Implementation: Virtual Multi-view Fusion for 3D Semantic Segmentation

Yanjie Ze, July 2021

Website: http://yanjieze.xyz

1 Method Overview

如下图所示，主要分为training和inference。

- 在training的时候，选择view和相机的intrinsic和extrinsic，进行render，获得2D data和ground truth。
- 在inference的时候，进行2D的semantic segmentation。

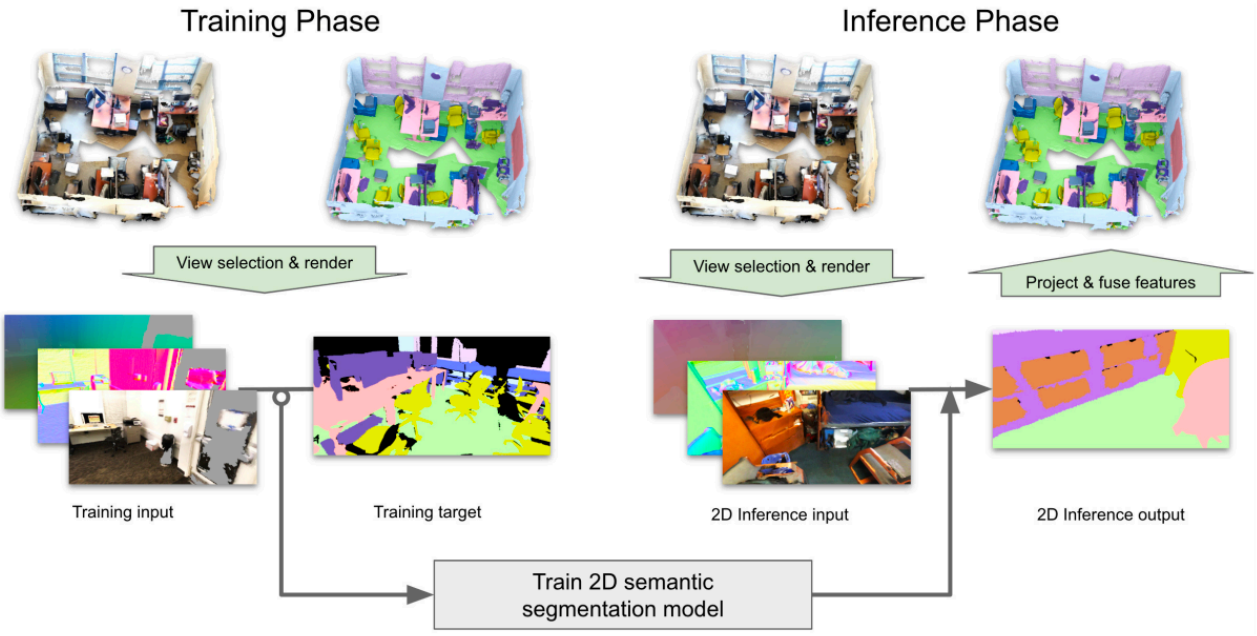

Fig. 1: Virtual multi-view fusion system overview.

2 Virtual View Selection

相机内参：用更大的FOV。

相机外参：如图2和图4所示，用了好几种增强的方法：

- 位置坐标用uniform sampling，视角是top-down的。
- scale-invariant sampling。

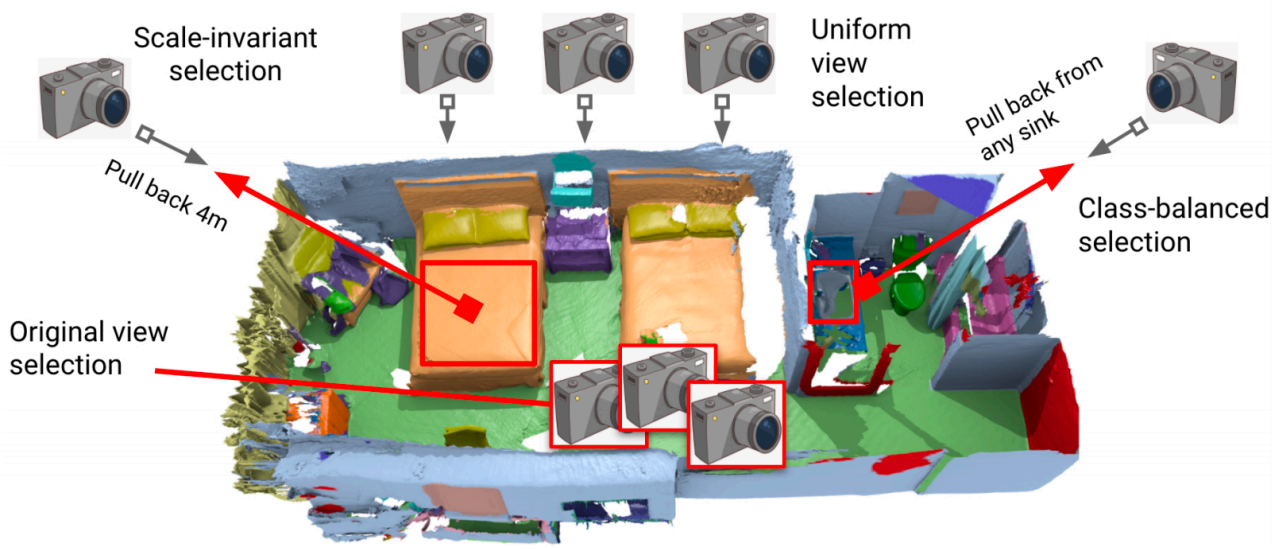

Fig. 2: Proposed virtual view selection approaches.

