Spatio Temporal Diffusion Model for Satellite Imagery

Prathap Nagaraj Kashyap[a], Alireza Javanmardi[b], Pragati Jaiswal[a,b], Gerd Reis[b], Alain Pagani[b], and Didier Stricker[a,b]

[a]RPTU Kaiserslautern, Gottlieb-Daimler-Straße 67663, Kaiserslautern, Germany
[b]DFKI, Trippstadter Str. 122, 67663, Kaiserslautern, Germany

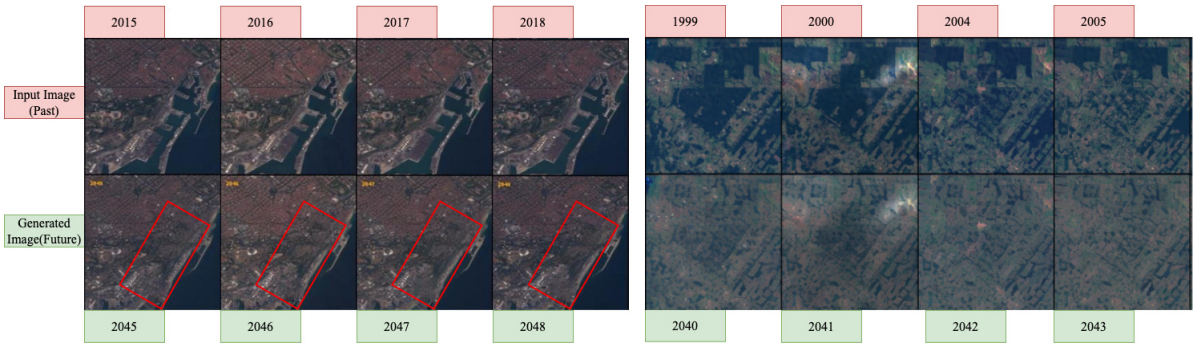

Figure 1. Given a sequence of input images and the desired future years, our model can generate a structurally consistent image sequence of a plausible future.

ABSTRACT

Generative AI has demonstrated strong capabilities in learning data distributions and producing realistic outputs. Although traditional approaches like Generative Adversarial Networks (GANs) have largely focused on static images and often fail to model coherent temporal sequences. Diffusion models have recently emerged as a more stable and effective alternative for generating time-sequenced data, yet their application to satellite imagery remains limited. Satellite data poses unique challenges such as multispectral channels and irregular temporal intervals that are poorly addressed by models trained on natural image datasets. To bridge this gap, we propose a spatio-temporal video diffusion model tailored for satellite-based forecasting tasks. Trained on curated datasets from the Landsat and Sentinel missions, our model generates temporally coherent sequences by conditioning on metadata like year, while effectively handling the spectral diversity and uneven sampling intervals characteristic of satellite imagery. When evaluated against retrained state-of-the-art baselines, our method demonstrates superior performance in modelling environmental changes, particularly deforestation, and achieves strong scores on perceptual quality metrics such as Fréchet Inception Distance (FID) and Learned Perceptual Image Patch Similarity (LPIPS), highlighting its effectiveness for geospatial generative tasks. Our code and data is available at github.com/dfki-av/STDS.

Keywords: Artificial Intelligence, Generative Model, Diffusion Model, Satellite Imagery, Spatio-Temporal Data

1. INTRODUCTION

Generative models have recently emerged as powerful tools capable of learning complex data distributions and producing realistic outputs across various domains. While Generative Adversarial Networks (GANs)[1] initially dominated progress in image synthesis, they suffer from limitations such as mode collapse and unstable training. Diffusion models have since gained prominence by offering stable training, greater diversity, and high-quality results through a denoising process, making them well-suited for tasks like remote sensing.

Corresponding author: Alireza Javanmardi, alierza.javanmardi@dfki.de

Eleventh International Conference on Remote Sensing and Geoinformation of the Environment (RSCy2025), edited by Andreas Christofi, et al., Proc. of SPIE Vol. 13816 · © The Authors. Published under a Creative Commons Attribution CC-BY 4.0 License · doi: 10.1117/12.3073127

This progress has extended to video generation, where models now capture both spatial and temporal dynamics, enabling the synthesis of coherent sequences that model scene evolution over time. Such capabilities are particularly valuable in geoinformation science, where satellite imagery is used to monitor environmental changes such as deforestation, urbanization, and natural disasters. However, generative modelling in this field remains underutilized due to challenges such as irregular time intervals, complex spatial structures, and limited labelled data. We address these challenges by introducing a novel framework for synthesizing the temporal evolution of satellite imagery. The goal is to generate realistic visualizations of how geographic regions change over time, enabling both historical analysis and future forecasting.

