Mip-Splatting复现过程记录

Mip-Splatting复现小记

Paper：2311.16493.pdf (arxiv.org)

Code：GitHub - autonomousvision/mip-splatting: Mip-Splatting: Alias-free 3D Gaussian Splatting

步骤

Step1：拉取code

```
git clone git@github.com:autonomousvision/mip-splatting.git
```

Step2：数据集

DataSet1：NeRF官方数据集nerf_synthetic文件夹 - Google 云端硬盘（1.56GB）

预处理数据：

```
python convert_blender_data.py --blender_dir nerf_synthetic/ --out_dir multi-scale
处理后数据输出到Multi-scale文件夹
```

DataSet2：Mip-NeRF360mip-NeRF 360 (jonbarron.info)

Step3：环境配置

```
①创建conda环境
conda create -y -n mip-splatting python=3.8
②激活
conda activate mip-splatting
③安装适合自己电脑的pytorch和cuda
pip install torch==1.12.1+cu113 torchvision==0.13.1+cu113
conda install cudatoolkit=11.3 -c conda-forge
④安装剩余环境
pip install -r requirements.txt
pip install submodules/diff-gaussian-rasterization
pip install submodules/simple-knn/
```

Step4：训练和预测

```
# single-scale training and single-scale testing on NeRF-synthetic dataset
python scripts/run_nerf_synthetic_stmt.py

# multi-scale training and multi-scale testing on NeRF-synthetic dataset
python scripts/run_nerf_synthetic_mtmt.py

# single-scale training and single-scale testing on the mip-nerf 360 dataset
python scripts/run_mipnerf360.py

# single-scale training and multi-scale testing on the mip-nerf 360 dataset
python scripts/run_mipnerf360_stmt.py
```

我直接运行脚本控制台一直等待，无任何输出

检查GPU状态和cuda　都可用

直接用下面的命令直接手动训练

```
训练train.py
set OMP_NUM_THREADS=4
set CUDA_VISIBLE_DEVICES=0,1
python train.py -s multi-scale/lego -m output/lego --eval --white_background --port 6209 --kernel_size 0.1
#输入是预处理的场景multi-scale/lego
#输出是训练过的点云数据output/lego

渲染render.py
python render.py -m {output_dir}/{scene} -r {scale} --data_device cpu --skip_train

评估指标metrics.py
python metrics.py -m {output_dir}/{scene}
```

训练过程：

```
Training progress:  96%|                                        | 28800/30000 [13:27<00:33, 36.21it/s, Loss=0.0047218]Computing 3D filter [25/01 23:06:01]
Training progress:  96%|                                        | 28900/30000 [13:29<00:29, 36.86it/s, Loss=0.0047575]Computing 3D filter [25/01 23:06:04]
Training progress:  97%|                                        | 29000/30000 [13:32<00:27, 36.35it/s, Loss=0.0045416]Computing 3D filter [25/01 23:06:07]
Training progress:  97%|                                        | 29100/30000 [13:35<00:24, 36.65it/s, Loss=0.0054599]Computing 3D filter [25/01 23:06:09]
Training progress:  97%|                                        | 29200/30000 [13:38<00:21, 36.43it/s, Loss=0.0040968]Computing 3D filter [25/01 23:06:12]
Training progress:  98%|                                        | 29300/30000 [13:41<00:19, 36.39it/s, Loss=0.0044441]Computing 3D filter [25/01 23:06:15]
Training progress:  98%|                                        | 29400/30000 [13:43<00:16, 36.19it/s, Loss=0.0047864]Computing 3D filter [25/01 23:06:18]
Training progress:  98%|                                        | 29500/30000 [13:46<00:13, 36.44it/s, Loss=0.0044415]Computing 3D filter [25/01 23:06:21]
Training progress:  99%|                                        | 29600/30000 [13:49<00:10, 36.73it/s, Loss=0.0042073]Computing 3D filter [25/01 23:06:23]
Training progress:  99%|                                        | 29700/30000 [13:52<00:08, 36.67it/s, Loss=0.0042502]Computing 3D filter [25/01 23:06:26]
Training progress:  99%|                                        | 29800/30000 [13:55<00:05, 36.30it/s, Loss=0.0047833]Computing 3D filter [25/01 23:06:29]
Training progress: 100%|                                        | 30000/30000 [14:00<00:00, 35.69it/s, Loss=0.0061895]

[ITER 30000] Evaluating test: L1 0.007192170043417185 PSNR 32.85328057527542 [25/01 23:06:38]

[ITER 30000] Evaluating train: L1 0.0032505741342902187 PSNR 39.7521469116211 [25/01 23:06:38]

[ITER 30000] Saving Gaussians [25/01 23:06:38]

Training complete. [25/01 23:06:39]

(mip-splatting) G:\project\mip-splatting>
```


评估过程：

```
(mip-splatting) G:\project\mip-splatting>python metrics.py -m output/lego
Setting up [LPIPS] perceptual loss: trunk [vgg], v[0.1], spatial [off]
D:\WorkStation\Anaconda\envs\mip-splatting\lib\site-packages\torchvision\models\_utils.py:208: UserWarning: The parameter 'pretrained' is deprecated since 0.13 and will be removed in 0.15, please use 'weights'
instead.
  warnings.warn(
D:\WorkStation\Anaconda\envs\mip-splatting\lib\site-packages\torchvision\models\_utils.py:223: UserWarning: Arguments other than a weight enum or `None` for 'weights' are deprecated since 0.13 and will be remov
ed in 0.15. The current behavior is equivalent to passing `weights=VGG16_Weights.IMAGENET1K_V1`. You can also use `weights=VGG16_Weights.DEFAULT` to get the most up-to-date weights.
  warnings.warn(msg)
Downloading: "https://download.pytorch.org/models/vgg16-397923af.pth" to C:\Users\Favor/.cache\torch\hub\checkpoints\vgg16-397923af.pth
100%|                                                            | 528M/528M [00:20<00:00, 27.0MB/s]
Loading model from: D:\WorkStation\Anaconda\envs\mip-splatting\lib\site-packages\lpips\weights\v0.1\vgg.pth

Scene: output/lego
Method: ours_30000
Metric evaluation progress: 100%|                                 | 800/800 [09:48<00:00,  1.36it/s]
  SSIM :    0.9823919
  PSNR :   32.7552834
  LPIPS:    0.0162020


(mip-splatting) G:\project\mip-splatting>
```


输出至output文件夹

3D 平滑滤波器融合到高斯参数

```
python create_fused_ply.py -m {model_dir}/{scene} --output_ply fused/{scene}_fused.ply"
```

viewer ：Mip-Splatting Viewer Examples (niujinshuchong.github.io)（google打开）

上传ply文件即可渲染

渲染结果