COSMIC: Compress Satellite Images Efficiently via Diffusion Compensation

Ziyuan Zhang[1]   **Han Qiu**[1]*   **Maosen Zhang**[1]   **Jun Liu**[1]*
Bin Chen[2]   **Tianwei Zhang**[3]   **Hewu Li**[1]

[1] Tsinghua University, China
[2] Harbin Institute of Technology, Shenzhen, China
[3] Nanyang Technological University, Singapore

{ziyuan-z23,zhangms24}@mails.tsinghua.edu.cn, {qiuhan,juneliu}@tsinghua.edu.cn
chenbin2021@hit.edu.cn, tianwei.zhang@ntu.edu.sg, lihewu@cernet.edu.cn

Abstract

With the rapidly increasing number of satellites in space and their enhanced capabilities, the amount of earth observation images collected by satellites is exceeding the transmission limits of satellite-to-ground links. Although existing learned image compression solutions achieve remarkable performance by using a sophisticated encoder to extract fruitful features as compression and using a decoder to reconstruct, it is still hard to directly deploy those complex encoders on current satellites' embedded GPUs with limited computing capability and power supply to compress images in orbit. In this paper, we propose COSMIC, a simple yet effective learned compression solution to transmit satellite images. We first design a lightweight encoder (i.e. reducing FLOPs by $2.6 \sim 5\times$) on satellite to achieve a high image compression ratio to save satellite-to-ground links. Then, for reconstructions on the ground, to deal with the feature extraction ability degradation due to simplifying encoders, we propose a diffusion-based model to compensate image details when decoding. Our insight is that *satellite's earth observation photos are not just images but indeed multi-modal data with a nature of Text-to-Image pairing* since they are collected with rich sensor data (e.g. coordinates, timestamp, etc.) that can be used as the condition for diffusion generation. Extensive experiments show that COSMIC outperforms state-of-the-art baselines on both perceptual and distortion metrics. The code is publicly available at https://github.com/Joanna-0421/COSMIC.

1 Introduction

The revival of the aerospace industry [19, 38], coupled with reduced costs of launching rockets [22], has fueled an exponential increase in the number of nanosatellites, resulting in massive growth in images collected in-orbit. For example, the Sentinel-3 missions can collect a maximum of 20 TB raw data on satellites (mainly earth observation images) every day [20]. However, the data transmission capability between satellites and ground stations has clear upper bounds [18, 45, 17]. This situation of the rapid growth of images collected by satellites versus the limited transmission capability to the ground requires effective image compression on satellites before transmission back to Earth.

Current industrial compression solutions for satellite images rely on JPEG [48], JPEG2000 [46], or CCSDS123 [29] (e.g. satellite BilSAT-1 [56]). These solutions are outperformed by various learned compression methods [12] in various cases. Existing learned image compression methods [39, 36, 53] use sophisticated encoders to extract fruitful features and then use a decoder to decompress [32, 54].

---

*Corresponding authors.

Although we notice a novel promising trend of *deploying embedded GPUs on satellites* in both academia [16, 17, 6, 45] and industry (e.g. satellite Phi-Sat-1 [25], Chaohu-1 [3], and Forest-1 [4]) which brings the potential opportunity of using learned compressors on satellites. It is still hard to directly adopt existing learned compression solutions for satellites since their sophisticated encoders are still too complex for GPUs on satellites (e.g. NVIDIA Jetson Xavier NX on Forest-1 [4]) which have limited computing capacity and power supply [1]. We have two insights to fill the above gaps. (1) We first design a lightweight encoder on satellites with a higher priority of compression ratio than feature extraction ability. (2) At the receiver's end on the ground, we deal with this simple encoder's feature extraction ability degradation by compensating image contents when decoding. We choose diffusion as compensation due to its powerful generation capability and, more importantly, *satellites' earth observation photos are not only images but enjoy a multi-modal nature in which rich real-time sensor information is the description of the corresponding photo*. For instance, in Figure 1, the coordinates (e.g. latitude and longitude) denote the location of the image which describes its main category (e.g. sea, city, etc.) and the timestamp describes the image's lightning-like day or night.

In this paper, we propose to **CO**mpress **S**atellite i**M**age via d**I**ffusion **C**ompensation (COSMIC), a novel learned image compression method for satellites. COSMIC has two key components, i.e., (1) a lightweight encoder for compression on satellite and (2) a sophisticated decompression process with a decoder and a diffusion model on the ground with sufficient GPUs. First, we design a lightweight convolution architecture to extract local features and apply convolution to obtain an attention map of global features, to realize a lightweight image compression encoder in terms of FLOPs. Please note that lightweight encoders usually extract fewer key features which increases the difficulty for the decoder to decompress. Thus, our second component, decompression, has two parts including decoding with a corresponding decoder and, more importantly, a compensation model. Inspired by the multi-modal nature of the satellite's images, we aim to build text-to-image pairs (image and its sensor information like Figure 1) and use diffusion as the compensation model.
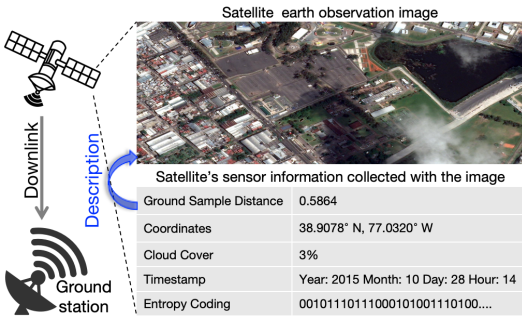

Figure 1: An example of the satellite's earth observation image and this image's corresponding sensor data as a description.

We compared COSMIC with 6 state-of-the-art (SOTA) baselines, 3 of which are based on generative models, considering both distortion and perceptual metrics. In addition, we constructed two image compression test sets based on satellite images by considering ordinary scenes and unique tile scenes in satellite imagery. Extensive experiments have proven that COSMIC significantly reduces the encoder's complexity to $2.6 \sim 5\times$ fewer FLOPs while achieving better performance on almost all metrics than baselines. Our contributions can be summarized as follows.

- We propose a novel idea that uses a lightweight image compression encoder on satellites and leverages satellite images' text-to-image pairing nature for compensation when decompressing.
- We propose a novel compensation model based on stable diffusion to compensate image details when decompressing with the unique sensor data of satellite images as descriptions.
- We analyzed the characteristics of satellite images in detail and incorporated them into the training and inference stages. In addition, we constructed two datasets under satellite image transmission scenarios, taking into account the typical satellite image transmission tasks like tile scenes.

2 Background

2.1 Earth observation missions on satellites

Earth observation missions (e.g. NASA's Landsat Program [49]) involve the use of satellite photos to monitor and collect data for tasks like forestry [5], agriculture [14, 41], land degradation [15], land use and land cover [40], biodiversity [34], and water resource [35, 47]. Traditional earth observation missions rely on a pipeline in which satellites take photos and then send them back to ground stations for analysis. Recently, along with the reduced cost of launching rockets and manufacturing

nanosatellites [21, 6], the photos on satellites are rapidly increasing which brings novel challenges for transmitting photos back to the ground. A recent promising approach is to deploy embedded GPUs on satellites to support DNN models to either filter useless data before transmission or make partial processing tasks on satellites. For instance, ESA's satellite Phi-Sat-1 [25] first deploy Intel VPU on satellite to support DNN models for filtering useless photos (e.g. covered by clouds) that can save 30%+ transmission volume. OroraTech has launched AI nanosatellites with the NVIDIA Jetson Xavier NX for wildfire detection [4], and Orbital Sidekick uses NVIDIA Jetson AGX Xavier as the AI engine at the edge of the satellite to detect gas pipeline leaks [2]. However, due to the inelastic computational capabilities of onboard satellites and limited power supply only from the sunshine (i.e., up to 15 Watt for GPUs on satellites [4]), a certain amount of images are still needed to be transmitted back to the ground. This brings an urgent need for satellite-specific image compression methods.

2.2 Neural image compression methods

Learned image compression methods have achieved remarkable rate-distortion performance compared with classical information theory-based image compression methods, to each of which consists of an encoder $\mathcal{E}$, a quantization $\mathcal{Q}$, and a decoder $\mathcal{D}$. The encoder, as the most critical part, extracts key features from the image as the latent representation. Higher quality representation extracted by the encoder means less content loss at compressing which is more likely to reconstruct a higher quality image when decompression. Thus, SOTA approaches explore introducing more complex modules into the encoder. [37] integrates the transformer into the CNN encoder and uses the transformer-CNN mixture block to extract rich global features. The other approaches aims to reduce the complexity of the decoders that are deployed on edge devices like smartphones. For instance, [54] adopts shallow or even linear decoding transforms to reduce the decoding complexity, compensated by more powerful encoder networks and iterative encoding.

Generative models for decompression. Although VAE-based methods have achieved good performances, optimizing solely for mean square error (MSE) can lead to excessive image smoothing, resulting in visual artifacts. More recent works [36, 53, 30, 24] have combined VAEs with generative models (e.g. diffusion) to achieve better visual results. [52] uses a conditional diffusion model as the image compression decoder which improves visual results. [30] and [24] decouple the compression task and augmentation task, sending the output of the VAE codec to diffusion to predict the residual.