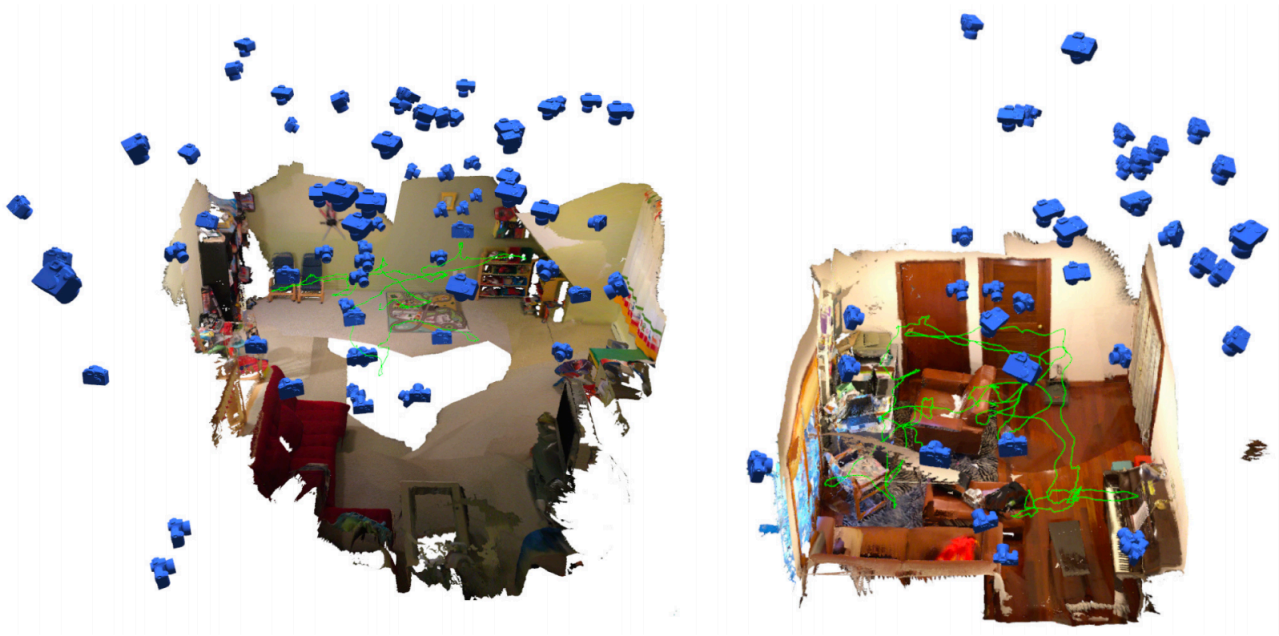

Fig. 4: Example virtual view selection on two ScanNet scenes. Green curve is the trajectory of the original camera poses; Blue cameras are the selected views with proposed approaches. Note that we only show a random subset of all selected views for illustration purposes.

3 Multi-view Fusion

3.1 2D semantic segmentation model

Feature extractor: **xception65**

Decoder: **DeepLabV3+**

Pretrain: **Classification Model on ImageNet**

3.2 3D fusion of 2D semantic features

将点云project到2D图像上，depth相同的点才对应。（depth check）

注意，这比从2D图像进行ray casting要快。

具体过程：

首先，根据下面这个公式，用相机内参、外参，将三维的点投影二维上，获得坐标。

$$\mathbf{x}_{k,i} = \mathbf{K}_i(\mathbf{R}_i\mathbf{X}_k + \mathbf{t}_i)$$

以及相机与这个三维点的距离，如下公式。

$$c_{k,i} = \left\|\mathbf{X}_k - \mathbf{R}_i^{-1}\mathbf{t}_i\right\|_2$$

然后，三维点从每个view采集获得feature vector，如下公式。

$$\mathcal{F}_k = \{\mathbf{f}_i(\mathbf{x}_{k,i}) \mid \mathbf{x}_{k,i} \in \mathcal{A}_i, |d_i(\mathbf{x}_{k,i}) - c_{k,i}| < \delta, \forall i \in \mathcal{I}\}$$

获得feature后，做一个**平均值**（而不是直接取最大值）。因为这个效果比较好。

4 实现思路

4.1 第一步，写dataloader和renderer

第一个问题，怎么从一个3D mesh来render一个图像?

先参考这篇medium上的文章试一试: How to render a 3D mesh and convert it to a 2D image using PyTorch3D

后来开始参照pytorch3d的官方文档，尝试写一个renderer。

renderer的具体结构如下: