3D SDFusion Animal Shape Generation

Stanford CS231N Project Report

Yunong Liu
Department of Computer Science
Stanford University
yunongl@stanford.edu

Siqi Ma
Department of Statistics
Stanford University
siqima@stanford.edu

Grace Yang
Department of Statistics
Stanford University
yanggeer@stanford.edu

Abstract

This project aims to tackle the challenge of generating high-quality 3D animal shapes. The SDFusion model is designed to generate new 3D shapes with rich emergent abilities. This project utilizes the original model architecture from SDFusion and applies the model to a smaller dataset of 3D animals instead of furniture. The training has three components: VQ-VAE training, diffusion process training, and the decoding phase. These decoded shapes are the output of the model. Training experiments are run with different configurations, and we identified key impact factors of model performance and constraints. The VQ-VAE model achieves good representations of animal shapes, and the diffusion model starts to recognize animal feet despite challenges like vanishing gradient and noise scheduling. This model targets applications in computer graphics, virtual environments, biological research, and science education.

1. Introduction

In current practice, creating detailed 3D models of animals is often constrained by the need for extensive manual labor or the requirement of detailed 3D annotations. This limitation is highlighted in works such as [12], where the focus has been on developing methods that, while innovative, remain reliant on substantial manual input and are thus limited to a relatively narrow range of animal species and creative applications.

This project aims to address these challenges and generate animal shapes by using a novel framework, the SDFusion model proposed by [3], designed to generate high-quality 3D animal shapes. The SDFusion model generates new 3D shapes, which are not mere replicas of the input data but are new instances created based on the learned features and patterns from the training data.

During the VQ-VAE training phase, the SDFusion model encodes 3D animal shapes into a latent space, capturing the essential features and patterns of the shapes in the animal training dataset. This latent space representation allows the model to understand the intricate details and variations found in different animal shapes.

The diffusion process introduces noise to the latent representations and employs a denoising process to gradually refine these representations. This approach enables the model to generate new variations of the animal shapes by exploring the latent space in a controlled manner. By manipulating the latent space and introducing noise, the model is capable of creating a wide variety of animal shapes, including novel ones that were not present in the original training set.

During the decoding phase, the refined latent representations are decoded back into 3D animal shapes using the decoder part of the VQ-VAE. The output of this model is high-quality 3D animal shapes that can be used in various applications.

This innovative approach targets several applications in computer graphics, virtual environments, biological research, and science education. By automating the generation of 3D animal shapes, the SDFusion model significantly reduces the need for manual labor and detailed annotations, thereby expanding the range of species and creative possibilities that can be explored and utilized in these fields.

2. Related Work

Diffusion models have recently gained prominence as a leading class of generative models, known for their high-quality samples. These models have demonstrated remarkable performance in various domains such as image synthesis [4], super-resolution [9], image editing [2], and text-to-image generation [1]. While diffusion models have been extensively explored for 2D data, their application to 3D

data is still in its early stages. Notable efforts include the adaptation of diffusion models to point clouds [6], which highlights the potential for further research in this area.

SDFusion [3] applies diffusion models on Signed Difference Field representations [7] and introduces a pioneering framework designed to simplify the process of 3D furniture generation for non-expert users. The approach supports interactive generation by accommodating various input modalities, such as images, text, partially observed shapes, and their combinations. This flexibility allows users to easily provide input and adjust the influence of each type. Central to their method is an encoder-decoder structure that compresses 3D shapes into a compact latent representation, which is then used to train a diffusion model. The model incorporates task-specific encoders with dropout and a cross-attention mechanism to handle multi-modal inputs effectively. Due to its adaptable design, the model excels in several tasks, outperforming previous methods in shape completion, image-based 3D reconstruction, and text-to-3D generation. Their framework can seamlessly integrate these tasks, enabling users to generate shapes using a combination of incomplete shapes, images, and textual descriptions, while providing control over the relative importance of each input.

