**《计算机视觉（2）》实验报告**

实验十 语义分割算法

|  |  |  |
| --- | --- | --- |
| **实验小组成员**  **（学号+班级+姓名）** | **分工及主要完成任务** | **成绩** |
| 201800820149+18数科+徐潇涵 | 论文研读、算法学习、论文讲解、实验报告 |  |
| 201800810253+18 数  科+王萦 | 论文研读、算法学习、算法实验、PPT制作 |  |
|  |  |  |

山东大学

2021年3月

完成综合实验，实现计算机视觉的基本任务——语义分割算法。采用的处理算法不限。

请大家按科技期刊论文模式写明算法的研究背景、目的意义，算法基本原理和研究方案，分割算法评价指标和与其他同类算法的实验对比，论文总结等（包括但不限于以上内容）。请注意行文表述要清晰并有条理，图、表、公式格式规范（具体要求可参考doc文件：“电子学报投稿模板20210208100630”）。

实验报告写在如下空白处，页数不限。

实验报告

# 1 研究背景

我们都知道，图像是由许多像素组成，而「语义分割」顾名思义就是将像素按照图像中表达语义含义的不同进行分组。2014年，来自UC Berkeley 的Trevor Darrell 组在2014年提出了全连接的卷积神经网络，开启了卷积神经网络应用在语义分割的先河。语义分割算法中，一般采取的卷积神经网络是FCNN。由于用于图像分类和检测的卷积神经网络关注的是图像级，所以在卷积网络的后面会有全连接层，用于降低网络的维度，输出我们想要的分类信息和位置信息；但是语义分割关注的是图像的像素级（Pixel-level），我们希望输入的是一张图片，输出的仍然是尺寸基本一致的图片，所以在FCNN中，去掉了一般卷积网络后面的全连接层。

同时，大规模且具有详细标注的数据集是充分释放深度学习算法在3D场景理解领域潜力的重要前提。然而，由于数据采集和数据标注的高昂成本，当前公开可用的数据集要么仅具有有限的语义标注，要么是在相对较小的空间范围内采集得到的，这在一定程度上限制了在大场景3D点云细粒度语义理解的发展。

所以，找到可以应用于大场景3D点云数据集，对其自动进行语义分类的基于卷积神经网络的语义分割算法，是实现大规模三维重建的关键点。

# 2 目的意义

在本篇论文中，构建了一个城市规模的摄影测量点云数据集，命名为SensatUrban，其中包含三个英国城市(伯明翰，剑桥以及约克)7.6平方公里中的近30亿具有详细语义标注的点，其中每个点都被标记为13个语义类别之一。相比于现存的3D数据集而言，SensatUrban是现有的可以捕获复杂城市结构和真实颜色信息的最大城市规模3D点云，为城市规模三维重建的进一步研究提供了关键数据。

此外，论文还在该数据集上对基于卷积神经网络的语义分割算法进行了性能评估，并对结果进行了全面分析，通过改变变量因子，从不同的角度和方向确定了适用于大规模点云的不同的语义分割算法，并提出了城市规模点云理解的几个关键挑战，其中包括如何对大规模点云进行采样和分区，颜色信息是否有助于语义分割，数据类别不平衡的影响及模型跨城市泛化。这些挑战与思考，可为不久的将来，自动驾驶飞行器智能地在人口稠密的城市和农村地区中导航时提供决定性帮助，精确的高分辨率三维实现地图也是新兴网络物理领域、智能城市、智能交通的必要组成部分。

