Alireza Limooie

Geometric Noise Augmentation for Roof Graph Reconstruction

Date: 21/07/2025

1. Objective & Background

Following the analysis in the previous report (14/07/2025), which identified significant instability and inter-component friction in our loss function, the primary objective of this work period was to fundamentally re-engineer the training loss architecture. The goal was to create a stable, balanced, and effective training signal that could handle the highly realistic and challenging data produced by our advanced geometric augmentation pipeline.

With a stable loss framework established, a secondary objective was to explore a more advanced, multi-step evaluation paradigm: **iterative refinement**. This method tests the model's ability to progressively improve a noisy graph by feeding its own output back as a new input, simulating a real-world refinement loop. This report details the successful redesign of the loss functions, the resulting performance improvements, and the new challenges and insights revealed by the iterative refinement evaluation.

2. Advancements in Methodology

The core of this period's work involved two major, interconnected advancements: a complete overhaul of the loss function components and a subtle but important enhancement to the data augmentation strategy.

2.1. Redesigned and Balanced Loss Functions

The previous loss functions, particularly the squared-distance connectivity term, were found to be overly aggressive, leading to graph collapse. The new framework addresses this by redesigning each component to be more robust and by introducing a normalization scheme to balance their influence.

- **Logarithmic Connectivity Loss:** The unstable squared-distance (L2) loss has been replaced with a **Logarithmic Connectivity Loss** (log(1 + T*d)). This function is steep for small junction errors, encouraging precise connections, but flattens for large errors. This critical change prevents the model from collapsing the entire graph to a single point, as it no longer receives an infinitely growing penalty for distant endpoints. It gently pulls junctions together rather than violently yanking them.
- **Linear Reconstruction Loss:** The reconstruction loss was changed from squared-distance to a simple **linear distance** penalty. This provides a constant gradient, preventing the vanishing gradient problem for small errors and avoiding exploding gradients for large errors, leading to more stable training.
- **Exponential Offset Regularization:** For models predicting offsets, a new **exponential regularization** term (B^||o|| - 1) was introduced. This strongly penalizes large, unnecessary corrections, encouraging the model to make minimal, precise adjustments, which is crucial for iterative refinement.

- **Automatic Loss Balancing:** A new --balance-losses flag was implemented. When active, it normalizes all loss components so they have an equal value at a user-defined "balance distance" (see Figure 1). The loss weights now act as intuitive multipliers on top of this balanced baseline, making the tuning process significantly more predictable and resolving the inter-component friction.

As shown in the loss visualization (Figure 1), these new functions are smoother and less prone to extreme values, creating a much more stable optimization landscape.

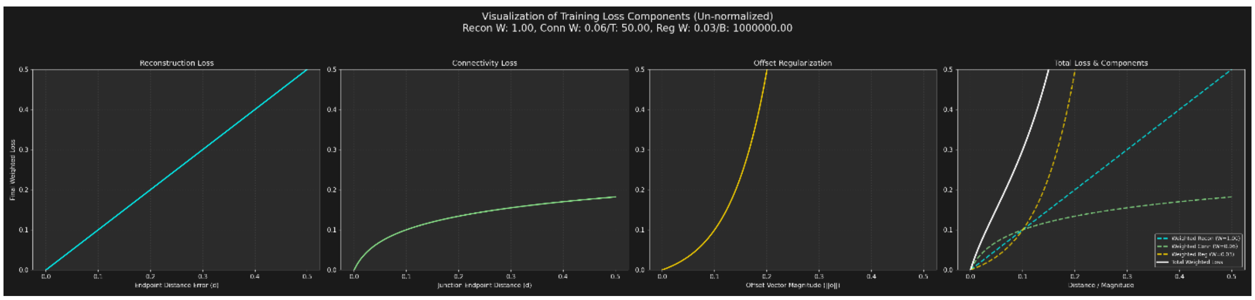


Figure 1: Visualization of the Redesigned and Balanced Training Loss Components. The new functions (linear reconstruction, logarithmic connectivity, exponential regularization) are designed to be less aggressive at large distances, preventing graph collapse and promoting stable training.

