GAMES 003 科研素养课

第六周：论文投稿管理与论文故事梳理

Sida Peng

Jun Gao

Songyou Peng

Qianqian Wang

GAMES 003 科研素养课

第六周：论文投稿管理与论文故事梳理

Sida Peng

Jun Gao

Songyou Peng

Qianqian Wang

前五周

初始化科研课题

建立领域视野 → 选择科研课题 → 设计技术方案

迭代技术方案

基于技术方案设计方法 ⇄ 基于实验结果提升方案

写作规划 → 论文评审

本次课

学术 … 习惯

前五周

初始化科研课题

建立领域视野 → 选择科研课题 → 设计技术方案

迭代技术方案

基于技术方案设计方法 ⇄ 基于实验结果提升方案

撰写学术论文

写作规划 → 故事梳理 → 论文画图 → 论文写作 → 论文评审

本次课程内容

掌握科研软技能

学术报告技巧

日常科研习惯

后六周

课程内容

- 论文投稿管理
  - 为什么需要论文投稿管理?
  - 如何进行论文投稿管理?
- 论文故事梳理
  - 为什么需要梳理论文故事?
  - 如何梳理好文章故事?

课程内容

- 论文投稿管理
  - 为什么需要论文投稿管理?
  - 如何进行论文投稿管理?

- 论文故事梳理
  - 为什么需要梳理论文故事?
  - 如何梳理好文章故事?

为什么需要论文投稿管理?

你有没有担心过...

- 离截止日期只剩一周了，我主要的实验有问题还要重新跑，文章也只有个题目？？
- 离截止日期只剩一天了，我的引言和方法怎么还是空的？？
- 离截止日期只剩一个小时了，我文章怎么还超了2页？？
- 离截止日期只剩一分钟了，怎么提交的系统宕机了？？
- 离截止日期已经过去了一小时了，哎？？完犊子了，我忘记把合作者的意见删了

一首《凉凉》拉给自己

为什么需要论文投稿管理

- **现实考虑**：能帮助你赶上截止日期！
- **身心健康**：能大大降低你的投稿的心理和生理压力，送你一个愉快的投稿经历
- **提升质量**：可以大大提升论文质量，提升中稿概率

> “My first drafts are so-so, but I think I re-write pretty well. **Good writing is re-writing**. This means you need to **start writing the paper early**!“
>
> -- Prof. Bill Freeman in *How to write a good CVPR submission*

如何高效进行论文投稿管理?

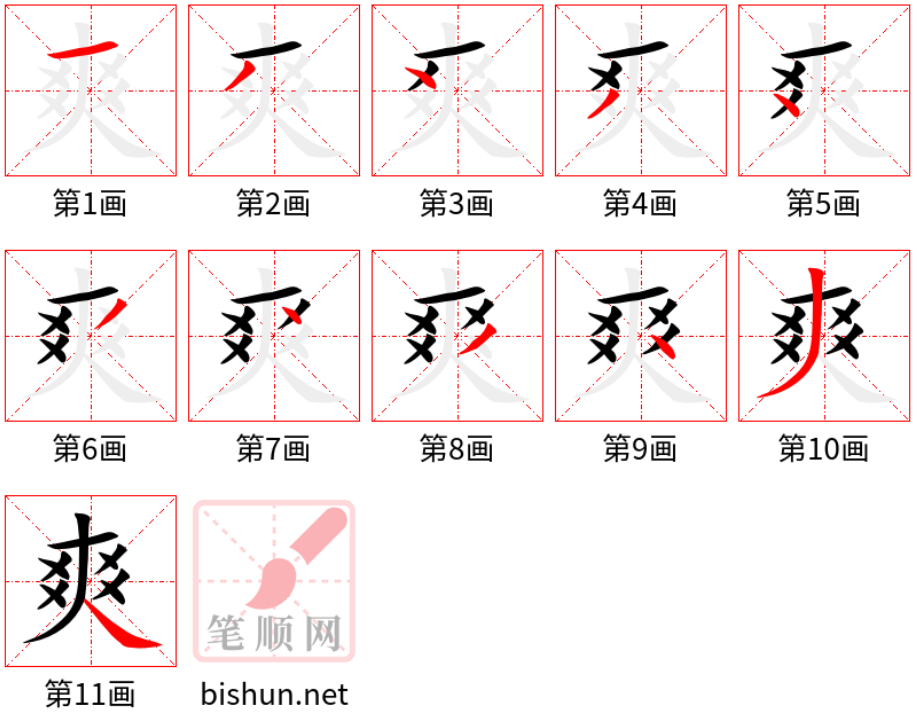
以分层方式迭代论文

核心方法：从粗到细，先列每小节的标题，再每个小节里面列要点，不断迭代完整要点，最后整合成这一小节的内容

- 为每个部分预留多次迭代

好处：

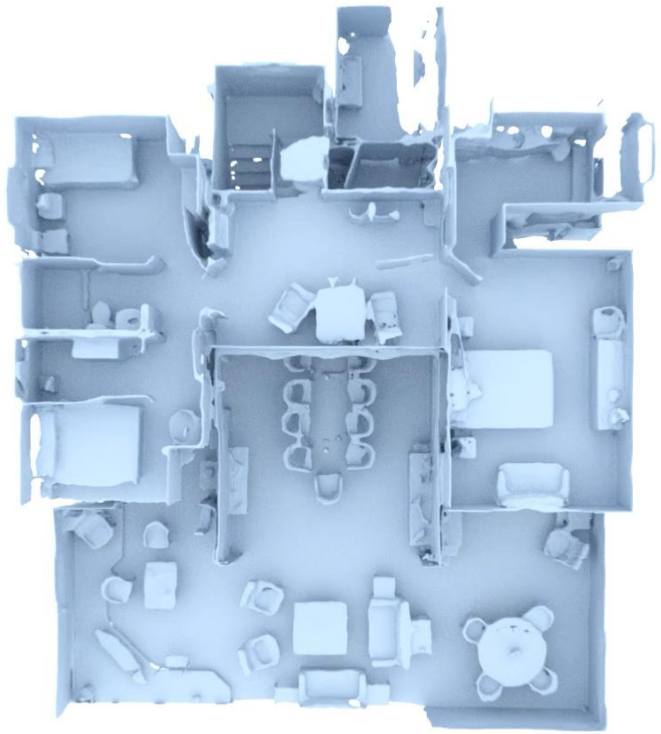
- 列要点本身很容易开始，防止拖延
- 更容易去确认上下文的逻辑
- 更轻松的和合作者一起迭代
- 由要点再转化成成段的文字非常容易

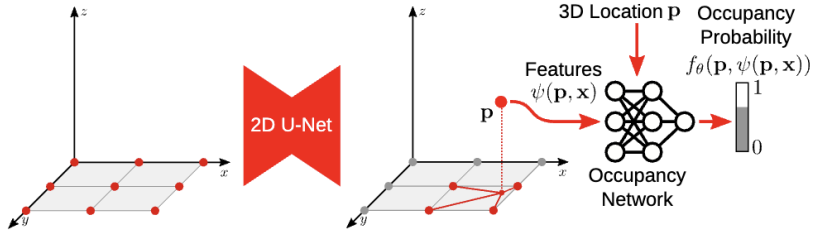


怎么操作?

