Diverse Hyperspectral Remote Sensing Image Synthesis with Diffusion Models

Liqin Liu, Bowen Chen, Hao Chen, Zhengxia Zou, and Zhenwei Shi*, *Member, IEEE*

***Abstract*—Hyperspectral image synthesis overcomes the limitations of imaging sensors and enables low-cost acquisition of hyperspectral images with high spatial resolution. Using RGB as a conditional input for hyperspectral generation is promising and valuable, as it can leverage abundant existing multispectral/RGB images without the intervention of hyperspectral sensors. However, most existing generation methods follow one-to-one mapping frameworks and ignore generation diversity. In addition, the current evaluation metrics of hyperspectral generation are based on the similarity with the reference image, which cannot reflect the diversity of the generated spectra. In this paper, we propose a novel method for diverse hyperspectral remote sensing image generation based on the diffusion model. The diffusion model uses a denoising model to gradually remove noise from the normal distribution and generates the hyperspectral data step-by-step with the conditional RGB image as input. To address the high-dimensional noise prediction problem caused by a large number of bands in the hyperspectral image, we introduce a conditional VQGAN that maps the high-dimension hyperspectral data into a low-dimension latent space and conduct the diffusion process in the latent space. The latent-diffusion process makes the diffusion process faster and more stable. The conditional VQGAN decodes hyperspectral images from the latent code generated by diffusion, with the conditional RGB image as input, which restricts the diversity to a specific object distribution. We also design two new metrics to evaluate the generation spectral diversity. Experiments on the IEEE *grss_dfc_2018* dataset demonstrate that our method can synthesize highly diverse hyperspectral data. In addition, the rationality of the proposed metrics is also verified.**

***Index Terms*—Remote sensing, hyperspectral image synthesis, diffusion model, diverse spectral synthesis**

I. Introduction

HYPERSPECTRAL remote sensing images have unique advantages over other remote sensing images in vegetation index mapping [1], mineralogical analysis [2], and water monitoring [3], benefiting from their high spectral resolution. The accuracy and precision of these applications depend, respectively, on the quality and spatial resolution of the hyperspectral images (HSIs). However, obtaining HSIs with high-quality and high spatial resolution is challenging. On the one hand, imaging sensors require high time and technical costs due to the strict requirements for platform stability and weather conditions [4]. On the other hand, the spatial and spectral resolutions of the remote sensing image are inversely related, and HSIs often suffer from low spatial resolution [5]. Hyperspectral image synthesis methods can overcome these challenges by using RGB images or multispectral images (MSIs) as conditional inputs to generate HSIs for specific scenes [6]. This way, hyperspectral image synthesis can achieve high-quality generation of high spatial resolution HSIs without the expensive costs of hyperspectral sensors.

Hyperspectral synthesis methods have received growing interest in recent years and the quality of the synthesized spectra has been continuously improving [7, 8]. However, existing hyperspectral synthesis methods aim to maximize the similarity between the synthesized and the real images in terms of Peak Signal-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM), while ignoring the purpose of improving application accuracy. For related downstream tasks, such as hyperspectral classification [9], target detection [10], and unmixing [11], the synthesized high-spatial-resolution HSI needs to closely resemble the real scene, especially the spectral variability. Spectral variability denotes the variation of spectral features of the same material caused by the effects of atmosphere, illumination, and other environmental factors as well as sensor imaging noise [12]. However, existing methods often follow the paradigm of one-to-one mapping, which cannot support diverse generation, due to the similarity-maximizing objective [13–15]. Therefore, it is important to study diverse hyperspectral synthesis methods. On the one hand, they can better simulate the phenomenon of spectral variation and generate data with a distribution closer to real-world data. On the other hand, they can enrich the hyperspectral data and reduce the risk of over-fitting, which is easily caused by a small number of samples.

Image synthesis methods based on generative models are mainly divided into three categories: Generative Adversarial Networks (GANs) [16–18], likelihood-based methods [19, 20], and diffusion models [21, 22]. GANs have achieved high-quality hyperspectral image synthesis [23, 24]. However, GANs may lack generative diversity. Likelihood-based methods, such as Variational AutoEncoders (VAEs) [19] and Normalizing Flows (NFs) [20], have not been applied to hyperspectral generation due to their inability to synthesize

The work was supported by the National Natural Science Foundation of China under Grant 62125102, the National Key Research and Development Program of China under Grant 2022ZD0160401, the Beijing Natural Science Foundation under Grant JL23005, the Academic Excellence Foundation of BUAA for PhD Students, and the Fundamental Research Funds for the Central Universities. *(Corresponding author: Zhenwei Shi (e-mail: shizhenwei@buaa.edu.cn))*

Liqin Liu, Bowen Chen and Zhenwei Shi are with the Image Processing Center, School of Astronautics, Beihang University, Beijing 100191, China, and with the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China, and also with Shanghai Artificial Intelligence Laboratory, Shanghai 200232, China.

Liqin Liu is also with Shen Yuan Honors College of Beihang University, Beijing 100191, China.

Hao Chen is with Shanghai Artificial Intelligence Laboratory, Shanghai 200232, China.

Zhengxia Zou is with the Department of Guidance, Navigation and Control, School of Astronautics, Beihang University, Beijing 100191, China, and also with Shanghai Artificial Intelligence Laboratory, Shanghai 200232, China.

high-quality images. Diffusion models have recently gained widespread attention, as they destroy data with successive addition of Gaussian noise and recover the data by reversing the noising process [21, 22]. Diffusion models consist of two processes: a forward process that gradually adds noise to the data until it becomes standard normal distribution noise, and an inverse diffusion process that starts from the noise and gradually predicts the added noise to recover the data. Diffusion models show great potential in diverse hyperspectral image synthesis, as they consider both quality and diversity in image generation [25–27]. However, to the best of our knowledge, no such diffusion-based hyperspectral synthesis method has been proposed with RGB image as conditional input.

Diffusion directly in the hyperspectral image data space poses several challenges. First, hyperspectral images have dozens or even hundreds of bands compared with RGB or multi-spectral images, and the dimension of the noise addition and removal in data space is proportional to the bands, which makes the noise prediction problem difficult. Second, the high dimension of the data space requires a lot of computing resources and time costs for the imperceptible and redundant details. Third, the value range of hyperspectral data is wide and most of them are located in a low range, which poses challenges for data normalization and the diffusion noise schedule design.

In this paper, we propose a diverse hyperspectral remote sensing image synthesis method based on diffusion models in latent space, which we call "Hyperspectral Latent space Diffusion Model (HyperLDM)". Given hyperspectral images and their corresponding RGB images, the method first trains a conditional VQGAN model that encodes the HSIs into a latent space and decodes them from the latent codes. Unlike traditional VQGAN (Vector Quantized Generative Adversarial Networks) [28], we input the corresponding RGB images as conditions along with the latent codes into the decoder, so that the decoder pays more attention to the spatial information in RGB images. Once the training is complete, the parameters of the VQGAN are frozen and a deterministic one-to-one bidirectional mapping between the high-dimensional HSI data and low-dimensional latent code is obtained. Therefore, the hyperspectral data is transferred to a perceptually equivalent and computationally suitable space that ignores some high-frequency details. Next, the method trains the diffusion model in the latent space. It takes the time step, noise image, and conditional image as input and shares the same parameters at each step. The diffusion model predicts the added noise and uses training loss with two terms: the Mean Squared Error (MSE) and structural similarity between the predicted noise and the real one. In the forward generation process, we predict a latent code from a randomly sampled noise through a multi-step denoising iteration of the denoising model with an RGB image as conditional input. Then the decoder of the conditional VQGAN recovers HSI from the denoised latent code with the same conditional RGB image.

Existing evaluation metrics only measure the similarity between synthetic data and the real one. To verify the diversity of the synthesized spectra, we propose two new evaluation metrics: spectral diversity (SD) and spectral diversity-multiple (SDM). Experiments on the IEEE *grss_dfc_2018* [29, 30] show that the HyperLDM method achieves diverse spectral synthesis [1]. The main contributions of the paper are summarized as follows:

1) We are the first to study the spectral variability problem in hyperspectral synthesis and introduce denoising diffusion models for diverse hyperspectral synthesis to the best of our knowledge. The diverse synthesized spectra better simulate the spectral variation in the real world and increase the modes of synthesized spectra, which can reduce the risk of overfitting in downstream processing tasks.
2) To avoid predicting high-dimensional noise in data space diffusion, we design a conditional VQGAN model that maps the hyperspectral data to latent space. With diffusion in the latent space, our denoising model achieves a faster speed and a better denoising quality with fewer model parameters and steps.
3) We propose two new metrics to measure the spectral diversity of the diverse spectral synthesis. The metrics are based on a pre-trained hyperspectral classification model and are more suitable for the purpose of hyperspectral synthesis than similarity metrics such as PSNR and SSIM.

The rest of the paper is organized as follows. In Section II, we review the related work on HSI synthesis methods and generative models. Section III details the HyperLDM method and the design of the newly proposed metrics. Section IV provides experimental evaluations on the synthesis diversity and quality. Finally, we draw conclusions in Section V.

II. Related Work

In this section, we briefly review the hyperspectral image synthesis methods, generative models and their applications in hyperspectral synthesis. Specifically, we focus on diffusion models in the field of image generation.

A. Hyperspectral Image Synthesis

Hyperspectral image synthesis methods aim to acquire high spatial-resolution HSIs [14, 31]. These methods are divided into hyperspectral super-resolution (SR) [32, 33], spectral super-resolution (SSR) [6, 34], and image fusion (IF) [35, 36] according to different conditional input images. Hyperspectral SR takes a low-resolution HSI (LR-HSI) as input, SSR takes a high-resolution multispectral image (HR-MSI) or RGB image as input, and image fusion takes both HR-MSI and LR-HSI as input. Depending on algorithms and techniques, hyperspectral synthesis methods mainly fall into the following categories: methods based on manual features and optimazation [31–33, 37], methods based on dictionary learning [35, 38, 39], methods based on tensor decomposition [40, 41] and methods based on neural networks [42–44]. In recent years, neural network-based hyperspectral synthesis has been developed rapidly, mainly in the following directions: powerful network

[1] The code is publically available at http://levir.buaa.edu.cn/Code.htm.

structure [13, 45, 46], attention to band correlation [47, 48], simulation of imaging process [43, 49] and utilization of imaging prior [42, 44].

Spectral super-resolution (SSR) is a hyperspectral image synthesis method with low imaging cost [6, 35]. We adopt it for diverse hyperspectral synthesis, which takes the RGB image as input. Early SSR methods use optimization or tensor decomposition methods [8, 31]. With the powerful modeling capabilities of neural networks, SSR methods learn spectral characteristics implied in RGB/MSI from a large amount of data [13, 14, 50]. These methods mainly design effective structures for deep feature learning [46, 51, 52]. Due to the limitations of fully data-driven methods, many SSR methods model physical characteristics [15, 45]. HSRNet incorporates the spectral response function (SRF) to group the bands [45]. Arad *et al.* correct the spectra after CNN prediction by combining the unmixing process [53]. SSRNet designs cross fusion network with HSI prior learning modules based on the imaging model [46].

These methods learn a one-to-one mapping between the input conditional image and the target HSI. The evaluation metrics are mainly based on the similarity between the reference image and the synthesized one, such as pixel-wise metrics, including Root Mean Square Errors (RMSE) [7, 14, 34, 42, 50], Mean Relative Absolute Error (MRAE) [8, 50], Peak Signal-Noise Ratio (PSNR) [8, 50], Spectral Angle Mapper (SAM) [8, 14, 50]. Other metrics, such as Structural Similarity Index Measure (SSIM) [8, 46, 50] and relevance metrics [8, 54] are also used for evaluation. Most of the metrics compare the synthesized images with the real ones in data space, which limits their ability to evaluate the synthesis quality without the real reference image and the generation diversity. Although these methods achieve high pixel-wise reconstruction accuracy, they hardly use advanced generative models, resulting in over-smoothed synthesis images.

In this paper, we use an advanced generative model to achieve diverse HSI generation, while taking advantage of imaging process and ground object spectra.

B. Generative Models

A good generative model is expected to have three properties: fast sampling, mode coverage or adequate sample diversity, and high-quality samples. However, existing generative models [55] face a trilemma, which mainly includes three kinds of methods: generative adversarial networks (GANs) [16], likelihood-based methods [20, 56], and diffusion models [21, 22].