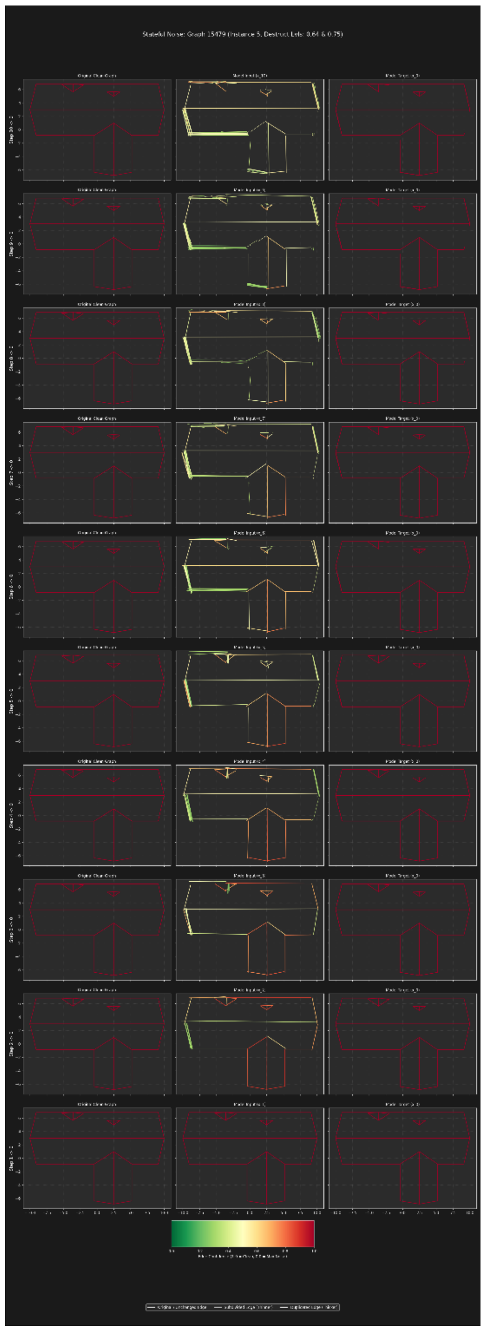
2.2. Structurally-Aware Augmentation Grouping

The geometric augmentation model was refined to create more realistic and structurally-aware noise. Edges are now probabilistically assigned to one of two groups based on the topological importance of their endpoints:

- **Low-Noise Group:** Edges forming the "core" of the roof structure (i.e., connecting two high-degree nodes) are more likely to be assigned to this group. They receive less severe geometric and topological noise.

- **High-Noise Group:** Edges on the "border" of the graph (i.e., connected to a degree-1 node) are more likely to be assigned here, undergoing more aggressive corruption.

This enhancement forces the model to learn that not all input edges are equally reliable, better simulating real-world scenarios where detector confidence varies across a roof.



3. Model Performance & New Evaluation Paradigm

3.1. Performance on Single-Step Denoising

Training the model with the new loss functions and augmented data yielded significant improvements. The total loss now converges smoothly and consistently, as seen in the convergence plot (Figure 2).