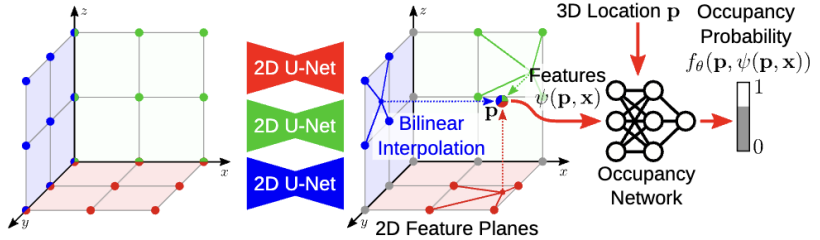
案例学习：ConvONet

第一篇神经隐式场应用到大场景三维重建的工作，还提出了tri-plane表征

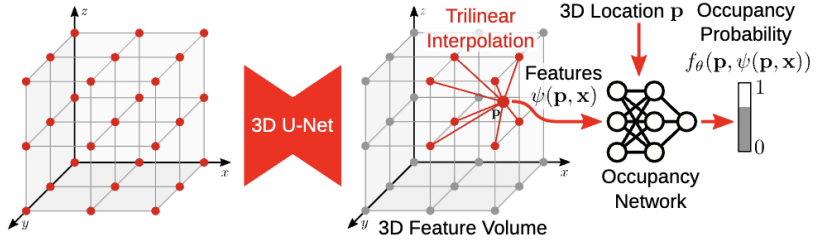




(c) Convolutional Single-Plane Decoder



(d) Convolutional Multi-Plane Decoder



(e) Convolutional Volume Decoder

Peng et al.: Convolutional Occupancy Networks. ECCV 2020

以分层方式迭代论文：ConvONet

第一步：列每小节的标题

Introduction
引言一般没有小节

以分层方式迭代论文：ConvONet

第一步：列每小节的标题

Related Work

- Voxels
- Point Clouds
- Meshes
- Implicit Representations

2 Related Work

Learning-based 3D reconstruction methods can be broadly categorized by the output representation they use.

Voxels: Voxel representations are amongst the earliest representations for learning-based 3D reconstruction [5,46,47]. Due to the cubic memory requirements of voxel-based representations, several works proposed to operate on multiple scales or use octrees for efficient space partitioning [8, 14, 25, 37, 38, 42]. However, even when using adaptive data structures, voxel-based techniques are still limited in terms of memory and computation.

Point Clouds An alternative output representation for 3D reconstruction is 3D point clouds which have been used in [9,21,34,49]. However, point cloud-based representations are typically limited in terms of the number of points they can handle. Furthermore, they cannot represent topological relations.

Meshes: A popular alternative is to directly regress the vertices and faces of a mesh [12, 13, 17, 20, 22, 44, 45] using a neural network. While some of these works require deforming a template mesh of fixed topology, others result in non-watertight reconstructions with self-intersecting mesh faces.

Implicit Representations: More recent implicit occupancy [3,26] and distance field [27,31] models use a neural network to infer an occupancy probability or distance value given any 3D point as input. In contrast to the aforementioned explicit representations which require discretization (e.g., in terms of the number of voxels, points or vertices), implicit models represent shapes continuously and naturally handle complicated shape topologies. Implicit models have been adopted for learning implicit representations from images [23, 24, 29, 41], for encoding texture information [30], for 4D reconstruction [28] as well as for primitive-based reconstruction [10, 11, 15, 32]. Unfortunately, all these methods are limited to

以分层方式迭代论文：ConvONet

第一步：列每小节的标题

Method

- Encoder
  - Plane Encoder
  - Volume Encoder
- Decoder
- Occupancy Prediction
- Training and Inference

3.1 Encoder

While our method is independent of the input representation, we focus on 3D inputs to demonstrate the ability of our model in recovering fine details and scaling to large scenes. More specifically, we assume a noisy sparse point cloud (e.g., from structure-from-motion or laser scans) or a coarse occupancy grid as input $\mathbf{x}$.

We first process the input $\mathbf{x}$ with a task-specific neural network to obtain a feature encoding for every point or voxel. We use a one-layer 3D CNN for voxelized inputs, and a shallow PointNet [35] with local pooling for 3D point clouds. Given these features, we construct planar and volumetric feature representations in order to encapsulate local neighborhood information as follows.

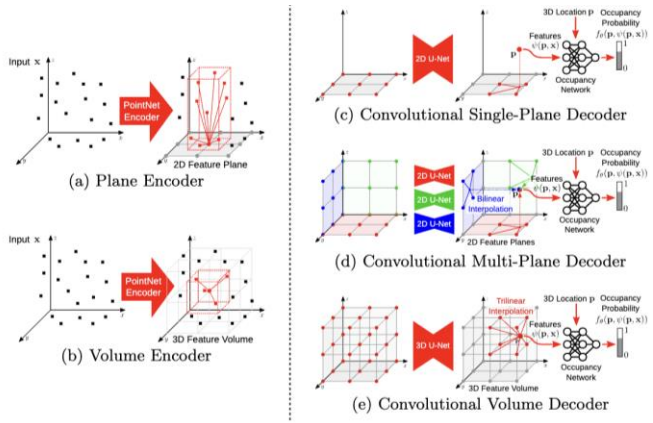


Fig. 2: **Model Overview.** The **encoder (left)** first converts the 3D input $\mathbf{x}$ (e.g., noisy point clouds or coarse voxel grids) into features using task-specific neural networks. Next, the features are projected onto one or multiple planes (Fig. 2a) or into a volume (Fig. 2b) using average pooling. The **convolutional decoder (right)** processes the resulting feature planes/volume using 2D/3D U-Nets to aggregate local and global information. For a query point $\mathbf{p} \in \mathbb{R}^3$, the point-wise feature vector $\psi(\mathbf{x}, \mathbf{p})$ is obtained via bilinear (Fig. 2c and Fig. 2d) or trilinear (Fig. 2e) interpolation. Given feature vector $\psi(\mathbf{x}, \mathbf{p})$ at location $\mathbf{p}$, the occupancy probability is predicted using a fully-connected network $f_\theta(\mathbf{p}, \psi(\mathbf{p}, \mathbf{x}))$.

Plane Encoder: As illustrated in Fig. 2a, for each input point, we perform an orthographic projection onto a canonical plane (i.e., a plane aligned with the axes of the coordinate frame) which we discretize at a resolution of $H \times W$ pixel cells. For voxel inputs, we treat the voxel center as a point and project it to the plane. We aggregate features projecting onto the same pixel using average pooling, resulting in planar features with dimensionality $H \times W \times d$, where d is the feature dimension.

Volume Encoder: While planar feature representations allow for encoding at large spatial resolution (128^2 pixels and beyond), they are restricted to two dimensions. Therefore, we also consider volumetric encodings (see Fig. 2b) which

3.2 Decoder

We endow our model with translation equivariance by processing the feature planes and the feature volume from the encoder using 2D and 3D convolutional hourglass (U-Net) networks [6, 39] which are composed of a series of down- and upsampling convolutions with skip connections to integrate both local and global information. We choose the depth of the U-Net such that the receptive field becomes equal to the size of the respective feature plane or volume.

Our single-plane decoder (Fig. 2c) processes the ground plane features with a 2D U-Net. The multi-plane decoder (Fig. 2d) processes each feature plane separately using 2D U-Nets with shared weights. Our volume decoder (Fig. 2e) uses a 3D U-Net. Since convolution operations are translational equivariant, our output features are also translation equivariant, enabling structured reasoning. Moreover, convolutional operations are able to "inpaint" features while preserving global information, enabling reconstruction from sparse inputs.

3.3 Occupancy Prediction

Given the aggregated feature maps, our goal is to estimate the occupancy probability of any point $\mathbf{p}$ in 3D space. For the single-plane decoder, we project each point $\mathbf{p}$ orthographically onto the ground plane and query the feature value through bilinear interpolation (Fig. 2c). For the multi-plane decoder (Fig. 2d), we aggregate information from the 3 canonical planes by summing the features of all 3 planes. For the volume decoder, we use trilinear interpolation (Fig. 2e).

