2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

UM3D: Towards a Unified Multimodal 3D Shape Generation Model

Xian-Feng Han[1] and Zecheng Zhang[1] and Xuran He[1] and Ming-Jie Wang [2]

***Abstract*— Vision-Language Pre-training models (VLMs) have emerged as a highly promising solution to the generative problem, achieving remarkable success in the field of 2D image generation. However, extending these 2D paradigms to 3D domains is still unexplored due to the scarcity of text-3D pairs and shape ambiguity. To address this challenge, we introduce UM3D, a two-stage pre-training architecture towards unified multimodal 3D shape generation. Our approach first optimizes a Finite Scalar Quantization based Autoencoder (FSQ-AE) to learn a compact yet powerful implicit representation with improved codebook utilization. We then encode sketch features into CLIP's multimodal embedding space to incorporate additional geometric information. This unified space conditions our well-designed Instance-Normalized Glow model (Glow-IN) to model the distribution of 3D shape representations while mitigating distribution shift issues. During inference, UM3D can accept individual text, image, sketch, or combined inputs to generate corresponding 3D shapes. Quantitative and qualitative evaluations confirm our method's effectiveness in synthesizing high-fidelity, input-consistent 3D geometries.**

I. Introduction

The primary objective of 3D shape generation is to synthesize diverse and realistic 3D object contents using manual [1] or automatic [2] methods, which serves as a cornerstone in a broad range of applications, such as Computer Aided Design (CAD) [3], entertainment [4] [5], gaming [6][7], robotics [8][9] and virtual reality [10][11]. Traditionally, manual creation of 3D shapes has been recognized as a time-consuming and labor-intensive task [12], typically requiring skilled designers proficient in specialized modeling software (e.g., Blender) to perform the generation process. Hence, there has been a growing focus on exploring automatic generation pipelines built on deep generative models. However, most of these methods often struggle to produce desired objects [13].

Recently, significant advancements in Vision-Language Pre-trained Models (VLMs) have revolutionized various tasks of 2D computer vision [14], especially opening up a new solution for high-fidelity 2D image content generation. By leveraging large-scale image-text pairs collected from the Internet to establish connections between vision and language modalities, VLMs have driven the success of text-guided image creation. This process, which aims to synthesize images semantically related to input text prompts or language descriptions [15] [16], has become a compelling research field, showcasing remarkable achievements.

This naturally raises a fundamental question: Can the text-to-2D paradigm be effectively extended to 3D domains? This proves to be a challenging problem due to the difficulty of collecting and annotating large-scale, high-quality text-3D paired data, which is time-consuming and labor-intensive. Recent studies [17][18] have leveraged the knowledge of 2D VLMs for 3D generation by using images as a bridge between text and 3D shapes. However, textual descriptions often fail to provide complete geometric information due to inherent ambiguities, inaccuracies, or missing details, leading to geometrically inconsistent 3D shapes.

To overcome these challenges, we learn a unified representation that integrates sketches, images, and texts to leverage their complementary strengths: texts provide semantic information, images capture appearance, and sketches control the geometry of shapes. Building upon this foundational concept, we propose a two-stage 3D shape generation architecture. This framework consists of a Finite Scale Quantization based Autoencoder for learning compact yet robust 3D representation, and a sketch-enhanced CLIP-Guided Instance Glow Model. The latter initially embeds sketch features into CLIP's visual-linguistic multimodal space via a contrastive learning strategy, subsequently conditioning on the Instance Normalization Glow framework to model the distribution of 3D representations. During inference, our model can accept image, sketch, text, or their combinations as input to produce high-fidelity 3D shapes, showcasing the effectiveness and superior performance of our proposed UM3D.

Our paper's contributions can be summarized as follows: (1) We present UM3D, a unified 3D generation model that is capable of processing individual text, image, sketch, or multimodal combination to generate 3D shapes of high visual quality while maintaining consistency with the given inputs. (2) We introduce an FSQ-based autoencoder framework to achieve 3D representations with high utilization. Subsequently, we present a Sketch-Enhanced CLIP-Guided Instance Glow Model, which aligns sketch features with the CLIP vision-language embedding space. This joint latent space serves as a guiding condition for modeling the shape representation distribution through our Instance Normalization Glow Model. (3) We perform extensive experiments under various settings, including zero-shot text-to-3D shape generation, multimodal-conditioned 3D shape generation, and novel-category 3D shape generation, to evaluate the efficacy of our proposed method.

*This work is supported by the Fundamental Research Funds for the Central Universities (SWU-KT24003)

[1]Xian-Feng Han, Zecheng Zhang and Xuran He are with Southwest University, Chongqing, China. xianfenghan@swu.edu.cn {nzdbgyx, dinosaur}@email.swu.edu.cn

[2]Ming-Jie Wang is with the Department of Mathematics, School of Science, Zhejiang Sci-Tech University, Hangzhou 310018, China mingjiew@zstu.edu.cn.

II. Related Work

Text-to-Image Generation. Recent years have witnessed extensive research on text-guided image synthesis. Early approaches have used Generative Adversarial Networks (GANs) [19] for domain-specific text-to-image generation [20], while other works have achieved impressive performance in creating corresponding images from text [21]. To eliminate textual dependency, Wang et al. [22] and Zhou et al. [23] have explored zero-shot text-to-image generation using Contrastive Language-Image Pre-Training (CLIP). Recent advancements in scalable generative architectures and large-scale text-image datasets [24] further enable high-fidelity zero-shot text-to-image synthesis.

3D Shape Representation. Existing studies on 3D shape generation explore diverse 3D representations. Voxel grids [25][26] employ a discrete volumetric structure, dividing the shape into uniform voxels that store attributes such as occupancy, density, or color. Their structured nature facilitates efficient convolutional operations for 3D shape synthesis. Point clouds [27][28] represent objects as unordered sets of 3D points, optionally with normals or colors, but require specialized network architectures for processing. Meshes [29], [30] define surfaces through vertices, edges, and faces, although they exhibit limited flexibility in modifying topology. Signed Distance Functions (SDFs) [31], [32] implicitly represent shapes as continuous functions that assign signed distances to the nearest surfaces. However, learning SDFs for complex shapes with neural networks remains challenging. Neural fields [33], [34] parameterize continuous coordinate-to-property mappings using neural networks. In our work, we introduce a finite scalar quantization strategy to acquire a concise 3D representation serving as shape priors for downstream conditional inference tasks.

Text to 3D Shape. Inspired by the success of text-to-image task, text-guided 3D shape generation has attracted growing attention recently. However, the scarcity of large-scale text-3D pairs poses a significant challenge. Consequently, some research efforts have concentrated on leveraging CLIP's image-text embeddings as prior knowledge. CLIP-Forge [18] and DreamStone [35] utilize image as a bridge to connect text and 3D shape. Dream3D [36] endeavors to incorporate explicit 3D shape priors into CLIP-guided 3D optimization methods. TAPS3D [13] employs a weighted objective combining low-level image regularization and high-level CLIP loss for diverse and fine-grained generation. In this paper, we introduce sketches into CLIP's embedding space to enrich the geometric information.

III. Method

Figure 1 illustrates our proposed UM3D's architecture, comprising two core components: a Finite Scalar Quantization Autoencoder (FSQ-AE) and a sketch-enhanced CLIP-Glow model. We detail both components in the following sections.

A. Finite Scale Quantization based Autoencoder

During the training stage, our approach first pre-trains a 3D autoencoder model. The primary objective is to compress the high-dimensional continuous 3D shape representation into a compact low-dimensional discrete space. Previous studies have employed VQ-VAE models to obtain a discrete representation, enabling the construction of robust 3D generation models. Nevertheless, enlarging the representation size may result in numerous unused codewords. To tackle this issue, we integrate a finite scalar quantization scheme into the 3D autoencoder framework, proposing our FSQ-based autoencoder architecture. Within this architecture, the encoder E_ψ is responsible for capturing the desired low-dimensional representation z_s' from the input 3D shape S, while the decoder $D\psi$ efficiently maps z_s' back to the 3D shape space $\hat{S}$. This process can be formally expressed as

$$z_s' = FSQ(E_\psi(\mathbf{S})), \hat{\mathbf{S}} = D_\psi(z_s'), \tag{1}$$

$$FSQ(z_s) \triangleq Round[(\left|\frac{L}{2}\right| \sigma(z_s)] \tag{2}$$

Here, the operation $FSQ(\cdot)$ represents a vector quantization step that discretizes a vector into a finite set of codewords, facilitated by the operation $Round[\cdot]$ that rounds a floating-point number to the nearest integer. σ denotes a $tanh$ function. This procedure serves to simplify the representation, yielding a concise 3D shape embedding with optimized utilization, improving the quality of the final reconstruction.

Similar to CLIP-Forge [18], we use the Mean Squared Error (MSE) loss function for training, which is formulated as

$$\mathcal{L} = \frac{1}{N} \sum ||S - \hat{S}||^2. \tag{3}$$

Where N represents the total number of training samples. Finally, the pre-trained FSQ-based autoencoder produces discrete low-dimensional representations with strong generalization capabilities, contributing to the generation of high-quality 3D models. Our architecture is agnostic to both encoder and decoder networks. The design choices are discussed in the ablation study.

B. Sketch-Enhanced CLIP-Guided Instance Glow Model

Aligning with CLIP Visual Features. From previous studies [37], [38], [39], we have found that exploiting the correlation and complementarity between hand-drawn sketches and RGB images can significantly improve the model's performance and generalization capabilities. Specifically, in the context of image-based 3D shape reconstruction, integrating sketches provides critical contour information that refines shape boundaries in RGB images, leading to more accurate 3D reconstructions. In addition, sketches often highlight essential shape details that may be obscured or less prominent in RGB images.

Based on this observation, we develop a cross-modal-like contrastive learning framework to align sketch embeddings with CLIP's joint vision-language feature space. As illustrated in Figure 1, our framework consists of a hand-drawn sketch encoder E_H, and an image encoder E_I. Given the

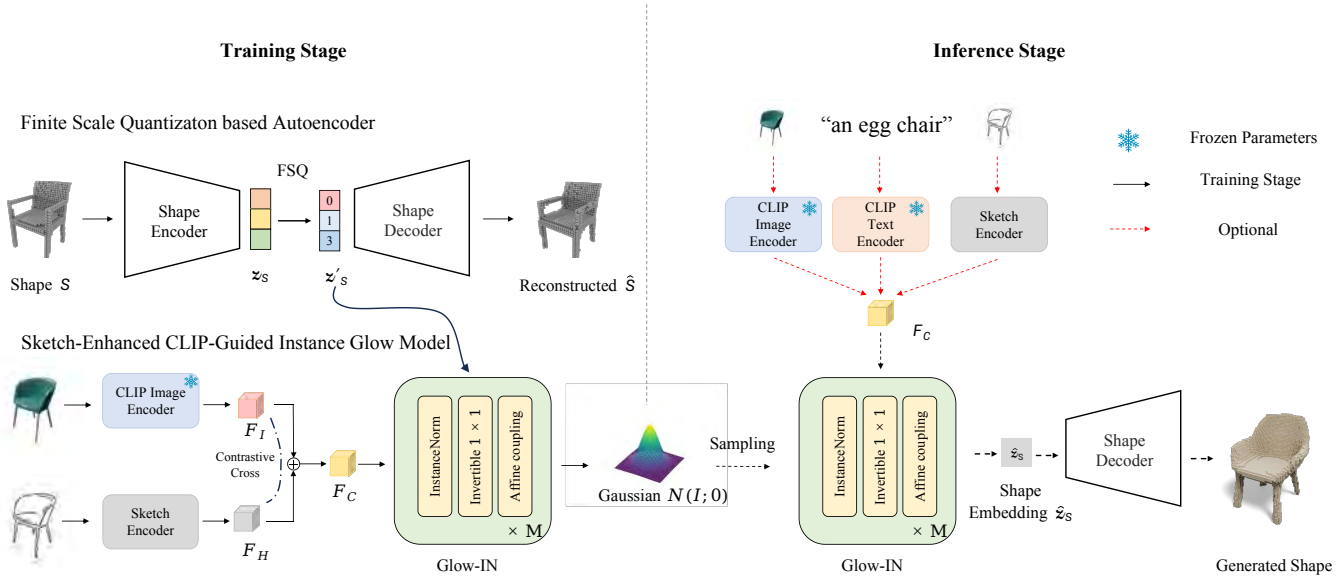

Fig. 1: The architecture of our proposed UM3D network.

inputs $\{\mathbf{I}_n, \mathbf{H}_n\}_{n=1}^{N}$, where $\mathbf{I}_n$ denotes the n-th rendered image from random camera viewpoints, and $\mathbf{H}_n$ represents the corresponding hand-drawn sketch generated with the Canny operator [40]. First, we extract image features $\mathbf{F}_I^n \in \mathbb{R}^{1\times C}$ and sketch features $\mathbf{F}_H^n \in \mathbb{R}^{1\times C}$, respectively, using E_I and E_H. Both encoders are initialized with weights from CLIP's visual encoder.

$$\mathbf{F}_I^n = E_I(\mathbf{I}_n) \tag{4}$$

$$\mathbf{F}_H^n = E_H(\mathbf{H}_n) \tag{5}$$

During training, we freeze the parameters of the CLIP image encoder E_I while keeping the sketch encoder E_H trainable. Subsequently, we employ a contrastive learning scheme to achieve alignment of sketch features with embedding space of CLIP by minimizing the distance between the corresponding image-sketch pairs. We define the contrastive loss InforNCE [41]between image features $\mathbf{F}_I^n$ and sketch features $\mathbf{F}_H^n$ as follows,

$$\mathcal{L}_{cross} = \frac{1}{2N}\sum_{n=1}^{N}(l_{cross}^n(\mathbf{F}_I^n, \mathbf{F}_H^n) + l_{cross}^n(\mathbf{F}_H^n, \mathbf{F}_I^n)) \tag{6}$$

$$l_{cross}^n(\mathbf{F}_H^n, \mathbf{F}_I^n) = -\log \frac{e(\mathbf{F}_H^n, \mathbf{F}_I^n)}{s_{cross}^n(\mathbf{F}_H^n, \mathbf{F}_I^n) - e(\mathbf{F}_H^n, \mathbf{F}_H^n)} \tag{7}$$

$$s_{cross}^n(\mathbf{F}_H^n, \mathbf{F}_I^n) = \sum_{k=1}^{N} e(\mathbf{F}_H^n, \mathbf{F}_H^k) + e(\mathbf{F}_H^n, \mathbf{F}_I^k) \tag{8}$$

Here, $e(a,b) = \exp(a \cdot b^T/\tau)$. We set the temperature hyperparameter $\tau = 0.7$.

Instance Normalization Glow Model. Recent studies [42][43] model 3D shapes as instances from an underlying distribution, employing flow-based generative architectures [44][18] such as RealNVP [45] and Glow [46] to capture this data distribution. Most of these models typically integrate the batch normalization (BN) technique to mitigate the instability issue encountered when training models. However, 3D shape data may exhibit a non-stationary distribution shift across batches, causing biased batch statistics estimation during batch normalization computation, thereby significantly degrading the model's performance. Previous works[47][48] demonstrate that instance normalization can serve as an effective solution to address the instability problem. Inspired by RevIN [49], we present a novel model based on reversible instance normalization flow, termed Glow-In, to alleviate the distribution shift challenge. We normalize the input feature $\mathbf{F}_C^n$ along the dth dimension as

$$\mathbf{F}_C^{'n,d} = \gamma^{(d)}\left(\left(\mathbf{F}_C^{n,d} - \mu^{(d)}\right)\left(\sigma^{2(d)} + \epsilon\right)^{-\frac{1}{2}}\right) + \beta^{(d)} \tag{9}$$

Where $\mathbf{F}_C^n = \mathbf{F}_I^n \oplus \mathbf{F}_H^n$. $\mu^{(d)}$ and $\sigma^{2(d)}$ denote the instance-specific mean and variance, respectively. $\gamma^{(d)}$ and $\beta^{(d)}$ are the learnable parameters of affine transformation.

To enhance transformation expressiveness, following previous studies [45][46], we incorporate the Invertible 1×1 Convolutions and Affine Coupling Layers into our model. Specifically, we use the coupling layers to operate on the features accroding to the following equation:

$$\begin{cases} y^{(1:d)} = z_s^{'(1:d)} \\ y^{(d+1:D)} = z_s^{'(d+1:D)} \odot exp(s([\mathbf{F}_C^{'}; z_s^{'(1:d)}])) \\ \quad +t([\mathbf{F}_C^{'}; z_s^{'(1:d)}]) \end{cases} \tag{10}$$

We let $d_c = \lceil D/2 \rceil$ represent the partitioning position. s and t correspond to the scale and the shift functions, respectively.

We construct the Glow-IN architecture by stacking Glow-IN modules. The model is trained using 3D shape embeddings z_s' conditioned on the image and sketch features from the first stage, which transforms z_s' into a simple multivariate normal distribution. The overall loss function $\mathcal{L}$ at this stage is defined as the combination of flow-based model loss $\mathcal{L}_{glow}$ and the cross-modal contrastive learning loss $\mathcal{L}_{cross}$,

$$\mathcal{L}_{glow}(\mathcal{D}) = -\frac{1}{|\mathcal{D}|}\sum_{x\in\mathcal{D}} log\ p_x(\boldsymbol{x}), \tag{11}$$

$$\mathcal{L} = \mathcal{L}_{glow} + \lambda \cdot \mathcal{L}_{cross} \tag{12}$$

Here, λ is a hyperparameter that balances the contributions between these two loss components. And we set it to be 0.8 in our experiment.

C. Inference

During training, image, sketch, and text features are implicitly encoded into a unified embedding space. As a result, at inference time, we can employ individual image, sketch, text, or any combination of them as input to obtain the condition vector, together with a feature vector sampled from our multivariate normal distribution, fed into our well-trained Glow-In model. The output shape embedding is decoded into the corresponding 3D shape via the shape decoder of our FSQ-based autoencoder.

IV. Experiments

Dataset. We evaluate the efficacy of our 3D shape generation model on the extensively utilized ShapeNet [50] benchmark, which provides 13 common real-world object categories. We use the data preprocessed by Mescheder et al. [51], which contains query points paired with their corresponding occupancy values, together with rendered images associated with each 3D object model. Furthermore, the Canny edge detection algorithm is applied to these images to obtain the sketches. In addition, for fair comparison, we primarily rely on the same train/test split as CLIP-Forge [18].

Implementation Details. Our proposed architecture is implemented using the PyTorch framework. All training and testing experiments are conducted on a single NVIDIA RTX 3090 GPU. Our UM3D model undergoes a two-stage training procedure. Initially, we optimize the FSQ-Based autoencoder (FSQ-AE) using the Adam optimizer with a learning rate of 0.0001 for 300 epochs. We set the batch size to 32. Following [52], the FSQ quantization levels are set to [8, 5, 5, 5].

In the second stage, we train our sketch-enhanced CLIP-guided instance normalization glow model for 100 epochs, employing an initial learning rate of 0.00003. The batch size is 32.

A. Quantitative Analysis

Table I displays the quantitative comparison of our model with state-of-the-art methods on the ShapeNet dataset across various configurations. The "Image" and "Sketch" columns indicate whether these input modalities are present during second-stage training. Our UM3D model achieves superior performance against supervised baseline methods, as these methods formulate this task as recognition/retrieval, and lack robust generalization capabilities. Furthermore, in all settings, UM3D demonstrates competitive results compared to CLIP-Forge. In addition, the introduction of sketches substantially enhances our model's ability to generate geometrically accurate 3D shapes. These quantitative findings verify the effectiveness of our proposed UM3D model.

TABLE I: Quantitative comparison of various methods on the ShapeNet dataset.

Method	Image	Sketch	FID↓	MMD↑	Acc.↑
Text2shape-CMA	-	-	16078.05	0.4992	4.27
Text2shape-supervised	-	-	14881.96	0.1418	6.84
CLIP-Forge	✓	×	2425.25	0.6607	83.33
	×	✓	5066.37	0.5531	59.82
	✓	✓	12482.01	0.5089	43.16
UM3D(Ours)	✓	×	2389.12	0.6982	82.18
	×	✓	4429.52	0.5827	63.56
	✓	✓	**2114.24**	**0.7009**	**85.04**

B. Qualitative Analysis

To further evaluate the generative capabilities of our proposed UM3D model across various tasks, we conduct the following experiments.

Text-to-3D Shape Generation. By leveraging the pre-trained vision-language model (CLIP), our model associates language semantic information with 3D shape representations using rendered images as a bridge. Therefore, our method enables zero-shot text-conditional 3D shape generation without requiring text as supervision during training. Figure 2 visualizes several examples of generated 3D shapes based on input text prompts. It can be observed that our method effectively captures deep semantic information and accurately produces 3D structures aligned with the provided textual descriptions, which shows its effectiveness and strong generalization capabilities.

Qualitative Comparison. We present visual comparisons between our UM3D model and the baseline method CLIP-Forge on the task of text-conditioned 3D generation in Figure 3. Both approaches can reconstruct the fundamental geometric structures of objects. However, CLIP-Forge tends to generate less accurate 3D shapes. For example, when generating "a lounge chair", CLIP-Forge creates a straight backrest, while our method yields a subtly curved backrest and a seat base that faithfully captures the essential characteristics of a lounge chair. Similarly, in the case of "a jeep", the size and overall shape of the model generated by our method align more closely with real jeeps compared to that produced by CLIP-Forge. These results clearly demonstrate our UM3D's capability to produce high-fidelity 3D geometry with realistic appearance.

Furthermore, while CLIP-Forge is capable of generating reasonable 3D geometry, its results display noticeable artifacts or noisy surfaces, such as tiny holes, a distorted base and a tilted stand in the lamp example. In contrast, our approach performs well on high-fidelity 3D shape generation. Notably, when it comes to detail preservation, our method recovers edge details and smooth surfaces, as seen in examples like "a digital display", "a mobile phone", and "a circular bench". These experimental results highlight the effectiveness of our model in producing text-consistent 3D shapes with superior geometric accuracy.

Fig. 2: Qualitative visualization of text-to-3D shape generation. The category names included in these text prompts originate from the annotation space of ShapeNet.

Fig. 3: Visual comparisons of text-to-3D shape generation. The shapes in cyan represent the CLIP-Forge's outputs, while the models in light brown are generated by our method.

Multimodal-Conditioned 3D Shape Generation. In addition to text-conditioned 3D generation, our method also facilitates 3D generation from image, sketch, or multiple conditioning modalities, since we have incorporated sketch features into CLIP's joint embedding space. Figure 4 presents qualitative comparisons of 3D generation under diverse input conditions. Key findings are as follows: First, given an image, our UM3D model generates a more accurate 3D shape that is faithful to the object shown in this image compared to CLIP-Forge. When provided with only a sketch, our method produces a 3D shape that accurately aligns with the sketch. In contrast, CLIP-Forge suffers from model collapse, resulting in outputs that diverge significantly from the input sketches. Second, in multimodal settings (such as combined text and image inputs), CLIP-Forge produces shapes inconsistent with the text prompts, while our method maintains consistency across all inputs, yielding 3D shapes that conform precisely to the given conditions. This indicates that our approach facilitates smooth transitions between single- and multi-modal inputs. Finally, our method achieves superior details and structural accuracy, particularly in handling the backrest and seat cushion of the bench compared to the CLIP-Forge method.

These experimental results demonstrate our method's su-

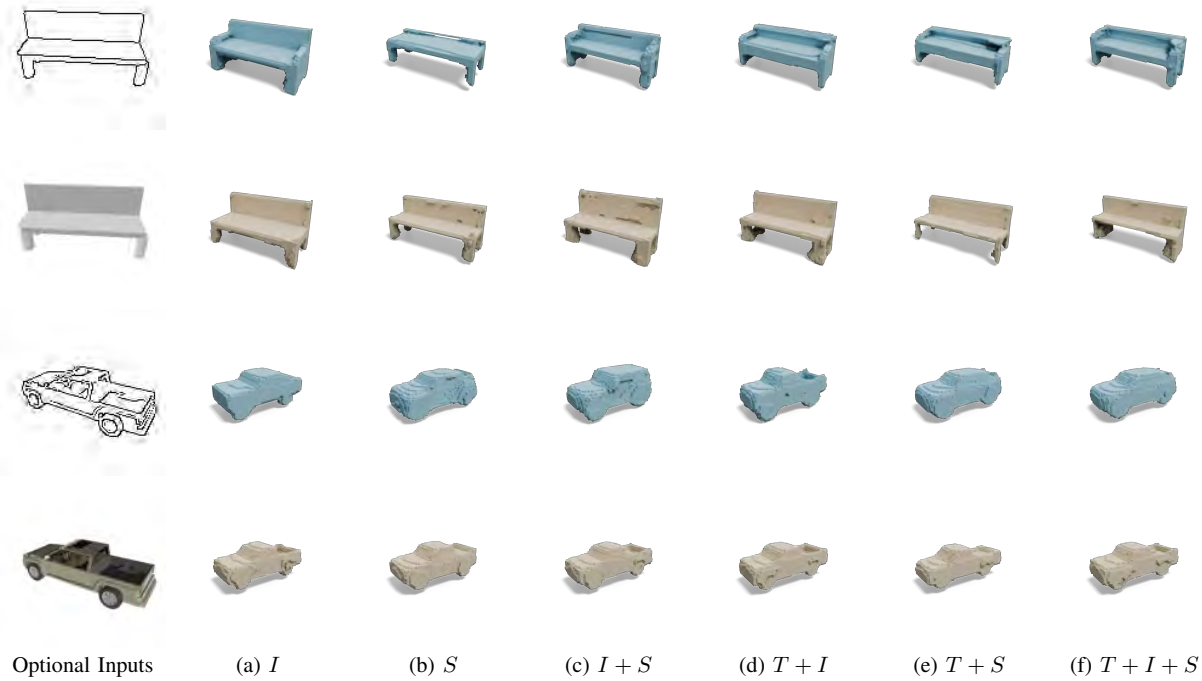

Fig. 4: Visualization of 3D shape generation results under diverse input conditions. The cyan shapes signify the outputs from CLIP-Forge, whereas the light brown models are the creations of our method. The first column provides color images and sketches for two categories, including bench and truck, with corresponding text prompts "a bench" and "a truck". I denotes color image, **S** represents sketch, T refers to text prompt.

Fig. 5: Visual results of novel-category 3D shape generation. The shapes in the first row are created by CLIP-Forge, while 3D models in the second row are the outputs of our UM3D.

perior performance in generative accuracy and structural detail, particularly when integrating multimodal inputs. The resulting 3D models exhibit highly plausible appearances with refined surface characteristics and geometrically coherent forms. This confirms that our method may be more suitable for complex 3D shape generation tasks and can effectively leverage heterogeneous input modalities to enhance the quality of the generated 3D geometries.

Novel-Category 3D Shape Generation. Figure 5 presents comparative results between our method and the baseline approach for novel category 3D object generation. It can be seen that when synthesizing shapes beyond the ShapeNet dataset, our approach outperforms the baseline, generating expected 3D objects consistent with text. This exhibits the superior generalization capability and adaptability of our proposed UM3D in handling novel-domain 3D shape generation tasks, enabling future cross-dataset and cross-domain applications.

TABLE II: Ablation study on the design choices of FSQ-based autoencoder.

Noise	Latent	Encoder	Decoder	VQ	IoU↑	MSE↓
✓	128	VoxEnc	RN-OccNet	FSQ	0.7339	0.00825
✓	128	VoxEnc	CBN-OccNet	-	0.7502	0.00849
✓	256	VoxEnc	CBN-OccNet	VQ-VAE	0.7502	**0.00718**
✓	256	VoxEnc	CBN-OccNet	FSQ	**0.7634**	0.00723

TABLE III: Ablation study on the design choices of flow model.

condition	prior	FID↓	MMD↑	Acc.↑
affine coupling	MAF	6052.62	0.6273	59.40
	Glow	2412.67	0.6778	82.07
	Glow-BN	2507.68	0.6596	81.45
	Glow-IN	**2389.12**	**0.6812**	**82.18**

C. Ablation Study

In this section, we perform ablation experiments to investigate the contributions of individual modules in our proposed model.

Design choice of FSQ-based autoencoder. First, we examine the impact of latent dimension size. As shown in Table II, the optimal 3D generation performance is achieved with a dimension of 256. Subsequently, we investigate the influence of different vector quantization (VQ) strategies, including VQ-VAE and FSQ. The results show a significant IoU improvement when incorporating the FSQ method.

Design choice of flow model. The ablation study results regarding the selection of the flow model are detailed in Table III. From these results, we can draw the following conclusions: the Glow network performs favorably against the MAF model [53]. Additionally, compared to the original Glow and batch normalization Glow (Glow-BN), introducing instance normalization in the Glow model enhances model's performance significantly. This improvement stems from its ability to model a more accurate data distribution, boosting the quality and diversity of the generated samples.

V. Conclusion

In this paper, we propose a two-stage pre-training framework, termed UM3D, for unified multimodal 3D shape generation. The first stage involves the development of a finite scalar quantization-based autoencoder network to capture an optimized 3D shape representation with enhanced efficacy. In the second stage, we introduce a novel Instance-Normalized Glow model to learn distributions of 3D shape representations conditioned by our sketch-enhanced joint embedding space. UM3D is capable of processing individual or combined inputs (text/image/sketch) to produce the corresponding 3D structures. The quantitative and qualitative experiments showcase the superior performance of our method in generating high-quality 3D shapes.

References

[1] B. Zhang, J. Tang, M. Niessner, and P. Wonka, "3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models," *ACM Transactions on Graphics (TOG)*, vol. 42, no. 4, pp. 1–16, 2023.

[2] T. Chen, C. Ding, L. Zhu, Y. Zang, Y. Liao, Z. Li, and L. Sun, "Reality3dsketch: Rapid 3d modeling of objects from single freehand sketches," *IEEE Transactions on Multimedia*, vol. 26, pp. 4859–4870, 2024.