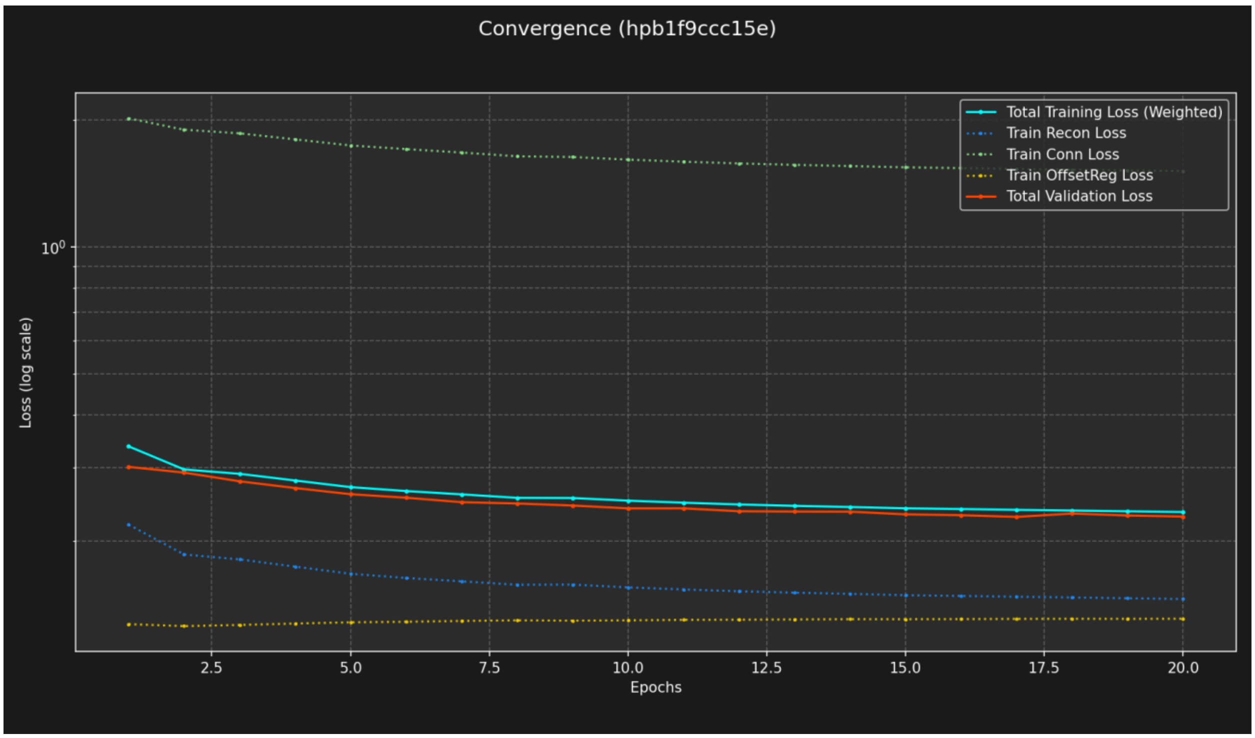


Figure 2: Convergence Plot with New Loss Architecture. The total training loss (cyan) shows stable and consistent convergence, a stark contrast to the fluctuating loss observed previously. The individual components also behave predictably.

Most importantly, the primary challenge from the last report has been solved: **the model now produces topologically sound and geometrically accurate reconstructions**. The logarithmic connectivity loss successfully enforces junctions without shrinking or distorting the overall graph shape. Qualitative evaluation (Figures 3 and 4) shows the model effectively handling complex, noisy inputs with subdivisions and duplications, correctly reconstructing the underlying clean geometry.

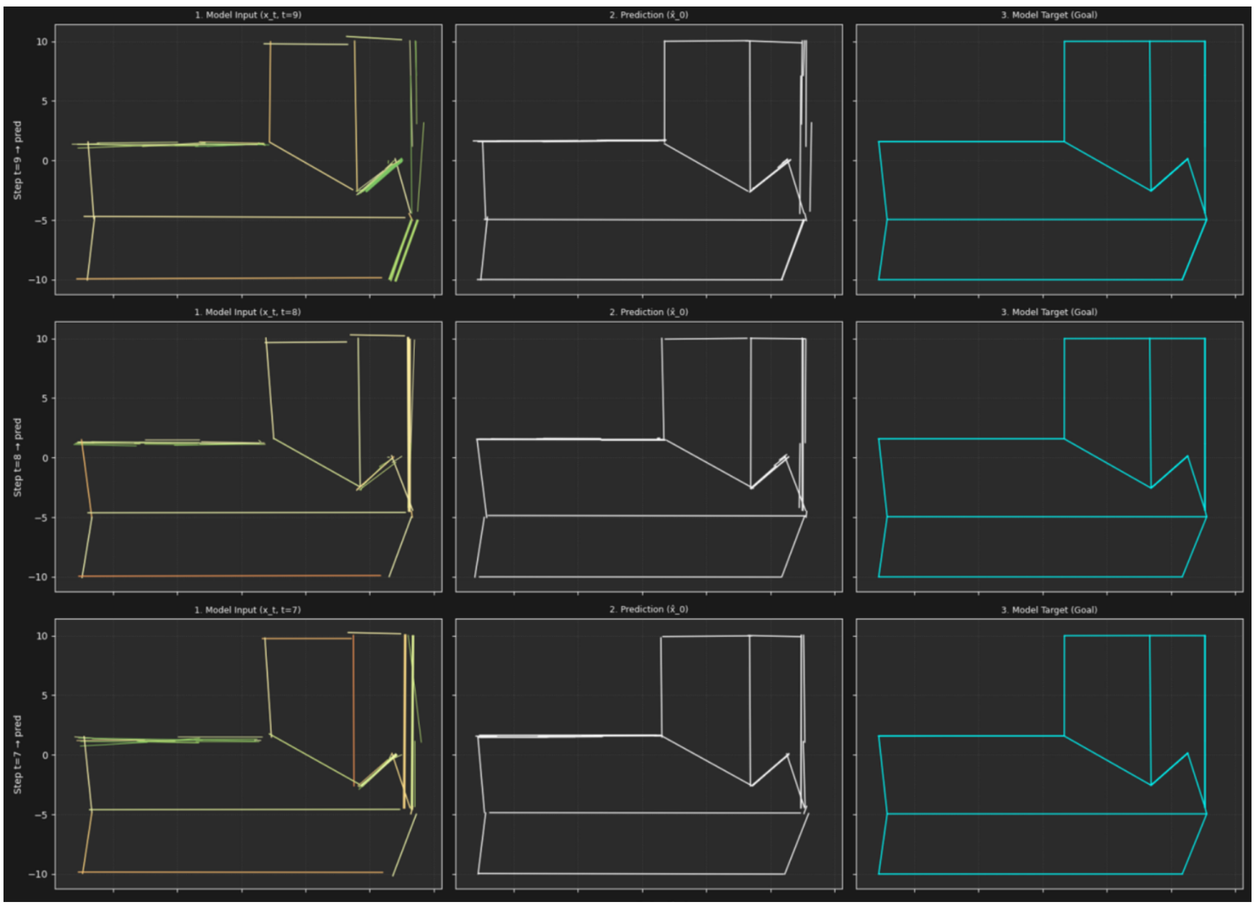


Figure 3-a

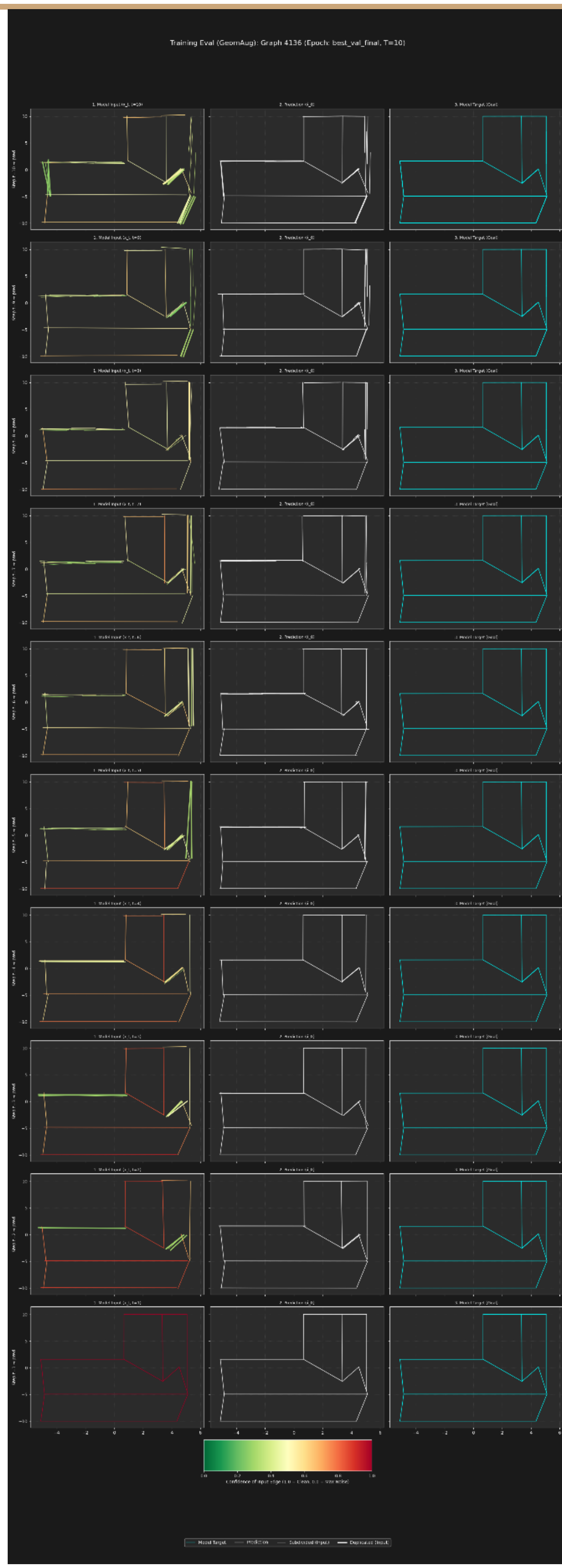


Figure 3-b

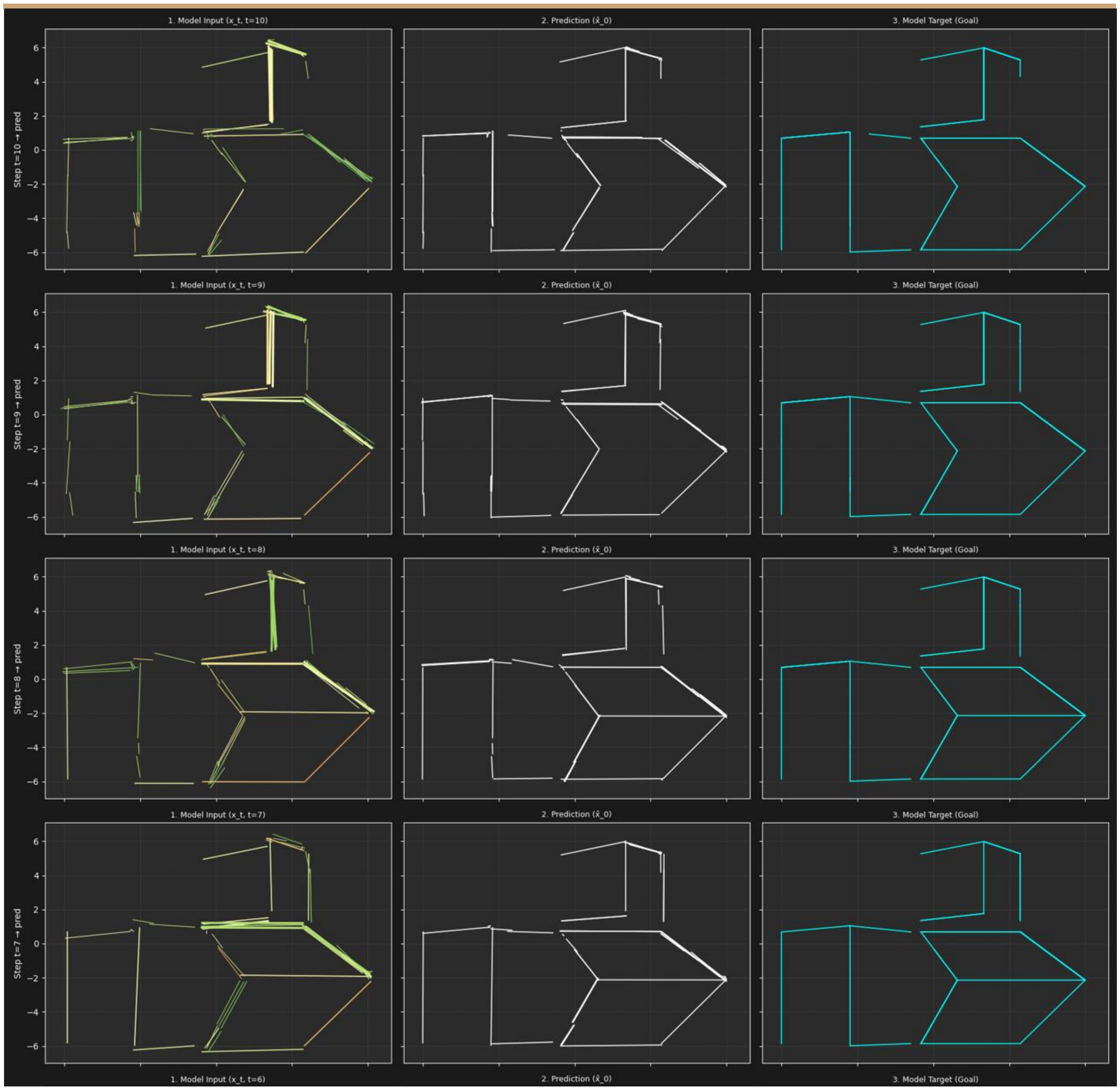

1. Model Input (x_t, t=10)
2. Prediction (x̂_0)
3. Model Target (Goal)
Step t=10 → pred
1. Model Input (x_t, t=9)
2. Prediction (x̂_0)
3. Model Target (Goal)
Step t=9 → pred
1. Model Input (x_t, t=8)
2. Prediction (x̂_0)
3. Model Target (Goal)
Step t=8 → pred
1. Model Input (x_t, t=7)
2. Prediction (x̂_0)
3. Model Target (Goal)
Step t=7 → pred
1. Model Input (x_t, t=6)
2. Prediction (x̂_0)
3. Model Target (Goal)

Figure 4-a


Training Eval (GeomAug): Graph 7297 (Epoch: best_val_final, T=10)
0.0
0.2
0.4
0.6
0.8
1.0
Model Target
Prediction