Satellite image compression method. There are some compression methods specifically for remote sensing images [57, 23, 50]. [50] uses discrete wavelet transform to divide image features into high-frequency features and low-frequency features, and design a frequency domain encoding-decoding module to preserve high-frequency information, thereby improving the compression performance. [23] explore local and non-local redundancy through a mixed hyperprior network to improve entropy model estimation accuracy. Few of these works focus on onboard deployment. [26] use the CAE model to extract image features and reduce the image dimension to achieve compression, and deploy the model on VPU. However, this method only considers the reduction of image dimension and does not consider the arithmetic coding process in the actual transmission process, resulting in the image compression rate can only be adjusted by changing the model architecture.

Limitations to use for satellites. Most of the above approaches don't consider lightweight compression encoders which makes them impractical to deploy on satellite's embedded GPUs constrained by computing capacity and power supply. The approach used on VPU is impractical as the image compression rate is highly related to model architecture. Besides, none of them pay attention to the multi-modal nature of satellite earth observation images to introduce conditions to further improve decompression quality.

3 Prelimiaries

3.1 Problem formulation

We formulate the basic process of learned image compression. The encoder $\mathcal{E}$ uses a non-linearly transformation to convert the input image $\mathbf{x}$ into the latent representation $\mathbf{y}$, which is subsequently discretized and entropy-coded by quantization $\mathcal{Q}$ under a learned hyper prior ζ. Under the stochastic Gaussian model, each discrete code $\lfloor \mathbf{y} \rfloor_i$ can be expressed as a Gaussian distribution with mean μ_i and variance σ_i given a hyper prior ζ_i: $\mathrm{p}\left(\lfloor \mathbf{y} \rfloor_i | \zeta_i\right) = \mathcal{N}\left(\mu_i, \sigma_i^2\right)$. The decoder $\mathcal{D}$ reconstructs the

discrete representation $\lfloor \mathbf{y} \rceil$ to the image $\hat{\mathbf{x}}$. The model can be optimized by the loss function (Eq. 1).

$$\mathcal{L}_{\text{IC}} = \text{R} + \lambda \text{D} = \mathbb{E}\left[-\log_2 \text{p}\left(\lfloor \mathbf{y} \rceil | \zeta\right) - \log_2 \text{p}\left(\zeta\right)\right] + \lambda \mathbb{E}\left[\text{d}\left(\mathbf{x}, \hat{\mathbf{x}}\right)\right], \tag{1}$$

where R is the bit rate of latent discrete coding, D is the distortion between the original and the reconstructed image (measured by MSE), and λ controls the trade-off between rate and distortion.

3.2 Diffusion model

Diffusion model is a type of generative model that can generate images from Gaussian noise through multi-step iterative denoising. These models include two Markov processes. First, the diffusion process gradually applies noise to the image until the image is destroyed and becomes complete Gaussian noise. Then, in the reverse stage, it learns the process of restoring the Gaussian noise to the original image. During the inference stage, given a random noise sample $\mathbf{x}_T \sim \mathcal{N}\left(0, 1\right)$, the diffusion model can denoise through T steps and gradually generate a photorealistic image $\mathbf{x}_0$. At each step $\text{t} \in \{0, 1, ..., \text{T}\}$, intermediate variable $\mathbf{x}_t$ can be expressed as $\mathbf{x}_t = \sqrt{1 - \beta_t}\mathbf{x}_{t-1} + \beta_t \boldsymbol{\epsilon}_t$, where $\beta_t \in (0, 1)$ is the variance hyperparameter of Gaussian distribution, and satisfies $\beta_1 < \beta_2 < ... < \beta_T$; $\boldsymbol{\epsilon}_t \sim \mathcal{N}\left(0, 1\right)$ is the Gaussian noise at step t. In the diffusion model, a noise prediction network ($\boldsymbol{\epsilon}_\theta$) is used to predict the noise at step t, and $\mathbf{x}_{t-1}$ can be obtained from $\mathbf{x}_t$.

The diffusion model can also understand the content of the given conditions, such as text and images, and generate images consistent with the conditions. In this case, the noise prediction network takes three parameters: intermediate sample $\mathbf{x}_t$, timestep t, and given condition ς as input. To guarantee the noise predicted by the noise prediction network at the t-th step of the reverse process has the same distribution as the noise introduced into the image during the diffusion process, diffusion model usually use $\mathcal{L}_2$ to optimize the network following $\nabla_\theta \left\| \boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta\left(\mathbf{x}_t, \text{t}, \varsigma\right) \right\|$.

Stable diffusion [42] is proposed to reduce the training cost, which implements the diffusion process in a low-dimensional latent space while retaining the high-dimensional information in the original pixel space for decoding. In this article, we aim to leverage the powerful generation ability of stable diffusion and its ability to maintain consistency with a given condition to provide compensation for the information lost by the image compression encoder.

4 Method

4.1 Framework of COSMIC

As illustrated in Figure 2, COSMIC consists of two components: a compression module and a compensation module. To adapt to the satellite scenarios, the compression module includes a lightweight image compression encoder $\mathcal{E}$ and an entropy model deployed on the satellite (Sec. 4.2), as well as an image compression decoder $\mathcal{D}$ deployed at the ground station. To fix the content detail loss caused by the lightweight encoder, a compensation module is proposed, which is entirely deployed at the ground station. It has an encoder $\tilde{\mathcal{E}}$, aiming to extract compensation information z_0 from original images, which is received by decoder $\mathcal{D}$ as compensation for latent representation $\mathbf{y}\prime$ extracted by $\mathcal{E}$ during training (Sec. 4.3). During the inference phase, the noise prediction network generates compensation information $z_0\prime$ from noise to simulate z_0 as a compensation (Sec. 4.4).

4.2 Lightweight image compression encoder

To make the image compression encoder $\mathcal{E}$ practical on satellites, we first make it lightweight. The main idea is to reduce the amount of calculation required for image compression in terms of FLOPs. We followed the classic architecture of image compression encoder [10], which is structured with downsampling convolutions with a stride of 2 and generalized divisive normalization (GDN) [9] arranged alternately (Figure 2 (a)). Since GDN provides the best performance for image compression when the number of channels is 192 [8], the increase in convolution filters will exponentially increase the calculation amount of convolution. To reduce the amount of computation required for image downsampling, we propose a lightweight convolution block (LCB), as in Figure 2 (d), which uses depthwise convolution to replace the ordinary convolution with a convolution kernel size of 5×5. To interact between different channels of the feature map, depthwise convolution is followed by a 1×1 convolution with a full number of channels. Inspired by [27], which proves that there is a lot of
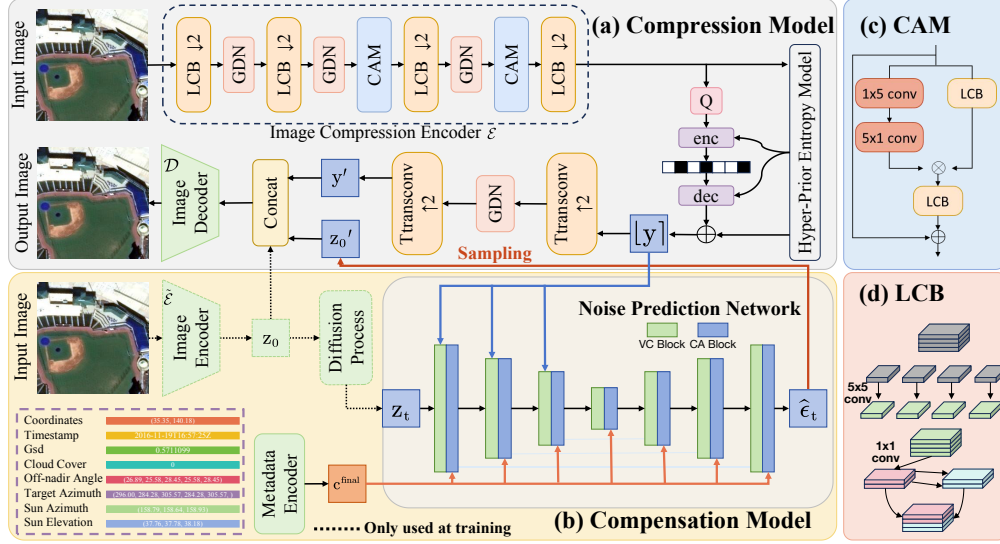
Figure 2: COSMIC **framework.** (a) Compression module for satellite images: a lightweight encoder and a compensation-based decoder (Sec. 4.3). (b) In the noise prediction network, each Cross-Attention (CA) block receives embedding of the Metadata Encoder (ME) (Sec. 4.4), and the Vanilla Convolution (VC) blocks use latent image discrete encoding to guide the prediction of noise for each diffusion step. (c) & (d) Convolution attention module and lightweight convolution block (Sec. 4.2).

redundancy in the features extracted by convolution, in LCB, only half of the output feature maps are obtained through 1×1 convolution, and the remaining half of the output feature maps are obtained through linear transformations with cheap cost using redundant features.

Transformer-based methods [58, 37] can outperform CNN-based methods as the attention can capture non-local information of the images. However, the computational complexity of self-attention has a quadratic relationship with the size of the input feature map, which is not computationally friendly. Inspired by [44], we use two one-dimensional convolutions in series. The first convolution is used to extract horizontal information. On this basis, the second convolution is used to vertically synthesize the previously extracted horizontal information to obtain the global attention map. Meanwhile, the other branch uses LCB with one stride to capture local information, as shown in Figure 2 (c).

4.3 Compensation-guided image compression

After giving a lightweight design of the image compression encoder $\mathcal{E}$, we note that the representation ability of this encoder is inevitably degraded. Specifically, the features contained in the latent representation $\lfloor \mathbf{y} \rceil$ received by the ground station are not enough to let the decoder reconstruct a high-quality image. Thus, we explore compensation for the degradation in encoding.