GANs consist of a generator that generates images and a discriminator that judges whether the generated images conform to the real image distribution [16]. The generator and the discriminator improve each other with alternating training, thus achieving realistic image generation. GANs are fast at sampling and have achieved high-quality hyperspectral image generation [23, 24, 57, 58]. However, these methods fail to synthesize diverse HSIs and suffer from unstable training processes. Likelihood-based methods, such as Variational AutoEncoders (VAEs) [56, 59] and Normalizing Flows (NFs) [20], learn a mapping from complex data distribution to a latent space. These models capture certain modes with fast sampling but fail to produce high-quality samples, making them unsuitable for high-quality HSI synthesis in high dimensions.

Diffusion models are inspired by nonequilibrium thermodynamics and follow a progressive decompression scheme that is interpreted as autoregressive denoising [21, 22]. Although diffusion models have low sampling speed, they have shown powerful ability in high-quality and diverse synthesis [60]. In this paper, we apply the excellent synthesis properties of diffusion to hyperspectral synthesis, which has not been explored to the best of our knowledge.

C. Diffusion Models

Denoising diffusion probability models (DDPMs) gradually add Gaussian noise to images until they reach a normal distribution in the forward process and denoise them step by step with noise prediction in the reverse process [21, 22]. The noise at the start and each step is randomly sampled from the Gaussian distribution, which results in high-quality samples with a variety of modes [25, 60]. Moreover, score-based generative models (SGMs) rely on a continuous diffusion process that gradually perturbs the data towards a tractable distribution, while the generative model learns to denoise [61, 62].

The multi-step denoising slows down the sampling efficiency, and several methods have been proposed to accelerate diffusion [55, 63–65]. The denoising diffusion implicit model (DDIM) designs a non-markovian diffusion process and optimizes it by the same surrogate objective as DDPM [64]. Variational diffusion models allow a learnable diffusion process that shortens the inverse process [63]. Salimans and Ho design a progressive distillation to gradually reduce the reverse diffusion steps to one step [65]. Moreover, denoising diffusion GAN approximates the reverse process by conditional GANs [55].

DDPM fails to predict high-dimension noise, and thus faces problems with high-resolution image generation. To solve this problem, diffusion models mainly provide three solutions. One is the cascaded generation [66], which first generates low-resolution images and then performs iterative super-resolution on them. The text-to-image models, such as GLIDE [67], DALL·E 2 [27] and Imagen [68], follow the cascaded generation. The other solution is diffusion in the latent space, which reduces the noise dimension and accelerates the diffusion process [62, 69]. The latent diffusion is used by stable diffusion [69], which has been setting off waves in image generation with unimaginable effect [70, 71]. The last one is simple diffusion, which works in an end-to-end manner through careful design of noise schedule and network architecture [72].

Benefits from the strong generative ability, diffusion models have been shining in various fields and applications, such as image editing [25, 73], super-resolution [26], semantic segmentation [74], video synthesis [75], medical image inverse problem solving [76] and 3D shape generation [77].

However, to the best of our knowledge, they have not been applied to hyperspectral synthesis even though they have great

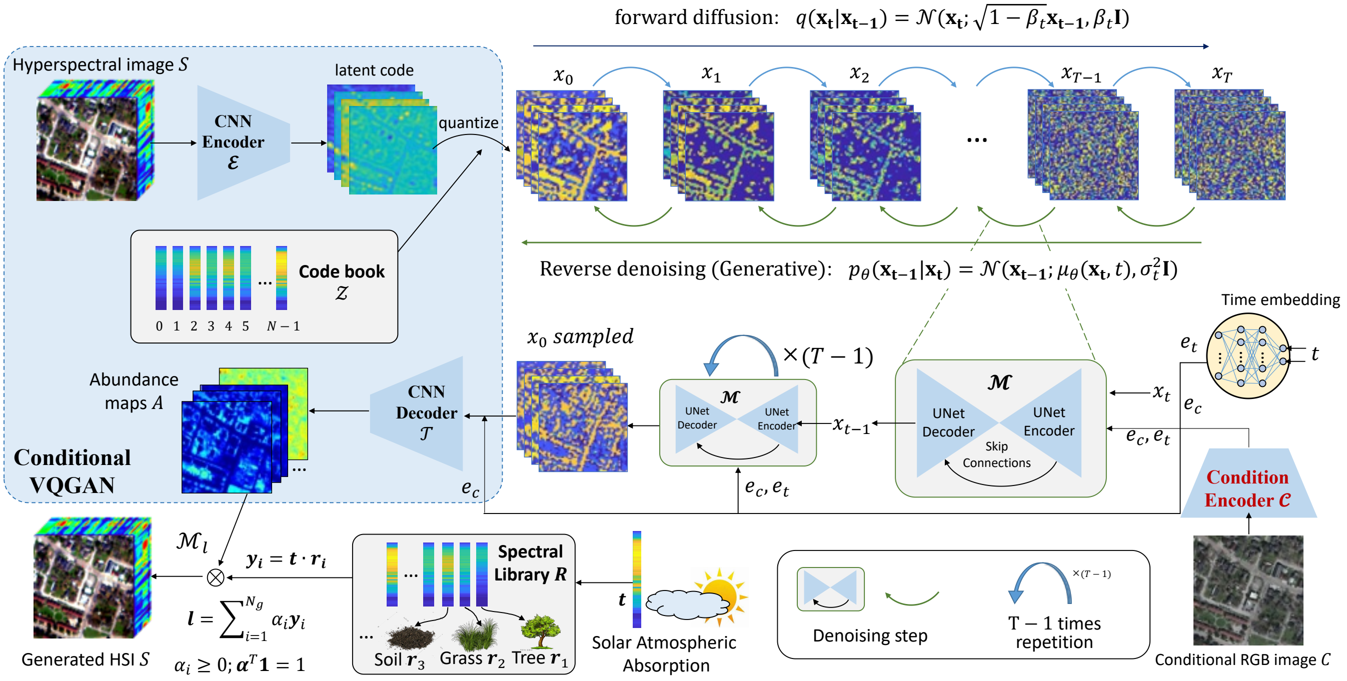

forward diffusion: $q(\mathbf{x_t}|\mathbf{x_{t-1}}) = \mathcal{N}(\mathbf{x_t}; \sqrt{1-\beta_t}\mathbf{x_{t-1}}, \beta_t\mathbf{I})$
Hyperspectral image S
CNN Encoder $\mathcal{E}$
latent code
quantize
x_0
x_1
x_2
x_{T-1}
x_T
Code book $\mathcal{Z}$
0 1 2 3 4 5 $N-1$
Reverse denoising (Generative): $p_\theta(\mathbf{x_{t-1}}|\mathbf{x_t}) = \mathcal{N}(\mathbf{x_{t-1}}; \mu_\theta(\mathbf{x_t}, t), \sigma_t^2\mathbf{I})$
Time embedding
x_0 sampled
$\times(T-1)$
$\mathcal{M}$
UNet Decoder
UNet Encoder
Skip Connections
x_t
x_{t-1}
e_t
t
e_c
e_c, e_t
Abundance maps A
CNN Decoder $\mathcal{T}$
Conditional VQGAN
Condition Encoder $\mathcal{C}$
$\mathcal{M}_l$
$y_i = t \cdot r_i$
$l = \sum_{i=1}^{N_g} \alpha_i y_i$
$\alpha_i \geq 0; \alpha^T \mathbf{1} = 1$
Spectral Library R
Soil r_3 Grass r_2 Tree r_1
Solar Atmospheric Absorption
Denoising step
T − 1 times repetition
Generated HSI S
Conditional RGB image C

Fig. 1. An overview of the proposed method. The conditional VQGAN maps the hyperspectral data to a latent space where the diffusion process operates. The conditional VQGAN is universal and only trained once before training the diffusion model. Both the diffusion model and the VQGAN decoder use the conditional RGB image as input. The generation process starts with noise and iteratively denoises it to obtain the latent code. The decoder $\mathcal{T}$ then takes the latent code as input and produces the abundance map A which is mapped to HSI through the Linear mixing model ($\mathcal{M}_l$).

potential in diverse and high-quality image synthesis. In this paper, we study the diffusion models for the hyperspectral synthesis and make improvements aiming at the high dimension of HSI.

III. Proposed Method

To avoid the high-dimensional noise prediction problem, we design a bidirectional mapping between the high-dimensional hyperspectral data and the low-dimensional latent space, and perform the diffusion process in the latent space. The Hyperspectral Latent space Diffusion Model (HyperLDM) has three advantages over diffusion in the data space: first, it greatly reduces the dimension of noise prediction; second, it speeds up the sampling by reducing the denoising model capacity; and third, it diffuses in the latent space, which is regular in numerical values and easy to denoise.

We design a conditional VQGAN to perform bidirectional mapping, and its decoder takes both the RGB image and the latent code as input. With conditional VQGAN, we specialize HyperLDM for hyperspectral image synthesis conditioned on RGB images. In this section, we start with the design of the conditional VQGAN and then introduce the denoising diffusion in the latent space. After that, we detail some implementation designs. Finally, the two newly proposed metrics that evaluate spectral diversity are introduced.

A. Design of the Conditional VQGAN

The conditional VQGAN is based on the VQGAN model, which has been widely used in natural image generation [28]. Here, we additionally introduce the conditional RGB image as input, thus enabling the VQGAN to perform conditional generation. Meanwhile, we adopt the hyperspectral imaging model proposed in our previous work [57], which obtains the corresponding abundance map while generating the hyperspectral image.

Specifically, given a hyperspectral image $I \in \mathbb{R}^{H \times W \times n_c}$ in hyperspectral data space, the encoder $\mathcal{E}$ encodes it into the latent space $\tilde{x} = \mathcal{E}(I) \in \mathbb{R}^{h \times w \times n_x}$. The encoder downsamples the image by a factor $f = H/h = W/w$, where $f = 2^m, m \in \mathbb{N}$ and the channel of the latent code n_x is usually smaller than the channels of the HSI n_c. To make the latent space more regular and robust, we quantize it with codes from a learnable, discrete codebook $\mathcal{Z} = \{z_k\}_{k=1}^K \subset \mathbb{R}^{n_x}$, which contains K codes with dimension n_x. The quantization assigns an element-wise closest code in $\mathcal{Z}$ for each pixel $\tilde{x}_{ij}$ at location (i, j) in code $\tilde{x}$. Here, the code z_k closest to $\tilde{x}_{ij}$ is chosen from $\mathcal{Z}$ according to L1 distance as:

$$x_q = \mathbf{q}(\tilde{x}) := \left(\underset{z_k \in \mathcal{Z}}{\arg\min} \|\tilde{x}_{ij} - z_k\|\right) \in \mathbb{R}^{h \times w \times n_x}. \quad (1)$$

The decoding process takes both the quantized code x_q and the conditional RGB image $C \in \mathbb{R}^{H \times W \times 3}$ as input. Since the conditional image has a different size from the latent code x_q, we design a condition encoder $\mathcal{C}$, and input the feature $\mathcal{C}(C)$ along with x_q into the decoder $\mathcal{T}$. The abundance map $A \in \mathbb{R}^{H \times W \times n_g}$ and the solar atmospheric spectrum $t \in \mathbb{R}^{n_c}$ are

$$A, t = \mathcal{T}(x_q, C) = \mathcal{T}(\mathbf{q}(\mathcal{E}(I)), \mathcal{C}(C)). \quad (2)$$

With the Linear Mixing Model (LMM) $\mathcal{M}_l$ proposed in [57] and the spectral library $R \in \mathbb{R}^{n_c \times n_g} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{n_g}]$, which has n_g spectra and each with n_c bands, the reconstructed hyperspectral image $\tilde{I} \approx I$ is

$$\tilde{I} = \mathcal{M}_l(A, t, R). \tag{3}$$

The linear mixing model $\mathcal{M}_l$ consists of simple tensor multiplication with no trainable parameters. Therefore, the trainable parameters of the VQGAN are $\mathcal{Z}, \theta_{\mathcal{E}}, \theta_{\mathcal{T}}$, and $\theta_{\mathcal{C}}$, where $\theta_{\mathcal{E}}, \theta_{\mathcal{T}}, \theta_{\mathcal{C}}$ denote parameters in $\mathcal{E}, \mathcal{T}, \mathcal{C}$ respectively.

Since the quantization operation is non-differentiable, we use a straight-through gradient estimator to perform backpropagation, which simply copies the gradients from the decoder $\mathcal{T}$ to the encoder $\mathcal{E}$. Therefore, all the parameters can be trained end-to-end by minimizing the loss function:

$$\begin{aligned}
&\mathcal{L}_{gen}(\theta_{\mathcal{E}}, \theta_{\mathcal{T}}, \theta_{\mathcal{C}}, \mathcal{Z}) = \mathcal{L}_{rec}(I, \tilde{I}) + \mathcal{L}_{qua}(I, x_q) \\
&\mathcal{L}_{rec}(I, \tilde{I}) = \lambda_1 \mathcal{L}_{l1}(I, \tilde{I}) + \lambda_2 \mathcal{L}_{cos}(I, \tilde{I}) \\
&\mathcal{L}_{qua}(I, x_q) = \|sg[\mathcal{E}(I)] - x_q\|_2^2 + \|sg[x_q] - \mathcal{E}(I)\|_2^2.
\end{aligned} \tag{4}$$

Here, $\mathcal{L}_{rec}$ is the reconstruction loss, which consists of pixel similarity and spectral angle similarity loss functions, same as in [57]. According to equations (2) and (3), the reconstruction loss $\mathcal{L}_{rec}$ depends on $\theta_{\mathcal{T}}, \theta_{\mathcal{C}}$ and $\mathcal{Z}$. $\mathcal{L}_{qua}(I, x_q)$ denotes the quantization error, depends on $\theta_{\mathcal{E}}$ and $\mathcal{Z}$. $sg[\cdot]$ denotes the stop-gradient operation.

$$\begin{aligned}
\mathcal{L}_{rec}(I, \tilde{I}) = \mathcal{L}_{rec}(I, \mathcal{M}_l(\mathcal{T}(x_q), R)) = \mathcal{L}_{rec}(\theta_{\mathcal{T}}, \theta_{\mathcal{C}}, \mathcal{Z}) \\
\mathcal{L}_{qua}(I, x_q) = \mathcal{L}_{qua}(\theta_{\mathcal{E}}, \mathcal{Z})
\end{aligned} \tag{5}$$

The quantized latent code x_q is determined by the codebook $\mathcal{Z}$ according to equation (1). During training, the parameters are optimized through gradient back-propagation, and the optimizations to x_q are reflected as updates to $\mathcal{Z}$.

We optimize the above models under the framework of Generative Adversarial Networks (GANs) to make the reconstruction of HSI more realistic. The discriminators include a spatial one $\mathcal{D}_{spat}$ and a spectral one $\mathcal{D}_{spec}$, which are same as [57]. Therefore, the final optimization objective is as follows:

$$\begin{aligned}
\min_{\theta_{\mathcal{E}}, \theta_{\mathcal{T}}, \theta_{\mathcal{C}}, \mathcal{Z}} \max_{\theta_{\mathcal{D}}^{spat}, \theta_{\mathcal{D}}^{spec}} \mathcal{L}_{total} \\
\mathcal{L}_{total} = \mathcal{L}_{gen} + \mathcal{L}_{adv} \\
\mathcal{L}_{adv} = \mathcal{L}_{adv}^{spat} + \mathcal{L}_{adv}^{spec},
\end{aligned} \tag{6}$$

where $\theta_{\mathcal{D}}^{spa}$, $\theta_{\mathcal{D}}^{spe}$ denote the parameters of $\mathcal{D}_{spat}$ and $\mathcal{D}_{spec}$ and $\mathcal{L}_{adv}^{spat}$, $\mathcal{L}_{adv}^{spec}$ denote their losses.

Once the conditional VQGAN is trained, the parameters $\mathcal{Z}, \theta_{\mathcal{E}}, \theta_{\mathcal{T}}$ and $\theta_{\mathcal{C}}$ are frozen and the bidirectional mapping between the hyperspectral data and the low-dimension latent space is established. Therefore, we can synthesize hyperspectral data through latent code generated by diffusion in the latent space.

B. Priliminary of Denoising Diffusion

Diffusion models learn a data distribution $p(x)$ by gradually denoising data from a normal distribution [21]. These models have two processes: a forward process that adds noise to the input data over time and a reverse process that generates data by denoising.

The forward noising process q consists of T steps and adds Gaussian noise with variance $\beta_t \in (0, 1)$ at step $t = 1, 2, \ldots, T$. Given the latent code distribution $p(x_0)$, the latent variables $x_1, x_2, \ldots, x_T$ are as follows:

$$\begin{aligned}
q(x_1, x_2, \ldots, x_T | x_0) &:= \prod_{t=1}^{T} q(x_t | x_{t-1}) \\
q(x_t | x_{t-1}) &:= \mathcal{N}(x_t; \sqrt{1 - \beta_t} x_{t-1}, \beta_t \mathbf{I}).
\end{aligned} \tag{7}$$

$\mathcal{N}(x; \mu, \sigma^2)$ denotes x follows a Gaussian distribution with mean μ and variance σ^2. Then, x_t is the marginalized conditional distribution given by

$$q(x_t | x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha_t}} x_0, (1 - \bar{\alpha_t}) \mathbf{I}) \tag{8}$$

where $\bar{\alpha_t}$ is a predefined constant that $\bar{\alpha_t} = \prod_{s=1}^{t} (1 - \beta_t)$. This implies that we can sample it by the reparameterization technique, without iteratively adding noise.

$$\begin{aligned}
x_t &= \sqrt{\bar{\alpha_t}} x_0 + \sqrt{1 - \bar{\alpha_t}} \epsilon \\
\epsilon &\sim \mathcal{N}(\mathbf{0}, \mathbf{I})
\end{aligned} \tag{9}$$

When the time step T is large enough and the noise schedule of β_t is well-designed such that $\bar{\alpha_T}$ approaches zero, the latent x_T becomes a Gaussian distribution with mean zero, identity covariance matrix, and zero cross-covariance, namely $q(x_T | x_0) \approx \mathcal{N}(\mathbf{0}, \mathbf{I})$.

For the reverse process, if we know the reverse distribution $q(x_{t-1} | x_t)$, we can sample $x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and then run it iteratively to obtain a sample following $p(x_0)$. We approximate $q(x_{t-1} | x_t)$ with a trainable neural network with parameters θ:

$$p_\theta(x_{t-1} | x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{10}$$

The joint distribution is

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1} | x_t). \tag{11}$$

To optimize the neural network, we form the variational upper bound commonly used for training variational autoencoders [19].

$$\mathbb{E}_{q(x_0)} \left[-log p_\theta(x_0) \right] \leq \mathbb{E}_{q(x_0) q(x_{1:T} | x_0)} \left[-log \frac{p_\theta(x_{0:T})}{q(x_{1:T} | x_0)} \right] =: L \tag{12}$$

Following DDPM [21], the Variational Lower Bound (VLB) can be written into three parts:

$$\begin{aligned}
L_{vlb} &:= \mathbb{E}_q \left[L_T + \sum_{t>1} L_{t-1} + L_0 \right] \\
L_T &:= D_{KL} \left(q(x_T | x_0) \| p(x_T) \right) \\
L_{t-1} &:= D_{KL} \left(q(x_{t-1} | x_t, x_0) \| p_\theta(x_{t-1} | x_t) \right) \\
L_0 &:= - \log p_\theta(x_0 | x_1)
\end{aligned} \tag{13}$$

We can see that L_T is irrelevant to θ and L_0 is equal to L_{t-1} when $t = 1$. Therefore, the L_{vlb} is determined by the expectation value of the sum of L_{t-1}. The tractable posterior distribution $q(x_{t-1}|x_t, x_0)$ is:

$$\begin{aligned} q(x_{t-1}|x_t, x_0) &= \mathcal{N}(x_{t-1}; \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t\mathbf{I}) \\ \tilde{\mu}_t(x_t, x_0) &:= \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}x_0 + \frac{\sqrt{1-\beta_t}(1-\bar{\alpha}_t)}{1-\bar{\alpha}_t}x_t \\ \tilde{\beta}_t &:= \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t \end{aligned} \tag{14}$$

Assuming that the Gaussian noise added at each step is small, $q(x_{t-1}|x_t, x_0)$ and $p_\theta(x_{t-1}|x_t)$ can be processed as normal distributions. We set the variance in the reverse process as constant and the trainable parameters only exist at $\mu_\theta(x_t, t)$. The KL (Kullback-Leibler) divergence L_{t-1} has a simple form:

$$\begin{aligned} L_{t-1} &= D_{KL}\left(q(x_{t-1}|x_t, x_0) \| p_\theta(x_{t-1}|x_t)\right) \\ &= \mathbb{E}_q\left[\frac{1}{2\sigma_t^2}\|\tilde{\mu}_t(x_t, x_0) - \mu_\theta(x_t, t)\|^2\right] + C \end{aligned} \tag{15}$$

Therefore, combining equations (9) and (14), the mean of the posterior distribution is

$$\tilde{\mu}_t(x_t, x_0) = \frac{1}{\sqrt{1-\beta_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\alpha_t}}\epsilon\right) \tag{16}$$

The mean of the denoising model can be represented using a noise-prediction network:

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{1-\beta_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\alpha_t}}\epsilon_\theta(x_t, t)\right) \tag{17}$$

Finaly, from equations (15), (16), (17), the VLB is written as

$$\begin{aligned} L_{t-1} &= \mathbb{E}_{x_0 \sim q(x_0), \epsilon \sim \mathcal{N}(0, \mathbf{I})}\left[\lambda_t\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right] \\ x_t &= \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon \\ \lambda_t &= \frac{\beta_t^2}{2\sigma_t^2(1-\beta_t)(1-\bar{\alpha}_t)}, \end{aligned} \tag{18}$$

where λ_t is a time-dependent weight and is often large for small t. DDPM [21] observes that simply setting $\lambda_t = 1$ improves the sample quality, so we optimize the network parameters θ using:

$$\begin{aligned} L_{simple} &= \mathbb{E}_{x_0 \sim q(x_0), \epsilon \sim \mathcal{N}(0, \mathbf{I}), t \sim \mathcal{U}(1, T)}\mathcal{L}_\epsilon \\ \mathcal{L}_\epsilon &= \|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\|^2 \end{aligned} \tag{19}$$

C. Denoising Diffusion in the Latent Space

The hyperspectral data typically follow a long-tail distribution, where most values are less than 0.2 times the maximum value. We perform diffusion in the latent space to cope with the high dimensionality and uneven range of the hyperspectral data.

Specifically, we first encode the hyperspectral image into latent space using the conditional VQGAN. Then we train a diffusion model in this space to generate latent codes with conditional RGB images as input. Finally, we decode the latent codes into hyperspectral images using the conditional VQGAN decoder.

To implement the denoising diffusion, we design a U-Net [78] $\mathcal{M}$ as the noise prediction model. The denoising model $\mathcal{M}$ is iterated for T steps to gradually recover the latent code of the hyperspectral image from a randomly sampled noise from $\mathcal{N}(\mathbf{0}, \mathbf{I})$, while using the conditional RGB image as input. In this section, we describe the design of the noise schedule, the denoising model, the loss functions, and the optimization and sampling processes.

1) Noise schedule: We design the noise schedule to satisfy that $\bar{\alpha}_T \to 0$, as the sampling process starts with a normal distribution of $\mathcal{N}(\mathbf{0}, \mathbf{I})$. The diffusion step is set to T=100 based on empirical experiments. Inspired by [60], we take a linear schedule with minimum 0.01 and maximum 0.2, where $\bar{\alpha}_T = 1.26 \times 10^{-5}$. Hence, $q(x_T|x_0) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and we can start the denoising process with a randomly sampled noise from the distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$.

2) Denoising network: The latent code compresses spatial information by a factor of 4^m ($H \times W$ to $h \times w$), which makes the latent diffusion process less constrained by fine spatial information in conditional RGB images. Therefore, we feed RGB images as condition to the denoising network to ensure its supervision of spatial information. Therefore, we put both the time step t and the conditional RGB image C as input to the noise prediction network. The time embedding module $\mathcal{E}_t$ is composed of two linear layers and outputs the time embedding $e_t = \mathcal{E}_t(t)$, as the same design in [60]. The condition encoder $\mathcal{C}$ encodes C to $e_c = \mathcal{C}(C) \in \mathbb{R}^{h \times w \times n_c}$ standing alone with that in the conditional VQGAN. The model $\mathcal{M}$ takes $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$, e_t and e_c as input. $\mathcal{M}$ is built mainly on Resblocks [79] and AttentionBlocks [80] with 2D convolution. Each Resblock is repeated 2 times for each resolution and has $[128, 128, 256, 256, 512, 512]$ channels for $[2^1, 2^2, 2^3, 2^4, 2^5, 2^6]$ times downsampled features, corresponding. The AttentionBlock is only used for the feature resolution at 8×8 and 4×4. The time-embedding e_t is used at each Resblock to rescale the feature with scale and shift before the skip connection. e_t is obtained by applying $\mathcal{E}_t$ to the time step t, and we project e_t to e_{scale}, e_{shift} as shown in equation (20). The feature f to be processed is then mapped multiplied by e_{scale} and added by e_{shift}.

$$\begin{aligned} e_{scale}, e_{shift} &= Linear(e_t) \\ f_{out} &= (e_{scale} + 1) \times f + e_{shift} \end{aligned} \tag{20}$$

The condition embedding e_c is concatenated with the latent input x_t in the channel dimension. Therefore, the predicted noise at time t is

$$\begin{aligned} \epsilon_\theta &= \mathcal{M}(x_t, e_t, e_c) \\ &= \mathcal{M}(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \mathcal{E}_t(t), \mathcal{C}(C)) \\ &= \mathcal{D}(t, x_t, C), \end{aligned} \tag{21}$$

where $\mathcal{D}$ denotes the denoising diffusion model, including $\mathcal{M}$, $\mathcal{C}$, and $\mathcal{E}_t$.