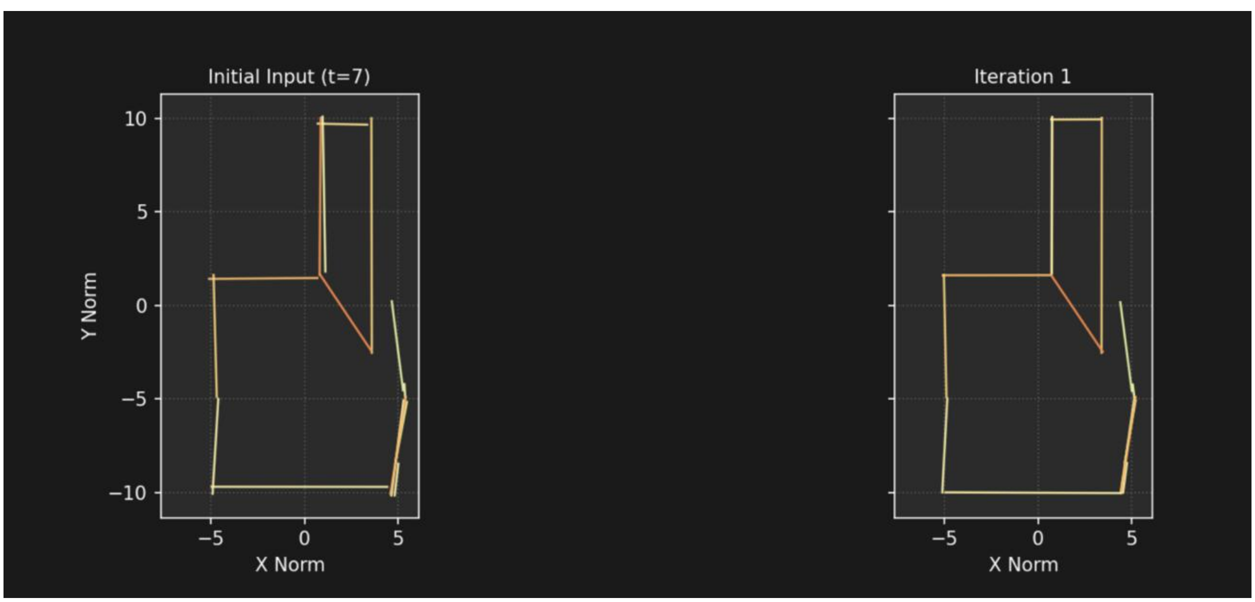
Figure 4-b

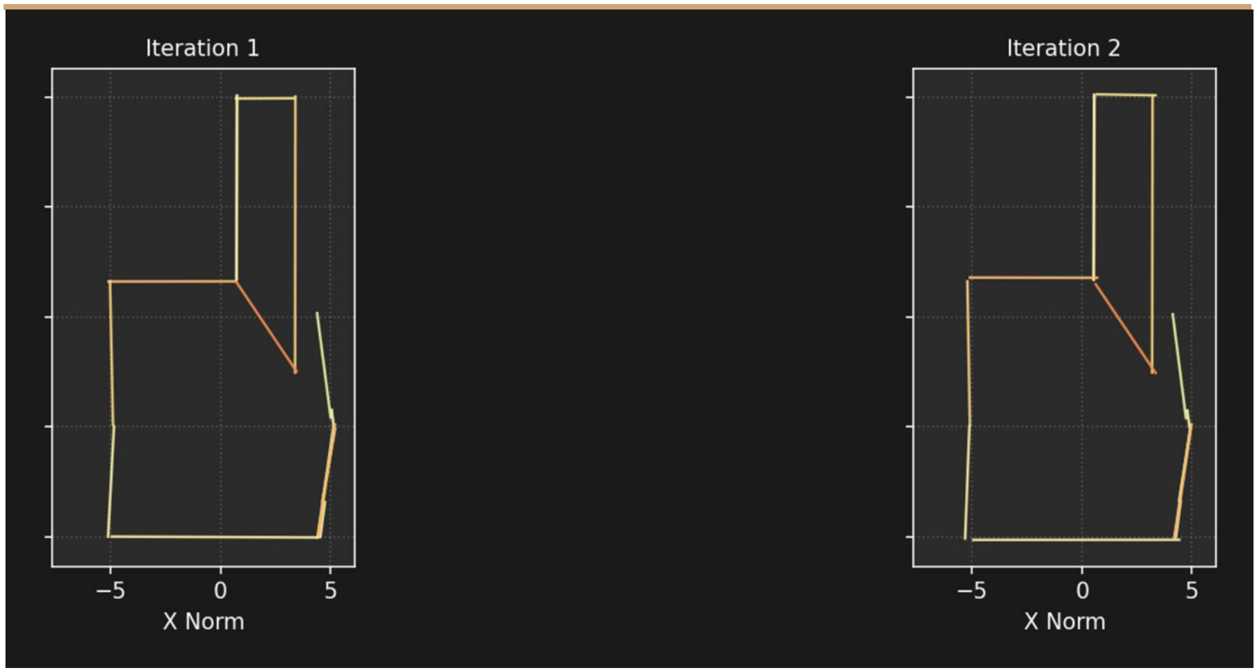
Figure 3 & 4: Training Evaluation Examples. The model demonstrates its ability to interpret heavily augmented inputs (left column) and produce high-quality predictions (middle column) that closely match the ground truth target (right column).

