3D无限场景的生成

第一周

图形学最基础知识，包括相机标定，colmap，sfm等方法

三维重建基础

LucidDreamer

论文解读

In this work, we propose a pipeline called LucidDreamer that utilizes Stable Diffusion and 3D Gaussian splatting to create diverse high-quality 3D scenes from various types of inputs such as text, RGB, and RGBD.

1. **局限**：dataset far form real world（特定领域）
    a. 利用stable diffusion生成3D模型，难以保证multi-view consistency
    b. diffusion模型学习：

    diffusion模型基础
2. **Step: dream and alignment**
    a. 投影点云，利用生成模型。根据深度提升三维空间
    b. Alignment algorithm：integrate
    c. 作为3dgs的inital points

简单来说，初始图像投影之后用diffusion模型生成，重新升维到3维空间后，用alignment算法和原先的点云群融合

3. **输入**：初始图像和深度图
4. **输出**：点云
5. **优点**：多场景，配合文字输入，生成更多视角
6. **具体过程**：
    a. 初始化点云，利用stable diffusion得到imag和深度图（深度图的生成根据zero-depth the monocular depth estimation model），把二维图像升维到三维
        i. $P_0 = \phi_{2\to3}([I0, D0], K, P0)$
    b. 聚合点云：把生成的点云和原来的点云聚合成更大的点云，需要满足一致性

c. dream过程：投影的image会有没有得到的点，因此利用mask来表示填充和没有填充的点，在没有填充的点继续进行stable diffusion和monocular depth的过程

i. 实际过程中，深度图可能会不一致（因为模型缺陷），所以需要optimal depth

$I_i = \mathcal{S}(\hat{I}, M_i), \hat{D}_i = \mathcal{D}(I_i), D_i = d_i \hat{D}_i$ 新的图像I，是从旧图像和mask利用S来生成的，depth图也可以用来生成，di是系数来控制

$$d_i = \arg\min_d \left(\sum_{M_i=1} \left\| \phi_{2\to 3}([I_i, d\hat{D}_i], K, P_i) - P_{i-1} \right\|_1 \right)$$ 这个是用来逼近di

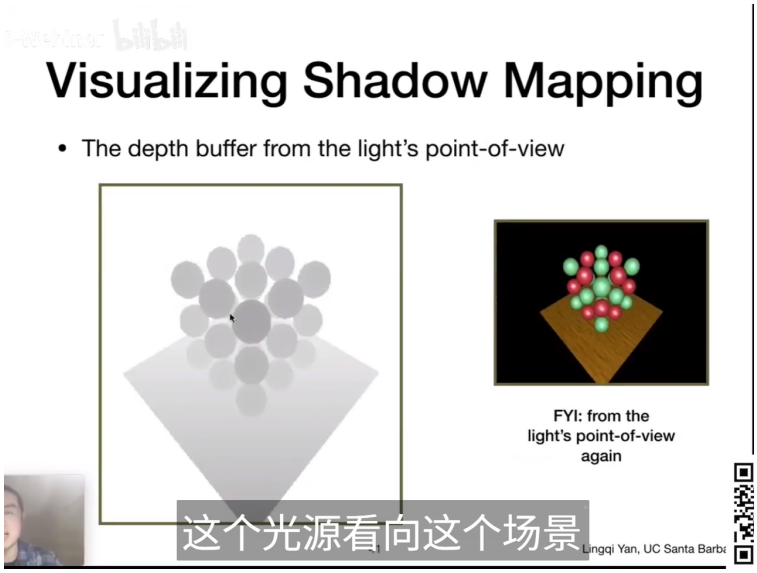
d. alignment：dream的过程是同时生成深度图和新图像，用off the shelf 的深度图生成方法会更好（适配更多的场景，更加的精确）

i. 需要解决consistency的问题，引入alignment算法。因为没有考虑多个depth之间的关系，直接移动点，让之间smooth。（需要计算vector），naive的移动会导致distort，需要利用差值算法解决

图形学基础（有差值算法）

i. 具体的过程：将图片沿着射线移动（因为深度是这个方向），找到最接近Pi-1位置的点，并计算深度改变的幅度（因为移动）

ii. 对于没有真实图像的地方，用线性差值的方法解决（深度吐的图片）



iii. 重复整个过程，就可以完成整个深度图的构造（其实整个alignment，就是在找更精确的di）

Algorithm 1: Constructing point cloud

Input: A single RGBD image $[\mathbf{I}_0, \mathbf{D}_0]$
Input: Camera intrinsic $\mathbf{K}$, extrinsics $\{\mathbf{P}_i\}_{i=0}^{N}$
Output: Complete point cloud $\mathcal{P}_N$