[3] Z. Liu, P. Dai, R. Li, X. QI, and C.-W. Fu, "Iss: Image as stepping stone for text-guided 3d shape generation," in *Proceedings of the International Conference on Learning Representations*, 2023.

[4] T. Huang, Y. Zeng, Z. Zhang, W. Xu, H. Xu, S. Xu, R. W. Lau, and W. Zuo, "Dreamcontrol: Control-based text-to-3d generation with 3d self-prior," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 5364–5373.

[5] Z. Liu, J. Hu, K.-H. Hui, X. Qi, D. Cohen-Or, and C.-W. Fu, "Exim: A hybrid explicit-implicit representation for text-guided 3d shape generation," *ACM Transactions on Graphics (TOG)*, vol. 42, no. 6, pp. 1–12, 2023.

[6] D. Di, J. Yang, C. Luo, Z. Xue, W. Chen, X. Yang, and Y. Gao, "Hyper-3dg: Text-to-3d gaussian generation via hypergraph," *International Journal of Computer Vision*, vol. 133, no. 5, pp. 2886–2909, 2025.

[7] Z. Zhao, W. Liu, X. Chen, X. Zeng, R. Wang, P. Cheng, B. FU, T. Chen, G. Yu, and S. Gao, "Michelangelo: Conditional 3d shape generation based on shape-image-text aligned latent representation," in *Proceedings of the Advances in Neural Information Processing Systems*, vol. 36, 2023, pp. 73 969–73 982.

[8] J. Cha, J. Kim, J. S. Yoon, and S. Baek, "Text2hoi: Text-guided 3d motion generation for hand-object interaction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 1577–1585.

[9] C. Li, C. Zhang, A. Waghwase, L.-H. Lee, F. Rameau, Y. Yang, S.-H. Bae, and C. S. Hong, "Generative ai meets 3d: A survey on text-to-3d in aigc era," *arXiv preprint arXiv:2305.06131*, 2023.

[10] X. He, J. Chen, S. Peng, D. Huang, Y. Li, X. Huang, C. Yuan, W. Ouyang, and T. He, "Gvgen: Text-to-3d generation with volumetric representation," in *European Conference on Computer Vision*, 2024, pp. 463–479.

[11] Y.-C. Cheng, H.-Y. Lee, S. Tulyakov, A. G. Schwing, and L.-Y. Gui, "Sdfusion: Multimodal 3d shape completion, reconstruction, and generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 4456–4465.

[12] F.-L. Liu, H. Fu, Y.-K. Lai, and L. Gao, "Sketchdream: Sketch-based text-to-3d generation and editing," *ACM Transactions on Graphics (TOG)*, vol. 43, no. 4, pp. 1–13, 2024.

[13] J. Wei, H. Wang, J. Feng, G. Lin, and K.-H. Yap, "Taps3d: Text-guided 3d textured shape generation from pseudo supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 16 805–16 815.

[14] J. Zhang, J. Huang, S. Jin, and S. Lu, "Vision-language models for vision tasks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 8, pp. 5625–5644, 2024.

[15] K. Huang, K. Sun, E. Xie, Z. Li, and X. Liu, "T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation," in *Proceedings of the Advances in Neural Information Processing Systems*, vol. 36, 2023, pp. 78 723–78 747.

[16] J. Xu, X. Liu, Y. Wu, Y. Tong, Q. Li, M. Ding, J. Tang, and Y. Dong, "Imagereward: Learning and evaluating human preferences for text-to-image generation," in *Proceedings of the Advances in Neural Information Processing Systems*, vol. 36, 2023, pp. 15 903–15 935.

[17] F. Yin, X. Chen, C. Zhang, B. Jiang, Z. Zhao, W. Liu, G. Yu, and T. Chen, "Shapegpt: 3d shape generation with a unified multi-modal language model," *IEEE Transactions on Multimedia*, vol. 27, pp. 4107–4120, 2025.

[18] A. Sanghi, H. Chu, J. G. Lambourne, Y. Wang, C.-Y. Cheng, M. Fumero, and K. R. Malekshan, "Clip-forge: Towards zero-shot text-to-shape generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 18 603–18 613.

[19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[20] T. Qiao, J. Zhang, D. Xu, and D. Tao, "Mirrorgan: Learning text-to-image generation by redescription," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1505–1514.

[21] X. Liu, C. Gong, L. Wu, S. Zhang, H. Su, and Q. Liu, "Fusedream: Training-free text-to-image generation with improved clip+gan space optimization," *arXiv preprint arXiv:2112.01573*, 2021.

[22] Z. Wang, W. Liu, Q. He, X. Wu, and Z. Yi, "Clip-gen: Language-free training of a text-to-image generator with clip," *arXiv preprint arXiv:2203.00386*, 2022.

[23] Y. Zhou, R. Zhang, C. Chen, C. Li, C. Tensmeyer, T. Yu, J. Gu, J. Xu, and T. Sun, "Towards language-free training for text-to-image generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 17 907–17 917.

[24] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, *et al.*, "Laion-5b: An open large-scale dataset for training next generation image-text models," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 25 278–25 294.