Denoting the feature vector for input $\mathbf{x}$ at point $\mathbf{p}$ as $\psi(\mathbf{p}, \mathbf{x})$, we predict the occupancy of $\mathbf{p}$ using a small fully-connected occupancy network:

$$f_\theta(\mathbf{p}, \psi(\mathbf{p}, \mathbf{x})) \rightarrow [0, 1] \tag{1}$$

3.4 Training and Inference

At training time, we uniformly sample query points $\mathbf{p} \in \mathbb{R}^3$ within the volume of interest and predict their occupancy values. We apply the binary cross-entropy loss between the predicted $\hat{o}_\mathbf{p}$ and the true occupancy values $o_\mathbf{p}$:

$$\mathcal{L}(\hat{o}_\mathbf{p}, o_\mathbf{p}) = -[o_\mathbf{p} \cdot \log(\hat{o}_\mathbf{p}) + (1 - o_\mathbf{p}) \cdot \log(1 - \hat{o}_\mathbf{p})] \tag{2}$$

以分层方式迭代论文

第一步：列每小节的标题

Experiments

- Datasets & Baselines & Metrics
- Object-Level Reconstruction
  - Reconstruct from Point Clouds
  - Voxel Super-Resolution
- Scene-Level Reconstruction
- Ablation Study
  - GPU Memory
  - Feature Interpolation
- Reconstruction on Real-World Data
  - ScanNet
  - Matterport3D

4 Experiments

We conduct three types of experiments to evaluate our method. First, we perform **object-level reconstruction** on ShapeNet [2] chairs, considering noisy point clouds and low-resolution occupancy grids as inputs. Next, we compare our approach against several baselines on the task of **scene-level reconstruction** using a synthetic indoor dataset of various objects. Finally, we demonstrate **synthetic-to-real generalization** by evaluating our model on real indoor scenes [1, 7].

Datasets:

ShapeNet [2]: We use all 13 classes of the ShapeNet subset, voxelizations, and train/val/test split from Choy et al. [5]. Per-class results can be found in supplementary.

Synthetic Indoor Scene Dataset: We create a synthetic dataset of 5000 scenes with multiple objects from ShapeNet (chair, sofa, lamp, cabinet, table). A scene consists of a ground plane with randomly sampled width-length ratio, multiple objects with random rotation and scale, and randomly sampled walls.

ScanNet v2 [7]: This dataset contains 1513 real-world rooms captured with an RGB-D camera. We sample point clouds from the provided meshes for testing.

Matterport3D [1]: Matterport3D contains 90 buildings with multiple rooms on different floors captured using a Matterport Pro Camera. Similar to ScanNet, we sample point clouds for evaluating our model on Matterport3D.

Baselines:

ONet [26]: Occupancy Networks is a state-of-the-art implicit 3D reconstruction model. It uses a fully-connected network architecture and a global encoding of the input. We compare against this method in all of our experiments.

PointConv: We construct another simple baseline by extracting point-wise features using PointNet++ [36], interpolating them using Gaussian kernel regression and feeding them into the same fully-connected network used in our approach. While this baseline uses local information, it does not exploit convolutions.

SPSR [18]: Screened Poisson Surface Reconstruction (SPSR) is a traditional 3D reconstruction technique which operates on oriented point clouds as input. Note that in contrast to all other methods, SPSR requires additional surface normals which are often hard to obtain for real-world scenarios.

Metrics:

Following [26], we consider Volumetric IoU, Chamfer Distance, Normal Consistency for evaluation. We further report F-Score [43] with the default threshold value of 1% unless otherwise specified. Details can be found in the supplementary.

4.1 Object-Level Reconstruction

Reconstruction from Point Clouds: Table 1 and Fig. 3 show quantitative and qualitative results. Compared to the baselines, all variants of our method achieve equal or better results on all three metrics. As evidenced by the training progression plot on the right, our method reaches a high validation IoU after only few iterations. This verifies our hypothesis that leveraging convolutions and local features benefits 3D reconstruction in terms of both accuracy and efficiency. The results show that, in comparison to PointConv which directly aggregates features from point clouds, projecting point-features to planes or volumes followed by 2D/3D CNNs is more effective. In addition, decomposing 3D representations from volumes into three planes with higher resolution (64^2 vs. 32^3) improves performance while at the same time requiring less GPU memory. More results can be found in supplementary.

Voxel Super-Resolution: Besides noisy point clouds, we also evaluate on the task of voxel super-resolution. Here, the goal is to recover high-resolution details

4.2 Scene-Level Reconstruction

To analyze whether our approach can scale to larger scenes, we now reconstruct 3D geometry from point clouds on our synthetic indoor scene dataset. Due to the increasing complexity of the scene, we uniformly sample 10000 points as input point cloud and apply Gaussian noise with standard deviation of 0.05. During training, we sample 2048 query points, similar to object-level reconstruction For our plane-based methods, we use a resolution to 128^2. For our volumetric approach, we investigate both 32^3 and 64^3 resolutions. Hypothesizing that the

4.3 Ablation Study

In this section, we investigate on our synthetic indoor scene dataset different feature aggregation strategies at similar GPU memory consumption as well as

Performance at Similar GPU Memory: Table 4a shows a comparison of different feature aggregation strategies at similar GPU memory utilization. Our multi-plane approach slightly outperforms the single plane and the volumetric approach in this setting. Moreover, the increase in plane resolution for the single plane variant does not result in a clear performance boost, demonstrating that higher resolution does not necessarily guarantee better performance.

Feature Interpolation Strategy: To analyze the effect of the feature interpolation strategy in the convolutional decoder of our method, we compare nearest neighbor and bilinear interpolation for our multi-plane variant. The results in Table 4b clearly demonstrate the benefit of bilinear interpolation.

4.4 Reconstruction from Point Clouds on Real-World Datasets

Next, we investigate the generalization capabilities of our method. Towards this goal, we evaluate our models trained on the synthetic indoor scene dataset on the real world datasets ScanNet v2 [7] and Matterport3D [1]. Similar to our previous experiments, we use 10000 points sampled from the meshes as input.

ScanNet v2: Our results in Table 5 show that among all our variants, the volumetric-based models perform best, indicating that the plane-based approaches are more affected by the domain shift. We find that 3D CNNs are more robust to noise as they aggregate features from all neighbors which results in smooth outputs. Moreover, all variants outperform the learning-based baselines by a significant margin.

The qualitative comparison in Fig. 6 shows that our model is able to smoothly reconstruct scenes with geometric details at various scales. While Screened PSR [18] also produces reasonable reconstructions, it tends to close the resulting meshes and hence requires a carefully chosen trimming parameter. In contrast, our method does not require additional hyperparameters.

