上海交通大学
SHANGHAI JIAO TONG UNIVERSITY

SJTU

3D内容生成与重建

机器学习工程实践

2024年10月22日

饮 水 思 源 • 爱 国 荣 校

课程分组

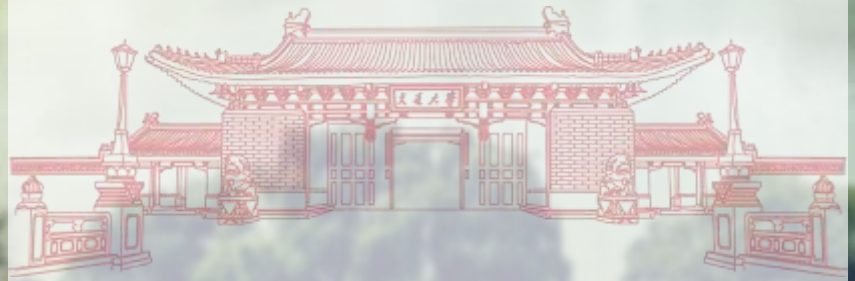
上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目 录

01

背景

Background



研究背景

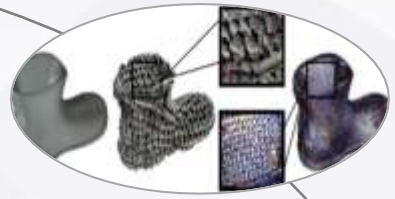
图像的生成与编辑已经取得了长足进展。

自从Diffusion model出现以来，大规模模型在多种任务上产生了爆炸效果。这是在2020以前想都不敢想的。

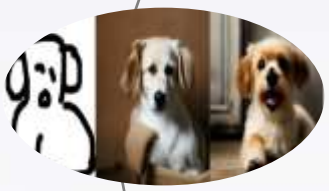
研究背景

线稿绘图

游戏人像建模



材质预测



草稿绘图

Stable Diffusion

纹理生成

人物姿态绘图

三维物体生成

语义绘图

研究背景

Microsoft

Copilot

智能办公
文档分析
幻灯片制作

OpenAI

DALL-E2

绘图模型
多场景优化

Adobe

FireFly

智能设计
内容生成
风格转换

Stable Diffusion

D-ID

D-ID

视频创作
人像生成
数字人

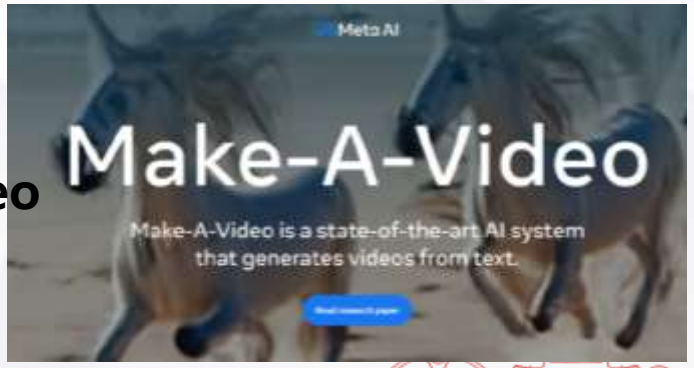
MidJourney

绘图机器人
真实感强

Meta AI

Make-A-Video

图-视频转换
视频创作



研究背景

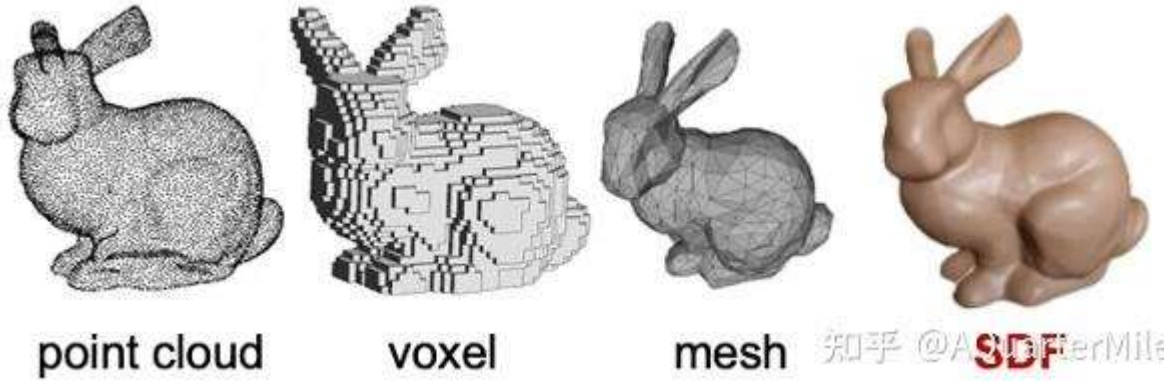
图像的生成与编辑能取得进展得益于三个因素：

- **大规模的图-文配对数据集（数以亿计）【基础】**
- **适应于大规模训练的模型（diffusion model）**
  - 语义上结构简单（省去encoder）
  - 非对抗性的损失函数
  - 训练稳定
- **PEFT技术**（ Parameter-Efficient Fine-Tuning ）

3D生成与重建的特征

3D数据

- 格式多样
- 维度高
- 计算开销大
- 数据收集难度高



- **与2D同等规模的图-文配对数据集？没有！**
- **→通用的3D生成模型？没有！**

3D生成与重建的特征

David Marr在他的视觉理论中把**计算机视觉终极问题**定义为：**输入二维图像，输出是由二维图像“重建”出来的三维物体的位置与形状。**而其他的一些我们现在常称为CV的任务，比如识别、检测等等，在Marr的理论中只能称作**“模式识别”**（Pattern Recognition）问题，不能被称作**“计算视觉”**（Computer Vision）

