RAILWAYGENERATOR: SYNTHETIC RARE EVENT GENERATION FOR RAILWAY SAFETY

A PREPRINT

Marius Dragic
R&D Department Thales-GTS
CentraleSupélec University
marius.dragic@student-cs.fr

Nicolas Chauveau
R&D Department Thales-GTS
IMT Atlantique University
nicolas.chauveau@urbanandmainlines.com

July 17, 2025

ABSTRACT

Railway vision systems must remain reliable in rare but critical situations, such as unexpected pedestrian crossings, obstacles on the tracks, barrier failures in case of level crossing, etc. These events are difficult to capture in real datasets due to rarity, cost, and GDPR consideration which limits the performance and robustness of object detection models in safety-critical scenarios. Thus, generative methods are promising approaches to address this problem and produce realistic synthetic data that can enrich training datasets with underrepresented and hazardous scenarios, without the need to capture them in the real world.

This paper offers a unified overview and experimental comparison of various generative families—StyleGAN2, vanilla diffusion, LoRA-enhanced diffusion, and mask guided diffusion—for synthesising rare railway scenes. Building on these insights, we propose **RailwayGenerator**, a mask and prompt-guided diffusion pipeline that generates targeted railway scenes by controlling the exact location of the generated content. We describe its design, training procedure, and integration into a data-augmentation workflow. All code, fine-tuned weights, and evaluation scripts will be released to foster reproducible research on railway safety.

Keywords Synthetic Data · Data Augmentation · Rare Event Generation · Diffusion Models · LoRA · Railway Safety · GAN · Mask Guided Generation

1 Introduction

Computer vision now plays a central role in railway operations, from obstacle detection on the track to automatic level-crossing supervision and autonomous driving support. To certify such systems, engineers must prove that detectors remain reliable not only in everyday traffic but also during *rare critical* situations—e.g. a half-closed barrier, a pedestrian crossing the tracks, workers on a level crossing, etc. Collecting real video for these cases is ethically sensitive, statistically unlikely, and logistically costly, resulting in poor performance on rare and safety-critical events.

Synthetic imagery can enrich training sets with controlled variations of rare events. Still, each generative family comes with specific trade-offs:

- *GANs* (e.g. StyleGAN2) generate images quickly but often lack geometric consistency in structured environments like railway crossings. They also cannot be easily controlled with prompts or spatial conditions.
- *Diffusion models* are known for producing high-quality and diverse images, but standard fine-tuning can be slow and requires large amounts of memory.
- *LoRA-enhanced diffusion* adds lightweight trainable layers to a frozen diffusion model. This makes it possible to fine-tune the generator on a small dataset with limited GPU resources, while keeping generation quality high.

- *Mask or edge-guided diffusion* (e.g. using ControlNet) allows precise control over scene layout by conditioning the generation process on binary mask, edge detections (like Canny), or other spatial guides. These methods can improve the structure of generated railway scenes, but they require pixel-level annotations, which are sometimes time-consuming to generate if not automated.

We present a practical comparison of several generative methods for railway safety, and introduce **RailwayGenerator**, an image generation pipeline designed to produce realistic railway scenarios. It is based on Stable Diffusion and uses ControlNet to guide the generation process using masks. This allows us to perform targeted inpainting in specific areas of an image—such as adding a rare event (e.g., debris, a pedestrian, animal, etc.) into realistic railway backgrounds—while keeping the rest of the image unchanged.

Unlike prompt-only diffusion methods, **RailwayGenerator** uses both a binary mask and a text prompt. This combination allows for precise control over what is generated and where it appears in the scene. It is especially useful in railway applications, where structure and layout must remain realistic.

Our contributions are as follows:

1. *Unified comparison*: we evaluate four families of generative models—GANs, standard diffusion, LoRA-tuned diffusion, and mask guided diffusion—on their ability to produce realistic and useful railway scenarios.
2. *RailwayGenerator system*: we design and train a ControlNet inpainting model for railway images generation. It allows generation of rare events directly within existing real scenes, improving realism and relevance.
3. *Open and reproducible tools*: we release our training code, and evaluation scripts to support further research on synthetic data for safety-critical vision in railways.

2 Related Work

This section reviews existing generative techniques, from GANs to diffusion, as well as recent adaptations such as LoRA and mask-guided diffusion. We also note applications, when available, to railway scene synthesis.