Matterport3D Dataset: Finally, we investigate the scalability of our method to larger scenes which comprise multiple rooms and multiple floors. For this

以分层方式迭代论文：ConvONet

第二步：在每个小节里面列要点

以Related Work里的Implicit Representations小节为例

- Voxels
- Point Clouds
- Meshes
- **Implicit Representations**

以分层方式迭代论文：ConvONet

第二步：在每个小节里面列要点

以Related Work里的Implicit Representations小节为例

- 近期工作，开始用网络预测占有场 (Occupancy) 或者符号距离场 (SDF)
- 另外一些工作开始用局部特征，但是只在二维的图像域上面应用
- 我们的方法把三维和二维的卷积网络用上
- 我们方法可以大场景重建
- 有两个同期的工作，一个只有三维卷积，一个只考虑图形先验

以分层方式迭代论文：ConvONet

第三步：不断扩充要点，填充内容

以Related Work里的Implicit Representations小节为例

- 近期的隐式表征工作，开始考虑用网络预测占有场(Occupancy) 或者符号距离场 (SDF)
  - 显示表征需要离散化，这会让结果分辨率低以及用更多内存
  - 他们的连续性可以模型相对复杂的拓扑结构，但是因为使用全局特征，过于平滑了

- 另外一些工作开始用局部特征，但是只在二维的图像域上面应用
  - PIFu and DISN 用了像素对准的局部特征来做单个物体的重建

- 我们的方法把三维和二维的卷积网络用上
  - 和前面的方法不一样，我们提出了在三维空间汇总局部特征，同时探索了二、三维
  - 这样，我们这个世界中心的表征不受相机位姿和输入的影响

- 我们方法可以大场景重建，就像在teaser figure里面展示的那样

- 最后，我们发现有两个同期的工作，一个只考虑三维卷积，一个只考虑图形先验
  - 一篇工作和我们的区别是他只考虑三维卷积做物体重建，而我们提出了三种方式
  - 另一篇考虑图形先验，但是他们需要点云的法线作为输入

以分层方式迭代论文：ConvONet

第四步：不断夯实要点，最后汇总要点，转化成这小节的文字

以Related Work里的Implicit Representations小节为例

- 近期的隐式表征工作，开始考虑用网络预测占有场(Occupancy) 或者符号距离场 (SDF)
  - 显示表征需要离散化，这会让结果分辨率低以及用更多内存
  - 他们的连续性可以模型相对复杂的拓扑结构，但是因为使用全局特征，过于平滑了

- 另外一些工作开始用局部特征，但是只在二维的图像域上面应用
  - PIFu and DISN 用了像素对准的局部特征来做单个物体的重建

- 我们的方法把三维和二维的卷积网络用上
  - 和前面的方法不一样，我们提出了在三维空间汇总局部特征，同时探索了二、三维
  - 这样，我们这个世界中心的表征不受相机位姿和输入的影响

- 我们方法可以大场景重建，就像在teaser figure里面展示的那样

- 最后，我们发现有两个同期的工作，一个只考虑三维卷积，一个只考虑图形先验
  - 一篇工作和我们的区别是他只考虑三维卷积做物体重建，而我们提出了三种方式
  - 另一篇考虑图形先验，但是他们需要点云的法线作为输入

以分层方式迭代论文：ConvONet

第四步：不断夯实要点，最后汇总要点，转化成这小节的文字

以Related Work里的Implicit Representations小节为例

近年来，更为先进的隐式占用和距离场模型使用神经网络，根据任意输入的三维点来推断占用概率或距离值。与之前提到的需要离散化（例如，按体素、点或顶点的数量）的显式表示不同，隐式模型能够连续地表示形状，并自然地处理复杂的拓扑结构。隐式模型已被用于从图像中学习隐式表示、编码纹理信息、进行四维重建以及基于基本体元的重建。然而，所有这些方法都局限于相对简单的单个物体的三维几何形状，无法扩展到更复杂或大规模的场景。主要的限制因素在于其简单的全连接网络架构，无法整合局部特征或引入如平移等变性等归纳偏置。

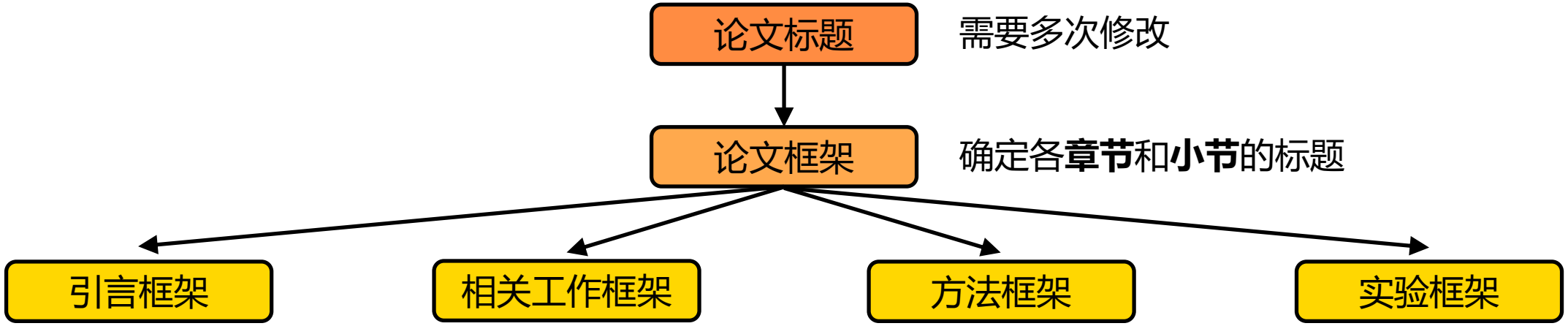
值得注意的例外是 PIFu 和 DISN，它们使用像素对齐的隐式表示来重建穿着衣服的人体或 ShapeNet 对象。虽然这些方法也利用了卷积操作，但所有的操作都在二维图像域中进行，这限制了这些模型只能接受基于图像的输入并重建单个物体。相反，在我们的工作中，我们提出了在物理三维空间中聚合特征，利用二维和三维卷积。因此，我们的以世界为中心的表示与相机视角和输入表示无关。此外，我们展示了如图 1c 所示的场景级隐式三维重建的可行性。

在同期的工作中，Chibane 等人提出了一个与我们的卷积体积解码器类似的模型。与我们相比，他们只考虑了单一变体的卷积特征嵌入（三维），对三维点云编码使用了有损离散化，并且仅在单个物体和人体上展示了结果，而非完整的场景。在另一项同期工作中，Jiang 等人利用形状先验进行场景级隐式三维重建。与我们不同的是，他们使用了三维点法线作为输入，并且在推理时需要进行优化。

如何高效进行论文投稿管理?

对文章整体迭代

论文迭代流程



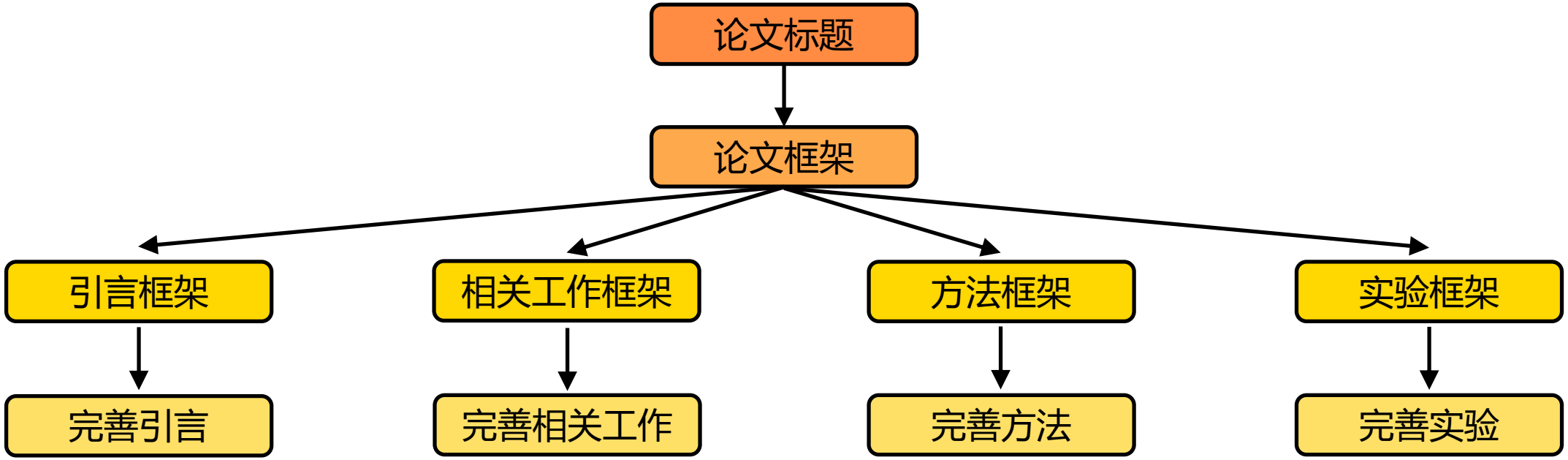
- 列出每段想表达的要点
- 不断填充和完善要点
- 合作者尽早开始讨论迭代
- 梳理故事思路在后面讲!