The proposed model is trained on sequences of time-stamped satellite images and learns to generate plausible future (or interpolated) frames based on past observations and specified target years, as shown in Fig. 1. This approach allows it to model the dynamics of landscape change, offering valuable applications in urban planning, deforestation tracking, and climate impact assessment. The model is trained on curated datasets representing real-world scenarios such as deforestation in the Amazon region and European urban growth, collected via Google Earth Engine (GEE)[2] and annotated with temporal metadata for conditioning. Furthermore, evaluation against state-of-the-art baselines shows that the model effectively learns to capture complex, often irregular, temporal progressions, generating coherent and realistic outputs.

The contributions of our work can be summarised as follows.

1. We introduce a novel model designed to synthesize realistic temporal progressions in satellite imagery, given a target year and a desired image sequence.
2. We present curated and processed datasets that capture dynamic geographic transformations, such as deforestation and urbanization over time.

2. RELATED WORKS

Our approach builds on recent advances in image-to-image translation, particularly Zero-Shot Image Translation[3] and Plug-and-Play Diffusion,[4] which use Denoising Diffusion Implicit Model (DDIM)[5] inversion to guide image editing while preserving structure. We extend this concept to the spatio-temporal domain by integrating DDIM inversion into a video diffusion framework, enabling the generation of structurally consistent and temporally coherent satellite image sequences for forecasting gradual landscape changes.

Unlike traditional video prediction models such as Masked Conditional Video Diffusion (MCVD) [6] and spatio-temporal Diffusion for Continuous Stochastic Video Prediction (STDiff),[7] which rely on motion cues like optical flow, our method addresses the unique challenges of satellite imagery, irregular sampling intervals, and long-term semantic shifts by conditioning generation on temporal metadata. This allows the model to produce context-aware and semantically meaningful predictions across time.

DiffusionSat[8] improves over motion-based models by conditioning on rich satellite metadata within a 3D ControlNet diffusion framework, enabling interpolation and forecasting. However, its reliance on numerous metadata inputs and one-shot generation limits scalability. Our model, by contrast, uses only temporal metadata and supports recursive generation of longer sequences, offering a more efficient and extensible alternative.

Spatio-Temporal Super Resolution for Satellite Imagery (STSR),[9] designed for super-resolution, synthesizes high-resolution outputs from low-resolution inputs using spatio-temporal cues. While it can extrapolate temporally, its focus remains on spatial fidelity at known timestamps. Our method is explicitly tailored for temporal progression, supporting diverse use cases like long-term forecasting and interpolated scene generation, making it better suited for dynamic Earth observation tasks.

3. METHOD

To model temporal progression in satellite imagery, we consider a sequence of past observations $\{I^{(t_1)}, \ldots, I^{(t_F)}\}$, where each frame $I^{(t)} \in \mathbb{R}^{C\times H\times W}$ represents a satellite image captured at time t. The goal is to synthesize future representations $\{\hat{I}^{(t')}\}$ for target time steps $t' > t_F$, making this a time-conditional image generation task. Leveraging a video diffusion framework, the input is structured as a tensor $\mathbb{R}^{F\times C\times H\times W}$, allowing the model to learn and reproduce temporally coherent changes grounded in historical geospatial patterns.