It is well-known that stable diffusion has powerful generation capabilities for specified content from noise under the guidance of text information [42]. The ground station can obtain not only the latent representation compressed by the encoder but also rich sensor data like the geographical location, time, camera parameters, etc. along with each image. We use these sensor data as conditions to guide diffusion generation to fix the missing image details. The training is divided into two stages. In the first stage, we train the compression model. As shown in (Figure 2 (b)), since the Image decoder $\mathcal{D}$ needs two parts of information (i.e. $\mathbf{y}\prime$ and $\mathbf{z_0}$ in Figure 2) for decoding, we introduce another image encoder $\tilde{\mathcal{E}}$ to extract compensation information $\mathbf{z_0}$ from the original image. In the first stage, $\mathcal{E}, \tilde{\mathcal{E}}$ and $\mathcal{D}$ are trained together. The reconstructed image $\hat{\mathbf{x}}$ can be expressed as in Eq. 2.

$$\hat{\mathbf{x}} = \mathcal{D}\left(\text{concat}\left(\text{transconv}\left(\lfloor \mathbf{y} \rceil\right), \tilde{\mathcal{E}}\left(\mathbf{x}_0\right)\right)\right) \tag{2}$$

In the second stage of training, we freeze the parameters of $\mathcal{E}, \tilde{\mathcal{E}}$ and $\mathcal{D}$, and train the noise prediction network, with the goal of making the information generated by the diffusion model as close to $\mathbf{z_0}$ as possible, denoted as $\mathbf{z_0}\prime$, so as to generate the compensation information required by the decoder.

5

During the inference phase, the trained diffusion model can generate compensation information $z_0\prime$. Therefore, we no longer need $\tilde{\mathcal{E}}$. The $z_0\prime$ generated by the diffusion model replaces the z_0 extracted by $\tilde{\mathcal{E}}$ to help the image decoder decompress the image.

4.4 Conditional diffusion model for loss compensation

As pointed out in Sec. 1, earth observation images collected by satellites are indeed multi-modal data. Here we consider that a satellite image x_0, when transmitted to the ground station, contains a discrete image coding $\lfloor y \rceil$ paired with its sensor information denoted as numerical metadata m. For $m \in \mathbb{R}^M$, just like diffusion handles timestep t, we use sinusoidal embedding (E_{sin}) to encode them to $c_j \in \mathbb{R}^{1 \times d}$ ($j = 1, 2, ...M$), where d is the dimension of the clip embedding, as we concatenate the metadata embedding together as a description of an image. This process is expressed as in Eq. 3.

$$c^{\text{final}} = \text{MLP}\left(\text{concat}\left(\left[E_{\sin}\left(m_1\right), ..., E_{\sin}\left(m_M\right)\right]\right)\right) \tag{3}$$

This final metadata condition will be incorporated into the latent representation using CrossAttention (CA) blocks to guide the generation process.

Stable diffusion was originally used for generative tasks, which have randomness. Here, we expect that stable diffusion generates image details that are not extracted by the satellite image encoder $\mathcal{E}$, and still retain the overall structure of the image. To address this problem, we inject the discrete image coding $\lfloor y \rceil$ into the Vanilla Convolution (VC) blocks of the noise prediction network to provide the structure information and improve content consistency, which can be described in Eq. 4.

$$f_i\prime = f_i + \text{projection}_i\left(\lfloor y \rceil\right), \tag{4}$$

where f_i is the i-th feature map of the U-Net backbone, and projection_i is the upsampling convolution used to align the dimensions between $\lfloor y \rceil$ and f_i. Guided by image coding and the metadata as a description, we use MSE loss to minimize the distance between target distribution and learned distribution in latent space as in Eq. 5.

$$\mathcal{L}_{\text{ldm}} = \mathbb{E}_{t,z_0,\epsilon \sim \mathcal{N}(0,1)}\left[\left\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta\left(z_t, t, \lfloor y \rceil, c^{\text{final}}\right)\right\|^2\right] \tag{5}$$

5 Experiments

5.1 Setup

Dataset. We use the function Map of the World (fMoW) [13], which has 62 categories, and in which each image is paired with different types of metadata features, as our training data and test data. For training data, we randomly crop the image to a resolution of 256×256 pixels. For information collected by satellite sensors as metadata, we choose Coordinates, Timestamp, GSD, Cloud cover, Off-nadir Angle, Target Azimuth, Sun Azimuth, and Sun Elevation provided by fMoW.

For testing data, we constructed two test sets with different resolutions. One is from the fMoW test set, where to ensure a comprehensive representation of categories, we randomly selected one image from each category, cropped it to a resolution of 256×256, and used it as a standard test set. For another test set, we considered the actual scenario of the satellite to construct a tile test set. Since the image captured by the satellite is a large geographic region, the computing resources on the satellite are limited, and large-size images cannot be processed directly, so images should first be cut into smaller sub-images, this process is known as tiling [31, 51, 17, 11]. In this paper, for images of size 2306×2306, we first divide each image into 81 smaller patches of size 256×256 each, and then compress and decompress each patch individually, as illustrated in Figure 5 (a). After obtaining all the decompressed patches, we reassemble them as one image for further evaluation.

Metrics. We use 4 metrics for quantitative measures following previous works [39, 36, 53]. For distortion comparison, we use the Peak Signal-to-Noise Ratio (PSNR) and Multi-Scale Structural Similarity Index Measure (MS-SSIM) to validate the pixel fidelity and measure brightness, contrast, and structural information at different scales. For perceptual comparison, we choose Learned Perceptual Image Patch Similarity (LPIPS) and Fréchet Inception Distance (FID).

Model training details. The training has two stages. First, we train the image compression encoder $\mathcal{E}$, image encoder $\tilde{\mathcal{E}}$ and image decoder $\mathcal{D}$ together using $\mathcal{L}_{\text{IC}}$ for 100 epochs with a batchsize of 32.

Second, we freeze the parameters of the model trained in the first stage, use the pretrained stable diffusion model for the noise prediction network, and finetune it using $\mathcal{L}_{ldm}$ for 10 epochs with a batchsize of 4. All the training experiments are performed on $10\times$ NVIDIA GeForce RTX 3090 using Adam optimizer with $lr = 1 \times 10^{-4}$ and $\lambda \in \{0.00067, 0.0013, 0.0026, 0.005\}$. During inference, we utilize the DDIM sampling [43] with 25 steps.

Baselines. We consider 6 baselines including traditional methods, VAE-based methods, and generative model based methods. **Elic** [28] proposes a multi-dimension entropy estimation model, which can effectively reduce the bit rate and improve the coding performance. **Hific** [39] pays more attention to the perception of the model reconstruction effect, obtaining visually pleasing reconstructed images. Based on Hific, **COLIC** [36] considers the semantic information of the image when designing the loss function, and treats structure and texture respectively. **CDC** [53] is the first work to use the diffusion model as an image compression decoder, performing the reverse process of diffusion in pixel space to reconstruct the image. **HL_RS** [50] is an image compression method, especially for remote sensing images, which processes the high-frequency part and the low-frequency part of the images separately to better preserve the important high-frequency features of remote sensing images. For these 5 baselines, we retrain their models with the fMoW dataset for a fair comparison. Besides, we choose JPEG2000, the industrial solution for satellite image compression [55], for comparison.

5.2 Comparison with baselines

RD performance. Figure 3 shows the comparison results with baselines on two test sets. The dotted line in the figure represents the baselines, and the solid line represents COSMIC. We demonstrate results from two perspectives, i.e., distortion and perception. Across all transmission rates, COSMIC surpasses the baselines in terms of LPIPS, and FID. At low bpp, the MS-SSIM of COSMIC is lower compared to the baseline. This is mainly because as the bpp decreases, the encoder extracts less information, and during the decompression process, there is a greater reliance on diffusion-based generation (more details in Sec. 5.4). Additionally, due to the degraded feature extraction of the lightweight encoder, the feature obtained at low bpp is insufficient to guide the diffusion process in generating high-fidelity images. As the bpp increases, the latent coding contains more features, resulting in a significant improvement in the MS-SSIM of COSMIC, demonstrating SOTA performance. Note that the sensor data is transmitted to the ground by default in earth observation missions. Besides, the volume of these sensor data is negligible compared with images so we do not consider them when counting bpp. We show more results of more metrics and baselines including other VAE-based methods in Appendix B which COSMIC achieves the SOTA performance on all 6 perceptual metrics.
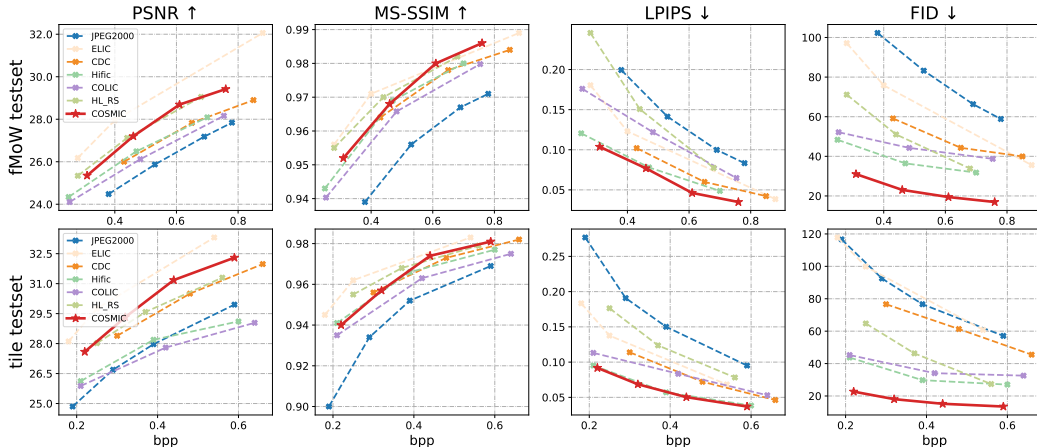

Figure 3: Trade-off between bitrates and different metrics on COSMIC and baselines. The $\uparrow$ ($\downarrow$) means higher (lower) is better. The first row is for the fMoW test set (image size 256×256). The second is for the tile test set by comparing between the stitched images and their original ones.