2.1 Generative Adversarial Networks

Generative Adversarial Networks (GANs) [1] train a generator G to produce images from noise that a discriminator D cannot distinguish from real data. StyleGAN2 [4] introduced a style-based architecture and adaptive augmentations (StyleGAN2-ADA) to improve stability and sample diversity. GANs are fast at inference but often struggle to maintain geometric consistency in structured environments. In the railway domain, CycleGAN [2] has been applied to style-transfer tasks such as adapting daytime to nighttime railway images, but purely GAN-based methods remain limited for precise scene control.

2.2 Diffusion Models

Denoising Diffusion Probabilistic Models (DDPMs) [3] iteratively add and remove Gaussian noise to learn the data distribution. Stable Diffusion [6] further scales diffusion to high-resolution images by operating in a latent space. Diffusion models yield high visual fidelity and diversity, but naïve fine-tuning can be slow and memory-intensive. In railway applications, diffusion-based pipelines have been explored for rare-event synthesis [8], showing qualitative improvements over GAN baselines.

2.3 Parameter-Efficient Tuning: LoRA for Image Generation

Low-Rank Adaptation (LoRA) [5] injects small, trainable low-rank matrices into the frozen weights of a diffusion model's U-Net layers, enabling rapid fine-tuning on domain-specific data with minimal extra parameters. More recently, DiffLoRA [9] treats LoRA adapters as outputs of a diffusion-based hypernetwork, allowing zero-shot personalization of image generators without any gradient updates. These techniques demonstrate that large pretrained diffusion backbones can be cheaply adapted to new visual domains—such as rare railway scenarios—while retaining high image quality and low memory overhead.

2.4 Mask-Guided Diffusion

ControlNet [7] extends diffusion models with side networks that accept spatial conditions—such as segmentation masks, depth maps, or Canny edges—to guide generation. Mask- or edge-guided diffusion enables precise inpainting and

layout control, which is crucial for inserting rare objects into complex railway scenes. While powerful, these methods require pixel-level annotations or precomputed edge maps.

3 Generative Methods

3.1 GAN-based Generation

3.1.1 GANs Theory

Generative Adversarial Networks (GANs) consist of two neural networks trained in opposition: a generator G learns to produce realistic images from random noise, while a discriminator D tries to distinguish between real and generated samples. The training is formulated as a minimax game:

Mathematical Viewpoint

The GAN training process is formulated as a two-player minimax game:

$$\min_G \max_D \; \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))] \tag{1}$$

This framework encourages the generator to create samples that are increasingly similar to the real distribution p_{data}, while the discriminator improves its classification accuracy.

StyleGAN2 [4] improves over previous GAN architectures by introducing a style-based generator with adaptive instance normalization and path regularization. These additions enhance visual quality, training stability, and controllability. The ADA (Adaptive Discriminator Augmentation) variant further stabilizes training on small datasets through learned data augmentations.

3.1.2 Experimental setup

We trained StyleGAN2-ADA using the official NVIDIA PyTorch implementation on a custom dataset of 1,500 railway level-crossing images. All images were resized to 512×512 resolution. Training was performed for 2000 kimg using the default hyperparameters for small datasets, with mirror and translation augmentations enabled. We used one NVIDIA RTX A4500 GPU with 20 GB memory.

3.1.3 Qualitative Results

While the generated images display coarse structures such as barriers, tracks, or trains, the overall quality remains insufficient for practical use. Severe geometric distortions, unnatural colors, and poor semantic coherence are observed across many samples. Vehicles and signs are often warped, shadows are inconsistent, and textures lack realism. Although it is sometimes possible to recognize the railway context, the results are far from the visual fidelity required for downstream applications such as data augmentation or simulation. These limitations highlight the weakness of GAN-based models, especially when trained on small, structured datasets like railway crossings.

Figure 1: Samples of level crossing scenes generated using StyleGAN2 trained on 128x128 images over 2000kimg.

3.2 Diffusion-Based Generation

3.2.1 Diffusion Models: General Theory

Diffusion models are a class of generative models that learn to reverse a stochastic forward process that gradually corrupts data by adding Gaussian noise. The learning objective is to denoise data step by step, eventually sampling high-quality images from random noise.

Mathematical Viewpoint

Formally, the forward process defines a Markov chain of latent variables $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_T$, where $\mathbf{x}_0$ is a clean image and $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is nearly pure noise. The forward process is:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \tag{2}$$

with variance schedule $\beta_t \in (0, 1)$.
Thanks to the Markov structure, we can directly sample any noisy step from the clean image:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}), \quad \text{where } \bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s) \tag{3}$$

The generative model learns to reverse this process by predicting the noise $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$ at each step with a neural network. The standard training loss is:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon},t}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)\right\|^2\right] \tag{4}$$

At inference time, the model denoises from $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$ back to $\mathbf{x}_0$ by iteratively applying learned reverse transitions. This yields high-quality samples with fine-grained control over the generation trajectory.