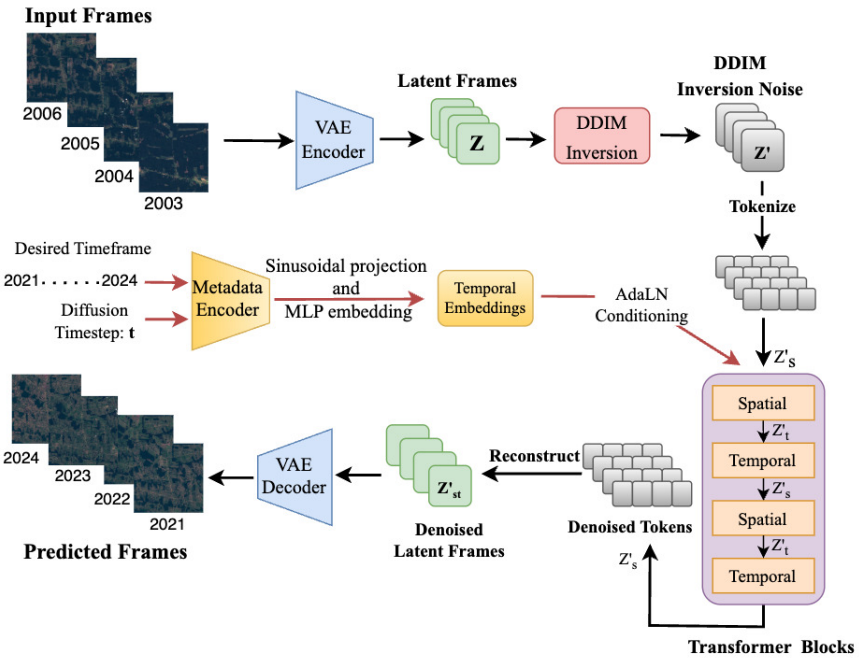

Figure 2. (A) The pipeline comprises three core stages: **Encoding**, **Conditioning**, and **Decoding**. (B) The illustrated architecture outlines the sampling process, which incorporates DDIM inversion. During training, this is replaced by the standard forward diffusion. (C) Z'_s, Z'_t, and Z'_{st} denote the spatial, temporal, and combined spatio-temporal token representations, respectively, following the denoising process.

3.1 Encoding

The architecture of our model, as seen in Section 3.1, is based on the Latte video generation framework, which builds on the Latent Diffusion Model (LDM) pipeline[10] by conducting the diffusion process in latent space for greater efficiency. As shown in Fig. 2, each input video frame is first encoded into a compressed latent representation using an encoder $\mathcal{E}$, allowing the model to learn spatial and temporal patterns in a reduced dimensionality space.

Let $V_L \in \mathbb{R}^{F\times C\times H\times W}$ denote the latent video representation, where F, C, H, and W represent the number of frames, channels, height and width respectively. This latent tensor is reshaped into a sequence of spatio-temporal tokens $\hat{z} \in \mathbb{R}^{n_f\times n_h\times n_w\times d}$, where n_f, n_h, and n_w are the number of tokens along temporal and spatial dimensions, and d is the token dimensionality.

To retain positional information, a spatio-temporal positional embedding p is added:

$$z = \hat{z} + p \tag{1}$$

The sequence z is reshaped into $z_s \in \mathbb{R}^{n_f\times t\times d}$, where $t = n_h \times n_w$, and fed into a spatial Transformer to capture local and global dependencies within each frame. The output is then reshaped to $z_t \in \mathbb{R}^{t\times n_f\times d}$ and processed through a temporal Transformer block, enabling the model to learn the dependencies among the frames.

3.2 Conditioning Data.

Each video frame in our dataset is associated with temporal metadata $\mathcal{T}$, such as year or month, which is continuous and numerical in nature. The aim is to model the conditional distribution $p(v \mid \mathcal{T})$, where v denotes the video. A naive strategy would be to embed $\mathcal{T}$ into short descriptive captions; however, this unnecessarily discretizes continuous variables and is constrained by the limitations of text encoders in representing numerical data accurately, as presented by Radford, A. et al.[11]

To overcome this, we adopt the numerical metadata conditioning technique introduced in Diffusion-Sat,[8] using sinusoidal timestep embeddings commonly employed in diffusion models. Specifically, the metadata is first normalized to the range $[0, 1000]$, aligning it with the diffusion timestep domain. It is then projected using the following sinusoidal functions:

$$\text{Project}(k, 2i) = \sin\left(k\,\Omega^{-\frac{2i}{d}}\right), \quad \text{Project}(k, 2i+1) = \cos\left(k\,\Omega^{-\frac{2i}{d}}\right) \tag{2}$$

where k is the normalized metadata value, i indexes the embedding dimension, d is the total dimensionality, and $\Omega = 10000$ is a scaling constant. The projected embedding is further processed by a Multi-Layer Perceptron (MLP), identical to the one used for encoding the diffusion timestep in Denoising Diffusion Probabilistic Model (DDPM),[12] enabling effective temporal conditioning within the model.

$$f_{\theta_f}(\mathcal{T}_f) = \text{MLP}\left([\text{Project}(\mathcal{T}_f)]\right) \tag{3}$$

where f_{θ_f} represents the learned MLP embedding for the metadata value $\mathcal{T}_f$ for frame f. The temporal metadata embedding m is then $m = f_\theta(\mathcal{T}_f) \in \mathbb{R}^D$, where D is the embedding dimension.

