Project Aether

Reinforcement Learning for Optimal Concept Steering in Diffusion Models

Team: Alkan, Durak, Chiucchiolo

Sapienza University of Rome • Fall 2025

The Problem

Diffusion models generate unsafe or undesired content

Limited control over latent representations during generation

Current Limitations:

- Retraining models is expensive and time-consuming
- Post-hoc filtering misses subtle unsafe content
- Existing methods lack fine-grained control

OUR GOAL

Steer latent representations during generation to avoid unsafe concepts without retraining

Key Innovation:

Use RL to learn an optimal steering policy with transport-based rewards

Our Approach: Three Phases

1 Linear Probing

Train probes to detect unsafe concepts in latent space at each timestep

- Collect latents from diffusion process
- Train binary classifiers per timestep
- Identify optimal intervention window

2 PPO Training

Learn a policy to steer latents away from unsafe regions

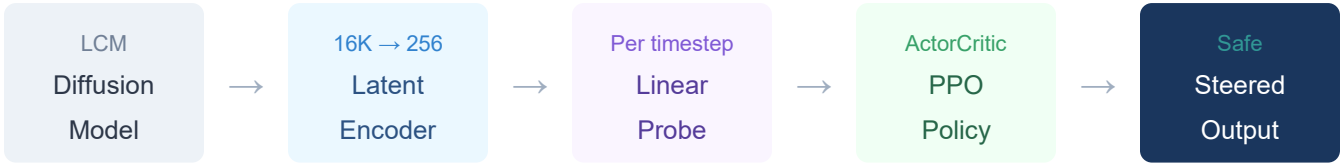
- Gymnasium environment wrapping SD
- Combined safety + transport reward
- ActorCritic network with PPO

3 Evaluation

Measure steering effectiveness and image quality preservation

- SSR: Safe Steering Rate
- FPR: False Positive Rate
- LPIPS: Perceptual quality

Technical Architecture



Safety Reward (R_safe)

Derived from probe predictions. Rewards steering away from detected unsafe concepts.

```
R = -log(p_unsafe) when unsafe detected
```

Transport Reward (R_transport)

Wasserstein-2 distance penalty to preserve semantic content during steering.

$$R = -\lambda \cdot W_2(z_{orig}, z_{steered})$$

Combined Reward: $R_{total} = R_{safe} + \lambda \cdot R_{transport}$ where λ balances safety vs. quality

Current Progress

COMPLETE

Phase 1: Linear Probing

Achieved **90% accuracy** at timesteps 2-3, **97% AUC** at t=1 • Optimal window: timesteps [2, 6]

COMPLETE

Phase 2: PPO Training

50K timesteps completed • Policy loss improved: -0.121 → -0.085 • Model saved

NEEDS WORK

Phase 3: Evaluation

Initial results on 30 samples • Metrics below targets • Extended training prepared

Evaluation Results (30 samples)

SSR (Safe Steering Rate)	FPR (False Positive Rate)	LPIPS (Perceptual)	Transport Cost
13.3%	26.7%	0.32	70.4
Target: >80%	Target: <5%	Target: <0.30	±46.5

Challenges & Solutions

PROBLEM

GPU Memory Constraints

Running the project locally caused OOM errors during training

SOLUTION

Memory Optimizations

Reduced batch (32→8), rollout (512→64), epochs (10→4), added CUDA cache clearing + Ran the solution on colab & kaggle

PROBLEM

Training Instability (NaN)

Observations caused NaN values in policy network during early training

SOLUTION

Observation Normalization

Added running mean/std normalization to latent observations before policy input

PROBLEM

Slow Inference (50 steps)

Standard Stable Diffusion requires 50 denoising steps per image

SOLUTION

LCM Model (8 steps)

Switched to Latent Consistency Model - 3x faster inference with similar quality

PROBLEM

Low SSR/High FPR

Initial evaluation shows policy not effectively steering (13% SSR, 27% FPR)

IN PROGRESS

Extended Training + Tuning

Prepared 150K timestep config, hyperparameter experiments (λ, LR, epochs)

Next Steps

Immediate Tasks

1. Extended Training
Run 150K timesteps with λ=0.3

2. Hyperparameter Sweep
9 experiments: λ, LR, epochs, policy size

3. Larger Evaluation
100+ samples for statistical significance

4. Unit Tests
Add test_env.py, test_rewards.py, test_ppo.py

Planned Experiments

Experiment	Values	Hypothesis
λ (transport)	0.3, 0.5, 0.8, 1.0	Lower = better SSR
Learning Rate	1e-4, 2e-4	Higher = faster conv.
PPO Epochs	4, 10	More = better policy
Policy Size	Small, Default	Smaller = faster

Target Outcomes

SSR >80% • FPR <5% • LPIPS <0.30

References

[1] Lamba, H., et al. (2025). "Alignment and Safety of Diffusion Models via Reinforcement Learning." *arXiv preprint*.

[2] Wagenmaker, A., et al. (2025). "Steering Your Diffusion Policy with Latent Space RL." *arXiv preprint*.

[3] Schulman, J., et al. (2017). "Proximal Policy Optimization Algorithms." *arXiv:1707.06347*.

[4] Alain, G. & Bengio, Y. (2016). "Understanding Intermediate Layers Using Linear Classifier Probes." *arXiv:1610.01644*.

[5] Holderrieth, P. & Erives, C. (2025). "MIT 6.S184: Generative AI with Stochastic Differential Equations." *MIT Course*.

[6] Luo, S., et al. (2023). "Latent Consistency Models." *arXiv:2310.04378*.

Thank You