Y. Zhu and X. Dong are with the State Key Laboratory of Physical Oceanography and the Faculty of Information Science and Engineering, Ocean University of China, Qingdao, 266100. (e-mail: zhuyifan@stu.ouc.edu.cn, xinghui.dong@ouc.edu.cn). Y. Wang was with the Faculty of Information Science and Engineering, Ocean University of China, Qingdao, 266100. (e-mail: wangyan6183@stu.ouc.edu.cn).

[1]The data set, model and source code are available at https://github.com/INDTLab/TG-TSGNet.

Lit-VQGAN, to address this problem using a super-resolution network. However, this network cannot be end-to-end trained with the Lit-VQGAN and cannot generate arbitrary-resolution (see Fig. 1) images. As a result, inconsistent image quality may be produced at different resolutions. In this situation, the existing deep learning-based image generation methods are probably not competent for generation of high-realism images.

On the other hand, the alignment between textual descriptions and images is normally challenging for text-to-image generation methods. For instance, Xu et al. [13] proposed an attention mechanism in order to improve text-image consistency while it struggled with aligning fine-grained textual details with specific image regions. In [14], semantic consistency was enhanced without extra captioning modules. However, complex spatial relationships that the text described were often missed. Moreover, the DALL-E [15] method tended to exhibit semantic drift when handling complex textual descriptions.

To address the above challenges, we propose a Text-Guided Arbitrary-Resolution Terrain Scene Generation Network, referred to as TG-TSGNet. This network consists of a ConvMamba-VQGAN, a Text Guidance Sub-network and an Arbitrary-Resolution Image Super-Resolution Module (ARSRM). Specifically, the ConvMamba-VQGAN follows the framework of the VQGAN [12]. However, both the encoder and decoder are built on top of the Conv-Based Local Representation Block (CLRB) and Mamba-Based Global Representation Block (MGRB). The two blocks are designed to extract local and global image features, respectively. Furthermore, the Text Guidance Sub-network contains a pre-trained text encoder and a Text-Image Alignment Module (TIAM), to inject textual semantics into image representation. Moreover, we adopt the ARSRM based on the Neural Operator [16] to overcome the bottleneck of high-resolution image generation. In contrast to the VQGAN [12] and Lit-VQGAN [10], our TG-TSGNet not only discards inefficient moving-window sampling and decoding by realizing end-to-end arbitrary-resolution[2] generation (see Fig. 1), but also captures both the local and global characteristics and the semantics of terrain scenes.

We further augment the Natural Terrain Scene Data Set (NTSD) [10] with a set of textual descriptions, namely, NTSD-TD, to fulfill the text-image terrain scene generation task. This data set associates each of the 36,672 real-world terrain scene images with a deliberately crafted textual description. Due to these high-quality terrain-specific descriptions, the utility of the NTSD for cross-modal learning becomes practical.

To our knowledge, terrain scene generation has not been performed using such a text-guided arbitrary-resolution generation manner. The contributions of this research can be summarized as fourfold.

- We introduce a Text-Guided Arbitrary-Resolution Terrain Scene Generation Network, i.e., TG-TSGNet. In particular, an Arbitrary-Resolution Image Super-Resolution Module (ARSRM) is adopted for the sake of generating images at various resolutions. Compared with the baseline methods, the TG-TSGNet can generate arbitrary-resolution images with the higher, or at least comparable, realism and semantic consistency at a proper computational cost.
- We build a ConvMamba-VQGAN on the basis of the Conv-Based Local Representation Block (CLRB) and the Mamba-Based Global Representation Block (MGRB). Both the blocks enable this network to efficiently learn local features and long-range dependencies of terrain scenes, respectively. As a result, the model trained is able to capture complicated terrain characteristics.
- To align text semantics and image characteristics, we design a Text-Image Alignment Module (TIAM). By combining the self-attention mechanism with Mamba, this module not only models the joint distribution of texts and images, but also reduces the number of parameters and computational complexity.
- We build a Natural Terrain Scene Textual Description Data Set, namely, NTSD-TD. This data set contains high-quality textual descriptions for the 36,672 terrain scene images, allowing the text-to-image terrain scene generation task to become a reality.

The structure of this paper is organized as follows. In Section II, we review the related literature. Our textual description data set is introduced in Section III. In Section IV, the proposed TG-TSGNet is presented. The experimental setup and results are introduced in Sections V and VI, respectively. Finally, we draw our conclusion in Section VII.

II. Related Work

A. Text-to-Image Generation

Many methods were introduced for text-to-image generation, aiming at generating images that accurately reflect the meaning of a given textual description. As the first method designed to generate images from natural language, AlignDRAW [17] evolved from the text-conditional GAN [18], which were followed by more studies [19, 20]. Autoregressive (AR) model-based approaches, e.g., DALL-E [15] and Pixart-σ [11], typically used a sequence-to-sequence generation manner to gradually predict pixels or latent codes of an image, which achieved precise text-to-image parsing and realistic image generation. Recently, Diffusion Models (DMs), such as Stable Diffusion [9], GLIDE [21] and DALL-E2 [22], have attracted attention due to the stepwise generation process. These methods controlled text conditioning using a pre-trained text encoder and flexibly manipulated the generated content by introducing negative prompts and multi-modal conditioning. To improve scalability and representation ability, Mamba and wavelet transforms were used in conjunction with diffusion models [23].

None of the aforementioned methods were focused on text-to-image generation of natural terrain scenes. To fulfill this task, we derived the NTSD-TD for the 36,672 real-world terrain scene images [10]. Those methods normally suffered from high computational complexity and lack of arbitrary-resolution generation ability. In contrast, our TG-TSGNet avoids the inefficient moving-window generation operation and supports

[2]The term "arbitrary-resolution" used in this paper refers to the ability to generate images at various resolutions under the ideal hardware configurations.

arbitrary-resolution image generation by introducing the ARSRM. In addition, we designed the TIAM, which produced the higher semantic consistency between textual descriptions and images, compared to the existing methods.

B. Latent Space-Based Generation

Typically, latent space-based generation methods were adopted on top of a two-stage strategy [9, 15, 24, 25]. The Vector Quantized Variational Autoencoder (VQVAE) [26] pioneered this strategy to learn discrete representations of images. The VQVAE-2 [27] enhanced the ability to model complex structures by designing a hierarchical latent space. In [12], the VQGAN was introduced on the basis of VQVAE, to optimize the latent representation using both the combating loss and perceptual loss functions. However, the moving-window sampling and decoding operation was computationally inefficient. To address this issue, Razavi et al. [27] proposed a Masked Generative Image Transformer (MaskGIT) using a two-way Transformer instead of the AR model. Li et al. [28] further introduced a pure Mamba-based autoregressive framework, i.e., Scalable Autoregressive Image Generation with Mamba (AiM), for scalable image generation. This modification improved the efficiency of the generation operation.

Progressive generation using diffusion models has also been integrated into two-stage approaches. In [29], the diffusion process was modeled in the latent space for the VQ-Diffusion model, which enabled precise control over the details of the images generated while improving the quality of these images. The Latent Diffusion Model (LDM) [9] conducted diffusion modeling in the continuous latent space, which reduced computational cost and could generate higher-resolution images. In contrast, the control conditions of the pre-trained model was incorporated into the latent space for the ControlNet [30]. Nevertheless, the depth of exploration and interpretability of the latent space were still poor.

Compared with these methods, the ConvMamba-VQGAN included in the proposed TG-TSGNet is able to learn discrete representations of images by integrating both local and global features, which enhances the quality of image reconstruction. Besides, the ARSRM enables end-to-end super-resolution without relying on the computationally intensive moving-window generation, thereby achieving the more flexible and efficient arbitrary-resolution generation.

C. Implicit Neural Representations for Super-Resolution

Image Super-Resolution (SR) aims to recover high-resolution images from a low-resolution image. Arbitrary-resolution image super-resolution approaches were developed on top of interpolation [31], meta-learning [32], dynamic convolution [33] and implicit neural representations [16, 34, 35]. Since implicit neural representations could map discrete values to continuous representations, they were applied to modeling the shape and appearance of 3D objects. Compared to discrete representations, continuous implicit representations capture fine shape details with fewer parameters.

Inspired by this finding, implicit neural representations were introduced into the Local Implicit Image Function (LIIF) [16].

Fig. 2. Twelve images contained in the NTSD [10] and the corresponding textual descriptions that we annotated in this study. It can be seen that diverse terrain categories and concise, accurate textual descriptions are included in the NTSD [10] and NTSD-TD, respectively.

This function achieved continuous representation by mapping the discrete pixel values to continuous values. Consequently, the LIIF realized arbitrary-resolution image super-resolution. In [34], the spatial coding technique was proposed, which improved the accuracy of high-frequency texture prediction. Lee and Jin [29] designed a Local Texture Estimator (LTE) by bringing together a frequency estimation module, allowing implicit functions to capture high-frequency details. In [35], implicit neural representation was inserted into the Super-Resolution-Neural-Operator (SRNO) for high-frequency texture and edge processing.