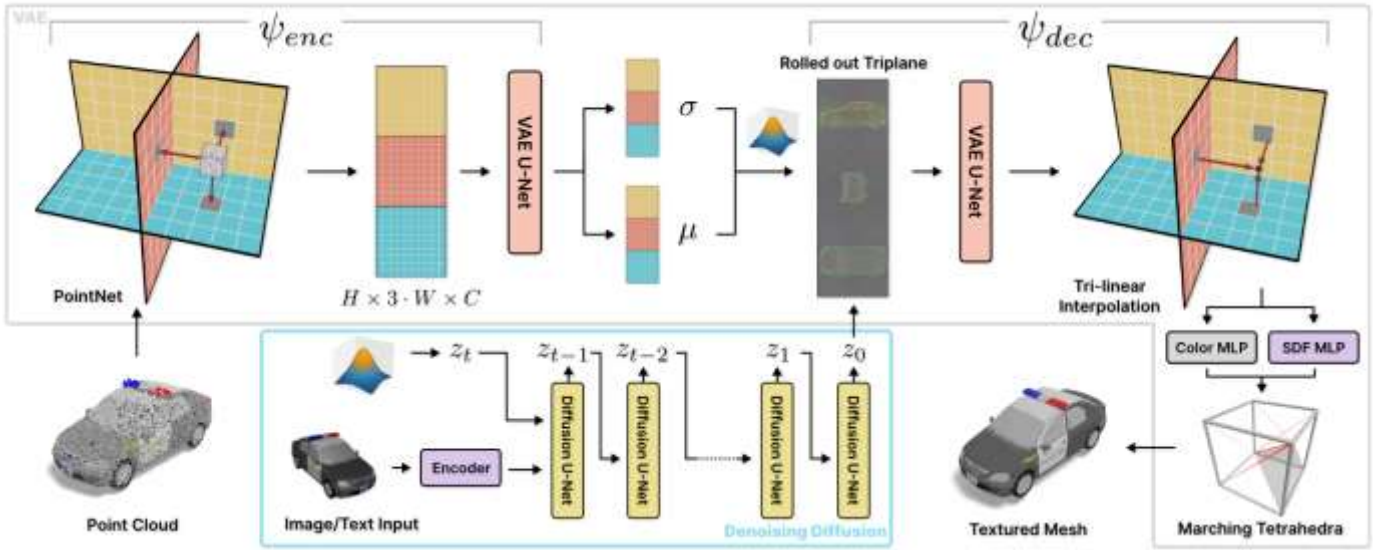
3D生成与重建的方法

3D内容生成与重建主要分成两种方法：

1. 【基于训练】在有限数据集上训练一个GAN、 Diffusion Model等的生成模型。
2. 【基于优化】对每个场景单独优化出一组参数来表征3D结构。

【基于训练】

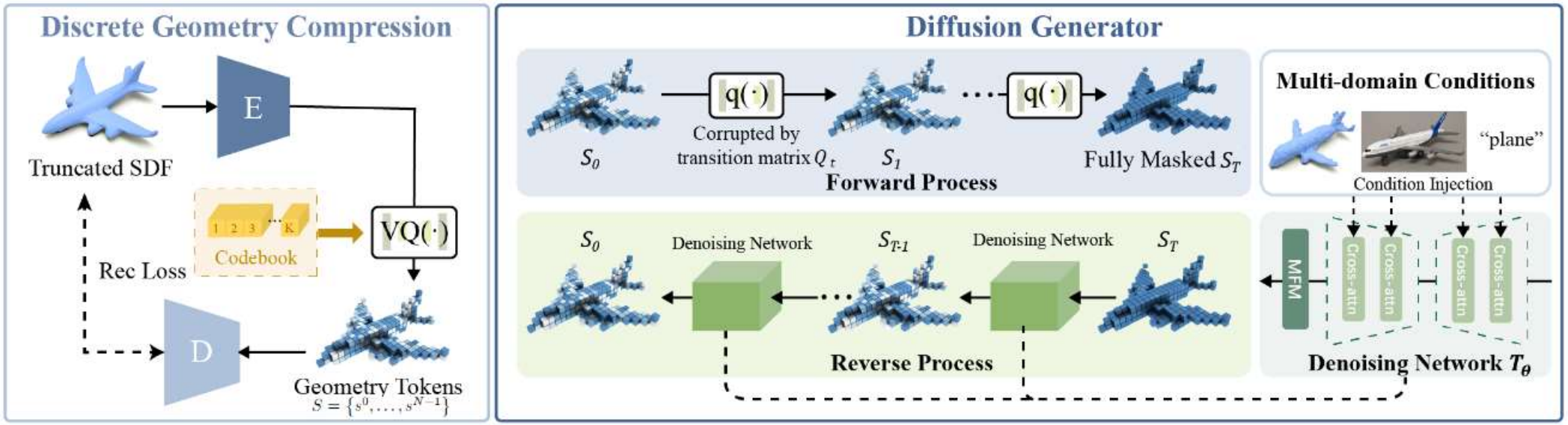
基于训练的方法，可以总结为：输出数据—处理为规则的数据格式—插入生成模型—解码形成3D结构



【1】3DGen: Triplane Latent Diffusion for Textured Mesh Generation

【基于训练】

基于训练的方法，可以总结为：输出数据—处理为规则的数据格式—插入生成模型—解码形成3D结构



【1】3DQD: Generalized Deep 3D Shape Prior via Part-Discretized Diffusion Process

【基于训练】

基于训练的方法：可以理解为2D的生成模型拓展到了3D上

要学习的是网络参数，网络一训练好，生成不同形状就不需要训练了

优势：1. 生成快(0.1s~10s per shape)