3) Loss functions: During the training process, the noise prediction network $\mathcal{D}$ predicts the noise ϵ_θ. Given x_0, c and t, the forward diffusion process gets $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$, with the noise ϵ sampled from $\mathcal{N}(\mathbf{0}, \mathbf{I})$. The predicted noise

$\epsilon_\theta(x_0, c, t)$ is called ϵ_θ for short. Accordingly, the $\tilde{x}_0$ predicted by the reverse diffusion model is calculated as follows:

$$\begin{aligned}\tilde{x}_0 &= \frac{1}{\sqrt{\alpha_t}}\left(x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta\right) \\ &= x_0 + \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\alpha_t}}(\epsilon - \epsilon_\theta)\end{aligned} \tag{22}$$

Besides the Mean Square Error (MSE) loss to measure the noise prediction error, we also use the structural similarity loss as in [81] to better capture the spatial information.

$$\begin{aligned}\mathcal{L}_{mse} &= \mathbb{E}_{x_0\sim q(x_0),\epsilon\sim\mathcal{N}(0,\mathbf{I}),t\sim\mathcal{U}(1,T)}\|\epsilon - \epsilon_\theta\|_2^2 \\ \mathcal{L}_{ssim} &= \mathbb{E}_{x_0\sim q(x_0),\epsilon\sim\mathcal{N}(0,\mathbf{I}),t\sim\mathcal{U}(1,T)}\left\{1 - SSIM(\tilde{x}_0, x_0)\right\}\end{aligned} \tag{23}$$

Therefore, the overall objective function is:

$$\begin{aligned}\mathcal{L}_{\mathcal{D}} &= \mathcal{L}_{mse} + \lambda_3\mathcal{L}_{ssim} \\ \theta^*_{\mathcal{D}} &= \arg\min_{\theta_{\mathcal{D}}}\mathcal{L}_{\mathcal{D}}.\end{aligned} \tag{24}$$

4) Processes of optimizing and sampling: To optimize the diffusion model, we take a random sample x_0, t, ϵ and update the noise prediction model $\mathcal{D}$ at each iteration. For the sampling process, we use equation (22) to calculate $\tilde{x}_0$ according to the predicted noise ϵ_θ. The x_{t-1} is then reparameterized according to equation (17). The posterior variance $\tilde{\beta}_t$ is taken as variance without prediction, that is $\sigma_t^2 = \tilde{\beta}_t$.

$$\begin{aligned}q(x_{t-1}|x_t, x_0) &= \mathcal{N}(x_{t-1}; \mu_\theta(x_t, \tilde{x}_0), \sigma^2\mathbf{I}) \\ \mu_\theta(x_t, \tilde{x}_0) &= \frac{1}{\sqrt{1-\beta_t}}(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta) \\ \sigma_t^2 = \tilde{\beta}_t &= \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t\end{aligned} \tag{25}$$

We summarize the optimization and sampling procedures of the latent diffusion model in algorithms 1 and 2, where r is the learning rate.

Algorithm 1 Training of Latent Diffusion Model

```
Input: $q(x_0)$, $C$ corresponding to $x_0$.
Output: The noise-predicting network with parameters $\theta_{\mathcal{D}}$.
repeat
    Sample $x_0 \sim q(x_0), \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), t \sim \mathcal{U}(\{1, \dots, T\})$
    Predict the noise with $\epsilon_\theta = \mathcal{D}(t, x_t, C)$, $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$
    Calculate gradient descent $\nabla_{\theta_{\mathcal{D}}}\mathcal{L}_{\mathcal{D}}$
    Update $\theta_{\mathcal{D}}$ with $\theta_{\mathcal{D}} \leftarrow \theta_{\mathcal{D}} - r\nabla_{\theta_{\mathcal{D}}}\mathcal{L}_{\mathcal{D}}$
until converged
```

Algorithm 2 Sampling of Latent Diffusion Model

```
Input: $C \sim q(C)$.
Output: latent $x_0 \sim q(x_0)$.
Sample $x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
for $t = T, \dots, 1$ do
    Calculate $\epsilon_\theta = \mathcal{D}(t, x_t, C)$
    Sample $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\epsilon = \mathbf{0}$
    $x_{t-1} = \frac{1}{\sqrt{1-\beta_t}}(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta) + \sigma_t\epsilon$
end for
```

D. Implementation Details

1) Parameter Settings: The hyperspectral image $I \in \mathbb{R}^{256\times256\times48}$ is downsampled by a factor of $2^m = 4$ times and the latent channel is $n_x = 16$, i.e. the latent code $\tilde{x} \in \mathbb{R}^{64\times64\times16}$. The codebook contains $K = 1024$ embeddings. For the decoder, we adopt a spectral library $R \in \mathbb{R}^{48\times345}$, which is the same as that in [57]. For the loss functions of the conditional VQGAN, we adopt $\lambda_1 = 100, \lambda_2 = 1000$.

In the latent diffusion model, the time-embedding $e_t \in \mathbb{R}^{1\times512}$ and the condition embedding $e_c \in \mathbb{R}^{64\times64\times32}$. The loss weight $\lambda_3 = 0.1$.

2) Spectral Library: We construct a spectral library based on the AVIRIS 2014 subset of the USGS spectral library V7 [82]. We select spectra of ground objects that might be observed in remote sensing views from the USGS V7 library. Meanwhile, we resample each spectrum from the AVIRIS sensor to the target hyperspectral sensor with linear interpolation. The resampling process is based on the wavelength of each band. We get a spectral library with 345 spectra, and each spectrum has the same number of bands as the target sensor.

3) Training details: During the training of the conditional VQGAN, we optimize the two discriminators once after every 3 iterations of optimization of the other parameters. The whole framework is trained with the Adam optimizer [83] and the cosine learning rate [84] that is initially set to 10^{-5} after 400 epochs with the max-iteration number of 1000. For the latent diffusion, we adopt the AdamW [85] optimizer and the training can converge with an ideal latent code generation after 4000 iterations.

E. The Spectral Diversity Metrics

To evaluate the diversity of the generated HSI, we propose two novel metrics that measure the diversity of the spectra. Unlike the previous metrics that rely on pixel-level comparisons with reference images, our metrics can handle cases where no reference image is available.

We use a pre-trained spectral classification network $\mathcal{S}$ to map the spectra from an HSI $I \in \mathbb{R}^{H\times W\times n_c}$ into semantic space inspired by Inception-Score(IS) [86]. The network consists of four 1D-CNN layers and one fully connection layer, and is trained on the same dataset as the synthesis method. It uses an area with pixel-wise annotation.

The network $\mathcal{S}$ takes a spectrum $s \in \mathbb{R}^{1\times n_c}$ as input, and outputs the softmax vector $v \in \mathbb{R}^{1\times n_s}$.

We introduce the SD metric to measure the spectral diversity within a single HSI, which is useful for methods that do not support multiple HSI generation with diversity under the same condition input. We also introduce the SDM metric to measure the diversity between multiple HSIs, which is useful for methods that support diversity synthesis under the same condition input. The design of the two metrics is as follows.

1) Spectral Diversity of a Single Image: Suppose the generated HSI I_g, the network $\mathcal{S}$ outputs pixel-wise softmax vector $V_g = \{v_g = \mathcal{S}(s)|s \in I_g\} \in \mathbb{R}^{HW\times n_s}$, which contains the semantic information.

We believe that a spectrum with higher quality has a more certain classification result, that is, the entropy of V_g is smaller, which is inspired by image generation [87]. Meanwhile, we hope spectra of the same class have various softmax vectors. Therefore, we divide the softmax vector V_g into n_s groups according to the class predicted by V_g. The marginal distribution of each class $p(y_i)$ is shown in the following equation:

$$\begin{aligned} V_i &= \{v_g | c_g = \underset{c_g=1,2,\ldots,n_s}{\arg\max} \; v_g = i\}, i = 1, 2, \ldots, n_s \\ p(y_i) &= \mathbb{E}_{s\sim p_g}\left[p(y_i|s)\right] \approx \frac{1}{N}\sum_{k=1}^{N} V_i(k) \end{aligned} \tag{26}$$

Each v_g in V_i is expected to be as far away from each other as possible, and we represent this distance by the KL divergence of each predicted conditional probability $p(y_i|s) = \{v_g|c_g = i\}$ and the marginal distribution $p(y_i)$:

$$\begin{aligned} D_{KL}(p(y_i|s) \| p(y_i)) = \mathbb{E}_{s\sim p_g} p(y_i|s) \log\left[p(y_i|s)\right] \\ - p(y_i) \log\left[p(y_i)\right] \end{aligned} \tag{27}$$

The first term in equation (27) represents the negative number of the entropy of V_g, thus maximizing $D_{KL}(p(y_i|s)\|p(y_i))$ will somewhat minimize the entropy of V_g. In other words, a large KL divergence represents a large spectral diversity and also, to some extent, a good spectrum generation quality. Thus the metric SD takes into account generation quality while evaluating the spectral diversity.

We define the spectra diversity with the mean exponential of the aforementioned distance of each class.

$$SD\left(I_g\right) = \frac{1}{n_s}\sum_{i=1}^{n_s} \exp\left(\mathbb{E}_{s\sim p_g} D_{KL}(p(y_i|s)\|p(y_i))\right) \tag{28}$$

2) Spectral Diversity of Multiple HSI Images Under the Same Condition: For methods that can generate multiple HSIs given the same conditional RGB image, we provide a computational metric SDM for the diversity synthesis. Given a conditional RGB image C, the method generates K HSIs $I_{g1}, I_{g2}, \ldots, I_{gK}$, with category softmax vectors $V_{g1}, V_{g2}, \ldots, V_{gK}$ output from $\mathcal{S}$. The conditional distribution is $p(y_i|c) = V_{gi}, i = 1, 2, \ldots K$. The spectral diversity is calculated by the following equation:

$$\begin{aligned} p(y) &= \mathbb{E}_{I_g\sim p_{I_g}} p(y_i|c) \\ SDM &= \exp\left(\mathbb{E}_{I_g\sim p_{I_g}} D_{KL}(p(y_i|c)\|p(y))\right) \end{aligned} \tag{29}$$

IV. Experiment

In this section, we present the datasets and experimental settings to validate the diverse hyperspectral generation, show the generation results of HyperLDM and conduct ablation studies on it. Meawhile, we design downstream experiments to prove that the diverse synthesis method is expected to improve the accuracy of downstream tasks without hyperspectral sensors, and achieve comparable results of real HSI.

A. Datasets and Experimental Setup

We evaluate our method on the IEEE *grss_dfc_2018* dataset, which was collected by the National Center for Airborne Laser Mapping (NCALM) from Houston University [29, 30]. The dataset contains an image scene with spatial size 4172×1202 and 48 bands, covering a wavelength range of 380-1050 nm. The bands 23,12,5 from the data are chosen as the conditional RGB input. The dataset is cropped into 27 paired patches of size 512×512, where 3 non-overlaping patches are used for testing and others for training. For the data processing, we simply normalize the data by dividing it by 4095 and input it to the conditional VQGAN.