Visual results. Figure 4 shows the example of reconstructed images at low bitrates and high bitrates. For fair comparison, we only show the results of optimizing for image perception. Figure 5 (b) shows an example of high-resolution image reconstruction by COSMIC and baselines. Due to the tiling and

stitching process in image compression, we pay particular attention to the seams where different small patches form a larger image. JPEG2000 exhibits noticeable misalignment at the image seams. Hific and COLIC can not accurately restore the details of the seam. For example, in the picture outlined in orange, the car headlight at the seam has been reconstructed into a red dot. The diffusion-based CDC reconstruction also exhibits some color differences between different sub-images and shows noticeable misalignment, such as the eaves at the seam in the picture outlined in red. Compared to baselines, COSMIC maintains a higher similarity in structure and color between different sub-images, resulting in significant visual improvements.
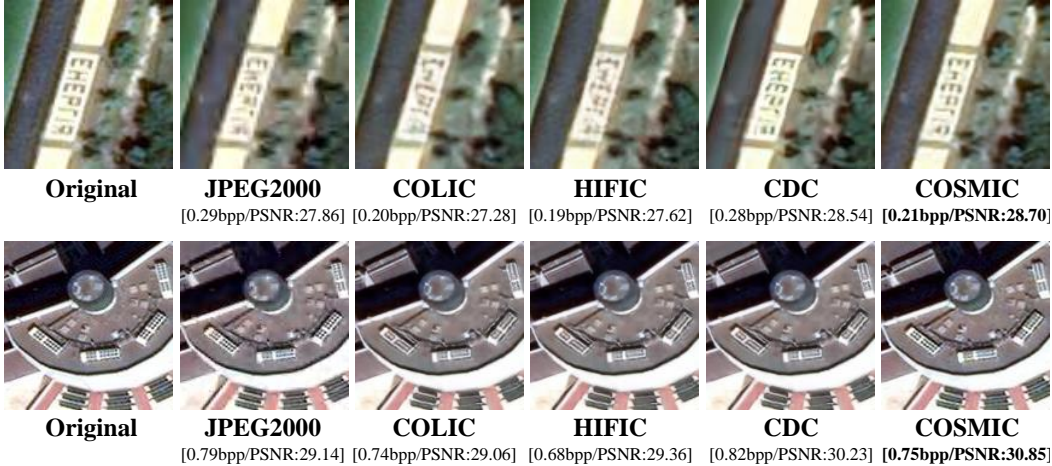

Original	**JPEG2000**	**COLIC**	**HIFIC**	**CDC**	**COSMIC**
	[0.29bpp/PSNR:27.86]	[0.20bpp/PSNR:27.28]	[0.19bpp/PSNR:27.62]	[0.28bpp/PSNR:28.54]	**[0.21bpp/PSNR:28.70]**

Original	**JPEG2000**	**COLIC**	**HIFIC**	**CDC**	**COSMIC**
	[0.79bpp/PSNR:29.14]	[0.74bpp/PSNR:29.06]	[0.68bpp/PSNR:29.36]	[0.82bpp/PSNR:30.23]	**[0.75bpp/PSNR:30.85]**

Figure 4: Decompressed fMoW images (full images in supplementary material). 1st row: comparison under low bitrates, COSMIC shows better visual effects. Compared with CDC, COSMIC still gets slightly better visual reconstruction with less bitrates. 2nd row: comparison under high bitrates.

5.3 Encoder efficiency analysis

We evaluate the lightweight encoder of COSMIC in terms of FLOPs in Table 1. Compared with baselines, the *on-satellite FLOPs* (including the encoder and entropy model) of COSMIC has been significantly reduced by roughly $2.6 \sim 5\times$ while the overall performance of COSMIC can still outperform baselines under a similar bitrate.

Table 1: Comparison of the on-satellite FLOPs with baselines on the tile test set. Best performances are highlighted in **bold**.

Method	FLOPs (G)	PSNR↑	MS-SSIM↑	LPIPS↓	FID↓	bpp↓
CDC [53]	13.1	31.98	0.982	0.0462	45.49	0.66
COLIC [36]	26.4	29.04	0.975	0.0530	32.56	0.64
Hific [39]	26.4	29.11	0.977	0.0384	27.01	0.60
Elic [28]	21.78	**33.31**	**0.983**	0.0683	60.80	0.54
HL_RS [50]	11.87	31.30	0.979	0.0782	27.38	0.56
COSMIC	**4.9**	32.11	0.980	**0.0359**	**13.50**	0.59

5.4 Ablation study

w/o DC. To demonstrate the compensatory role of diffusion in the image reconstruction process, we remove the diffusion compensation module, and the results are shown in Figure 6. We remove the compensation module and retrain the model to show the quantitative metrics. The result shows that the diffusion model plays an important role in decompressing images to get better perceptual metrics. To show the compensatory role of the diffusion model more clearly, we directly remove the diffusion model and only use the output information of the lightweight encoder $\mathcal{E}$ to reconstruct the image. We find that at low bitrate, diffusion compensation is more important. Due to the insufficient feature extraction capability of the lightweight encoder, many image content details are lost to save the transmission rate, and diffusion needs to reconstruct much of the image content guided by the limited output of the encoder, along with the metadata. As the bitrate increases, more features extracted by the encoder can be retained, and at this point, diffusion only needs to compensate for some image details that the encoder failed to capture. The visual results are shown in Figure 7. The results indicate that using the diffusion model as compensation is very useful, especially with a small bitrate.

(a) Illustration for tile testset

Original

(b) Visual result

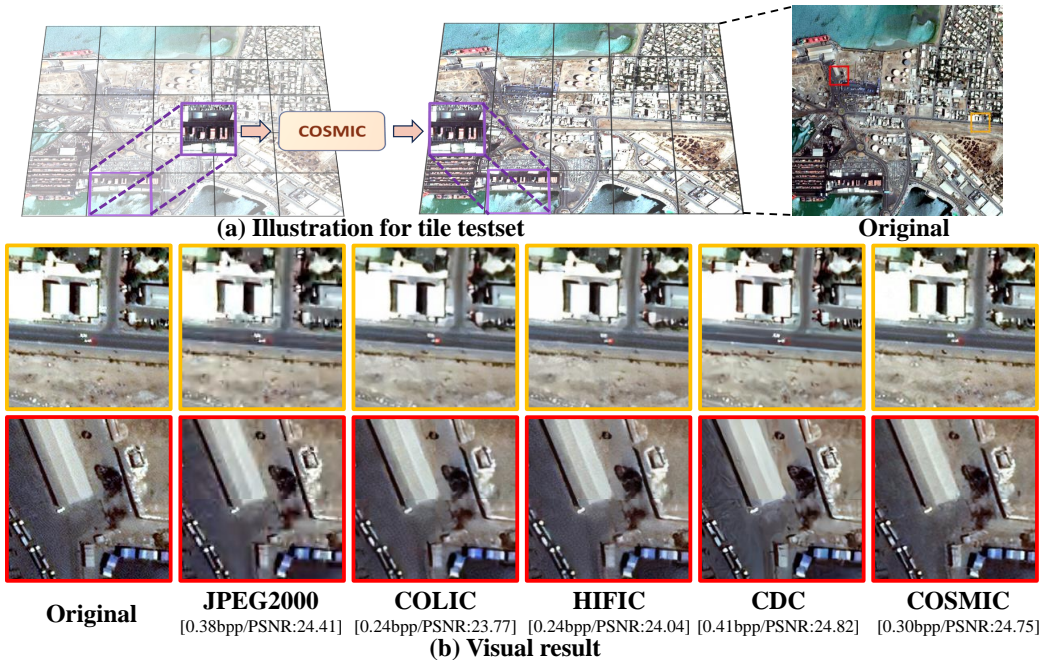| Original | JPEG2000 [0.38bpp/PSNR:24.41] | COLIC [0.24bpp/PSNR:23.77] | HIFIC [0.24bpp/PSNR:24.04] | CDC [0.41bpp/PSNR:24.82] | COSMIC [0.30bpp/PSNR:24.75] |

Figure 5: (a) Illustration of the tile test set. A high-resolution image is divided into many small sub-images (or patches), each of which is compressed individually. The reconstructed sub-images are then placed back in their original positions and stitched together to form a high-resolution reconstructed image. (b) On the tile test set, we provide two detailed views of a stitching area (outlined in orange and red). The visual comparison between COSMIC and the baseline shows that COSMIC achieves the best visual effects in terms of texture alignment and consistency in color brightness.

Table 2: Effect on image classification model.