As with the reshaping of z into z_s and z_t for the spatial and temporal transformers, respectively, the temporal embeddings are reshaped in a similar manner. The temporal embedding vector is then added to the embedded timestep t as $t = f_\theta(t) \in \mathbb{R}^D$, so that the final conditioning vector c is:

$$c = m + t \tag{4}$$

3.3 Conditioning Method

We adopt Adaptive Layer Normalization with zero initialization (AdaLN-Zero) for conditioning the Transformer blocks, following the Diffusion Transformer (DiT)[13] framework. In standard Layer Normalization (LN), the activations are normalized and scaled using learned affine parameters γ and β:

$$\text{LN}(x) = \frac{\gamma(x - \mu)}{\sigma} + \beta \tag{5}$$

where μ and σ represent the mean and standard deviation of the input activations. In AdaLN, these parameters become conditional and are predicted from an external conditioning vector c. Specifically, a Multi-Layer Perceptron (MLP) maps the conditioning input to a pair of vectors:

$$\gamma_c, \beta_c = \text{MLP}_{\text{cond}}(c) \tag{6}$$

In a Transformer block, these AdaLN-normalized activations are used as inputs to operations such as Attention and MLP layers. The outputs are then combined with the original input through residual connections. In the unconditioned case, this would look like:

$$x = x + \text{AttentionOutput}, \quad x = x + \text{MLPOutput} \tag{7}$$

In AdaLN-Zero, these residual updates are further modulated by an additional learnable scaling factor α_c, also predicted from the conditioning vector c. The conditioned form becomes:

$$x = x + \alpha_c \cdot \text{Operation}(\text{AdaLN}(x \mid c)) \tag{8}$$

where *Operation* refers to either the Attention or MLP sub-layer.

The "zero" initialization ensures that α_c starts near zero, means that the conditioning has little effect at the start of training. This stabilizes optimization by allowing the model to first learn the underlying structure of the data before conditioning influences the outputs. As training progresses, the influence of c increases, allowing the model to adapt based on temporal or contextual metadata.

3.4 Decoding

After passing through the Transformer backbone, a crucial step involves decoding the video token sequence to generate both the predicted noise and the predicted covariance. The shapes of these two outputs match the dimensions of the input $\mathbf{V_L} \in \mathbb{R}^{F \times H \times W \times C}$. In line with prior research,[13] this is achieved by using a conventional linear decoder along with a reshaping operation.

3.5 Sampling

During sampling, the model follows a process similar to training, but employs DDIM inversion to ensure structural consistency, as it does not rely on image conditioning. Starting with the input frames v, an encoder $\mathcal{E}$ transforms them into a latent representation v_0. Instead of sampling from random noise, the model deterministically reconstructs the corresponding noise vector v_T through iterative DDIM inversion:

$$\text{predicted}_\epsilon = \epsilon_\theta(v_{t-1}, t-1) \tag{9}$$

$$v_t \approx \sqrt{\bar{\alpha}_t} \cdot v_0 + \sqrt{1-\bar{\alpha}_t} \cdot \text{predicted}_\epsilon \tag{10}$$

Here, $\bar{\alpha}_t$ denotes the cumulative product of the noise schedule coefficients up to timestep t, indicating the retained signal proportion after t steps in the forward diffusion process. This reconstruction continues until timestep T, resulting in the latent noise v_T. Instead of initializing generation with random noise, the model uses this inverted noise along with a sequence of target years $\mathcal{T}$ as conditioning input. For example, if the input frames span 2015–2020, the model is conditioned on years 2021–2026 to guide generation. From this point, the model leverages the pre-trained spatial and temporal transformer blocks (as described in Section 3.1) to synthesize future outputs aligned with the given temporal context, as illustrated in Fig. 1.

4. EXPERIMENTS

4.1 Dataset

This study utilizes satellite imagery from the Landsat and Sentinel series, combining Landsat's extensive historical archive (since 1984) with Sentinel-2's higher-resolution data. Both provide multispectral information beyond RGB, including bands like Near Infrared (NIR) and Short Wave Infrared (SWIR), critical for environmental and urban analyses.

Two primary datasets were curated: one capturing rapid deforestation in the Amazon (1997–2024) using Landsat data and spectral indices like NDVI, and another focused on more gradual urban expansion across European cities (2015–2025) with Sentinel-2 imagery. The deforestation dataset highlights visually distinct vegetation loss, while urban change is subtler, characterized by structural developments such as road expansions and new buildings, influenced by complex socioeconomic factors.

Training on the urban dataset, the model learns broad spatial distributions and development patterns distinct from the organic, irregular changes of deforestation. While effective at capturing overall urban layouts, it shows limitations in reconstructing finer geometric details like roads and coastlines, indicating areas for future refinement to handle complex urban structures more accurately.

