3D Shape Generation Through Voxel Diffusion

Chealyfey Vutha
Brown University
chealyfey_vutha@brown.edu

November 27, 2025

Abstract

Procedural modeling offers a powerful means to generate infinite geometric variations, yet it lacks the semantic flexibility and learnability inherent to deep generative models. However, training such models for niche 3D domains, specifically complex calligraphic scripts like Khmer numerals, is often intractable due to the scarcity of large-scale volumetric datasets. To address this challenge, we propose a **Voxel Density Diffusion** framework that learns to mimic a procedural generator, compressing explicit procedural rules into implicit neural weights. We hypothesize that by training on continuous density fields rather than discrete binary voxels, a diffusion model can internalize smooth surface gradients and complex topological features, such as loops and curves. In our experiments, we demonstrate that our model captures the underlying manifold of the procedural generator, producing novel 3D Khmer numerals from pure noise. The generated shapes achieve high structural fidelity and surface smoothness without requiring massive 3D asset libraries or high-resolution grids.

1 Introduction

Can we train generative models to synthesize intricate 3D calligraphy, such as Khmer numerals? How about generating stylistic variations across rotations, scales, or shear transformations without losing the script's unique topological identity? While state-of-the-art generative algorithms have shown immense success on large-scale object categories (e.g., ShapeNet chairs or cars) [1, 2], the latter scenario remains challenging due to the scarcity of high-quality volumetric data for niche cultural domains. The dependence on massive 3D asset libraries makes learning specialized scripts impossible due to the high cost of digitization. Conversely, traditional procedural modeling offers infinite data but requires the manual, ad-hoc design of rigid geometric rules, which lack the semantic flexibility required for modern creative workflows.

We present a Voxel Density Diffusion framework, which utilizes a custom procedural generator to supervise a Denoising Diffusion Probabilistic Model (DDPM) in learning a continuous density field representation of Khmer numerals. We hypothesize that diffusion models, when trained on continuous density fields rather than discrete binary voxels, can internalize smooth surface gradients and complex topological features (such as loops and variable stroke widths) even at modest resolutions [3, 4]. This allows the model to behave both *faithfully* to the procedural constraints and *naturally* to surface smoothness, avoiding the *blocky* artifacts common in low-resource 3D generation. Our method is distinct in its direct operation on the spatial domain via density fields,

rather than employing Latent Diffusion Models (LDMs) [5, 6] which often sacrifice fine-grained topological fidelity for compression. By avoiding latent abstraction, our approach preserves the specific geometric integrity required for calligraphic scripts.

2 Methodology

2.1 Procedural Data Generation

To train the model, we developed a custom procedural pipeline that synthetically generates 3D volumetric training data.

- **Rasterization & Extrusion:** We rasterize Khmer glyphs from the *Kantumruy Pro* font into 64x64 2D images. These are extruded into 3D space with a depth of 16 voxels.
- **Density Fields over Binary Voxels:** A critical design choice was avoiding binary (0 or 1) voxel occupancy. Instead, we apply a 3D Gaussian filter ($\sigma = 1.0$) and a depth-based falloff factor ($1 - 0.3 \cdot \frac{|z-\text{center}|}{\text{half\_depth}}$). This creates a *soft* density field, providing the neural network with richer gradient information during training.
- **Augmentation:** To prevent the model from memorizing the limited font glyphs, we apply cubic-interpolated affine transformations, including rotation ($\pm 15°$), scaling ($0.9 \times - 1.1\times$), and shearing.

2.2 Architecture: 3D U-Net with Time Embeddings

We adapted the standard 2D U-Net architecture [7] for volumetric data [8]. The network takes a 1×64^3 voxel grid as input and progressively downsamples it to an 8^3 bottleneck using 3D convolutions and max pooling (channels: $32 \to 64 \to 128 \to 256$).

Crucially, the diffusion timestep t is injected into every residual block using sinusoidal position embeddings [9, 10] projected via an MLP. This allows the network to modulate its processing based on the noise level—focusing on coarse structure at high noise levels ($t \approx 50$) and fine detail at low noise levels ($t \approx 0$).