[25] Z. Shi, Z. Meng, Y. Xing, Y. Ma, and R. Wattenhofer, "3d-retr: End-to-end single and multi-view 3d reconstruction with transformers," in *32nd British Machine Vision Conference (BMVC 2021)*, 2021, p. 405.

[26] A. Sanghi, P. K. Jayaraman, A. Rampini, J. Lambourne, H. Shayani, E. Atherton, and S. Asgari Taghanaki, "Sketch-a-shape: Zero-shot sketch-to-3d shape generation," *arXiv preprint arXiv:2307.03869*, 2023.

[27] Z. Wu, Y. Wang, M. Feng, H. Xie, and A. Mian, "Sketch and text guided diffusion model for colored point cloud generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 8929–8939.

[28] Z. Xiang, Z. Huang, and K. Khoshelham, "Synthetic lidar point cloud generation using deep generative models for improved driving scene object recognition," *Image and Vision Computing*, vol. 150, p. 105207, 2024.

[29] K. Wu, F. Liu, Z. Cai, R. Yan, H. Wang, Y. Hu, Y. Duan, and K. Ma, "Unique3d: High-quality and efficient 3d mesh generation from a single image," in *Advances in Neural Information Processing Systems*, vol. 37, 2024, pp. 125 116–125 141.

[30] M. Liu, C. Zeng, X. Wei, R. Shi, L. Chen, C. Xu, M. Zhang, Z. Wang, X. Zhang, I. Liu, H. Wu, and H. Su, "Meshformer : High-quality mesh generation with 3d-guided reconstruction model," in *Advances in Neural Information Processing Systems*, vol. 37, 2024, pp. 59 314–59 341.

[31] K. Yamazaki, T. Hanyu, K. Vo, T. Pham, M. Tran, G. Doretto, A. Nguyen, and N. Le, "Open-fusion: Real-time open-vocabulary 3d mapping and queryable scene representation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 9411–9417.

[32] S. Li, J. Zhou, B. Ma, Y.-S. Liu, and Z. Han, "Learning continuous implicit field with local distance indicator for arbitrary-scale point cloud upsampling," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 38, no. 4, 2024, pp. 3181–3189.

[33] Y. Lan, F. Tan, Q. Xu, D. Qiu, K. Genova, Z. Huang, S. Fanello, R. Pandey, T. Funkhouser, C. C. Loy, *et al.*, "Loc3diff: Local diffusion for 3d human head synthesis and editing," in *European Conference on Computer Vision*, 2024, pp. 49–66.

[34] A. Chetan, G. Yang, Z. Wang, S. Marschner, and B. Hariharan, "Accurate differential operators for hybrid neural fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025, pp. 530–539.

[35] Z. Liu, P. Dai, R. Li, X. Qi, and C.-W. Fu, "Dreamstone: Image as a stepping stone for text-guided 3d shape generation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 12, pp. 14 385–14 403, 2023.

[36] J. Xu, X. Wang, W. Cheng, Y.-P. Cao, Y. Shan, X. Qie, and S. Gao, "Dream3d: Zero-shot text-to-3d synthesis using 3d shape prior and text-to-image diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 20 908–20 918.

[37] P. Sangkloy, J. Lu, C. Fang, F. Yu, and J. Hays, "Scribbler: Controlling deep image synthesis with sketch and color," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5400–5409.

[38] C. Xiao and H. Fu, "Customsketching: Sketch concept extraction for sketch-based image synthesis and editing," in *Computer Graphics Forum*, vol. 43, no. 7, 2024, p. e15247.

[39] S. Lee, H. Jung, B. Koh, Q. Huang, S. H. Yoon, and S. Kim, "Pasta: Part-aware sketch-to-3d shape generation with text-aligned prior," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2025, pp. 18 585–18 595.

[40] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 679–698, 1986.

[41] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[42] G. Yang, X. Huang, Z. Hao, M.-Y. Liu, S. Belongie, and B. Hariharan, "Pointflow: 3d point cloud generation with continuous normalizing flows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 4541–4550.

[43] R. Klokov, E. Boyer, and J. Verbeek, "Discrete point flow networks for efficient point cloud generation," in *European Conference on Computer Vision*, 2020, pp. 694–710.

[44] A. Pumarola, S. Popov, F. Moreno-Noguer, and V. Ferrari, "C-flow: Conditional generative flow models for images and 3d point clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 7949–7958.

[45] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," in *International Conference on Learning Representations*, 2017.

[46] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," in *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[47] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[48] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1501–1510.

[49] T. Kim, J. Kim, Y. Tae, C. Park, J.-H. Choi, and J. Choo, "Reversible instance normalization for accurate time-series forecasting against distribution shift," in *International Conference on Learning Representations*, 2021.

[50] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, 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.

[51] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3d reconstruction in function space," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4460–4470.

[52] F. Mentzer, D. Minnen, E. Agustsson, and M. Tschannen, "Finite scalar quantization: Vq-vae made simple," in *International Conference on Learning Representations*, 2024.

[53] G. Papamakarios, T. Pavlakou, and I. Murray, "Masked autoregressive flow for density estimation," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.