HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Qi Yang[1], Le Yang[2], Geert Van Der Auwera[3], Zhu Li[1]

qiyang@umkc.edu

[1]: University of Missouri-Kansas City

[2]: University of Canterbury

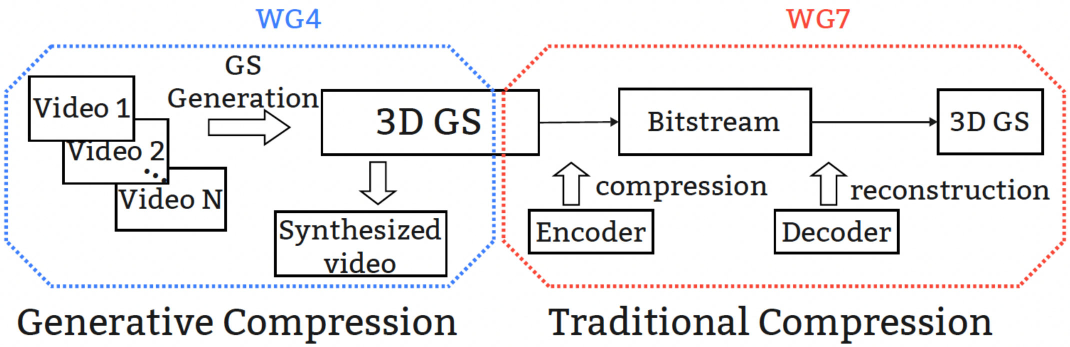
[3]: Qualcomm

Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Motivation

- GS compression has two different branches: generative compression and traditional compression
  - Generative compression: MPEG WG4
    - input/output: ground truth/synthetic images
    - 3D GS and its variants as intermediates, such as implicit feature domain
  - Traditional compression: MPEG WG7
    - Input/output: 3D GS
    - Using point cloud encoder to generate bitstream

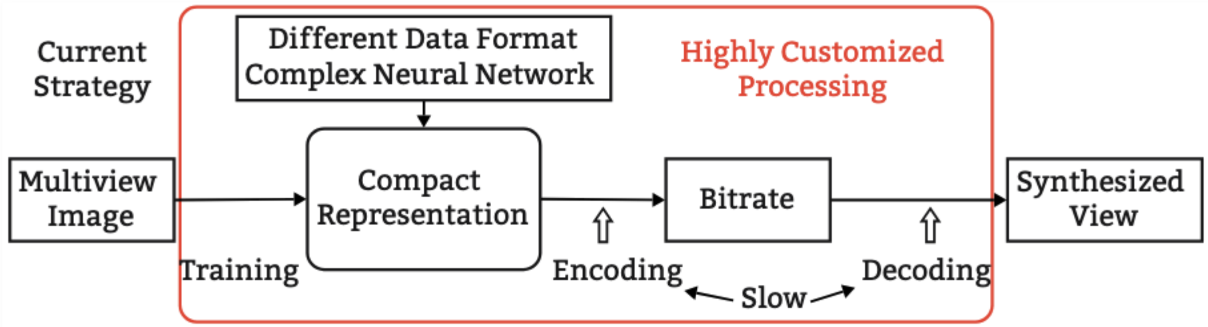


Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Motivation

- Generative compression
  - high compression ratio + high PSNR
  - Very slow encoding and decoding speed: operation on the implicit data field



- Traditional compression
  - High coding and decoding speed
  - Easy understanding and stable operation on the explicit data field
  - Low compression ratio and low PSNR

