OSMGen: Highly Controllable Satellite Image Synthesis using OpenStreetMap Data

Amir Ziashahabi* **Narges Ghasemi*** **Sajjad Shahabi**
John Krumm **Salman Avestimehr** **Cyrus Shahabi**
University of Southern California
{ziashaha, nghasemi, sajjadsh, jkrumm, salman, shahabi}@usc.edu

Abstract

Accurate and up-to-date geospatial data are essential for urban planning, infrastructure monitoring, and environmental management. Yet, automating urban monitoring remains difficult because curated datasets of specific urban features and their changes are scarce. We introduce OSMGen, a generative framework that creates realistic satellite imagery directly from raw OpenStreetMap (OSM) data. Unlike prior work that relies on raster tiles, OSMGen uses the full richness of OSM JSON, including vector geometries, semantic tags, location, and time, giving fine-grained control over how scenes are generated. A central feature of the framework is the ability to produce consistent before–after image pairs: user edits to OSM inputs translate into targeted visual changes, while the rest of the scene is preserved. This makes it possible to generate training data that addresses scarcity and class imbalance, and to give planners a simple way to preview proposed interventions by editing map data. More broadly, OSMGen produces paired (JSON, image) data for both static and changed states, paving the way toward a closed-loop system where satellite imagery can automatically drive structured OSM updates. Source code is available at `https://github.com/amir-zsh/OSMGen`.

1 Introduction

Urban planning can greatly benefit from accurate and timely geospatial data [26, 18]. A readily available source is OpenStreetMap (OSM) [23], a collaborative project that offers more than just a visual map: it provides a detailed, structured JSON format containing rich information lost in simple image tiles, such as precise vector geometries and semantic tags for every feature. While this detailed data structure is ideal for conditioning a generative process, it has been largely underexplored for this purpose, partly due to the complexity of the JSON files.

In this work, we leverage the full depth of OSM data for image generation. The core of our contribution is a novel generative model that synthesizes high-fidelity satellite imagery by conditioning on the structured information within OSM JSON. In contrast to methods that use rendered map images, our approach utilizes a richer set of inputs including feature tags, location, and date, enabling highly controllable and precise synthesis. We also introduce a method to leverage this model for controlled scene manipulation: by editing the OSM-derived inputs, we can synthesize a corresponding "after" image that is perfectly co-registered to its "before" state, thereby isolating the visual impact of a single, defined change.

Our approach enables two powerful applications. First, it can generate vast, pixel-perfect labeled datasets to address data scarcity in geospatial AI, improving downstream models for tasks like building footprint segmentation [20] and land-use classification [40]. Second, it serves as a dynamic

*These authors contributed equally.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: UrbanAI 2025.

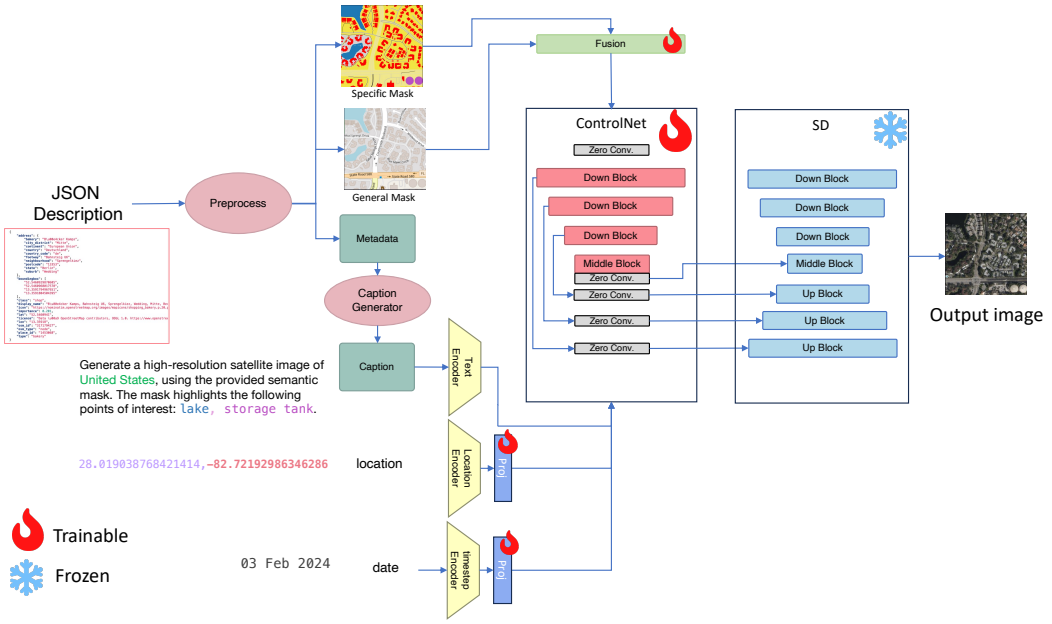

Figure 1: Overview of our ControlNet pipeline. Semantic masks are fed into ControlNet to generate control feature maps that are added into the U-Net; spatial and temporal embeddings are summed into the timestep embedding; the text prompt is injected via cross-attention.

simulation tool, allowing urban planners to visualize the impact of proposed developments, such as new parks or infrastructure, by simply editing the JSON map data, thus supporting data-driven decision-making.