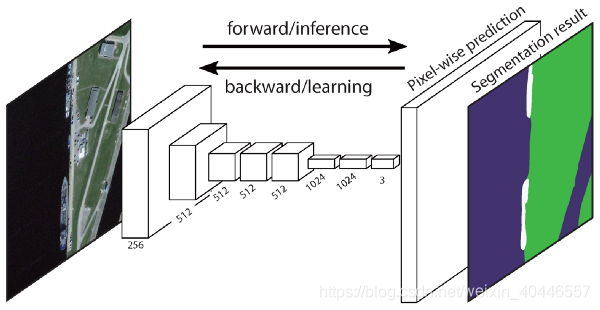
# 3 算法基本原理

## 3.1 语义分割算法基本原理

现有的语义分割算法主要被分为三类探讨：基于区域的语义分割、全卷积网络语义分割和弱监督语义分割，由于论文的应用性，这里我们主要来讨论全卷积语义分割。

原始的完全卷积网络（FCN）学习从像素到像素的映射，而不提取区域建议。刚才也提到，FCN是经典CNN的延伸。其主要思想是使经典的CNN以任意大小的图像作为输入。CNN仅接受和生产特定尺寸输入的标签的限制来自完全连接的固定层。与之相反，FCN只有卷积层和池层，它们能够对任意大小的输入进行预测。

在经过一个全卷积的预训练的网络之后，比如说VGG，由于池化操作降低了图像空间维度，特征map仍然需要被上采样。与简单的双线性插值不同，反卷积层可以学习插值。该层也叫上卷积（upconvolution），全卷积（full convolution），转置卷积（transpose convolution）或者分数化卷积（fractionally-strided convolution）。下图就是FCN的基本构架。



图表 1 FCN的基本构架

## 3.2 KPConv算法基本原理

KPConv算法其实就是相对于PointNet提出的一种基于Kernel Point的新型点云卷积。相当于将Kernel Point当成了每个点的参照物，去计算与这些Kernel Point的权重去更新每个点。提出了两种不同类型的Kernel Point，一种是刚性的Rigid Kernel，相当于均匀分布在每个点的周围球面；一种是可变的Deformable Kernel ，通过在刚性的基础上通过网络去学习它的位置变化。

这个算法相对于PointNet来说整个网络GPU占用少、速度快，可以在不定个数的点云上直接训练，因此效果也好。

### 3.2.1 定义点云上的核函数

首先定义一下点云上的某个点以及其对应的特征，通常由所定义的点云的卷积可以写成如下的形式：

其中代表其某个局部邻域，可以看做是某个中心点以半径 画出的一个球型邻域，即

在不同的工作中，往往上面公式中的核函数才是关键，取以为中心的相邻点作为输入。我们将局部的区域中心化，对于每一个点，可以通过中心化将其转换成。最终，点云上的核函数可以定义为：

### 3.2.2 核心点网络层

使用采样策略来控制每一层输入点的密度。为了保证采样点位置的空间一致性，他们采用网格采样（即通过空间划分网格，对每个网格内采取中心点）。因此每一个网格的重心，被作为采样点。

## 3.3 RandLA-Net算法基本原理

### 3.3.1 随机降采样

对于一个覆盖数百米范围、由百万量级的点组成的大场景点云而言，如果希望将其直接输入到深度神经网络中进行处理，那么持续有效地对点云进行逐步地降采样，同时尽可能地保留有用的几何结构信息是非常有必要的。

