

QKAN-GS: Quantum-Empowered 3D Gaussian Splatting

Fujihashi, Takuya; Kuwabara, Akihiro; Koike-Akino, Toshiaki

TR2025-156 October 29, 2025

Abstract

The recent 3D Gaussian Splatting (3DGS) method has been expected for high-quality 3D scene rendering. However, due to its explicit representation, which requires geometry and attributes for millions of individual 3D Gaussians, 3DGS requires a significant data size for storage and transmission. For data reduction, some studies have proposed signal-processing-based and generative-based 3DGS compression methods. The generative-based methods utilize multi-layer perceptrons (MLPs) with fixed activation functions for individual attributes to reduce the locally redundant Gaussians. In this paper, we propose a novel approach, QKAN-GS, to represent the individual Gaussian attributes with a small model size. For this purpose, we design a Quantum-inspired Kolmogorov-Arnold Network (QKAN), which is quantum-empowered learnable activation functions on model edges, to maintain the model's expressive power with fewer parameters. Experiments show that the proposed QKAN-GS achieves better 3D reconstruction quality than the generative 3DGS compression method, regardless of whether it uses fixed or learnable activation functions, under the same data size.

ACM Multimedia Workshop 2025

QKAN-GS: Quantum-Empowered 3D Gaussian Splatting

Takuya Fujihashi
The University of Osaka
Suita, Osaka, Japan
fujihashi.takuya@ist.osaka-u.ac.jp

Akihiro Kuwabara
The University of Osaka
Suita, Osaka, Japan
kuwabara.akihiro@ist.osaka-u.ac.jp

Toshiaki Koike-Akino
Mitsubishi Electric Research
Laboratories
Cambridge, MA, United States
koike@merl.com

Abstract

The recent 3D Gaussian Splatting (3DGS) method has been expected for high-quality 3D scene rendering. However, due to its explicit representation, which requires geometry and attributes for millions of individual 3D Gaussians, 3DGS requires a significant data size for storage and transmission. For data reduction, some studies have proposed signal-processing-based and generative-based 3DGS compression methods. The generative-based methods utilize multi-layer perceptrons (MLPs) with fixed activation functions for individual attributes to reduce the locally redundant Gaussians. In this paper, we propose a novel approach, QKAN-GS, to represent the individual Gaussian attributes with a small model size. For this purpose, we design a Quantum-inspired Kolmogorov-Arnold Network (QKAN), which is quantum-empowered learnable activation functions on model edges, to maintain the model's expressive power with fewer parameters. Experiments show that the proposed QKAN-GS achieves better 3D reconstruction quality than the generative 3DGS compression method, regardless of whether it uses fixed or learnable activation functions, under the same data size.

CCS Concepts

• **Computing methodologies** → **Image compression**; **Computer vision**; • **Computer systems organization** → **Quantum computing**.

Keywords

3D-GS, Generative 3D-GS Compression, QNN, KAN

ACM Reference Format:

Takuya Fujihashi, Akihiro Kuwabara, and Toshiaki Koike-Akino. 2018. QKAN-GS: Quantum-Empowered 3D Gaussian Splatting. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 5 pages. <https://doi.org/XXXXXXX.XXXXXXX>

1 Introduction

The rendering of three-dimensional (3D) scenes in a photorealistic and real-time manner has been a central goal in the fields of computer vision and graphics. The recent introduction of 3D Gaussian Splatting (3D-GS) [5, 7] has marked a significant breakthrough,

achieving state-of-the-art rendering quality and speed by representing 3D scenes as a large number of 3D Gaussians with attributes including position, rotation, scale, color, and opacity. This method utilizes a tile-based rasterizer that has been optimized for modern GPUs. This allows for efficient rendering without the costly ray-marching required by many neural radiance fields. However, 3D-GS causes a significantly large model size. Storing and transmitting the attributes of millions of Gaussians requires hundreds of megabytes or even gigabytes of storage and network capacity for complex scenes. Thus, deploying 3D-GS models on resource-constrained devices, such as wearable devices and mobile devices, is challenging.

To address this challenge, existing studies on 3D-GS compression methods have proposed methods to compress Gaussian primitives. They can be broadly categorized into two distinct classes: generative and traditional compression methods. The traditional compression methods convert the learned 3D Gaussians into a bitstream suitable for storage and transmission by applying signal-processing-based compression techniques. A pioneering work is Graph-based Gaussian Splatting Compression (GGSC) [21], which utilizes Graph Signal Processing (GSP). They regard the Gaussian primitives as a graph signal and define a graph Fourier transform (GFT) based on the graph.