Crucially, our framework is unique in its ability to generate complete, corresponding pairs of (JSON, image) data for both before and after states. This provides the exact data required to train the next generation of cartographic models that can truly "close the loop": detecting changes in new satellite imagery to suggest automated, structured updates to OSM JSON. This capability promises to significantly reduce the manual effort needed to keep the world's map current and accurate [3].

Background. Denoising diffusion models synthesize images by learning to reverse a fixed process of gradually adding noise [11, 29]. By training a neural network to perform this reverse denoising operation, the model can generate new, high-fidelity images starting from pure noise. To enable precise, training-free image editing, we leverage Denoising Diffusion Implicit Models (DDIM) [30], which introduce a deterministic variant of the diffusion process. This determinism is crucial because it allows the process to be inverted. Given a real image, DDIM inversion can trace the denoising path backward to find the unique latent code that generates it. This inversion capability is the key mechanism that enables high-fidelity, targeted modifications. The technical details of these processes are detailed further in Appendix A.

2 Methodology

This work presents an end-to-end pipeline for generating satellite imagery conditioned on rich information derived from raw OSM JSON data. Our approach addresses key shortcomings of methods that rely on simpler inputs such as raster tiles or bounding boxes, which lack the precise geometries and detailed tag-level semantics available in the source OSM data [33, 19]; see Appendix C.1 for a detailed analysis. By leveraging the source JSON, we enable fine-grained, controllable, and spatially accurate synthesis. Please refer to Appendix D for a detailed literature review on image generation, satellite image generation, and image editing.

Data Collection and Preprocessing. To ensure broad geographic coverage, we sample approximately 20,000 points from the Functional Map of the World (FMoW) benchmark [5], spanning urban centers, suburbs, and rural areas. For each point, we fix a zoom level z in advance (typically choosing $z = 18$ to capture fine-grained structural details and $z = 15$ for wider contextual views) and compute the exact 256×256-pixel tile bounds around the center latitude and longitude via standard Web Mercator tile formulas [1]. Using this bounding box and zoom, we retrieve a 256×256 satellite image tile and its corresponding raw OSM JSON for each data point. From the JSON, we extract a multimodal set of conditions designed to provide comprehensive guidance to the generative model. The primary conditions are two segmentation masks derived from the raw vector geometries: (1) the **general mask**, which groups features into a small number of broad categories such as roads, water bodies, vegetation, buildings, and other primary surface types, capturing high-level concepts; and (2) the **specific mask**, which assigns each fine-grained point-of-interest (POI) subtype (e.g., lakes, rivers, storage tanks, solar farms) its own mask color so the model can learn the nuances of each type. To capture spatiotemporal context, we encode the tile's geographic coordinates using SatCLIP [17] and its capture date using Date2Vec [28]. Finally, we generate a textual summary of the tile's most salient categories and encode it via a frozen CLIP text encoder [25] to provide high-level semantic guidance (see Appendix B.1 for details).

Generation Framework. As illustrated in Figure 1, our framework augments a frozen Stable Diffusion U-Net [27] with a trainable ControlNet branch [34]. The general and specific masks are fused via a convolutional layer and provided to ControlNet to enforce geometric fidelity. Spatial and temporal embeddings are each passed through a linear projection and then added to the diffusion timestep embedding, while the text embedding is injected through cross-attention. We train the ControlNet component, the mask-fusion layer, and the linear projections for spatial and temporal conditioning, using the standard diffusion loss:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{x_0,\, t,\, \epsilon} \Big\| \epsilon \; - \; \epsilon_\theta\big(x_t,\, t \mid M, \mathbf{e}_{\text{loc}}, \mathbf{e}_{\text{time}}, \mathbf{e}_{\text{text}}\big) \Big\|_2^2 .$$

Here, the network ϵ_θ is trained to predict the ground-truth noise ϵ from the noisy image latent x_t at timestep t, given the set of conditions: the fused mask M and the location, time, and text embeddings $(\mathbf{e}_{\text{loc}}, \mathbf{e}_{\text{time}}, \mathbf{e}_{\text{text}})$.

Controlled Change Generation. To create consistent before/after image pairs, we use DDIM inversion [30]. This choice is driven by three factors. First, cross-attention–based editing methods are unsuitable here because they do not account for our nontextual conditions (masks, spatial, and temporal information) [10]. Second, DDIM inversion is straightforward to implement and agnostic to the model's specific architecture. Third, strong spatial conditioning from the masks allows us to reduce the classifier-free guidance (CFG) scale, mitigating a known limitation of DDIM inversion, instability at high CFG scales [22, 12], and yielding high-fidelity results. Please refer to Appendix A.2 for details.