There has also been previous work in generating datasets for 3D meshes from 2D images. MagicPony [10] proposed a framework to learn articulated 3D animal shapes from single-view images in the wild, demonstrating the potential of 3D animal generation from a single image.

Our project leverages the SDF representation and diffusion-based generative modeling techniques from SDFusion paper to enable the generation of high-quality 3D animal shapes. We modify the implementation by fine-tuning the pre-trained SDFusion model on the Animal3D dataset [11] and adapting it to the animal domain.

3. Methods

3.1. 3D Shape Representation using SDF

We follow a systematic approach involving several key steps to preprocess 3D meshes using Signed Distance Fields (SDF) and create HDF5 files for subsequent training. SDF is a representation where each point in a 3D space stores the distance to the nearest surface of the shape, with the sign indicating whether the point is inside (negative) or outside (positive) the surface. This method offers several advantages over traditional point cloud representations, as it provides a continuous field that can represent surfaces more accurately and handle complex geometries effectively. Using SDF allows for better interpolation and integration with various machine learning models, especially in tasks requiring high-resolution shape representation.

Firstly, we modify the provided script to process the 3D mesh data from the animal dataset. The script begins by normalizing the meshes using the trimesh library to ensure consistent scaling and centering. Next, we generate SDF values for the 3D shapes, representing the distance from each point in a volumetric grid to the nearest surface of the mesh. This process involves sampling the 3D space and interpolating the SDF values to create a high-resolution representation of the shape. To manage computational complexity and storage, the resolution of the SDF grid is reduced, compressing the 3D data into a more compact form.

The SDF values and associated parameters are then saved into HDF5 files for efficient storage and retrieval. These files contain the original point cloud, sampled SDF values, normalization parameters, and the SDF parameters. This structured and compressed format facilitates efficient data handling and is suitable for subsequent modeling tasks such as shape synthesis and texture mapping.

3.2. VQVAE for dimensional reduction

We then use a 3D-variant of the Vector Quantised-Variational AutoEncoder (VQ-VAE) [8] to encode the SDF into a lower-dimensional latent space, making it feasible to apply diffusion models in subsequent steps. This enables the application of diffusion models in a lower-dimensional space. Specifically, the 3D VQ-VAE includes an encoder E_ϕ to encode the 3D shape into the latent space and a decoder D_τ to decode the latent vectors back to 3D space. Given an input shape represented by the SDF $X \in \mathbb{R}^{D\times D\times D}$, we have:

$$z = E_\phi(X)$$
$$X' = D_\tau(VQ(z))$$

where $z \in \mathbb{R}^{d\times d\times d}$ is the latent vector, the latent dimension d is smaller than the 3D shape dimension D, and VQ is the quantization step mapping the latent variable z to the nearest element in the codebook Z. The encoder E_ϕ, decoder D_τ, and codebook Z are optimized jointly. We pretrain the VQ-VAE using reconstruction loss, commitment loss, and the VQ objective using the Animal dataset as described in section 4.

3.3. 3D Difussion model for SDF

A diffusion model is then trained over the latent space to manage the generative process, providing a method to transition from a noise distribution to a structured 3D output.

Using the trained encoder E_ϕ, we can encode any given SDF into a compact and low-dimensional latent variable $z = E_\phi(X)$. This allows us to train a diffusion model on this latent space. Essentially, a diffusion model learns to sample from a target distribution by reversing a progressive noise addition process. Starting with a sample z, we produce z_t for $t \in \{1, \ldots, T\}$ by gradually introducing Gaussian noise following a specific variance schedule. For the

denoising step, we employ a time-conditional 3D UNet ϵ_θ. The training of this denoising 3D UNet is guided by the simplified objective proposed by [5]

$$L_{\text{simple}}(\theta) := \mathbb{E}_{z,\epsilon\sim N(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t)\|^2\right].$$

During inference, we generate $\hat{z}$ by iteratively denoising a variable initially sampled from the standard normal distribution $N(0, 1)$. We then use the trained decoder D_τ to convert the denoised code $\hat{z}$ back into a 3D SDF shape representation $\hat{X} = D_\tau(\hat{z})$.

