

3D Gaussian Splatting (3DGS) 完整部署与展示指南

一、文档目的

本指南用于完整记录一次从零开始的 3D Gaussian Splatting (3DGS) 部署、训练与展示流程，强调环境可复现性、每一步的作用以及生成文件的意义。

二、整体流程概览

整体流程分为五个阶段：

1. 视频抽帧生成图片
2. COLMAP 计算相机位姿与稀疏点云
3. 3DGS 训练高斯表示
4. 新视角渲染与结果查看
5. Web 交互式 Demo 展示

三、环境配置（已验证）

操作系统：Ubuntu 24.04

GPU：NVIDIA RTX 5090（计算能力 sm_120）

CUDA：12.8

PyTorch：Nightly / Dev（支持 sm_120）

Python：3.11

Conda 环境名：3dgs_5090

四、Conda 与 PyTorch 安装

```
conda create -n 3dgs_5090 python=3.11 -y
```

```
conda activate 3dgs_5090
```

```
pip install --pre torch torchvision torchaudio --index-url  
https://download.pytorch.org/whl/nightly/cu128
```

验证：`torch.cuda.get_device_capability(0)` 应返回 (12, 0)

五、3DGS 代码获取

```
git clone https://github.com/graphdeco-inria/gaussian-splatting.git  
cd gaussian-splatting  
git submodule update --init --recursive
```

六、Python 依赖安装

```
pip install -r requirements.txt  
pip install plyfile opencv-python tqdm imageio
```

若提示 `ModuleNotFoundError`, 直接安装对应库即可。

七、CUDA 扩展编译（关键步骤）

```
cd submodules/diff-gaussian-rasterization  
pip install -v . --no-build-isolation --no-cache-dir
```

这里需要单独从 git 上下载 zip 解压该文件，不然会不全

```
cd ../simple-knn  
pip install -v . --no-build-isolation --no-cache-dir
```

注意：必须在当前 PyTorch 环境下编译，否则会出现 `undefined symbol` 错误。

八、数据准备（视频抽帧）

使用 `ffmpeg` 将视频转为图片序列：

```
ffmpeg -i video.mp4 -qscale:v 2 data/images/frame_%05d.png
```

生成目录：data/images/

九、COLMAP 输出说明

COLMAP 输出位于 data/sparse/0：

- cameras.bin：相机内参
- images.bin：相机外参（位姿）
- points3D.ply：稀疏几何点云（仅用于初始化）

十、3DGS 训练

训练命令示例：

```
python train.py -s data -m data/output --resolution 2
```

核心输出：

```
data/output/point_cloud/iteration_xxx/point_cloud.ply
```

十一、PLY 文件区别

COLMAP 的 points3D.ply 表示几何稀疏点云；

3DGS 的 point_cloud.ply 表示高斯外观（颜色、透明度、尺度），不是几何模型。

十二、结果查看方式

render.py：离线渲染指定相机轨迹，适合生成视频；

PlayCanvas Gaussian Viewer：实时交互查看 3DGS 场景，适合展示。

十三、常见问题说明

图像发白、模糊通常由相机视角超出训练分布引起，并非模型或数据错误。

十四、展示建议

推荐组合：

1. PlayCanvas 交互式 Demo
2. 原始视频与新视角渲染对比
3. 简要说明 3DGS 为外观重建方法