缺点：1. 类别受局限，只能生成自己见过的类似的物体

（ShapeNet椅子飞机等）

2. 不是通用的prior

不适应于通用的任务

【1】3DQD: Generalized Deep 3D Shape Prior via Part-Discretized Diffusion Process

【基于优化】

基于优化的方法：对每个形状/3D场景用一组参数表征，

要学习的不是网络参数，而是物体对应的表征参数

每次重建新的3D物体，都要花几十分钟重新优化

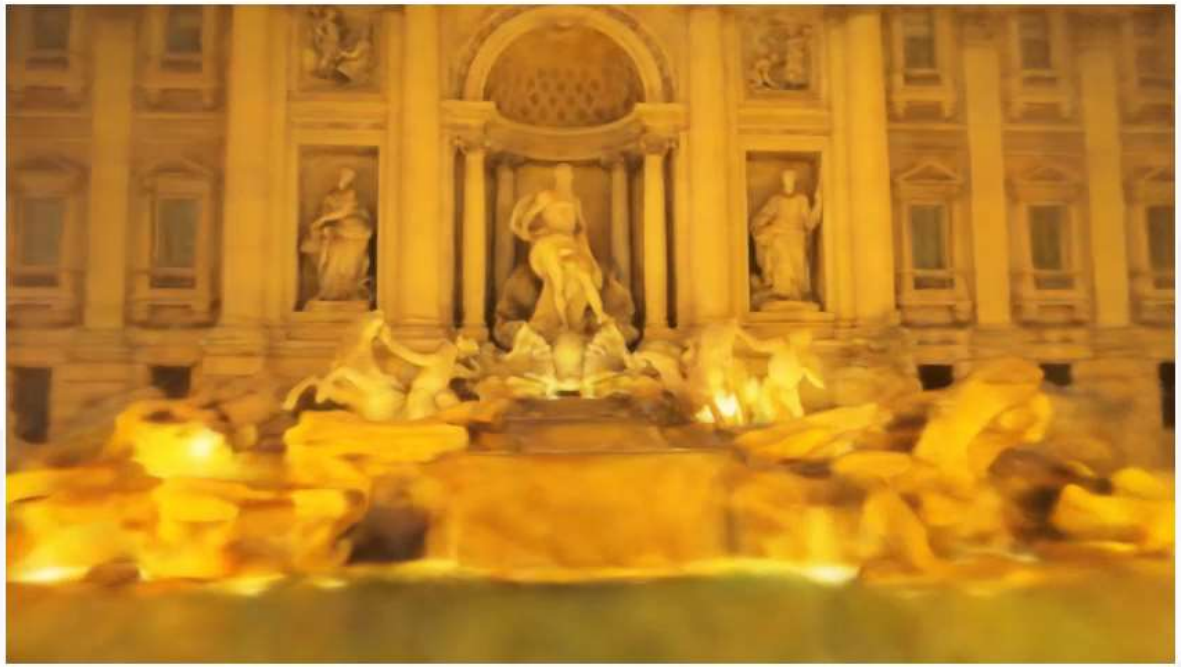
缺点：1. 生成慢(每个物体都要单独优化)

优点：1. 支持生成整个3D场景

2. 各种物体都可以用来优化，通用的

【1】 Fantasia3D: Disentangling Geometry and Appearance for High-quality Text-to-3D Content Creation

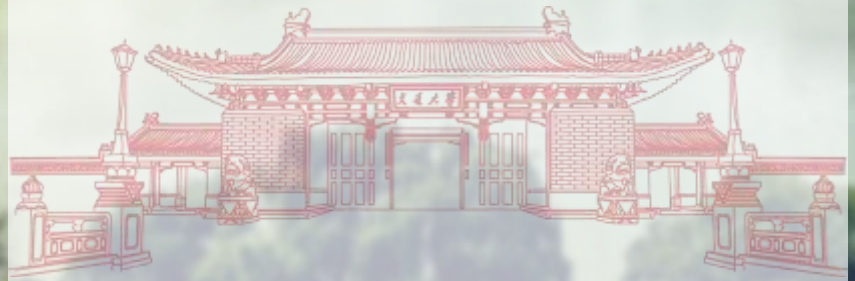
【基于优化】



02

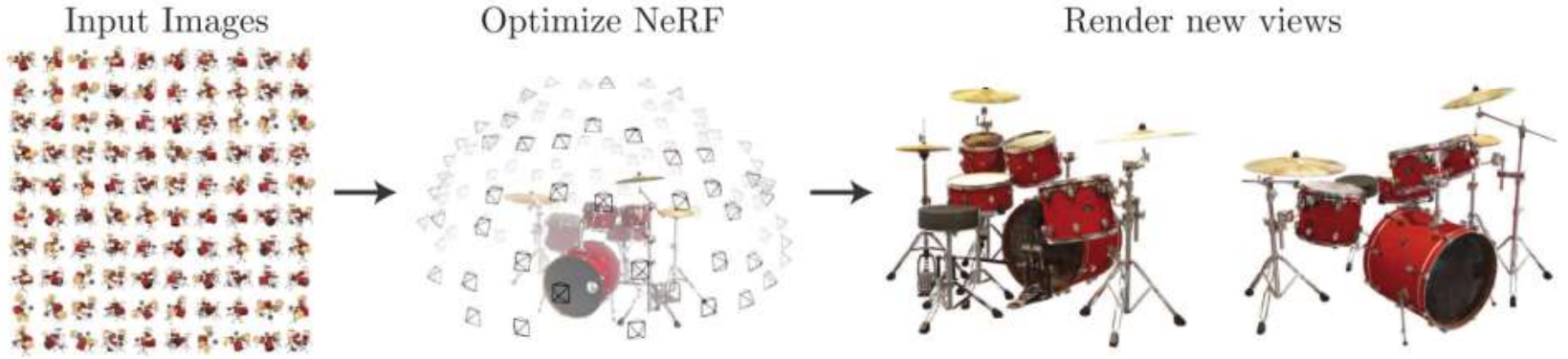
多视角重建

Multi-view Reconstruction



多视角重建

任务要求：对于一个场景，给出N张（几十张）它不同视角下的图片，及其对应的视角参数。优化一个3D结构，这个3D结构可以表征这个场景，并能给出新的视角下，这个场景的渲染图片.