3 Experiments

Experimental Setup We trained the model for 500 epochs using a batch size of 2048. Our evaluation uses samples from a held-out test set of approximately 2,000 locations from our FMoW-derived dataset. For each location, we generate a 256x256 pixel tile using the full multi-modal conditioning framework described previously. All synthesis operations were performed on a single NVIDIA A100 GPU.

Qualitative Results. Figure 2 shows representative outputs, demonstrating that the model (i) accurately reproduces road networks and building footprints from the general mask and (ii) renders rare POI classes (e.g., stadiums, storage tanks) with correct shapes and context from the specific mask. A brief analysis of seasonality under temporal conditioning is deferred to Appendix E. **Consistency via DDIM Inversion:** We apply DDIM inversion and re-denoising with an edited mask to generate "after" images in which regions outside the edited area remain consistent with the "before" state. Figure 4 demonstrates samples produced using this method for edits that add, remove, or modify elements. The pipeline produces consistent pairs without introducing artifacts outside the intended changes.

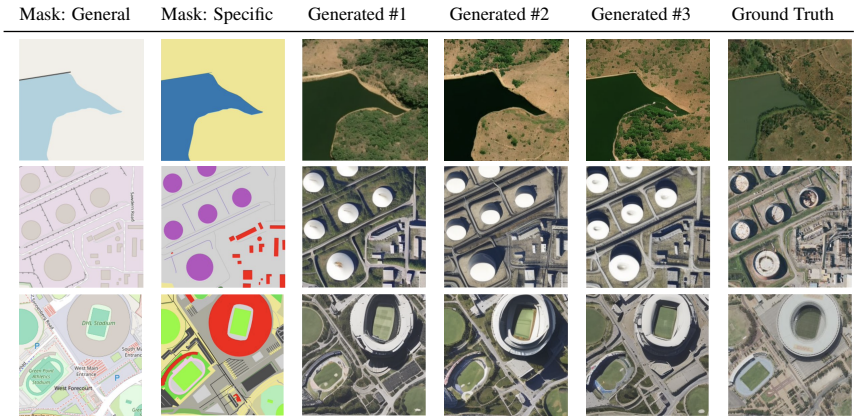

Figure 2: Qualitative evaluation on held-out FMoW locations. This layout highlights both the model's ability to reproduce large-scale structure and to capture fine-grained POI details in context.

Figure 3: Editing via DDIM inversion. Edits are applied locally while preserving consistency outside the modified region.

4 Conclusion

We have presented a novel approach for high-fidelity satellite image synthesis conditioned on OSM JSON. This framework achieves both structural fidelity and semantic richness, opening new avenues for interactive geospatial content creation and data augmentation. This framework paves the way for generating large labeled datasets, including static imagery and co-registered before/after pairs, to be used in downstream tasks, such as segmentation, change detection, and automated proposals for OSM JSON updates based on satelite image changes.

5 Acknowledgments

This research has been funded in part by NSF grants IIS-2128661, 1956435, and DMS-2428039, and NIH grant 5R01LM014026. Opinions, findings, conclusions, or recommendations expressed are those of the author(s) and do not necessarily reflect the views of any sponsors, such as NSF or NIH.

References

[1] Slippy map tilenames. OpenStreetMap Wiki, 2025. Accessed May 9, 2025.

[2] Stability AI. Stable diffusion v2. `https://stability.ai/blog/stable-diffusion-v2-release`, 2022.

[3] Favyen Bastani, Songtao He, Satvat Jagwani, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, Sam Madden, and Mohammad Amin Sadeghi. Updating street maps using changes detected in satellite imagery. In *Proceedings of the 29th International Conference on Advances in Geographic Information Systems*, pages 53–56, 2021.

[4] Jiaxin Cheng, Xiao Liang, Xingjian Shi, Tong He, Tianjun Xiao, and Mu Li. Layoutdiffuse: Adapting foundational diffusion models for layout-to-image generation. *arXiv preprint arXiv:2302.08908*, 2023.

[5] 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.

[6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[7] Miguel Espinosa and Elliot J Crowley. Generate your own scotland: Satellite image generation conditioned on maps. *arXiv preprint arXiv:2308.16648*, 2023.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27, 2014.

[9] Sen He, Wentong Liao, Michael Ying Yang, Yongxin Yang, Yi-Zhe Song, Bodo Rosenhahn, and Tao Xiang. Context-aware layout to image generation with enhanced object appearance. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15049–15058, 2021.

[10] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control.(2022). *URL https://arxiv.org/abs/2208.01626*, 1, 2022.