In contrast, we adopted the ARSRM by integrating the multi-scale features extracted using the encoder with the neural operator, which enhanced the ability to capture image details. In addition, the ARSRM can be jointly trained with the ConvMamba-VQGAN, which optimizes the parameters of the encoder with regard to the generation and super-resolution tasks. Therefore, the performance of both the tasks conducted using our TG-TSGNet is further improved.

III. Natural Terrain Scene Textual Description Data Set

This study was focused on generating images of the natural terrain scene in terms of a textual description. To our knowledge, however, there is not a publicly available terrain text-image pair data set. The scarcity of such a data set poses a challenge to the task of text-to-image terrain scene generation. To overcome this challenge, we deliberately derived a set of textual descriptions for the 36,672 images included in the Natural Terrain Scene Data Set (NTSD) [10]. (Refer to Fig. 2 for examples). This data set is referred to as Natural Terrain Scene Textual Description Data Set, or NTSD-TD for short. Since the NTSD-TD provides terrain-specific textual annotations, text-image alignment can be learned using them, which supports the text-to-image terrain scene generation task.

TABLE I
THE 38 TERRAIN CATEGORIES OF THE NTSD [10] AND THE NUMBER OF THE IMAGES CONTAINED IN EACH CATEGORY.

Terrain Category	Number of Images	Terrain Category	Number of Images	Terrain Category	Number of Images
Adarce	1,182	Coast	1,017	Glacier	1,041
Beach	920	Coniferous Forest	963	Islands	935
Canyon	899	Danxia	1,017	Lake	1,118
Cavern	1,204	Desert	1,002	Loess Plateau	817
Earth Forest	861	Farmland	1,018	Mangrove	1,812
Flowerland	1,032	Meander	858	Mountain	880
Oasis	824	Peak Forest Plain	1,210	Prairie	830
Pier	868	Rainforest	849	Reef	1,033
River	774	Saline Soil	508	Sandbar	778
Sand Dunes	807	Sands	1,091	Sea	1,027
Snow Mountain	864	Snowfield	1,074	Stone Forest	815
Terrace	1,168	Volcano	945	Waterfall	1,001
Wetland	925	Yardang	705		

Fig. 3. Four terrain scene images contained in the NTSD [10], each of which is followed by two textural descriptions. The top description is generated using the OFA [36] model. The bottom description is further optimized to obtain the more consistent meaning with the scene, in which the red font indicates the details that have been appended while the blue font indicates the text that has been modified.

A. Natural Terrain Scene Data Set

The images used in this study were directly sourced from the NTSD [10], which covered a wide variety of land and oceanic terrains, e.g., Desert, Glacier and Pier, and unique landforms, e.g., Danxia and Yadan. They were divided into 38 categories. The details of these categories and the number of each category are shown in Table I. In contrast to aerial or satellite data sets, the NTSD [10] contains ground-level photographs that capture intricate details of terrain morphology, lighting and contextual characteristics. These details are crucial for generating realistic terrain scenes from a textual description. In Fig. 2, an image of 12 categories is presented along with the textual description.

B. Annotation and Optimization of Textual Descriptions

To support text-image cross-modal learning, we first used the pre-trained OFA [36] model to generate a raw textual description for each image. Then we inspected whether or not each description accurately described the content of the associated image by manual screening. It was found that the generated descriptions often suffered from problems, such as hallucination and omission of the background information.

We further manually optimized each textual description in order to guarantee that it matched the scene in semantics. We also modified the descriptions using some terrain terminologies for the sake of ensuring that they accurately reflected the terrain categories, geographic characteristics, subject-object landscapes and other key elements shown in the images. Finally, the optimized textual descriptions have an average length of 9.59 words, with a vocabulary size of 2,563 unique words. In terms of four terrain scene images [10], the descriptions generated using the OFA [36] model and those optimized by the authors are compared in Fig. 3. While maintaining the conciseness of the text, we managed to retain the transmission of information. Thus, high-quality textual descriptions were derived which can be used for cross-modal learning.

IV. TEXT-GUIDED ARBITRARY-RESOLUTION TERRAIN SCENE GENERATION NETWORK

With regard to existing image generation methods, the computational cost is usually very high while images are generated at a fixed resolution. Since this study aims to generate realistic terrain scene images from a textual description, image realism and the modality gap between textual descriptions and terrain images are also challenging. To address the above challenges, we introduce a Text-Guided Arbitrary-Resolution Terrain Scene Generation Network, referred to as TG-TSGNet, motivated by the two-stage strategy [12, 26]. This network comprises a ConvMamba-VQGAN, a Text Guidance Subnetwork and an Arbitrary-Resolution Image Super-Resolution Module (ARSRM). In particular, the ARSRM is designed for the sake of reducing the computational cost of the generation process. Compared with the two-stage methods [12] that repeatedly perform sampling and decoding in a moving-window manner, our TG-TSGNet only runs this operation once and then uses the ARSRM to generate an image at the arbitrary resolution. Fig. 4 shows the pipeline of the TG-TSGNet.

A. ConvMamba-VQGAN

Inspired by the success that the VQGAN [12] has achieved, we introduce a ConvMamba-VQGAN, which consists of a ConvMamba-VQVAE and a discriminator. The architecture of the ConvMamba-VQGAN is presented in Fig. 5(a).

1) ConvMamba-VQVAE: The ConvMamba-VQVAE includes an encoder, a Vector Quantization (VQ) module and a decoder. The encoder comprises a Conv-Based Local Representation Block (CLRB) and four consecutive Mamba-Based Global Representation Blocks (MGRBs). Similarly, the decoder contains four MGRBs and one CLRB symmetrically. Within the bottleneck, there is the VQ module.

Conv-Based Local Representation Block. To effectively extract local features from terrain scene images, we design a lightweight Conv-Based Local Representation Block (CLRB), as illustrated in Fig. 5(b). This block consists of a depthwise separable convolution sub-block and a Multilayer Perceptron (MLP) sub-block. Each sub-block has a residual connection. The CLRB can be formulated as follows:

$$\tilde{F}_c^i = \mathrm{Conv}_{1\times1}\left(\mathrm{DWConv}\left(\mathrm{Conv}_{1\times1}\left(F_c^i\right)\right)\right) + F_c^i, \quad (1)$$

$$\tilde{\tilde{F}}_c^i = \mathrm{MLP}\left(\tilde{F}_c^i\right) + \tilde{F}_c^i, \quad (2)$$

where F_c^i denotes the feature maps fed into the i-th CLRB and $\tilde{\tilde{F}}_c^i$ represents the feature maps produced by this CLRB. Compared with the convolutional blocks that the VQGAN [12] utilizes, the CLRB contains fewer parameters.

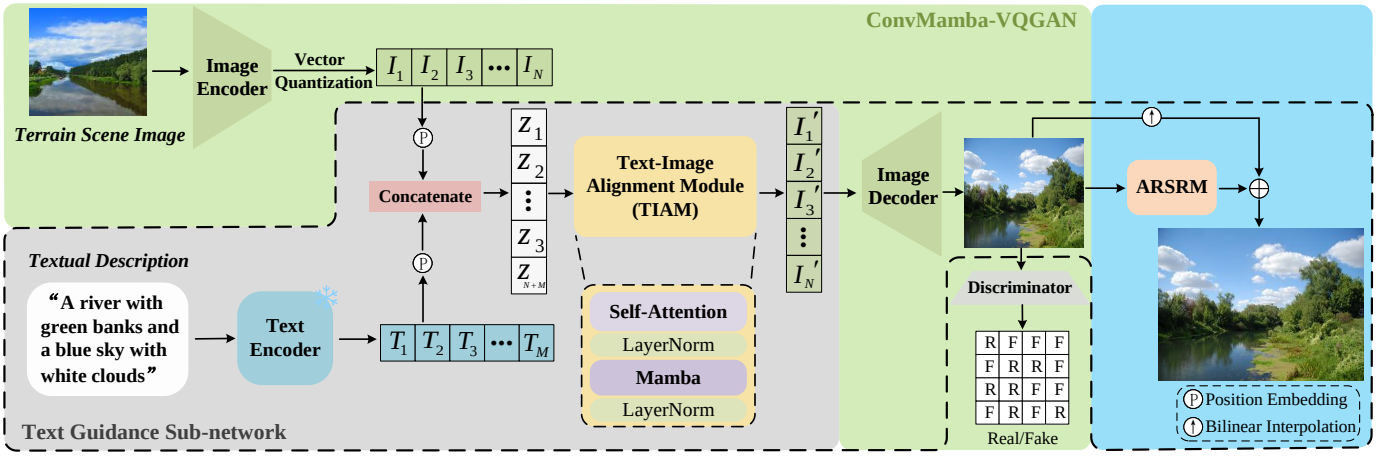

Fig. 4. Pipeline of the proposed TG-TSGNet, whose training process consists of two steps. Within the first step, the ConvMamba-VQGAN and ARSRM are end-to-end trained. During the second step, the Text Guidance Sub-network is trained with the trained ConvMamba-VQGAN and ARSRM frozen. In the inference process (indicated by the dashed box), a text token sequence and an empty image code sequence are fed into the TIAM. The image that the decoder generates is sent to the ARSRM and the resultant image is added with the upsampled image, to obtain an arbitrary-resolution image.