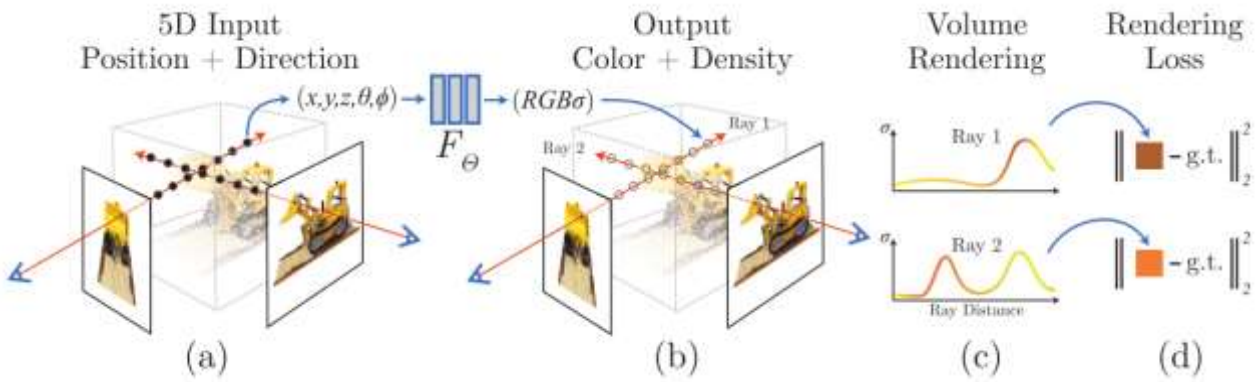


【1】 Representing Scenes as Neural Radiance Fields for View Synthesis

多视角重建

思路：设计一种3D结构，该3D结构按对应相机参数渲染出图像，与给定的真实图像计算损失，反向传播优化，使3D结构满足所需

结构　　渲染　　监督



【1】 Representing Scenes as Neural Radiance Fields for View Synthesis

NeRF的3D表征结构

神经辐射场（Neural Radiance Field, NeRF）

假设空间中每个点都有两个属性：**体密度sigma，与颜色color。**

怎么得到每个点对应的属性呢？假设有一个函数，这个场景表示为一个 **5D 向量值函数（vector-valued function）**：

- 输入是 3D 位置 $\mathbf{x} = (x, y, z)$ 和 2D 视角方向 (θ, ϕ)
- 输出是发射颜色 $\mathbf{c} = (r, g, b)$ 和体积密度 σ。

我们用一个 MLP 网络来近似这个连续的 5D 场景表示。 $F_\Theta : (\mathbf{x}, \mathbf{d}) \rightarrow (\mathbf{c}, \sigma)$

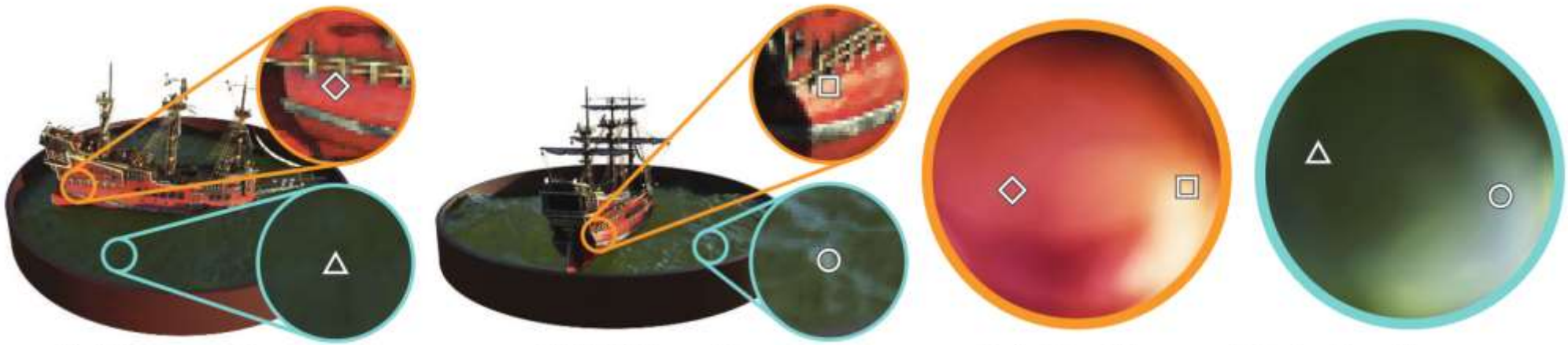
鼓励场景多视角连续

首先，我们希望这个场景表征是多视角连续的。什么是多视角连续？不断移动相机，物体是连贯的。同一个部位在不同视角下是一致的。（书包正着看和侧着看）

NeRF约束**预测体积密度 σ 的网络的输入仅仅是位置 X**，而**预测 RGB 颜色 C 的网络的输入是位置和视角方向**，鼓励场景多视角连续

如何理解？不同视角下的同一部位，不同高光导致颜色可能变，但物体几何不变！

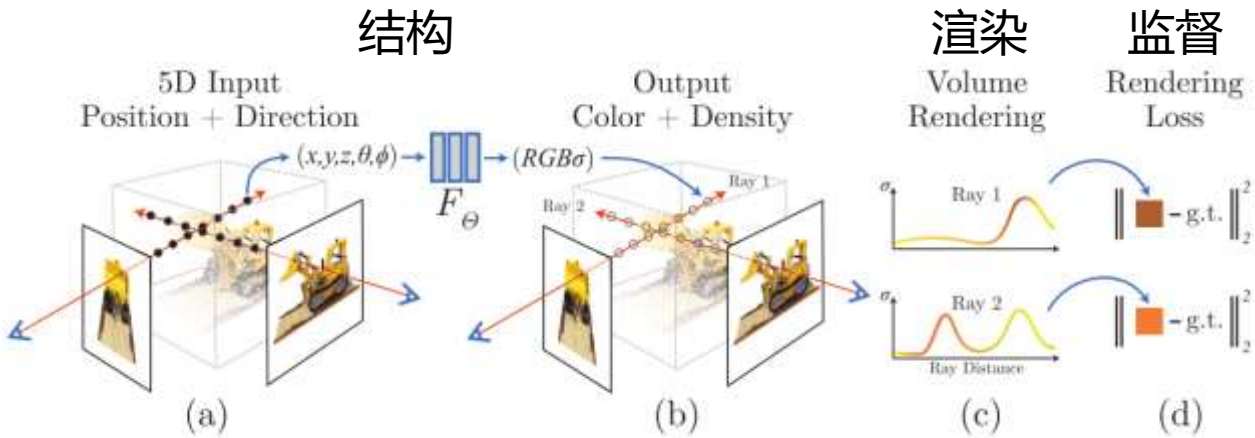
鼓励场景多视角连续



(a) View 1 (b) View 2 (c) Radiance Distributions

NeRF

思路：设计一种3D结构，该3D结构按对应相机参数渲染出图像，与给定的真实图像计算损失，反向传播优化，使3D结构满足所需



【1】 Representing Scenes as Neural Radiance Fields for View Synthesis

NeRF的3D表征结构

•在具体实现上，MLP 首先用 8 层的全连接层（使用 ReLU 激活函数，每层有 256 个通道），处理 3D 坐标 x，得到 σ和**一个 256 维的特征向量**。

•这个 256 维的特征向量，**与视角方向一起拼接起来**，喂给另一个全连接层（使用 ReLU 激活函数，每层有 128 个通道），输出方向相关的 RGB 颜色。

