

SuperGS: Super-Resolution 3D Gaussian Splatting via Latent Feature Field and Gradient-guided Splitting

概要

- タスクはHRNVS(High Resolution Novel View Synthesis)
 - 低画質画像を入力として高画質の新視点画像を生成するタスク
- SuperGS(Super Resolution 3DGS)を提案
- 事前学習済みの低解像度のシーン表現を初期化として利用して2段階のcoarse to fine training frameworkを採用
- MFGS(Multi-resolution Feature Gaussian Splatting)を提案
 - latent feature fieldを活用して任意の位置でガウス特徴をサンプリング可能に
- GSS(Gradient-guided Selective Splitting)を提案
 - プリミティブを適切にアップサンプリングして一貫性のある詳細な3D表現で高解像度シーンをレンダリング

HRNVSの課題

- 普通の3DGSでは性能が大幅に低下
- ガウスプリミティブを直接アップサンプリングすることが困難
- 低解像度の画像で最適化されたプリミティブは高解像度レンダリングにおいて粗すぎる
- detailの欠如が目立つ
- 各ガウスにSHパラメータを関連付けるとメモリ消費量が膨大
- view consistency

関連研究

NVS(Novel View Synthesis)

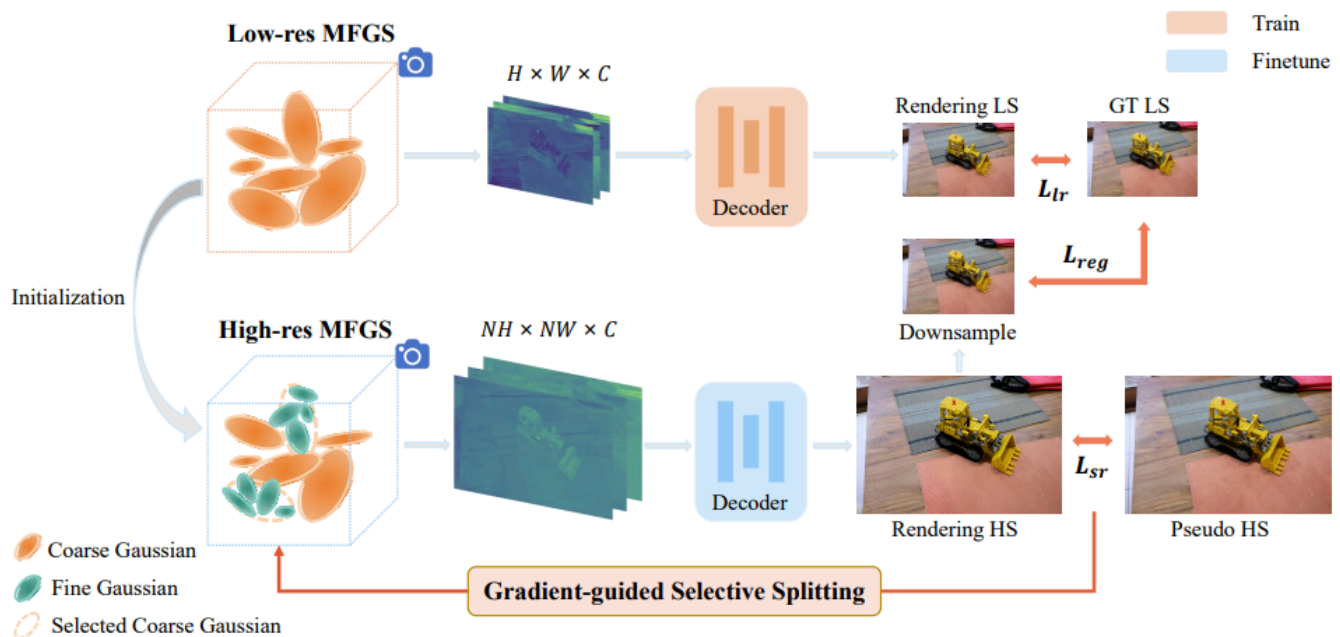
- NeRF
 - Mip-NeRF
 - InstantNGP

- 3DGS

3D Scene Super Resolution, HRNVS(High Resolution Novel View Synthesis)

- NeRF-SR
- RefSR-NeRF
- GaussianSR
- SRGS

SuperGS



(論文より引用)

MFGS(Multi-resolution Feature Gaussian Splatting)

- InstangNGPから思いついたらしい
- 複数の解像度のグリッドで特徴量ベクトルを取得する
- 下図のようにハッシュテーブルをインデックスとして利用して線形補間を行うことで任意の位置でGaussian featuresにアクセスできる
- 各Gaussianに対して異なる解像度のグリッドに格納された学習可能な特徴ベクトルを取得して線形補間を実行することでその特徴表現を導出する
- 異なる解像度のグリッドはそれぞれ個別のハッシュテーブルに対応している