Mamba-Based Global Representation Block. Since terrain scene images normally manifest large-scale topological structure, long-range dependencies are particularly useful for representation of these images. However, the encoder and decoder of the VQGAN [12] only comprise convolutional layers, which cannot capture long-range dependencies well. Although Transformer can utilize these data, the computational cost is high. Recently, Mamba [37], designed based on the Structured State Space Sequence (S4) model, has attracted the attention of researchers, as it addresses the limitations of Transformer in computational complexity and long sequence modeling. Liu et al. [38] further extended it to Vision Mamba, which showed the ability to capture global spatial dependencies.

To efficiently capture long-range dependencies, we adopt a Mamba-Based Global Representation Block (MGRB) (see Fig. 5(c)) inspired by the existing work [39]. The MGRB contains a Variable State Space Module (VSSM) [38] and a Channel Attention Block (CAB) [40], along with two residual connections, respectively. Since a single VSSM may lead to the problem of local pixel forgetting and channel redundancy [38], the CAB is used to recover local characteristics. The MGRB can be expressed as follows:

$$\hat{F}_m^j = \text{LayerNorm}\left(\tilde{\tilde{F}}_c^i\right) + \text{VSSM}\left(\text{LayerNorm}\left(\tilde{\tilde{F}}_c^i\right)\right), \quad (3)$$

$$F_m^j = \text{Conv}_{1\times1}\left(\text{LayerNorm}\left(\hat{F}_m^j\right)\right) + \text{CAB}\left(\text{Conv}_{1\times1}\left(\text{LayerNorm}\left(\hat{F}_m^j\right)\right)\right), \quad (4)$$

where F_m^j represents the feature maps produced by the j-th MGRB.

Vector Quantization. We use the discrete representation learning method which has been utilized in the VQVAE [26]. The input image x is fed into the encoder. The result is a set of feature vectors, denoted as $F = [F_1, F_2, F_3, \cdots, F_N]$. Then each feature vector F_p is mapped to the closest code T_q in the terrain codebook (i.e., Terrainbook or $B_{Terrain}$), to obtain a quantized code sequence I, which can be expressed as:

$$I = \text{VQ}(F) := \text{argmin}_{F_p \in F, T_q \in B_{Terrain}} \|F_p - T_q\|. \quad (5)$$

As a result, each terrain image is uniquely represented by a sequence of codes of length N, denoted as $I = [I_1, I_2, I_3, \cdots, I_N]$ ($I_p \in B_{Terrain}$). To reconstruct the image, I is passed to the decoder $Dec(\cdot)$, which can be written as:

$$\hat{x} = Dec(I). \quad (6)$$

2) Discriminator: The patch-based discriminator [41] that Esser et al. [12] used for the VQGAN is utilized, which aims to distinguish reconstructed images from real images. The discriminator learns the subtle differences between the real and reconstructed images by performing a patch-level analysis on the input images. Therefore, the accuracy of image reconstruction is improved.

B. Text Guidance Sub-network

The Text Guidance Sub-network comprises a text encoder and a Text-Image Alignment Module (TIAM). This subnetwork extracts text tokens from the textual description and aligns these with the quantized codes, to guide the image generation process using the semantics of the description.

1) Text Encoder: In [42], it was found that the pre-trained Byte Pair Encoding (BPE) model improved the generalization of unregistered words. In addition, this model does not depend on specific language rules and has good linguistic generality, which makes it suit text encoding tasks. Therefore, we utilize the pre-trained BPE model as the text encoder. Given a textual description, it is sent to this encoder. The result is a token sequence of length M, denoted as $T = [T_1, T_2, T_3, \cdots, T_M]$.

2) Text-Image Alignment Module: Since the text token and image code sequences lie in different feature spaces, a text-image alignment operation is necessary. To establish a semantic consistency between the token and code sequences, we treat this operation as an autoregressive problem. Transformer has been widely applied to autoregressive modeling because it can

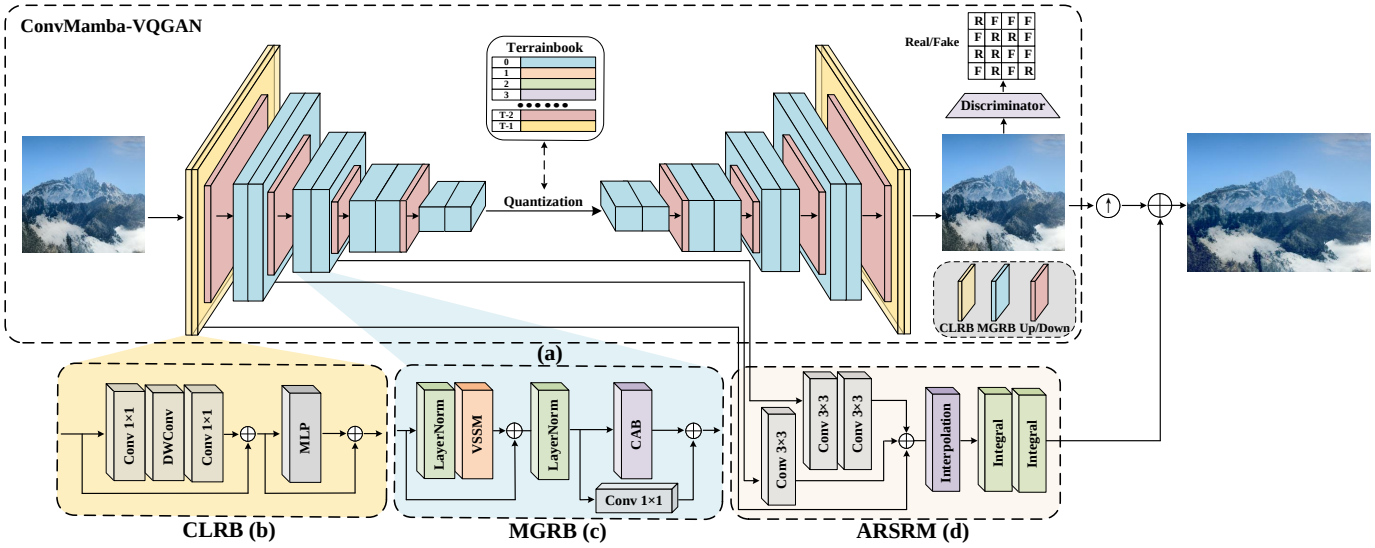

Fig. 5. The architecture of the ConvMamba-VQGAN (a), which comprises a ConvMamba-VQVAE and a discriminator. The encoder and decoder are built using the Conv-Based Local Representation Block (CLRB) (b) and Mamba-Based Global Representation Block (MGRB) (c). An Arbitrary-Resolution Image Super-Resolution Module (ARSRM) (d) is also adopted based on the Neural Operator [16], to reduce the computational cost of the generation process.

capture long-range dependencies. However, the self-attention mechanism that Transformer utilizes suffers from the quadratic complexity with respect to the length of sequences.

Although recent studies [43, 44] addressed this issue by sparsifying the attention matrix, applying low-rank approximations or using linear projections, they often compromised alignment accuracy due to the limited context modeling ability. In contrast, Mamba [37] avoids the explicit computation of the global attention matrix and achieves a linear computational complexity in terms of the length of sequences. We are motivated to propose the Text-Image Alignment Module (TIAM) (Fig. 4). This module brings together the self-attention mechanism and Mamba [37], to achieve a proper trade-off between the accuracy of alignment and computational complexity.

To encode the relative and absolute position data, we use Rotary Position Embedding (RoPE) [45] to embed the position data of the image code sequence I and the text token sequence T. They are concatenated into a single sequence, denoted as:

$$Z = [z_1, z_2, \cdots, z_{N+M}], \quad z_i \in \mathbb{R}^d. \tag{7}$$

The sequence Z is first passed through the Multi-head Self-attention (MHSA) layer, to capture long-range dependencies. The output is then processed using a Layer Normalization (LN) unit. For the purpose of capturing the sequential structure and facilitating precise semantic alignment, the output of the LN unit is further fed into a Mamba layer. Due to the causality constraints required in the autoregressive generation process, we utilize the unidirectional Mamba. This choice ensures the validity and consistency of the sequence generation.

In contrast to modeling long-range dependencies using self-attention alone, which may suffer from the context dilution issue [46], the joint use of Mamba enables the more effective retention of the historical context through its continuous state update mechanism. Hence, the TIAM exploits both the merit of self-attention in capturing cross-modal long-range dependencies and the strength of Mamba in modeling long sequences. As a result, it can achieve high alignment accuracy while keeping a relatively small number of parameters when processing long sequences. Finally, a sequence of aligned tokens, i.e., $I' = [I'_1, I'_2, I'_3, \cdots, I'_N]$, is produced, which can be fed into the image decoder to generate an image.

