重庆理工大学学报(自然科学)
Journal of Chongqing University of Technology(Natural Science)
2022 年 第 36 卷 第 2 期
Vol. 36 No. 2 2022
doi: 10.3969/j.issn.1674-8425(z).2022.02.017

基于 RGB-D 图像的移动端点云分割方法研究

余方洁[1,2],王 斌[1]

(1. 中国科学院长春光学精密机械与物理研究所,长春 130033;
2. 中国科学院大学,北京 100049)

摘 要: 近年来,深度传感器和三维激光扫描仪的普及推动了三维点云处理方法的快速发展。针对传统的前后端分离的点云分割模式,提出了一种使用移动端设备进行三维数据采集与处理的一体化技术方案。基于谷歌的 AR Core 开发平台,进行了安卓设备上的深度图获取实验,深度图可进一步转换为点云数据;通过对模型轻量化方法的研究,改进了 PointNet 网络,使模型参数量减少为原来的 1/5,同时具有约 73% 的分割精度;最后利用 TensorFlow Lite 移动端深度学习框架,将改进的 PointNet 网络成功部署到了安卓智能手机上,量化后的 tflite 模型仅 268 kB 大小,在启用 GPU 加速后,对单幅场景点云数据的推断速度约为 0.7 s。实验结果表明了提出方法的可行性。

关 键 词: 点云分割;深度图获取;深度学习框架
中图分类号: TP391 **文献标识码:** A **文章编号:** 1674-8425(2022)02-0126-09

点云是一种重要的三维数据结构,它是具有相同空间参考的一组物体表面特征点集合。点云数据语义分割作为三维场景理解的关键性技术,在无人驾驶、数字城市、高精地图、VR、AR 等领域得到了广泛的应用。传统的点云分割流程包括前期数据采集与桌面端后期处理 2 个阶段,其中常用的采集设备有激光雷达、TOF 相机等,它们通常价格昂贵且有安装结构要求,此外这种前后端分离的工作模式在实时性方面也表现较差。本文通过对深度图获取原理及点云分割方法的研究,提出了一种仅凭借智能手机等轻量级移动端设备就可以实现三维数据采集和准实时语义分割的技术方案,旨在进行前后端的无缝集成。

语义分割是计算机视觉的基本研究任务之一,是将输入数据映射到现实世界中事物的可解释类别的技术。近年来,随着大数据的出现与计算机硬件性能的大幅提升,基于深度学习的点云分割方法已经成为当前的主流。根据对三维点云数据处理方式的不同,又可分为间接语义分割法和直接语义分割法。间接语义分割法主要借鉴了以往二维图像分割的经验,通过将点云数据转换为多视图或体素网格,很大程度上利用二维的网络模型间接达到分割的目的。在多视图的方法中,Su 等[1] 提出了 MVCNN(multi-view convolution-

收稿日期: 2021-05-20
基金项目: 国家自然科学基金项目(11703024)
作者简介: 余方洁,男,硕士研究生,主要从事深度学习和计算机视觉研究,E-mail: 907040864@qq.com;**通讯作者** 王斌,男,博士,研究员,主要从事图像恢复、深度学习、波前探测等研究,E-mail: 175969722@qq.com。

本文引用格式: 余方洁,王斌. 基于 RGB-D 图像的移动端点云分割方法研究[J]. 重庆理工大学学报(自然科学),2022,36(2):126-134.
Citation format: YU Fangjie, WANG Bin. Research on point cloud segmentation method of mobile devices based on RGB-D image[J]. Journal of Chongqing University of Technology(Natural Science),2022,36(2):126-134.

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

al neural network)，将三维目标投影为多个不同视角下的二维图像，对每个视图进行特征提取并经过特征聚合得到最终的分割结果; Feng 等[2] 在 MVCNN 的基础上提出了 GVCNN(group-view convolutional neural network)，对不同视图提取的特征进行分组，提高了网络性能; Zeng 等[3] 基于 FCN (fully convolutional networks)[4] 并结合 HHA 投影技术实现了对 RGB-D 数据的语义分割; Wu 等[5] 借鉴 SqueezeNet[6] 的设计思路提出了 SqueezeSeg 网络，使用球面投影的方法将三维点云转换为二维图像输入到 SqueezeSeg 中进行分割。不同于多视图方法对数据进行降维的操作，体素化方法致力于构建三维语义分割网络，Maturana 等[7] 将卷积运算推广到三维空间，最早提出了基于体素数据的 VoxNet 模型; 针对体素网格分辨率低的限制，Tchapmi 等[8] 提出了 SegCloud 网络; 为了更加合理地利用点云数据的特点并减少计算量，Riegler 等[9] 基于八叉树结构提出了 OctNet，Klokov等[10] 基于 Kd-tree 结构提出了 Kd-Net。在点云直接语义分割方法中，Qi 等[11] 在 CVPR2017 上开创性的提出了 PointNet 网络，它不需要对输入数据做任何变换，可以直接通过端到端训练的方式实现点云分割，为三维场景理解指明了新的发展方向; 由于对空间邻域信息感知较少，Qi 等[12] 又提出了 PointNet 的改进版本 PointNet ++，通过对数据进行采样、分组的方式提取局部特征，并使用 MSG (multi-scale grouping) 、MRG (multi-resolution grouping) 等策略自适应处理密度不均匀的点云数据。