2.3 Training Strategy: The Two-Phase Curriculum

Training diffusion models from scratch on high-dimensional voxel grids is notoriously unstable. To mitigate this, we devised a two-phase training curriculum:

Phase 1: Autoencoder Pre-training (50 Epochs). We first trained the network to perform pure reconstruction ($t = 0$) without adding noise. This forces the model to learn the manifold of valid Khmer shapes before tackling the denoising task. The model achieved a strong baseline of near-perfect reconstruction (MSE $\approx 10^{-5}$) within just 8 epochs.

Phase 2: Diffusion Fine-tuning (2000 Epochs). We then introduced the diffusion process using a cosine noise schedule [11], which preserves signal longer than a linear schedule. We optimized a combined loss function:

$$\mathcal{L} = w_{\text{noise}} \cdot \text{MSE}(\epsilon_{\text{pred}}, \epsilon_{\text{true}}) + w_{\text{recon}} \cdot \text{MSE}(x_{\text{pred}}, x_0)$$

To ensure stability, we clipped the model's raw predictions to the range $[-0.95, 0.95]$ (using `tanh`) before calculating loss, preventing diverging gradients during the highly noisy stages of training.

3 Results and Analysis

3.1 Training Dynamics

The model demonstrated stable convergence, with the validation reconstruction MSE dropping from an initial 0.0053 (Epoch 1001) to a best of **0.00277** (Epoch 1584). Notably, the loss did not decrease monotonically; we observed fluctuations where the model traded off noise prediction accuracy for reconstruction fidelity, a common characteristic of adversarial-style training dynamics.