C. Arbitrary-Resolution Image Super-Resolution Module

To overcome the challenge of high computational complexity that two-stage methods [12] normally encounter, we apply image super-resolution to the generation process. This choice avoids computationally intensive moving-window sampling and decoding. As existing super-resolution methods [32, 33] typically process discrete pixels, the model cannot produce an arbitrary-resolution image. A solution is to learn the mapping from the Low-Resolution (LR) image to the High-Resolution (HR) image. To this end, we use the Neural Operator [16] to convert an image from the discrete pixel matrix into a continuous representation. This process can be expressed as:

$$S_\theta : H\left(\Omega_{lr}\right) \rightarrow H\left(\Omega_{hr}\right), \tag{8}$$

where S_θ denotes a Neural Operator parameterized by θ, and $H(\Omega_{lr})$ and $H(\Omega_{hr})$ represent the functions of continuous representations defined over the LR and HR domains, i.e., Ω_{lr} and Ω_{hr}, respectively.

On top of the Neural Operator, we adopt an Arbitrary-Resolution Image Super-Resolution Module (ARSRM) (see Fig. 5(d)), which contains three convolutional layers, an interpolation layer and two integral layers. Given an image x, the multi-scale features extracted at the image encoder are fused via the convolutional layers. Due to the multi-scale representation, the richer image characteristics can be captured by the ARSRM. The fused features $Enc_\varphi(x)$ are then fed into the interpolation and integral layers. The interpolation layer is used to construct a continuous representation while the integral

layer aims to learn the mapping relationship between the LR and HR images. The final HR image x_{HR} is obtained as:

$$x_{HR} = \hat{x}_{up} + ARSRM(Enc_{\varphi}(x)), \tag{9}$$

where $\hat{x}_{up}$ is the coarse HR image obtained by upsampling the generated image and $ARSRM(Enc_{\varphi}(x))$ is the super-resolution image. The fusion of the two HR images restores high-frequency details and ensures global consistency.

D. Model Training

The training operation of the TG-TSGNet is performed in two steps. During the first step, both the ConvMamba-VQGAN and ARSRM are end-to-end trained. The loss function of the ConvMamba-VQGAN consists of three components, including the reconstruction loss, perceptual loss and adversarial loss. These loss functions are defined as:

$$L_{Rec} = |x - \hat{x}|^2, \tag{10}$$

$$L_{Per} = \sum_{l} \lambda_l \left|\varphi_l(x) - \varphi_l(\hat{x})\right|_2^2, \tag{11}$$

$$L_{Adv} = \log D(x) + \log(1 - D(\hat{x})), \tag{12}$$

respectively, where $\varphi_l(x)$ and $\varphi_l(\hat{x})$ are the features extracted at the l-th layer of the pre-trained VGG-VD-16 [47] model, λ_l is the weighting factor, and $D(\cdot)$ denotes the discriminator.

The loss of the ConvMamba-VQGAN is defined as:

$$\begin{aligned} L_{GAN} = {} & L_{Rec} + L_{Per} + \lambda L_{Adv} \\ & + \|sg\,(F) - I\|_2^2 + \|sg\,(I) - F\|_2^2, \end{aligned} \tag{13}$$

where $sg\,[\cdot]$ denotes the stop-gradient operation, and λ represents an adaptive weight, which is defined as:

$$\lambda = \frac{\nabla_G \left(L_{Rec} + L_{Per}\right)}{\nabla_G L_{Adv}}, \tag{14}$$

$\|sg\,(F) - I\|_2^2$ updates the parameters of Terrainbook, and $\|sg\,(I) - F\|_2^2$ compensates for the backpropagation using the straight-through gradient estimator. On the other hand, the ARSRM uses the L_1 loss function.

Hence, the loss function used in the first step is defined as:

$$L_{First} = \beta L_{GAN} + (1 - \beta)\, L_1, \tag{15}$$

where β is the weighting coefficient that regulates the balance between the two components. Both the ConvMamba-VQGAN and ARSRM will be jointly trained using this loss function.

Within the second step, the Text Guidance Sub-network is trained with the weights in the encoder, decoder, Terrainbook and ARSRM are frozen. Given the text sequence T, the text-image alignment task is considered an autoregressive problem. Its objective is to maximize the conditional log-likelihood of the predicted image code sequence I':

$$\max\, \log P(I' \mid T) = \sum_{t=1}^{N+M} \log P\left(I'_t \mid I'_{<t}, T\right), \tag{16}$$

where $P\left(I'_t \,|I'_{<t}, T\right)$ denotes the probability for prediction of I'_t based on the preceding image codes and text tokens. To fulfill this objective, we use the Mean Squared Error (MSE) as the training loss to minimize the discrepancy between the predicted code sequence I' and the original code sequence I:

$$L_{MSE} = \frac{1}{N+M} \sum_{t=1}^{N+M} \|I'_t - I_t\|^2. \tag{17}$$

Once the training operation of the Text Guidance Sub-network is complete, it can be used to align the image code sequence and the text token sequence.

E. Model Inference

During the inference process, only a textual description is received, from which a sequence of text tokens T is extracted. The sequence processed by RoPE [45] is concatenated with an empty image code sequence before being fed into the TIAM. The prediction of I' is performed code by code. At each time, the most likely next code is computed using the TIAM. The Temperature Sampling strategy [48] is used to ensure that the resulting code sequence matches the data distribution and maintains a certain level of diversity based on the existing image codes and text tokens. This operation is iterated until a complete code sequence has been produced. The sequence is sent to the decoder and a new image is generated.

To derive an arbitrary-resolution image, the new image is resized to the desired resolution. This image is also fed into the ARSRM and an image at the desired resolution is predicted. Both images are fused using the element-wise addition operation. In essence, the predicted image compensates for the missing high-frequency details. As a result, a higher-quality image is derived at the desired resolution.

V. Experimental Setup

In this section, we present the data set, evaluation metrics and implementation details involved in our experiments.

A. Data Set

Within the first step of the training process, only the Natural Terrain Scene Data Set (NTSD) [10] was used. Both the NTSD and our NTSD-TD were utilized in the second step of the training process. Two thirds of the images and textual descriptions contained in both data sets were randomly selected from each category as the training data, while the remaining data was employed as the test data. In total, the training set contained 24,437 pairs of textual descriptions and images, and the test set comprised 12,235 pairs. Note that category labels were not utilized. All the training images were resized to a resolution of 256×256 pixels. They were pre-processed using the same procedure as that used for the VQGAN [12].

B. Evaluation Metrics

We used six evaluation metrics, including Fréchet Inception Distance (FID) [49], Inception Score (IS) [50], No-Reference Image Quality Assessment (NIQE) [51], Zero-Shot Classification Overall Accuracy (Zero-Shot Classification OA) [52], Visual-Question-Answering (VQA) Score [53] and Human Preference Score v2 (HPSv2.1) [54]. These metrics were

computed across the test set in the case that 256×256 images were generated, if not specified.

Both FID and IS were used to evaluate the quality of images generated. The Inception-V3 [55] model pre-trained on the NTSD [10] was used for these metrics. NIQE evaluated the naturalness and realism of images by comparing their statistical features with a predefined distribution of natural images. Both Zero-Shot Classification OA and VQA Score were used to assess the semantic consistency between textual descriptions and images. Regarding the Zero-Shot Classification OA metric, we trained ResNet-18 [56] using 12,235 generated images and performed zero-shot classification on 36,672 real images. Considering that the text encoder of CLIP [57] can act as a "Bag of Words" [53], we used a VQA model to produce an alignment score. In addition, HPSv2.1 was used to evaluate text-image alignment by averaging the scores computed between each generated image and the associated textual description, providing a measure of semantic consistency.

C. Implementation Details

During the first training step, we used the Adam optimizer with an initial learning rate of 4.5e-6, where β_1 and β_2 were set to 0.5 and 0.9, respectively. The Terrainbook contained 1,024 codes. We set the weighting coefficient, β, to 0.6. The batch size was set to 4. The training operation was conducted on two GeForce RTX 3090 GPUs for 200 epochs. During the second training step, the settings of the optimizer were kept unchanged. The number of text tokens, M, was set to 256. We set the batch size to 32. The training operation was carried out on a single GeForce RTX 4090 GPU for 150 epochs.

VI. Experimental Results

In this section, we report the results obtained in the quantitative evaluation, qualitative evaluation, performance analysis and ablation study.

A. Quantitative Evaluation

To assess the effectiveness of the proposed method, we used 12 state-of-the-art text-to-image generation networks as baselines, including Lafite [58], GALIP [59], Attn-GAN [25], DF-GAN [42], DM-GAN [37], U-ViT [60], Dimba [61], Pixart-σ [11], VQGAN-CLIP [62], DALL-E [15], Txt2Img-MHN [52] and Infinity [63]. These baselines were designed on top of Diffusion Models (DMs), GAN and Autoregressive (AR) models. For the purpose of fair comparison, all the networks were trained on the NTSD [10] and our NTSD-TD by following the setup and hyperparameters described in the original publications.