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[12] Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly ddpm noise space: Inversion and manipulations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12469–12478, 2024.

[13] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[14] Seyed Mehran Kazemi, Rishab Goel, Sepehr Eghbali, Janahan Ramanan, Jaspreet Sahota, Sanjay Thakur, Stella Wu, Cathal Smyth, Pascal Poupart, and Marcus Brubaker. Time2vec: Learning a vector representation of time. *arXiv preprint arXiv:1907.05321*, 2019.

[15] Samar Khanna, Patrick Liu, Linqi Zhou, Chenlin Meng, Robin Rombach, Marshall Burke, David Lobell, and Stefano Ermon. Diffusionsat: A generative foundation model for satellite imagery. *arXiv preprint arXiv:2312.03606*, 2023.

[16] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[17] Konstantin Klemmer, Esther Rolf, Caleb Robinson, Lester Mackey, and Marc Rußwurm. Satclip: Global, general-purpose location embeddings with satellite imagery. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(4):4347–4355, Apr. 2025.

[18] Jae-Gil Lee and Minseo Kang. Geospatial big data: challenges and opportunities. *Big Data Research*, 2(2):74–81, 2015.

[19] Yaxian Lei, Xiaochong Tong, Chunping Qiu, Haoshuai Song, Congzhou Guo, and He Li. Spatial-aware remote sensing image generation from spatial relationship descriptions. *IEEE Geoscience and Remote Sensing Letters*, 22:1–5, 2025.

[20] Jiayi Li, Xin Huang, Lilin Tu, Tao Zhang, and Leiguang Wang. A review of building detection from very high resolution optical remote sensing images. *GIScience & Remote Sensing*, 59(1):1199–1225, 2022.

[21] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

[22] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6038–6047, 2023.

[23] OpenStreetMap contributors. Planet dump retrieved from https://planet.osm.org . `https://www.openstreetmap.org`, 2017.

[24] Leigang Qu, Shengqiong Wu, Hao Fei, Liqiang Nie, and Tat-Seng Chua. Layoutllm-t2i: Eliciting layout guidance from llm for text-to-image generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 643–654, 2023.

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[26] Darcy Reynard. Five classes of geospatial data and the barriers to using them. *Geography compass*, 12(4):e12364, 2018.

[27] 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.

[28] Surya Kant Sahu. Date2vec: Pretrained embeddings for date-time. `https://github.com/ojus1/Date2Vec`, 2021. Accessed: 2025-09-01.

[29] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

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

[31] Datao Tang, Xiangyong Cao, Xingsong Hou, Zhongyuan Jiang, Junmin Liu, and Deyu Meng. Crs-diff: Controllable remote sensing image generation with diffusion model. *IEEE Transactions on Geoscience and Remote Sensing*, 2024.

[32] Qi Zang, Jiayi Yang, Shuang Wang, Dong Zhao, Wenjun Yi, and Zhun Zhong. Changediff: A multi-temporal change detection data generator with flexible text prompts via diffusion model. *arXiv preprint arXiv:2412.15541*, 2024.

[33] Qi Zang, Jiayi Yang, Shuang Wang, Dong Zhao, Wenjun Yi, and Zhun Zhong. Changediff: A multi-temporal change detection data generator with flexible text prompts via diffusion model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 9763–9771, 2025.

[34] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023.

[35] Mu Zhang, Yunfan Liu, Yue Liu, Yuzhong Zhao, and Qixiang Ye. Cc-diff: enhancing contextual coherence in remote sensing image synthesis. *arXiv preprint arXiv:2412.08464*, 2024.

[36] Bo Zhao, Weidong Yin, Lili Meng, and Leonid Sigal. Layout2image: Image generation from layout. *International journal of computer vision*, 128(10):2418–2435, 2020.

[37] Guangcong Zheng, Xianpan Zhou, Xuewei Li, Zhongang Qi, Ying Shan, and Xi Li. Layout-diffusion: Controllable diffusion model for layout-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22490–22499, 2023.

[38] Zhuo Zheng, Stefano Ermon, Dongjun Kim, Liangpei Zhang, and Yanfei Zhong. Changen2: Multi-temporal remote sensing generative change foundation model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[39] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[40] Xiao Xiang Zhu, Devis Tuia, Lichao Mou, Gui-Song Xia, Liangpei Zhang, Feng Xu, and Friedrich Fraundorfer. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE geoscience and remote sensing magazine*, 5(4):8–36, 2017.