相比于其它降采样方式，随机降采样具有以下两个优点：

1)计算效率高, 因为是常数计算复杂度, 与输入点数无关

2)内存开销少，采样过程并不需要额外的内存消耗

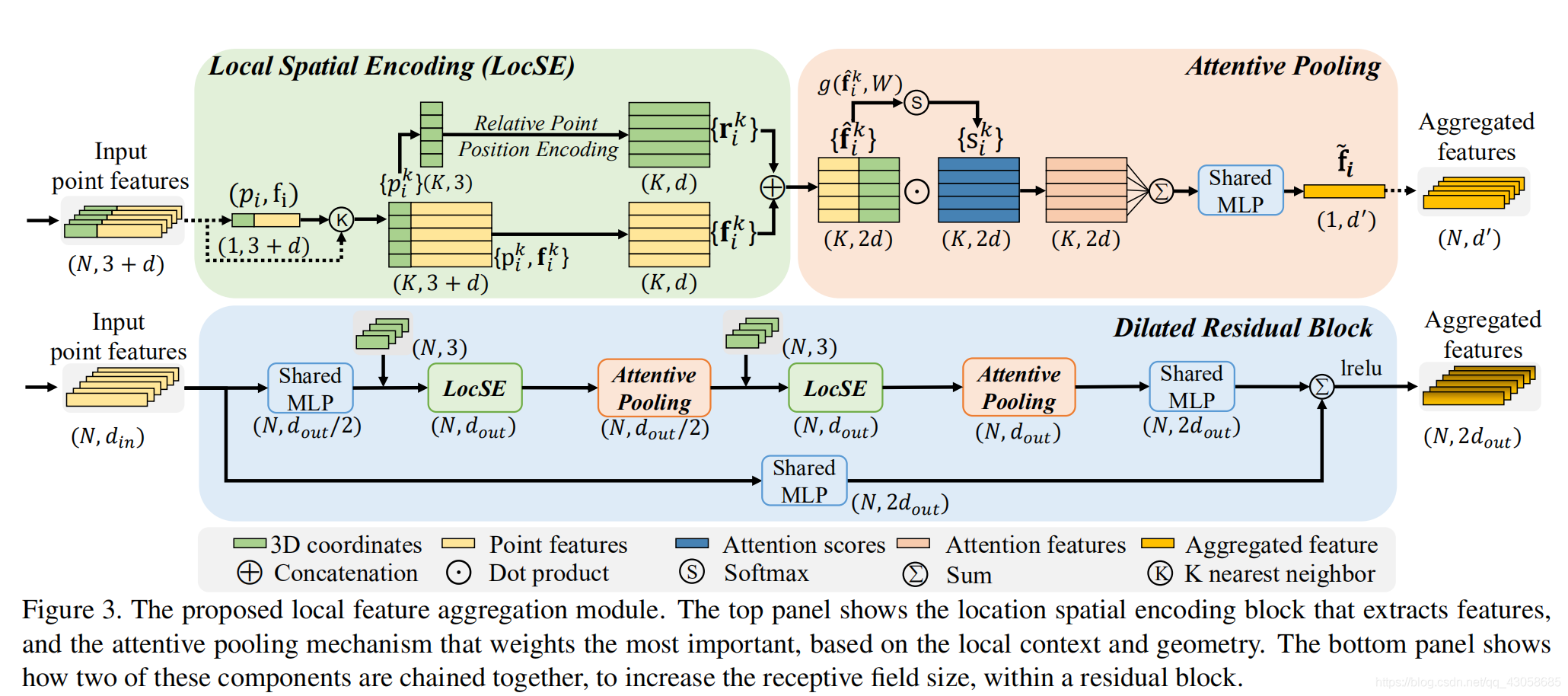
### 3.3.2 局部特征聚合

随机地对点云进行降采样势必会导致有用的信息被丢失，为了缓解这个问题，我们进一步提出了与随机采样互补的局部特征聚合模块(Local feature aggregation)。如图所示，该模块主要包括三个子模块:

1)局部空间编码(LocSE)