As shown in Table XI, our method achieved the best performance in terms of the FID [49], NIQE [51], Zero-Shot Classification OA metrics [52, 58] and HPSv2.1 [54]. It is indicated that our method produced the better semantic consistency and visual quality, compared with its counterparts. With regard to the IS [50] and VQA Score [53] metrics, the proposed method derived the third and fourth best results, respectively. It should be noted that the TG-TSGNet outperformed the other AR-based methods. Regarding the terrain scene generation task, the generated images should not only exhibit high visual realism but also accurately capture terrain characteristics while maintaining semantic consistency. In this context, our method is able to produce visually realistic terrain scene images which present accurate semantic alignment with textual descriptions, offering a competitive solution to terrain scene generation.

TABLE II
COMPARISON BETWEEN OUR METHOD AND 12 BASELINES ON TERRAIN SCENE GENERATION IN TERMS OF FIVE EVALUATION METRICS. FOR EACH METRIC, THE BEST, SECOND BEST AND THIRD BEST RESULTS ARE HIGHLIGHTED IN **RED**, *Cyan* AND BLUE FONTS, RESPECTIVELY. THIS CONTINUES IN TABLE III.

Method	Type	FID ↓	IS ↑	NIQE ↓	Zero-Shot Classification OA ↑	VQA Score ↑	HPSv2.1 ↑
Lafite [58]		*16.5*	**18.89**	5.00	*56.29*	*0.68*	*21.33*
GALIP [59]		41.5	14.70	*4.73*	54.85	0.63	20.26
Attn-GAN [13]	GAN	77.1	9.31	6.16	20.56	0.28	13.54
DF-GAN [14]		75.1	9.89	4.92	30.26	0.29	14.30
DM-GAN [64]		64.1	9.65	5.20	30.17	0.28	14.42
U-ViT [60]		50.5	10.34	5.62	48.51	0.50	19.19
Dimba [61]	DM	38.0	13.72	5.23	48.37	0.53	19.11
Pixart-σ [11]		29.8	13.70	6.98	53.62	0.53	15.87
VQGAN-CLIP [62]		112.6	9.04	7.46	37.55	0.55	20.28
DALL-E (VQGAN) [15]		44.20	10.00	7.21	41.46	0.39	17.31
Txt2Img-MHN (VQVAE) [52]	AR	140.6	1.10	13.85	2.21	0.26	12.10
Txt2Img-MHN (VQGAN) [52]		117.9	4.31	5.83	26.70	0.41	13.65
Infinity [63]		37.2	*15.20*	5.26	51.20	0.59	*23.18*
TG-TSGNet (Ours)		**15.6**	15.13	**4.61**	**60.17**	0.62	**24.96**

In general, GAN-based methods, except Lafite [58] and GALIP [59], were trained by directly optimizing image quality [15] and manifested the better generalization ability in low-resolution scenes. This finding explains the relatively better performance of these methods on the FID and IS metrics, compared to the AR-based methods. In particular, both Lafite [58] and GALIP [59] utilized the pre-trained CLIP [57] model as the constraint during the training process, which benifited them in the VQA Score metric. However, GANs may suffer from training instability and mode collapse, specifically when they are applied to complex structured data. The inferior performance of Attn-GAN [13], DF-GAN [14] and DM-GAN [64] should be attributed to these issues. In contrast, our TG-TSGNet maintained stable training dynamics through a two-stage training process, which improved both the generation of high-quality terrain scene characteristics and the preservation of structural consistency.

Since DM-based methods generated images by stepwise denoising, they normally produced the higher image quality than the three GAN-based and four AR-based baselines. However, DM-based methods often encounter a high computational cost and a large number of parameters. In contrast, the proposed TG-TSGNet utilizes the significantly fewer parameters and has the lower computational complexity. Therefore, it is more efficient and scalable for both high- and low-resolution data sets than its DM-based counterparts.

AR-based methods sequentially generate images, which allows them to capture fine details. However, they usually struggle with modeling long-range dependencies, which impairs the representation of global characteristics of terrain scenes, such as river paths and mountain continuity. This shortage should be responsible for the poor performance of the AR-based baselines. In contrast, our method takes advantage of the long-range dependency modeling ability of Mamba, which improves the representation of complex terrain structures.

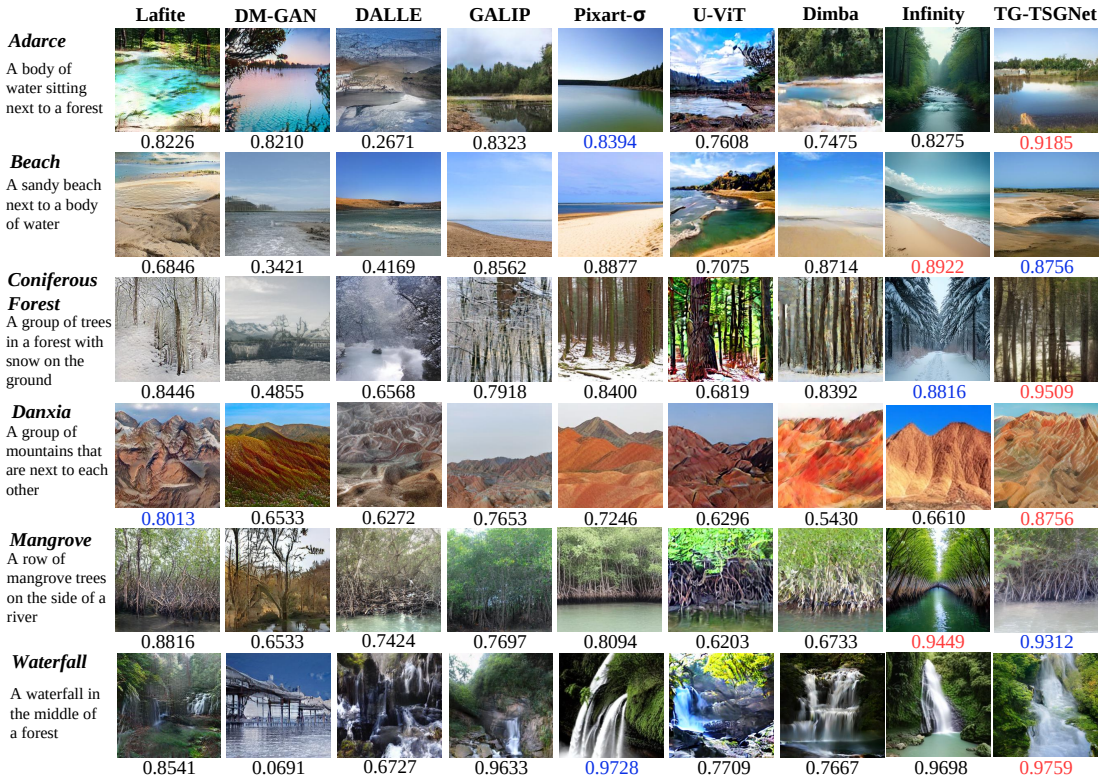

Fig. 6. Comparison of the images generated using seven baselines and our method in terms of a given textual description. The corresponding VQA Score value is shown below each image. Here, the highest and second highest scores are indicated in red and blue, respectively.

B. Qualitative Evaluation

The subjective evaluation of generated images by human observers is also important to topographic image generation tasks, where the details and naturalness of these images are critical for practical applications. Fig. 6 shows the terrain scene images generated using seven baselines and our method with regard to the textual descriptions in nine different categories. It can be observed that our TG-TSGNet demonstrates the superior performance to its counterparts across multiple categories in terms of visual quality and image realism, which is further augmented by the VQA Score metric.

Regarding the textual description in the Adarce category, the image generated using our TG-TSGNet not only preserves complete scene elements but also accurately reconstructs the details of water reflection. This phenomenon can only be observed in the images generated by our model and GALIP [59]. In terms of the textual description in the Coniferous Forest category, the scene involves both snow and forest, demanding a model that can capture the layering of trees in the snow and the variations in light reflections. Our method, together with Pixart-σ [11], were able to generate the snow on the ground better and produced the more natural transition between the snow and trees. However, the rest of the baselines generated the blurring snow area and even lacked snow.

With regard to the textual description in the Mangrove category, it is more difficult to generate realistic images due to the complexity and natural distribution of tree branches. Some baselines were unable to adequately capture the details of intertwined branches, resulting in the chaotic and blurred content that lacked naturalness and hierarchy. In contrast, our method accurately produced the intertwining and interlacing of branches and retained the complex ecological characteristics of mangrove forests. Considering the textural description of the Waterfall category, the proposed method reproduced the transparency and dynamics of the water flow and naturally blended the waterfall with the elements of the surrounding vegetation better than its counterparts.

It is noteworthy that the images that some baselines generated lack element integrity and spatial structure from the perspective of human visual perception, even if they produced high scores in terms of quantitative metrics. In contrast, the images generated using the proposed TG-TSGNet not only show visual realism and semantic consistency with regard to terrain characteristics but also present high quality in line with human visual perception.

C. Performance Analysis

We computed the computational complexity (FLOPs) and the number of parameters for each model using THOP[3]. The average inference time per image was also calculated in terms of the generation of 256×256 images across the test set. To ensure a fair comparison, the inference time was evaluated on the same GeForce RTX 3090 GPU. The results are shown in Table III. It can be seen that the proposed TG-TSGNet has the fewer parameters and lower computational complexity compared to the DM-based and AR-based methods.