- 今、グリッドの解像度が L レベルまでとする
- i 番目のGaussianにおける l 番目のレベルで補完された特徴ベクトルを f_i^l とする
- このときgaussianの多解像度特徴ベクトル表現(multi resolution feature vector representation)は以下

$$f_{i_vi} = f_i^1 \otimes f_i^2 \otimes \cdots \otimes f_i^L$$

- このときグリッドに収められるように位置ベクトルはcontracted spaceにnormalizeする
 - Mip-NeRF360の手法

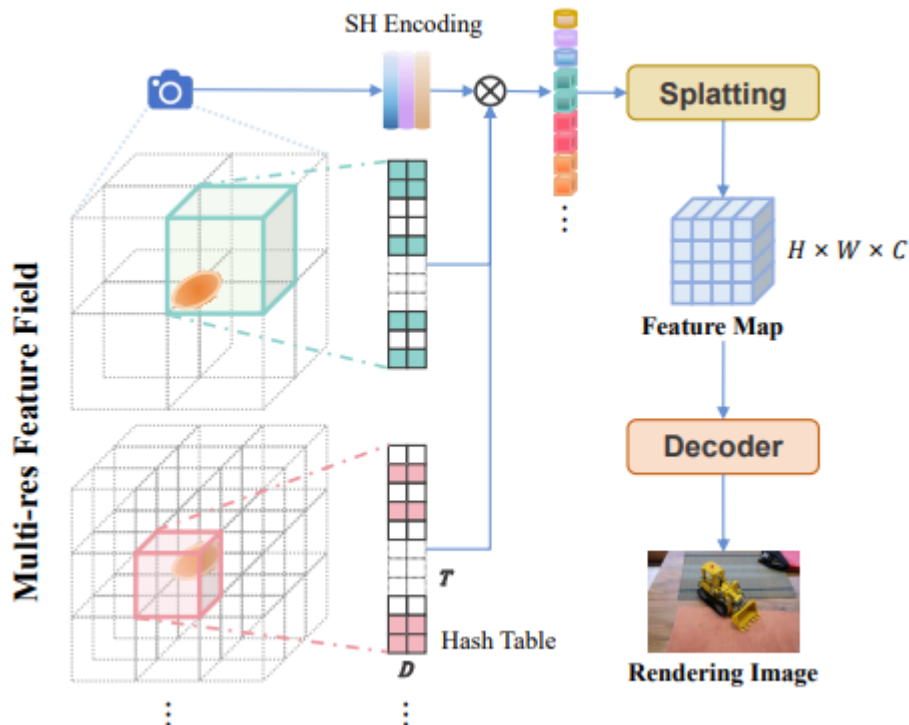
$$\text{contract}(p_i) = \begin{cases} p_i, & \|p_i\| \leq 1, \\ \left(2 - \frac{1}{\|p_i\|}\right) \left(\frac{p_i}{\|p_i\|}\right), & \|p_i\| \geq 1. \end{cases}$$

- $p_i \in \mathbb{R}^3$ は i 番目のGaussianの位置
- \otimes はconcatenation
- d_i はdirection
- 最後にview dependent featureを得るためにSH関数を用いてencodeして2層のMLPに入れる

$$f_i = \text{MLP}(f_{i_vi} \otimes \text{SH}(d_i))$$

- カメラポーズ v が与えられたときにピクセル p のfeatureは以下

$$F_{v,p} = \sum_{i \in N} f_i a_i \prod_{j=1}^{i-1} 1 - a_j$$

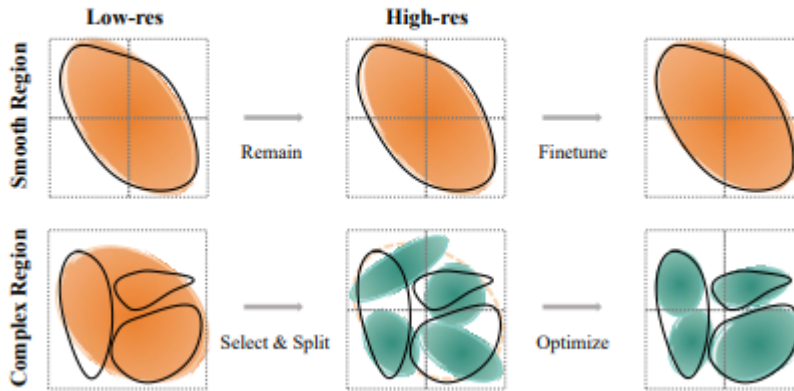


(論文より引用)

GSS(Gradient-guided Selective Splitting)

- smoothな領域ではGaussianを固定してComplexな領域ではgaussianを増やすアルゴリズム

3.2 Gradient-guided Selective Splitting



- 勾配とスケールがしきい値を超えるとcomplexとみなしてgaussianを増やす
- 視点の個数を M , k 番目の視点におけるlossを \mathcal{L}^k , ピクセル空間上のgaussianの座標を $u_i^k = (u_{i,x}^k, u_{i,y}^k)$ とする
- このとき100 training iterationごとに以下のaverage view space positional gradientを計算する

$$\nabla_{u_i} \mathcal{L} = \frac{1}{M} \sum_{k=1}^M \sqrt{\left(\frac{\partial \mathcal{L}^k}{\partial u_{i,x}^k} \right)^2 + \left(\frac{\partial \mathcal{L}^k}{\partial u_{i,y}^k} \right)^2}$$