4. Dataset and Features

Our approach will leverage the Animal3D dataset [11], which provides 3D scans, meshes, and annotations for 40 common quadruped species. This dataset is particularly valuable because it includes a wide range of diverse poses and anatomical features, allowing us to develop and test our models on realistic and varied animal shapes. The dataset's comprehensive coverage of different species and postures makes it an excellent resource for creating robust and generalizable models.

Additionally, the dataset statistics shown in Table 1 show the breakdown of the training and testing sets across all species. The training set includes an average of 76.62 samples per species with a standard deviation of 107.44, totaling 3065 samples. The test set, on the other hand, includes an average of 8.00 samples per species with a standard deviation of 11.98, totaling 320 samples. This distribution provides a substantial amount of data for training while maintaining a reasonable test set size for evaluating model performance.

By utilizing the Animal3D dataset, we aim to enhance our 3D shape synthesis and modeling capabilities, ultimately contributing to more accurate and flexible 3D representations of quadruped animals. The rich variety of the dataset ensures that our models can learn from a broad spectrum of shapes and poses, improving their ability to generalize to unseen data. However, the dataset is very small compared to the variety of data used in the original SDFusion paper.

Dataset	Average	Standard Deviation	Total
Train	76.62	107.44	3065
Test	8.00	11.98	320

Table 1. Training and Testing Data Size for Each Species

5. Results

During data preprocessing phase, we normalized the 3D mesh data, calculated the Signed Distance Functions (SDF) for each mesh, and converted these into formats suitable for neural network processing. Following this, the data was stored in HDF5 files, ensuring efficient access during training. Building on this foundation, we proceed to train the Vector Quantized Variational AutoEncoder (VQ-VAE) model. This training phase was focused on enabling the VQ-VAE to accurately reconstruct animal shapes from the processed data inputs.

5.1. VQVAE Results

We trained the VQ-VAE network to extract lower dimensional feature representation of the 3D models for 9000 steps based on the checkpoint provided by the SDFusion paper with batch size 2. The reconstructed mesh (Figure 1) from one batch of the testing dataset and ground truth mesh Figure 2 demonstrate the VQ-VAE model's success in capturing and reproducing the complex geometries of various animal forms. The reconstruction error achieved by the VQ-VAE is 0.003941.

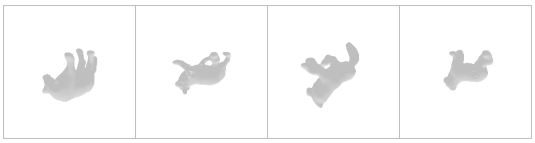

Figure 1. Reconstructed Mesh

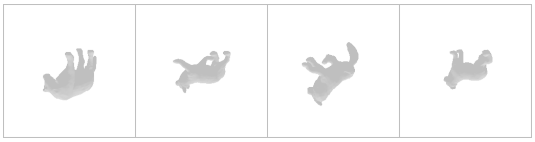

Figure 2. Ground Truth Mesh

5.2. Diffusion Model Performance

5.2.1 Diffusion model trained from scratch

In one attempt, we train VQ-VAE and SDFusion from scratch. While the VQ-VAE training yielded satisfactory results, the performance of the diffusion model was suboptimal. We train the diffusion model for 7500 steps with batch size 2. The generated shapes from the diffusion model in Figure 3 do not resemble reasonable animal forms, indicating that the model struggles to learn the distribution of the latent space effectively.