[3]https://github.com/Lyken17/pytorch-OpCounter

TABLE III
COMPARISON BETWEEN 12 STATE-OF-THE-ART BASELINES AND OUR METHOD IN TERMS OF THE NUMBER OF PARAMETERS, COMPUTATIONAL COMPLEXITY (FLOPS) AND INFERENCE TIME.

Method	Num. of Param. (B) ↓	FLOPs (G) ↓	Inference Time (S) ↓
Lafite [58]	0.23	170	*0.09*
GALIP [59]	0.32	180	0.10
Attn-GAN [13]	0.23	140	0.93
DF-GAN [14]	**0.019**	**17**	**0.06**
DM-GAN [64]	*0.046*	32	0.55
U-ViT [60]	0.44	133	5.75
Dimba [61]	0.86	210	1.56
Pixart-σ [11]	0.60	168	1.12
VQGAN-CLIP [62]	0.90	195	263.21
DALL-E[15]	12.00	250	1.91
Txt2Img-MHN [52]	0.11	75	1.92
Infinity [63]	0.13	40	0.18
TG-TSGNet (Ours)	0.11	37	1.36

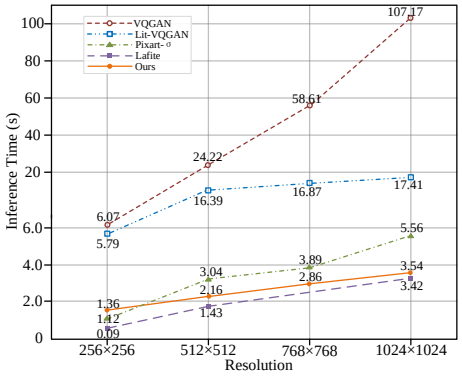

Fig. 7. Comparison of the inference times used by five models for generating images at varying resolutions. The VQGAN [10] and Lit-VQGAN [12] were run for unconditional image generation. Note that Lafite [58] cannot generate images at a resolution of 768×768 pixels due to the inherent limitation.

The inference times of different models are also plotted in terms of four resolutions in Fig. 7. As can be seen, our TG-TSGNet always used the shorter time than that used by the VQGAN [10] and Lit-VQGAN [12]. Compared to Pixart-σ [11], this was also the case when images over 256×256 pixels were generated while our method maintained a flatter growth trend as the resolution increases. In addition, Lafite [58] generated images faster than our method, which should be due to the fact that Lafite [58] directly generated images at a given resolution. However, the gap became smaller and smaller as the resolution increased. It should be noted that our TG-TSGNet can be used to generate arbitrary-resolution images once it has been trained. In contrast, its counterparts have to be retrained for a specific resolution. Thus, our method can perform image generation more flexibly. Referring to the results reported in Table XI, it is demonstrated that our method achieved a proper trade-off between the generation quality and semantic consistency, and the model size, computational complexity and inference time.

D. Ablation Study

To investigate the impact of different modules of our TG-TSGNet on its performance, we conducted a series of ablation experiments, which were performed using the textual prompts contained in the test set of our NTSD-TD, generating a total of 12,235 images.

TABLE IV
COMPARISON BETWEEN OUR CONVMAMBA-VQGAN AND ITS FOUR VARIANTS DERIVED BY REPLACING THE MGRB BY DIFFERENT BLOCKS FOR THE IMAGE RECONSTRUCTION TASK.

Variant	L_{Rec} ↓	L_{Per} ↓
MGRB → Conv Block [12]	0.481	0.449
MGRB → VSSM [38]	0.473	0.339
MGRB → EFFB [10]	0.472	0.342
MGRB → RSSB [39]	0.481	0.342
ConvMamba-VQGAN (Ours)	**0.452**	**0.332**

TABLE V
IMPACT OF THE NUMBER OF MGRBS FOR THE IMAGE SUPER-RESOLUTION TASK.

Number of MGRBs	1	2	3	4
PSNR ↑	27.62	**30.61**	28.51	27.96

1) Impact of the MGRB on Image Reconstruction: To assess the impact of the MGRB on our ConvMamba-VQGAN, we replaced it by different modules, including the convolution block in the VQGAN [10], the VSSM in Vision Mamba [38], the EFFB in the Lit-VQGAN [10] and the RSSB in the MambaIR [39], separately. Correspondingly, we constructed four variants of the ConvMamba-VQGAN. They were applied to the image reconstruction task. We computed the reconstruction loss and perceptual loss between the original and reconstructed images as the evaluation metrics. The results are presented in Table IV. It can be seen that our ConvMamba-VQGAN achieved the superior performance to that derived using the four variants. It is demonstrated that our MGRB is able to produce the better reconstruction quality than its counterparts.

2) Impact of the Encoder on Image Super-Resolution: As shown in Fig. 5, the encoder of the ConvMamba-VQGAN was used as the feature extractor of the ARSRM. Given that the CLRB is used, different MGRBs in the encoder can be used. To examine the impact of the encoder on the image super-resolution task, we used different MGRBs with the ARSRM. The features extracted at the CLRB and different MGRBs were fused in a similar manner to that shown in Fig. 5. Peak Signal-to-Noise Ratio (PSNR) was used as the evaluation metric. As reported in Table V, the best result was produced when two MGRBs, i.e., the first and second MGRBs, were utilized. Furthermore, the PSNR values derived using and without the CLRB were 32.02 and 30.61, respectively. It is suggested that the super-resolution task is particularly sensitive to the shallow features and the utilization of the CLRB is useful for our ARSRM to perform this task.

3) Impact of the ConvMamba-VQGAN on Image Generation: To evaluate the impact of the ConvMamba-VQGAN on the task of image generation, we replaced it with four encoder-decoder networks [10, 12, 24, 27] individually, while keeping the other modules unchanged. We also tested the proposed TG-TSGNet by removing the ARSRM from it. In total, we derive five variants of the TG-TSGNet. The results derived using the variants and our method are compared in Table VI. It can be seen that the use of ConvMamba-VQGAN without the ARSRM produced images with the better quality and semantic consistency, compared to the four encoder-decoder networks.

TABLE VI
COMPARISON BETWEEN OUR TG-TSGNET AND ITS FIVE VARIANTS OBTAINED BY REPLACING THE CONVMAMBA-VQGAN WITH DIFFERENT NETWORKS OR REMOVING THE ARSRM FOR IMAGE GENERATION.

Method	FID ↓	IS ↑	NIQE ↓	Zero-Shot Classification OA ↑	VQA Score ↑	HPSv2.1 ↑
→ DiscreteVAE [27]	109.3	7.35	7.54	25.63	0.38	14.49
→ VQGAN [12]	35.7	11.0	5.28	46.05	0.50	17.58
→ Lit-VQGAN [10]	17.9	14.23	4.74	55.75	0.59	17.82
→ ViT-VQGAN [24]	184.4	3.16	13.5	18.15	0.33	17.54
TG-TSGNet (w/o ARSRM)	17.4	14.68	4.63	55.95	0.59	22.98
TG-TSGNet (Ours)	**15.6**	**15.13**	**4.61**	**60.17**	**0.62**	**24.96**

TABLE VII
COMPARISON BETWEEN THE TG-TSGNET AND ITS FOUR VARIANTS OBTAINED BY REPLACING THE TIAM BY A DIFFERENT MODULE OR REMOVING ROPE [45] FROM THE TIAM FOR IMAGE GENERATION.

Variant	FID ↓	IS ↑	NIQE ↓	Zero-Shot Classification OA ↑	VQA Score ↑	HPSv2.1 ↑
→ Hopfield Layer [52]	27.1	13.94	4.94	50.73	0.53	18.31
→ Transformer [15]	19.5	14.62	4.95	52.87	0.44	17.49
→ Mamba [37]	21.9	14.02	4.79	48.97	0.43	17.98
w/o RoPE	24.4	12.78	4.85	49.22	0.44	21.31
w/ RoPE	**15.6**	**15.13**	**4.61**	**60.17**	**0.62**	**24.96**

TABLE VIII
COMPARISON BETWEEN THE TG-TSGNET AND PIXART-σ [11] AND LAFITE [58] FOR IMAGE GENERATION AT DIFFERENT RESOLUTIONS.

Method	Resolution	FID ↓	IS ↑	NIQE ↓	Zero-Shot Classification OA ↑	VQA Score ↑	HPSv2.1 ↑
Lafite [58]	512 × 512	**18.5**	14.77	5.36	50.78	**0.55**	21.24
Pixart-σ [11]	512 × 512	38.6	10.22	5.49	49.30	0.52	15.61
TG-TSGNet (Ours)	512 × 512	18.8	**14.89**	**5.19**	**55.88**	**0.55**	**24.51**
Lafite [58]	1024 × 1024	19.8	**14.89**	9.64	54.81	0.46	21.28
Pixart-σ [11]	1024 × 1024	52.8	3.57	6.74	46.79	0.46	15.34
TG-TSGNet (Ours)	1024 × 1024	**19.2**	14.34	**5.99**	**55.70**	**0.55**	**22.01**
TG-TSGNet (Ours)	512 × 256	17.8	16.04	5.87	56.84	0.59	23.22
TG-TSGNet (Ours)	512 × 1024	19.3	15.03	5.77	57.30	0.56	23.37

TABLE IX
COMPARISON BETWEEN THE RESULTS OBTAINED USING THE TEXTUAL DESCRIPTIONS DIRECTLY GENERATED USING OFA [36] AND BLIP2 [65] AND MANUALLY OPTIMIZED BY THE AUTHORS.

