AlignDiff: Spatiotemporal Alignment for Remote Sensing Image Recovery via Terrain-Aware Diffusion

Anonymous submission

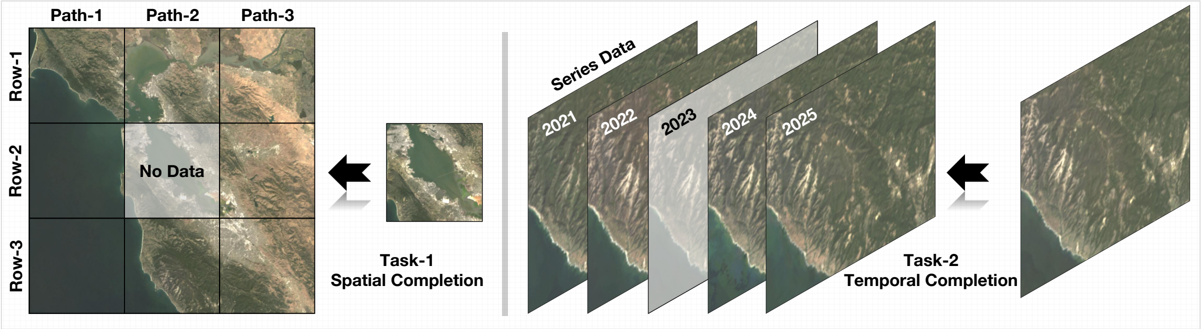


Figure 1: Motivation. We propose *AlignDiff*, a diffusion-based framework that formulates satellite image recovery as a spatiotemporal alignment task. The model addresses key challenges in remote sensing imagery—including cloud occlusion, acquisition gaps, and spectral inconsistency—by aligning missing regions with both spatial and temporal context through guided generation and geophysical priors.

Abstract

Remote sensing imagery plays a critical role in environmental monitoring and land assessment, but is often degraded by large missing regions due to cloud occlusion, sensor failure, or acquisition gaps. Existing interpolation and generative approaches struggle to reconstruct such data while maintaining spatial, spectral, and temporal coherence. We propose *AlignDiff*, a diffusion-based framework that formulates remote sensing image reconstruction as a **spatiotemporal alignment problem**. The model adopts a **three-way alignment strategy** to recover missing regions in a physically and semantically consistent manner: (1) **spatial alignment** through Digital Elevation Model (DEM) conditioning, (2) **semantic alignment** via prompt-based modulation, and (3) **distributional alignment** using a VGG-Adapter with feature-level consistency loss. Experiments on multi-region Landsat-8 imagery and EarthNet2021 demonstrate that *AlignDiff* outperforms state-of-the-art baselines on both spatial and temporal completion tasks. Our method enables scalable and reliable satellite image recovery. The dataset and code will be released upon acceptance.

Introduction

Remote sensing imagery serves as a critical data source for environmental monitoring, agricultural planning, and disaster response (Zhu et al. 2017; Zhang 2010). However, satellite observations are frequently disrupted by cloud cover, sensor failure, or acquisition discontinuities, particularly in high-resolution or high-frequency scenarios (Belgiu and Drăguţ 2016; Gorelick et al. 2017). These disruptions result in temporally and spatially incomplete datasets that degrade the accuracy of downstream analyses, hinder multi-source fusion, and weaken the capacity for reliable long-term observation. In climate-vulnerable and data-scarce regions, such deficiencies exacerbate global inequities in access to environmental intelligence.

Challenges. Traditional approaches, including interpolation methods and deep learning-based image completion models, typically operate at the pixel level, focusing on visually plausible reconstruction (Li et al. 2023; Zhang et al. 2023). However, they often fail to preserve structural, spectral, or temporal consistency, particularly when recovering semantically complex or geographically diverse regions (Gui et al. 2024). Many methods also overlook the physical priors inherent in remote sensing data—such as terrain morphology or acquisition geometry—and rely heavily on large-scale labeled datasets. Consequently, restored images may diverge from the underlying geophysical reality, limiting their usability in scientific and operational contexts.

Problem formulation. In this work, we reformulate satellite image reconstruction as a *spatiotemporal alignment problem*. Instead of treating missing data as isolated gaps to be filled, we aim to infer content that is structurally and semantically consistent with both the spatial landscape and the temporal dynamics of the scene. This formulation emphasizes context-aware reasoning and geophysically grounded alignment, moving beyond standard completion paradigms.

Proposed approach. We introduce ***AlignDiff***, a diffusion-based framework designed to model spatiotemporal alignment in remote sensing imagery. *AlignDiff* integrates Digital Elevation Models (DEMs) as geospatial priors and employs structured prompts that encode spatial location and acquisition time to guide the generation process. To ensure distribution-level consistency between generated and observed regions, we further incorporate a lightweight alignment adapter module that constrains statistical deviations during sampling.

Contributions.

- We formulate remote sensing image reconstruction as a **spatiotemporal alignment task**, unifying spatial and temporal recovery under a structured modeling perspective;
- We propose ***AlignDiff***, a diffusion-based generation framework that integrates terrain priors and prompt-based conditioning to align missing regions with physical and semantic context;
- We develop an **alignment adapter** that enforces distributional consistency, improving generalization across diverse terrain types and missing patterns.

Related Work

Data Issues and Traditional Approaches

Remote sensing imagery is widely used in earth observation and environmental monitoring, yet data issues such as cloud cover, sensor failures, and temporal discontinuities frequently lead to missing or corrupted data (Ju and Roy 2008; Xie et al. 2016; Daras et al. 2024). These gaps are particularly problematic for high-resolution imagery and long-term monitoring tasks, and disproportionately affect regions with persistent cloud coverage or limited data infrastructure—hindering access to timely and accurate environmental information.

Traditional methods for filling missing data rely on interpolation techniques such as nearest-neighbor (Rukundo and Cao 2012; Xing, Song, and Cheng 2022), bilinear (Gribbon and Bailey 2004; Yan et al. 2021), spline (McKinley and Levine 1998; Sun et al. 2023), and kriging (Oliver and Webster 1990; Jang, Kim, and Choo 2024) interpolation. While effective for small and smooth missing regions, these methods fail to reconstruct large gaps or preserve complex spatial patterns, often introducing blurriness or spectral inconsistencies (Kakar, Sudha, and Ser 2011; De Luca et al. 2024). These limitations restrict their applicability in real-world satellite pipelines that support environmental risk assessment, land management, or policy monitoring.

Machine Learning Methods

In recent years, machine learning and deep learning have revolutionized remote sensing image reconstruction. Convolutional Neural Networks (CNNs) (Gu et al. 2018) have been widely applied in image restoration due to their ability to capture spatial features. Partial Convolutions (Liu et al. 2018) and Generative Adversarial Networks (GANs) (Goodfellow et al. 2014) further advanced image completion, with methods such as Contextual Attention GAN (Yu et al. 2018) and the Globally and Locally Consistent Image Completion model (Iizuka, Simo-Serra, and Ishikawa 2017) demonstrating strong performance in generating visually plausible image completions.

Variational Autoencoders (VAEs) (Kingma 2013) have also been applied to image restoration by learning latent distributions, though they tend to generate over-smoothed results, limiting their applicability in high-resolution remote sensing imagery (Kingma, Welling et al. 2019).

Spatio-temporal models, such as Spatio-Temporal Convolutional Neural Networks (ST-CNN) (He, Chow, and Zhang 2019) and Spatio-Temporal Graph Convolutional Networks (ST-GCN) (Song et al. 2020), have been introduced to address dependencies in time-series satellite data. However, they often require large annotated datasets and struggle to generalize across geographical regions or sensor types, making deployment in low-resource or dynamic environments challenging.

Diffusion Models

Recent advances in generative modeling, particularly diffusion models, have significantly improved image synthesis tasks (Song and Ermon 2020; Dhariwal and Nichol 2021; Rombach et al. 2022). These models iteratively add and remove noise to generate high-quality images, with Denoising Diffusion Probabilistic Models (DDPM) (Ho, Jain, and Abbeel 2020) demonstrating remarkable success in natural image inpainting.

However, diffusion models face challenges in complex structured data. While methods such as Stable Diffusion (Rombach et al. 2022) optimize high-resolution image generation, they do not explicitly address spectral consistency across multiple bands, a critical factor in remote sensing. Additionally, ensuring that generated content aligns with physical ground truth data remains an open problem, particularly in domains like environmental monitoring where geospatial consistency is essential.