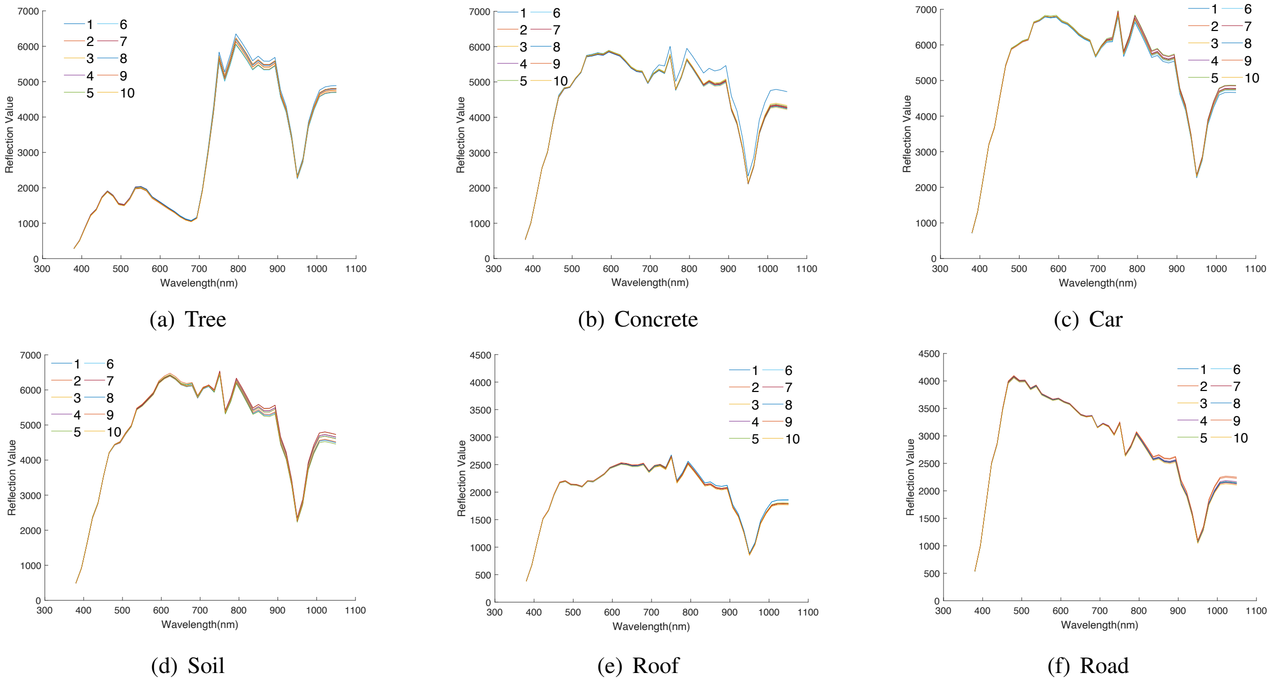
The proposed HyperLDM method is compared with five state-of-the-art methods, including MSCNN [14], HSCNN+ [88], FMNet [13], HASIC-Net [51] and HSR-Net [45]. The methods all follow a data reconstruction framework with residual blocks. HSCNN+ [88] uses multiple residual blocks for feature mapping. MSCNN [14] designs a multiscale deep convolution network. HASIC-Net [51] proposes structure information consistency with attention layers. FMNet [45] designs an adaptive receptive field and HSR-Net [13] groups the bands according to the spectral response function (SRF). For a fair competition, all the experiments are conducted on a desktop PC with the Intel (R) Core (TM) i7-7700K CPU @4.2GHz and an NVIDIA GeForce RTX 3090 GPU card, optimized adequately and the best parameters are selected. The training process of HyperLDM consists of two stages: the training of conditional VQGAN and the diffusion in latent space. The training time of conditional VQGAN costs about 4 hours and that of the diffusion in latent space costs 1.5 hours.

B. Diversity Generation of Hyperspectral Images

TABLE I
Similarity metrics to the reference HSI of 10 runs given the same conditional RGB image.

Time	RMSE ↓	MRAE ↓	SAM ↓	MSSIM ↑	MPSNR ↑
1	388.74	0.0736	0.0585	0.9854	46.996
2	388.97	0.0736	0.0585	0.9854	46.990
3	386.43	0.0731	0.0581	0.9855	47.028
4	391.49	0.0741	0.0587	0.9854	46.959
5	388.25	0.0736	0.0585	0.9855	47.002
6	387.45	0.0735	0.0584	0.9855	47.010
7	389.34	0.0739	0.0586	0.9854	46.986
8	391.66	0.0741	0.0589	0.9853	46.959
9	388.42	0.0735	0.0584	0.9855	46.998
10	387.56	0.0735	0.0584	0.9855	47.012

Given a conditional RGB image, we synthesize diverse latent codes in the latent space using the diffusion model, and then decode them to various hyperspectral data by the decoder $\mathcal{T}$. The synthesized data share similarities with each other and with the reference HSI, but also exhibit pixel-wise differences. Table I shows the similarity metrics between the reference real one and the 10 synthesized images under the same RGB image. We use pixel-wise metrics RMSE, MRAE, SAM and MPSNR, where SAM measures the spectral curve shape and the quality of spectral information. We also use the structure similarity index metric MSSIM for evaluating the quality of spatial information. The results indicate that the diversity is within a reasonable range. Fig. 2 shows the spectral curves at the same pixel location generated by 10 runs. We can see that the curves have small fluctuations and are very close to each other.



(a) Tree (b) Concrete (c) Car

(d) Soil (e) Roof (f) Road

Fig. 2. Spectral diversity: spectra at the same location generated by HyperLDM 10 times, spectra of six different objects are shown in (a)-(f).

C. Comparison with Other Methods

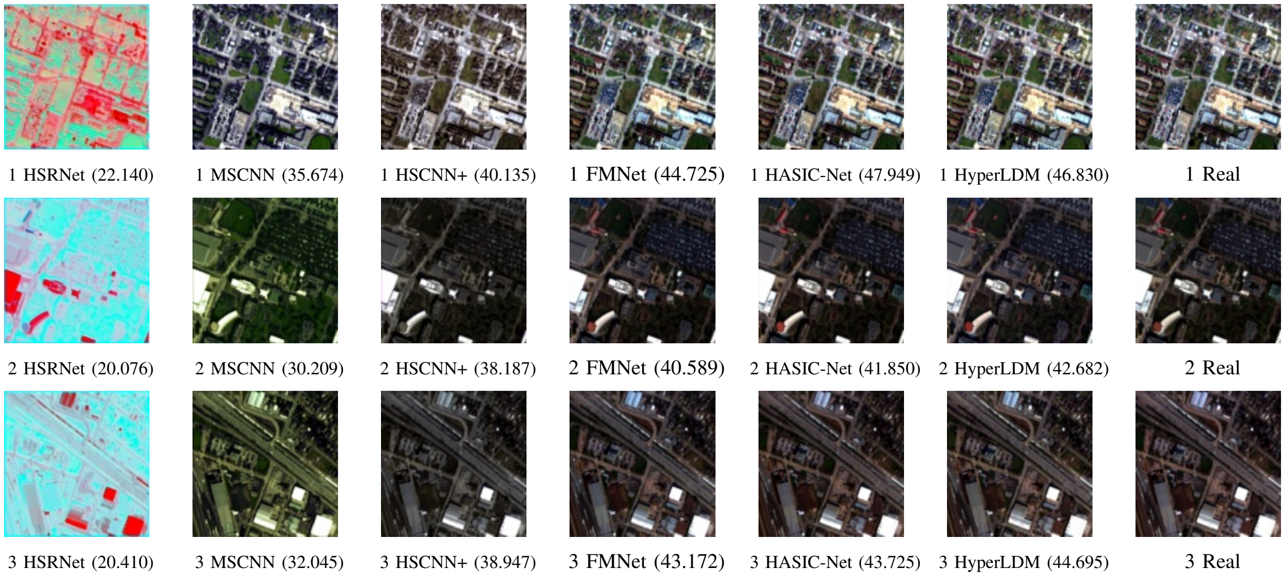
The false-color images (bands 23, 12, and 5) of different methods are shown in Fig. 3, along with their MPSNR values. We can observe that HSRNet [45] fails to reconstruct the color information of RGB bands accurately, and preserve only a small amount of spatial structure. MSCNN [14] recovers more spatial information than HSRNet [45], but introduces color bias in false-color images especially for those with highlighted buildings. By incorporating residual structure and deepening the network, HSCNN+ [88] further alleviates the loss of spatial and color information in false-color images. It can be seen that except for the color deviation of test image # 1 and the red building in the bottom left corner of test image # 2, the rest of the color and spatial information are restored. FMNet [45], HASIC-Net [51] and HyperLDM (Ours) produce false-color images that are visually similar to the real ones, and HyperLDM (Ours) has highest MPSNR values for two of the three images.

TABLE II
ACCURACY OF DIFFERENT METHODS ON THE DATASET. FOR RMSE, MRAE, AND SAM, A LOWER SCORE INDICATES BETTER, WHILE FOR MSSIM, MPSNR, AND SD, A HIGHER SCORE IS BETTER.

Method	RMSE ↓	MRAE ↓	SAM ↓	MSSIM ↑	MPSNR ↑	SD ↑
HSRNet	33380.94	9.0159	1.0416	0.5581	20.875	1.0000
MSCNN	2117.14	0.2859	0.1696	0.9555	37.945	2.4447
HSCNN+	1015.51	0.1667	0.1559	0.9439	39.089	2.4583
FMNet	697.94	0.1177	0.0875	0.9729	42.829	2.4890
HASIC-Net	887.06	0.1116	0.1812	0.9642	44.508	1.5297
HyperLDM	**515.99**	**0.1037**	**0.0724**	**0.9797**	**44.736**	**2.5014**

We present the similarity metrics between different methods and the reference image in Table II, where the metrics are the same as those in Section IV-B. We also report the spectral diversity (SD) of each method, which quantifies the spectral variation within a single image. A higher SD value indicates better diversity. HSRNet [13] losts a lot of spectral and spatial information, resulting in low similarity metrics. The SD value is small because the spectral curves generated by HSRNet [13] are severely distorted and can hardly be correctly classified by the pre-trained classification network $\mathcal{S}$. MSCNN [14] greatly improves the realism of spectral information synthesis and outperforms HSRNet [13] in various metrics. At the same time, due to the more accurate spectral curves, the spectral diversity value is increased by 144% (1.00 to 2.44). HSCNN+ [88] reduces RMSE by half compared with MSCNN [14] (2117.14 to 1015.51) and improves MRAE by 0.12, with the help of deep network and residual blocks. Meanwhile, HSCNN+ [88] performs similarly to MSCNN [14] on other metrics. FMNet [45] improves the shape of the spectral curves by adaptive receptive field, reduces SAM by nearly half of HSCNN+ (0.1559 to 0.0875), and improves the spatial information of the generated HSI, which leads to a large improvement in the structure similarity (from 0.9555 to 0.9729). HASIC-Net [51] improves MPSNR from 42.829 to 44.508, but decreases in other similarity metrics. Meanwhile, the spectral diversity SD is greatly decreased. HyperLDM (Ours) further improves the similarity metrics while enhancing synthesis diversity. Most importantly, HyperLDM (Ours) can generate multiple HSIs from the same conditional RGB image input, as shown in section IV-B. However, the other methods can only give a fixed output HSI without variation.

Fig 4 shows the spectral curves of different methods. HSRNet [45] fails to generate the correct spectral curve, and many of its bands have reflectance values outside the range of spectral reflectance, as illustrated by Fig 4(a). For

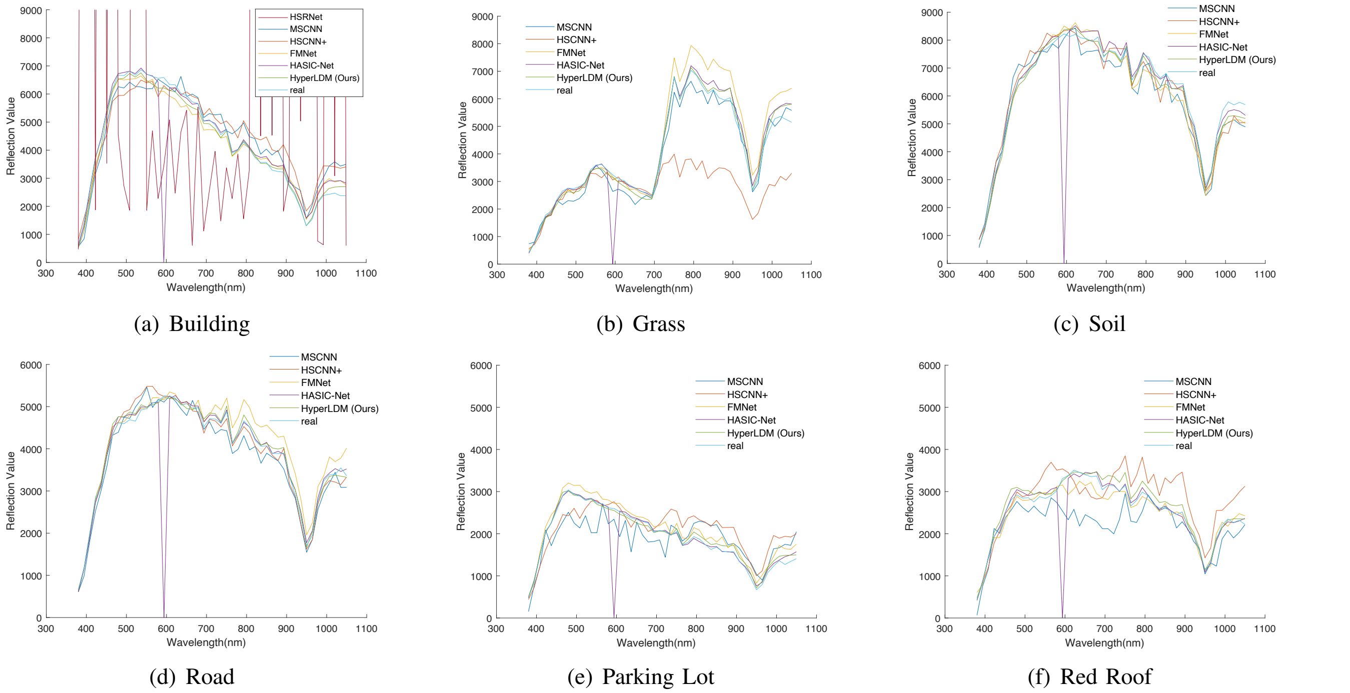


1 HSRNet (22.140) 1 MSCNN (35.674) 1 HSCNN+ (40.135) 1 FMNet (44.725) 1 HASIC-Net (47.949) 1 HyperLDM (46.830) 1 Real

2 HSRNet (20.076) 2 MSCNN (30.209) 2 HSCNN+ (38.187) 2 FMNet (40.589) 2 HASIC-Net (41.850) 2 HyperLDM (42.682) 2 Real

3 HSRNet (20.410) 3 MSCNN (32.045) 3 HSCNN+ (38.947) 3 FMNet (43.172) 3 HASIC-Net (43.725) 3 HyperLDM (44.695) 3 Real

Fig. 3. False-color visualization (band No. 23, 12, and 5) of the synthesis hyperspectral image with different methods: HSRNet [45], MSCNN [14], HSCNN+ [88], FMNet [13], HASIC-Net [51] and HyperLDM (Ours). The reconstruction MPSNR is given along with the image ID. For example, 1 HSRNet (22.140) means the result of HSRNet [23] on test image #1 with MPSNR equals 22.140.