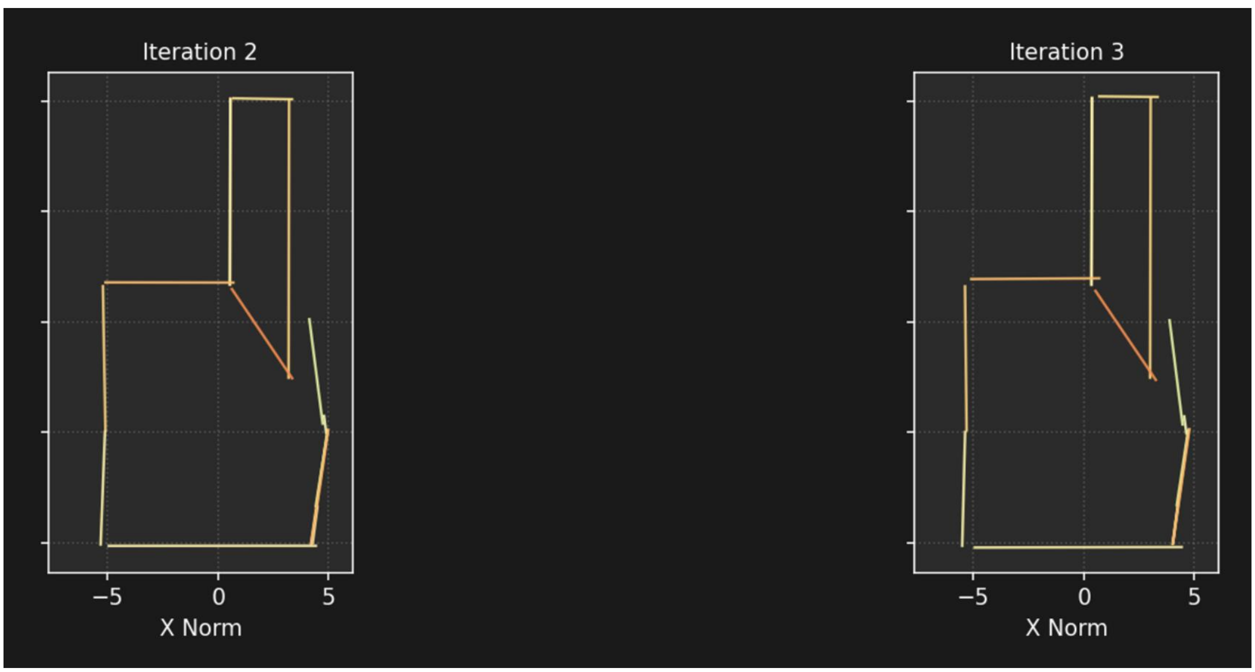
3.2. Iterative Refinement: A New Evaluation and a New Challenge

To push the model's capabilities further, an iterative refinement evaluation was conducted. Here, the model's output from one step is used as the input for the next, with the goal of progressively improving the result. The initial results of this experiment are mixed, revealing a new, more subtle challenge.

- **Initial Success:** The first iteration almost always produces a significant improvement over the initial noisy input, demonstrating the model's strong single-step denoising capability.
- **Subsequent Divergence:** However, in subsequent iterations (2, 3, and beyond), the model often fails to improve the result further. In many cases, it actively *diverges*, degrading the already-good output from the first iteration by introducing new errors and distortions (see Figure 5).