Latent Diffusion Models [6] improves computational efficiency by operating in a learned latent space $\mathcal{Z}$, where images are encoded via a variational autoencoder $E : \mathbb{R}^{H\times W\times C} \to \mathcal{Z}$. Diffusion is applied in $\mathcal{Z}$, and decoded to pixel space

using a decoder $D : \mathcal{Z} \to \mathbb{R}^{H \times W \times C}$, enabling faster training and lower memory usage without sacrificing image quality.

3.2.2 Prompt-Guided Diffusion (Stable Diffusion)

Theory Prompt-guided diffusion models generate images conditioned on natural language prompts. A text encoder—typically CLIP—is used to convert a user-defined prompt into a fixed embedding vector $\mathbf{c}$. This embedding is injected into the denoising U-Net via cross-attention layers at each timestep, guiding the generation process according to the prompt.

Mathematical Viewpoint

Given a noisy latent $\mathbf{z}_t$ at timestep t and a prompt embedding $\mathbf{c}$, the model learns to predict the noise ϵ added during the forward diffusion process by minimizing:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{\mathbf{z},\epsilon,t}\left[\|\epsilon - \epsilon_\theta(\mathbf{z}_t, t, \mathbf{c})\|^2\right]$$

At inference, classifier-free guidance improves prompt adherence by interpolating the unconditional (ϵ_{uncond}) and conditional (ϵ_{cond}) denoising outputs:

$$\epsilon_{\text{guided}} = \epsilon_{\text{uncond}} + s \cdot (\epsilon_{\text{cond}} - \epsilon_{\text{uncond}})$$

where $s > 1$ controls the guidance strength.

This approach enables the model to steer the generation towards prompt-relevant content while preserving flexibility in details. However, it is fundamentally limited by the representational power of the text encoder: prompts involving rare or technical concepts (e.g., "level crossing") may not yield meaningful generations if underrepresented in the pretraining data.

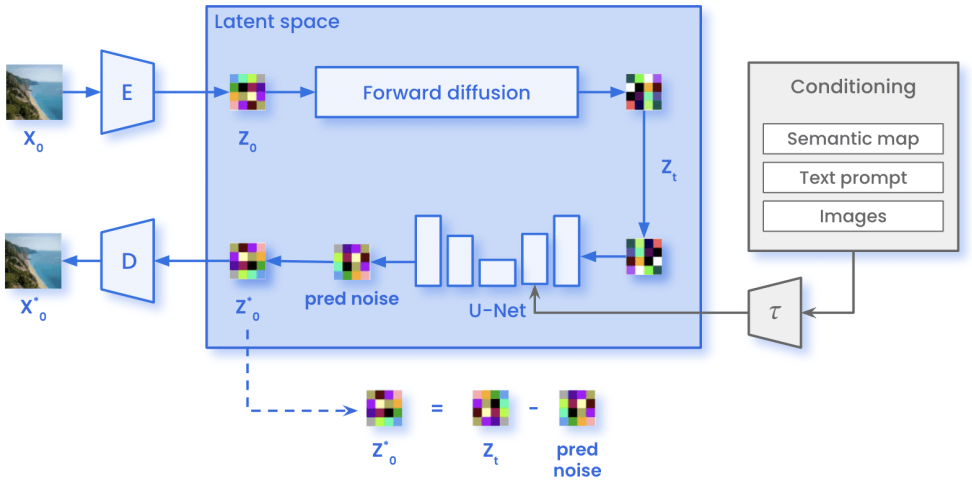


Figure 2: Architecture of a prompt-guided latent diffusion model. The diffusion process operates in latent space $\mathcal{Z}$, guided by prompt embeddings from CLIP illustrated by τ encoder.

Experimental Setup We use the open-source checkpoint `runwayml/stable-diffusion-v1-5`, which is based on the latent diffusion framework. Inference is performed on a RTX A4500 GPU with 20 GB VRAM using FP16 precision to save memory and speed up generation, though at a slight cost in image quality. The prompts are tailored to rare railway events, such as:

- **Pedestrian on the tracks**: `"A realistic person walking away on train tracks, seen from a train driver's cabin, centered, full body, cinematic lighting"`
- **Pedestrian at a level crossing**: Same prompt with `"walking on a level crossing with closed barriers"`

Negative prompts are used to suppress common generation errors like anatomical glitches or unrealistic details: `"blurry, distorted, cartoon, extra limbs, cropped, bad hands, glitch, artifact..."`

(a) Prompted diffusion: person walking on railway tracks.