4.2 Model Evaluation and Comparative Analysis

A key requirement of our model is to produce outputs that maintain structural consistency with input frames while exhibiting plausible temporal progression. Structural similarity is measured using the Structural Similarity Index (SSIM), and temporal consistency is evaluated via Fréchet Video Distance (FVD), which assesses coherence across video sequences.

We benchmark our model on the Amazon deforestation dataset due to its clear annual changes in forest cover, which offer both visual and quantitative evaluation. A temporal holdout strategy is employed by excluding specific years (e.g., 2005, 2015, 2022–2024) from training, allowing us to test the model's capacity to generate these held-out years from earlier observations.

Our model is compared against three baselines in spatio-temporal image generation: DiffusionSat,[8] STSR,[9] and STDiff.[7] While DiffusionSat and STSR are tailored for satellite imagery, STDiff is a general-purpose video prediction model without satellite-specific metadata conditioning. Comparisons are limited to scenarios where the target year follows the input sequence, as some baselines are not designed for arbitrary temporal prediction.

All baselines were retrained on the deforestation dataset per our evaluation strategy. Missing metadata required by models like DiffusionSat, such as geo-coordinates, cloud cover, and Ground Sampling Distance (GSD), were approximated using empirical values to ensure compatibility.

As shown in Fig. 3, among the baselines, DiffusionSat aligns most closely with our objective, generating future frames from three inputs and metadata. STSR was adapted from super-resolution to temporal interpolation but suffers in fidelity and long-term forecasting. STDiff, dependent on motion cues, often replicates input frames due to the static nature of satellite imagery.

Each baseline presents limitations: DiffusionSat's heavy metadata reliance is a practical barrier, STSR offers efficient inference but lacks detail and accuracy, and STDiff struggles to generalize without explicit motion. In contrast, our model requires only a short image sequence and target year, free of metadata, while outperforming others in capturing meaningful spatio-temporal changes, thus providing a more scalable and effective solution for temporal satellite image generation. As reported in Table 1, our method achieves a substantially lower FID of 55.5827 and outperforms the baselines on LPIPS, L1, and PSNR, underscoring its capacity to produce temporally consistent satellite images.

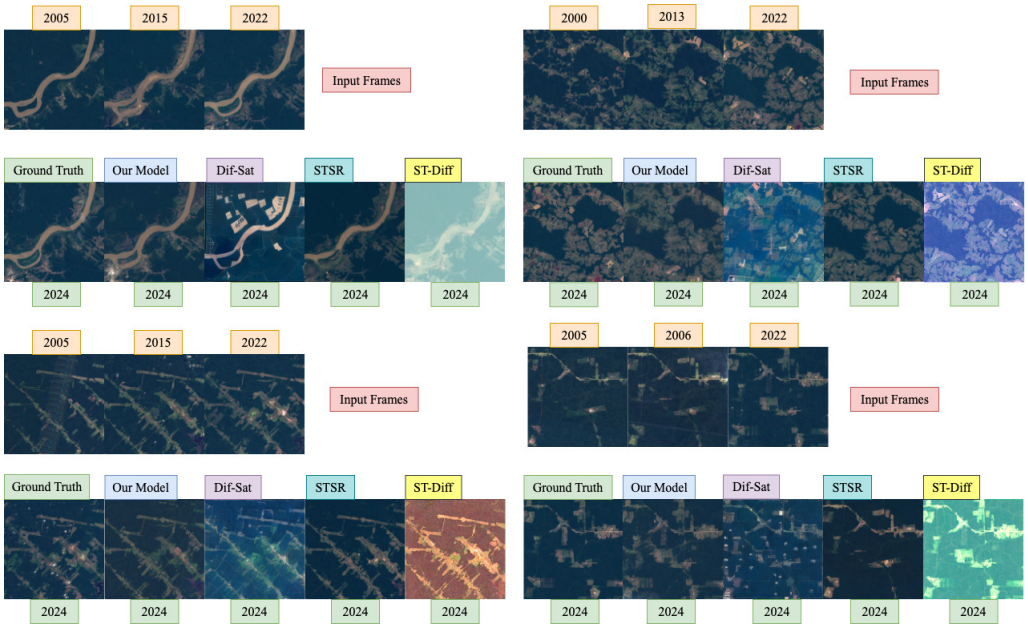

Figure 3. Comparison of results generated by our model and baseline models. While several baselines tend to replicate input frames with minimal variation, indicating limited temporal understanding, our model captures clear temporal progression with meaningful changes aligned to expected deforestation trends.