- このとき判定基準は以下

$$\nabla_{u_i} \mathcal{L} > \tau_p \quad \text{AND} \quad \|S_i\| > \tau_S$$

- アルゴリズムは以下
 - 祖なgaussianが G_{coarse}
 - 増やされるのが G_{fine}
 - splitting number N_s は G_{fine} に分割するときの個数であり、super resolutionの倍率に比例させて設定する

Algorithm 1: Gradient-guided selective splitting

Input: Feature Field \mathcal{F} , Splitting Number N_s , G_{coarse} **Output:** \mathcal{F} , G_{coarse} , G_{fine}

```
1:  $G_{fine} \leftarrow \emptyset$ 
2: for  $i \leftarrow 0$  to Coarse2FineIteration do
3:    $\mathcal{L} \leftarrow \text{UpdateLoss}()$ 
4:   if IsSplitIteration( $i$ ) then
5:     for  $\mathcal{G}_j \in G_{coarse}$  do
6:       if  $\nabla_{\mu_j} \mathcal{L} > \tau_\mu$  and  $\|S_j\| > \tau_s$  then
7:          $G_{N_s}(p', \Sigma', \alpha', f') = \text{SplitCoarse}(p, \Sigma, \alpha, \mathcal{F})$ 
8:          $G_{fine} = G_{fine} \cup G_{N_s}$ 
9:          $G_{coarse} = G_{coarse} - \mathcal{G}_j$ 
10:      end if
11:    end for
12:  end if
13: end for
14: DensifyFine( $G_{fine}$ )
15: Finetune( $G_{coarse}$ ,  $\mathcal{F}$ )
```

cross-view consistency constraint

- lossはSRGSの提案手法で以下

$$\begin{aligned}\mathcal{L} &= (1 - \lambda_{reg}) \mathcal{L}_{hr} + \lambda_{reg} \mathcal{L}_{reg} \\ \mathcal{L}_{hr} &= (1 - \lambda_{ssim}) \mathcal{L}_1 \left(I_{hr}^{render}, I_{hr}^{pseudo} \right) + \lambda_{ssim} \mathcal{L}_{D-SSIM} \left(I_{hr}^{render}, I_{hr}^{pseudo} \right) \\ \mathcal{L}_{reg} &= (1 - \lambda_{ssim}) \mathcal{L}_1 \left(I_{lr}^{render}, I_{lr}^{gt} \right) + \lambda_{ssim} \mathcal{L}_{D-SSIM} \left(I_{lr}^{render}, I_{lr}^{gt} \right)\end{aligned} \quad (1)$$

- I_{lr}^{gt} は低画質画像の正解
- I_{hr}^{pseudo} は I_{lr}^{gt} を2D pretrained SRモデルに入力したときの出力
- I_{hr}^{render} はこのモデルの出力
- I_{lr}^{render} は出力をダウンサンプリングしたもの

results

- データセットは13個
 - Mip-NeRF360 : 9
 - Deep Blending : 2

◦ Tanks&Temples : 2

- 比較対象は以下

Dataset Method&Metric	Mip-NeRF360 ×2			Mip-NeRF360 ×4			Deep Blending ×2			Deep Blending ×4		
	PSNR	SSIM	Mem	PSNR	SSIM	Mem	PSNR	SSIM	Mem	PSNR	SSIM	Mem
3DGS	23.41	0.727	598	20.38	0.582	475	27.88	0.863	395	24.80	0.769	410
3DGS-SwinIR	24.77	0.708	662	23.47	0.605	539	27.56	0.846	459	25.65	0.789	474
Scaffold-GS	23.37	0.723	156	20.30	0.584	134	28.79	0.885	48	27.51	0.862	49
Scaffold-SwinIR	24.83	0.703	220	23.49	0.600	198	28.09	0.854	112	27.45	0.838	113
GaussianSR	-	-	-	25.60	0.663	-	-	-	-	28.28	0.873	-
SRGS	26.24	0.768	559	25.16	0.710	464	28.98	0.885	427	28.80	0.875	437
Ours	26.69	0.766	203	25.52	0.716	204	29.61	0.889	101	29.21	0.880	100

英単語

- autonomous navigation : 自律航行
- coarse : 粗い、粗な
- opacity : 不透明度
- vanilla : 基本的な
- bypass : 迂回する
- granularity : 粒度
- pseudo : 疑似
- discrepancy : 不一致
- isotropic : 等方性
- anisotropic : 異方性(方向によって異なる)
- notably : 特に
- anchor : 錨(いかり)(船をとめるもの)