5.2.2 Finetune Diffusion model pretrained on furniture datasets

We suspect that the poor performance of the diffusion model was due insufficient data. To address this, we used the provided stable diffusion checkpoint trained on furniture datasets and fine-tuned it further on comparatively smaller

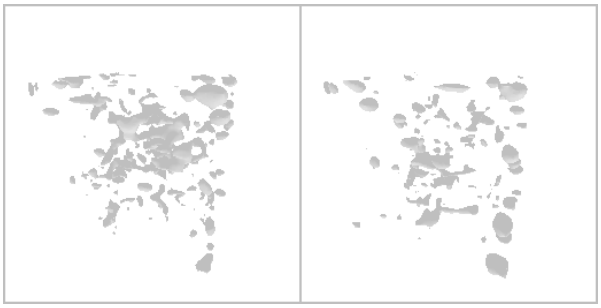

Figure 3. Generated Mesh from Diffusion Model trained from scratch

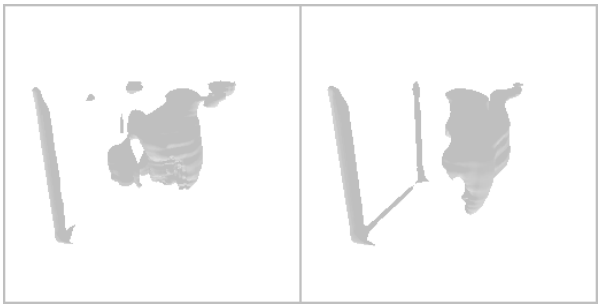

Figure 5. Generated Mesh from finetuned Diffusion Model at step 5000

animal dataset. Additionally, we tested the original VQ-VAE checkpoint on our dataset and compared the reconstruction loss with our current model to rule out any issues in the VQ-VAE model. The output of the fine-tuned diffusion model, while still suboptimal, shows a noticable improvement compared to earlier iterations. Between steps 4000 and 6000, shapes resembling animals start to emerge, indicating some level of pattern recognition Fig 4 and Fig 5. However, this trend dissipates as training progresses, with the model eventually generating nonsensical shapes. The loss also starts to rise back to 1 after 8000 steps and stays high. We hypothesized that this issue arises due to the aggressive denoising steps employed during the process and inappropriate learning rate for UNet training due to reduced batch size. To address this, we adjust relevant hyperparameters during training. Specifically, we modify the scheduler that controls the magnitude of the noise added to the data and reduced the learning rate to better accommodate smaller batch size. These adjustments aim to refine the model's ability to consistently generate coherent and meaningful outputs.

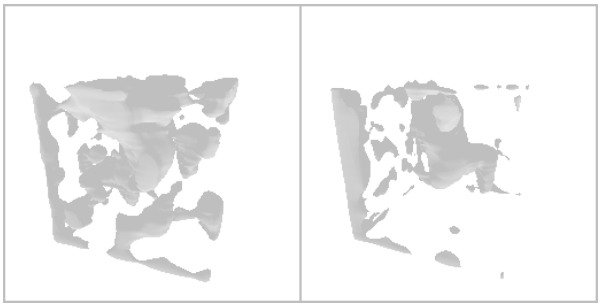

Figure 4. Generated Mesh from finetuned Diffusion Model at step 4500

5.2.3 Finetune diffusion model with modified hyperparameters

In training a 3D diffusion model, the Noise Scheduler plays a crucial role in controlling the noise addition process during the diffusion and denoising phases. While the original paper employs a linear scheduler, cosine scheduler appears to perform better than a linear one.

Cosine Noise Scheduler Start defines the initial amount of noise added to the latent variables at the beginning of the diffusion process. We start with a small noise level to ensure that the model begins its training from a relatively clean and less noisy state, allowing it to learn the structure of the 3D data effectively.

Cosine Noise Scheduler End determines the final amount of noise added to the latent variables at the end of the diffusion process. We gradually increasing the noise level to this final value enables the model to learn to handle and denoise progressively noisier inputs, which is essential for the model to generalize well and perform robust denoising during inference.

Cosine Noise Scheduler Steps specifies the total number of steps over which the noise level transitions from the start value to the end value. The steps indicate the number of increments in the noise level, following a cosine schedule. A cosine schedule smoothly transitions the noise level, starting slowly, accelerating in the middle, and then decelerating towards the end. This smooth transition helps in stabilizing the training process and ensures that the model gradually adapts to increasing levels of noise, improving its denoising capabilities.