Table 1. Performance comparison of our model against existing methods. Arrows indicate whether higher (↑) or lower (↓) values are preferred for each metric.

S.No	Method	FID ↓	L1 ↓	SSIM ↑	LPIPS ↓	PSNR ↑
1	Dif-Sat[8]	154.2632	2.96×10^{-4}	0.4042	0.5483	19.6337
2	STSR[9]	82.80	2.8×10^{-4}	**0.5968**	0.3971	21.184
3	STDiff[7]	117.5076	1.31×10^{-3}	0.4200	0.5693	8.6653
4	Our Model	**55.5827**	$\mathbf{2.0 \times 10^{-4}}$	0.5571	**0.3722**	**23.1605**

4.3 Ablation Studies

To evaluate the impact of varying input configurations and conditioning strategies, we conducted a series of ablation studies.

Single-Frame Input

We first tested whether the model could learn temporal progression from a single frame by reformulating the task as image diffusion with $F = 1$ in the input tensor $V_L \in \mathbb{R}^{F\times C\times H\times W}$. Each image was conditioned on its corresponding year Y during training. At inference, the same frame was conditioned on a future year $Y' > Y$. However, the outputs often resembled the input image with minimal variation, an outcome attributed to DDIM inversion, indicating that single-frame conditioning is insufficient for modelling meaningful temporal change. A few examples did exhibit minor signs of adaptation (Fig. 4, left).

Two-Frame Input

Expanding the input to two sequential frames was intended to provide minimal temporal context. Similar to the single-frame setup, the model primarily reconstructed the input images without exhibiting strong evidence of learning or extrapolating future transitions (Fig. 4, right).

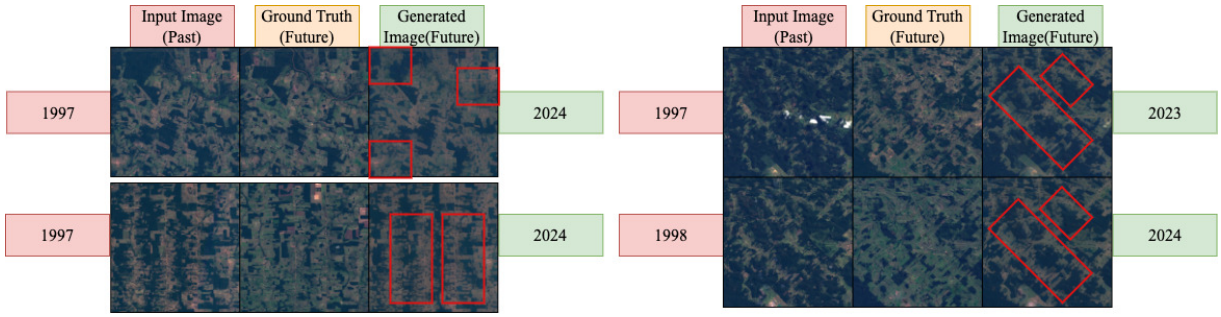

Figure 4. (Left)Comparison of the results vs ground-truth from a 1-frame model. The top and the bottom sequences represent two separate examples. (Right) Comparison of results vs ground truth from a 2-frame model. Both models exhibit limited temporal learning. The images in the top and the bottom row exhibit the temporal evolution of the same region.

Four-Frame Input

Introducing four frames with temporal conditioning provided a richer temporal context. The model showed improved ability to learn temporal progression, generating samples that align well with future ground-truth frames while preserving structural consistency with the input sequence (Fig. 5, left).

Cross-Attention Conditioning

We trained a variant using standard cross-attention instead of AdaLN. Each frame was conditioned on its timestamp through cross-attention layers. Although the results maintained high visual fidelity, they exhibited minimal temporal evolution, closely resembling the inputs regardless of target year (Fig. 5, right). This suggests that AdaLN conditioning is more effective in learning temporal dynamics.

Table 2. Image evaluation of different frame-based methods. (i) CA = Cross Attention, (ii) Y = Year conditioning, (iii) Y+SD* = Year + Structural Difference.

S.No.	Method	FID ↓	L1 ↓	SSIM ↑	LPIPS ↓	PSNR ↑
1	1-Frame	66.66	2.2×10^{-4}	0.5203	0.4179	22.25
2	2-Frames	66.64	2.2×10^{-4}	0.5206	0.4177	22.25
3	4-Frames	58.31	1.9×10^{-4}	0.5511	0.3884	23.18
4	4-Frames (CA*)	56.26	$\mathbf{1.86 \times 10^{-4}}$	**0.5636**	0.3767	23.46
5	6-Frames (Y*)	**48.38**	2.1×10^{-4}	0.5510	0.3738	**22.72**
6	6-Frames (Y+SD*)	55.58	2.0×10^{-4}	0.5571	**0.3722**	23.16

Table 3. Video evaluation metrics of different frame-based models. (i) CA = Cross Attention, (ii) Y = Year conditioning, (iii) Y+SD* = Year + Structural Difference.