(a) Building (b) Grass (c) Soil

(d) Road (e) Parking Lot (f) Red Roof

Fig. 4. Spectral curves on six objects generated by different methods: HSRNet [45], MSCNN [14], HSCNN+ [88], FMNet [13], HASIC-Net [51] and HyperLDM (Ours).

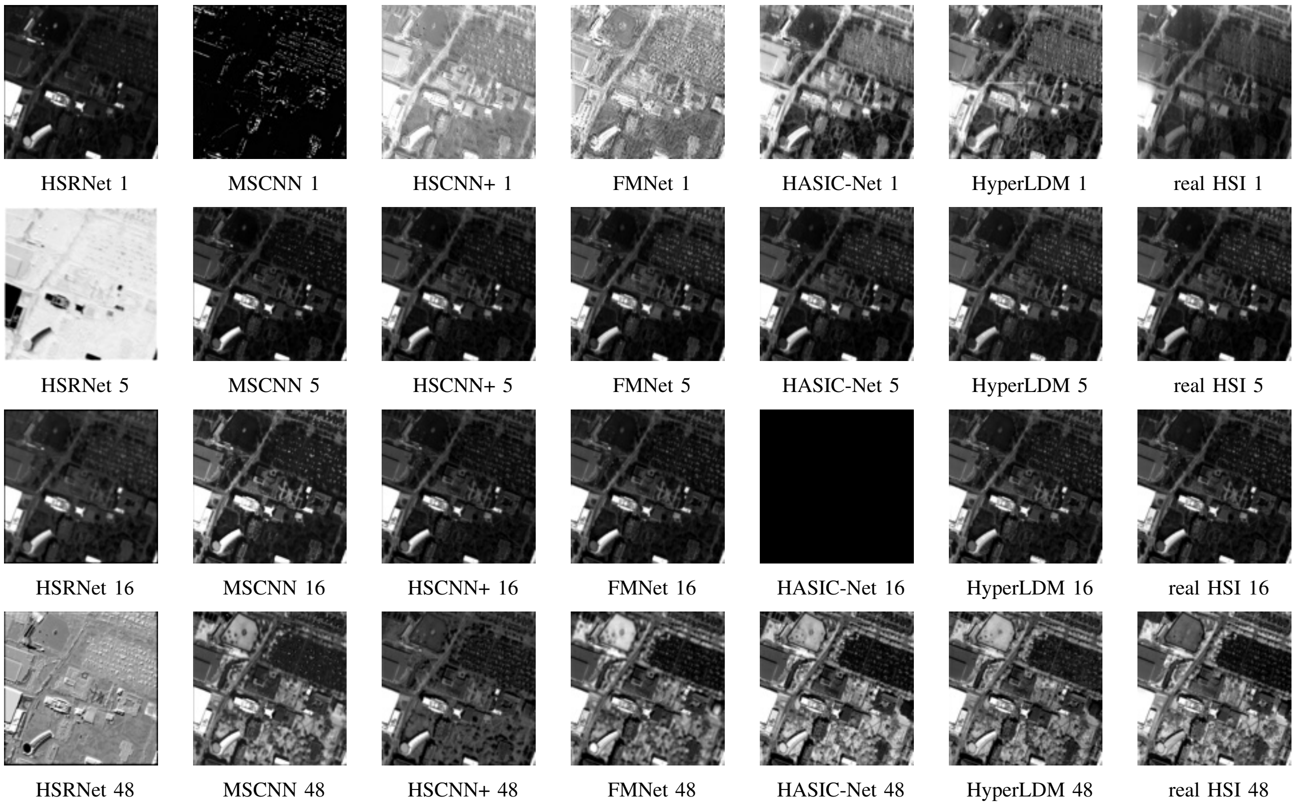


Fig. 5. Band compare of the generated hyperspectral images. Each row contains a particular band generated by different methods: HSRNet [45], MSCNN [14], HSCNN+ [88], FMNet [13], HASIC-Net [51] and HyperLDM (Ours). Each column shows different bands of one method.

a clearer comparison with other methods, the spectral curves generated by HSRNet [45] are omitted in Fig 4(b)-(f). HASIC-Net [51] loses information in a band at wavelength 593.8nm. MSCNN [14] fails to recover spectral curves close to the true ones in the visible wavelength range (400-780 nm), as shown by Fig 4 (e)(f). As a result, severe color distortion on the false-color image of MSCNN, as demonstrated by Fig 3. HSCNN+ [88] has a similar problem in the red and near-infrared bands (700-1100 nm), as indicated by Fig 4 (b)(e) and (f). FMNet [13] and HyperLDM (Ours) generate spectral curves that correctly simulate the shape and absorption peak of the real spectral curve. However, FMNet [13] produces biased spectral curves compared with the true radiation values on two types of ground objects, Grass and road, as revealed by Fig 4 (b)(d). HyperLDM (Ours) reduces these biases and achieves a realistic generation of spectral curves.

The typical bands synthesized by different methods are shown in Fig 5. HSRNet has an abnormal brightness in some bands, namely band 5, that exceeds the range of reasonable reflection values, resulting in the outliers on spectral curves in Fig 4 (a). MSCNN [14], HSCNN+ [88] and FMNet [13] fail to synthesize realistic band 1, since they use a one-to-one fitting framework that cannot fit well in start bands. HSCNN+ [88] also fails to fit the last band (band 48). HASIC-Net [51] fails to synthesize band 16 and results in an abnormal value in the spectral curves as shown in Fig. 4. HyperLDM (Ours) generates each band of the HSI more realistically, and even produces bands with a lower noise level than the real ones, as shown by band 1 in Fig 5.

D. Ablation Studies

To evaluate the design of the conditional VQGAN and the diffusion model, we conduct ablation studies on the following aspects: (1) The dimension n_x and size K of the codebook, the downsampling factor $f = 2^m$ and the conditional input of the decoder for the conditional VQGAN. (2) The steps, noise schedule and the SSIM loss $\mathcal{L}_{ssim}$ for the diffusion process.

1) Design of the Conditional VQGAN: We use the similarity metrics between the real images and the ones reconstructed by conditional VQGAN after the encoding and decoding process to evaluate its design. The results are shown in Table III, where the default parameter settings are the same as in experiment 2.

The dimension of the latent code in the codebook affects the reconstruction accuracy. As experiments 1,2, and 3 in Table III show, a larger n_x preserves more information and leads to a better reconstruction of HSI. However, the reconstruction accuracy does not improve when the dimension of the latent code reaches 32, so we set $n_x = 16$.

In the experiments 4 and 5, we examine the influence of the downsampling factor on the reconstruction accuracy. The reconstruction accuracy decreases as the downsampling factor increases, which implies that a lower downsampling factor retains more spatial information in the latent code and enhances the reconstruction accuracy. Since the reconstruction accuracy does not improve significantly and the diffusion

TABLE III
ABLATION STUDIES OF THE CONDITIONAL VQGAN, † DENOTES THE FINAL SETTING.

Name	n_x	$f = 2^m$	K	condition	λ_1	λ_2	RMSE	MRAE	SAM	MSSIM	MPSNR
1	8						376.30	0.0801	0.0590	0.9846	45.920
2†	16	2^2	1024	✓	100	1000	356.34	**0.0704**	**0.0522**	0.9865	46.656
3	32						362.43	0.0740	0.0550	0.9860	46.369
4		2^1					**349.15**	0.0742	0.0551	**0.9872**	**46.781**
5		2^3					423.35	0.0825	0.0602	0.9824	45.346
6			512				378.72	0.0788	0.0585	0.9849	46.035
7				✗			1231.9	0.2719	0.1118	0.8922	35.653
8					10	100	500.42	0.1117	0.0786	0.9756	43.097
9					100	100	426.69	0.0980	0.0717	0.9808	44.644

computation increases by 4 times when downsampling by $f = 2^1$ instead of $f = 2^2$, we choose $f = 2^2$ as a trade-off between accuracy and efficiency.

In experiment 6 in Table III, we reduce the codebook size to 512 and observe a decrease in accuracy due to the loss of information in the latent code. In the experiment 7, we remove the conditional RGB input of the decoder and decode the hyperspectral image only based on the latent code. This leads to a significant drop in the similarity metrics, because the HSI has a large number of bands that are hard to encode into the latent space only, where both spatial and channel dimensions are greatly compressed. Without the conditional RGB image as auxiliary information, the decoder cannot recover the HSI from the latent code with high MPSNR. In experiments 8 and 9, we change the loss weights λ_1, λ_2 and find the set of $\lambda = 100, \lambda_2 = 1000$ achieves the optimal similarity metrics.

In conclusion, the conditional branch is the only design factor that greatly affects the reconstruction effect, and the VQGAN design is fully robust. Therefore, we adopt the design of experiment 2 in Table III, where the reconstruction accuracy reaches an MPSNR of 46.656, and we assume that this encoding and decoding process can enable the bidirectional conversion between HSI and latent code with minimal information loss during the conversion.

2) Design of the Diffusion Model: For the design of the diffusion process, we conduct ablation study on the following factors: (1) The diffusion steps; (2) The noise schedule; (3) The weight λ_3 of SSIM loss $\mathcal{L}_{ssim}$; (4) The attention layers. The spectral diversity and similarity metrics are shown in Table IV. We use linear noise schedules and vary their max values. For the attention layers, we add them to features of resolution $4, 8$.

We experiment with three different diffusion steps: 50, 100, and 200. We observe that increasing the number of steps from 50 to 100 (experiments 1,3) improves the similarity and the spectral diversity metrics, especially the spectral diversity of multiple HSIs (SDM). Increasing the diffusion step from 100 to 200 (experiments 5,6) does not affect any of the metrics significantly. Therefore, we chose $T = 100$ as the optimal diffusion step.

For the noise schedule, we standardize it for different diffusion steps by setting the maximum noise value to M and the minimum value is $0.005M$, and then compute the noise in the tth step of the diffusion as $0.005M + \frac{M-0.005M}{T-1}t$. We varied the maximum value as 0.01, 0.02 and 0.04 respectively. From experiments 2 and 3 in Table IV, we observe that increasing the maximum noise value from 0.01 to 0.02 substantially improved the diversity of the generated images, especially the SDM which increases from 4.8920 to 5.2420. Moreover, when the noise maximum was increased from 0.02 to 0.04, the diversity metrics were improved (experiments 5,7). Taking into account the trade-off between the diversity and similarity of the generated image and the real image, we selected $M = 0.02$ as the optimal maximum.

As shown in experiments 3 and 4, the introduction of the SSIM loss $\mathcal{L}_{ssim}$ with $\lambda = 0.1$ leads to an improvement in the similarity metrics and a decrease in spectral diversity SDM. Additionally, the addition of the Attention layers has little effect on the similarity with negligible difference in diversity metrics, as shown in experiments 4 and 5. Meanwhile, from experiments 5 and 8, we find little change in the metrics when we increase the weight of $\mathcal{L}_{ssim}$ to $\lambda_3 = 1$. In conclusion, considering the trade-off between diversity and similarity of the generated HSI and the reference ones, we choose to include attention layers and $\mathcal{L}_{ssim}$ with $\lambda_3 = 0.1$.