2) attentive pooling

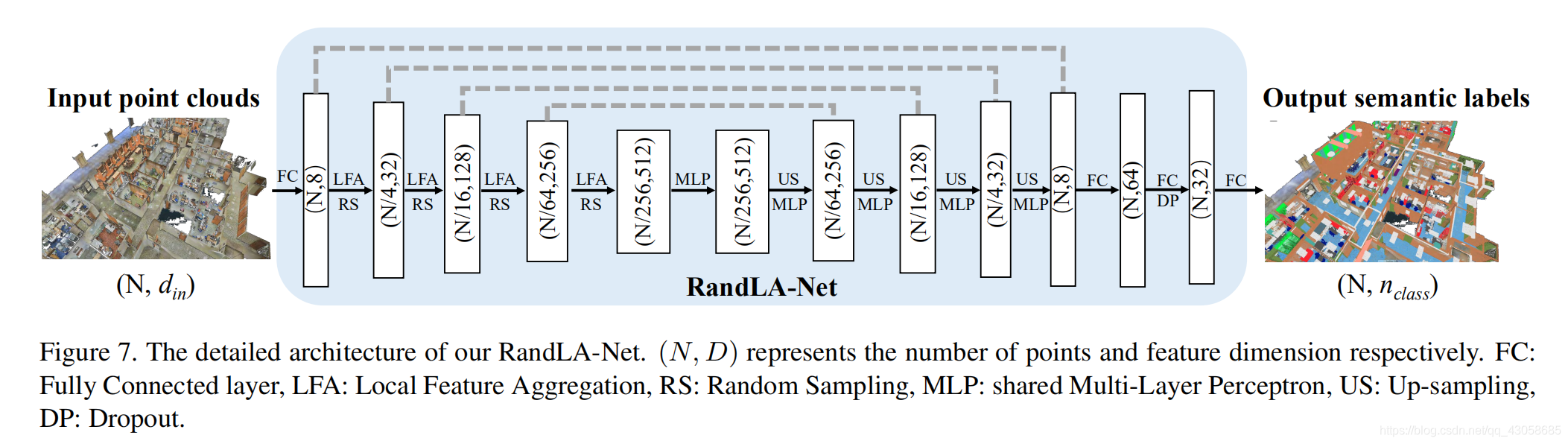
3)扩张残差块(dilated residual block)



图表 2 局部特征聚合模块

### 3.3.3 RandLA-Net网络结构

最后，我们将随机采样以及局部特征聚合模块组合到一起，基于标准的encoder-decoder结构组建了RandLA-Net。网络的详细结构如下图所示，可以看到，输入的点云在RandLA-Net中持续地进行降采样以节约计算资源及内存开销。此外，RandLA-Net中的所有模块都由简单高效的feed-forward MLP组成，因此具有非常高的计算效率。最后，在解码器中的上采样阶段，不同于广泛采用的三线性插值(trilinear interpolation)，我们选择了更加高效的最近邻插值(nearest interpolation)，进一步提升了算法的效率。



图表 3 RandLA-Net网络结构

# 4 研究方案

## 4.1 语义分割算法的选择

基于神经网络的语义学习算法一般分为三种

1）基于体素的方法(Voxel-based)。可以轻松地应用成熟的三维卷积神经网络架构，但是这些技术通常需要大量的计算和内存使用，因此无法轻松地扩展到城市规模的点云。

2）基于2D投影的方法(Projection-based)。关键的几何信息很可能会在投影步骤中丢失，因此不适合学习城市规模方案中相对较小的对象类别。

3）基于点的架构(Point-based)。这类技术主要基于简单的MLP学习每点语义。相比而言，这些管道在计算上趋于高效，并且有可能保留每个3D点的语义。

## 4.2 实验的基准与四个相关实验

首先我们以上面提到的三类算法中的代表性算法对数据集进行实验，以此来作为实验的基准，选择的算法有以下这些：

* Sparseconv (Voxel-based)
* TagentConv (Projection-based)
* PointNet/PointNet++/SPGraph/KPConv/RandLA-Net (Point-based).

这些将会在6 同类算法实验对比中提到。

同时，我们还会在7 论文总结中，更进一步地分析这些算法在调整一些参数后，对于提出的四个问题的解决实现与思考。

# 5 分割算法评价指标

为了在论文中定量地分析语义分割算法的优劣，特在此总结一些常用的评价指标。

## 5.1 mIoU均交并比

语义分割说到底还是一个分类任务，既然是分类任务，预测的结果往往就是四种情况：

表1 混淆矩阵

|  |  |  |  |
| --- | --- | --- | --- |
|  | | 真实值 | |
| Positive | Negative |
| 预测值 | Positive | True Positive（TP）真阳性 | False Positive（FP）假阳性 |
| Negative | False Negative（FN）假阴性 | True Negative（TN）真阴性 |