体渲染

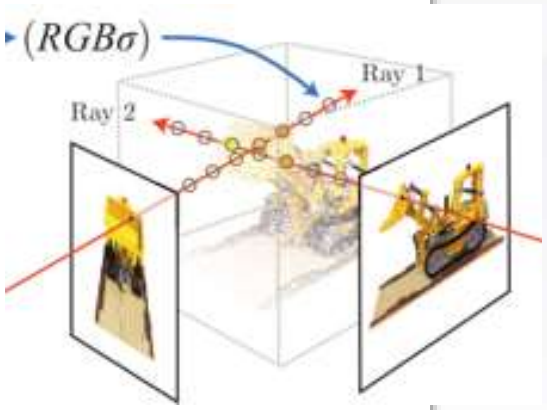
通过上面的设计，我们用一组MLP（辐射场）来表征了3D空间的物体结构。具体来说，空间中每个点都有两个属性：**体密度sigma，与颜色color。**
但这样的物体结构如何产生我们人眼看到的图片呢？

传统技术 体渲染！
使用经典的体渲染的原理，我们可以渲染出任意射线穿过场景的颜色。体积密度 σ(x) 可以解释为：**光线停留在位置 x 处的无穷小粒子的可导概率**

体渲染

使用经典的体渲染的原理，我们可以渲染出任意射线穿过场景的颜色。体积密度 σ(x) 可以解释为：**光线停留在位置 X 处的无穷小粒子的可导概率。**

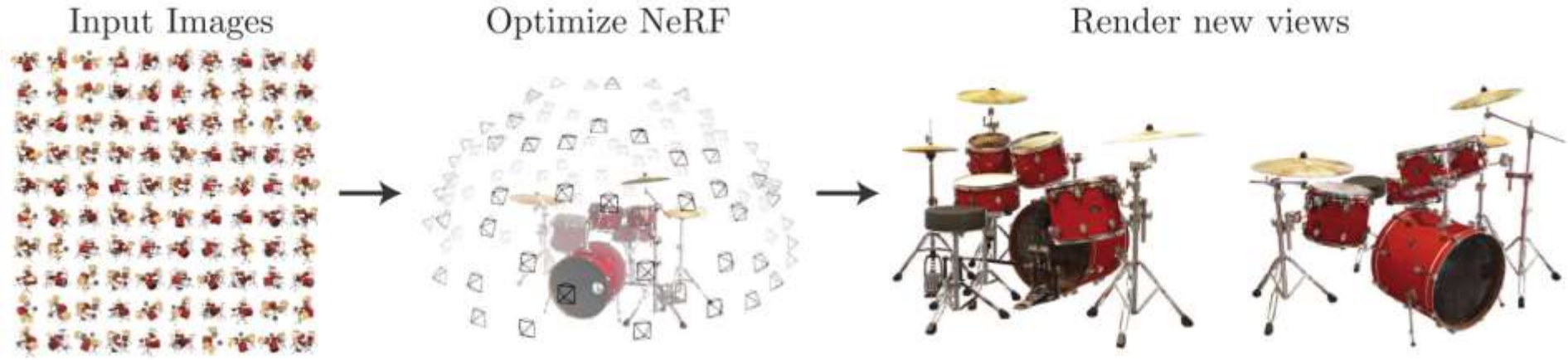
$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt, \text{ where } T(t) = \exp\left(-\int_{t_n}^{t} \sigma(\mathbf{r}(s))ds\right)$$



通俗来说就是：取一条光线，光线上采很多点。
采点的颜色会根据它前面的不透明度（体密度），
而对最终**该光线**的颜色颜色产生影响。
函数 T(t) 表示沿着光线从 tn 到 t 所**累积的透明度（accumulated transmittance）**，即**光线从 tn 出发到 t，穿过该路径的概率**。

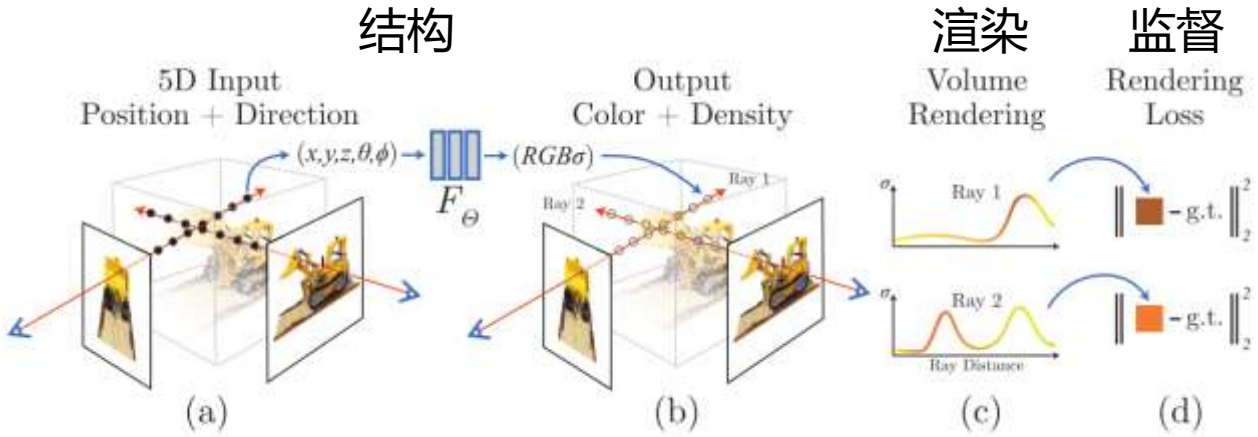
监督

最后呢，通过体渲染，我们得到了在**某个相机视角**下，该3D物体的**渲染图**。与我们真实的数据（**groud truth image**）计算误差，反向传播优化。
误差是MSE loss



【1】 Representing Scenes as Neural Radiance Fields for View Synthesis

NeRF总结



用一组MLP（辐射场）来表征了3D空间的物体结构。具体来说，空间中每个点都有两个属性：**体密度sigma，与颜色color。**
使用经典的**体渲染**的原理，渲染出任意**射线穿过场景的颜色。**
3D物体的**渲染图**与真实的数据（**groud truth image**）计算误差，进行优化。