On the other hand, the generative methods are designed to construct more compact scene representations by optimizing 3D-GS parameters under specific constraints or by learning a compact parameter representation. To obtain the compact representation, many studies aim to use lightweight multi-layer perceptrons (MLPs) for generating Gaussian attributes from inputs, including embeddings or structural features [2, 6, 11, 19]. Other studies consider that many Gaussian primitives share similar attributes, and thus they use codebooks based on K-means for quantization [3, 9, 12–14].

The generative studies achieved a compact 3D-GS representation by utilizing small MLPs to realize neural Gaussians for each attribute of Gaussian primitives. However, the performance may be limited since their MLPs used a fixed activation function for generating Gaussian attributes. In this paper, we introduce QKAN-GS, inspired by Kolmogorov-Arnold Network (KAN) [10, 18] and quantum neural networks (QNNs) [1, 4, 16]. Unlike typical MLP architectures, KANs implement learnable activation functions at the edges of the network, enabling them to capture complex functions. In addition, QNN is a promising technique for accelerating computation while reducing the number of parameters. Our quantum-inspired KAN (QKAN) architecture aims to combine these advantages to achieve high-quality rendering and a compact representation in 3D-GS compression.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference acronym 'XX, Woodstock, NY

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-XXXX-X/2018/06
<https://doi.org/XXXXXXX.XXXXXXX>

Experiments using an open 3D-GS dataset demonstrate that our QKAN-based architecture achieves better rate-distortion performance compared with the state-of-the-art generative method, Scaffold-GS [11], and KAN-based architecture.

Our main contributions can be summarized as follows:

- We propose QKAN-GS, a new paradigm for generative 3D-GS compression, to realize quantum-empowered learnable activation functions for generating Gaussian attributes.
- We are the first to propose the concept of QKAN for parameter-efficient 3D scene representation.

2 QKAN-GS

Fig. 1 (a) shows the overview of the proposed QKAN-GS. Our QKAN-GS leverages the fundamental architecture of Scaffold-GS, which is an anchor-based generative 3D-GS compression. More specifically, 3D Gaussian primitives are divided into anchors, each with corresponding 3D coordinates and attributes consisting of features, scalings, and offsets. The anchor features are fed into the proposed QKAN architecture of each attribute. The features are first passed to a linear layer. The output of the linear layer, $\mathbf{h}$, is then fed into an element-wise quantum activation function, $\hat{h}_i = \text{Qact}_i(h_i)$, where h_i is an element of $\mathbf{h}$. Finally, another linear layer is used to generate the corresponding attribute from the extracted features $\hat{\mathbf{h}}$.

2.1 Preliminaries

2.1.1 3D Gaussian Splatting. 3D-GS represents a 3D scene using numerous Gaussians and renders viewpoints through differentiable splatting and tile-based rasterization. Each Gaussian is initialized from Structure from Motion (SfM) and defined by a 3D covariance matrix $\Sigma \in \mathbb{R}^{3 \times 3}$ and location (mean) $\mu \in \mathbb{R}^3$:

$$G(\mathbf{x}) = \exp\left(-\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1}(\mathbf{x} - \mu)\right), \quad (1)$$

where $\mathbf{x} \in \mathbb{R}^3$ is a random 3D point, and Σ is defined by a diagonal matrix $S \in \mathbb{R}^{3 \times 3}$ representing scaling and rotation matrix $R \in \mathbb{R}^{3 \times 3}$ to guarantee its positive semi-definite characteristics, such that $\Sigma = RSS^T R^T$. To render an image from a random viewpoint, 3D Gaussians are first splatted to 2D, and render the pixel value $C \in \mathbb{R}^3$ using α -composed blending.

$$C = \sum_{i \in I} c_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j), \quad (2)$$

where $\alpha \in \mathbb{R}^1$ measures the opacity of each Gaussian after 2D projection, $\mathbf{c} \in \mathbb{R}^3$ is view-dependent color modeled by Spherical Harmonics (SH) coefficients, and I is the number of sorted Gaussians for rendering.