(b) Prompted diffusion: person at a level crossing (failure case).

Figure 3: Stable Diffusion v1.5 generation for rare railway scenarios. Left: partially successful generation of a pedestrian on tracks. Right: failed result when prompting a pedestrian at a level crossing.

Qualitative Results The image of a pedestrian on railway tracks shows acceptable realism, with coherent layout and recognizable elements, despite minor artifacts such as soft details and slight anatomical inaccuracies. However, the generation of a pedestrian at a level crossing fails: the scene lacks semantic structure and the notion of "level crossing" seems poorly understood by the model. These limitations illustrate the dependency of prompt-guided diffusion on the quality and coverage of CLIP's pretraining data, making it unreliable for domain-specific scenarios without additional adaptation.

3.2.3 LoRA-Enhanced Diffusion

Theory Low-Rank Adaptation (LoRA) [5] enables efficient fine-tuning of large diffusion models by injecting small trainable low-rank matrices into the frozen weights of a pretrained U-Net—especially in cross-attention and feed-forward layers. This approach drastically reduces the number of trainable parameters required for domain adaptation, while preserving the original model's generalization.

Mathematical Viewpoint

Given a pretrained weight matrix $W \in \mathbb{R}^{D\times D}$, LoRA introduces two low-rank trainable matrices $A \in \mathbb{R}^{D\times r}$ and $B \in \mathbb{R}^{r\times D}$, with $r \ll D$, so that the effective weight becomes:

$$W' = W + \Delta W \qquad \Delta W = BA$$

where $\mathrm{rank}(\Delta W) \leq r$, reducing trainable parameters from D^2 to $2Dr$.
The LoRA adapters A, B are optimized via the diffusion loss:

$$\mathcal{L}(\theta, A, B) = \mathbb{E}_{\mathbf{z}_0,\boldsymbol{\epsilon},t} \left\| \boldsymbol{\epsilon} - \epsilon_{\theta+\Delta\theta}(\mathbf{z}_t, t, \mathbf{c}) \right\|^2,$$

where θ are the frozen pretrained weights and $\Delta\theta$ collects all LoRA updates across layers.

By constraining the weight updates to a low-dimensional subspace, LoRA achieves rapid, memory-efficient adaptation, with minimal risk of catastrophic forgetting.

Experimental Setup We start from the Stable Diffusion v1.5 checkpoint (`runwayml/stable-diffusion-v1-5`) and insert LoRA adapters of rank $r = 4$ into all cross-attention and MLP layers of the U-Net. Our fine-tuning dataset

consists of 100 manually curated railway images (including both normal and rare event scenes at level crossings and trackside) resized to 512×512. We train for 12 epochs with a learning rate of $1e{-}4$ on the adapter weights only (base model frozen), using a batch size of 16 and FP16 precision on a single NVIDIA A4500 GPU (20 GB VRAM). Classifier-free guidance is preserved with a guidance scale of 7.5 during inference.

Qualitative Results As shown in Figure 4, LoRA-based fine-tuning leads to a clear improvement in the model's ability to understand and generate the complex structure of level crossings. Compared to prompt-only diffusion, the synthesized images demonstrate a much better understanding of the key elements that define a railway crossing—such as barriers, tracks, and signaling equipment. This highlights the significant contribution of LoRA in teaching diffusion models to represent domain-specific concepts that are underrepresented in the pretraining data.

However, several generation failures remain. Some images exhibit unrealistic context, such as barriers being open while a train is passing, tracks oriented incorrectly, or deformed barriers. These artifacts reveal the ongoing challenges in synthesizing scenes where geometric and semantic constraints are strict. Level crossings are particularly difficult to model, due to their complex structure and highly variable pixel distribution. Maybe in this case fine-tuning would me more relevant and would improve results, but it requires substantial GPU resources and high-quality, curated datasets to be fully effective.

Overall, LoRA fine-tuning substantially boosts prompt adherence and scene realism for challenging concepts, but perfect generation of structured scenes like level crossings remains an open problem, with frequent context or geometry errors that limit their use in downstream safety-critical applications.