A Technical Background

A.1 DDPM

Diffusion Process [11, 29] consists of two processes: a tractable forward process that gradually adds Gaussian noise to data, and a learned reverse process that recovers clean data from noisy inputs. Specifically, the forward process is a Markov chain defined as

$$q(x_t \mid x_{t-1}) = \mathcal{N}\big(x_t; \sqrt{1-\beta_t}\, x_{t-1},\, \beta_t \mathbb{I}\big),$$

with a variance schedule $\beta_{1:T}$. This yields the joint distribution

$$q(x_{1:T} \mid x_0) = \prod_{t=1}^{T} q(x_t \mid x_{t-1}),$$

and a tractable marginal for x_t given x_0:

$$q(x_t \mid x_0) = \mathcal{N}\big(x_t; \sqrt{\bar{\alpha}_t}\, x_0,\, (1-\bar{\alpha}_t)\mathbb{I}\big),$$

where $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$. While the forward process is fixed, the true reverse process $q(x_{t-1} \,|\, x_t)$, which is required for generation, is intractable to compute directly; in DDPM (*Denoising Diffusion Probabilistic Models*; [11]) this reverse step is approximated by a parameterized model p_θ, typically implemented as a neural network:

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}\big(x_{t-1} \mid \mu_\theta(x_t, t),\, \Sigma_\theta(x_t, t)\big).$$

When side information c (e.g., text, mask, layout) is available, the reverse model can be conditioned on c:

$$p_\theta(x_{t-1} \mid x_t, c) = \mathcal{N}\big(x_{t-1} \mid \mu_\theta(x_t, t, c),\, \Sigma_\theta(x_t, t, c)\big).$$

A.2 DDIM Inversion

Denoising Diffusion Implicit Models (DDIM) [30] relax the Markovian assumption of DDPM by introducing a family of non-Markovian transitions. Using the reparameterization trick, the reverse step can be written as

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}} \underbrace{\left(\frac{\boldsymbol{x}_t - \sqrt{1-\alpha_t}\, \epsilon_\theta^{(t)}(\boldsymbol{x}_t, c)}{\sqrt{\alpha_t}} \right)}_{\text{predicted } \boldsymbol{x}_0} + \underbrace{\sqrt{1-\alpha_{t-1}-\sigma_t^2}\, \epsilon_\theta^{(t)}(\boldsymbol{x}_t, c)}_{\text{direction toward } \boldsymbol{x}_t} + \underbrace{\sigma_t\, \epsilon_t}_{\text{random noise}},$$

where $\epsilon_t \sim \mathcal{N}(0, \mathbb{I})$ is independent noise at step t. Setting

$$\sigma_t = \sqrt{\frac{1-\alpha_{t-1}}{1-\alpha_t}} \sqrt{1 - \frac{\alpha_t}{\alpha_{t-1}}}$$

recovers the stochastic DDPM sampler (Markovian), and setting $\sigma_t = 0$ yields a deterministic update, i.e., the DDIM transition. DDIM allows skipping intermediate noise levels for faster sampling and, due to its determinism, enables precise inversion for image editing.

Inversion. Choose an inversion depth $t^\star \in \{0, \ldots, T\}$ that sets the *edit strength*. Starting from an observed image x_{obs} and a *reference* condition c_{ref}, apply the deterministic DDIM forward (noise-adding) updates only up to $t^\star$:

$$x_{t+1} = \sqrt{\alpha_{t+1}} \left(\frac{x_t - \sqrt{1-\alpha_t}\, \epsilon_\theta^{(t)}(x_t, c_{\text{ref}})}{\sqrt{\alpha_t}} \right) + \sqrt{1-\alpha_{t+1}}\, \epsilon_\theta^{(t)}(x_t, c_{\text{ref}}), \qquad t = 0, \ldots, t^\star - 1,$$

initialized at $x_0 = x_{\text{obs}}$. The latent $x_{t^\star}$ is the inversion endpoint: $t^\star{=}T$ gives full inversion; smaller $t^\star$ preserves more of x_{obs} (weaker edits). With $\sigma_t{=}0$, running the matching DDIM denoiser from $t^\star$ back to 0 under the same condition approximately reconstructs x_{obs}.

Redenoising with new conditions. To edit, re-denoise the inverted latents under a *target* condition c_{new} using the DDIM update

$$\tilde{x}_{t-1} \;=\; \sqrt{\alpha_{t-1}} \left(\frac{x_t - \sqrt{1-\alpha_t}\, \epsilon_\theta^{(t)}(x_t, c_{\text{new}})}{\sqrt{\alpha_t}} \right) + \sqrt{1-\alpha_{t-1}}\, \epsilon_\theta^{(t)}(x_t, c_{\text{new}}), \qquad t = t^\star, \ldots, 1,$$

initialized at the encoded $x_{t^\star}$ from the inversion pass. If $c_{\text{new}} = c_{\text{ref}}$ and the same $t^\star$ and schedule are used, the procedure approximately reconstructs x_{obs}; otherwise, it produces an edited output consistent with c_{new}.

Edit strength. The single knob $t^\star$ implements the fidelity–edit trade-off introduced above: smaller $t^\star$ leads to higher fidelity / weaker edits; larger $t^\star$ (up to T) leads stronger edits.