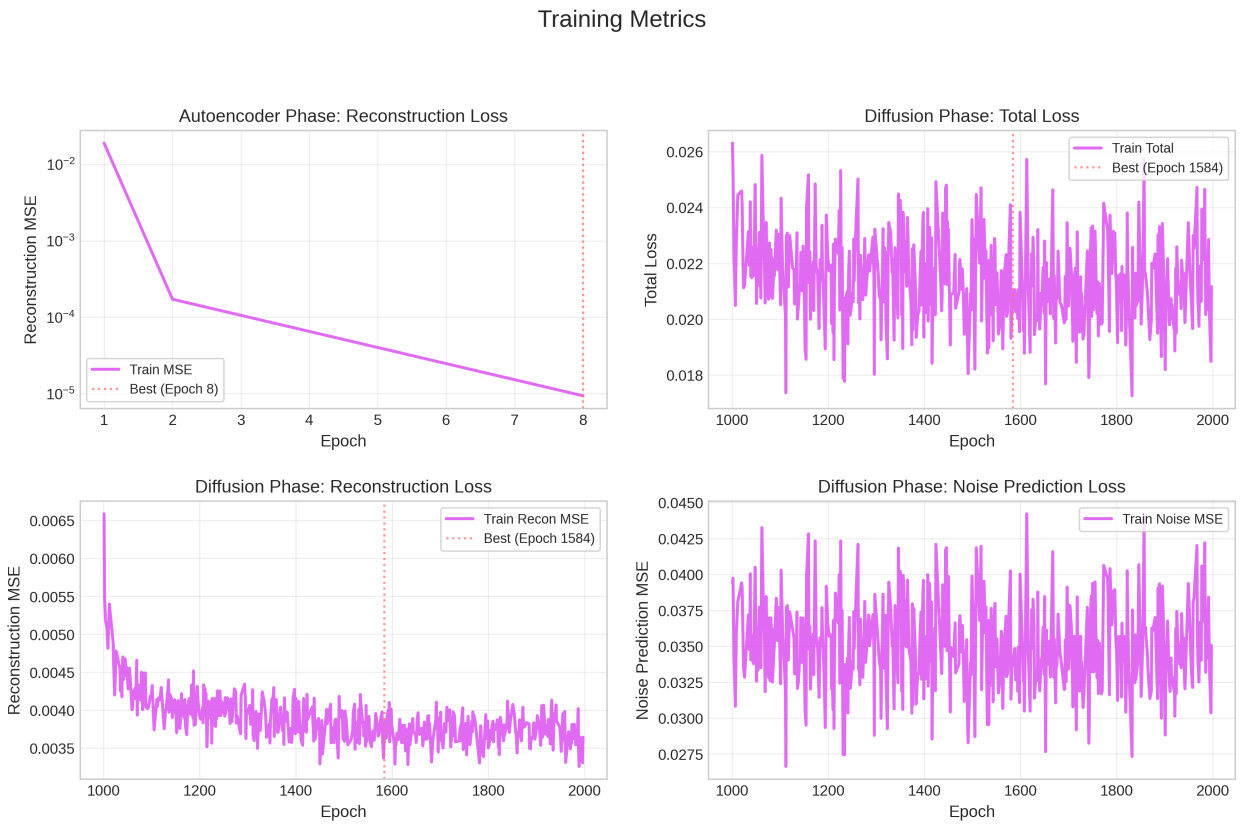

Figure 1: Training metrics during model training. (a) Autoencoder phase reconstruction loss (MSE) on training set. The model converged after 8 epochs with a final reconstruction MSE of 9.37×10^{-6}. (b) Diffusion phase total loss combining noise prediction and reconstruction objectives. (c) Diffusion phase reconstruction loss, measuring the model's ability to reconstruct clean 3D shapes from noisy inputs. (d) Diffusion phase noise prediction loss, the core denoising objective. Training was conducted for 2,000 epochs with a learning rate of 1×10^{-4} and batch size of 16.