2.1.2 Scaffold-GS. Scaffold-GS extends the framework of 3DGS and introduces a more storage-friendly anchor-based approach. More specifically, it utilizes anchors to cluster Gaussians and deduce their attributes from the attributes of attached anchors through MLPs, rather than directly storing them. Specifically, each anchor consists of a location $\mathbf{x}^a \in \mathbb{R}^3$ and anchor attributes $\mathcal{A} = \{\mathbf{f}^a \in \mathbb{R}^{D^a}, \mathbf{l} \in \mathbb{R}^6, \mathbf{o} \in \mathbb{R}^{3K}\}$, where each component represents anchor feature, scaling, and offsets, respectively. Here, K is the number of offsets per anchor. During rendering, $\mathbf{f}^a$ is fed into MLPs to

generate attributes for Gaussians, whose locations are determined by adding $\mathbf{x}^a$ and $\mathbf{o}$, where $\mathbf{l}$ is utilized to regularize both locations and shapes of the Gaussians.

2.2 QKAN

In contrast to the typical MLPs with fixed activation functions, KAN shifts the paradigm from “learnable weights and fixed activations” to “learnable weights and activations”. Although traditional KAN architectures utilized B-splines and fully-connected layers, our QKAN architecture extends this concept by harnessing the power of QNNs. This approach may have the potential to improve the rate-distortion performance in generative 3D-GS compression by creating more expressive and parameter-efficient activation functions.

Fig. 1 (b) shows the proposed QKAN architecture. It is applied independently to generate each Gaussian attribute, namely for $\Phi_{\text{cov}}(\mathbf{f}^a; \theta_{\text{cov}})$, opacity $\Phi_{\text{opacity}}(\mathbf{f}^a; \theta_{\text{opacity}})$, and color $\Phi_{\text{color}}(\mathbf{f}^a; \theta_{\text{color}})$. Here, θ_* represents the set of learnable parameters for each specific attribute. The QKAN architecture mainly consists of three main components: an initial linear layer, a channel-wise quantum activation layer, and a final linear layer.

First, the input anchor feature $\mathbf{f}^a$ is projected into an embedding vector $\mathbf{h} \in \mathbb{R}^M$ by an initial layer:

$$\mathbf{h} = \mathbf{W}\mathbf{f}^a + \mathbf{b}, \quad (3)$$

where $\mathbf{W} \in \mathbb{R}^{M \times D^a}$ and $\mathbf{b} \in \mathbb{R}^M$ are trainable weight and bias. The embedding vector is then processed by the element-wise quantum activation layer $\text{Qact}_i(h_i)$. More specifically, the forward pass of $\text{Qact}_i(h_i)$ can be described as:

$$\hat{h}_i = \text{Qact}_i(h_i), \quad \text{for } i = 1, \dots, M. \quad (4)$$

The quantum activation layer for each element h_i consists of embedding and entangling layers. For the embedding layer, we use the amplitude embedding to encode each element of the embedding vector $\mathbf{h}$ into a single qubit. The entangling layer is based on a parameterized quantum circuit in [17]. Specifically, the parameterized circuit sequentially performs Z-rotation and X-rotation on the single qubit. Here, each rotation gate is controlled based on the parameter set θ_* . A number of entangling layers L are sequentially cascaded. These embedding and entangling layers are iterated over a few layers, with a shuffled extension of the data re-uploading trick [15]. Finally, the expectation value or the probability value of the qubit’s measurement is taken as the output of the quantum activation layer $\hat{h}_i$.

After the activation layer, we concatenate the outputs $\{\hat{h}_i\}_{i=1}^M$ to form an activated vector $\hat{\mathbf{h}} \in \mathbb{R}^M$. The vector is finally fed into another linear layer to obtain the corresponding attribute for each anchor.

2.3 Training and Loss Function

We train the network by optimizing all learnable parameters, including those in the linear layers and the quantum activation layer. Our training objective follows the methodology of Scaffold-GS [11], utilizing a composite loss function that balances photometric accuracy and geometric compactness. The total loss $\mathcal{L}$ is a weighted sum of three components: a photometric loss, the structural similarity (SSIM) loss $\mathcal{L}_{\text{SSIM}}$, and a volume regularization loss $\mathcal{L}_{\text{vol}}$. The

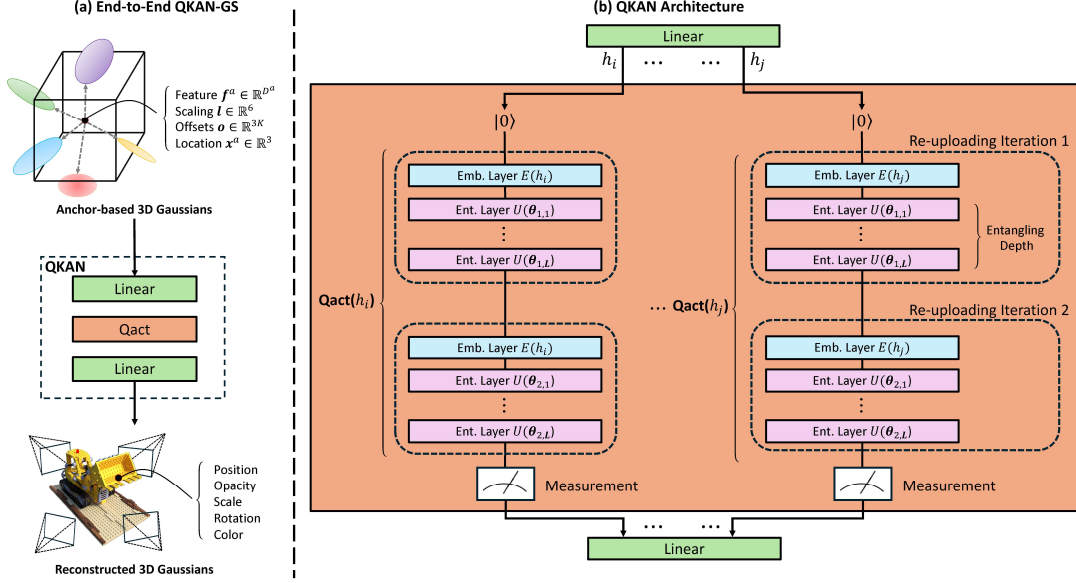


Figure 1: Overview architecture of QKAN-GS. (a) end-to-end architecture. (b) our QKAN architecture.

function is given by:

$$\mathcal{L} = \mathcal{L}_1 + \lambda_{\text{SSIM}} \mathcal{L}_{\text{SSIM}} + \lambda_{\text{vol}} \mathcal{L}_{\text{vol}}, \quad (5)$$

where $\lambda_{\text{SSIM}} = 0.2$ and $\lambda_{\text{vol}} = 0.001$ are weighting coefficients.

3 Evaluation

3.1 Setting

Dataset and Metric: We select two scenes from an open dataset: “train” and “truck” from Tanks & Temples [8] as real-world dataset.

We used the Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [20], Learned Perceptual Image Patch Similarity (LPIPS) [22], and compressed data size of the anchors and corresponding MLPs for each attribute in megabytes (MB) as performance metrics.

Baseline: We consider Scaffold-GS and KAN-based Scaffold-GS (KAN-GS) as the baselines. For the implementation of both schemes, we use the implementations provided by the official Scaffold-GS’s GitHub repository. Here, Scaffold-GS used MLPs with a fixed activation function for each attribute. KAN-GS used a channel-wise FC layer as a learnable activation function. For our proposed QKAN-GS, we adopted the optimal QKAN architecture. This was found by selecting, for each attribute, whether to use the proposed quantum activation function or a fixed activation function, and then choosing the combination that yielded the best rate distortion performance. Here, we fixed the number of entangling layers, L , to 1 and tested up to five re-uploading iterations to choose the best combination.

3.2 Baseline Performance

Figs. 2 (a)-(f) show the PSNR, SSIM, and LPIPS as a function of data sizes in “train” and “truck” scenes, respectively. For both scenes, QKAN-GS achieves a better trade-off between compressed data size and reconstruction quality across all three metrics. Especially, in a

large data size, QKAN-GS provides a higher PSNR and SSIM, and a lower (better) LPIPS value than both the Scaffold-GS and KAN-GS baselines.

Figs. 3 (a)-(d) and Figs. 4 (a)-(d) show the qualitative evaluation of the baselines for different scenes. As illustrated in Fig. 3 for the “train” scene, the reconstructed image by QKAN-GS restores sharper details and more accurate color representation compared to Scaffold-GS and KAN-GS, which is reflected in its higher PSNR and SSIM scores. Similarly, the “truck” scene shown in Fig. 4 shows that although the PSNR for QKAN-GS is the same as for KAN-GS, our QKAN-GS achieves better SSIM and LPIPS scores. Although the visual difference between QKAN-GS and the other baselines in the “truck” scene is slight, it indicates higher perceptual quality with better preservation of structural details and textures.