- 列出相关的主题
- 每个主题下列出要讨论的论文
- 要点简述打算如何评价

- 列每个小节的标题和要点
- 展示你将如何引导读者理解你的方法
- 为流程图预留位置

- 说明实验设置的要点。
- 列出小节标题和需要支持的论点。
- 为图表和表格预留位置
- 列出计划进行的分析内容(无需实际结果，迭代)

论文迭代流程



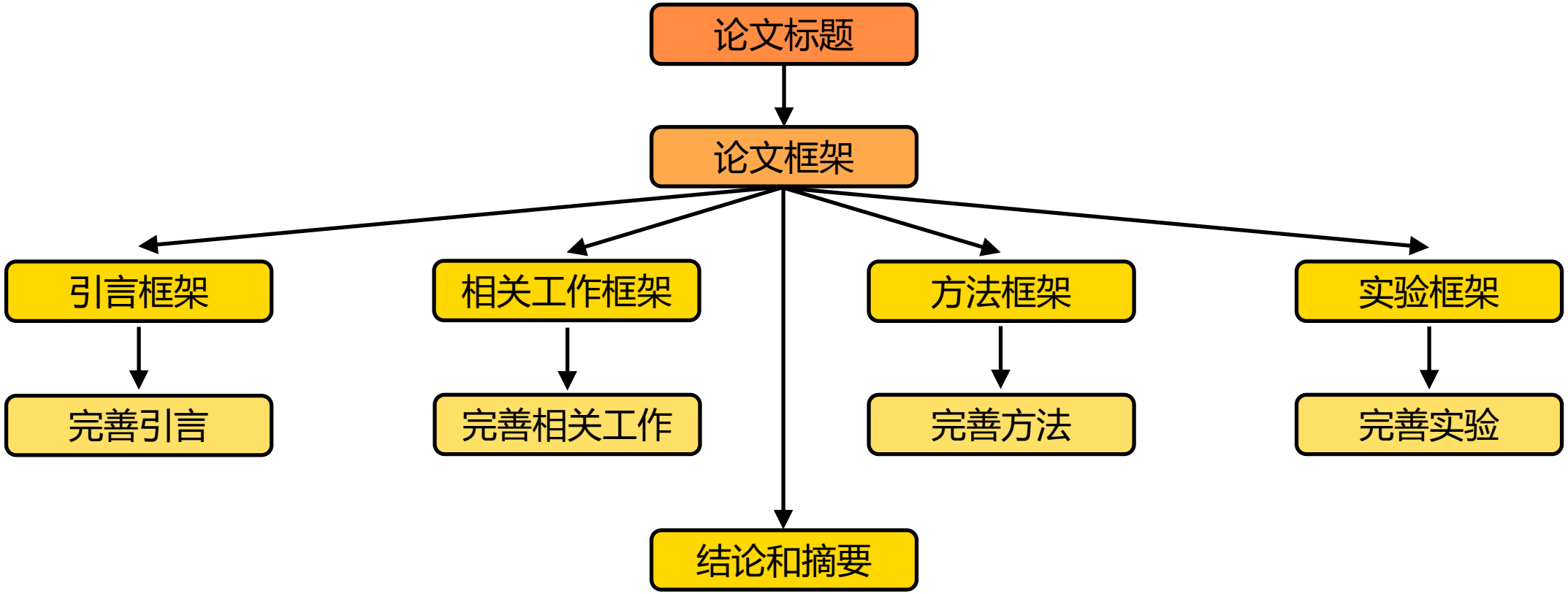
- 要点详尽后写完整句子
- 形成初稿，需要多次迭代
- 画teaser figure

- 将框架中的内容填充完整。

- 方法一般长且复杂
- 可以分块展开写
- 画流程图和展示图

- 填入实际的句子, 数据和结果
- 强调获取什么新的认知

论文迭代流程



论文投稿管理==项目管理

项目管理 [编辑]

57种语言

条目 讨论 汉 漢 大陆简体 阅读 编辑 查看历史 工具

此条目需要**精通或熟悉相关主题的编者**参与及协助编辑。
请邀请适合的人士改善本条目。更多的细节与详情请参见讨论页。

此条目**可参照英语维基百科相应条目来扩充。**
若您熟悉来源语言和主题，请协助参考外语维基百科扩充条目。请勿直接提交机械翻译，也不要翻译不可靠、低品质内容。依版权协议，译文需在编辑摘要注明来源，或于讨论页顶部标记 `{{Translated page}}` 标签。

项目管理是领导一个团队在规定时间内实现目标和达到成功标准的过程，其主要挑战在于在给定

学会“领导”你的合作者/导师

- 可以请资深的合作者帮忙写引言和相关工作等章节，画流程图和teaser figure
- 提前请合作者帮忙一起跑实验 (跑基线方法，跑消融实验，等等)
- 有策略的寻求帮助：比如问能不能帮忙给某一段或者列的要点提提建议

给自己设置小的截止日期

我读博期间的CVPR投稿时间线（理想情况），aka 一个让你不用熬夜的方法

1. **投稿一个月前**：3-4页左右初稿，需要包含引言和方法列好的要点，一定要有计划好的每一个实验。和合作者/导师讨论一遍这里面的点
2. **投稿前两周**：7-8页左右的比较完整的稿，流程图，teaser figure，实验的图表都已经有一些初步版本了，可能还欠缺一些实验
3. **投稿前一周**：有一个8-10页的稿，绝大部分实验已经完成，合作者最好已经看过以及修改了一遍文字和图，之后主要精力就专注打磨文章和图表
4. **投稿前2-3天**：非常完整的终稿，不再跑任何实验，专注文章润色，对于细节修改。导师过最后一遍，投稿，开香槟

一些额外的建议

- **借鉴**：精读这个领域之前的几篇你觉得最好的文章，分析他们写作逻辑，分析优劣，模仿
- **简单文字**：避免用大词，如果能用最简单的词就能说清楚的，就够了
- **视觉吸引力**：多花时间在提升文章里面图的质量 （下周课程）
- **井井有条**：把所有和这个项目相关的材料都组织好

免责声明

- 这是我的投稿管理方式，但是即便如此我自己每篇文章的情况都不一样，也不是都能遵守
- 希望你们可以自由探索，找到适合自己的方式
- 可以努力沿着这个指导方针做，文章相信会更好

课程内容

- 论文投稿管理
  - 为什么需要论文投稿管理?
  - 如何进行论文投稿管理?

- 论文故事梳理
  - 为什么需要梳理论文故事?
  - 如何梳理好文章故事?

课程内容

- 论文投稿管理
  - 为什么需要论文投稿管理?
  - 如何进行论文投稿管理?

- 论文故事梳理
  - 为什么需要梳理论文故事?
  - 如何梳理好文章故事?

为什么需要梳理论文故事?