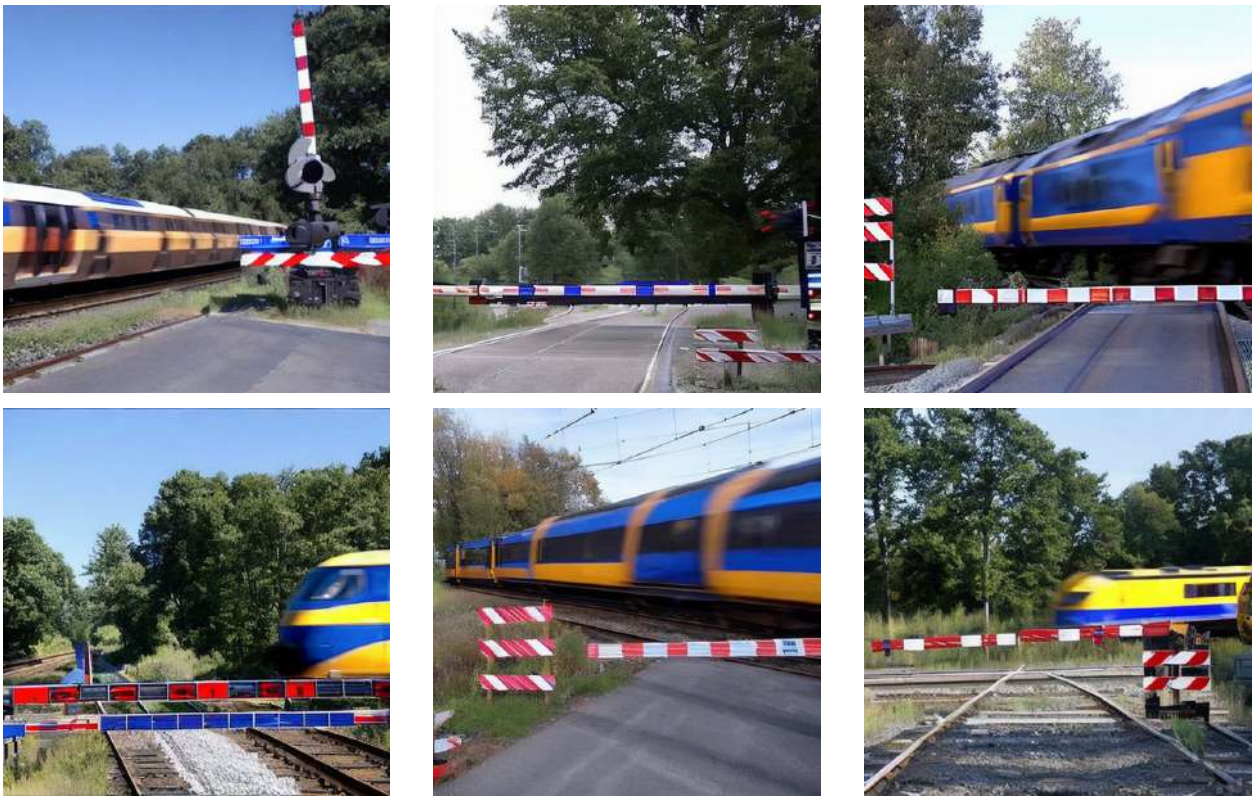


Figure 4: Examples of level crossing generations using LoRA fine-tuned diffusion models.

3.2.4 Mask-Guided Diffusion (ControlNet)

Theory ControlNet [7] extends diffusion models by introducing explicit spatial conditioning through auxiliary inputs, such as binary masks, edge maps, or segmentation maps. The model augments the standard diffusion U-Net with a parallel conditional branch that processes the structural guidance input $\mathbf{c}_{\text{struct}}$, while keeping the original diffusion weights frozen. At each layer, residual features from the structural branch are injected into the main U-Net, enabling precise control over the generated image layout without forgetting the learned distribution.

This architecture allows the model to jointly leverage both *semantic* guidance (via prompt embedding $\mathbf{c}_{\text{text}}$) and *spatial* guidance (via $\mathbf{c}_{\text{struct}}$), as illustrated in Figure 5. The result is a diffusion process that faithfully follows both textual and spatial constraints, producing edits or insertions exactly at the prescribed locations.

Mathematical Viewpoint

Let $\mathbf{x}$ denote the input image, which is first encoded into the latent space via $\mathcal{E}(\mathbf{x}) = \mathbf{z}_0$ using a VAE encoder. During diffusion, we consider a noisy latent $\mathbf{z}_t$ at timestep t, and two types of conditioning: a spatial condition $\mathbf{c}_{\text{struct}}$ (e.g., segmentation map, mask) and a semantic/textual condition $\mathbf{c}_{\text{text}}$.
The ControlNet-augmented denoising U-Net predicts the added noise at each step as:

$$\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t, \mathbf{c}_{\text{text}}, \mathbf{c}_{\text{struct}}) = \text{U-Net}_\theta\left(\mathbf{z}_t, t, \mathbf{c}_{\text{text}}\right) + \mathcal{F}_\theta\left(\mathbf{z}_t, t, \mathbf{c}_{\text{struct}}\right)$$

where $\mathcal{F}_\theta$ is the parallel conditional branch (ControlNet) processing the structural map, whose features are injected at each U-Net block.
The training objective is the standard diffusion loss with both conditions:

$$\mathcal{L}_{\text{ControlNet}} = \mathbb{E}_{\mathbf{x},t,\boldsymbol{\epsilon},\mathbf{c}_{\text{struct}}}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\left(\mathbf{z}_t, t, \mathbf{c}_{\text{text}}, \mathbf{c}_{\text{struct}}\right)\right\|^2\right]$$

At inference, the decoder $\mathcal{D}$ maps the final latent back to pixel space, and the spatial conditioning $\mathbf{c}_{\text{struct}}$ enforces strict control over the structure of the generated image, as illustrated in the figure.

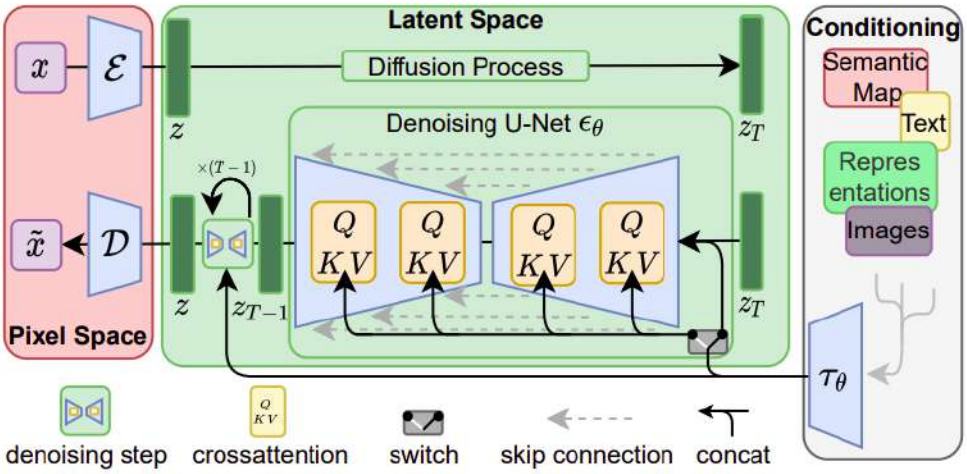


Figure 5: ControlNet: structural conditioning (e.g., masks) is injected into the denoising U-Net through residual connections at each layer.