Cosine Noise Scheduler with its Start, End, and Steps parameters orchestrates how noise is added during the training of the 3D diffusion model. This controlled noise addition is vital for teaching the model to effectively reverse the noise process, leading to accurate generation and reconstruction of 3D shapes from noisy inputs.

Our best SDFusion model is trained using the following hyperparameters shown in Table 2 : VQ-VAE is trained from the provided checkpoint pretrained on furniture datasets for 9000 steps. Diffusion model is trained from the provided checkpoint for 25000 steps. The generated output is shown in Figure 6 and Figure 7. The generated shapes exhibit a somewhat recognizable structure, such as four discernible legs, but they are crude and lack fine details.

In addition to unconditional generation, we evaluated the diffusion model's capabilities in shape completion. The model was given partial shapes with missing parts and tasked with generating the missing portions to complete the shape. This task is particularly challenging as it requires the model to understand and infer the context and structure of the partial input to produce coherent and realistic completions. Both tasks were designed to comprehensively evaluate the model's versatility and effectiveness in generating high-quality 3D shapes under different scenarios. The ground truth of one sample is given in Figure 8, and the generated outputs for completion are shown in Figure 9. We can discern abstract shapes resembling heads and legs in the generated outputs; however, they lack fine details and exhibit discontinuities in the overall structure.

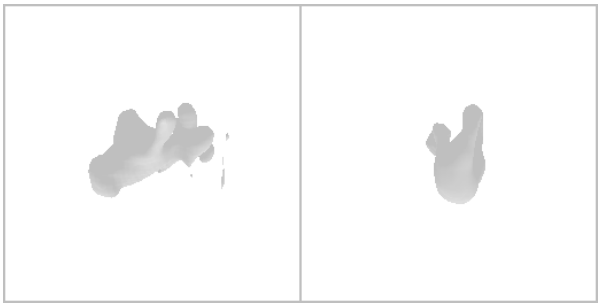

Figure 6. Generated Mesh from finetuned Diffusion Model at training step 17500 with modified hyperparameters

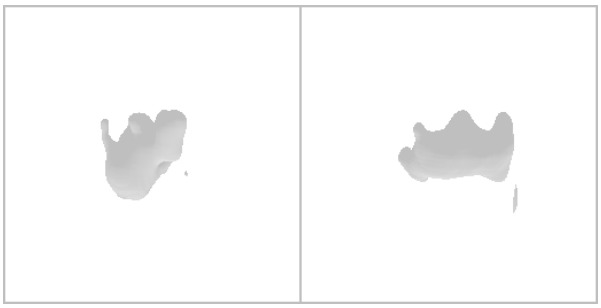

Figure 7. Generated Mesh from finetuned Diffusion Model at testing step 17500 with modified hyperparameters

Hyperparameter	Value
Learning Rate	$2e-5$
Batch Size	2
Cosine Noise Scheduler Start	$1e-4$
Cosine Noise Scheduler End	$2e-2$
Cosine Noise Scheduler Steps	$1e3$

Table 2. Configuration and Hyperparameters for Training

Figure 8. Shape completion ground truth

Figure 9. Shape completion using step 17000 checkpoint

5.2.4 Gradient Vanishing problem

However, we observe that the 3D diffusion model experiences severe vanishing gradient problems and the violin plot of gradients versus step size is shown in Figure 10. We examine our model to identify potential modifications. The input data is normalized. For the activation function, we used siLu as activation to prevent the vanishing gradient problem intentionally. Since the batch size is limited to 2 due to GPU memory constraints, batch normalization is not possible. The original architecture already deployed residual blocks. In addition to these modifications, we propose the vanishing gradient problem could be due to several factors.

3D data is inherently high-dimensional. When modeling 3D shapes, especially in high resolutions, the number of parameters increases significantly, this increased complexity can exacerbate the vanishing gradient problem because

the gradient signal diminishes as it propagates through the many layers required to process such data. To capture the intricacies of 3D shapes, deep networks are required. The deeper the network, the more likely it is to suffer from vanishing gradients. As gradients are backpropagated through many layers, they can become exceedingly small, making it difficult to update the weights effectively during training. The complexity and high dimensionality of 3D data can lead to unstable training dynamics. This instability can result in vanishing gradients , making it challenging to find a stable solution during the training process.