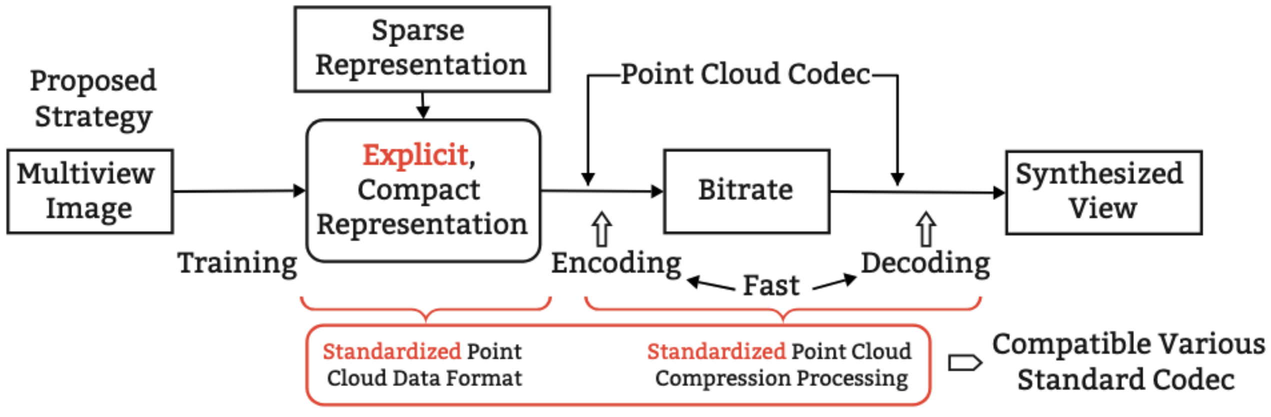
HybridGS

Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Motivation

- How about combining the above two methods?
- Therefore, we propose HybridGS
  - Generating compact and explicit 3D GS data first
  - Using a mature point cloud encoder to code

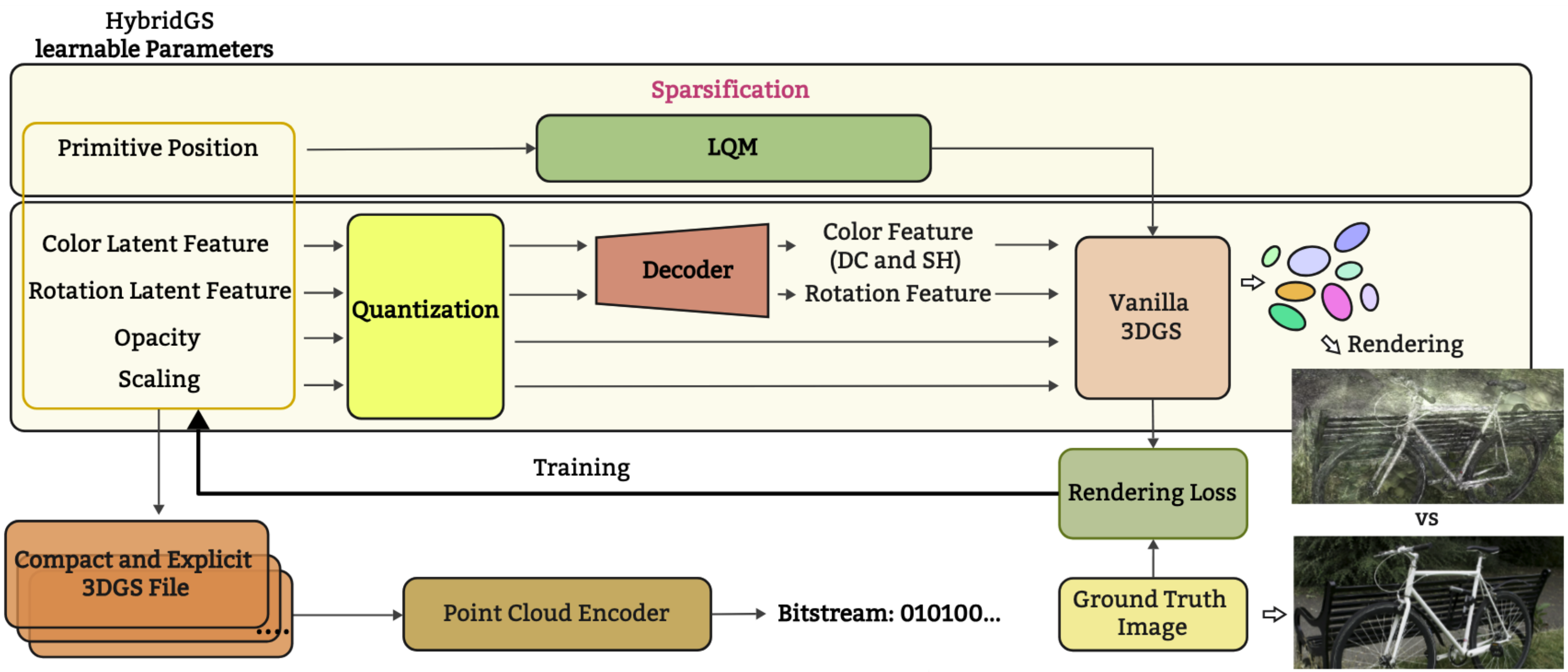


Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Framework

Dual-channel sparse representation: 1) sparse representation of feature; 2) primitive sparsification