Textual Descriptions	FID ↓	IS ↑	NIQE ↓	Zero-Shot Classification OA ↑	VQA Score ↑	HPSv2.1 ↑
Original (OFA [36])	21.9	13.08	4.98	38.16	0.40	17.82
Original (Blip2 [65])	21.7	13.12	4.77	42.45	0.42	17.83
Optimized (Ours)	**15.6**	**15.13**	**4.61**	**60.17**	**0.62**	**24.96**

In addition, the incorporation of the ARSRM into the TG-TSGNet further improved the performance.

4) Impact of the TIAM on Image Generation: To assess the effect of our proposed TIAM on text-image alignment, we replaced it with three modules individually, including the Hopfield Layer used by the Txt2Img-MHN [52], the Transformer used by the DALL-E [15] and Mamba [37]. Correspondingly, we constructed three variants of the TG-TSGNet. In addition, we removed RoPE from the TIAM and derived a fourth variant. The four variants were applied to the image generation task. As reported in Table VII, the results show that the TIAM with RoPE manifested the stronger capability of text-image alignment, compared to its counterparts. In particular, the removal of RoPE caused a performance drop, which indicated that the use of proper positional encoding plays an important role in improving the accuracy of text-image alignment.

5) Impact of the Resolution on Image Generation: To evaluate the impact of the resolution on image generation, we used the TG-TSGNet, together with Pixart-σ [11] and Lafite [58], to generate images at different resolutions. The results are reported in Table VIII. It can be seen that the performance of these methods dropped as the resolution increased while our TG-TSGNet normally produced the better results than its counterparts at the same resolution. It is noteworthy that Pixart-σ [11] and Lafite [58] had to be retrained in order to generate 512×512 or 1024×1024 images. In contrast, our TG-TSGNet was only trained once because it could generate images at arbitrary-resolution.

6) Impact of the Optimization of Textual Descriptions on Image Generation: To examine the effect of the optimization process of textual descriptions, we repeated the image generation experiment on three sets of textual descriptions, including the original descriptions generated by OFA [36], the original descriptions generated by BLIP-2 [65], and the manually optimized descriptions that we obtained. As shown in Table IX, we derived much better results using the manually optimized descriptions than those obtained using each set of original descriptions. It is demonstrated that the naturalness of textual descriptions is important to text-to-image generation. In addition, terrain scene images often contain unique or rare landforms that remain challenging even for state-of-the-art vision-language models to describe accurately. In this situation, manual optimization is still necessary for precise semantic alignment and high-quality text-to-image generation.

7) Comparison with Pre-trained Generative Models: We compared our model with four state-of-the-art pre-trained generative models, including Stable Diffusion XL (SD-XL) [67], Stable Diffusion 3 Medium (SD3-Medium) [66], Pixart-σ [11] and Infinity [63], for terrain scene image generation. These models were directly used to generate images without fine-tuning. As shown in Table X, our TG-TSGNet outperformed, or at least performed comparable to, the four pre-trained models, while our method used fewer parameters. It is suggested that the domain-specific adaption to terrain scenes is effective and is even superior to large pre-trained models.

8) Comparison with Fine-Tuned Generative Models: To investigate the effect of the fine-tuning operation of generative models on their performance, we fine-tuned the pre-trained SD3-Medium [66], SD-XL [67], Pixart-σ [11] and Infinity [63] models. As shown in Table XI, the fine-tuned model normally achieved the better performance, compared to the corresponding pre-trained model, in terms of different evaluation metrics. This finding demonstrates that the domain-specific adaption to terrain scenes is useful. In addition, our TG-TSGNet still performed the best with regard to three metrics, including FID [49], NIQE [51] and Zero-Shot Classification OA [52]. It should be noted that our TG-TSGNet only used 0.11B parameters, which were fewer than those utilized by its counterparts. It is suggested that the proposed TG-TSGNet achieves a good trade-off between accuracy and complexity.

VII. CONCLUSION

In this paper, we proposed a Text-Guided Arbitrary-Resolution Terrain Scene Generation Network (TG-TSGNet), which comprised a ConvMamba-VQGAN, a Text Guidance Sub-network and an Arbitrary-Resolution Image Super-Resolution Module (ARSRM). Specifically, the ConvMamba-VQGAN was built using a Conv-Based Local Representation

TABLE X
COMPARISON BETWEEN OUR TG-TSGNET AND FOUR PRE-TRAINED GENERATIVE MODELS ON TERRAIN SCENE IMAGE GENERATION.

Method	Num. of Param. (B) ↓	FID ↓	IS ↑	NIQE ↓	Zero-Shot Classification OA ↑	VQA Score ↑	HPSv2.1 ↑
SD3-Medium [66]	2.00	52.5	10.65	7.36	40.76	**0.62**	24.27
SD-XL [67]	2.60	35.5	**18.57**	5.22	48.96	0.60	**25.81**
Pixart-σ [11]	0.60	27.8	12.55	5.08	44.59	0.61	23.25
Infinity [63]	0.13	64.9	15.21	5.34	35.56	0.58	23.75
TG-TSGNet (Ours)	**0.11**	**15.6**	15.13	**4.61**	**60.17**	**0.62**	24.96

TABLE XI
COMPARISON BETWEEN OUR TG-TSGNET AND FINE-TUNED GENERATIVE MODELS ON TERRAIN SCENE IMAGE GENERATION.

Method	Num. of Param. (B) ↓	FID ↓	IS ↑	NIQE ↓	Zero-Shot Classification OA ↑	VQA Score ↑	HPSv2.1 ↑
SD3-Medium [66]	2.00	39.6	13.22	6.45	47.77	0.69	25.31
SD-XL [67]	2.60	22.1	**20.58**	5.47	49.36	0.69	24.35
Pixart-σ [11]	0.60	26.0	16.23	5.87	54.11	**0.71**	24.16
Infinity [63]	0.13	61.3	15.21	5.22	41.03	0.61	**25.37**
TG-TSGNet (Ours)	**0.11**	**15.6**	15.13	**4.61**	**60.17**	0.62	24.96

Block (CLRB) and a Mamba-Based Global Representation Block (MGRB). Thus, this network was able to learn both local features and long-range dependencies. The Text Guidance Sub-network aimed to inject textual semantics into image representation. To this end, we designed a Text-Image Alignment Module (TIAM). For the purpose of generating images at various resolutions, we adopted the ARSRM. Furthermore, we derived a set of textual descriptions for the 36,672 images in the Natural Terrain Scene Data Set (NTSD) to facilitate the text-to-image generation task. Experimental results showed that our TG-TSGNet generated the better, or at least competitive, images in terms of image realism and semantic consistency at a proper computational cost, compared to the baselines. However, our method could generate arbitrary-resolution images. These promising results should result from the capability of the TG-TSGNet to not only encode both the local and global image characteristics and the semantics of terrain scenes but also decrease the computational cost.

REFERENCES

[1] K. Pavelka Jr and M. Landa, "Using virtual and augmented reality with gis data." *ISPRS International Journal of Geo-Information*, vol. 13, no. 7, 2024.

[2] J. Rambach, G. Lilligreen, A. Schäfer, R. Bankanal, A. Wiebel, and D. Stricker, "A survey on applications of augmented, mixed and virtual reality for nature and environment," in *International conference on human-computer interaction*. Springer, 2021, pp. 653–675.

[3] L. Gu, H. Zhang, and X. Wu, "Surveying and mapping of large-scale 3d digital topographic map based on oblique photography technology," *Journal of Radiation Research and Applied Sciences*, vol. 17, no. 1, p. 100772, 2024.

[4] J. Cannon, "The fractal geometry of nature. by benoit b. mandelbrot," *The American Mathematical Monthly*, vol. 91, no. 9, pp. 594–598, 1984.

[5] G. S. P. Miller, "The definition and rendering of terrain maps," *ACM SIGGRAPH Computer Graphics*, vol. 20, no. 4, p. 39–48, Aug 1986.

[6] C. Dachsbacher, *Interactive terrain rendering: towards realism with procedural models and graphics hardware*. Friedrich-Alexander-Universitaet Erlangen-Nuernberg (Germany), 2006.

[7] A. D. Kelley, M. C. Malin, and G. M. Nielson, "Terrain simulation using a model of stream erosion," *ACM SIGGRAPH Computer Graphics*, vol. 22, no. 4, p. 263–268, Aug 1988.

[8] F. K. Musgrave, C. E. Kolb, and R. S. Mace, "The synthesis and rendering of eroded fractal terrains," in *Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, 1989, p. 41–50.