To introduce explicit control, ControlNet (Zhang, Rao, and Agrawala 2023) integrates external conditioning (e.g., edge maps, depth maps) into diffusion models, enabling stable and structured image generation. GeoSynth (Sastry et al. 2024) attempted to use basemaps as conditioning information, but such methods remain limited to qualitative reconstruction rather than precise quantitative remote sensing tasks.

Unlike natural image datasets, remote sensing images involve multi-band spectral information and strict spatial-temporal consistency, which traditional diffusion models do not explicitly handle. Moreover, most existing approaches lack integration with geographic priors such as terrain or vegetation indices, limiting their interpretability and deployment in socially critical applications such as food security estimation, climate resilience, and disaster preparedness. These limitations motivate the design of more control-

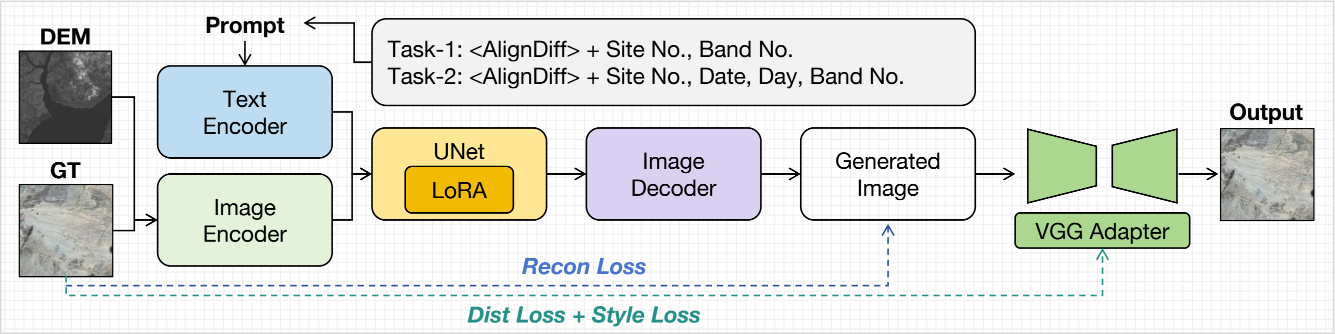


Figure 2: The *AlignDiff* pipeline integrates DEM-guided diffusion generation, LoRA-based prompt adaptation, and VGG-based distribution alignment for reconstructing missing satellite data across spatial and temporal gaps.

lable and context-aware generative frameworks for geospatial data restoration.

Proposed Framework: *AlignDiff*

We propose ***AlignDiff***, a unified diffusion-based generative framework that reconstructs missing satellite observations through three complementary alignment mechanisms: (1) **spatial alignment** via terrain-aware priors; (2) **distribution alignment** to enforce statistical and perceptual consistency; and (3) **semantic alignment** through prompt-conditioned generation. *AlignDiff* is designed to address both spatially incomplete observations across regions (**Task-1**) and temporally missing observations within spatiotemporal sequences (**Task-2**).

Problem Formulation and Task Setting

Let $x \in \mathbb{R}^{C \times H \times W}$ denote the complete ground-truth remote sensing image, and x_{obs} be its partially observed counterpart with missing regions $\mathcal{M} \subset \Omega$. The goal is to generate a completed image $\hat{x}$ that restores the missing content while preserving spatial structure, spectral distribution, and contextual coherence.

Formally, the model learns the conditional distribution:

$$p(\hat{x} \mid x_{\text{obs}}, a_{\text{spatial}}, a_{\text{dist}}, a_{\text{semantic}}) \tag{1}$$

where a_{spatial} denotes the spatial alignment input, typically represented by Digital Elevation Models (DEMs) that encode topographic structure; a_{dist} represents the distributional alignment objective, which enforces the alignment of feature distributions and textures between generated outputs and reference samples; and a_{semantic} corresponds to the semantic conditioning signal introduced via textual prompts, incorporating contextual information such as season, region, or spectral band type.

We consider two complementary reconstruction tasks:

(1) Task-1 (Spatial Completion): At a fixed timepoint t, reconstruct missing regions across spatially distinct tiles using DEM-guided priors and prompt-specific adaptation.

(2) Task-2 (Temporal Completion): At a fixed location s, recover observations at missing timepoints $\{t_k\}$ by leveraging seasonal semantics and historical dynamics.

Three-Way Alignment Strategy

To address the complex and heterogeneous nature of missing patterns in satellite imagery, *AlignDiff* integrates three alignment modules: spatial alignment with terrain priors, distribution alignment for perceptual fidelity, and semantic alignment via prompt-aware adaptation. Each module contributes complementary inductive biases, as detailed below.

Spatial Alignment with DEM Conditioning. To incorporate physical topography and enforce terrain-aware realism, we adopt a ControlNet-style branch (Zhang, Rao, and Agrawala 2023) that conditions the diffusion process on Digital Elevation Models (DEMs). This spatial alignment guides generation in mountainous, coastal, and varied terrain regions.

Let c_{spatial} denote the DEM-based conditioning input. The conditional denoising process is formulated as:

$$p(\hat{x} \mid c_{\text{spatial}}) = \prod_{t=1}^{T} p(x_t \mid x_{t-1}, c_{\text{spatial}}) \tag{2}$$

where x_t denotes the latent image at timestep t. The spatial reconstruction is supervised via a pixel-wise MSE loss:

$$\mathcal{L}_{\text{recon}} = \frac{1}{N} \sum_{i=1}^{N} \|x_i - \hat{x}_i\|^2 \tag{3}$$

Distribution Alignment with VGG Adapter. To correct brightness imbalance, texture distortion, and spectral inconsistencies, we introduce a VGG-based adapter that aligns feature distributions between generated and ground-truth images.

Let $\phi(\cdot)$ denote features extracted from a pretrained VGG-19 network. We minimize the Maximum Mean Discrepancy (MMD) between the empirical feature distributions:

$$\mathcal{L}_{\text{dist}} = \|\mathbb{E}_x[\phi(x)] - \mathbb{E}_{\hat{x}}[\phi(\hat{x})]\|^2 \tag{4}$$

Additionally, we include a multi-level style loss to regularize fine-grained texture via Gram matrix alignment:

$$\mathcal{L}_{\text{style}} = \sum_{l=1}^{L} \frac{1}{4C_l^2 H_l^2 W_l^2} \left\|G_{\hat{x}}^l - G_x^l\right\|_F^2 \tag{5}$$

where G^l is the Gram matrix of VGG features at layer l.

Semantic Alignment with Prompt. To condition the generation process on high-level semantics—such as location, seasonality, or band type—we introduce a lightweight adaptation module using Low-Rank Adaptation (LoRA) (Hu et al. 2021). A task-specific token `<AlignDiff>` is prepended to the text encoder, enabling fine-grained prompt control across domains.

For each task, the injected prompt is structured as:
Task 1: `<AlignDiff>Region, Band`
Task 2: `<AlignDiff>Region, Date, Day, Band`

LoRA adapts the attention weight matrix $W \in \mathbb{R}^{d\times d}$ by injecting a low-rank perturbation:

$$W' = W + AB, \quad A \in \mathbb{R}^{d\times r}, \quad B \in \mathbb{R}^{r\times d} \tag{6}$$

This design ensures efficient parameter tuning while maintaining generalization and cross-domain adaptability.

Unified Loss Function and Training

To jointly optimize spatial realism, statistical fidelity, and semantic control, we formulate a unified training objective that balances all three alignment mechanisms. The total loss is defined as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \lambda_{\text{dist}}\mathcal{L}_{\text{dist}} + \lambda_{\text{style}}\mathcal{L}_{\text{style}} \tag{7}$$

Here, λ_{dist} and λ_{style} are hyperparameters used to control the influence of perceptual distribution and style consistency. In practice, we set $\lambda_{\text{dist}} = 1.0$ and $\lambda_{\text{style}} = 100.0$ based on validation performance. The training process follows a gradient-based optimization, using DEM priors and task-specific prompts as conditional inputs. The full pipeline is detailed in Algorithm 1.