S.No.	Method	FID-VID (FVD-3DRN50) ↓	FVD (FVD-3DInception) ↓
1	2-Frames	21.24	216.32
2	4-Frames	11.09	168.15
3	4-Frames (CA*)	14.43	217.26
4	6-Frames (Y*)	**10.42**	**159.37**
5	6-Frames (Y+SD*)	10.69	176.64

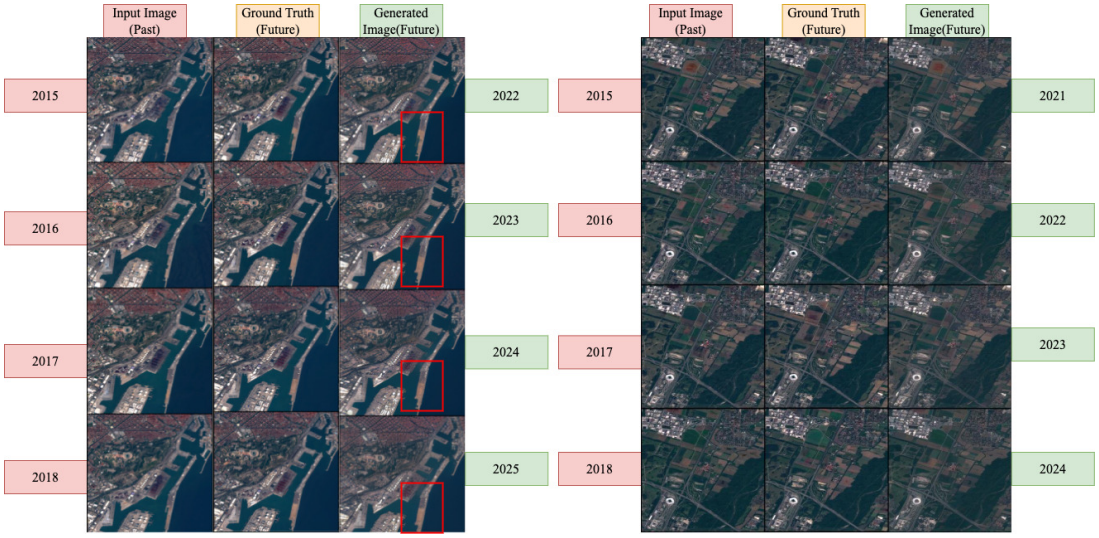

Figure 5. (Left)Comparison of ground-truth vs actual results from a 4-frame model. (Right) Comparison of ground-truth vs actual results from a 4-frame cross-attention conditioned model.

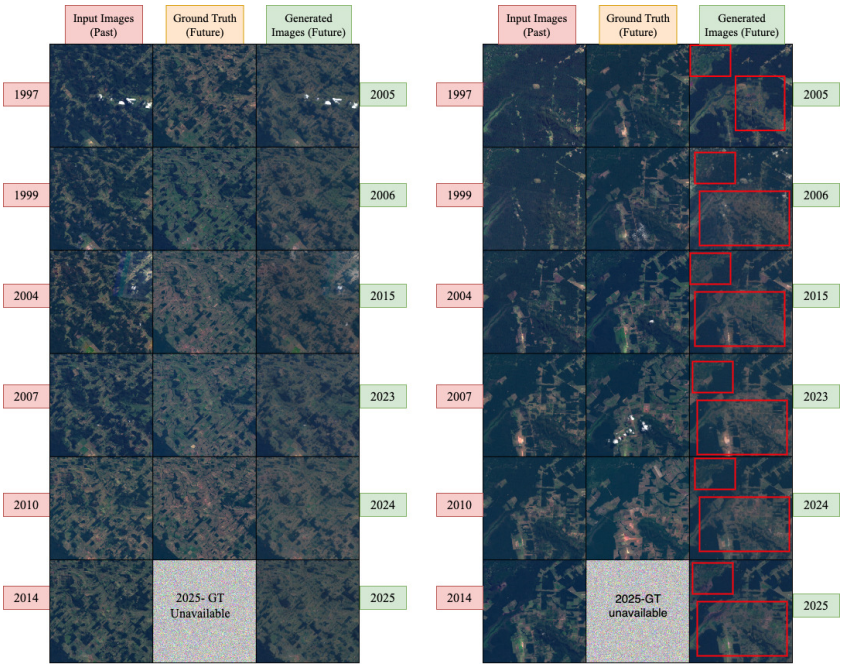

Figure 6. (Left)Comparison of ground-truth vs actual results from a 6-frame model conditioned on temporal metadata alone (years). (Right) Comparison of ground-truth vs actual results from a 6-frame conditioned on years and structural differences based model.

Six-Frame Input

Extending the input to six frames yielded noticeable improvements in both structural accuracy and temporal coherence. The model captured realistic progression patterns while preserving spatial fidelity across frames (left, Fig. 6).

Structural Difference Conditioning:

To further enhance structural alignment, we introduced a variant incorporating structural difference information defined as $(1 - \text{SSIM})$, normalized between 0 and 1000 (see Section 3.2). The overall conditioning vector is defined as:

$$c = m + t + s \tag{11}$$

where m, t, and s represent the timestep, temporal, and structural embeddings, respectively. During inference, since future structural differences are unknown, we approximated this component by averaging differences across input frames. This variant showed slight improvements over the 6-frame baseline in terms of structural consistency with future ground-truth samples (Fig. 6, right). Quantitative evaluations for all setups are presented in Table 2 and Table 3.

5. CONCLUSION

We propose a Spatio-Temporal Diffusion model for satellite imagery, built on the Latte framework, which generates temporally coherent image sequences by learning from multiple input frames conditioned on temporal metadata. A DDIM inversion-based sampling strategy helps preserve spatial structure while enabling realistic future predictions. The model performs best when conditioned on temporally closer frames, making it particularly effective for filling gaps caused by missing or corrupted data (e.g., cloud cover). Although not pixel-perfect, the outputs are plausible and valuable for applications like deforestation tracking, urban growth monitoring, and climate impact assessment. Future work could refine urban-scale details with higher-resolution priors and shorter-context training, and incorporate drivers such as temperature, precipitation, and socio-economic data to improve interpretability and extend the framework to domains like medical imaging and precision agriculture.

6. ACKNOWLEDGEMENT

This research has been partially funded by the EU project AI-Observer (EU: 101079468).

REFERENCES

[1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., "Generative adversarial networks," *Communications of the ACM* **63**(11), 139–144 (2020).

[2] Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., and Moore, R., "Google earth engine: Planetary-scale geospatial analysis for everyone," *Remote Sensing of Environment* (2017).

[3] Parmar, G., Singh, K. K., Zhang, R., Li, Y., Lu, J., and Zhu, J., "Zero-shot image-to-image translation," in [*ACM SIGGRAPH 2023 Conference Proceedings*], 1–11 (2023).

[4] Tumanyan, N., Geyer, M., Bagon, S., and Dekel, T., "Plug-and-play diffusion features for text-driven image-to-image translation," in [*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*], 1921–1930 (2023).

[5] Song, J., Meng, C., and Ermon, S., "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502* (2020).

[6] Voleti, V., Jolicoeur-Martineau, A., and Pal, C., "Mcvd: Masked conditional video diffusion for prediction, generation, and interpolation," *Advances in Neural Information Processing Systems* **35**, 23371–23385 (2022).

[7] Ye, X. and Bilodeau, G., "Stdiff: Spatio-temporal diffusion for continuous stochastic video prediction," in [*Proceedings of the AAAI Conference on Artificial Intelligence*], **38**(7), 6666–6674 (2024).

[8] Khanna, S., Liu, P., Zhou, L., Meng, C., Rombach, R., Burke, M., Lobell, D., and Ermon, S., "Diffusionsat: A generative foundation model for satellite imagery," *arXiv preprint arXiv:2312.03606* (2023).

[9] He, Y., Wang, D., Lai, N., Zhang, W., Meng, C., Burke, M., Lobell, D., and Ermon, S., "Spatial-temporal super-resolution of satellite imagery via conditional pixel synthesis," *Advances in Neural Information Processing Systems* **34**, 27903–27915 (2021).

[10] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B., "High-resolution image synthesis with latent diffusion models," in [*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*], 10684–10695 (2022).

[11] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al., "Learning transferable visual models from natural language supervision," in [*Proceedings of the 38th International Conference on Machine Learning*], 8748–8763, PMLR (2021).

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

[13] Peebles, W. and Xie, S., "Scalable diffusion models with transformers," in [*Proceedings of the IEEE/CVF International Conference on Computer Vision*], 4195–4205 (2023).