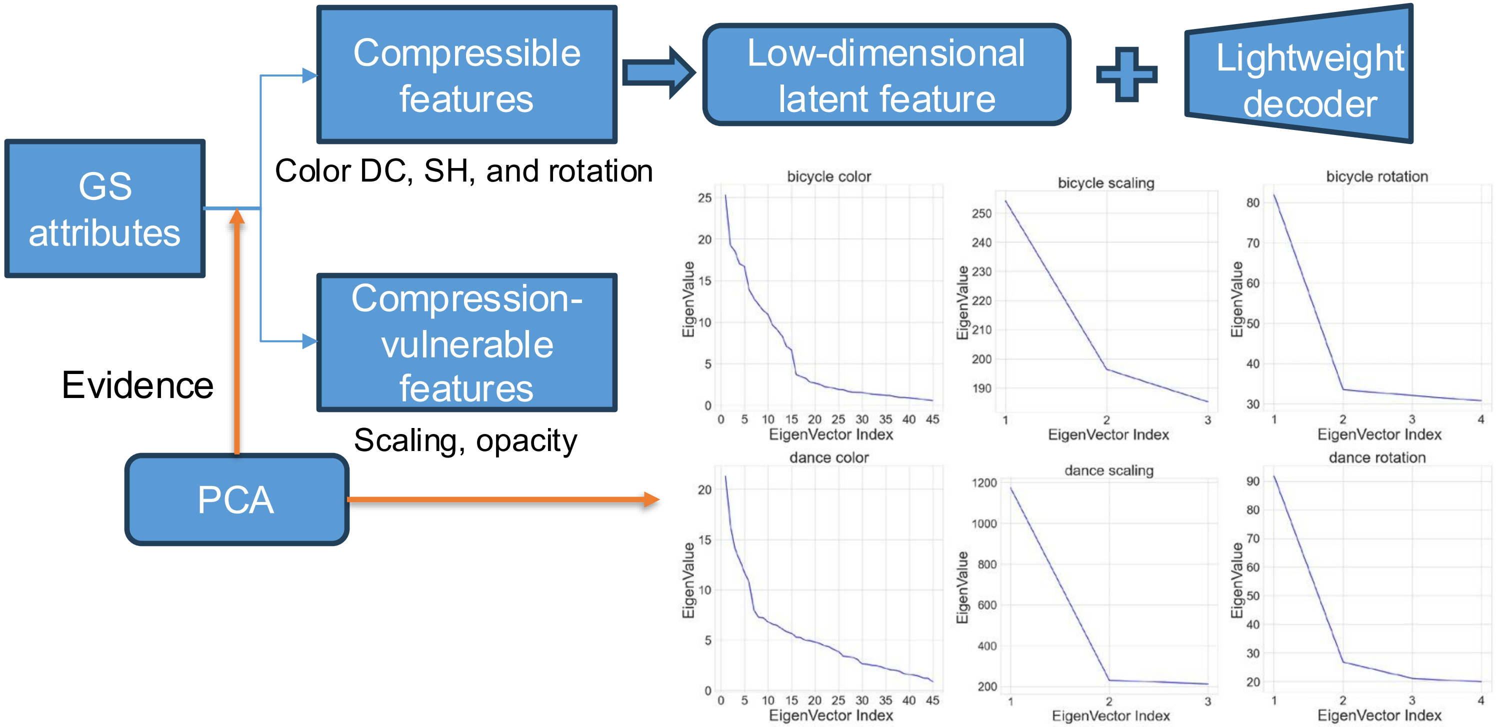


Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Framework

- Sparse representation of feature

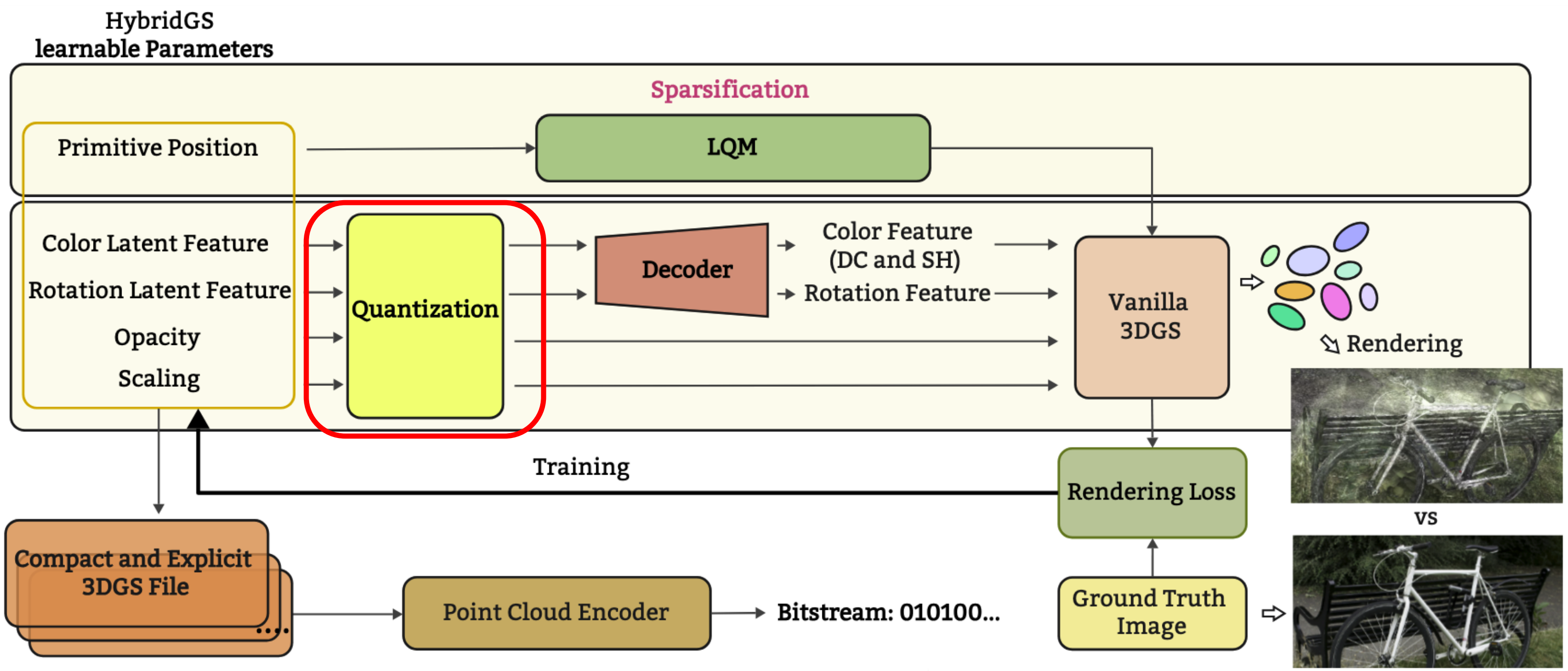


Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Framework