均交并比（Mean Intersection over Union）：为语义分割的标准度量。

首先假定数据集中有类，0通常表示背景。使用表示原本为类同时预测为类，即真阳性（TP）和真阴性（TN）。表示原本为类被预测为类，即假阳性（FP）和假阴性（FN）。

如果第类为正类，当时，那么表示TP，表示TN，表示FP，表示FN。

IoU计算真实值和预测值两个集合的交集和并集之比。这个比例可以变形为TP（交集）比上TP、FP、FN之和（并集）。即：

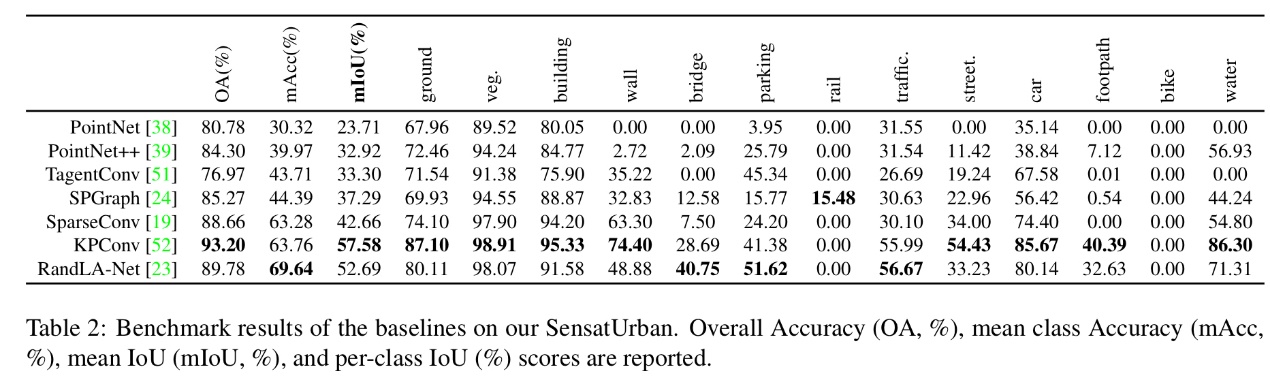
Mean IoU是在所有类别的IoU上取平均值。其公式如下：

# 6 同类算法实验对比

上文提到，我们实验基准选择的算法有以下这些：

* Sparseconv (Voxel-based)
* TagentConv (Projection-based)
* PointNet/PointNet++/SPGraph/KPConv/RandLA-Net (Point-based).

为确保公平，我们遵循了这些方法原始的实验设置及参数。下表展示的为定量结果：



图表 4 基于SensatUrban数据集实验所得的基准结果

从图表中我们可以看到，基于语义分割算法评价指标，KPConv算法取得了最高的mIoU结果，然而整体分割效果仍然远不能令人满意。数据集中桥梁，铁路，街道，人行道的分割结果依然较差。

# 7 论文总结

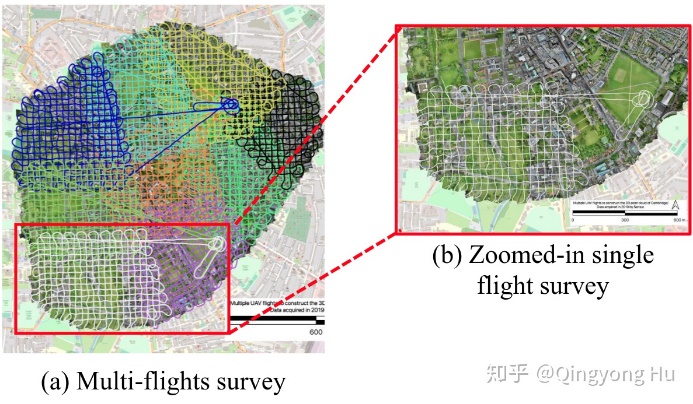
本篇论文通过构建了一个庞大而丰富的城市规模数据集，用于半监督学习方案，通过广泛的基准测试，提出了相关的挑战，并对其进行了一些思考。