$\mathcal{P}_0 \leftarrow \phi_{2\to3}\left([\mathbf{I}_0, \mathbf{D}_0], \mathbf{K}, \mathbf{P}_0\right)$
for $i \leftarrow 1$ **to** N **do**
  $\hat{\mathbf{I}}_i, \mathbf{M}_i \leftarrow \phi_{3\to2}\left(\mathcal{P}_{i-1}, \mathbf{K}, \mathbf{P}_i\right)$
  $\mathbf{I}_i \leftarrow \mathcal{S}\left(\hat{\mathbf{I}}_i, \mathbf{M}_i\right), \hat{\mathbf{D}}_i \leftarrow \mathcal{D}\left(\mathbf{I}_i\right)$
  $d_i \leftarrow 1$
  while *not converged* **do**
    $\tilde{\mathcal{P}}_i \leftarrow \phi_{2\to3}\left(\left[\mathbf{I}_i, d_i\hat{\mathbf{D}}_i\right], \mathbf{K}, \mathbf{P}_i\right)$
    $\mathcal{L}_d \leftarrow \frac{1}{\|\mathbf{M}_i=1\|}\sum_{\mathbf{M}_i=1}\left\|\tilde{\mathcal{P}}_i - \mathcal{P}_{i-1}\right\|_1$
    Calculate $\nabla_d \mathcal{L}_d$
    $d_i \leftarrow d_i - \alpha\nabla_d\mathcal{L}_d$
  end
  $\mathbf{D}_i \leftarrow d_i\hat{\mathbf{D}}_i$
  $\hat{\mathcal{P}}_i \leftarrow \phi_{2\to3}\left([\mathbf{I}_i, \mathbf{D}_i|\mathbf{M}_i = 0], \mathbf{K}, \mathbf{P}_i\right)$
  $\mathcal{P}_i \leftarrow \mathcal{P}_{i-1} \cup \mathcal{W}\left(\hat{\mathcal{P}}_i\right)$
end

e. 最后就是利用3dgs高斯来完成训练和渲染（整个luciddreammer其实关注的就是初始化点云）
   i. 值得注意的是：高斯损失函数只关注有真实结果的，mask=0的地方不会计入损失函数

高斯学习相关知识

7. **疑问：**
   a. 相机的内参和外参如何得到：The camera intrinsic matrix and the extrinsic matrix of I0 are denoted as K and P0, respectively. For the case where I0 and D0 are generated from the diffusion model, we set the values of K and P0 by convention regarding the size of the image.
   b. 高感知质量是什么：high perceptual quality
   c. 训练的时候添加M照片是为了什么（实验经验吗）：For the images to train the model, we use additional M images as well as (N + 1) images for generating the point cloud, since the initial (N + 1) images are not sufficient to train the network for generating the plausible output. The M new images and the masks are generated by reprojecting from the point cloud PN by a new camera sequence of length M, denoted as PN+1, ..., PN+M.
   d. alingnment移动点，为什么就可以解决一致性的问题，直接使用差值不可以么

8. **和项目的关系**：luciddreammer是利用stable diffusion和3dgs的整体框架，可以利用stable diffusion和深度图来提供多视角，新的三维场景。但是我们的项目是想要做超分，相当于并不是实现横向的新场景添加，是要实现纵向的超分效果。提供了一种思路，就是利用stable生成点云和alignment进行结合。因为lucciddreamer本身有提供框架和代码，所以在luciddreamer上结合power of ten的思路，可以尝试实现。

Power of ten

论文解读

We achieve this through a joint multi-scale diffusion sampling approach that encourages consistency across different scales while preserving the integrity of each individual sampling process.

our method enables deeper levels of zoom than traditional super-resolution methods that may struggle to create new contextual structure at vastly different scales

1. **优点**：相比传统的超分结构，在contextual上，上下文结构上具有更好的效果。现有的文生图模型，无法实现在zoom level上的consistent
    a. 传统的超分仅仅依靠图像信息，因此无法再deeper上实现细节，但是我们有text promot可以解决这个问题
    b. 每个scale上的plausible image，同时每个scale之间又是连续
    c. 传统的超分也可以实现zoom in和zoom out的效果，但是上下文关系薄弱，递归的时候会产生问题
    d. 同时生成整个consistent sequence
    e. 总的来说，我希望在zoom in和zoom out过程中，整个语音可以被保持
2. **难点**：zoom in的问题是语义semantic：因为我们生成的图片之间应该要有很多的过度关系（比如一个人的手掌放大会有皱纹）
3. **输入**：text提示词（会包括不同的scale的提示词）
4. **过程**：
    a. 先介绍了diffusion模型生成图像的过程，逐步添加高斯噪声来生成图片（train），如果train完了，就需要利用denooise的过程，来得到干净的图像
        i. 我对stable diffusion模型的学习 diffusion模型的核心原理就是预测噪声，利用噪声消除来得到
    b. 目标：利用生成模型生成一组图片，图片之间的在不同的 zoom level上（大图片经过裁剪），能够保持consist

consistent way. This means that the image $\mathbf{x}_i$ at any specific zoom level p_i, should be consistent with the center $H/p \times W/p$ crop of the zoomed-out image $\mathbf{x}_{i-1}$.

c. 因此提出了两个方法：multi-scale joint sampling和 zoom stack representation