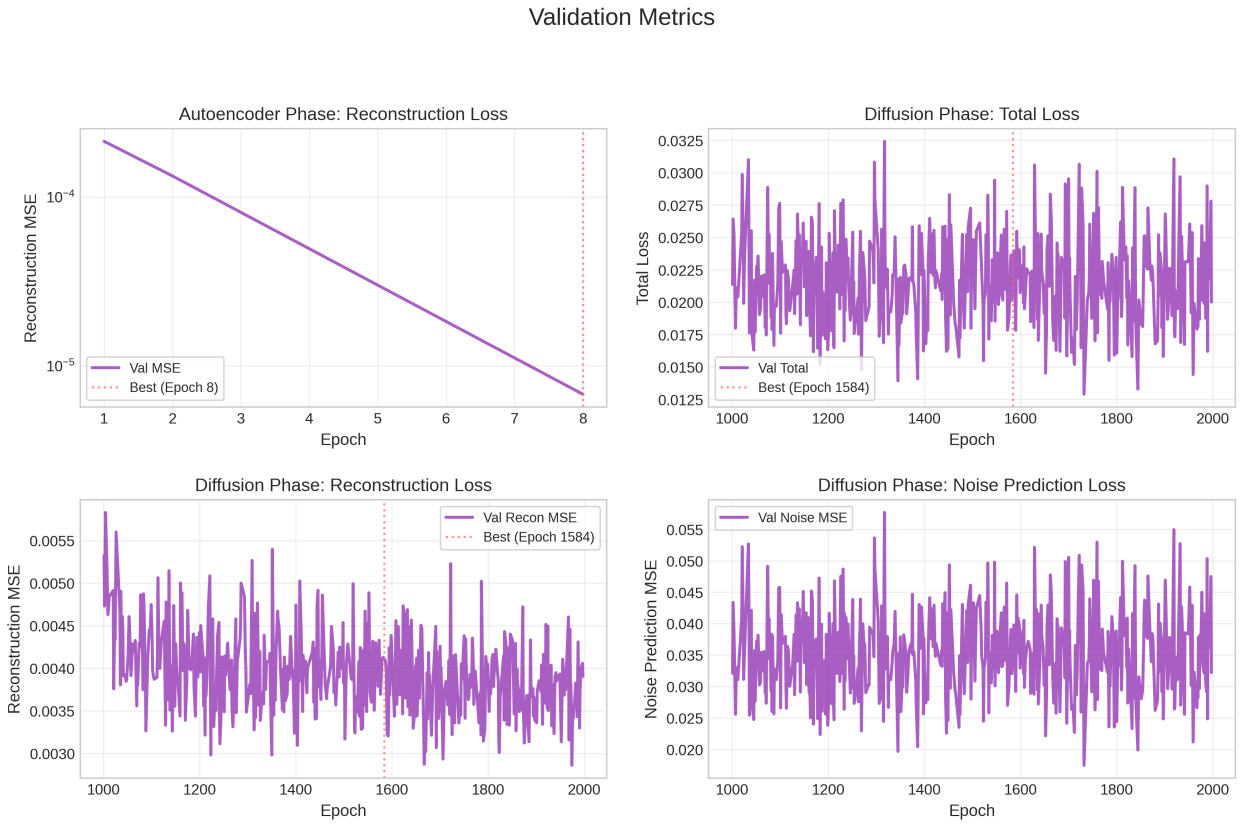

Figure 2: Validation metrics during model training. (a) Autoencoder phase reconstruction loss on validation set. (b) Diffusion phase total loss on validation set. (c) Diffusion phase reconstruction loss on validation set. The best model was selected at epoch 1,584 based on validation reconstruction MSE of 0.0028. (d) Diffusion phase noise prediction loss on validation set. The validation set consists of 400 samples (20 per numeral) from the Khmer numeral dataset.

3.2 Generation Quality

Using DDIM sampling with 50 steps, the model successfully generates novel 3D numeral shapes.

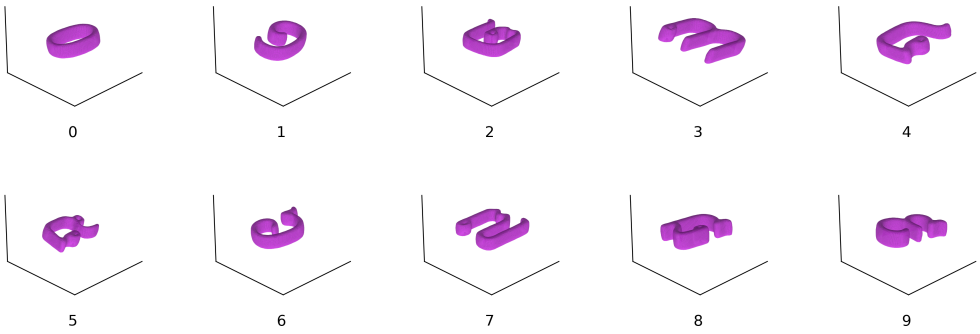

Figure 3: Gallery of generated 3D Khmer numerals (0–9). These samples were synthesized from pure Gaussian noise using the trained diffusion model.

- **Smoothness:** The decision to train on continuous density fields resulted in exceptionally smooth surfaces. The generated shapes lack the blocky artifacts common in binary voxel generation.
- **Diversity vs. Memorization:** We compared generated samples to their nearest neighbors in the training set (L2 distance ≈ 0.08). While the model captures the general style of the procedural generator, the low L2 distance suggests it is interpolating closely within the training distribution rather than extrapolating wild new forms. This is expected given the dataset size (2400 samples).

4 Conclusion

This work demonstrates that 3D diffusion models can effectively learn to mimic procedural generators, effectively compressing the procedural rules into neural weights. The resulting model produces high-quality, smooth volumetric shapes of Khmer numerals.

References

[1] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, H. Qixing, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, *et al.*, "ShapeNet: An information-rich 3D model repository," *arXiv preprint arXiv:1512.03012*, 2015.

[2] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall, "DreamFusion: Text-to-3D using 2D diffusion," in *International Conference on Learning Representations*, 2023.

[3] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "DeepSDF: Learning continuous signed distance functions for shape representation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 165–174, 2019.

[4] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *European Conference on Computer Vision*, pp. 405–421, Springer, 2020.

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

[6] H. Jun and A. Nichol, "Shap-E: Generating conditional 3D implicit functions," *arXiv preprint arXiv:2305.02463*, 2023.

[7] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, Springer, 2015.

[8] Ö. Çiçek, A. Abdulkadir, S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: learning dense volumetric segmentation from sparse annotation," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 424–432, Springer, 2016.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.

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

[11] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *International Conference on Machine Learning*, pp. 8162–8171, PMLR, 2021.