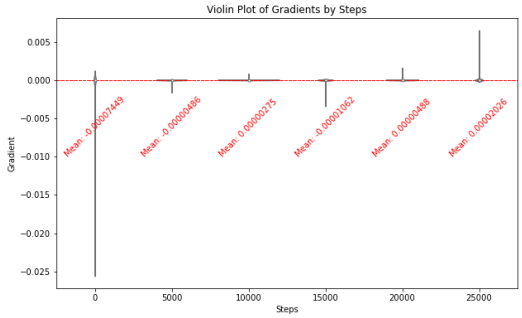

Figure 10. Gradient Vanishing Plot

5.2.5 Training Curves

Figure 11 shows different loss components including embedding loss, nagative log likelihood, reconstruction loss and the total loss of the two VQ-VAE model (one trained from scratch and the other trained from the provided checkpoint) as training progresses. This plot illustrates the training dynamics of the VQ-VAE models and highlights the benefits of using a pre-trained checkpoint for further training. Both models display fast convergent to local optimal.

Figure 12 shows different loss components including simple loss and total loss of two diffusion models (one trained from scratch and the other trained from the provided checkpoint) as training progresses. There are occasional spikes in the loss values throughout the training, which could be due to the stochastic nature of the training process or specific challenging batches of data. Both models follow similar trends, with the checkpoint model generally showing slightly lower or comparable loss values compared to the scratch model. This suggests that starting from a checkpoint might offer a slight advantage in terms of stability and convergence.

5.3. Evaluation

5.3.1 VQVAE Model

To evaluate the performance of the VQVAE model on the testing set, Intersection over Union (IoU) is used to describe

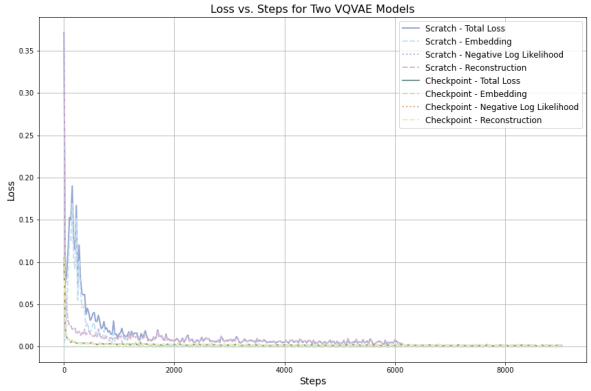

Figure 11. Different Loss Curves Versus Training Steps for Two VQVAE Model on Training Data

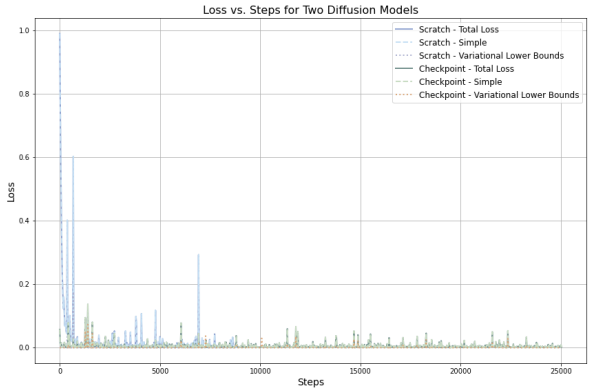

Figure 12. Different Loss Curves Versus Training Steps for Two Diffusion Model on Training Data

how much the model captures the original features from the mesh. A line plot with standard error bars for the two models is displayed in Figure 13. The figure shows the consistent result as the loss curve, indicating that training from checkpoint achieves a better performance.