[9] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[10] Y. Wang, H. Zhou, and X. Dong, "Terrain scene generation using a lightweight vector quantized generative adversarial network," *IEEE Transactions on Big Data*, pp. 1–13, 2025.

[11] J. Chen, C. Ge, E. Xie, Y. Wu, L. Yao, X. Ren, Z. Wang, P. Luo, H. Lu, and Z. Li, "Pixart-σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation," in *European Conference on Computer Vision*. Springer, 2025, pp. 74–91.

[12] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2021.

[13] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "Attngan: Fine-grained text to image generation with attentional generative adversarial networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018.

[14] M. Tao, H. Tang, F. Wu, X.-Y. Jing, B.-K. Bao, and C. Xu, "Df-gan: A simple and effective baseline for text-to-image synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 16 515–16 525.

[15] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," in *International conference on machine learning*. Pmlr, 2021, pp. 8821–8831.

[16] Y. Chen, S. Liu, and X. Wang, "Learning continuous image representation with local implicit image function," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8628–8638.

[17] E. Mansimov, E. Parisotto, J. Ba, and R. Salakhutdinov, "Generating images from captions with attention," in *ICLR*, 2016.

[18] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," *Cornell University - arXiv,Cornell University - arXiv*, May 2016.

[19] W. Liao, K. Hu, M. Y. Yang, and B. Rosenhahn, "Text to image generation with semantic-spatial aware gan," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 18 187–18 196.

[20] M. Kang, J.-Y. Zhu, R. Zhang, J. Park, E. Shechtman, S. Paris, and T. Park, "Scaling up gans for text-to-image synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 10 124–10 134.

[21] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, "Glide: Towards photorealistic image generation and editing with text-guided diffusion models," 2022.

[22] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.

[23] H. Phung, Q. Dao, T. Dao, H. Phan, D. Metaxas, and A. Tran, "Dimsum: Diffusion mamba – a scalable and unified spatial-frequency method for image generation," 2025.

[24] J. Yu, X. Li, J. Koh, H. Zhang, R. Pang, J. Qin, A. Ku, Y. Xu, J. Baldridge, and Y. Wu, "Vector-quantized image modeling with improved vqgan," *arXiv*, Oct 2021.

[25] R. Rombach, P. Esser, and B. Ommer, "Network-to-network translation with conditional invertible neural networks," *Neural Information Processing Systems,Neural Information Processing Systems*, Jan 2020.

[26] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," *Advances in neural information processing systems*, vol. 30, 2017.

[27] A. Razavi, A. Van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with vq-vae-2," *Advances in neural information processing systems*, vol. 32, 2019.

[28] H. Li, J. Yang, K. Wang, X. Qiu, Y. Chou, X. Li, and G. Li, "Scalable autoregressive image generation with mamba," 2025.

[29] J. Lee and K. H. Jin, "Local texture estimator for implicit representation function," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1929–1938.

[30] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 3836–3847.

[31] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1646–1654.

[32] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, and J. Sun, "Meta-sr: A magnification-arbitrary network for super-resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 1575–1584.

[33] L. Wang, Y. Wang, Z. Lin, J. Yang, W. An, and Y. Guo, "Learning a single network for scale-arbitrary super-resolution," in *Proceedings of*

the IEEE/CVF international conference on computer vision, 2021, pp. 4801–4810.
[34] X. Xu, Z. Wang, and H. Shi, "Ultrasr: Spatial encoding is a missing key for implicit image function-based arbitrary-scale super-resolution," *arXiv preprint arXiv:2103.12716*, 2021.
[35] M. Wei and X. Zhang, "Super-resolution neural operator," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 18 247–18 256.
[36] P. Wang, A. Yang, R. Men, J. Lin, S. Bai, Z. Li, J. Ma, C. Zhou, J. Zhou, and H. Yang, "Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework," in *International conference on machine learning*. PMLR, 2022, pp. 23 318–23 340.
[37] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," *arXiv preprint arXiv:2312.00752*, 2023.
[38] Y. Liu, Y. Tian, Y. Zhao, H. Yu, L. Xie, Y. Wang, Q. Ye, J. Jiao, and Y. Liu, "Vmamba: Visual state space model," *Advances in neural information processing systems*, vol. 37, pp. 103 031–103 063, 2024.
[39] H. Guo, J. Li, T. Dai, Z. Ouyang, X. Ren, and S.-T. Xia, "Mambair: A simple baseline for image restoration with state-space model," in *European conference on computer vision*, 2025, pp. 222–241.
[40] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018.
[41] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017.
[42] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Jan 2016.
[43] B. Chen, T. Dao, E. Winsor, Z. Song, A. Rudra, and C. Ré, "Scatterbrain: Unifying sparse and low-rank attention," *Advances in Neural Information Processing Systems*, vol. 34, pp. 17 413–17 426, 2021.
[44] Q. Fan, H. Huang, and R. He, "Breaking the low-rank dilemma of linear attention," *arXiv preprint arXiv:2411.07635*, 2024.
[45] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu, "Roformer: Enhanced transformer with rotary position embedding," *Neurocomputing*, vol. 568, p. 127063, 2024.
[46] A. Ziyaden, A. Yelenov, and A. Pak, "Long-context transformers: A survey," in *2021 5th Scientific School Dynamics of Complex Networks and their Applications (DCNA)*. IEEE, 2021, pp. 215–218.
[47] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[48] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *Advances in neural information processing systems*, vol. 29, 2016.
[49] 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," *Advances in neural information processing systems*, vol. 30, 2017.
[50] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *Advances in neural information processing systems*, vol. 29, 2016.
[51] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, 2012.
[52] Y. Xu, W. Yu, P. Ghamisi, M. Kopp, and S. Hochreiter, "Txt2img-mhn: Remote sensing image generation from text using modern hopfield networks," *IEEE Transactions on Image Processing*, 2023.
[53] Z. Lin, D. Pathak, B. Li, J. Li, X. Xia, G. Neubig, P. Zhang, and D. Ramanan, "Evaluating text-to-visual generation with image-to-text generation," in *European Conference on Computer Vision*, 2024, pp. 366–384.
[54] X. Wu, Y. Hao, K. Sun, Y. Chen, F. Zhu, R. Zhao, and H. Li, "Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis," 2023.
[55] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826.
[56] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
[57] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021.
[58] Y. Zhou, R. Zhang, C. Chen, C. Li, C. Tensmeyer, T. Yu, J. Gu, J. Xu, and T. Sun, "Towards language-free training for text-to-image generation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 17 907–17 917.
[59] M. Tao, B.-K. Bao, H. Tang, and C. Xu, "Galip: Generative adversarial clips for text-to-image synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 14 214–14 223.
[60] F. Bao, S. Nie, K. Xue, Y. Cao, C. Li, H. Su, and J. Zhu, "All are worth words: A vit backbone for diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 22 669–22 679.
[61] Y. Teng, Y. Wu, H. Shi, X. Ning, G. Dai, Y. Wang, Z. Li, and X. Liu, "Dim: Diffusion mamba for efficient high-resolution image synthesis," *arXiv preprint arXiv:2405.14224*, 2024.
[62] K. Crowson, S. Biderman, D. Kornis, D. Stander, E. Hallahan, L. Castricato, and E. Raff, "Vqgan-clip: Open domain image generation and editing with natural language guidance," in *European Conference on Computer Vision*. Springer, 2022, pp. 88–105.
[63] J. Han, J. Liu, Y. Jiang, B. Yan, Y. Zhang, Z. Yuan, B. Peng, and X. Liu, "Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025, pp. 15 733–15 744.
[64] M. Zhu, P. Pan, W. Chen, and Y. Yang, "Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5802–5810.
[65] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," 2023.
[66] P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saini, Y. Levi, D. Lorenz, A. Sauer, F. Boesel, D. Podell, T. Dockhorn, Z. English, K. Lacey, A. Goodwin, Y. Marek, and R. Rombach, "Scaling rectified flow transformers for high-resolution image synthesis," 2024.
[67] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, "Sdxl: Improving latent diffusion models for high-resolution image synthesis," 2023.

Yifan Zhu received the bachelor's degree in Information Science and Computational Mathematics from Qingdao Agricultural University, Shandong Province, China, in 2023. She is currently a post-graduate student at Ocean University of China working toward her master's degree in Computer Science. Her research interests include computer vision, deep learning, image generation, and image super-resolution.

Yan Wang received the bachelor's degree in Computer Science and Technology from Jining Medical College, Shandong Province, China in 2021. She is currently a post-graduate student at Ocean University of China working toward her master's degree in Computer Science. Her research interests include computer vision, deep learning, image generation, and image super-resolution.

Xinghui Dong received the PhD degree from Heriot-Watt University, U.K., in 2014. He worked with the Centre for Imaging Sciences, the University of Manchester, U.K., between 2015 and 2021. Then he joined Ocean University of China in 2021. He is currently a professor at the Ocean University of China. His research interests include computer vision, defect detection, texture analysis, underwater image processing and visual perception.