## 7.1 数据集的构建

### 7.1.1 数据采集

考虑到通过无人机摄影测量技术相比于类似的制图技术在成本上具有明显的优势，使用了经济高效的固定翼无人机Ebee X来采集高分辨率的航空图像序列。为了完全均匀地覆盖测量区域，所有飞行路径均以网格形式预先规划，并由飞行控制系统（e-Motion）自动执行。

该相机具有拍摄斜角(oblique)和天底(nadir)照片的能力，可确保正确捕获垂直表面。由于电池容量的限制，每次飞行持续40-50分钟，因此并行执行多个单独的飞行以捕获整个区域。而后，我们使用高精度机载实时Realtime Kinemtic (RTK) GNSS系统对这些多个航空图像序列进行地理位置的对应。此外，我们还安排了专业测量师配备高精度GNSS设备测量地面验证点，以评估数据的准确性和质量。数据的分辨率取决于输入图像的数量和3D重建设置。下图展示了覆盖剑桥市所选区域的预先计划的多次飞行的路径。



图表 5 剑桥地区的飞行survey。通过所有的9个飞行计划（左图）整理在一起以覆盖图中所示区域。 不同颜色的线表示无人机的不同飞行路径。圆形路径是起飞和降落的模式。

### 7.1.2 3D点云重建

对于三维点云的重建，采用了软件Pix4D，基于Structure from Motion(SfM) 和 dense image matching原理，实现从捕获的航空图像序列中重建密集和彩色的3D点云。对于伯明翰周边的市区，我们将所有捕获的连续图像输入到Pix4D，总共生成569,147,075个3D点，覆盖了1.6平方公里的面积。 同样，我们为剑桥市附近的市区重建了2,278,514,725点，面积约为4.6平方公里。此外，我们在约克郡也采集了3.2平方公里内的904,155,619个点。

### 7.1.2 标注语义标签

两个原则：

1）每个类别都应具有清晰明确的语义。

2）不同类别的几何结构或外观应有显著差异。

在标注过程中，采用现成的点云标记工具，每个点都被标注为13个语义类别之一，所有标注都经过人工交叉检查，以确保一致性和高质量。



图表 6 本文提出的SensatUrban数据集部分数据示意图。 不同的颜色代表不同的语义标注。可以看到，我们的数据集增加了一些以往数据集中不存在的语义类别，比如桥梁，铁路，水（河流）等。

## 7.2 实验中的四个挑战

四个挑战：

如何高效有效地预处理大量点以输入神经网络？

如何克服神经网络中的数据不平衡问题？

颜色信息是否有助于城市区域的语义分割？

如何将模型进行跨城市泛化？

### 7.2.1 挑战一：分块操作

由于GPU内存的限制，处理城市规模点云的首要挑战就是对整个点云进行分块。 早期的PointNet/PointNet++通常将点云划分为1m×1m的点云块。 然而对于城市规模的点云，采取这种方式将会非常耗时，并且会导致场景中目标的几何形状受到影响。 另一方面，如果将原始点云划分为非常大的点云块，则无法直接将大量的点输入到GPU中。 为了减少每个块中的总点数，KPConv和RandLA-Net中采用了网格或随机下采样， 许多其他方法倾向于使用不同的采样和分块操作。 总的来说，目前还并没有一个统一的方案或预处理步骤来实现对大规模点云数据的分块操作。