- Sparse representation of feature

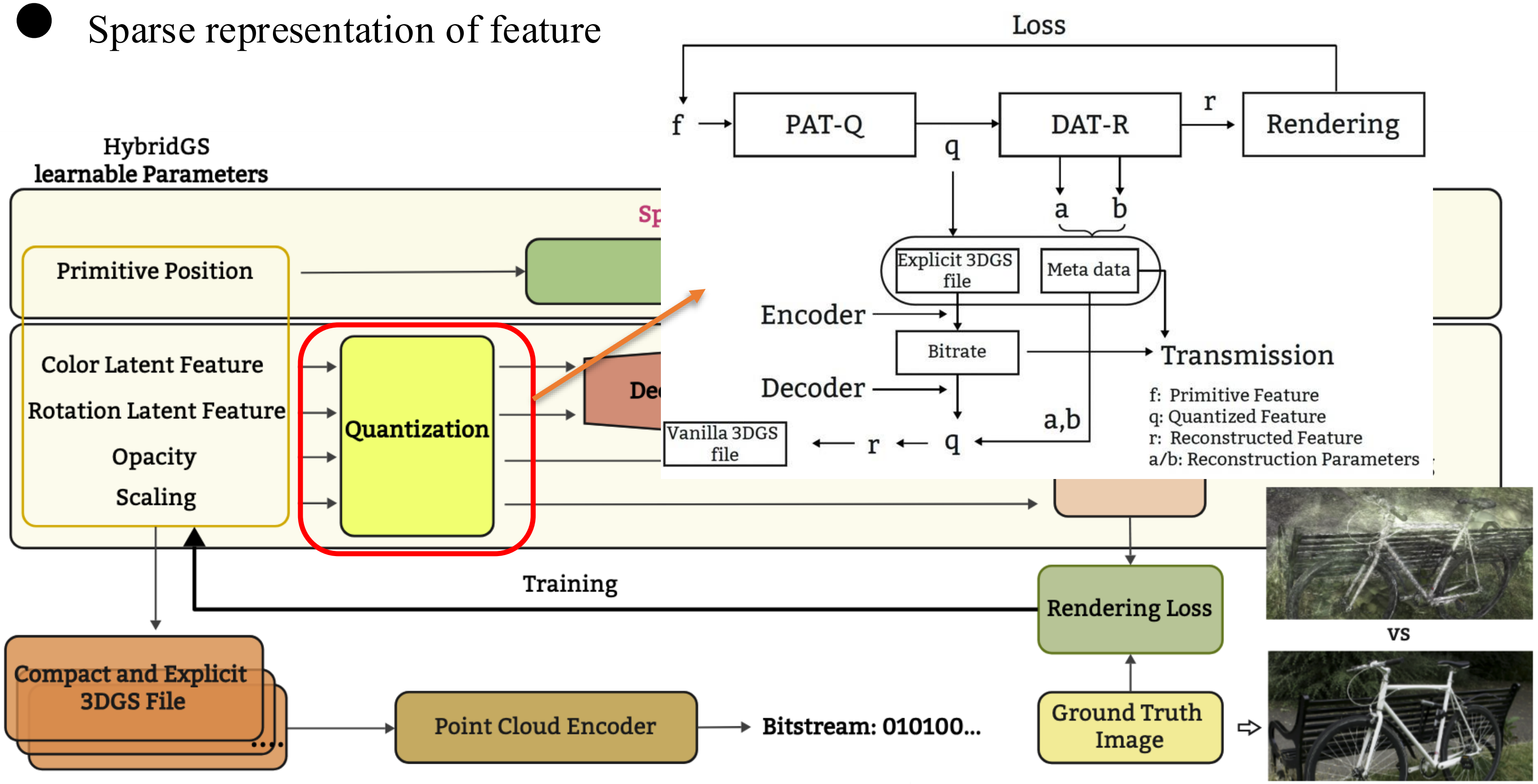


Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Framework

- Sparse representation of feature

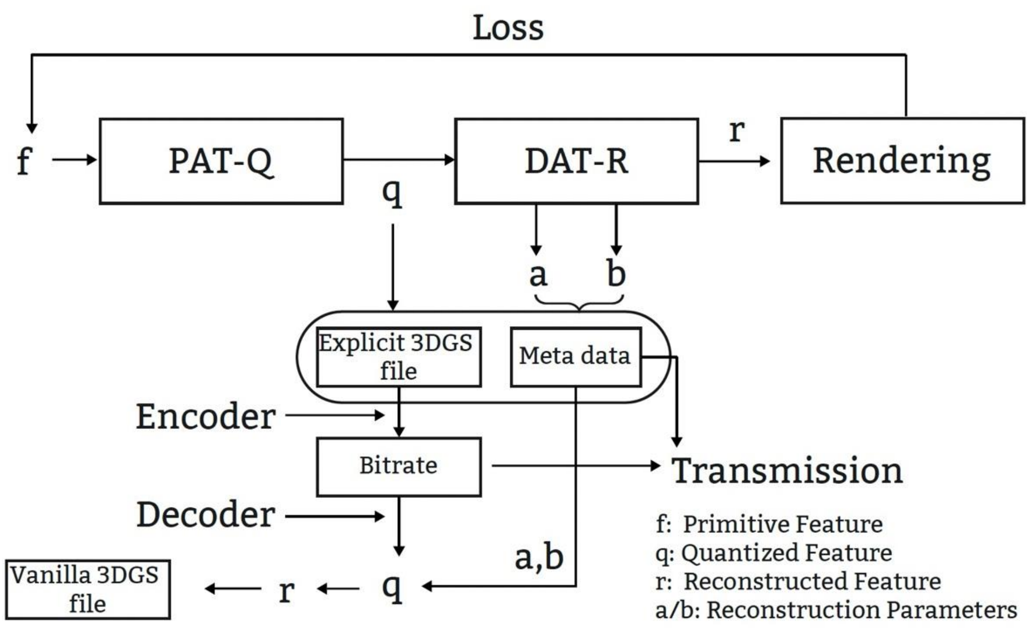


Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Framework

- Sparse feature Quantization [1]



•**PAT-Q:** the quantization of a feature f is formulated as

$$q = A(f) + \sigma, \quad A(f) = \frac{f - f_{min}}{f_{max} - f_{min} + \epsilon} \cdot (2^N - 1),$$
$$\sigma = round(A(f)) - A(f), \quad \sigma \in [-0.5, 0.5].$$