Classes	Original	JPEG2K [46]	Elic [28]	COLIC [36]	HIFIC [39]	CDC [53]	HL_RS [50]	COSMIC
10	98.95%	-4.21%	-5.27%	-1.06%	-1.06%	-2.11%	-2.11%	-1.06%
15	97.92%	-4.17%	-3.84%	-0.70%	-0.70%	-1.40%	-1.39%	-0.70%
20	98.42%	-3.16%	-3.68%	-0.53%	-0.53%	-1.06%	-2.1%	-1.06%

w/o CAM. To show the CAM module can capture non-local information, which can help the encoder $\mathcal{E}$ to get higher quality representation. We remove the CAM module and show the results in Figure 6. The result shows that CAM module is effective and can achieve better RD performance.

w/o ME. To show that sensor data can guide the generation of diffusion, we remove the metadata encoder and only use image encoding for diffusion generation. Please note that there are mainly two kinds of sensor data here including the camera parameters such as the off-nadir angle and target azimuth used during image capture and the data on cloud cover, illumination, and ground sample distance, etc. Figure 6 indicates that sensor data helps guide the diffusion in reconstructing the image.

Influence of decoding steps. We further investigate the impact of different denoising step counts in the reverse diffusion process on the reconstructed image quality, as shown in Figure 8. We find that as the number of denoising steps increases, the perceptual metrics of the generated images gradually improve, which is consistent with the exploration of the relationship between denoising steps and FID scores in DDIM [43]. While the perceptual results improve, distortion metrics such as PSNR experience a slight decline, showcasing the trade-off between perceptual quality and distortion. In practical deployment and downstream tasks, the number of denoising steps can be selected based on different emphases on distortion and perceptual performance to suit actual applications.

5.5 Compression influence on downstream tasks

To confirm that COSMIC will not affect the downstream remote sensing tasks, such as image classification, we choose the image classification task in [7] to show the effect caused by compression, as shown in Table 2. The same test images are compressed at a low bitrate level and decompressed, then processed through the classification model to obtain the accuracy changes on different numbers of
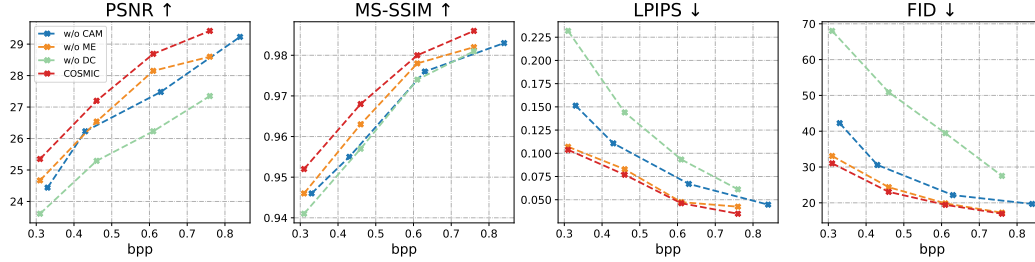
Figure 6: Ablation study with different variants of COSMIC. "w/o DC" indicates that diffusion compensation is not used during decoding. "w/o CAM"denotes the CAM module is removed in the encoder. "w/o ME" denotes that we remove the metadata encoder.

classes. The result shows that JPEG2000 and Elic reduce the accuracy by the most and COSMIC is outstanding in learning methods.
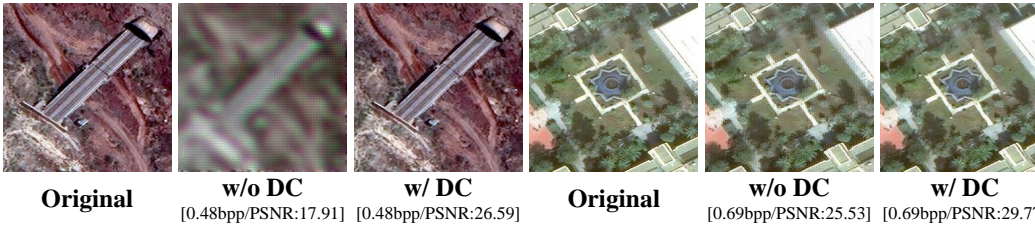

| **Original** | **w/o DC** | **w/ DC** | **Original** | **w/o DC** | **w/ DC** |
| | [0.48bpp/PSNR:17.91] | [0.48bpp/PSNR:26.59] | | [0.69bpp/PSNR:25.53] | [0.69bpp/PSNR:29.77] |

Figure 7: Contrast visual results. "w/ DC" means using of diffusion for compensation during decoding, while "w/o DC" indicates that diffusion compensation was not used during decoding.
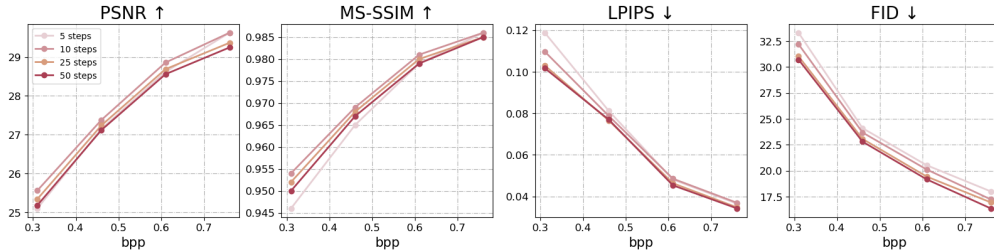

Figure 8: Compression performance with different numbers of decoding step.

6 Conclusion

We present COSMIC, a novel approach to compress images for satellite earth observation missions. We first design a lightweight encoder to adapt to the limited resources on satellites. Then, we introduce a conditional latent diffusion model by using the sensor data of satellite as instructions to compensate the missing details due to the lightweight encoder's degradation of feature extraction. Extensive results indicate that COSMIC can not only achieve SOTA compression performance for 2 typical satellite tasks but also guarantee the accuracy of satellite images' downstream tasks.

Limitations & Future work. Even though COSMIC compensates for encoder limitations using diffusion, at extremely low bpp (e.g., less than 0.1bpp), the information provided by latent image coding may not be enough to support diffusion in generating high-fidelity images. We think the main reason is that we only finetune the pretrained Stable Diffusion model, which lacks sufficient prior knowledge of satellite images. As a prospective solution, we will train a diffusion model specifically for satellite images or use historical satellite images as a reference for further improvements.

References

[1] 1. `https://www.nvidia.cn/autonomous-machines/embedded-systems/jetson-xavier-nx/`.

[2] 2. `https://blogs.nvidia.com/blog/orbital-sidekick/`.

[3] 3. `http://www.stardetect.cn/h-col-127.html`.

[4] 4. Shoot from the stars: Startup provides early detection of wildfires from space. `https://blogs.nvidia.com/blog/ororatech-wildfires-from-space/`.

[5] Frédéric Achard, Hans-Jürgen Stibig, Hugh D Eva, Erik J Lindquist, Alexandre Bouvet, Olivier Arino, and Philippe Mayaux. Estimating tropical deforestation from earth observation data. *Carbon Management*, 1(2):271–287, 2010.

[6] Caleb Adams, Allen Spain, Jackson Parker, Matthew Hevert, James Roach, and David Cotten. Towards an integrated gpu accelerated soc as a flight computer for small satellites. In *2019 IEEE aerospace conference*, pages 1–7. IEEE, 2019.

[7] Ali Bahri, Sina Ghofrani Majelan, Sina Mohammadi, Mehrdad Noori, and Karim Mohammadi. Remote sensing image classification via improved cross-entropy loss and transfer learning strategy based on deep convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 17(6):1087–1091, 2020.

[8] Johannes Ballé. Efficient nonlinear transforms for lossy image compression. In *2018 Picture Coding Symposium (PCS)*, pages 248–252. IEEE, 2018.

[9] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. Density modeling of images using a generalized normalization transformation. *arXiv preprint arXiv:1511.06281*, 2015.

[10] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436*, 2018.

[11] Marc Bosch, Kevin Foster, Gordon Christie, Sean Wang, Gregory D Hager, and Myron Brown. Semantic stereo for incidental satellite images. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1524–1532. IEEE, 2019.

[12] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7939–7948, 2020.

[13] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6172–6180, 2018.

[14] Laura Crocetti, Matthias Forkel, Milan Fischer, František Jurečka, Aleš Grlj, Andreas Salentinig, Miroslav Trnka, Martha Anderson, Wai-Tim Ng, Žiga Kokalj, et al. Earth observation for agricultural drought monitoring in the pannonian basin (southeastern europe): current state and future directions. *Regional Environmental Change*, 20:1–17, 2020.

[15] Rogier de Jong, Sytze de Bruin, Michael Schaepman, and David Dent. Quantitative mapping of global land degradation using earth observations. *International Journal of Remote Sensing*, 32 (21):6823–6853, 2011.

[16] Bradley Denby and Brandon Lucia. Orbital edge computing: Nanosatellite constellations as a new class of computer system. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 939–954, 2020.

[17] Bradley Denby, Krishna Chintalapudi, Ranveer Chandra, Brandon Lucia, and Shadi Noghabi. Kodan: Addressing the computational bottleneck in space. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3*, pages 392–403, 2023.

[18] Kiruthika Devaraj, Ryan Kingsbury, Matt Ligon, Joseph Breu, Vivek Vittaldev, Bryan Klofas, Patrick Yeon, and Kyle Colton. Dove high speed downlink system. 2017.

[19] Lauren Dreyer. Latest developments on spacex's falcon 1 and falcon 9 launch vehicles and dragon spacecraft. In *2009 IEEE Aerospace conference*, pages 1–15. IEEE, 2009.