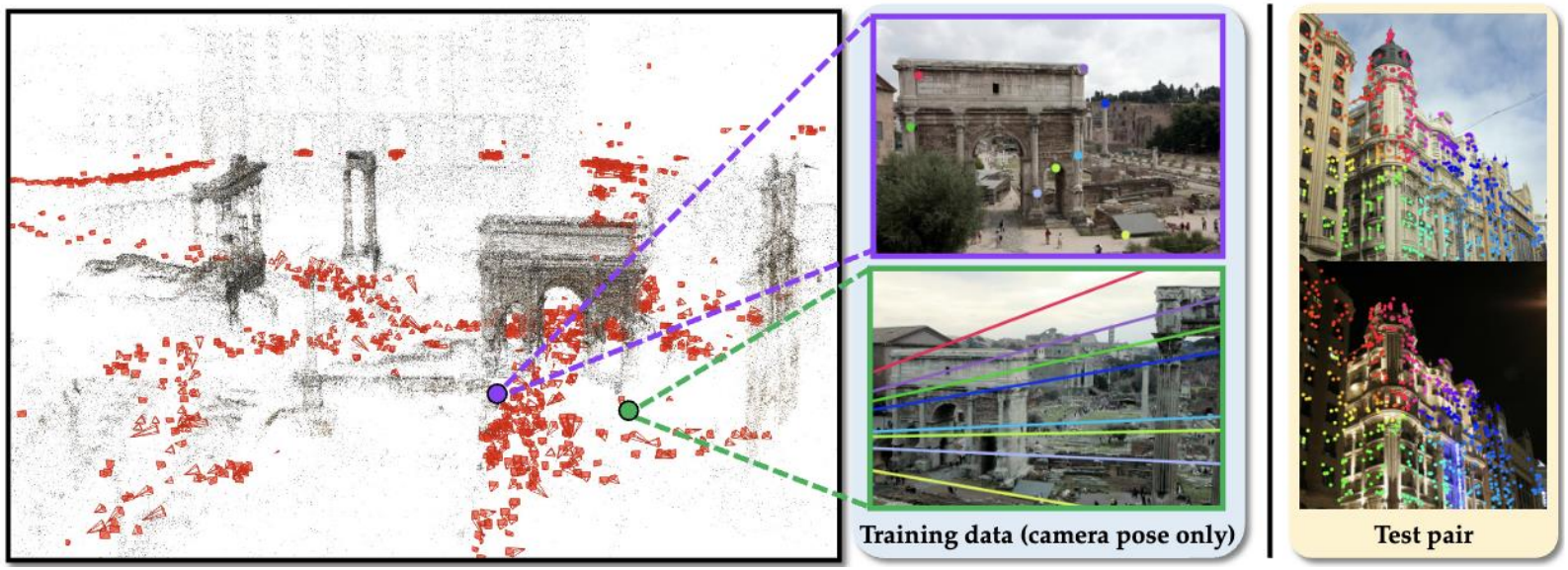
为什么需要梳理论文故事?

再好的想法，如果没找准卖的点也可能会被拒

Learning Feature Descriptors using Camera Pose Supervision

Qianqian Wang[1,2], Xiaowei Zhou[3], Bharath Hariharan[1], and Noah Snavely[1,2]

[1]Cornell University [2]Cornell Tech [3]Zhejiang University



为什么需要梳理论文故事?

Learning Feature Descriptors using Camera Pose Supervision

Qianqian Wang[1,2], Xiaowei Zhou[3], Bharath Hariharan[1], and Noah Snavely[1,2]
[1]Cornell University [2]Cornell Tech [3]Zhejiang University

再好的想法，如果没找准卖的点也可能会被拒

- **CVPR 2020 投稿**
  - **卖点**：一个可微分的匹配层，由粗到细来优化，以及用相机位姿做监督
  - **结果**：3个borderline，最后被拒
    - 评审人喷这个可微分的匹配层在别的领域已经有人做了，由粗到细来优化也没创新!

- **ECCV 2020 投稿**
  - **卖点**：单纯就卖相机位姿做监督这个最核心的点，之前也没有人做过
  - **结果**：接收为 **Oral Presentation**

如何梳理好文章故事?

前置：把读者当做第一次来你家的客人

- 预见他们可能的需求
  - 房子会不会太冷了？
  - 他们是想喝热水吗？
  - 吃饭他们喜欢吃辣吗？
  - 吃完饭他们是想打麻将还是唱歌？
- 设想他们完全不了解你
  - 他们很容易忘记你说的事情
  - 他们不是很耐心
  - 他们可能和你有代沟，不理解你说的

Generated by ChatGPT 4o

本页受启发于: How to write a good CVPR submission, Bill Freeman

梳理文章故事

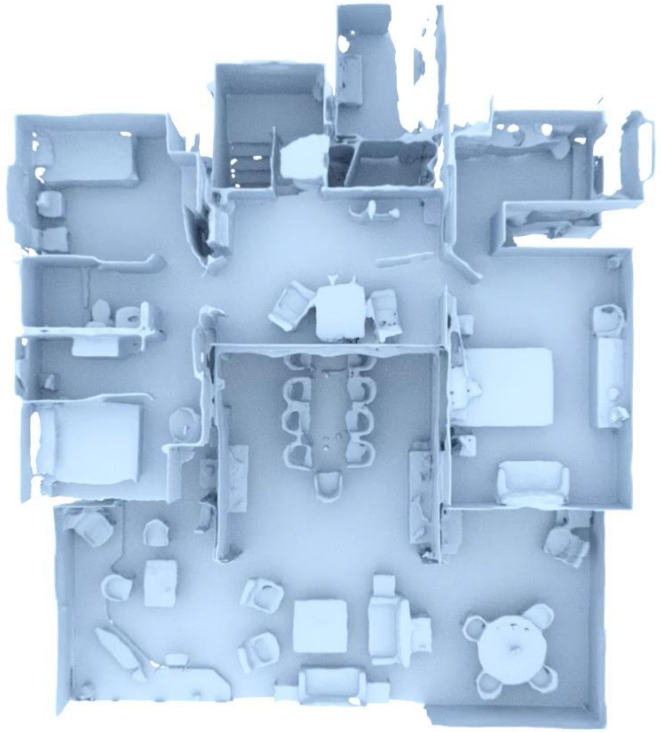
如何找准最好的故事角度？核心是回答以下的问题：

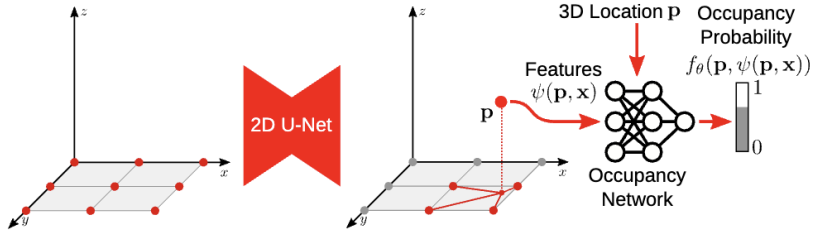
- 我们在解决的问题是什么？
- 为什么这个问题很重要？
- 之前的方法有哪些，他们有什么问题？
- 我们方法核心是什么，有什么是只有我们可以做到的？
- 我们（将）获得什么新的认知？