深度图可以等效为三维密集点云，针对移动端深度图像获取的任务，谷歌的 Valentin 等[13] 提出了 depth-from-motion 算法，它可以从运动中恢复深度信息; 商汤科技的 Yang 等[14] 基于多视图关键帧的深度估计方法，提出了一个手机端实时单目三维重建系统 Mobile3Drecon。移动端设备由于资源和算力等受限，因此在对深度学习模型进行部署时，通常需要经过网络压缩处理，减少其参数量与运算量。在此方面，研究者们也提出了许多模型压缩方法，如紧凑网络、参数剪枝、低秩分解、知识蒸馏等。

同时，许多轻量化网络也被证明具有相当好的效果，已经在移动端设备上进行了成功的应用，如 SqueezeNet[6]、MobileNet[15]、ShuffleNet[16]、Xception[17] 等。

本文利用移动端设备轻量、便捷的优势，对安卓平台的点云分割方法进行研究，主要贡献如下:

1) 提出了一种仅使用移动端设备进行三维数据采集与处理的一体化解决方案，包括了深度图像获取、点云转换、模型压缩、移动端部署与加速整个处理流程，改变了以往前后端分离的应用模式。

2) 对点云语义分割网络 PointNet 进行轻量化设计，使参数量减少为原来的 1/5，同时在测试集上的平均分割精度达到了 73%，在启用 GPU 加速后，对单幅场景点云数据的推断速度约为 0.7 s。

系统整体架构如图 1 所示。

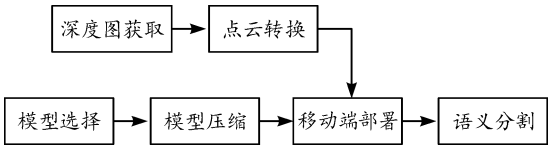


图 1 系统整体架构

1 三维点云分割方法

1.1 点云特点

1) 无序性

点云是点数据的集合，这意味着其中的数据是没有顺序的，因此对于按照不同排列输入的同一集合的数据，点云处理模型应该对于这种变化具有不变性。

2) 稀疏性

点云的本质是对空间中物体形状的低分辨率重采样，而且由于采集过程中存在的目标遮挡等情况，其获得的几何信息是片面、不完整的。

3) 密度不均匀性

不同方式获取的点云数据，其点间距、密集程度等往往相差很大，即使同一批次采集的点云数据，也经常存在密集和稀疏的区域。

1.2 点云分割方法

随着深度学习技术的不断成熟，其在点云语义分割领域也得到了越来越多的应用，取得了比传统方法更优的效果。根据点云数据处理方式的不同，基于深度学习的点云语义分割方法可分为间接语义分割法和直接语义分割法[18]。

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

1.2.1 间接语义分割法

间接语义分割法需要将点云数据转换为其他的表示方式,通过这种数据结构的转变间接实现点云数据语义分割,根据转换类型的不同,分为二维多视图法和三维体素化法两类。

二维多视图法的基本思路是对点云数据进行投影,得到多个视图的二维图像,再对这些二维图像使用经典的语义分割模型如 FCN[18]、U-Net[19]、Segnet[20]、DeepLab[21]等进行处理。多视图法很好地克服了点云数据的非结构化特性,通过卷积神经网络提取各个视图的特征,并将多个特征的信息整合为更高级的语义特征,得到最终的分割结果。但是多视图的方法在简化点云数据处理的同时,也丢失了原始数据中包含的大量关键空间信息,影响了点云分割的精度。

体素(Voxel)是二维空间中像素概念的推广,体素化的本质是将无序、非结构的点云数据规则化,这解决了点云数据的特征学习问题,但点云的稀疏性导致了体素网格数据冗余多、占用空间大、分割效率低。此外体素化方法相比二维图像增加了一个维度,计算过程中的资源开销更大,这也一定程度上限制了体素网格的分辨率。总的来说,体素化方法在现阶段实用性相对较低。

1.2.2 直接语义分割法

为了充分利用点云数据本身的特性并降低计算复杂度,人们开始研究直接对原始点云数据进行处理的网络模型,其中具有开创性和代表性的是 Qi 等[12]提出的 PointNet 网络。