Iteration 1
Iteration 2
−5
0
5
X Norm
−5
0
5
X Norm


Iteration 2
Iteration 3
−5
0
5
X Norm
−5
0
5
X Norm

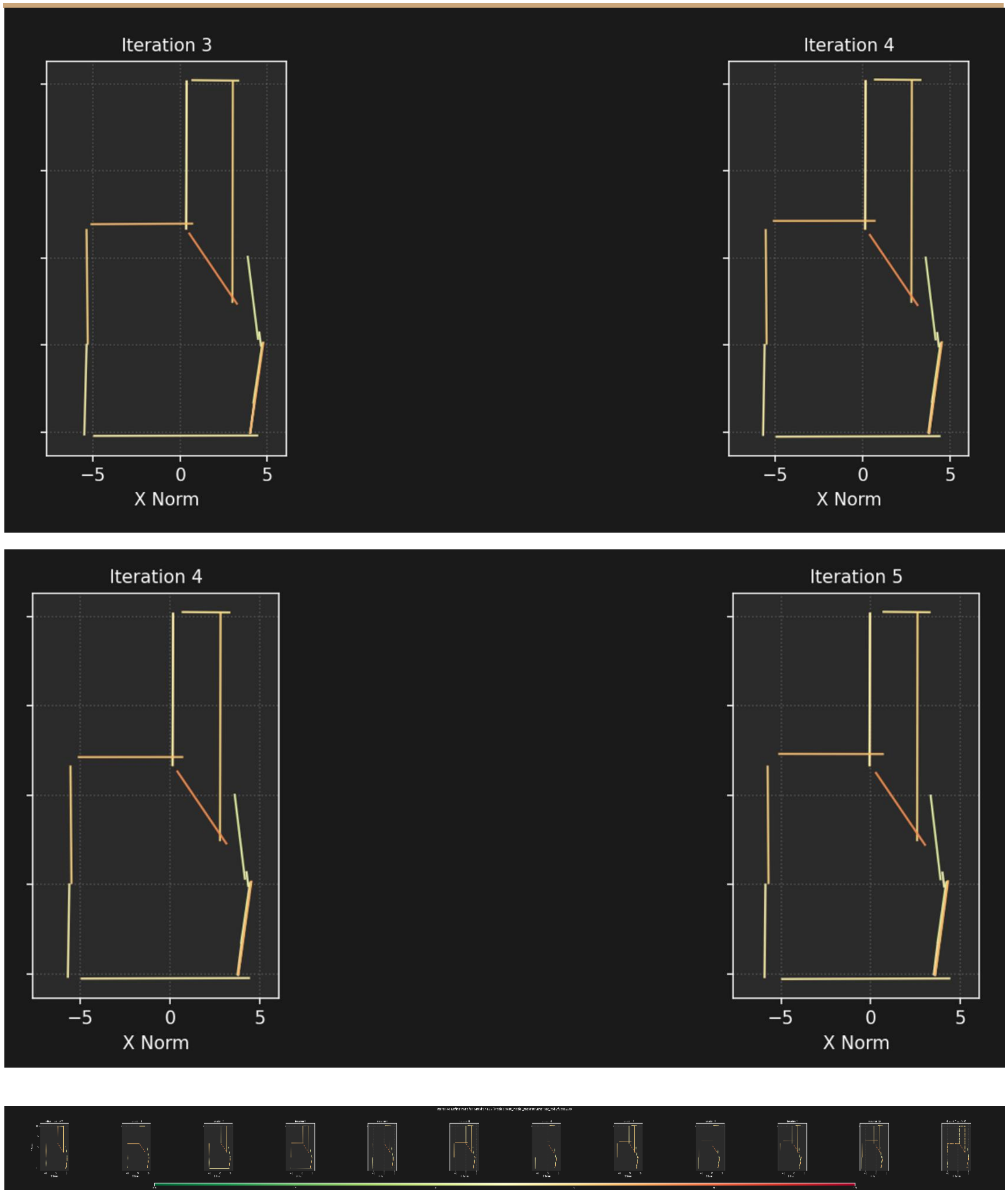


Figure 5: Iterative Refinement Visualization. The model starts with a very noisy input. The first iteration produces a much-improved graph. However, subsequent iterations fail to refine it further and begin to diverge, making the reconstruction worse.

3.3. Analysis of Divergence: The Static Confidence Problem

The leading hypothesis for this divergence is the **static nature of the input confidence scores**. The confidence score for each edge, which is a key input feature, is calculated only once from the initial noisy state. In the iterative loop:

1. **Iteration 1:** The model receives a very noisy graph and low confidence scores. It correctly performs a large, corrective step.
2. **Iteration 2:** The model receives a geometrically *cleaner* graph (the output of Iteration 1). However, it is still being fed the *original, low confidence scores*. This creates a critical mismatch. The model is effectively told, "This input is highly unreliable and noisy," when in fact the geometry is quite good. In response, it attempts to apply another large, corrective step, over-correcting the already-clean geometry and causing the observed divergence. The model lacks the mechanism to recognize that its own previous output is now a high-confidence input.

4. Future Work & Next Steps

The immediate priority is to solve the iterative divergence problem to unlock the model's full refinement potential. Our leading hypothesis is that providing a static, outdated confidence score actively misleads the model in later refinement steps. Therefore, the next phase of work will focus on testing this hypothesis by forcing the model to rely purely on geometric context.

1. **Establish a "Confidence-Agnostic" Baseline:** The primary next step is to conduct an experiment to determine if the confidence score is a net-negative for iterative tasks.
   - **Action:** A new model variant will be trained from scratch with the confidence_score feature completely removed from the input.
   - **Hypothesis:** By removing the potentially misleading static confidence signal, the model will be forced to infer the quality of the input graph solely from its geometric and topological properties. This may lead to more stable and robust behavior during the multi-step refinement process, as it will no longer be "told" that an already-clean graph is unreliable.
   - **Evaluation:** This confidence-agnostic model will be evaluated on both single-step reconstruction accuracy and, crucially, its performance in the iterative refinement task. We will directly compare its ability to converge to a stable solution against the divergence seen in the current model.
2. **Re-evaluate the Role of Confidence:** Based on the outcome of the confidence-agnostic experiment, we will decide on the future role of this feature.
   - **If Divergence is Solved:** If the new model performs well iteratively, it suggests that for refinement tasks, no confidence is better than wrong confidence. Future work would then focus on other improvements, like refinement-aware training.

- **If Performance Declines:** If the confidence-agnostic model has poor single-step accuracy or still diverges, it indicates that confidence is a necessary feature, but it must be dynamic. We would then proceed with the previously outlined strategies:

3. **Explore Refinement-Aware Training:** In parallel, we can investigate changes to the training objective itself. The current model is always trained to perform a full denoising step (e.g., $x_t \rightarrow x_0$). This may be too aggressive for gentle refinement. Future experiments could explore training the model on intermediate tasks (e.g., $x_t \rightarrow x_0$ with 10 or 20 steps)., which would explicitly teach it how to make the smaller, incremental improvements needed in later stages of an iterative chain.