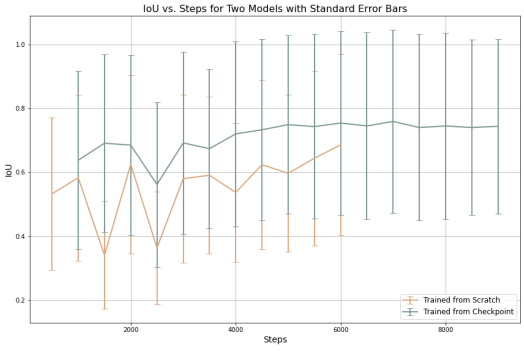

Figure 13. IoU plot for VQVAE Model

5.3.2 Diffusion Model

We use Uni-Directional Hausdorff Distance(UHD) and the Trimmed Mean Distance (TMD) to compare the similarity between a generated shape and a ground truth shape.

UHD is a variation of the Hausdorff Distance that measures the maximum distance from a point in one shape to the nearest point in another shape, but only in one direction. Typically, this metric is used to assess how far the points of a generated shape are from the points of a reference shape (or vice versa), providing a sense of the worst-case discrepancy in one direction. Given two sets of points A and B, the unidirectional Hausdorff distance from A to B is defined as:

$$\text{UHD}(A \to B) = \max_{a \in A} \min_{b \in B} \|a - b\|$$

In the context of evaluating shape generation quality, UHD can reveal how well the generated shape covers the ground truth shape. A high UHD value indicates that there are parts of the generated shape that are far from the reference shape, suggesting poor quality in those regions.

TMD is a metric used to measure the average distance between points in two shapes after excluding a certain percentage of the most extreme values. This trimming process reduces the influence of outliers or noise, providing a more robust and representative measure of similarity between the shapes.

Together, UHD and TMD provide complementary insights into the quality of shape generation. UHD ensures that significant deviations are minimized, while TMD gives a robust measure of overall similarity, making them valuable tools in evaluating and improving the performance of shape-generation algorithms. The evaluation result is shown in Table 5.3.2. A better model should have smaller UHD and larger TMD.

Metric	Value
UHD ↓	137.3643
TMD ↑	0.02835

Table 3. Evaluation Results for Shape Generation Quality

6. Discussion

The experiments suggested several key factors in training the VQVAE network and 3D diffusion network. Firstly, the start and end of the noise scheduler of the diffusion model impact training more than the scheduler type. An appropriate learning rate reduced proportionally to the batch size is the main impact factor for training a diffusion model. The effect of inappropriate hyperparameters can be identified early in the training stage with the help of learning curves. While the VQVAE model will achieve the same level of loss quickly regardless of whether training from scratch or from existing checkpoints, these two approaches converge to a different level of testing phase IoU. This suggests that the VQVAE model will certainly benefit from a larger sample size, while the training loss does not fully capture this benefit.

In the following sections, we will analyze the reasons behind the unsatisfactory performance of the SDFusion model on the animal dataset. The model's inability to maintain coherence in generated shapes indicates underlying issues. By examining potential factors contributing to this suboptimal performance, we aim to identify key areas for improvement and propose strategies to enhance the model's ability to generate accurate and meaningful representations of animal shapes.

6.1. Insufficient scale of the dataset

One possible reason for the unsatisfactory performance of the SDFusion model in generating 3D animal shapes is the lack of diversity and quantity in the animal mesh dataset used for training. The dataset may not provide a broad enough range of examples to enable the diffusion model to learn a meaningful distribution in the latent space. In contrast, the model in the original paper was trained on a significantly larger dataset sourced from multiple domains, which likely contributed to its superior performance. Consequently, the results of our model are not directly comparable to those of the original, highlighting the critical importance of extensive and varied training data in achieving high quality generative outputs.

MagicPony [10] proposed a framework to learn articulated 3D animal shapes from single-view images in the wild, demonstrating the potential of 3d animal generations from a single image. One future direction is to utilize the techniques introduced in MagicPony to generate 3D meshes as a data augmentation step for 3D animal shape generation. MagicPony's approach, which effectively predicts the 3D shape and viewpoint from single-view images, will enhance the Animal3D dataset by providing articulated 3D meshes. By incorporating these meshes, we expect to achieve superior performance in reconstructing 3D animal shapes from limited input data.