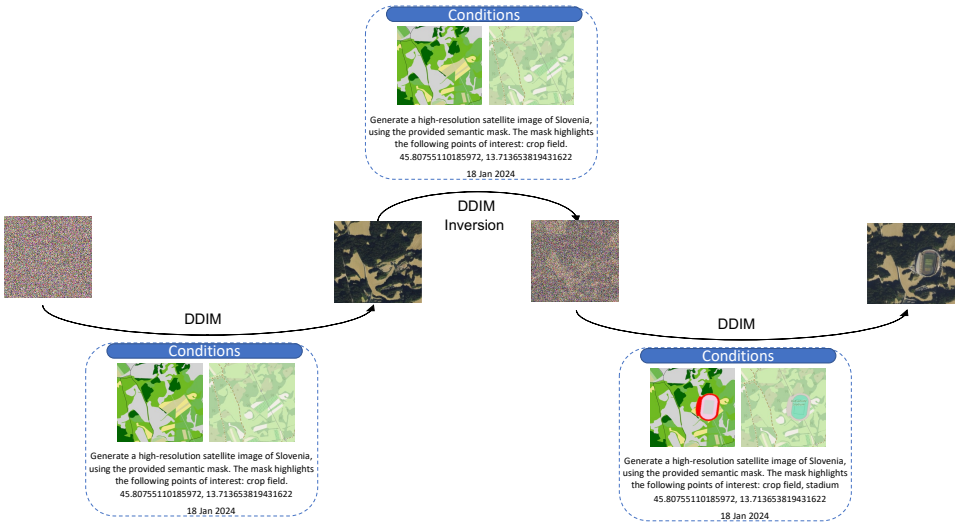

Figure 4: Change synthesis via DDIM inversion.

Figure 4 illustrates this pipeline within our framework to produce consistent edits.

B Detailed Methodology

B.1 Component Encoders and Conditioning

Mask Conditioning via ControlNet We first combine the general and specific segmentation masks by stacking them into a multi-channel tensor and passing this tensor through a small multilayer perceptron (MLP) that projects the concatenated mask channels down to the control-image dimension required by ControlNet. The resulting fused mask embedding is then supplied to ControlNet's image encoder. At each U-Net block, ControlNet produces control features from this embedding and adds them to the corresponding feature maps in the diffusion network, ensuring that the generated output strictly follows the prescribed geometries and class layout.

Spatial Conditioning (Location Encoder) We encode the tile center's longitude–latitude pair (λ, ϕ) using SatCLIP [17]. SatCLIP projects λ and ϕ into a multi-scale sinusoidal basis and refines them via a two-layer MLP to produce a D-dimensional vector $\mathbf{e}_{\text{loc}}$. We then pass $\mathbf{e}_{\text{loc}}$ through a learnable linear projection before adding it to the diffusion timestep embedding at every denoising iteration, thereby injecting geographic context into the noise schedule.

Temporal conditioning (time encoder). To capture seasonal and illumination effects, we encode the capture timestamp using Date2Vec[28], a pretrained, Time2Vec [14]–inspired encoder. Given a 6-D timestamp vector (hh:mm:ss, yyyy-mm-dd), Date2Vec produces an embedding comprising a learned linear component and periodic (sinusoidal) components. As with the spatial pathway, we apply a learnable linear projection to this embedding before adding it to the model's timestep embedding.

Textual Conditioning (Prompt Encoder) We generate a concise prompt for each tile, for example:

> "Generate a high-resolution satellite image in *Country*, using semantic masks highlighting POI_1, POI_2, ..."

This prompt is tokenized and encoded by a frozen CLIP text encoder [25] into $\mathbf{e}_{\text{text}}$. In the main U-Net branch, $\mathbf{e}_{\text{text}}$ is injected via cross-attention, complementing the strict mask and embedding conditioning with flexible, human-readable guidance.

C Problem Context and Literature Review

C.1 Limitations of Existing Satellite Image Generation Methods

Most prior work conditions satellite image synthesis on raster map tiles or on text prompts augmented with bounding-box masks. In the raster-tile approach [33, 38], rich vector data are flattened into a fixed palette of colors and symbols, erasing tag-level distinctions among related feature types (for example, different road classes or categories of commercial establishments) and forcing generative models to treat semantically distinct entities as identical. Consequently, these methods cannot selectively render subtypes of interest at inference time.

When using text descriptions alongside simple rectangular masks[19, 31], each geographic feature is reduced to an axis-aligned box that encloses its true shape. This approximation discards the precise polygonal outlines of buildings, the continuous centerlines of roads, and the irregular boundaries of land-use areas, resulting in spatial misalignment in the synthesized imagery.

Moreover, reliance on a fixed iconography and color scheme limits extensibility. Introducing new feature classes or applying custom styling to existing ones requires manual tile redesign or retraining on freshly collected raster datasets, which undermines the goal of fine-grained, user-driven control. These inherent shortcomings motivate our shift to conditioning directly on raw OSM JSON, where exact geometries and explicit semantic tags remain fully available.