3) Comparison of diffusion in latent space and data space: For the dataset, the values of the hyperspectral data follow a long-tail distribution, where most of them are smaller than 10000 (97.71%) but the biggest value is 50898. For diffusion in data space, the pre-processing of the data, especially the normalization is challenging since it should satisfy two requirements: 1) The data should follow $N(0, 1)$ after N-step diffusion. Therefore, the variance of the noise schedule at each step should exceed a certain threshold. Moreover, a suitable signal-to-noise ratio (SNR) at each step is essential for successful prediction of high-dimensional noise. Consequently, the normalized data should have a large magnitude. 2) The data should not exceed the reasonable range at each step, and should be constrained within the interval (-1,1). We attempt three normalization methods as follows and find that the clamp method obtains the best results.

1) Linear: Linear normalizing the data to (-1,1) results in the data being overwhelmed by noise, and the SNR is very low, making it difficult to predict the noise.
2) Clamp: We assume that the data follows a Gaussian distribution and calculate the mean μ and standard deviation σ, normalize the data by subtracting μ and dividing by 3σ, and then clamp the data to (-1,1). This normalization maintains 99.20% data but lose information at big values,

TABLE IV
ABLATION STUDIES OF THE DIFFUSION MODEL, † DENOTES THE FINAL SETTING.

Name	step	noise max	λ_3	Attention	RMSE	MRAE	SAM	MSSIM	MPSNR	SD	SDM
1	50	0.02	0		539.82	0.1068	0.0743	0.9792	44.424	2.5114	4.5914
2	100	0.01	0		531.65	0.1061	0.0744	0.9793	44.501	2.4924	4.8920
3	100	0.02	0		524.13	0.1067	0.0752	0.9791	44.617	2.5037	**5.2420**
4	100	0.02	0.1		517.58	**0.1014**	**0.0715**	0.9799	44.660	**2.5155**	4.5026
5†	100	0.02	0.1	✓	**515.99**	0.1037	0.0724	0.9797	**44.736**	2.5014	4.5500
6	200	0.02	0.1	✓	517.33	0.1029	0.0726	0.9796	44.693	2.4975	4.6204
7	100	0.04	0.1	✓	526.18	0.1049	0.0727	0.9796	44.607	2.5122	4.7139
8	100	0.02	1	✓	522.52	0.1021	0.0716	**0.9800**	44.634	2.5018	4.5737

where the diffusion results may suffer big errors.

3) Nonlinear: Using a nonlinear function to map the values to (-1,1) results in low gradients near large values that are difficult to optimize.

TABLE V
TIME COST AND SIMILARITY METRICS OF THE DIFFUSION IN DATA SPACE AND LATENT SPACE

Method	RMSE ↓	MRAE ↓	SAM ↓	MSSIM ↑	MPSNR ↑	Time ↓
Data	944.90	0.2418	0.1744	0.9606	39.491	47.162 s
Latent	**515.99**	**0.1037**	**0.0724**	**0.9797**	**44.736**	**0.7743 s**

The time cost and similarity metrics of the diffusion in data space and latent space are shown in Table V. We can find that diffusion in latent space has better similarity metrics and faster synthesis than that in data space, which improves the synthesis speed by over 60 times.

E. Rationality Analysis of Diversity Metrics

To verify the rationality of the designed diversity evaluation metric, we examine it from the properties of the relationship between the diffusion diversity and the noise variance. According to the sampling process of the diffusion model (step 5 in Algorithm 2), the variance σ_t of the random noise $\sigma_t \boldsymbol{z}$ added at each step follows the posterior variance $\tilde{\beta}_t$, which is strongly correlated with the β_t in the noise schedule, as Eq. (25) shows. A bigger variance $\tilde{\beta}_t$ at each step increases both the uncertainty of the sampled image, and the diversity of multiple synthesized images under the same input conditional image.

We analyze the relationship between the proposed diversity metrics (SD, SDM) and the noise schedule, with the ablation experiments on the noise schedule in Table IV. For experiments 2 and 3, the maximum of the noise schedule is increased twofold, which means the variance of the noise is doubled at each sampling step. The spectral diversity metrics, both SD and SDM, have a numerical increase, which represent the improvement of the synthetic spectrum diversity, consistent with the intuitive diversity change when the noise increases. Similarly, doubling the noise variance of experiment 7 relative to experiment 5 increases the diversity of its synthetic spectrum. The two diversity metrics (SD, SDM) also reflect a similar numerical increase, which indicates their rationality.

F. Improvement on Downstream Task

To evaluate the impact of the hyperspectral image (HSI) synthesis, particularly the diversity synthesis on the downstream tasks, we design an experiment involving a hyperspectral classification task. This experiment allows us to verify the performance promotion of HSI synthesis in the downstream tasks.

We repartition the training and testing sets from the *grss_dfc_2018* dataset for hyperspectral generation according to the pixel-wise annotations area and ensure that the labeled data is all located in the testing set for the generation task. The labeled data consists of 4 596 × 601 patches with 20 classes. 75% of the labeled data is used for classification training and 25% for testing. The classification model is designed with a U-Net [78] consists of 10 residual blocks.

We use three metrics to measure the downstream classification accuracy on different types of data: overall accuracy (OA), average accuracy (AA), and Kappa coefficient (κ) [9, 89]. Table VI shows the results for three types of data: RGB data, synthetic HSI data, and real HSI data. For synthetic HSI data, we vary the synthesis times to generate diverse HSI from 1 time to 10 times for training. For example, 'Synthesis 2×' means that we conduct HyperLDM 2 times to produce twice as many hyperspectral images as the real HSI for training. The testing data is also generated from the testing RGB images.

TABLE VI
ACCURACY OF THE DOWNSTREAM CLASSIFICATION TASK

Name	Training Data	Testing data	OA ↑	AA ↑	κ ↑
1	RGB	RGB	0.7366	0.3529	0.4598
2	Synthesis 1×	Synthesis 1×	0.7868	0.4643	0.5732
3	Synthesis 2×	Synthesis 1×	0.7866	0.4739	0.5724
4	Synthesis 3×	Synthesis 1×	0.8312	0.5130	0.6454
5	Synthesis 5×	Synthesis 1×	0.8378	**0.5244**	**0.6594**
6	Synthesis 10×	Synthesis 1×	**0.8399**	0.4931	0.6578
7	Real HSI	Real HSI	0.8335	0.5074	0.6524

Table VI shows that the pixel-level classification using HSI images generated with RGB images as conditional input achieves higher values of all three metrics (OA, AA, and κ) than using RGB images alone (Experiment 2). Doubling the training data by running HyperLDM twice (Experiment 3), does not affect the classification metrics significantly. However, increasing the number of synthetic HSI training images by three times (Experiment 4) leads to a comparable classification accuracy with real HSI images (Experiment 6), benefiting from the diverse spectral synthesis. Remarkably, running HyperLDM on synthetic data multiple times to further enhance the diversity of training images results in even higher accuracy than that on real HSI data. Enlarging the amount of training data from 5× to 10× does not improve the classification accuracy further (Experiments 5 and 6), suggesting that the spectral diversity reaches the saturation point at 5× synthesis.

In summary, the synthetic hyperspectral images improve the accuracy of the pixel classification compared with using the RGB images. Moreover, generating diverse HSI enhances the classification accuracy even more, matching or surpassing the accuracy of the real HSI data. HyperLDM can potentially lower the reliance of object recognition tasks on costly HSI imaging and attain similar recognition accuracy by using only RGB data and diverse HSI generation.

V. Conclusion

Hyperspectral imaging is expensive and often suffers from low spatial resolution. Hyperspectral synthesis has emerged as an important means to reduce imaging costs and improve spatial resolution, especially using RGB images as input. In this paper, we propose a novel HSI synthesis method based on the diffusion model, which supports diverse hyperspectral image synthesis for the first time, as far as we know. The diffusion model takes both generation quality and diversity into account, and the diverse generation simulates the phenomenon of spectral variation in real imaging. To avoid direct prediction of the noise with the same high dimension as HSI, we design a conditional VQGAN that maps the hyperspectral image into the latent space and it reduces the dimension of noise prediction, speeds up the diffusion inference process, and improves its stability. Furthermore, we propose two new metrics, SD and SDM, to measure the diversity of generated spectra in semantic space, which is inspired by the Inception score (IS). We verify their rationality by analyzing their relationship with noise variance in the diffusion sampling process. Finally, the experimental results show that HyperLDM generates spectra that are both accurate and diverse. The downstream classification experiments demonstrate that diverse HSI synthesis is of great significance to downstream tasks. This is expected to achieve similar ground object observation and recognition results without using expensive HSI imaging. In the future, we aim to accelerate the synthesis process by applying model distillation methods [65].

VI. Acknowledgement

The authors would like to thank the National Center for Airborne Laser Mapping and the Hyperspectral Image Analysis Laboratory, University of Houston, for acquiring and providing the data used in this study and the IEEE GRSS Image Analysis and Data Fusion Technical Committee.

References

[1] G. Yang, K. Huang, W. Sun, X. Meng, D. Mao, and Y. Ge, "Enhanced mangrove vegetation index based on hyperspectral images for mapping mangrove," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 189, pp. 236–254, 2022.

[2] J. M. Meyer, R. F. Kokaly, and E. Holley, "Hyperspectral remote sensing of white mica: A review of imaging and point-based spectrometer studies for mineral resources, with spectrometer design considerations," *Remote Sensing of Environment*, vol. 275, p. 113000, 2022.

[3] J. Cai, J. Chen, X. Dou, and Q. Xing, "Using machine learning algorithms with in situ hyperspectral reflectance data to assess comprehensive water quality of urban rivers," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–13, 2022.

[4] M. B. Stuart, L. R. Stanger, M. J. Hobbs, T. D. Pering, D. Thio, A. J. McGonigle, and J. R. Willmott, "Low-cost hyperspectral imaging system: Design and testing for laboratory-based environmental applications," *Sensors*, vol. 20, no. 11, p. 3293, 2020.

[5] J. Jia, J. Chen, X. Zheng, Y. Wang, S. Guo, H. Sun, C. Jiang, M. Karjalainen, K. Karila, Z. Duan *et al.*, "Tradeoffs in the spatial and spectral resolution of airborne hyperspectral imaging systems: A crop identification case study," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–18, 2021.

[6] K. V. Mishra, M. Cho, A. Kruger, and W. Xu, "Spectral super-resolution with prior knowledge," *IEEE Transactions on Signal Processing*, vol. 63, no. 20, pp. 5342–5357, 2015.

[7] Y. Jia, Y. Zheng, L. Gu, A. Subpa-Asa, A. Lam, Y. Sato, and I. Sato, "From RGB to spectrum for natural scenes via manifold-based mapping," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 4715–4723.

[8] C. Yi, Y.-Q. Zhao, and J. C.-W. Chan, "Spectral super-resolution for multispectral image based on spectral improvement strategy and spatial preservation strategy," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 11, pp. 9010–9024, 2019.

[9] L. Liu, Z. Shi, B. Pan, N. Zhang, H. Luo, and X. Lan, "Multiscale deep spatial feature extraction using virtual rgb image for hyperspectral imagery classification," *Remote sensing*, vol. 12, no. 2, p. 280, 2020.

[10] Z. Zou and Z. Shi, "Hierarchical suppression method for hyperspectral target detection," *IEEE transactions on geoscience and remote sensing*, vol. 54, no. 1, pp. 330–342, 2015.

[11] X. Xu, Z. Shi, and B. Pan, "L0-based sparse hyperspectral unmixing using spectral information and a multi-objectives formulation," *ISPRS journal of photogrammetry and remote sensing*, vol. 141, pp. 46–58, 2018.

[12] A. Zare and K. Ho, "Endmember variability in hyperspectral analysis: Addressing spectral variability during spectral unmixing," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 95–104, 2013.

[13] L. Zhang, Z. Lang, P. Wang, W. Wei, S. Liao, L. Shao, and Y. Zhang, "Pixel-aware deep function-mixture network for spectral super-resolution," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 12 821–12 828.

[14] Y. Yan, L. Zhang, J. Li, W. Wei, and Y. Zhang, "Accurate spectral super-resolution from single rgb image using multi-scale CNN," in *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*. Springer, 2018, pp. 206–217.

[15] R. Hang, Q. Liu, and Z. Li, "Spectral super-resolution network guided by intrinsic properties of hyperspectral imagery," *IEEE Transactions on Image Processing*, vol. 30, pp. 7256–7265, 2021.

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[17] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[18] 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*, 2017, pp. 1125–1134.