记住：在你的项目的每个阶段都要回答这些问题，而不是留到最后

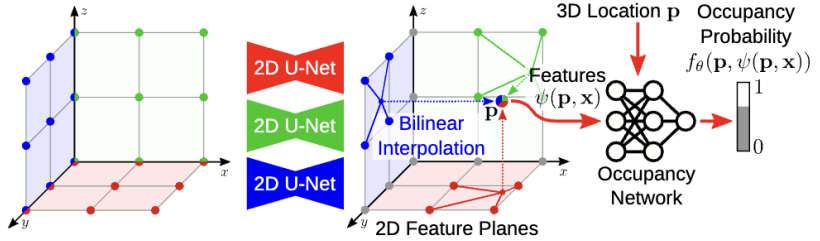
案例学习：ConvONet

第一篇神经隐式场应用到大场景三维重建的工作，还提出了tri-plane表征

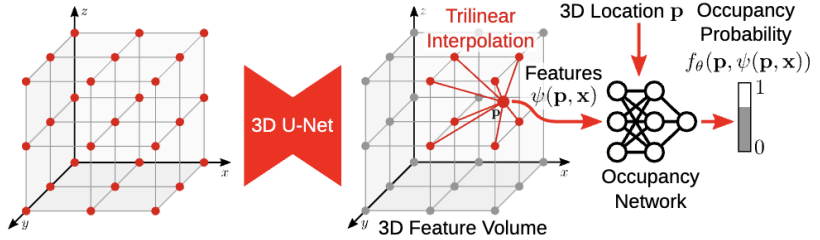




(c) Convolutional Single-Plane Decoder



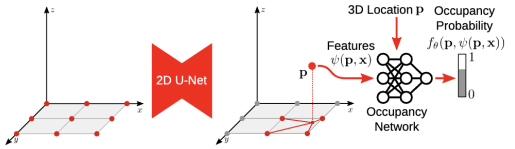
(d) Convolutional Multi-Plane Decoder



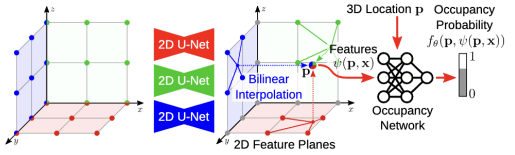
(e) Convolutional Volume Decoder

Peng et al.: Convolutional Occupancy Networks. ECCV 2020

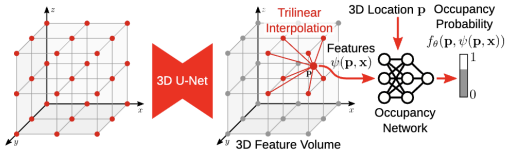
案例学习：ConvONet



(c) Convolutional Single-Plane Decoder



(d) Convolutional Multi-Plane Decoder



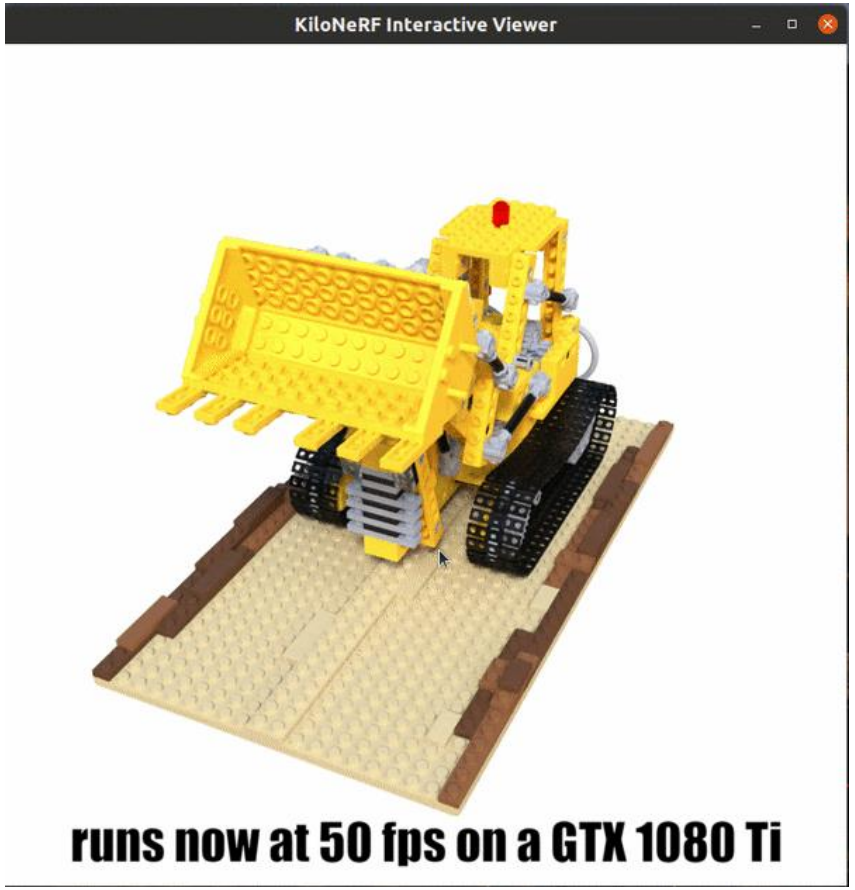
(e) Convolutional Volume Decoder

- **我们在解决的问题是什么?**
  - 如何获得精细的三维重建
- **为什么这个问题很重要?**
  - 经典三维视觉和图形学的问题，对虚拟现实，游戏，自动驾驶都有很大意义
- **之前的方法有哪些，他们有什么问题?**
  - Occupancy Networks: 全局潜在表征 (global latent code) 导致过于平滑的重建，只能物体
- **我们方法核心是什么，有什么是只有我们可以做到的?**
  - 提出三种局部表征 (local latent code) 的方式，然后对应使用卷积网络去加强表达能力
- **我们（将）获得什么新的认知?**
  - 不仅大大提升物体重建质量，卷积网络的平移相等性（translation equivariance）也能直接使得大场景重建成为可能。同时，我们提出的tri-plane 表征能同时降低内存以及提升表达

Peng et al.: Convolutional Occupancy Networks. ECCV 2020

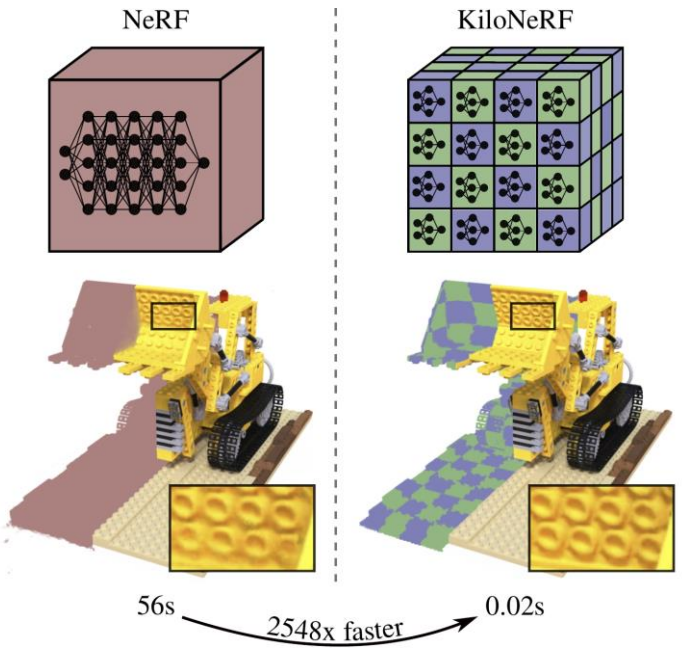
案例学习：KiloNeRF

加速神经辐射场 (NeRF) 2500倍到可以实时渲染



Reiser et al.: KiloNeRF: Speeding up Neural Radiance Fields with Thousands of Tiny MLPs. ICCV 2021