C.2 Advantages of Conditioning on OSM JSON

Directly using OSM JSON grants access to rich, structured metadata that raster tiles or bounding box approaches lack. Every feature in the JSON (nodes, ways, and relations) is annotated with comprehensive key–value tags (e.g. `amenity=restaurant`, `shop=bakery`), as well as hierarchical relations and auxiliary attributes. This level of detail enables the generator to distinguish and render closely related subclasses of features, rather than treating them as uniform pixels.

Moreover, OSM JSON encodes exact vector geometries such as polygonal footprints for buildings and land-use areas, and linear centerlines for roads, which we convert into masks to preserve spatial fidelity. The JSON schema's flexibility supports fine-grained control over the conditioning inputs: users can select specific feature types, adjust per-class palettes, or introduce and use various categories of entities. This deterministic mapping from vector layout to image conditioning ensures fully controllable and repeatable generation.

D Detailed Related Work

D.1 Image Generation

Diffusion models [29, 11] introduced a new powerful approach for image generation and have demonstrated superior performance over existing methods such as Generative Adversarial Networks (GANs) [8] and Variational Auto Encoders (VAEs) [16], often yielding higher-fidelity images [6]. Ho et al. [11] put diffusion models on the map by demonstrating their capability in generating high-quality images. Latent Diffusion Models (LDM) [27] improved upon image generation using diffusion by employing the diffusion process in the latent space instead of the pixel space, resulting in reduced computational complexity and higher fidelity generated images. Furthermore, they used a text encoder alongside cross-attention to enable conditional generation of images based on input text. Stable Diffusion [2] uses the same architecture as LDM and trains the model on a much larger dataset. Furthermore, they improve upon the text encoding process by using CLIP [25] as the text encoder. ControlNet [34] utilizes the Stable Diffusion model and adds extra conditions to the generation process besides text. They enforce the extra conditioning by copying and freezing the pretrained base model. The copied version is trainable and used for learning new conditions, and the output of this model is added back to the frozen model using zero convolution. This enables the model for

fine-grained, structure-preserving generation guided by various conditioning inputs like edge maps, depth information, segmentation masks, or human pose estimations.

D.2 Satellite Image Generation

Recent advances in layout-to-image generation have paved the way for controllable synthesis. In these works, input layouts, which are often represented as bounding boxes, segmentation masks, or other structured formats, serve as the primary condition for image generation. Several works have employed rectangular box layouts for controllable generation, including [37, 4, 9, 36, 24]. In the context of satellite image generation, one approach is to convert spatial relationship descriptions into structured layouts that direct image synthesis. For example, Lei et al.[19] propose a two-stage framework that transforms spatial relationship descriptions into structured layouts and then synthesizes the final image using an enhanced diffusion model with positional prompts and layout attention. Although this method produces highly spatially accurate results, its reliance on fixed layouts and a limited set of classes restricts flexibility in diverse regions. CC-Diff [35] enhances contextual coherence by integrating a dual resampler with foreground-aware attention to align the generated foreground with the background, yet it does not incorporate additional metadata that could improve output control.

Another approach is generating satellite images via image-to-image translation. Some works use GANs for this purpose, including CycleGAN [39] and pix2pix [13]. In particular, pix2pix employs a conditional adversarial framework with a U-Net generator and a PatchGAN discriminator to translate input images into outputs. Although this method can be applied to map-to-satellite conversion, it struggles to deliver high-quality results when the input maps are complex. Diffusion-based approaches have also been explored. ChangeDiff [32] uses a two-stage diffusion process: a text-to-layout model generates layouts via multi-class prompts, which a layout-to-image model converts into images. It yields coherent, diverse outputs, but its narrow vocabulary limits use in complex scenes. Similarly, Changen2 [38] simulates semantic change events in a scene's mask and then employs a diffusion transformer to generate the post-event image. Self-supervised training with SAM-extracted contours enables robust zero-shot performance, though its coarse masks limit fine detail. Tang et al. [31] proposed CRS-Diff, a controllable remote sensing generative model that leverages diffusion models with multi-condition guidance. CRS-Diff supports text, metadata, and image conditions, such as sketch, segmentation mask, HED, and road maps, through a new conditional control mechanism that fuses multi-scale features, achieving precise and realistic remote sensing image synthesis. Additionally, Espinosa and Crowley [7] propose a ControlNet-based method to synthesize satellite images from OSM maps. Their dataset primarily features green vegetation, and reliance on fixed zoom-level maps without additional temporal or semantic metadata limits its adaptability. Finally, DiffusionSat [15] integrates numerical metadata into latent diffusion models for satellite imagery, yielding high-resolution, temporally diverse, and contextually accurate images; however, it fails for prompts with multiple entity classes.