4 Conclusion

In this paper, we introduced QKAN-GS, a novel generative compression method for 3D-GS. By replacing the standard MLPs and their fixed activation functions with a QKAN architecture featuring quantum-empowered learnable activation functions, we aimed to create a more parameter-efficient and expressive model for representing Gaussian attributes. Our approach builds upon the anchor-based framework of Scaffold-GS, enhancing its compression capabilities without sacrificing reconstruction quality. Our experimental results on the Tanks & Temples dataset demonstrate the effectiveness of QKAN-GS. It outperformed the Scaffold-GS, which is a fixed activation function, and a KAN-based variant across different data sizes, demonstrating superior rate-distortion performance in terms of PSNR, SSIM, and LPIPS metrics. Although QKAN-GS shows promising results, we acknowledge that its evaluation is currently limited to two scenes.

For future work, we plan to explore the application of QKAN to other aspects of 3D scene representation and investigate further

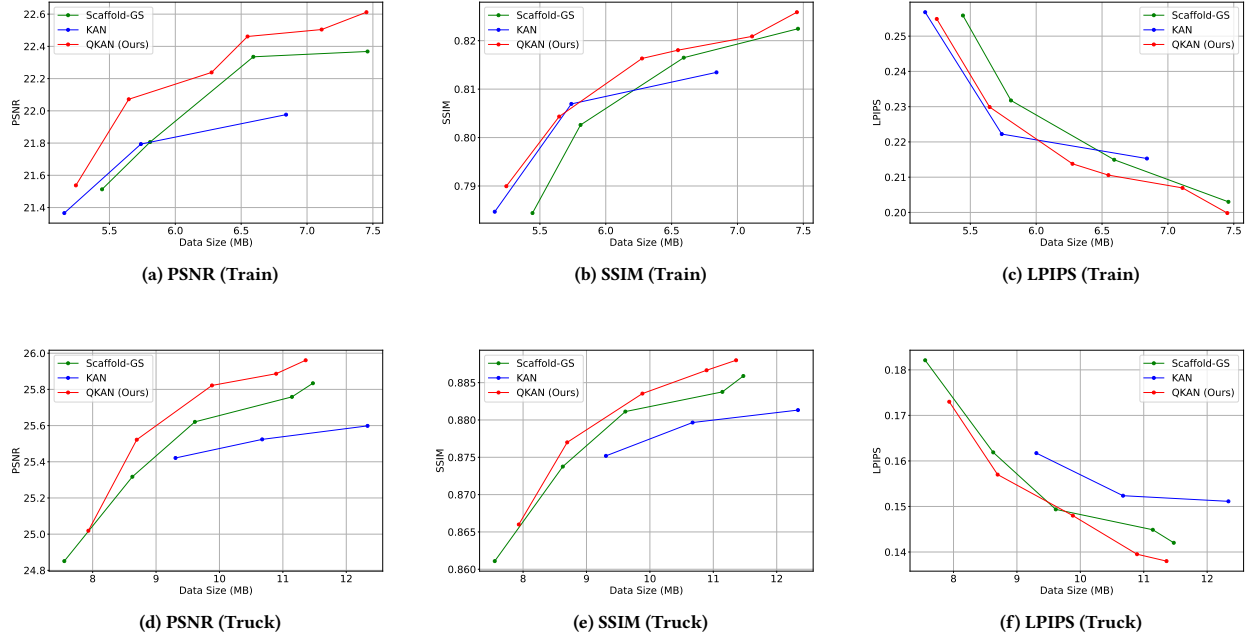


Figure 2: Reconstruction quality of each scheme as a function of data sizes in different scenes. (a)-(c): “Train”, (d)-(f): “Truck”.



(a) GT

(b) Scaffold-GS
PSNR: 20.2dB
SSIM: 0.837
LPIPS: 0.214



(c) KAN-GS
PSNR: 20.7dB
SSIM: 0.849
LPIPS: 0.208

(d) QKAN-GS (Ours)
PSNR: 21.9dB
SSIM: 0.857
LPIPS: 0.204

Figure 3: Snapshots of the reconstructed “train” scene. (a) Ground Truth, (b) Scaffold-GS, (c) KAN-GS, (d) QKAN-GS.



(a) GT

(b) Scaffold-GS
PSNR: 25.9dB
SSIM: 0.848
LPIPS: 0.153



(c) KAN-GS
PSNR: 26.3dB
SSIM: 0.849
LPIPS: 0.156

(d) QKAN-GS (Ours)
PSNR: 26.3dB
SSIM: 0.853
LPIPS: 0.143

Figure 4: Snapshots of the reconstructed “truck” scene. (a) Ground Truth, (b) Scaffold-GS, (c) KAN-GS, (d) QKAN-GS.