Experimental Setup

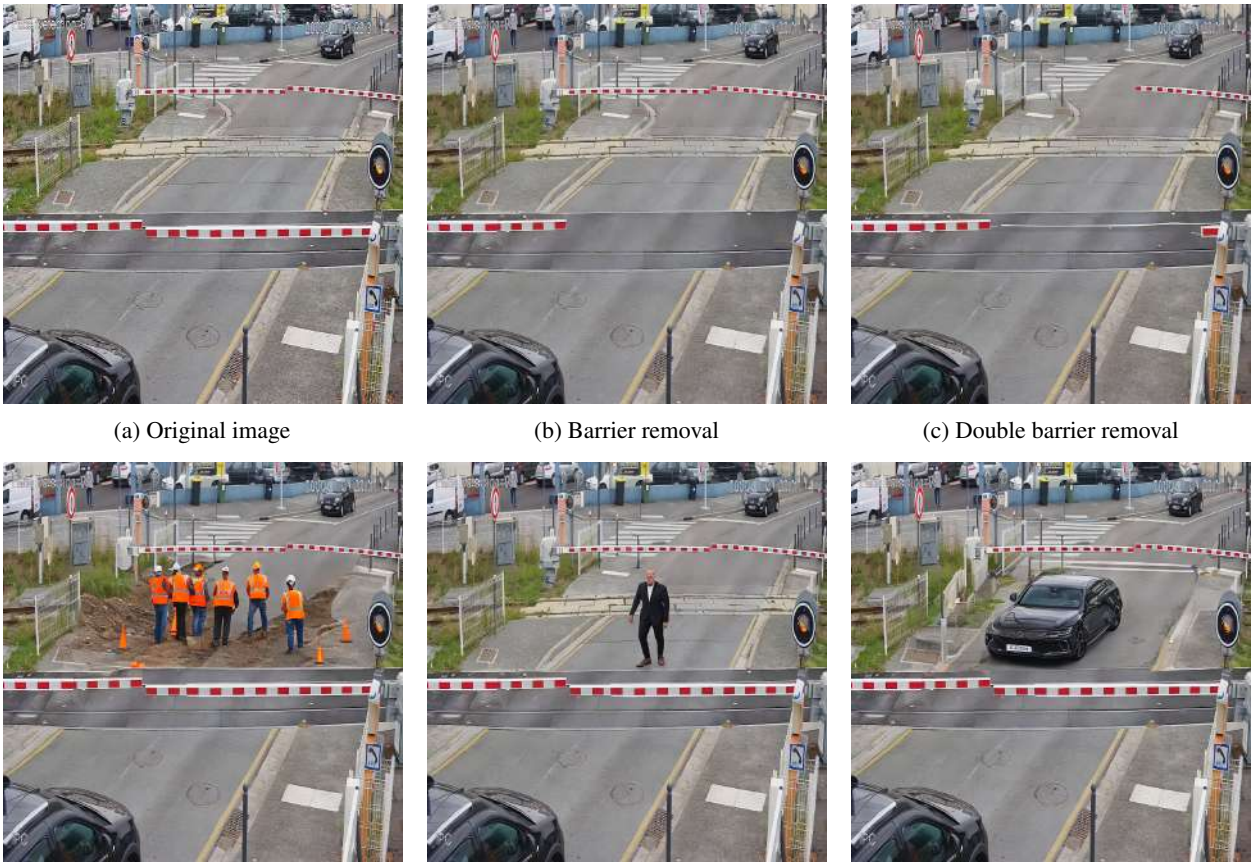
Prompt + Mask-Guided Inpainting We conducted a first set of experiments using prompt-based generation guided by binary masks. These masks specify the regions to inpaint—either to remove unwanted elements or insert new objects. We used the open-source `StableDiffusionInpaintPipeline` combined with a ControlNet backbone trained for inpainting tasks (`xinsir-controlnet-union-sdxl-1`). All generations were performed with the `stabilityai/sd2-inpainting` model using FP16 precision on an NVIDIA A4500 GPU (20 GB VRAM), yielding a generation time of approximately 5 seconds per image at 768×768 resolution and 25 denoising steps.

Different masks were used depending on the scenario. For instance, to simulate infrastructure failures, we erased a barrier using a mask and a prompt such as `"a missing barrier, signs of recent impact..."`. To simulate human presence, we masked out an empty region of the level crossing and used prompts like `"a realistic pedestrian, full body, walking across..."`. This setup enables targeted generation while preserving the realism of the background.

(a) Original image (b) Mask for barrier removal (c) Mask for pedestrian insertion

Figure 6: Different masks used for ControlNet inpainting: barrier suppression and pedestrian generation.

Qualitative Results The generations obtained via ControlNet inpainting show high visual fidelity and strong prompt adherence. The inserted pedestrians exhibit realistic anatomy, lighting consistency, and plausible integration into the scene, even under strict viewpoint constraints. In the case of barrier removal, the inpainted regions maintain background coherence, preserving texture continuity in the road and track structures. No major artifacts or semantic inconsistencies are observed. Thanks to precise spatial guidance through binary masks, the model avoids hallucinations and respects the desired location of edits. Overall, these results confirm that mask-guided diffusion can produce highly realistic and controllable augmentations of rare railway scenes.



(a) Original image (b) Barrier removal (c) Double barrier removal

(d) Group worker (e) Pedestrian (f) Car

Figure 7: Examples of mask-guided inpainting with ControlNet for rare railway events on a railroad crossing.

4 Comparative Evaluation of Generative Methods

To provide a comprehensive and reproducible benchmark, we compare all tested generative methods on both qualitative and quantitative metrics, including sample fidelity, human-perceived realism, controllability, as well as training and

inference costs. For comparison purpose experiments were conducted on a single NVIDIA RTX A4500 GPU (20GB VRAM), ensuring consistency in resource reporting.

Method	FID↓	Human Realism↑	Prompt Adherence↑	Mask Adherence↑	Train Time (h)↓	Inf. time (s)↓
StyleGAN2-ADA 512px	413.4	0.5 ± 0.2	–	–	36.5	**0.02**
Stable Diffusion v1.5	51.2	2.6 ± 0.7	57%	–	–	2.2
Stable Diffusion v1.5 + LoRA	36.7	3.8 ± 0.5	79%	–	**2.2**	2.3
ControlNet 768 px	**29.3**	**4.3** ± 0.4	**92%**	**98%**	–	2.6

Table 1: **Quantitative and qualitative comparison of generative models for rare event synthesis in railway imagery. FID** (Fréchet Inception Distance, lower is better) quantifies distributional similarity between real and generated images over 500 samples. **Human Realism** is the average score assigned by three annotators on a random subset of 50 images (1 = unrealistic, 5 = fully realistic). **Prompt Adherence** is the percentage of generations judged as accurately reflecting the intended prompt. **Mask Adherence** is the percentage of samples where the object appears correctly within the masked region (not applicable to unconditioned methods). **Train Time** and **Inference Time** are measured on a single NVIDIA RTX A4500; inference time is per 512×512 image, averaged over 100 runs. Best results for each metric are in **bold**.