Algorithm 1: Training *AlignDiff*

```
Require: Dataset $X$, DEM $D$, Prompt $P$, Learning rate $\eta$,
    Epochs $E$
Initialize model parameters $\theta$
for $epoch = 1$ to $E$ do
    for each mini-batch $(x, d, p)$ in $(X, D, P)$ do
        Extract observed image $x_{\text{obs}}$ from $x$
        Generate reconstruction: $\hat{x} \leftarrow f_\theta(x_{\text{obs}}, d, p)$
        Compute loss $\mathcal{L}_{\text{total}}$ with Eq. (10)
        Update parameters: $\theta \leftarrow \theta - \eta\nabla_\theta\mathcal{L}_{\text{total}}$
    end for
end for
return Trained model $\theta$
```

Experiment

Dataset Description

Task 1 – Landsat-8 Multi-Region Dataset. We collect Landsat-8 imagery and Digital Elevation Model (DEM) data across 10 geographically diverse regions spanning various climate zones and land covers, including urban, agricultural, and natural areas. Table A1-A3 summarizes the selected regions, while Figure 3 shows their spatial distribution. All imagery was pre-filtered using Google Earth Engine (GEE) to ensure less than 1% cloud cover. The images (30m resolution) are tiled into 512 × 512 patches with 50% overlap to enhance spatial diversity and reduce edge effects. Detailed preprocessing scripts, including GEE filtering and masking routines, are available in Appendix 4.

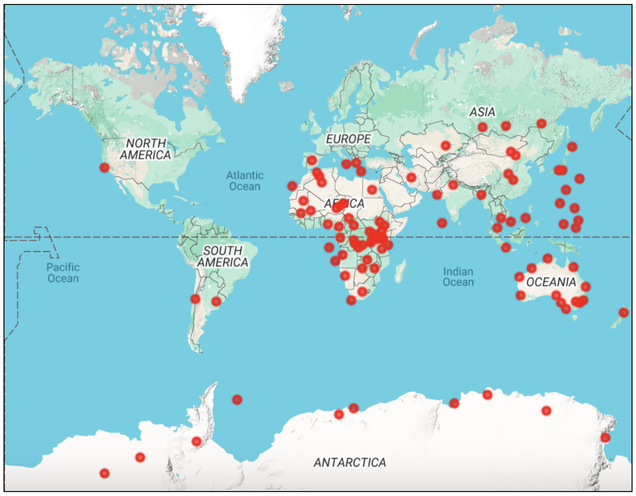


Figure 3: Global distribution of selected regions for Landsat imagery. The imagery was downloaded using Google Earth Engine (GEE) from various countries, ensuring diverse geographical coverage for comprehensive analysis.

Task 2 – EarthNet2021 Spatiotemporal Dataset. EarthNet2021 (Requena-Mesa et al. 2021) provides over 200,000 Sentinel-2 image sequences, offering rich temporal context for evaluating spatiotemporal modeling. We remove samples with more than 5% invalid pixels (e.g., clouds, shadows) based on quality masks. The data split follows the official protocol, including IID-test and OOD-test subsets, enabling rigorous assessment of generalization under in-distribution and out-of-distribution settings.

Masking Strategy. To simulate missing data and evaluate reconstruction performance, we adopt a standardized masking strategy. **Cloud Removal**: Pixels with high cloud probability are excluded to ensure clean ground truth. **Random Masking**: We randomly mask 10%–50% of valid pixels in each image to simulate missing regions. **Reproducibility**: The masked indices are stored and shared to ensure consistent evaluation and reproducibility. This masking strategy is uniformly applied across Task 1 and Task 2 to facilitate cross-dataset comparison.

Experimental Settings

All experiments were conducted on a single NVIDIA A100 GPU with 80 GB of memory. The diffusion model was trained using an initial learning rate of 5×10^{-5} and a target image resolution of 512 × 512 pixels. For inference, we adopted the DDIM scheduler with 50 steps, a guidance strength of 0.9, classifier-free guidance scale of 1.0, and $\eta = 1.0$. In baseline comparisons, the Spatio-Temporal CNN (STCNN) was trained using a batch size of 16, learning rate of 1×10^{-4}, and 100 training epochs. All mod-

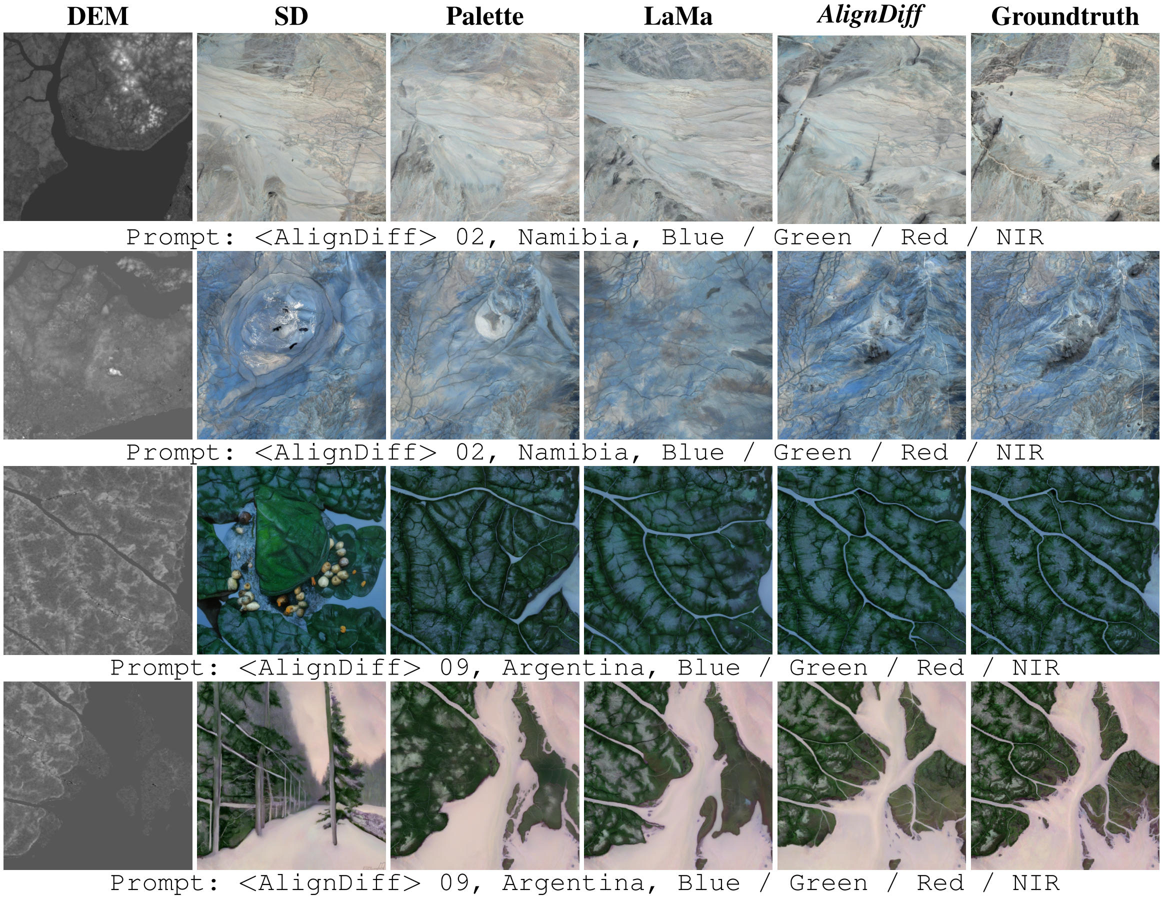


Figure 4: Comparison for Task-1, addressing missing data in specific regions over a fixed time period.

els were trained and evaluated using an 80%–20% train-test split for both Task 1 and Task 2.

Evaluation Metrics

To evaluate the reconstruction performance of *AlignDiff*, we use five standard metrics: Root Mean Square Error (RMSE) (Willmott and Matsuura 2005) and Mean Absolute Error (MAE) (Chai and Draxler 2014) measure pixel-level accuracy, with lower values indicating better fidelity. Peak Signal-to-Noise Ratio (PSNR) (Hore and Ziou 2010) and Structural Similarity Index Measure (SSIM) (Wang et al. 2004) assess image quality and structural similarity, where higher values reflect better reconstruction. Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al. 2018) evaluates perceptual similarity using deep features, with lower scores indicating closer alignment to the reference.

Comparison with Existing Methods

Task-1: Spatial Completion across Diverse Regions. We compare with state-of-the-art methods including natural image inpainting models, unconditioned diffusion models, and terrain-aware diffusion baselines.

Palette (Saharia et al. 2022) and LaMa (Suvorov et al. 2022), originally designed for natural image inpainting, demonstrate limited transferability to the satellite domain. Despite LaMa showing improved robustness among the two, both methods fail to preserve geophysical structure in large or topographically varied missing areas.

Stable Diffusion (SD) (Rombach et al. 2022), used as an unconditioned diffusion baseline, achieves the highest SSIM (0.5402 ± 0.012) among all models. However, its low PSNR (17.1599 ± 0.84) and high RMSE (0.1448 ± 0.007) reflect blurred reconstructions and spectral distortion, due to the absence of terrain- or time-specific conditioning.

ControlNet (Zhang, Rao, and Agrawala 2023), adapted with DEM guidance, significantly improves generation quality, yielding PSNR of 21.1847 ± 0.72 and RMSE of 0.0873 ± 0.005. This demonstrates the value of integrating physical priors into the generation process.

AlignDiff improves upon ControlNet by introducing a VGG-based adapter for distribution alignment, achieving the best PSNR (23.04 ± 0.69), RMSE (0.0713 ± 0.004), and MAE (0.0500 ± 0.002). Despite slightly lower SSIM, qualitative results (Fig. 4) show more coherent and terrain-consistent outputs, confirming the benefit of combining geospatial priors with perceptual alignment.

Task-2: Spatiotemporal Completion under Structured Constraints. In this setting, we evaluate the model's ability to restore missing observations across both spatial and tem-

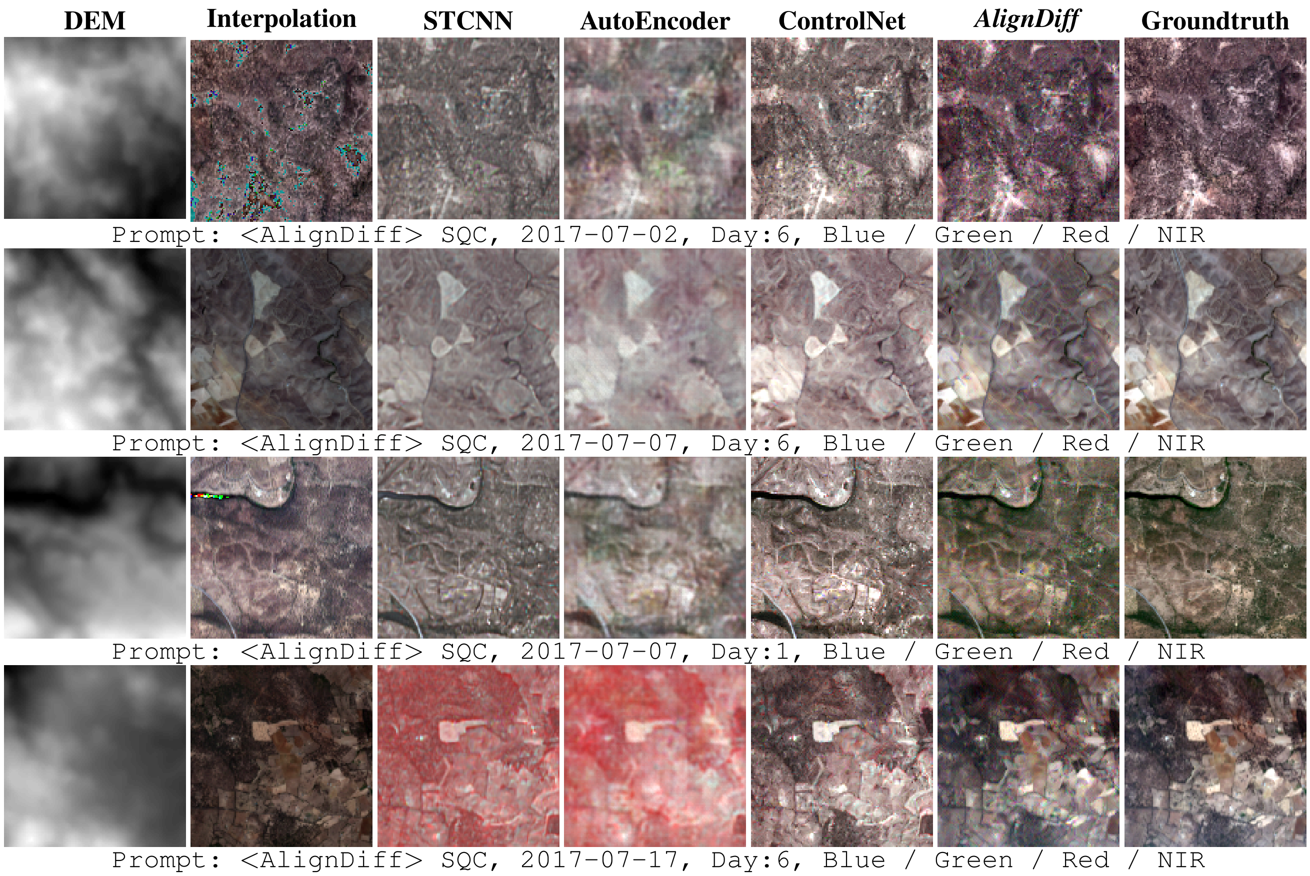


Figure 5: Comparison for Task-2 using the EarthNet2021 dataset with selected missing data days. The above images are the results after brightness adjustment and gamma correction, with a coefficient of 1.2. The original image was used for calculating the evaluation metrics.

poral dimensions. We consider two reconstruction scenarios: (1) temporal prediction from prior frames, and (2) spatial reconstruction from DEM priors. The results are averaged across 100 globally sampled locations with varied seasonal and spectral conditions.

Under the temporal setting, interpolation-based methods exhibit poor performance, with PSNR below 12.0 and high RMSE above 0.25, failing to exploit semantic or physical consistency. STCNN improves moderately (PSNR=14.5317 ± 0.65), but still lacks high-frequency details. AutoEncoder (Rifai et al. 2011) yields strong metrics (SSIM=0.6090 ± 0.013, PSNR=18.2038 ± 0.78), yet suffers from hallucination and texture mismatch in regions with significant temporal gaps.

In spatial reconstruction, diffusion models demonstrate superior stability. ControlNet achieves SSIM of 0.3787 ± 0.011 and PSNR of 22.7866 ± 0.73, substantially better than unconditioned models (SD: SSIM=0.2819 ± 0.012). However, ControlNet alone fails to ensure distributional consistency, often introducing spectral shifts and texture inconsistencies.

AlignDiff addresses these issues holistically by combining spatial priors, semantic prompts, and perceptual alignment. It outperforms all baselines in every metric—SSIM ($\mathbf{0.5704 \pm 0.011}$), PSNR ($\mathbf{24.3429 \pm 0.71}$), RMSE ($\mathbf{0.0642 \pm 0.003}$), MAE ($\mathbf{0.0479 \pm 0.002}$), and LPIPS ($\mathbf{0.0469 \pm 0.002}$). Compared to ControlNet, this represents a 50.68% improvement in SSIM and 11.56% reduction in RMSE. The integration of alignment modules enables *AlignDiff* to handle complex missing patterns across seasons, land types, and spectral bands—making it a robust candidate for real-world satellite observation recovery.

Ablation Study

VGG-Adapter Module. The proposed VGG-Adapter serves as a style alignment module to mitigate distributional discrepancies between generated outputs and reference data. As shown in Figure 6, omitting the adapter results in artifacts such as overexposed regions and brightness shifts, particularly in areas with complex land cover and elevation variability. Quantitatively, the mean brightness of reconstructed images *without* the adapter reaches $\mu = 123.63 \pm 6.02$, significantly deviating from the reference ($\mu = 95.19 \pm 3.87$). In contrast, *AlignDiff with VGG-Adapter* restores this value to $\mu = 95.27 \pm 3.82$, closely matching the ground truth distribution.

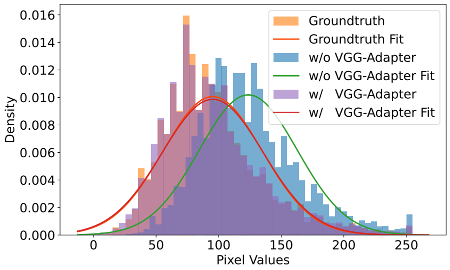
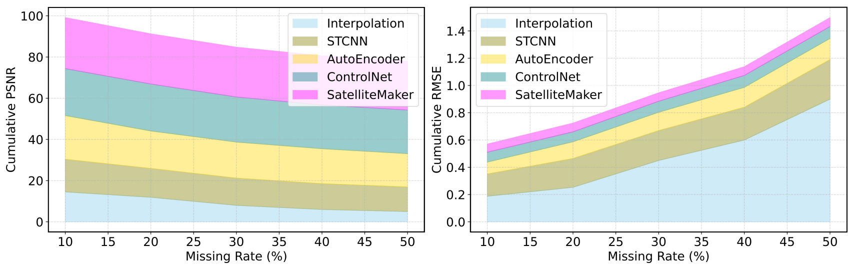


Figure 6: Ablation study on the impact of adding the VGG-Adapter.