[20] Thomas Esch, Soner Üreyen, Julian Zeidler, Annekatrin Metz-Marconcini, Andreas Hirner, Hubert Asamer, Markus Tum, Martin Böttcher, S Kuchar, Vaclav Svaton, et al. Exploiting big earth data from space–first experiences with the timescan processing chain. *Big Earth Data*, 2 (1):36–55, 2018.

[21] M Esposito, BC Dominguez, M Pastena, N Vercruyssen, SS Conticello, C van Dijk, PF Manzillo, and R Koeleman. Highly integration of hyperspectral, thermal and artificial intelligence for the esa phisat-1 mission. In *Proceedings of the International Astronautical Congress IAC, Washington, DC, USA*, pages 21–25, 2019.

[22] Warren Frick and Carlos Niederstrasser. Small launch vehicles-a 2018 state of the industry survey. 2018.

[23] Chuan Fu and Bo Du. Remote sensing image compression based on the multiple prior information. *Remote Sensing*, 15(8):2211, 2023.

[24] Noor Fathima Khanum Mohamed Ghouse, Jens Petersen, Auke J Wiggers, Tianlin Xu, and Guillaume Sautiere. Neural image compression with a diffusion-based decoder. 2022.

[25] Gianluca Giuffrida, Luca Fanucci, Gabriele Meoni, Matej Batič, Léonie Buckley, Aubrey Dunne, Chris Van Dijk, Marco Esposito, John Hefele, Nathan Vercruyssen, et al. The ϕ-sat-1 mission: The first on-board deep neural network demonstrator for satellite earth observation. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2021.

[26] Giorgia Guerrisi, Fabio Del Frate, and Giovanni Schiavon. Artificial intelligence based on-board image compression for the ϕ-sat-2 mission. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2023.

[27] Kai Han, Yunhe Wang, Qi Tian, Jianyuan Guo, Chunjing Xu, and Chang Xu. Ghostnet: More features from cheap operations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1580–1589, 2020.

[28] Dailan He, Ziming Yang, Weikun Peng, Rui Ma, Hongwei Qin, and Yan Wang. Elic: Efficient learned image compression with unevenly grouped space-channel contextual adaptive coding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5718–5727, 2022.

[29] Miguel Hernández-Cabronero, Aaron B Kiely, Matthew Klimesh, Ian Blanes, Jonathan Ligo, Enrico Magli, and Joan Serra-Sagrista. The ccsds 123.0-b-2 "low-complexity lossless and near-lossless multispectral and hyperspectral image compression" standard: A comprehensive review. *IEEE Geoscience and Remote Sensing Magazine*, 9(4):102–119, 2021.

[30] Emiel Hoogeboom, Eirikur Agustsson, Fabian Mentzer, Luca Versari, George Toderici, and Lucas Theis. High-fidelity image compression with score-based generative models. *arXiv preprint arXiv:2305.18231*, 2023.

[31] Bohao Huang, Daniel Reichman, Leslie M Collins, Kyle Bradbury, and Jordan M Malof. Tiling and stitching segmentation output for remote sensing: Basic challenges and recommendations. *arXiv preprint arXiv:1805.12219*, 2018.

[32] Nick Johnston, Elad Eban, Ariel Gordon, and Johannes Ballé. Computationally efficient neural image compression. *arXiv preprint arXiv:1912.08771*, 2019.

[33] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. *Advances in Neural Information Processing Systems*, 35:23593–23606, 2022.

[34] Claudia Kuenzer, Marco Ottinger, Martin Wegmann, Huadong Guo, Changlin Wang, Jianzhong Zhang, Stefan Dech, and Martin Wikelski. Earth observation satellite sensors for biodiversity monitoring: potentials and bottlenecks. *International Journal of Remote Sensing*, 35(18): 6599–6647, 2014.

[35] Richard Lawford, Adrian Strauch, David Toll, Balazs Fekete, and Douglas Cripe. Earth observations for global water security. *Current Opinion in Environmental Sustainability*, 5(6): 633–643, 2013.

[36] Meng Li, Shangyin Gao, Yihui Feng, Yibo Shi, and Jing Wang. Content-oriented learned image compression. In *European Conference on Computer Vision*, pages 632–647. Springer, 2022.

[37] Jinming Liu, Heming Sun, and Jiro Katto. Learned image compression with mixed transformer-cnn architectures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14388–14397, 2023.

[38] Henry Martin, Conor Brown, Tristan Prejean, and Nathan Daniels. Bolstering mission success: Lessons learned for small satellite developers adhering to manned spaceflight requirements. 2018.

[39] Fabian Mentzer, George D Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. *Advances in Neural Information Processing Systems*, 33:11913–11924, 2020.

[40] Prem Chandra Pandey, Nikos Koutsias, George P Petropoulos, Prashant K Srivastava, and Eyal Ben Dor. Land use/land cover in view of earth observation: Data sources, input dimensions, and classifiers—a review of the state of the art. *Geocarto International*, 36(9):957–988, 2021.

[41] George P Petropoulos, Prashant K Srivastava, Maria Piles, and Simon Pearson. Earth observation-based operational estimation of soil moisture and evapotranspiration for agricultural crops in support of sustainable water management. *Sustainability*, 10(1):181, 2018.

[42] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[43] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[44] Yehui Tang, Kai Han, Jianyuan Guo, Chang Xu, Chao Xu, and Yunhe Wang. Ghostnetv2: enhance cheap operation with long-range attention. *Advances in Neural Information Processing Systems*, 35:9969–9982, 2022.

[45] Bill Tao, Om Chabra, Ishani Janveja, Indranil Gupta, and Deepak Vasisht. Known knowns and unknowns: Near-realtime earth observation via query bifurcation in serval. In *21st USENIX Symposium on Networked Systems Design and Implementation (NSDI 24)*, pages 809–824, 2024.

[46] David S Taubman, Michael W Marcellin, and Majid Rabbani. Jpeg2000: Image compression fundamentals, standards and practice. *Journal of Electronic Imaging*, 11(2):286–287, 2002.

[47] Soner Uereyen and Claudia Kuenzer. A review of earth observation-based analyses for major river basins. *Remote Sensing*, 11(24):2951, 2019.

[48] Gregory K Wallace. The jpeg still picture compression standard. *Communications of the ACM*, 34(4):30–44, 1991.

[49] Michael A Wulder, David P Roy, Volker C Radeloff, Thomas R Loveland, Martha C Anderson, David M Johnson, Sean Healey, Zhe Zhu, Theodore A Scambos, Nima Pahlevan, et al. Fifty years of landsat science and impacts. *Remote Sensing of Environment*, 280:113195, 2022.

[50] Shao Xiang and Qiaokang Liang. Remote sensing image compression based on high-frequency and low-frequency components. *IEEE Transactions on Geoscience and Remote Sensing*, 2024.

[51] Chen Xu, Xiaoping Du, Zhenzhen Yan, and Xiangtao Fan. Cloud-based parallel tiling algorithm for large scale remote sensing datasets. In *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*, pages 4030–4033. IEEE, 2022.

[52] Ruihan Yang and Stephan Mandt. Lossy image compression with conditional diffusion models. *arXiv preprint arXiv:2209.06950*, 2022.

[53] Ruihan Yang and Stephan Mandt. Lossy image compression with conditional diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024.

[54] Yibo Yang and Stephan Mandt. Computationally-efficient neural image compression with shallow decoders. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 530–540, 2023.

[55] Xiaqiong Yu, Jinxian Zhao, Tao Zhu, Qiang Lan, Lin Gao, and Lingzhi Fan. Analysis of JPEG2000 compression quality of optical satellite images. In *2022 2nd Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS)*, pages 500–503. IEEE, 2022.

[56] Gokhan Yuksel, Onder Belce, and Hakan Urhan. Bilsat-1: First earth observation satellite of turkey-operations and lessons learned. In *Proceedings of 2nd International Conference on Recent Advances in Space Technologies, 2005. RAST 2005.*, pages 846–851. IEEE, 2005.

[57] Lei Zhang, Xugang Hu, Tianpeng Pan, and Lili Zhang. Global priors with anchored-stripe attention and multiscale convolution for remote sensing images compression. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2023.

[58] Renjie Zou, Chunfeng Song, and Zhaoxiang Zhang. The devil is in the details: Window-based attention for image compression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17492–17501, 2022.

A   Details about COSMIC

A.1   Details of train and inference

We present a more detailed explanation of our two-stage training and inference pipeline in algorithm 1 and algorithm 2.

Algorithm 1: Two-stage training of COSMIC.