案例学习：KiloNeRF



- **我们在解决的问题是什么?**
  - 如何在保证高质量渲染的同时，加速神经辐射场 (NeRF) 到实时
- **为什么这个问题很重要?**
  - 实时渲染可以大大缩短游戏电影制作周期和成本
- **之前的方法有哪些，他们有什么问题?**
  - NeRF: 单独一个大的网络，每一次渲染过程都要前传整个网络，非常慢
- **我们方法核心是什么，有什么是只有我们可以做到的?**
  - 把空间分割成一千多个小的区域，每个里面都有一个非常小的网络，直接帮助大大加速
- **我们（将）获得什么新的认知?**
  - 这样的分治法 (Divide and Conquer) 能大大降低运算成本的同时直接提升运算速度

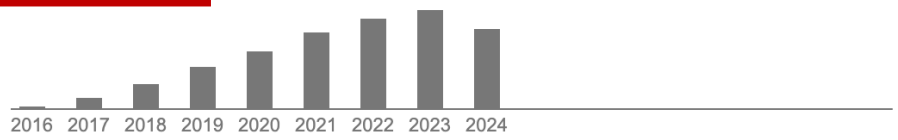
Reiser et al.: KiloNeRF: Speeding up Neural Radiance Fields with Thousands of Tiny MLPs. ICCV 2021

案例学习： ResNet

人类历史至今，计算机领域最高引文章

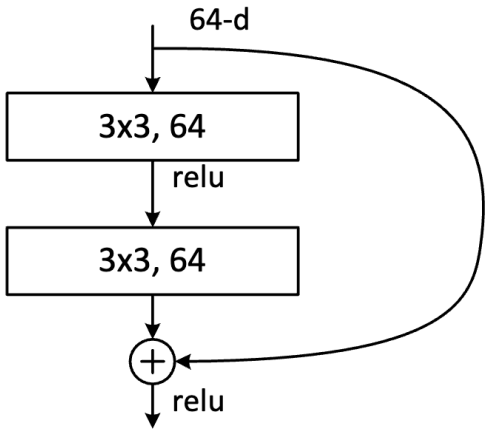
Deep Residual Learning for Image Recognition

Authors	Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun
Publication date	2016
Conference	Computer Vision and Pattern Recognition (CVPR), 2016
Description	Deeper neural networks are more difficult to train. We present a residual learning framework to ease the training of networks that are substantially deeper than those used previously. We explicitly reformulate the layers as learning residual functions with reference to the layer inputs, instead of learning unreferenced functions. We provide comprehensive empirical evidence showing that these residual networks are easier to optimize, and can gain accuracy from considerably increased depth. On the ImageNet dataset we evaluate residual nets with a depth of up to 152 layers---8x deeper than VGG nets but still having lower complexity. An ensemble of these residual nets achieves 3.57% error on the ImageNet test set. This result won the 1st place on the ILSVRC 2015 classification task. We also present analysis on CIFAR-10 with 100 and 1000 layers. The depth of representations is of central importance for many visual recognition tasks. Solely due to our extremely deep representations, we obtain a 28% relative improvement on the COCO object detection dataset. Deep residual nets are foundations of our submissions to ILSVRC & COCO 2015 competitions, where we also won the 1st places on the tasks of ImageNet detection, ImageNet localization, COCO detection, and COCO segmentation.
Total citations	Cited by 246584



He et al.: Deep Residual Learning for Image Recognition. CVPR 2016 (Best Paper)

案例学习：ResNet



- **我们在解决的问题是什么?**
  - 深度网络比浅层网络难训练非常多
- **为什么这个问题很重要?**
  - 越深的网络表现力越强，如果能解决训练问题，太多应用了
- **之前的方法有哪些，他们有什么问题?**
  - AlexNet, VGG Net, 一旦网络加深就训练不动
- **我们方法核心是什么，有什么是只有我们可以做到的?**
  - 残差很容易学习，第一次可以把一个152层的网络训练起来
- **我们（将）获得什么新的认知?**
  - 巨多的实验上全部都有效，巨多的消融实验去分析，牛!

He et al.: Deep Residual Learning for Image Recognition. CVPR 2016 (Best Paper)

Extra Tips

- 主人翁意识：别期待合作者帮你想出故事，你对你的项目是最懂的，积极主动自己想出你觉得最好的故事
- 确保核心想法可以清楚的传达到：通过你的题目，摘要, 引言，图等等，不断强化
- 需要迭代很多次，所以尽早开始梳理你的故事！

Q&A：回答同学们的一些问题

问题 1：引言和相关工作板块中对相关工作的讨论有什么区别?

回答:

- 引言：讨论的相关工作是服务于你的故事的，通常只会非常简略的讲几篇最相关的工作
- 相关工作：这是更广阔的和你的论文相关的工作，它可以不服务于你的故事线，强调的是详尽度

Q&A：回答同学们的一些问题

问题 2：是否建议idea与实验不太完善的文章尝试投稿?

回答：分情况。

- 如果你觉得文章本身的确有很有意思的发现，可以考虑投一个相关的workshop，一般workshop都会有4页的投稿，这比较适合。

- 但是如果文章本身暂时还没有太多有意思的东西，个人不建议投稿，这不仅会浪费审稿人的时间，同时如果还放arXiv的话，也会浪费读的人的时间

相关问题：方法自认为还有可以改进的地方，我是继续改进呢，还是把继续改进的点放到下一篇工作?

回答：取决于你的"可以改进"的点是不是足够支撑新的文章，或者能让当前文章变得更强。也要看你的时间够不够

Q&A：回答同学们的一些问题

问题 3：在文章投稿出版后，发现文章有错误（不影响模型和结论），应该如何处理，是否会造成“学术污点”？

回答：

- 如果有错误，但是不影响结果和结论的话，请立刻更改论文，并列举修改的所有东西，然后立刻联系对应的期刊和会议提出更改，然后arXiv也请更新，并在comment里面加修改了的要点
- 如果是结论有错，那请立即请求撤稿

Q&A：回答同学们的一些问题

问题 4：有没有某作者/某篇文章的storytelling是您个人十分推崇、值得学习的

回答：

- 绝大多数知名的大组的文章质量都不会差，story telling本身也肯定都有值得学习的地方
- 我个人很喜欢时不时做一些天马行空工作的教授的文章，比如Noah Snavely，Bill Freeman, 的文章写得很写意。
- Stefan Roth的文章，永远会你可能会有的问题在未来几句话以内解决了

Q&A：回答同学们的一些问题

问题 4：有没有某作者/某篇文章的storytelling是您个人十分推崇、值得学习的

5.1. Experimental results

Tables 1 and 2 summarize the quantitative results and Figure 7 shows qualitative comparisons.

Comparison to alternative approaches. In Table 1 we

Does recognizing objects help scene completion? Previous work has shown scene completion is possible without

Does scene completion help in recognizing objects? To answer this question, we trained a model with a loss only ac-

Does synthetic data help? To investigate the effect of using synthetic training data, we compared models trained

Does a bigger receptive field help? In Table 3, the networks labeled [Basic] and [Basic+D] have the same number

Does multi-scale aggregation help? Comparing the network performance with and without the aggregation layer

Do different encodings matter? The last three rows in Table 3 compare different volumetric encodings: projec-

Is data balancing necessary? To balance the empty and occupied voxel examples, we proposed to sample the empty

Limitations. Firstly, we do not use any color information, so objects missing depth such as "windows" are hard to han-

Song et al.: Semantic Scene Completion from a Single Depth Image. CVPR 2017

Q&A：回答同学们的一些问题

问题 5：论文投稿周期一般定在多久比较合理；论文撰写和实验的时间按什么比例分配比较合理？论文撰写穿插实验进行会让效率变得更高吗？

回答：非常看项目，可长可短，但是不建议为了赶文章而赶。论文撰写尽早开始，这样也会给你理清思路，更知道论文需要什么实验。

谢谢！

Sida Peng

Jun Gao

Songyou Peng

Qianqian Wang