Acknowledgment

optimizations of the quantum circuit design for enhanced rate-distortion performance.

This work was partly supported by JSPS KAKENHI Grant Number JP22H03582 and JST-ASPIRE Grant Number JPMJAP2432.

References

- [1] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. 2017. Quantum machine learning. *Nature* 549, 7671 (2017), 195–202.
- [2] Yihang Chen, Qianyi Wu, Weiyao Lin, Mehrtash Harandi, and Jianfei Cai. 2024. Hac: Hash-grid assisted context for 3d gaussian splatting compression. In *European Conference on Computer Vision*. Springer, 422–438.
- [3] Zhiwen Fan, Kevin Wang, Kairun Wen, Zehao Zhu, Dejia Xu, Zhangyang Wang, et al. 2024. Lightgaussian: Unbounded 3d gaussian compression with 15x reduction and 200+ fps. *Advances in neural information processing systems* 37 (2024), 140138–140158.
- [4] Edward Farhi and Hartmut Neven. 2018. Classification with quantum neural networks on near term processors. *arXiv preprint arXiv:1802.06002* (2018).
- [5] Ben Fei, Jingyi Xu, Rui Zhang, Qingyuan Zhou, Weidong Yang, and Ying He. 2024. 3d gaussian splatting as new era: A survey. *IEEE Transactions on Visualization and Computer Graphics* (2024).
- [6] Sharath Girish, Kamal Gupta, and Abhinav Shrivastava. 2024. Eagles: Efficient accelerated 3d gaussians with lightweight encodings. In *European Conference on Computer Vision*. Springer, 54–71.
- [7] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics* 42, 4 (July 2023). <https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/>
- [8] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. 2017. Tanks and Temples: Benchmarking Large-Scale Scene Reconstruction. *ACM Transactions on Graphics (ToG)* 36, 4 (2017), 1–13.
- [9] Joo Chan Lee, Daniel Rho, Xiangyu Sun, Jong Hwan Ko, and Eunbyung Park. 2024. Compact 3d gaussian representation for radiance field. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 21719–21728.
- [10] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljacic, Thomas Y Hou, and Max Tegmark. 2025. KAN: Kolmogorov–Arnold Networks. In *The Thirteenth International Conference on Learning Representations*.
- [11] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. 2024. Scaffold-GS: Structured 3D gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 20654–20664.
- [12] K Navaneet, Kossar Pourahmadi Meibodi, Soroush Abbasi Koohpayegani, and Hamed Pirsiavash. 2023. Compact3d: Compressing gaussian splat radiance field models with vector quantization. *arXiv preprint arXiv:2311.18159* 4 (2023).
- [13] Simon Niedermayr, Josef Stumpfegger, and Rüdiger Westermann. 2024. Compressed 3d gaussian splatting for accelerated novel view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10349–10358.
- [14] Panagiotis Papantonakis, Georgios Kopanas, Bernhard Kerbl, Alexandre Lanvin, and George Drettakis. 2024. Reducing the memory footprint of 3d gaussian splatting. *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 7, 1 (2024), 1–17.
- [15] Adrián Pérez-Salinas, Alba Cervera-Lierta, Elies Gil-Fuster, and José I Latorre. 2020. Data re-uploading for a universal quantum classifier. *Quantum* 4 (2020), 226.
- [16] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. 2015. An introduction to quantum machine learning. *Contemporary Physics* 56, 2 (2015), 172–185.
- [17] Sukin Sim, Peter D Johnson, and Alán Aspuru-Guzik. 2019. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Advanced Quantum Technologies* 2, 12 (2019), 1–18.
- [18] Shriyank Somvanshi, Syed Aaqib Javed, Md Monzurul Islam, Diwas Pandit, and Subasish Das. 2025. A survey on kolmogorov-arnold network. *Comput. Surveys* (2025).
- [19] Yufei Wang, Zhihao Li, Lanqing Guo, Wenhan Yang, Alex Kot, and Bihan Wen. 2024. Contextgs: Compact 3d gaussian splatting with anchor level context model. *Advances in neural information processing systems* 37 (2024), 51532–51551.
- [20] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.
- [21] Qi Yang, Kaifa Yang, Yuke Xing, Yiling Xu, and Zhu Li. 2024. A benchmark for gaussian splatting compression and quality assessment study. In *Proceedings of the 6th ACM International Conference on Multimedia in Asia*. 1–8.
- [22] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 586–595.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009