[19] D. P. Kingma and M. Welling, "Auto-encoding variational

bayes," in *International Conference on Learning Representations*, 2014.

[20] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *International conference on machine learning*. PMLR, 2015, pp. 1530–1538.

[21] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[22] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8162–8171.

[23] L. Liu, S. Lei, Z. Shi, N. Zhang, and X. Zhu, "Hyperspectral remote sensing imagery generation from RGB images based on joint discrimination," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 7624–7636, 2021.

[24] Y. Shi, L. Han, L. Han, S. Chang, T. Hu, and D. Dancey, "A latent encoder coupled generative adversarial network LE-GAN for efficient hyperspectral image super-resolution," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–19, 2022.

[25] C. Saharia, W. Chan, H. Chang, C. Lee, J. Ho, T. Salimans, D. Fleet, and M. Norouzi, "Palette: Image-to-image diffusion models," in *ACM SIGGRAPH 2022 Conference Proceedings*, 2022, pp. 1–10.

[26] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[27] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, 2022.

[28] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12 873–12 883.

[29] 2018 IEEE GRSS Data Fusion Contest. Online: http://www.grss-ieee.org/community/technical-committees/data-fusion.

[30] Y. Xu, B. Du, L. Zhang, D. Cerra, M. Pato, E. Carmona, S. Prasad, N. Yokoya, R. Hänsch, and B. Le Saux, "Advanced multi-sensor optical remote sensing for urban land use and land cover classification: Outcome of the 2018 IEEE GRSS data fusion contest," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 6, pp. 1709–1724, 2019.

[31] R. Wu, W.-K. Ma, X. Fu, and Q. Li, "Hyperspectral super-resolution via global-local low-rank matrix estimation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 10, pp. 7125–7140, 2020.

[32] T. Akgun, Y. Altunbasak, and R. Mersereau, "Super-resolution reconstruction of hyperspectral images," *IEEE Transactions on Image Processing*, vol. 14, no. 11, pp. 1860–1875, 2005.

[33] H. Irmak, G. B. Akar, and S. E. Yuksel, "A MAP-based approach for hyperspectral imagery super-resolution," *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 2942–2951, 2018.

[34] B. Arad and O. Ben-Shahar, "Sparse recovery of hyperspectral signal from natural RGB images," in *European Conference on Computer Vision*. Springer, 2016, pp. 19–34.

[35] M. A. Veganzones, M. Simões, G. Licciardi, N. Yokoya, J. M. Bioucas-Dias, and J. Chanussot, "Hyperspectral super-resolution of locally low rank images from complementary multisource data," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 274–288, 2016.

[36] X.-H. Han, B. Shi, and Y. Zheng, "Self-similarity constrained sparse representation for hyperspectral image super-resolution," *IEEE Transactions on Image Processing*, vol. 27, no. 11, pp. 5625–5637, 2018.

[37] R. C. Patel and M. V. Joshi, "Super-resolution of hyperspectral images: Use of optimum wavelet filter coefficients and sparsity regularization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 4, pp. 1728–1736, 2015.

[38] H. Kwon and Y.-W. Tai, "RGB-guided hyperspectral image upsampling," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 307–315.

[39] C. Yi, Y.-Q. Zhao, and J. C.-W. Chan, "Hyperspectral image super-resolution based on spatial and spectral correlation fusion," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 7, pp. 4165–4177, 2018.

[40] Y. Xu, Z. Wu, J. Chanussot, and Z. Wei, "Nonlocal patch tensor sparse representation for hyperspectral image super-resolution," *IEEE Transactions on Image Processing*, vol. 28, no. 6, pp. 3034–3047, 2019.

[41] R. Dian, S. Li, and L. Fang, "Learning a low tensor-train rank representation for hyperspectral image super-resolution," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2672–2683, 2019.

[42] S. Nie, L. Gu, Y. Zheng, A. Lam, N. Ono, and I. Sato, "Deeply learned filter response functions for hyperspectral reconstruction," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4767–4776.

[43] L. Zhang, J. Nie, W. Wei, Y. Li, and Y. Zhang, "Deep blind hyperspectral image super-resolution," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 6, pp. 2388–2400, 2021.

[44] Y. Fu, T. Zhang, Y. Zheng, D. Zhang, and H. Huang, "Joint camera spectral response selection and hyperspectral image recovery," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 1, pp. 256–272, 2022.

[45] J. He, J. Li, Q. Yuan, H. Shen, and L. Zhang, "Spectral response function-guided deep optimization-driven network for spectral super-resolution," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[46] R. Dian, T. Shan, W. He, and H. Liu, "Spectral super-resolution via model-guided cross-fusion network," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2023.

[47] X. Wang, J. Ma, and J. Jiang, "Hyperspectral image super-resolution via recurrent feedback embedding and spatial-spectral consistency regularization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–13, 2022.

[48] P. V. Arun, K. M. Buddhiraju, A. Porwal, and J. Chanussot, "CNN-based super-resolution of hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 9, pp. 6106–6121, 2020.

[49] W. Wei, J. Nie, L. Zhang, and Y. Zhang, "Unsupervised recurrent hyperspectral imagery super-resolution using pixel-aware refinement," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[50] J. Li, C. Wu, R. Song, W. Xie, C. Ge, B. Li, and Y. Li, "Hybrid 2-D-3-D deep residual attentional network with structure tensor constraints for spectral super-resolution of RGB images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 3, pp. 2321–2335, 2021.

[51] J. Li, S. Du, R. Song, C. Wu, Y. Li, and Q. Du, "Hasic-net: Hybrid attentional convolutional neural network with structure information consistency for spectral super-resolution of rgb images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[52] J. Li, S. Du, C. Wu, Y. Leng, R. Song, and Y. Li, "Drcr net: Dense residual channel re-calibration network with nonlocal purification for spectral super resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1259–1268.

[53] P. V. Arun, K. M. Buddhiraju, A. Porwal, and J. Chanussot, "Cnn based spectral super-resolution of remote sensing images," *Signal Processing*, vol. 169, p. 107394, 2020.

[54] U. B. Gewali, S. T. Monteiro, and E. Saber, "Spectral super-

resolution with optimized bands," *Remote Sensing*, vol. 11, no. 14, p. 1648, 2019.

[55] Z. Xiao, K. Kreis, and A. Vahdat, "Tackling the generative learning trilemma with denoising diffusion GANs," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=JprM0p-q0Co

[56] 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.

[57] L. Liu, W. Li, Z. Shi, and Z. Zou, "Physics-informed hyperspectral remote sensing image synthesis with deep conditional generative adversarial networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[58] L. Liu, Z. Zou, and Z. Shi, "Hyperspectral remote sensing image synthesis based on implicit neural spectral mixing models," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–14, 2023.

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

[60] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.

[61] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=PxTIG12RRHS

[62] A. Vahdat, K. Kreis, and J. Kautz, "Score-based generative modeling in latent space," *Advances in Neural Information Processing Systems*, vol. 34, pp. 11 287–11 302, 2021.

[63] D. Kingma, T. Salimans, B. Poole, and J. Ho, "Variational diffusion models," *Advances in neural information processing systems*, vol. 34, pp. 21 696–21 707, 2021.

[64] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=St1giarCHLP

[65] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=TIdIXIpzhoI

[66] J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans, "Cascaded diffusion models for high fidelity image generation." *J. Mach. Learn. Res.*, vol. 23, no. 47, pp. 1–33, 2022.

[67] A. Q. 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," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 162. PMLR, 17–23 Jul 2022, pp. 16 784–16 804.

[68] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *Advances in Neural Information Processing Systems*, vol. 35, pp. 36 479–36 494, 2022.

[69] 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.

[70] L. Zhang and M. Agrawala, "Adding conditional control to text-to-image diffusion models," *arXiv preprint arXiv:2302.05543*, 2023.

[71] C. Wu, S. Yin, W. Qi, X. Wang, Z. Tang, and N. Duan, "Visual chatgpt: Talking, drawing and editing with visual foundation models," *arXiv preprint arXiv:2303.04671*, 2023.

[72] E. Hoogeboom, J. Heek, and T. Salimans, "simple diffusion: End-to-end diffusion for high resolution images," *arXiv preprint arXiv:2301.11093*, 2023.

[73] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon, "Sdedit: Guided image synthesis and editing with stochastic differential equations," in *International Conference on Learning Representations*, 2021.

[74] D. Baranchuk, I. Rubachev, A. Voynov, V. Khrulkov, and A. Babenko, "Label-efficient semantic segmentation with diffusion models," *arXiv preprint arXiv:2112.03126*, 2021.

[75] W. Harvey, S. Naderiparizi, V. Masrani, C. D. Weilbach, and F. Wood, "Flexible diffusion modeling of long videos," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022. [Online]. Available: https://openreview.net/forum?id=0RTJcuvHtIu

[76] Y. Song, L. Shen, L. Xing, and S. Ermon, "Solving inverse problems in medical imaging with score-based generative models," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=vaRCHVj0uGI

[77] L. Zhou, Y. Du, and J. Wu, "3d shape generation and completion through point-voxel diffusion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5826–5835.

[78] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[79] 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*, 2016, pp. 770–778.

[80] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[81] L. Liu, Z. Shi, Y. Zao, and H. Chen, "Hyperspectral image generation from rgb images with semantic and spatial distribution consistency," in *IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*, 2022, pp. 1804–1807.

[82] R. Kokaly, R. Clark, G. Swayze, K. Livo, T. Hoefen, N. Pearson, R. Wise, W. Benzel, H. Lowers, R. Driscoll *et al.*, "Usgs spectral library version 7 data: Us geological survey data release," *United States Geological Survey (USGS): Reston, VA, USA*, vol. 61, 2017.

[83] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[84] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *ICLR (Poster)*, 2016.

[85] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiCqY7

[86] M. J. Chong and D. Forsyth, "Effectively unbiased fid and inception score and where to find them," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 6070–6079.

[87] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016.

[88] Z. Shi, C. Chen, Z. Xiong, D. Liu, and F. Wu, "Hscnn+: Advanced cnn-based hyperspectral recovery from rgb images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 939–947.

[89] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6690–6709, 2019.

Liqin Liu received her B.S. degree from Beihang University, Beijing, China in 2018. She is currently working toward her doctorate degree in the Image Processing Center, School of Astronautics, Beihang University. Her research interests include hyperspectral image processing, machine learning and deep learning.

Bowen Chen received his B.S. degree from China University of Petroleum East China, Qingdao, Shandong, China, in 2022. He is currently working toward his master's degree in the Image Processing Center, School of Astronautics, Beihang University.

His research interests include machine learning and pattern recognition.

Hao Chen received his B.S. and Ph.D. degrees from the Image Processing Center, School of Astronautics, Beihang University in 2017 and 2023, respectively. He is currently a Junior Researcher at Shanghai AI Laboratory. His research interests include geospatial machine learning, remote sensing, earth monitoring, and prediction.

Zhengxia Zou received his B.S. degree and his PhD degree from the Image Processing Center, School of Astronautics, Beihang University in 2013 and 2018, respectively. He is currently an Associate Professor at the School of Astronautics, Beihang University. During 2018-2021, he was a postdoc research fellow at the University of Michigan, Ann Arbor. His research interests include computer vision and related problems in remote sensing and autonomous driving. He has published more than 20 peer-reviewed papers in toptier journals and conferences, including TPAMI, TIP, TGRS, CVPR, ICCV, AAAI. His research has been featured in more than 30 global tech media outlets and adopted by multiple application platforms with over 50 million users worldwide. His personal website is https://zhengxiazou.github.io/.

Zhenwei Shi (Member IEEE) received a Ph.D. degree in mathematics from Dalian University of Technology, Dalian, China, in 2005.

He was a Post-Doctoral Researcher with the Department of Automation, Tsinghua University, Beijing, China, from 2005 to 2007. He was Visiting Scholar with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, USA., from 2013 to 2014. He is currently a Professor and Dean of the Image Processing Center, School of Astronautics, Beihang University, Beijing. He has authored or co-authored over 200 scientific articles in refereed journals and proceedings, including the IEEE Transactions on Pattern Analysis and Machine Intelligence, the IEEE Transactions on Image Processing, the IEEE Transactions on Geoscience and Remote Sensing, the IEEE Geoscience and Remote Sensing Letters, the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) and the IEEE International Conference on Computer Vision (ICCV). His current research interests include remote sensing image processing and analysis, computer vision, pattern recognition, and machine learning.

Dr. Shi serves as an Editor for the Pattern Recognition, the ISPRS Journal of Photogrammetry and Remote Sensing, and the Infrared Physics and Technology, etc. His personal website is http://levir.buaa.edu.cn/.