•**DAT-R:** the reconstruction (de-quantization) is cast into a ridge regression problem. The regularization factor λ is introduced, and the de-quantization is realized via solving

$$\min_{a,b} \frac{1}{2M}||a \cdot q + b - f||^2 + \frac{\lambda}{2}a^2,$$

where M is the dimension of f. Taking the partial derivatives with respect to a and b, and setting the results to zero yield the stationary points, which are

$$a = \frac{Cov_{fq}}{Var_q + \lambda}, \quad b = \bar{f} - a\bar{q},$$

where $\bar{}$, Cov, and Var represent the averaging, covariance, and variance operators. Finally, the reconstruction of the learnable feature is calculated using

$$r = R(q) = a \cdot q + b = \bar{f} + \frac{Cov_{fq}}{Var_q + \lambda}(q - \bar{q}).$$

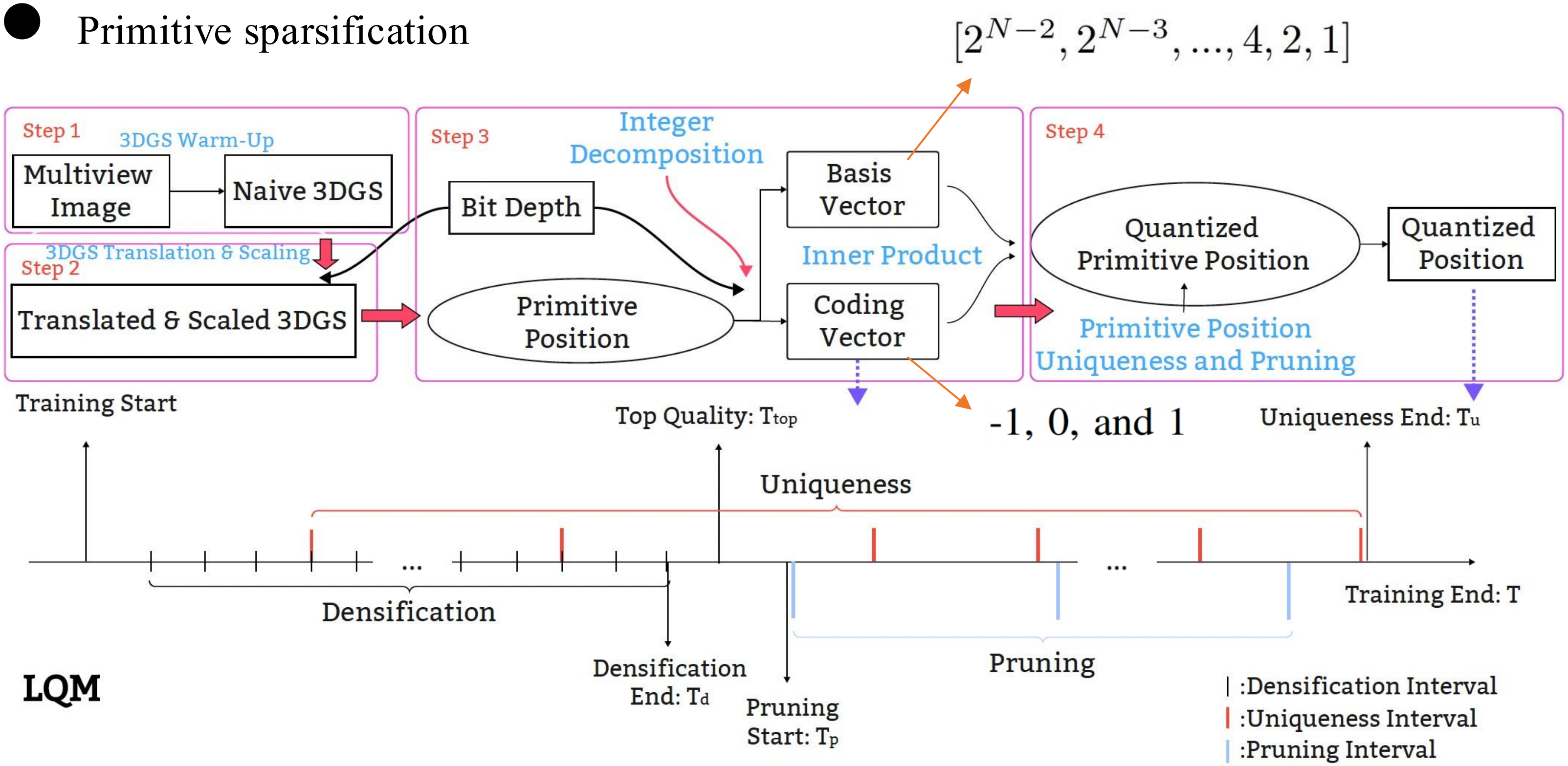
[1] Ye, Chengxi, et al. "Robust Training of Neural Networks at Arbitrary Precision and Sparsity." *arXiv preprint arXiv:2409.09245* (2024).

Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Framework

- Primitive sparsification



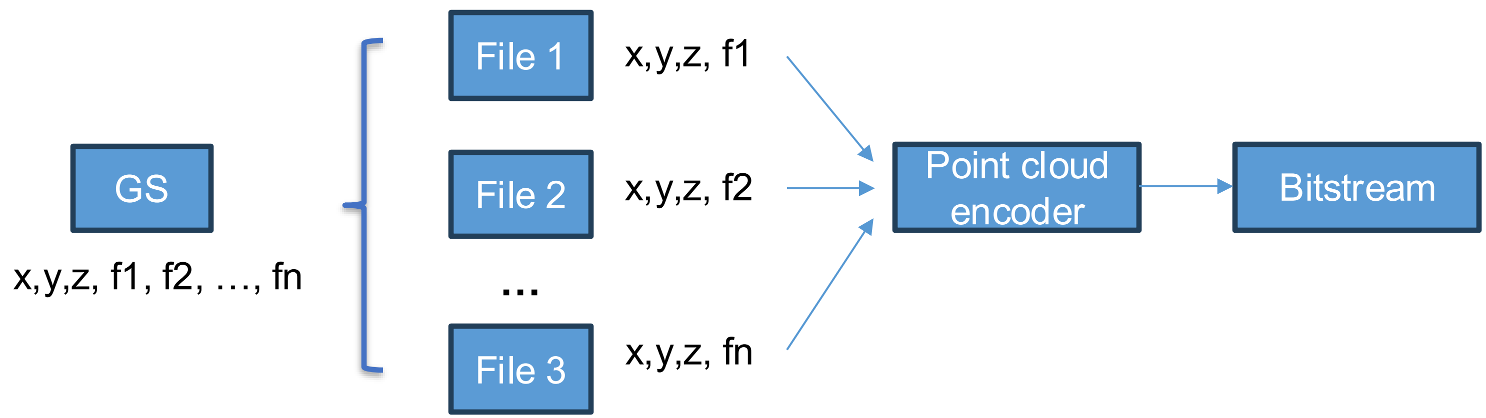
LQM

Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

High-efficiency coding

- Two implementations: 1) revise the point cloud encoder interface; 2) split explicit GS into multiple "xyzf" or "xyzrbg" file



Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

High-efficiency coding

- Two rate control methods

1) controlling primitive number

$$\max_{n} \quad Q(GS),$$

$$s.t., R(GS) \leq B \quad \text{and} \quad R(GS) = \frac{n \cdot P_{bit}}{L}$$

$$P_{bit} = 3 \cdot (BD_p + BD_s) + k_c \cdot BD_c + BD_o + k_r \cdot BD_r$$

2) adapting feature bit depth (BD)

$$\max_{\Delta} \quad Q(GS),$$

$$s.t., R(GS) \leq B \quad \text{and} \quad R(GS) = \frac{N \cdot (p_{bit} - \Delta p_{bit})}{L}$$