---

input   : A satellite image $\mathbf{x}$, referenced image $\mathbf{x}$(the same as satellite image, used only for training), metadata $\mathrm{m_i} \in \mathbb{R}^{\mathrm{M}}$
output : Reconstructed image $\hat{\mathbf{x}}$
Parameters: Image encoder on satellite $\mathcal{E}$, image encoder of diffusion $\tilde{\mathcal{E}}$, image decoder $\mathcal{D}$, noise prediction network ϵ_θ

   /* *Training stage 1*: Train $\mathcal{E}$, $\tilde{\mathcal{E}}$, and $\mathcal{D}$ together.   */
1 $\mathbf{y} = \mathcal{E}\left(\mathbf{x}\right)$;
2 $\zeta = \mathrm{entropy\_model}(\mathbf{y})$;
3 $\mu_{\mathrm{z}}, \sigma_{\mathrm{z}} = \tilde{\mathcal{E}}\left(\mathrm{x_0}\right)$;
4 $\varepsilon_{\mathrm{z}} \sim \mathcal{N}\left(0, \mathrm{I}\right)$;
5 $\mathrm{z_0} = \mu_{\mathrm{z}} + \sigma_{\mathrm{z}} * \varepsilon_{\mathrm{z}}$;
6 $\hat{\mathbf{x}} = \mathcal{D}\left(\mathrm{concat}\left(\mathrm{z_0}, \mathrm{deconv}\left(\lfloor\mathbf{y}\rceil\right)\right)\right)$;
7 optimize the parameters of $\mathcal{E}$, $\tilde{\mathcal{E}}$, and $\mathcal{D}$ following:
   $\mathcal{L}_{\mathrm{IC}} = \mathrm{R} + \lambda\mathrm{D} = \mathbb{E}\left[-\log_2 \mathrm{p}\left(\lfloor\mathbf{y}\rceil|\zeta\right) - \log_2 \mathrm{p}\left(\zeta\right)\right] + \lambda\mathbb{E}\left[\mathrm{d}\left(\mathbf{x}, \hat{\mathbf{x}}\right)\right]$

   /* *Training stage 2*: Freeze the parameters of $\mathcal{E}$, $\tilde{\mathcal{E}}$, and $\mathcal{D}$, and only
      update the parameters of ϵ_θ.   */
8 **repeat**
9   sample $\mathrm{x}, \mathrm{m} \sim \mathrm{dataset}$;
10     $\mathrm{t} \sim \mathcal{U}\left(1, 2, ..., \mathrm{T}\right)$;
11     $\epsilon_{\mathrm{t}} \sim \mathcal{N}\left(0, \mathrm{I}\right)$;
12     $\mathrm{c}^{\mathrm{final}} = \mathrm{metadata\_encoder}(m)$;
13   optimize the parameters of ϵ_θ following:
       $\mathcal{L}_{\mathrm{ldm}} = \mathbb{E}_{\mathrm{t}, \mathrm{z_0}, \epsilon \sim \mathcal{N}(0,1)}\left[\left\|\boldsymbol{\epsilon}_{\mathrm{t}} - \boldsymbol{\epsilon}_\theta\left(\mathrm{z_t}, \mathrm{t}, \lfloor\mathbf{y}\rceil, \mathrm{c}^{\mathrm{final}}\right)\right\|^2\right]$
14 **until** *converge*;

Algorithm 2: Compress and decompress pipeline of COSMIC.

---

input   : A satellite image $\mathbf{x}$, metadata $\mathrm{m_i} \in \mathbb{R}^{\mathrm{M}}$
output : Reconstructed image $\hat{\mathbf{x}}$
Parameters: Image encoder on satellite $\mathcal{E}$, image decoder $\mathcal{D}$, noise prediction network ϵ_θ

   /* *Compress*   */
1 $\mathbf{y} = \mathcal{E}\left(\mathbf{x}\right)$;
2 $\zeta = \mathrm{entropy\_model}(\mathbf{y})$;
3 $\lfloor\mathbf{y}\rceil = \mathrm{Quantization}\left(\mathrm{y}, \zeta\right)$;
   /* *Decompress*   */
4 sample $\mathrm{z_T} \sim \mathcal{N}\left(0, \mathrm{I}\right)$;
5 **for** $\mathrm{t} = \mathrm{T}, ..., 1$ **do**
6     $\epsilon_{\mathrm{t}} \leftarrow \epsilon_\theta\left(\mathrm{z_t}, \mathrm{t}, \lfloor\mathbf{y}\rceil, \mathrm{c}^{\mathrm{final}}\right)$;
7     $\mathrm{z_{t-1}} \leftarrow \mathrm{DDIM}\left(\mathrm{z_t}, \mathrm{t}, \epsilon_{\mathrm{t}}\right)$;
8 **end**
9 return $\mathrm{z_0}\prime$;
10 $\hat{\mathbf{x}} = \mathcal{D}\left(\mathrm{concat}\left(\mathrm{z_0}\prime, \mathrm{deconv}\left(\lfloor\mathbf{y}\rceil\right)\right)\right)$;
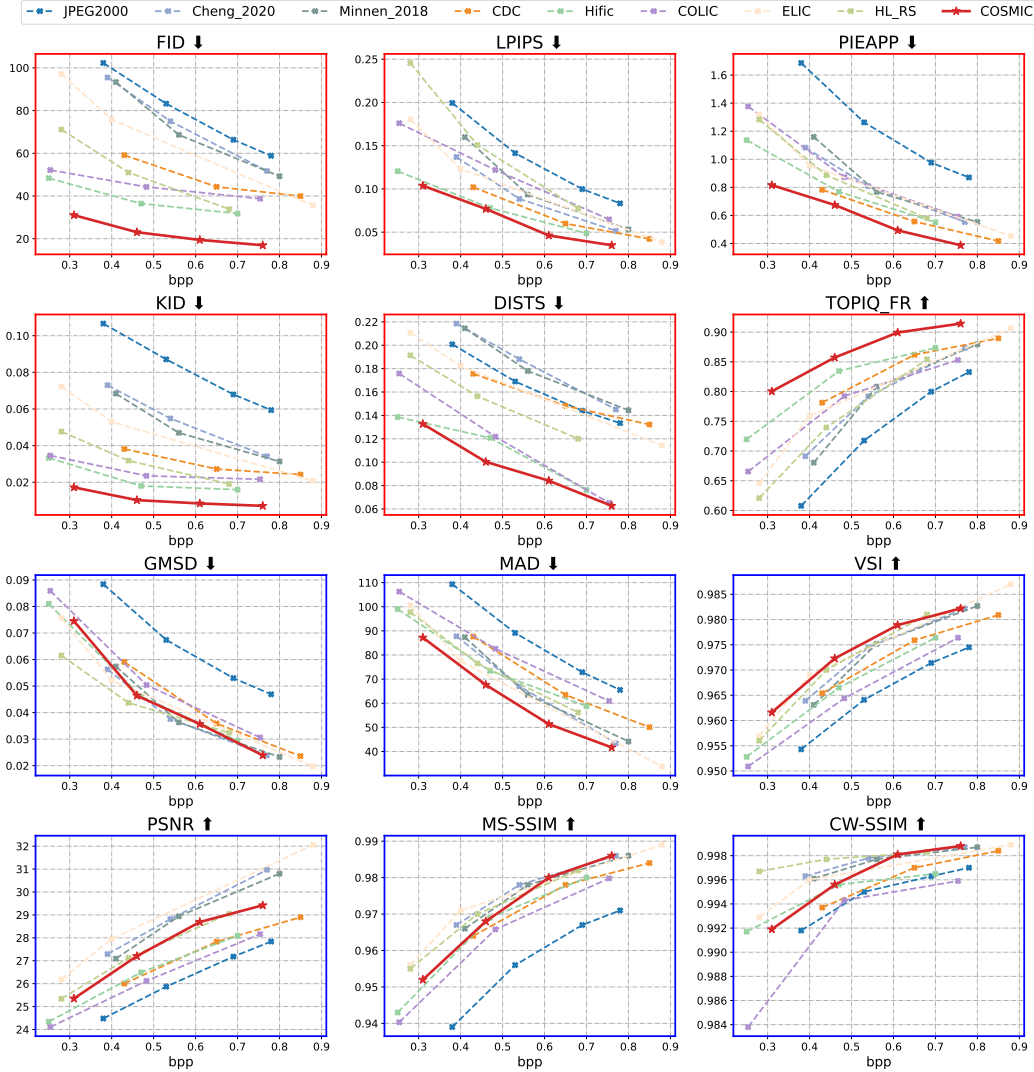

Figure 9: Rate-Distortion(Perception) for fMoW dataset.