Model	Params (M)	VRAM Inference (GB)	Dataset Size	Annotation
StyleGAN2-ADA 512px	32.6	1.4	1,500 images	None
Stable Diffusion v1.5	860	3.5	–	–
Stable Diffusion v1.5 + LoRA	8.4	3.5	100 images	Minimal (train)
ControlNet 768px	14	13.5	–	High (inference)

Table 2: **Resource and data requirements for each generative method at** 512×512 **resolution.**
Params is the number of parameters updated during training (in millions). **VRAM Inference** is the typical GPU memory used to generate a 512×512 image with batch size 1. **Dataset Size** is the number of images used for training or fine-tuning if training was necessary. **Annotation** reflects the manual effort required.

Discussion. Table 1 highlights the significant differences between generative methods for rare railway event synthesis. ControlNet-based diffusion consistently achieves the best overall performance, with the lowest FID, highest realism, and exceptional adherence to both prompts and spatial masks—demonstrating its suitability for safety-critical data augmentation where precise control is required. However, this comes at the cost of greater annotation effort (pixel-level masks) for inference and slightly increased inference memory usage.

LoRA-enhanced diffusion provides an effective compromise, offering strong prompt adherence and realism (almost matching ControlNet) while requiring only a fraction of trainable parameters (8.4M) and a small curated dataset. Its minimal annotation needs and moderate VRAM usage make it especially attractive for domains with limited resources or tight project timelines.

Prompt-only diffusion (Stable Diffusion v1.5) improves realism and FID over StyleGAN2, but still suffers from limited controllability and can struggle to generate rare or structured railway scenarios without further adaptation.

StyleGAN2, despite its negligible inference footprint, is clearly outperformed on all qualitative and quantitative criteria. The method fails to produce visually plausible or prompt-aligned scenes at high resolution, highlighting the limits of GANs for this structured domain.

Table 2 shows that modern diffusion methods, especially LoRA, achieve a good trade-off between performance and resource use. The memory required for inference is low enough that these methods can run efficiently on standard GPUs, without special hardware.

In summary, ControlNet provides the best overall performance when precise spatial control is needed, while LoRA fine-tuning offers a fast and efficient solution for adapting to new domains with minimal data and compute.

References

[1] Ian Goodfellow et al. "Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems (NeurIPS)* 27 (2014), pp. 2672–2680.
https://arxiv.org/abs/1406.2661.

[2] Jun-Yan Zhu et al. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 2223–2232.
https://arxiv.org/abs/1703.10593.

[3] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising Diffusion Probabilistic Models". In: *Proceedings of the 34th International Conference on Neural Information Processing Systems (NeurIPS)* (2020), pp. 6840–6851.
https://arxiv.org/abs/2006.11239.

[4] Tero Karras, Samuli Laine, and Timo Aila. "Analyzing and Improving the Image Quality of StyleGAN". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 8110–8119.
https://arxiv.org/abs/1912.04958.

[5] Edward J. Hu et al. "LoRA: Low-Rank Adaptation of Large Language Models". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2021, pp. 2233–2243.
https://arxiv.org/abs/2106.09685.

[6] Robin Rombach et al. "High-Resolution Image Synthesis with Latent Diffusion Models". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 10684–10695.
https://arxiv.org/abs/2112.10752.

[7] Ming Zhang et al. "Adding Conditional Control to Text-to-Image Diffusion Models". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2023.
https://arxiv.org/abs/2302.05543.

[8] Andrei–Robert Alexandrescu et al. "ContRail: A Framework for Realistic Railway Image Synthesis using ControlNet". In: *arXiv preprint arXiv:2412.06742* (Dec. 2024). Published 9 December 2024.
https://arxiv.org/abs/2412.06742.

[9] Yujia Wu et al. "DiffLoRA: Generating Personalized Low-Rank Adaptation Weights with Diffusion". In: *arXiv preprint arXiv:2408.06740* (2024).
https://arxiv.org/abs/2408.06740.