(a) PSNR (b) RMSE

Figure 7: Comparison of different methods using PSNR and RMSE.

Table 1: Ablation study on the impact of the VGG-Adapter module in the *AlignDiff* framework across 100 sites. **Bold** indicates the best result; Underline denotes the second-best. The AVG row is computed as the mean of the four bands.

Band	**w/o VGG-Adapter**					**w/ VGG-Adapter**				
	SSIM↑	PSNR↑	RMSE↓	MAE↓	LPIPS↓	SSIM↑	PSNR↑	RMSE↓	MAE↓	LPIPS↓
Blue	0.3842 ± 0.024	17.8211 ± 1.102	**0.1295 ± 0.012**	**0.1047 ± 0.009**	0.1244 ± 0.008	0.5662 ± 0.020	24.1039 ± 0.954	0.0662 ± 0.006	0.0489 ± 0.004	0.0469 ± 0.005
Green	**0.3861 ± 0.025**	17.7244 ± 1.031	0.1302 ± 0.013	0.1055 ± 0.010	0.1217 ± 0.007	0.5774 ± 0.018	**24.1578 ± 0.822**	**0.0647 ± 0.005**	**0.0479 ± 0.004**	0.0478 ± 0.006
Red	0.3613 ± 0.029	15.1034 ± 1.354	0.1759 ± 0.015	0.1598 ± 0.012	0.1369 ± 0.009	**0.5829 ± 0.019**	**26.4456 ± 1.021**	**0.0503 ± 0.005**	**0.0387 ± 0.003**	**0.0388 ± 0.004**
NIR	0.3771 ± 0.022	17.0143 ± 0.889	0.1407 ± 0.011	0.1173 ± 0.008	**0.1181 ± 0.006**	0.5524 ± 0.023	23.0387 ± 0.978	0.0741 ± 0.006	0.0555 ± 0.004	0.0511 ± 0.005
AVG	0.3772 ± 0.025	16.9160 ± 0.874	0.1441 ± 0.012	0.1218 ± 0.009	0.1253 ± 0.007	0.5697 ± 0.020	24.4365 ± 0.844	0.0638 ± 0.006	0.0478 ± 0.004	0.0462 ± 0.005

Table 2: Performance comparison on **Task-1**. **Bold** indicates the best, underline the second-best result. All values are mean ± std over multiple trials.

Method	**SSIM↑**	**PSNR↑**	**RMSE↓**	**MAE↓**	**LPIPS↓**
SD	**0.5412 ± 0.023**	17.2635 ± 1.084	0.1452 ± 0.011	0.0916 ± 0.009	0.3378 ± 0.013
Palette	0.4350 ± 0.025	18.7321 ± 1.021	0.1218 ± 0.009	0.0821 ± 0.007	0.3424 ± 0.014
LaMa	0.4924 ± 0.018	20.8812 ± 0.994	0.0915 ± 0.008	0.0641 ± 0.006	0.2905 ± 0.012
ControlNet	0.4898 ± 0.020	21.2276 ± 0.935	0.0871 ± 0.007	0.0589 ± 0.005	**0.2872 ± 0.011**
AlignDiff	0.4563 ± 0.017	**23.1072 ± 0.879**	**0.0708 ± 0.006**	**0.0496 ± 0.004**	0.3417 ± 0.010

Table 3: Performance comparison on **Task-2**. The upper section uses the previous timestep as input; the lower uses DEM guidance. **Bold** denotes best, underline second-best. Values are reported as mean ± std over multiple trials.

Method	**SSIM↑**	**PSNR↑**	**RMSE↓**	**MAE↓**	**LPIPS↓**
Interpolation	0.5254 ± 0.028	11.9225 ± 0.845	0.2534 ± 0.022	0.2023 ± 0.019	0.3380 ± 0.017
STCNN	0.4047 ± 0.021	14.5317 ± 0.779	0.1877 ± 0.015	0.1498 ± 0.012	0.6835 ± 0.026
Autoencoder	**0.6090 ± 0.030**	**18.2038 ± 0.935**	**0.1230 ± 0.010**	**0.0981 ± 0.009**	**0.2487 ± 0.014**
STCNN	0.2232 ± 0.020	13.9769 ± 0.788	0.2107 ± 0.017	0.1723 ± 0.014	0.4208 ± 0.021
Autoencoder	0.2514 ± 0.024	14.7913 ± 0.801	0.2040 ± 0.016	0.1684 ± 0.013	0.3073 ± 0.019
SD	0.2819 ± 0.022	16.1943 ± 0.832	0.1550 ± 0.013	0.1335 ± 0.011	0.1468 ± 0.012
ControlNet	0.3787 ± 0.019	22.7866 ± 0.741	0.0726 ± 0.006	0.0570 ± 0.005	0.0721 ± 0.006
AlignDiff	**0.5704 ± 0.018**	**24.3429 ± 0.695**	**0.0642 ± 0.005**	**0.0479 ± 0.004**	**0.0469 ± 0.005**

Table 1 (averaged over 100 sites) further corroborates this improvement across all metrics. Notably, the red band—which is highly sensitive to spectral shifts—shows the most significant gain in PSNR (+11.01 dB) and RMSE reduction (-0.12), as illustrated in Figure 5. These results confirm that the VGG-Adapter contributes both perceptual and statistical alignment, and plays a critical role in enhancing output consistency across diverse geographies.

Missing Ratio. We also assess model robustness under varying missing ratios (10% to 50%), using PSNR and RMSE as primary indicators (see Figure 7). Traditional interpolation methods exhibit the lowest resilience, with PSNR plummeting from 14.53 ± 1.27 (at 10% missing) to below 5.00 ± 0.91 (at 50%), and RMSE increasing sharply. Such instability renders them unreliable in large-scale remote sensing scenarios.

STCNN and AutoEncoder maintain moderate robustness. For example, STCNN shows limited degradation (PSNR drops from 16.24 to 14.01), while AutoEncoder's PSNR drops more significantly (from 21.24 to 16.19), indicating weaker generalization under high-missing conditions. By contrast, diffusion-based models—especially *AlignDiff*—exhibit strong resilience. AlignDiff achieves 24.76 ± 0.73 PSNR at 10% missing and maintains 23.71 ± 0.81 at 50%, with the lowest RMSE across all levels. These results demonstrate that *AlignDiff* consistently preserves reconstruction quality, outperforming both deterministic and deep learning baselines under severe data loss.

Conclusion

We introduce *AlignDiff*, a diffusion-based generative framework tailored for remote sensing image reconstruction under missing data conditions. Addressing both spatial completion within fixed periods and temporal recovery in spatiotemporal sequences, *AlignDiff* leverages Digital Elevation Models (DEMs) as structural priors to guide terrain-consistent generation. In addition, a VGG-Adapter module is proposed to reduce distributional shifts via perceptual alignment, enhancing both realism and spectral fidelity. Extensive experiments on globally distributed Landsat-8 imagery and the EarthNet2021 benchmark demonstrate consistent improvements over existing inpainting and generative baselines, including Stable Diffusion, ControlNet, and AutoEncoder, across SSIM, PSNR, RMSE, MAE, and LPIPS. The proposed framework offers a scalable, generalizable, and geophysically grounded solution for satellite image recovery, particularly in regions with persistent data scarcity.

References

Belgiu, M.; and Drăguţ, L. 2016. Random forest in remote sensing: A review of applications and future directions. *ISPRS journal of photogrammetry and remote sensing*, 114: 24–31.

Chai, T.; and Draxler, R. R. 2014. Root mean square error (RMSE) or mean absolute error (MAE)?–Arguments against avoiding RMSE in the literature. *Geoscientific model development*, 7(3): 1247–1250.

Daras, G.; Shah, K.; Dagan, Y.; Gollakota, A.; Dimakis, A.; and Klivans, A. 2024. Ambient diffusion: Learning clean distributions from corrupted data. *Advances in Neural Information Processing Systems*, 36.

De Luca, G.; Carotenuto, F.; Genesio, L.; Pepe, M.; Toscano, P.; Boschetti, M.; Miglietta, F.; and Gioli, B. 2024. Improving PRISMA hyperspectral spatial resolution and geolocation by using Sentinel-2: development and test of an operational procedure in urban and rural areas. *ISPRS Journal of Photogrammetry and Remote Sensing*, 215: 112–135.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34: 8780–8794.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Gorelick, N.; Hancher, M.; Dixon, M.; Ilyushchenko, S.; Thau, D.; and Moore, R. 2017. Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote sensing of Environment*, 202: 18–27.