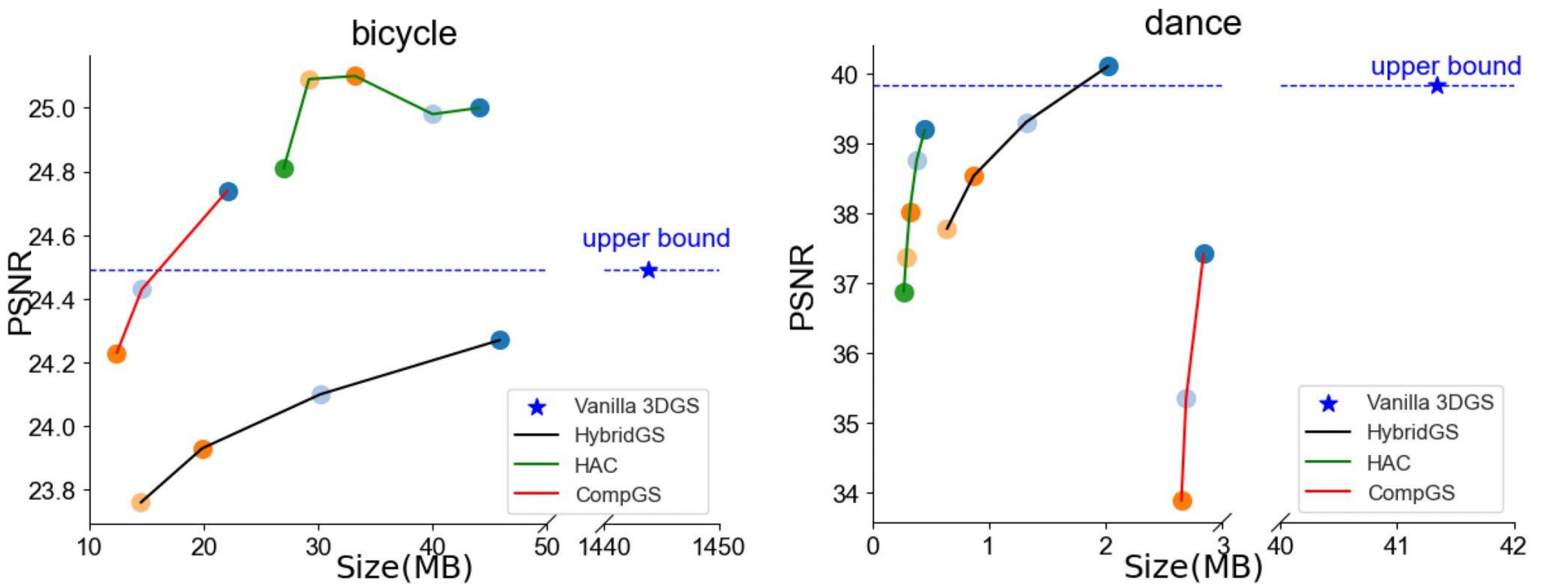
$$\Delta p_{bit} = \Delta \cdot (k_c + 1 + 3 + k_r),$$

Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Experiment

- Performance



Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Experiment

- Coding time

Method	bicycle	room
	Enc/Dec (second)	
HAC	85.03/80.09	17.04/15.90
CompGS(MM)	36.29/22.47	7.82/6.28
HGSC HR	132.32/72.13	35.65/16.45
HGSC LR	124.22/52.94	31.64/13.39
HybridGS HR	1.67/1.77	0.44/0.47
HybridGS LR	0.66/0.92	0.26/0.46
-position	0.09/0.08	0.03/0.04
-attribute	0.57/0.84	0.23/0.32

Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Experiment

- Rate allocation

Components	bicycle	dance
Total Size	22.88 MB	1.04 MB
-primitive number	1,286,284	56,490
-position	2.72MB (7.36)	0.16 MB (0.32)
-color (latent feature)	6.39 MB (7.36)	0.28 MB (0.32)
-opacity	2.40 MB (2.45)	0.11 MB (0.11)
-scaling	7.07 MB (7.36)	0.31 MB (0.32)
-rotation (latent feature)	4.30 MB (4.91)	0.18 MB (0.22)
-metadata	4 KB	4 KB
-color decoder weights	13 KB	13 KB
-rotation decoder weights	4 KB	4 KB

Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Experiment

- Rate control

Dataset	**train**			
Type	Method 1		Method 2	
Target Rate	PSNR	Real Rate	PSNR	Real Rate
4 MB	21.13	3.96	18.20	2.20
6 MB	21.60	5.82	21.54	5.15
8 MB	21.44	7.56	21.75	7.33
10 MB	21.54	8.59	21.75	9.86
	dance			
0.5 MB	37.40	0.54	*	*
1 MB	38.99	1.06	38.80	0.77
1.5 MB	39.76	1.59	39.86	1.40
2 MB	39.90	2.07	39.96	1.95

HybridGS

Introduction

HybridGS: High-Efficiency Gaussian Splatting Data Compression using Dual-Channel Sparse Representation and Point Cloud Encoder

Conclusion

- We propose a new 3DGS compression method, HybridGS, which first generates an explicit and compact 3DGS file and then uses canonical point cloud encoders to realize high-efficient coding and flexible rate control
- HybridGS reports comparable performance with SOTA methods and faster encoding and decoding (from over 1 min to 0.4s-1.7s)
- HybridGS demonstrates characteristics of interpretability, compatibility, and alignment with the demands of standardization

Limitation

The optimal compression efficiency of HybridGS is lower than end-to-end methods