D.3 Image Editing

The ultimate goal of this project is to generate a change dataset. For the effective execution of this task, merely generating satellite images is insufficient. This is because, in a change dataset, the "before" and "after" images must be highly correlated in unmodified areas, and a proper change generation method should preserve this dependency. This challenge is quite similar to the classic problem of image editing, where edits must be applied in a manner that maintains the characteristics of the original image. A significant challenge in this task is the scarcity of training data for "before" and "after" edits, which has spurred the development of numerous training-free approaches. These methods can be broadly categorized into two main groups: (1) methods utilizing cross-attention and (2) methods based on inversions. Cross-attention-based methods gained popularity with the introduction of the Prompt-to-Prompt paper [10]. The core idea is that during text-based image generation, the cross-attention between text tokens and latent features provides a powerful mechanism that can be altered to generate edits. A drawback of these methods is their reliance on the internal architecture of the diffusion model, which can limit their applicability. Furthermore, these methods can only edit images generated by the same model and cannot be applied to arbitrary images.

The other approach for training-free editing employs inversion methods. SDEdit [21] is one of the pioneers in this area. Here, starting from a condition (such as a rough sketch), noise is added to the image, but sparingly, so that some information about the original image is retained. This noisy

image is then denoised to produce the edited image. A similar approach is adopted by the img2img functionality of Stable Diffusion [2], which also incorporates support for text conditioning. A major drawback of this approach is that selecting the optimal amount of added noise to balance the trade-off between faithfulness and realism (or diversity) is quite challenging. One way to address this problem is by using inversions based on a deterministic method like DDIM [30]. In DDIM inversion, after the noise addition phase, if conditions (such as the caption) remain unchanged, the denoised image is guaranteed to be identical to the initial image. This offers the strong fidelity that was lacking in previous methods. However, DDIM inversion relies on the assumption that noise at steps t and $t + 1$ are very close. This assumption can lead to inaccuracies, which are amplified when using classifier-free guidance. More recent works, such as null-text inversion [22] and DDPM inversion [12] have been proposed to address this issue.

E Seasonal Variation

To isolate the effect of temporal conditioning, we fix the semantic masks and text description for a single location and vary only the date input. This cleanly changes season-specific appearance (e.g., vegetation density, color palette, and lighting) while leaving geometry unchanged. Sample images are provided in Figure 5.

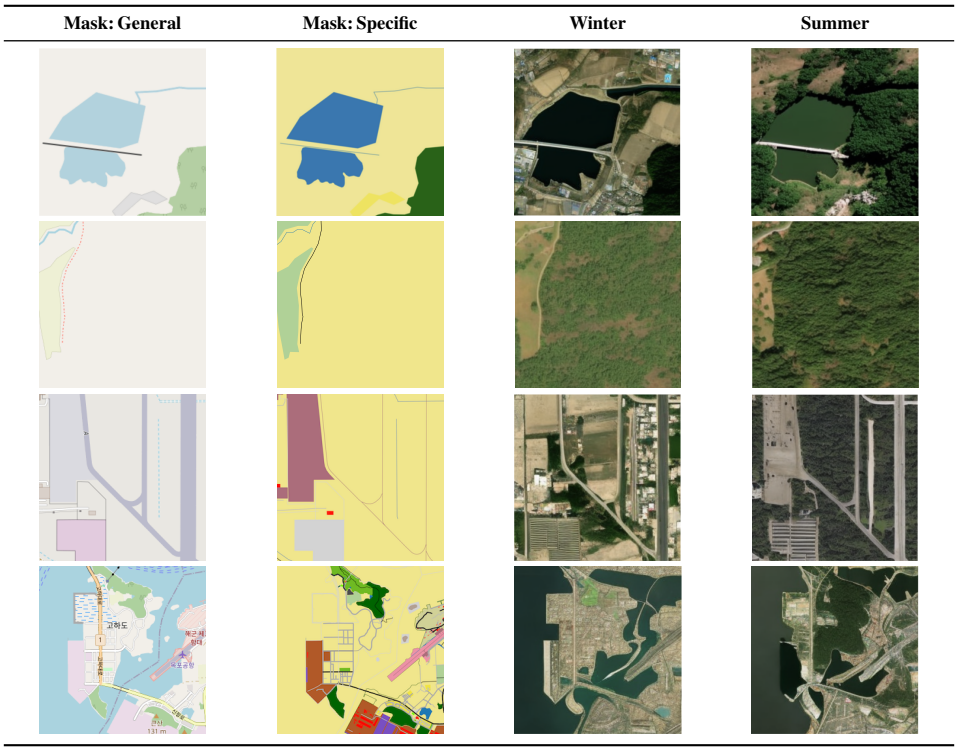

Figure 5: Seasonal conditioning: for fixed masks, varying the date input produces distinct winter and summer images.