B   Additional Rate-Distortion(Perception) Results

C   Failure cases analysis

Stable diffusion was originally used for generative tasks, which can produce textures and content that do not exist in reality. In COSMIC, we ensure the consistency of content and texture by injecting discrete latent coding $\lfloor \mathbf{y} \rceil$ into the Vanilla Convolution (VC) blocks of the noise prediction network to provide structural information. However, in some cases, especially at extremely low bitrates, the latent discrete coding may lack structural information, leading to the diffusion generating non-existent textures. As illustrated in Figure 10 (a), when a satellite image captures an ocean scene, the insufficient information in the latent coding at low bitrates requires heavy reliance on diffusion for generation, resulting in textures that do not exist in the reconstructed image. In the case of high bitrates, as described in the main paper, the reliance on diffusion reduces, and in the same scenario, the appearance of non-existent textures can be controlled. However, in practice, remote sensing downstream tasks are more concerned with regions of interest (ROIs). For example, in remote sensing tasks for oceans, such as ship detection , only the parts where ships are present are of interest. As shown in Figure 10 (b), even at low bitrates, while the reconstruction of the sea surface may introduce non-existent textures, the reconstruction of the ship parts remains intact. Therefore, although COSMIC may have minor visual flaws, it does not affect downstream detection tasks.
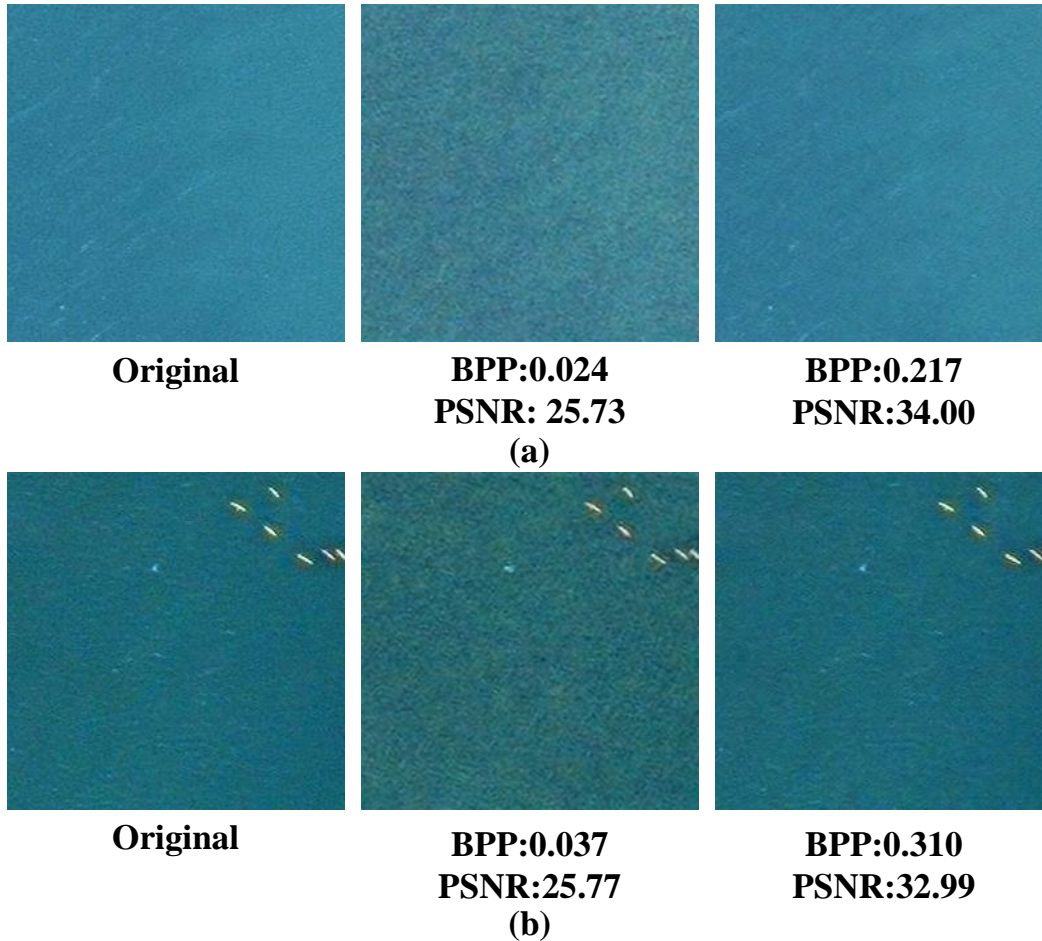
16

Figure 10: Failure examples.

D   Why not artifact correction for JPEG2000?

The most direct way to improve the quality of satellite images is to use artifact correction on JPEG2000 compressed images. This approach does not introduce the computational burden on the satellite and can also improve image quality to a certain extent. We use DDRM [33], which is an image restoration method, for JPEG2000 compressed images to remove the artifact. As shown in Figure 11, JPEG2000+DDRM is slightly better than JPEG2000 but the improvement is limited, and there is still a big gap with COSMIC. We believe that this is mainly due to the fact that traditional compression methods based on static compression procedures cannot fit the data well like neural networks, and therefore lose a large amount of information, making image restoration methods difficult to improve the image quality. As a result, it is necessary to design a learned compression method to improve satellite image compression quality without introducing excessive computational burden on the satellite.

E   Influence of random seed

To show the influence of different initial gaussian noise on the quality of reconstructed images, we randomly select 5 different seeds for each bit rate and show the 2-Sigma Error Bars results in Table 3. The results demonstrate that our method has small variances in various quantitative metrics, proving the robustness of our method to initialized random Gaussian noise.
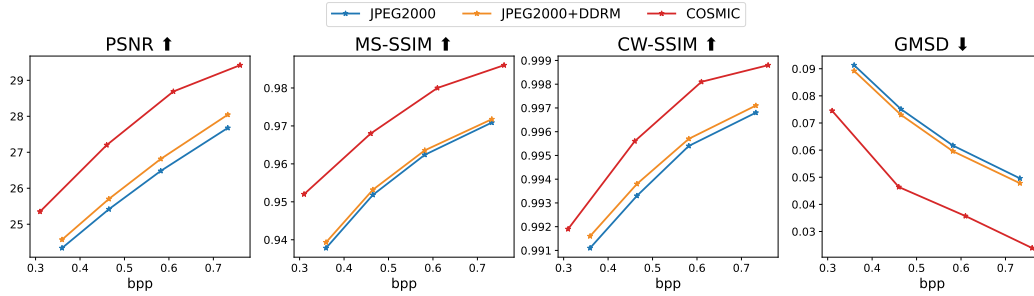
Figure 11: Comparison results of COSMIC, JPEG2000 and JPEG2000+DDRM.

Table 3: The results of different random seeds for fMoW dataset.

bpp	PSNR	MS-SSIM	LPIPS	FID
0.31	25.3698 ± 0.0447	0.9521 ± 0.0004	0.1040 ± 0.0010	31.2750 ± 0.4328
0.46	27.2411 ± 0.0222	0.9682 ± 0.0004	0.0761 ± 0.0006	23.0243 ± 0.2253
0.61	28.6654 ± 0.0251	0.9799 ± 0.0004	0.0461 ± 0.0004	19.4264 ± 0.1059
0.76	29.3617 ± 0.0608	0.9852 ± 0.0006	0.0363 ± 0.0007	16.9090 ± 0.1683

F   Additional visualization of reconstructed images

We show the full images of Figure 4 and Figure 5in the main paper, and give more visualization results, as shown in Figure 12 ∼ 25. We select visualization results for two different test sets under higher bitrates and lower bitrates from each dataset. For the tile test set, under lower bitrates, as shown in Figure 5 in the main paper, the reconstructed images by the baselines exhibit noticeable discontinuities at the stitching seams.

Original

COSMIC [0.21bpp/PSNR:28.70]

CDC [0.28bpp/PSNR:28.54]

JPEG2000 [0.29bpp/PSNR:27.86]

HIFIC [0.19bpp/PSNR:27.62]

COLIC [0.20bpp/PSNR:27.28]

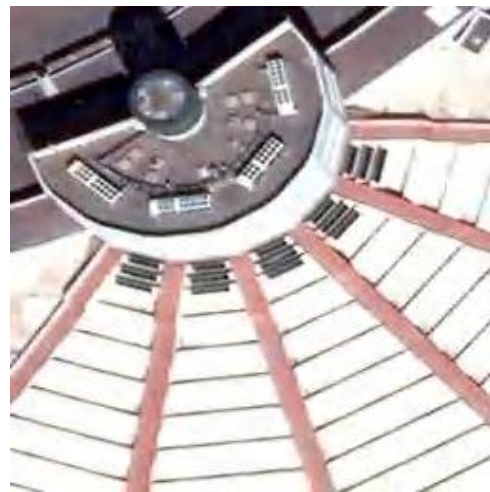Figure 12: Reconstructed fMoW images under low bitrates.

Original

COSMIC [0.75bpp/PSNR:30.85]

CDC [0.82bpp/PSNR:30.28]

JPEG2000 [0.79bpp/PSNR:29.14]

HIFIC [0.68bpp/PSNR:29.36]

COLIC [0.74bpp/PSNR:29.06]

Figure 13: Reconstructed fMoW images under high bitrates.

Figure 14: Ground truth.

Figure 15: **Low bpp**: COSMIC, PSNR=24.75, bpp=0.30.

Figure 16: **Low bpp**: HIFIC, PSNR=24.04, bpp=0.24.

Figure 17: **Low bpp**: COLIC, PSNR=23.77, bpp=0.24.

Figure 18: **Low bpp**: CDC, PSNR=24.82, bpp=0.41.

Figure 19: **Low bpp**: JPEG2000, PSNR=24.41, bpp=0.38.

Figure 20: Ground truth.

Figure 21: **High bpp**: COSMIC, PSNR=30.73, bpp=0.67.

Figure 22: **High bpp**: HIFIC, PSNR=29.77, bpp=0.65.

Figure 23: **High bpp**: COLIC, PSNR=29.82, bpp=0.70.

Figure 24: **High bpp**: CDC, PSNR=30.21, bpp=0.75.

Figure 25: **High bpp**: JPEG2000, PSNR=28.95, bpp=0.69.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have polished our abstract and introduction to accurately reflect our main contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations are discussed in Section 6

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: The paper does not propose a theory and does not have any theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Experiments are described in details, including the dataset we use, all the hyperparameters and the model training details.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
    (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
    (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
    (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
    (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The source code will be released upon acceptance with detailed instructions including data access and preparation, the exact command and the environment needed.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: All the detailed information are provided in experiment setup.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The random factor that affects our results is the sampling of Gaussian noise initialized for diffusion. We stochastically sample different Gaussian noise for the decompression process and show quantitative error metrics in the supplemental material.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We show the computer resources in experimental details ($10\times$ Nvidia GeForce RTX 3090 GPUs) in Section 5.1 and FLOPs of different methods in experiment results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: All the researches in this paper conform the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [No]

Justification: Our goal is as simple as to improve existing compression methods of transmitting images from satellites to grounds. This is not a security, safety, or privacy related research direction and is not related to any potential harmful or malicious usage in future.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All assets we used are properly cited.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not release new assets. The codes and model checkpoints will be released upon acceptance.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.