将分块操作分为两个步骤：

1）对原始点云进行降采样。文献中有两种选择：

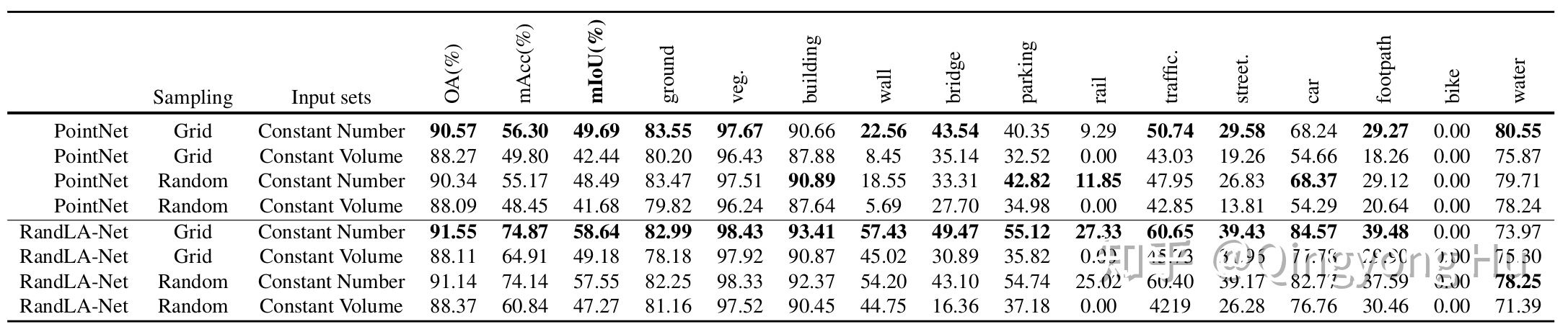
* 网格下采样
* 随机下采样。

两者都可以显著的减少点的总量，但是各有优点。

1. 将独立的点云块输入到网络中。也有两种选择：

* 恒定数量输入集(constant-density input)，我们通过使用预先构建的KDTree来查询中心点的固定数量的相邻点作为输入来实现。
* 恒定体积输入集(constant-volume input), ，我们首先在中心点周围裁剪固定大小的体积（例如8m×8m块），然后随机进行下（上）采样以获得输入集。

把PointNet和RandLA-Net作为baseline，评估了步骤1和步骤2的四种不同组合对整体分割性能的影响。



图表 7 使用不同输入，由PointNet和RandLA-Net获得的定量结果

由图表我们可以得出以下结论：

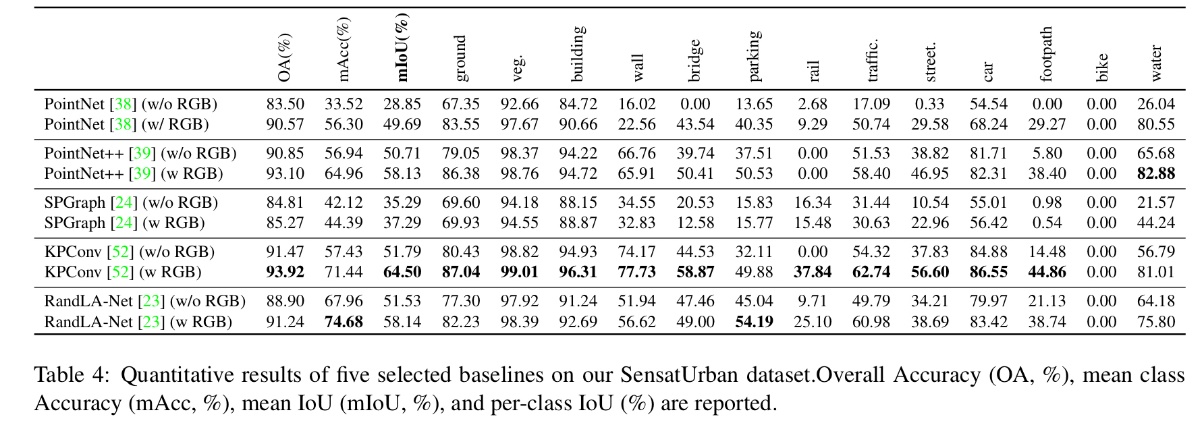
1.第一步使用网格下采样能得到更好的结果；

2.相比恒定体积输入集，在恒定密度输入时基于PointNet或RandLA-Net的框架均能获得更好的分割结果；

3.总的来说，数据准备对于处理大规模城市点云确实非常重要。不同的预处理步骤带来的性能差距在10%(mIoU)以上，因此有必要进一步探索更为有效、统一的数据准备方法。