一些提升效果的trick: 位置编码与层次化采样

【1】 Representing Scenes as Neural Radiance Fields for View Synthesis

3DGS的3D表征结构

三维高斯泼溅（3D Gaussian Splatting, 3DGS）

- 我们知道，足够密集的像素矩阵能够表示一幅高清图片
- 同理，用大量足够密集的三维“小基元”也可以清晰刻画一个三维物体
- 用**三维高斯椭球**作为这个小基元

3DGS的3D表征结构

每个高斯椭球都要存储以下信息：

- 均值 $\boldsymbol{\mu}$

  也就是椭球中心的坐标 $\boldsymbol{\mu} = (x, y, z)$

- 协方差矩阵 Σ

  Σ 的正定性让它难以直接被用于训练，所以一般拆成 $\Sigma = RSS^T R^T$，其中R为旋转矩阵，S为缩放矩阵，分别训练

- 密度 δ

- 颜色

  为了使高斯球在不同方向上能够展现出不同的颜色，3DGS用球谐函数拟合颜色，这里存储球谐函数的参数 $\boldsymbol{f}$

3DGS的渲染

与NeRF一样，都是采取α-blending的策略

$$C = \sum_{i=1}^{N} T_i \alpha_i \mathbf{c}_i,$$

with

$$\alpha_i = (1 - \exp(-\sigma_i \delta_i)) \text{ and } T_i = \prod_{j=1}^{i-1} (1 - \alpha_j).$$

各点渲染密度 α 的计算公式：

$$\alpha = \delta G(x')$$

$$G(x') = e^{-\frac{1}{2}(x')^T \Sigma'^T (x')}$$

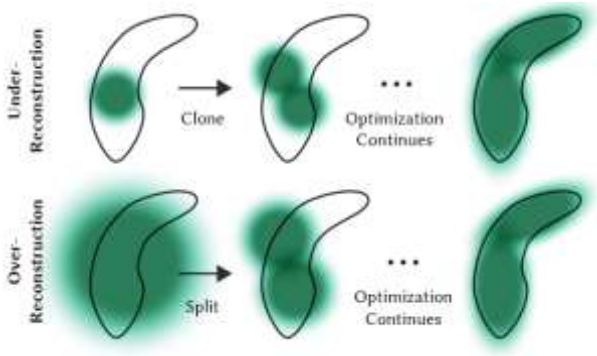
$$\Sigma' = JW \Sigma W^T J^T$$

【1】 Representing Scenes as Neural Radiance Fields for View Synthesis

致密化与剪枝

问：什么地方该放高斯球？放多少高斯球？

答：不知道，但我们可以引入三个操作：致密化和剪枝



致密化 (Densification)

对于训练过程中梯度大的地方（即渲染结果与GT相差大的地方），增加高斯点

剪枝 (Pruning)

删除掉密度δ太小的高斯点，同时引入定期的“密度重置”操作作为辅助

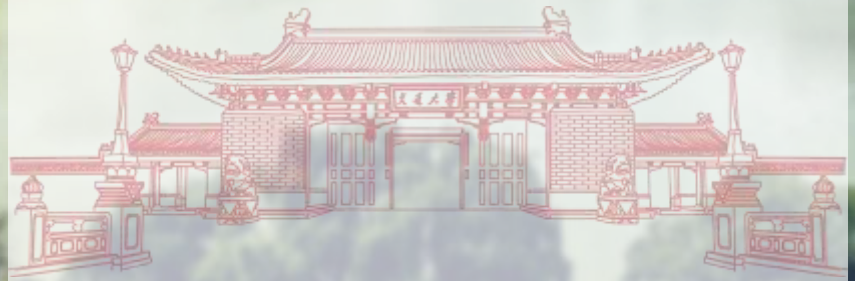
3DGS的效果展示

【1】 Representing Scenes as Neural Radiance Fields for View Synthesis

03

单视图重建

Single-view



单视图重建

给定图像的一个视角图片，还原出整个3D物体。就叫单视图重建

单视图重建是一个病态的问题，因为我只有一张物体的图片，缺失了过多信息。这个物体的背景长什么样子呢？有多种可能性。

解不是唯一的。

我们希望模型像人一样，能够有先验知识，知道这个物体背面（图片上看不到的面），长什么样子，并自动补齐。

One-2-3-45

One-2-3-45: Any Single Image to 3D Mesh in 45 Seconds without Per-Shape Optimization

Minghua Liu[1*] Chao Xu[2*] Haian Jin[3,4*] Linghao Chen[1,4*]
Mukund Varma T[5] Zexiang Xu[6]
Hao Su[1]

[1] UC San Diego [2] UCLA [3] Cornell University [4] Zhejiang University [5] IIT Madras [6] Adobe

【1】 One-2-3-45: Any Single Image to 3D Mesh in 45 Seconds without Per-Shape Optimization

One-2-3-45

One-2-3-45成功的前提：

像人一样，能够有先验知识的模型：ZERO-123

它可以给定一张图片，想象出这个物体在不同视角的图片。

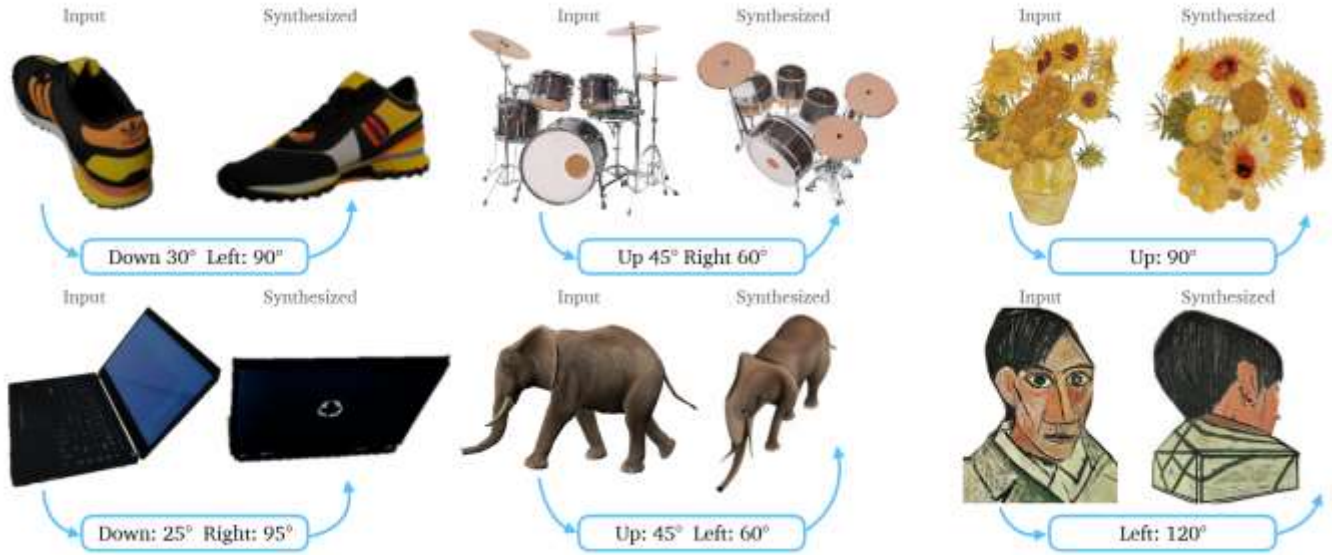
（其实是基于stable diffusion fine-tune的）

【1】 One-2-3-45: Any Single Image to 3D Mesh in 45 Seconds without Per-Shape Optimization

Zero-1-to-3

Zero-1-to-3: Zero-shot One Image to 3D Object (ICCV23)

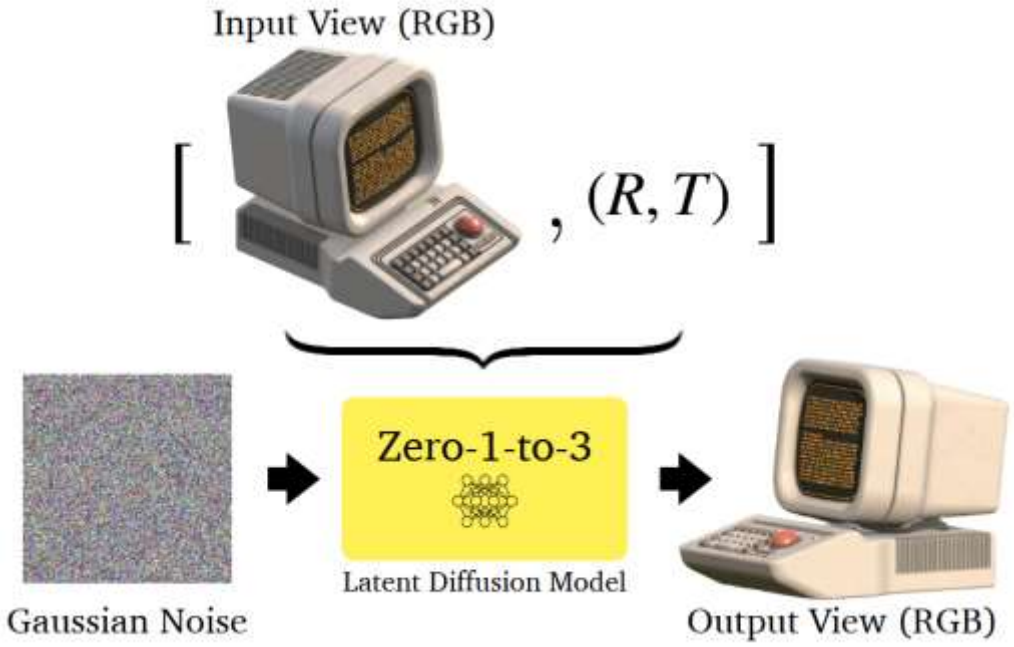
给一张图描述一个物体，这个物体在转特定角度之后应该长什么样子?



Zero-1-to-3

Zero-1-to-3: Zero-shot One Image to 3D Object (ICCV23)

- **原理：**训练一个条件生成模型，以原图和希望旋转+平移的参数作为条件，生成目标结果
- **实现方法：**微调 stable diffusion，训练数据来源于objaverse (800K)



【1】 One-2-3-45: Any Single Image to 3D Mesh in 45 Seconds without Per-Shape Optimization

One-2-3-45

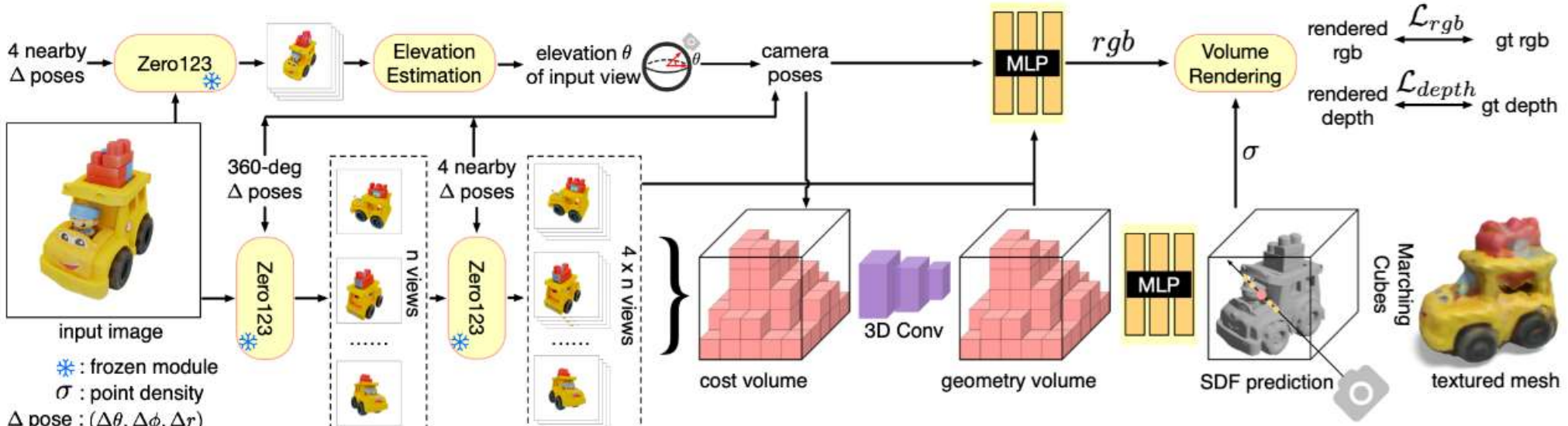
能不能直接用 Zero-1-to-3 生成的多视角图片训练一个NeRF?

不行！ 因为 Zero-1-to-3 生成的三维图片并不具有三维一致性，以此作为GT优化出的NeRF质量很低

【1】 One-2-3-45: Any Single Image to 3D Mesh in 45 Seconds without Per-Shape Optimization

One-2-3-45

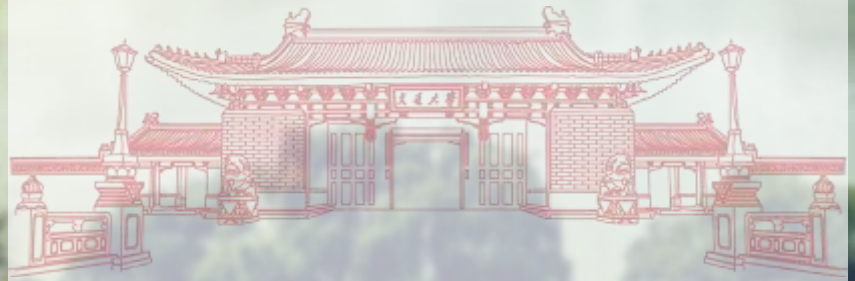
关键：用Zero-123从单视角补充成多视角，获得多视角训练集，
然后训练一个生成模型，生成目标物体的SDF表征



04

项目作业

Project



项目内容与目标

项目内容:

- 熟悉三维数据的格式与使用;
- 收集足够多的三维数据;
- 了解三维数据在深度学习模型中的表征;
- 微调以三维数据为表征的AI生成模型;

项目目标:

- 了解数据收集，数据清洗，模型训练，模型测试等全流程开发;
- 对三维领域的内容生成及编辑方法获得宏观的认知和理解;

项目内容与目标

三维数据集

- 熟悉三维数据的格式与使用
- 按照提供的爬虫例子收集并清洗项目相关数据；
- 数据类别清晰、数量达标；
- 综合考虑数据难度和数据完成度作为评分依据

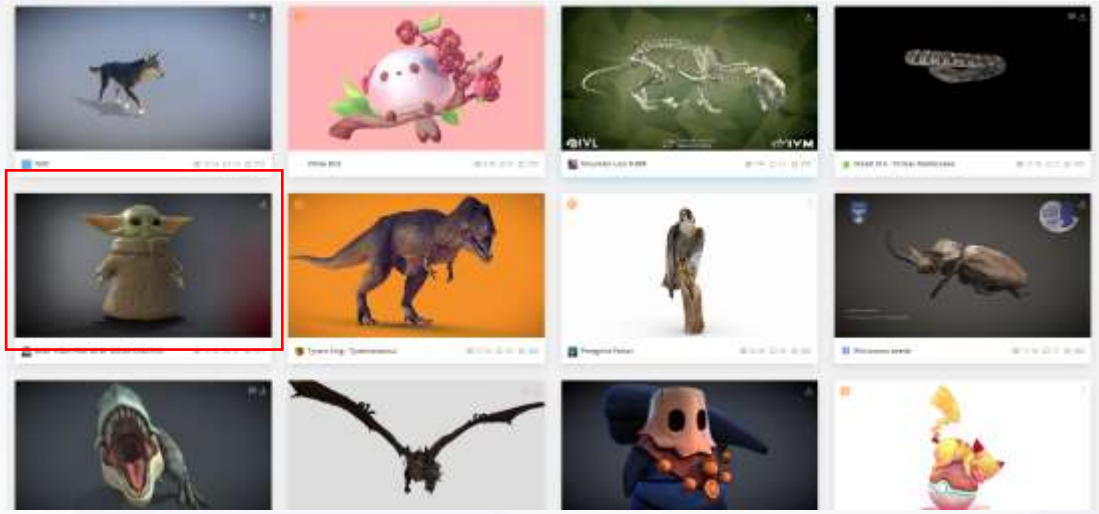
三维生成与重建模型

- 熟悉并理解三维物体生成与重建的原理
- 基于收集的数据，**微调推荐的单视图重建模型！研究单视图重建现有模型在小众子类上的泛化能力。**
- 分析结果质量（多种评价指标，如与真实3D几何的误差，渲染结果的图像误差等）并给出可能的因素对结果的影响
- 采用代码+项目报告作为评分依据
- 鼓励同学创新设计

数据集具体要求

任务要求

1. 数据爬取：助教小组会提供爬虫示例脚本、示例网站与示例下载数据。示例脚本展示了一些常见的爬虫场景，请每个小组仔细学习后，了解爬虫基本思路之后再针对数据源网站自行撰写对应爬虫脚本，进行数据爬取。
2. 数据清洗：爬取之后需要把不符合类别的数据清除，并添加中文描述，如红框的中文描述是尤达
3. 渲染收集到的数据，并按照预训练模型需要的输入文件结构进行整理



数据要求

1. 两个小组分别收集“动/植物”、“文玩”两类模型，每个类别的数据量要求：5000-10000
2. 数据样本与类别名直接关联
3. 将文件以下面形式压缩提交

   组号_组长学号.zip

   |---收集到的数据（mesh+texture+文字描述+下载链接）

   |---清洗后的数据

   |---渲染结果（按照预训练模型输入结构存放）

算法与工具推荐

- 三维物体生成与重建
  - 算法推荐：One-2-3-45: Any Single Image to 3D Mesh in 45 Seconds without Per-Shape Optimization等
  - 数据源网站：https://sketchfab.com/；https://3d.si.edu/；https://poly.cam/explore；github.com；等三维数据开源网站（鼓励自由拓展）
  - 不同的网站对不同类别有偏好，部分类别在某个网站数量很少，但可能在另一个网站的数量很多

- 数据收集工具
  - 提供爬虫的例子；

> 1. 爬虫思路：分析网址源代码，定位文件信息
> 2. 对于类别利用“子类别”，“实例”，“同义词”等信息，扩充向数据源网站发送的关键词库，这样可以尽可能将所有相关数据收集到，扩增下载数据储备

成绩评定

每个阶段提交结果，独立评分，互不影响

数据收集与清洗：十一月中旬提交数据（评分占比40%）

模型训练与测试：十二月底提交代码与模型（评分占比30%）

项目报告：期末提交（评分占比30%）


上海交通大學
SHANGHAI JIAO TONG UNIVERSITY
感谢聆听
饮 水 思 源 爱 国 荣 校