6.2. Complexity of animal shape topology, distribution shift

Another factor contributing to the unsatisfactory performance of the SDFusion model in generating 3D animal shapes is the intricate and varied geometries of animal forms. These complex structures may pose significant challenges for the diffusion model to capture and generate coherent shapes.

Additionally, there is a domain shift from furniture to animals. The SDF representation that works well with simpler furniture shapes may not transfer effectively to the more

complex geometries of animals. To address these issues, we can explore different representations in future work. For example, we might consider using voxel grids, which offer a more uniform approach to representing 3D space. Mesh-based representations, which explicitly model the surfaces of objects, could also provide a more accurate and detailed way to generate intricate animal shapes. Exploring these alternative representations may help improve the model's ability to generate coherent and realistic 3D animal shapes.

6.3. Time constraints and computational resources

The original SDFusion paper reported extensive training times, with the VQ-VAE being trained for 7 days and the diffusion model for 7-14 days on a single GPU with 32 to 48 GB memory. Given our limited time and computational resources, reproducing these results from scratch proved challenging. Our shorter training duration may have led to an insufficient convergence of the models, which is likely a key factor in the unsatisfactory performance observed. The lack of prolonged and extensive training prevents the models from fully learning the intricate patterns and distributions necessary to generate coherent 3D animal shapes. Future work should consider either securing additional computational resources or optimizing the training process to achieve better results within the given constraints.

The insights gained from this analysis will guide our decision-making process and help us prioritize our efforts in improving the quality of the generated 3D animal shapes.

References

[1] O. Avrahami, D. Lischinski, and O. Fried. Blended Diffusion for Text-Driven Editing of Natural Images. pages 18208–18218, 2022. 1

[2] S.-I. Cheng, Y.-J. Chen, W.-C. Chiu, H.-Y. Tseng, and H.-Y. Lee. Adaptively-Realistic Image Generation from Stroke and Sketch with Diffusion Model, Aug. 2022. 1

[3] Y.-C. Cheng, H.-Y. Lee, S. Tulyakov, A. Schwing, and L. Gui. Sdfusion: Multimodal 3d shape completion, reconstruction, and generation, 2023. 1, 2

[4] P. Dhariwal and A. Nichol. Diffusion Models Beat GANs on Image Synthesis, May 2021. 1

[5] J. Ho, A. Jain, and P. Abbeel. Denoising Diffusion Probabilistic Models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. 3

[6] S. Luo and W. Hu. Diffusion Probabilistic Models for 3D Point Cloud Generation, June 2021. arXiv:2103.01458 [cs]. 2

[7] A. Marić, Y. Li, and S. Calinon. Online Learning of Continuous Signed Distance Fields Using Piecewise Polynomials, May 2024. arXiv:2401.07698 [cs]. 2

[8] A. v. d. Oord, O. Vinyals, and K. Kavukcuoglu. Neural Discrete Representation Learning, May 2018. arXiv:1711.00937 [cs]. 2

[9] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi. Image Super-Resolution via Iterative Refinement, Apr. 2021. 1

[10] S. Wu, R. Li, T. Jakab, C. Rupprecht, and A. Vedaldi. Magicpony: Learning articulated 3d animals in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8792–8802, June 2023. 2, 7

[11] J. Xu, Y. Zhang, J. Peng, W. Ma, A. Jesslen, P. Ji, Q. Hu, J. Zhang, Q. Liu, J. Wang, W. Ji, C. Wang, X. Yuan, P. Kaushik, G. Zhang, J. Liu, Y. Xie, Y. Cui, A. Yuille, and A. Kortylewski. Animal3d: A comprehensive dataset of 3d animal pose and shape, 2024. 2, 3

[12] S. Zuffi, A. Kanazawa, and M. J. Black. Lions and tigers and bears: Capturing non-rigid, 3d, articulated shape from images. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 3955–3963, 2018. 1