### 7.2.2 挑战二：颜色信息

进一步探索颜色信息对分割性能的影响：



图表 8 报告了数据集上五个选定基线的定量结果

由图表我们可以得到以下结论：

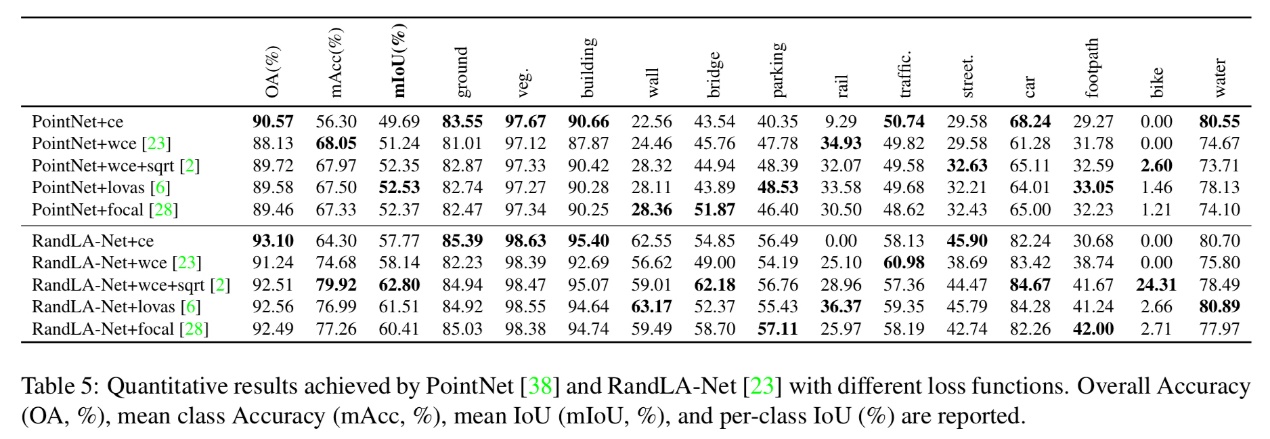
1.输入几何坐标以及颜色信息时，PointNet/PointNet ++，KPConv和RandLA-Net均可实现更高的分割精度。 这是因为如果仅提供3D坐标，许多城市中的语义类别在本质上就无法区分（例如，桥梁，水与地面）。

2.对于SPGraph，由于几何分区性能主要取决于输入点云的几何坐标，因为增加RGB信息后并不会对性能有明显的提升。

### 7.2.3 挑战三：缓解类别不平衡

为了缓解类别不平衡所带来的问题，常用的解决方案是使用不同的损失函数。

在这里，我们以PointNet和RandLA-Net为基准，评估了五个现成的损失函数的有效性，包括：Cross Entropy Loss(CE) 交叉熵损失、weighted corss entropy（WCE）加权交叉熵损失、WCE+sqrt、Focal loss、balanced corss entrop (BCE)均衡交叉熵损失等。



图表 9 具有不同损失函数的PointNet和Rand-Net获得的定量结果

通过观察图表，我们可以发现，使用更加有针对性的损失函数确实可以提高分割性能，其中mIoU分数最高可提升5％。

### 7.2.4 挑战四：跨城市泛化

当将训练后的模型直接应用于未曾见过的城市场景时，所有基线的分割性能都会大大下降。大多数方法的mIoU得分差距高达20％。其中，地面和建筑等主要类别的性能下降并不严重，而铁路，街道和水等类别的下降非常明显。

我们认为：

1、不均衡的语义类别分布可能是阻碍模型泛化能力的关键因素，因为模型倾向于拟合主要类别，而无法很好地学习到次要类别的鲁棒性；

2、很难将某些城市类别的形态变化从一个数据集推广到另一个数据集。