Gribbon, K. T.; and Bailey, D. G. 2004. A novel approach to real-time bilinear interpolation. In *Proceedings. DELTA 2004. Second IEEE international workshop on electronic design, test and applications*, 126–131. IEEE.

Gu, J.; Wang, Z.; Kuen, J.; Ma, L.; Shahroudy, A.; Shuai, B.; Liu, T.; Wang, X.; Wang, G.; Cai, J.; et al. 2018. Recent advances in convolutional neural networks. *Pattern recognition*, 77: 354–377.

Gui, S.; Song, S.; Qin, R.; and Tang, Y. 2024. Remote sensing object detection in the deep learning era—a review. *Remote Sensing*, 16(2): 327.

He, Z.; Chow, C.-Y.; and Zhang, J.-D. 2019. STCNN: A spatio-temporal convolutional neural network for long-term traffic prediction. In *2019 20th IEEE international conference on mobile data management (MDM)*, 226–233. IEEE.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Hore, A.; and Ziou, D. 2010. Image quality metrics: PSNR vs. SSIM. In *2010 20th international conference on pattern recognition*, 2366–2369. IEEE.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Iizuka, S.; Simo-Serra, E.; and Ishikawa, H. 2017. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)*, 36(4): 1–14.

Jang, D.; Kim, N.; and Choo, H. 2024. Kriging Interpolation for Constructing Database of the Atmospheric Refractivity in Korea. *Remote Sensing*, 16(13): 2379.

Ju, J.; and Roy, D. P. 2008. The availability of cloud-free Landsat ETM+ data over the conterminous United States and globally. *Remote Sensing of Environment*, 112(3): 1196–1211.

Kakar, P.; Sudha, N.; and Ser, W. 2011. Exposing digital image forgeries by detecting discrepancies in motion blur. *IEEE Transactions on Multimedia*, 13(3): 443–452.

Kingma, D. P. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kingma, D. P.; Welling, M.; et al. 2019. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4): 307–392.

Li, Z.-L.; Wu, H.; Duan, S.-B.; Zhao, W.; Ren, H.; Liu, X.; Leng, P.; Tang, R.; Ye, X.; Zhu, J.; et al. 2023. Satellite remote sensing of global land surface temperature: Definition, methods, products, and applications. *Reviews of Geophysics*, 61(1).

Liu, G.; Reda, F. A.; Shih, K. J.; Wang, T.-C.; Tao, A.; and Catanzaro, B. 2018. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European conference on computer vision (ECCV)*, 85–100.

McKinley, S.; and Levine, M. 1998. Cubic spline interpolation. *College of the Redwoods*, 45(1): 1049–1060.

Oliver, M. A.; and Webster, R. 1990. Kriging: a method of interpolation for geographical information systems. *International Journal of Geographical Information System*, 4(3): 313–332.

Requena-Mesa, C.; Benson, V.; Reichstein, M.; Runge, J.; and Denzler, J. 2021. EarthNet2021: A large-scale dataset and challenge for Earth surface forecasting as a guided video prediction task. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1132–1142.

Rifai, S.; Vincent, P.; Muller, X.; Glorot, X.; and Bengio, Y. 2011. Contractive auto-encoders: Explicit invariance during feature extraction. In *Proceedings of the 28th international conference on international conference on machine learning*, 833–840.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Rukundo, O.; and Cao, H. 2012. Nearest neighbor value interpolation. *arXiv preprint arXiv:1211.1768*.

Saharia, C.; Chan, W.; Chang, H.; Lee, C.; Ho, J.; Salimans, T.; Fleet, D.; and Norouzi, M. 2022. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 conference proceedings*, 1–10.

Sastry, S.; Khanal, S.; Dhakal, A.; and Jacobs, N. 2024. GeoSynth: Contextually-Aware High-Resolution Satellite Image Synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 460–470.

Song, C.; Lin, Y.; Guo, S.; and Wan, H. 2020. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 914–921.

Song, Y.; and Ermon, S. 2020. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33: 12438–12448.

Sun, M.; Lan, L.; Zhu, C.-G.; and Lei, F. 2023. Cubic spline interpolation with optimal end conditions. *Journal of Computational and Applied Mathematics*, 425: 115039.

Suvorov, R.; Logacheva, E.; Mashikhin, A.; Remizova, A.; Ashukha, A.; Silvestrov, A.; Kong, N.; Goka, H.; Park, K.; and Lempitsky, V. 2022. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2149–2159.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612.

Willmott, C. J.; and Matsuura, K. 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research*, 30(1): 79–82.

Xie, K.; Ning, X.; Wang, X.; Xie, D.; Cao, J.; Xie, G.; and Wen, J. 2016. Recover corrupted data in sensor networks: A matrix completion solution. *IEEE Transactions on Mobile Computing*, 16(5): 1434–1448.

Xing, Y.; Song, Q.; and Cheng, G. 2022. Benefit of interpolation in nearest neighbor algorithms. *SIAM Journal on Mathematics of Data Science*, 4(2): 935–956.

Yan, F.; Zhao, S.; Venegas-Andraca, S. E.; and Hirota, K. 2021. Implementing bilinear interpolation with quantum images. *Digital Signal Processing*, 117: 103149.

Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; and Huang, T. S. 2018. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5505–5514.

Zhang, J. 2010. Multi-source remote sensing data fusion: status and trends. *International Journal of Image and Data Fusion*, 1(1): 5–24.

Zhang, L.; Rao, A.; and Agrawala, M. 2023. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3836–3847.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 586–595.

Zhang, W.; Wang, Y.; Ni, B.; and Yang, X. 2023. Fully context-aware image inpainting with a learned semantic pyramid. *Pattern Recognition*, 143: 109741.

Zhu, X. X.; Tuia, D.; Mou, L.; Xia, G.-S.; Zhang, L.; Xu, F.; and Fraundorfer, F. 2017. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE geoscience and remote sensing magazine*, 5(4): 8–36.

Reproducibility Checklist

Instructions for Authors:

This document outlines key aspects for assessing reproducibility. Please provide your input by editing this `.tex` file directly.

For each question (that applies), replace the “Type your response here” text with your answer.

Example: If a question appears as

```
\question{Proofs of all novel claims
are included} {(yes/partial/no)}
Type your response here
```

you would change it to:

```
\question{Proofs of all novel claims
are included} {(yes/partial/no)}
yes
```

Please make sure to:

- Replace ONLY the “Type your response here” text and nothing else.
- Use one of the options listed for that question (e.g., **yes**, **no**, **partial**, or **NA**).
- **Not** modify any other part of the `\question` command or any other lines in this document.

You can `\input` this .tex file right before `\end{document}` of your main file or compile it as a stand-alone document. Check the instructions on your conference’s website to see if you will be asked to provide this checklist with your paper or separately.

---

1. General Paper Structure

1.1. Includes a conceptual outline and/or pseudocode description of AI methods introduced (yes/partial/no/NA) yes

1.2. Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results (yes/no) yes

1.3. Provides well-marked pedagogical references for less-familiar readers to gain background necessary to replicate the paper (yes/no) yes

2. Theoretical Contributions

2.1. Does this paper make theoretical contributions? (yes/no) no

If yes, please address the following points:

2.2. All assumptions and restrictions are stated clearly and formally (yes/partial/no) NA

2.3. All novel claims are stated formally (e.g., in theorem statements) (yes/partial/no) NA

2.4. Proofs of all novel claims are included (yes/partial/no) NA

2.5. Proof sketches or intuitions are given for complex and/or novel results (yes/partial/no) NA

2.6. Appropriate citations to theoretical tools used are given (yes/partial/no) NA

2.7. All theoretical claims are demonstrated empirically to hold (yes/partial/no/NA) NA

2.8. All experimental code used to eliminate or disprove claims is included (yes/no/NA) NA

3. Dataset Usage

3.1. Does this paper rely on one or more datasets? (yes/no) yes

If yes, please address the following points:

3.2. A motivation is given for why the experiments are conducted on the selected datasets (yes/partial/no/NA) yes

3.3. All novel datasets introduced in this paper are included in a data appendix (yes/partial/no/NA) yes

3.4. All novel datasets introduced in this paper will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no/NA) yes

3.5. All datasets drawn from the existing literature (potentially including authors' own previously published work) are accompanied by appropriate citations (yes/no/NA) yes

3.6. All datasets drawn from the existing literature (potentially including authors' own previously published work) are publicly available (yes/partial/no/NA) yes

3.7. All datasets that are not publicly available are described in detail, with explanation why publicly available alternatives are not scientifically satisficing (yes/partial/no/NA) NA

4. Computational Experiments

4.1. Does this paper include computational experiments? (yes/no) yes

If yes, please address the following points:

4.2. This paper states the number and range of values tried per (hyper-) parameter during development of the paper, along with the criterion used for selecting the final parameter setting (yes/partial/no/NA) yes

4.3. Any code required for pre-processing data is included in the appendix (yes/partial/no) yes

4.4. All source code required for conducting and analyzing the experiments is included in a code appendix (yes/partial/no) yes

4.5. All source code required for conducting and analyzing the experiments will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no) yes

4.6. All source code implementing new methods have comments detailing the implementation, with references to the paper where each step comes from (yes/partial/no) yes

4.7. If an algorithm depends on randomness, then the method used for setting seeds is described in a way sufficient to allow replication of results (yes/partial/no/NA) yes

4.8. This paper specifies the computing infrastructure used for running experiments (hardware and software), including GPU/CPU models; amount of memory; operating system; names and versions of relevant software libraries and frameworks (yes/partial/no) yes

4.9. This paper formally describes evaluation metrics used and explains the motivation for choosing these metrics (yes/partial/no) yes

4.10. This paper states the number of algorithm runs used to compute each reported result (yes/no) yes

4.11. Analysis of experiments goes beyond single-dimensional summaries of performance (e.g., average; median) to include measures of variation, confidence, or other distributional information (yes/no) yes

4.12. The significance of any improvement or decrease in performance is judged using appropriate statistical tests (e.g., Wilcoxon signed-rank) (yes/partial/no) no

4.13. This paper lists all final (hyper-)parameters used for each model/algorithm in the paper's experiments (yes/partial/no/NA) partial

Landsat Dataset

The data used in this study, including the Digital Elevation Model (DEM) and Landsat 8 imagery, were sourced from the Google Earth Engine (GEE) platform. GEE provides a vast repository of satellite imagery and geospatial datasets, which can be accessed programmatically using its API. Below is the code used for downloading the selected data for the study. This script extracts the required images, applies the necessary filters, and prepares the data for analysis. This code filters and downloads Landsat 8 imagery with less than 1% cloud cover and the corresponding DEM data for the specified region of interest (ROI). The data is then exported to Google Drive for further analysis. We selected 10 regions, with the range of the ROIs shown in Table **??**, and their spatial distribution illustrated in Fig. 3.

```

// Using Google Earth Engine JavaScript API to retrieve DEM and Landsat 8 imagery for specific regions

// Define Areas of Interest (AOI) with corresponding country/region names

// Region 8: Near Seattle, Washington, United States
var region8 = ee.Geometry.Polygon([
  [20.0, 64.0], [20.5, 64.0], [20.5, 64.5], [20.0, 64.5]
]);

// Using Region 8 as an example
var region = region8;

// Retrieve SRTM DEM data and clip it to the Area of Interest (AOI)
var srtm = ee.Image(’USGS/SRTMGL1_003’).clip(region).toFloat();

// Retrieve Landsat 8 imagery with cloud cover less than 1%, filtered by date and region, including all bands
var landsat = ee.ImageCollection(’LANDSAT/LC08/C02/T1_TOA’)
  .filterBounds(region)
  .filterDate(’2013-01-01’, ’2022-12-31’)
  .filter(ee.Filter.lt(’CLOUD_COVER’, 1))
  .median()
  .clip(region)
  .toFloat();

// Visualization settings
var srtmVis = {min: 0, max: 3000, palette: [’#440154’, ’#3b528b’, ’#21908d’, ’#5ec962’, ’#fde725’]};
var landsatVis = {bands: [’B4’, ’B3’, ’B2’], min: 0, max: 3000, gamma: 1.4, palette: [’#8c510a’, ’#bf812d’, ’#dfc27d’, ’#f6e8c3’, ’#c7eae5’, ’#80cdc1’, ’#35978f’, ’#01665e’]};

// Add SRTM and Landsat 8 imagery to the map
Map.centerObject(region, 10);
Map.addLayer(srtm, srtmVis, ’SRTM DEM’);
Map.addLayer(landsat, landsatVis, ’Landsat 8’);

// Export SRTM DEM data to Google Drive
Export.image.toDrive({
  image: srtm,
  description: ’srtm_dem_region8’,
  folder: ’output_gee_data’,
  scale: 30,
  region: region,
  fileFormat: ’GeoTIFF’
});

// Export Landsat 8 imagery to Google Drive
Export.image.toDrive({
  image: landsat,
  description: ’landsat8_image_region8’,
  folder: ’output_gee_data’,
  scale: 30,
  region: region,
```

```
  fileFormat: 'GeoTIFF'
});

// Export AOI vector shapefile to Google Drive
Export.table.toDrive({
  collection: ee.FeatureCollection(ee.Feature(region)),
  description: 'aoi_shapefile_region8',
  folder: 'output_gee_data',
  fileFormat: 'SHP'
});
```

Listing 1: Landsat Dataset Download Code

Figures

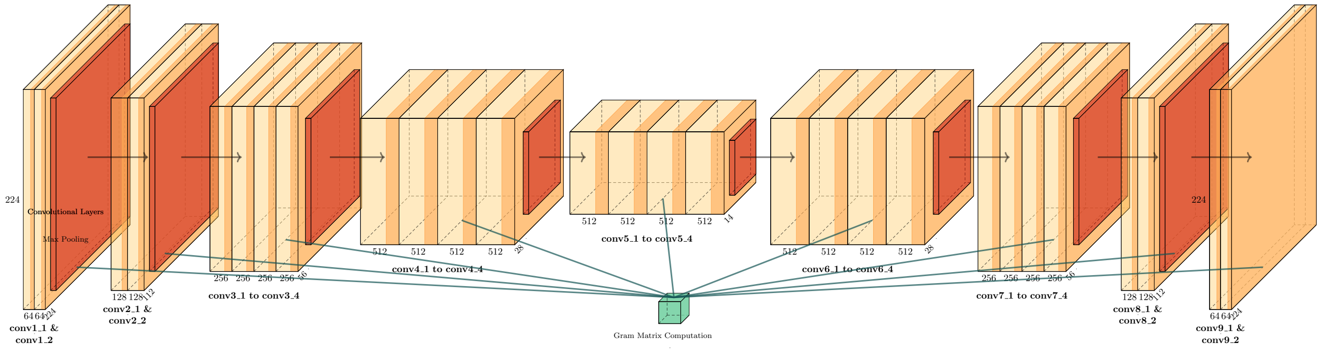


Figure A1: VGG-Adapter Module Architecture. The VGG-Adapter module is designed to reduce distribution shifts and improve consistency in remote sensing image reconstruction. Built upon the VGG-19 network, it extracts multi-scale feature representations from convolutional layers ($conv1_1, conv2_1, conv3_1, conv4_1, conv5_1$) and aligns them with the reference image. The module incorporates distribution loss to minimize feature discrepancies, ensuring improved stability and perceptual quality in generated images.

Tables

Table A1: Data Information-1. Notes: cloud cover $\leq 10\%$. The regions were randomly selected, but it was required to ensure that there were available Landsat images within these areas.

No.	Country / Region	Coordinates (Center Point)	Latitude Range	Longitude Range
01	USA	(-115.89, 40.90)	40.65 – 41.15	-116.14 – -115.64
02	Canada	(-113.67, 54.29)	54.04 – 54.54	-113.92 – -113.42
03	Brazil	(-49.08, -15.61)	-15.86 – -15.36	-49.33 – -48.83
04	Argentina	(-66.22, -33.09)	-33.34 – -32.84	-66.47 – -65.97
05	Peru	(-74.82, -12.21)	-12.46 – -11.96	-75.07 – -74.57
06	UK	(-2.45, 53.81)	53.56 – 54.06	-2.70 – -2.20
07	France	(1.78, 46.72)	46.47 – 46.97	1.53 – 2.03
08	Germany	(10.57, 50.94)	50.69 – 51.19	10.32 – 10.82
09	Russia	(45.91, 53.30)	53.05 – 53.55	45.66 – 46.16
10	India	(79.94, 23.94)	23.69 – 24.19	79.69 – 80.19

Table A2: Data Information-2. Notes: cloud cover $\leq 10\%$. The regions were randomly selected, but it was required to ensure that there were available Landsat images within these areas.

No.	**Country / Region**	**Coordinates (Center Point)**	**Latitude Range**	**Longitude Range**
11	China	(111.37, 30.76)	30.51 – 31.01	111.12 – 111.62
12	Australia	(147.45, -34.42)	-34.67 – -34.17	147.20 – 147.70
13	South Africa	(24.99, -29.55)	-29.80 – -29.30	24.74 – 25.24
14	Kenya	(36.81, -0.97)	-1.22 – -0.72	36.56 – 37.06
15	Egypt	(30.70, 26.40)	26.15 – 26.65	30.45 – 30.95
16	Iran	(53.29, 32.02)	31.77 – 32.27	53.04 – 53.54
17	Kazakhstan	(72.71, 46.12)	45.87 – 46.37	72.46 – 72.96
18	Indonesia	(106.93, -6.17)	-6.42 – -5.92	106.68 – 107.18
19	Japan	(137.71, 35.65)	35.40 – 35.90	137.46 – 137.96
20	New Zealand	(174.77, -39.38)	-39.63 – -39.13	174.52 – 175.02
21	Asia	(148.8, 9.24)	8.99 – 9.49	148.55 – 149.05
22	Asia	(93.36, 23.71)	23.46 – 23.96	93.11 – 93.61
23	Asia	(145.25, 45.64)	45.39 – 45.89	145.0 – 145.5
24	Asia	(127.8, 54.3)	54.05 – 54.55	127.55 – 128.05
25	Asia	(67.52, 23.81)	23.56 – 24.06	67.27 – 67.77
26	Asia	(110.26, 43.86)	43.61 – 44.11	110.01 – 110.51
27	Asia	(141.57, 26.21)	25.96 – 26.46	141.32 – 141.82
28	Asia	(104.34, 10.56)	10.31 – 10.81	104.09 – 104.59
29	Asia	(102.18, 5.57)	5.32 – 5.82	101.93 – 102.43
30	Asia	(70.69, 7.82)	7.57 – 8.07	70.44 – 70.94
31	Asia	(118.43, 10.88)	10.63 – 11.13	118.18 – 118.68
32	Asia	(112.5, 42.3)	42.05 – 42.55	112.25 – 112.75
33	Asia	(93.74, 53.11)	52.86 – 53.36	93.49 – 93.99
34	Asia	(138.17, 19.29)	19.04 – 19.54	137.92 – 138.42
35	Asia	(146.69, 16.18)	15.93 – 16.43	146.44 – 146.94
36	Asia	(147.29, 5.61)	5.36 – 5.86	147.04 – 147.54
37	Asia	(140.2, 7.16)	6.91 – 7.41	139.95 – 140.45
38	Asia	(149.37, 31.39)	31.14 – 31.64	149.12 – 149.62
39	Asia	(109.85, 8.69)	8.44 – 8.94	109.6 – 110.1
40	Asia	(107.08, 53.47)	53.22 – 53.72	106.83 – 107.33
41	Africa	(1.95, 29.77)	29.52 – 30.02	1.70 – 2.20
42	Africa	(12.55, 17.46)	17.21 – 17.71	12.30 – 12.80
43	Africa	(10.37, 16.08)	15.83 – 16.33	10.12 – 10.62
44	Africa	(20.28, 6.63)	6.38 – 6.88	20.03 – 20.53
45	Africa	(-10.26, 13.84)	13.59 – 14.09	-10.51 – -10.01
46	Africa	(-1.35, 34.61)	34.36 – 34.86	-1.60 – -1.10
47	Africa	(17.07, 11.13)	10.88 – 11.38	16.82 – 17.32
48	Africa	(36.49, 4.21)	3.96 – 4.46	36.24 – 36.74
49	Africa	(34.28, -1.02)	-1.27 – -0.77	34.03 – 34.53
50	Africa	(22.84, -6.15)	-6.40 – -5.90	22.59 – 23.09

Table A3: Data Information-3. Notes: cloud cover $\leq 10\%$. The regions were randomly selected, but it was required to ensure that there were available Landsat images within these areas.

No.	Country / Region	Coordinates (Center Point)	Latitude Range	Longitude Range
51	Africa	(31.76, -17.72)	-17.97 – -17.47	31.51 – 32.01
52	Africa	(35.95, -6.83)	-7.08 – -6.58	35.70 – 36.20
53	Africa	(13.42, -10.48)	-10.73 – -10.23	13.17 – 13.67
54	Africa	(6.11, -5.77)	-6.02 – -5.52	5.86 – 6.36
55	Africa	(-8.65, 20.26)	20.01 – 20.51	-8.90 – -8.40
56	Africa	(27.82, -12.89)	-13.14 – -12.64	27.57 – 28.07
57	Africa	(14.36, 19.24)	18.99 – 19.49	14.11 – 14.61
58	Africa	(30.95, 1.92)	1.67 – 2.17	30.70 – 31.20
59	Africa	(37.47, 6.73)	6.48 – 6.98	37.22 – 37.72
60	Africa	(39.59, -4.41)	-4.66 – -4.16	39.34 – 39.84
61	Africa	(32.48, -1.18)	-1.43 – -0.93	32.23 – 32.73
62	Africa	(29.75, -3.04)	-3.29 – -2.79	29.50 – 30.00
63	Africa	(18.31, -33.99)	-34.24 – -33.74	18.06 – 18.56
64	Africa	(15.64, 38.56)	38.31 – 38.81	15.39 – 15.89
65	Africa	(24.91, -17.76)	-18.01 – -17.51	24.66 – 25.16
66	Africa	(36.38, 0.52)	0.27 – 0.77	36.13 – 36.63
67	Africa	(9.62, -13.65)	-13.90 – -13.40	9.37 – 9.87
68	Africa	(5.33, 7.10)	6.85 – 7.35	5.08 – 5.58
69	Africa	(-4.43, 15.26)	15.01 – 15.51	-4.68 – -4.18
70	Africa	(21.19, 39.36)	39.11 – 39.61	20.94 – 21.44
71	Africa	(0.49, 32.85)	32.60 – 33.10	0.24 – 0.74
72	Africa	(28.88, 3.19)	2.94 – 3.44	28.63 – 29.13
73	Africa	(19.91, -1.43)	-1.68 – -1.18	19.66 – 20.16
74	Africa	(34.78, 8.64)	8.39 – 8.89	34.53 – 35.03
75	Africa	(12.01, -0.36)	-0.61 – -0.11	11.76 – 12.26
76	Africa	(11.56, 6.21)	5.96 – 6.46	11.31 – 11.81
77	Africa	(23.92, 35.14)	34.89 – 35.39	23.67 – 24.17
78	Africa	(25.13, -4.83)	-5.08 – -4.58	24.88 – 25.38
79	Africa	(-15.38, 28.26)	28.01 – 28.51	-15.63 – -15.13
80	Africa	(20.72, -3.64)	-3.89 – -3.39	20.47 – 20.97
81	Australia	(115.86, -31.95)	-32.20 – -31.70	115.61 – 116.11
82	Australia	(138.60, -34.92)	-35.17 – -34.67	138.35 – 138.85
83	Australia	(149.13, -35.28)	-35.53 – -35.03	148.88 – 149.38
84	Australia	(145.75, -16.92)	-17.17 – -16.67	145.50 – 146.00
85	Australia	(130.84, -12.46)	-12.71 – -12.21	130.59 – 131.09
86	Australia	(152.97, -27.47)	-27.72 – -27.22	152.72 – 153.22
87	Australia	(135.86, -31.94)	-32.19 – -31.69	135.61 – 136.11
88	Australia	(141.45, -37.83)	-38.08 – -37.58	141.20 – 141.70
89	Australia	(115.13, -22.05)	-22.30 – -21.80	114.88 – 115.38
90	Australia	(122.20, -17.96)	-18.21 – -17.71	121.95 – 122.45
91	Antarctica	(11.80, -70.67)	-70.92 – -70.42	11.55 – 12.05
92	Antarctica	(77.57, -68.58)	-68.83 – -68.33	77.32 – 77.82
93	Antarctica	(164.02, -74.70)	-74.95 – -74.45	163.77 – 164.27
94	Antarctica	(-46.42, -67.61)	-67.86 – -67.36	-46.67 – -46.17
95	Antarctica	(-69.88, -75.25)	-75.50 – -75.00	-70.13 – -69.63
96	Antarctica	(-122.95, -79.30)	-79.55 – -79.05	-123.20 – -122.70
97	Antarctica	(20.21, -69.50)	-69.75 – -69.25	19.96 – 20.46
98	Antarctica	(-102.40, -77.36)	-77.61 – -77.11	-102.65 – -102.15
99	Antarctica	(96.84, -66.72)	-66.97 – -66.47	96.59 – 97.09
100	Antarctica	(130.24, -70.05)	-70.30 – -69.80	129.99 – 130.49