PointNet 网络设计时充分考虑了点云的无序性、旋转和平移不变性、空间相关性,并经过严格的推理证明提出了 2 个定理:一是证明了 PointNet 网络能够拟合任意的连续集合函数,二是其网络结构对于有噪声和数据缺失的点云同样具有鲁棒性。对于点云无序性,PointNet 使用对称函数来提取点云数据的特征;为了保证旋转和平移不变性,PointNet 网络通过训练一个小型网络 T-Net 得到转换矩阵,并用来对输入的点云数据进行空间变换;为了有效利用点云之间的空间关系,PointNet 网络使用跳跃连接(skip connection)的方式将浅层特征与高层特征相结合。

虽然 PointNet 网络在点云分割方法中取得了突破性的进展,但由于对各个点的操作过于独立,以及未考虑点云的密度不一致性,因此仍有很大的改进和提升空间。后来研究者们在 PointNet 网络的基础上又提出了一系列的优化算法,这些方法主要有基于邻域学习的方法、基于图卷积神经网络的方法、基于注意力机制的方法、基于循环神经网络的方法等。

2 深度图获取与转换

2.1 常用获取方法

深度图(Depth map)是一种特殊的二维灰度图像,它的每个像素值反映的是场景中各个物点距离传感器的实际距离,是物体可见表面几何形状的写真。深度图像通常是与彩色 RGB 图像经过配准的,被合称为 RGB-D 图像,它们的像素点之间具有一一对应的关系。

深度图的获取有多种方法,根据传感器工作原理的不同,分为主动式和被动式两类。主动式方法主要有激光雷达成像法、结构光法、TOF 相机、莫尔条纹法、坐标测量机法等,它们的共同特点是在获取深度信息时需要激光等光源主动向物体发射信号并解析回波,这类方法也是目前在实用中被研究最深入、使用最广泛的一类;此外还有被动式的单目、双目、多目立体视觉的方法,这些都是基于多视图几何原理的通过对图像的理解来恢复三维结构,其相对主动式的方法而言成本更低也更加便捷,因此近年来也受到了越来越多的关注。

2.2 depth-from-motion 算法

本文通过对深度图获取方法的研究,探索了在移动端设备上创建深度图的解决方案,结果表明 depth-from-motion 算法具有可行性,其实质是属于单目深度估计算法的一种,可以从运动中恢复深度信息。

depth-from-motion 算法由谷歌的 Valentin 等[13]研究者于 2018 年提出,是一种专门针对智能手机设备获取深度图任务设计的算法,使得在大多数智能手机上仅通过标准的彩色摄像头就可以在单核 CPU 上创建 30 Hz 频率的密集、低延迟的深度图。

如图 2 所示,展示了 depth-from-motion 算法的基本处理流程。算法的第一步是从过去的图像帧中选择适合与当前帧进行立体匹配的关键帧,接

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

着使用关键帧和当前帧之间的相对 6 DoF(自由度) 位姿进行极性图像校正，再根据视差就可以得到估计的稀疏深度图。将稀疏深度图送到快速双边求解器(fast bilateral solver) 中进行求解，生成与之相应的双边深度网格(bilateralgrid of depth) ，双边深度网格可以根据需要转换为经过时空平滑的密集深度图。为了保证生成的深度图与 RGB 图像对齐，在生成双边深度网格后，还需要使用最新的图像帧对其进行切片，再经过后期渲染就能得到密集的平滑深度图。

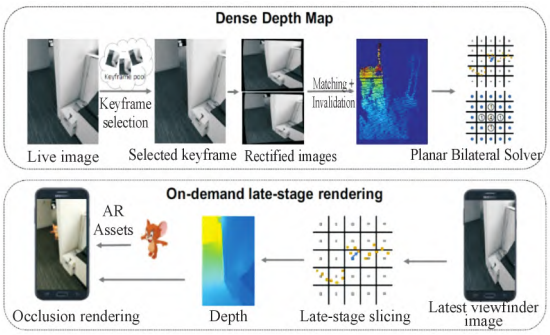
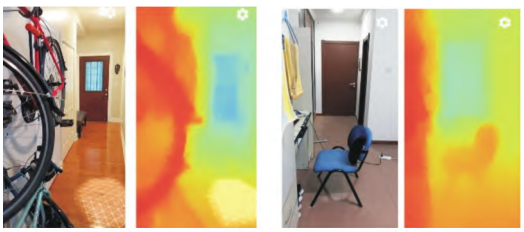


图 2 depth-from-motion 算法流程

谷歌现已开源了集成有 depth-from-motion 算法的增强现实开发平台 ARCore，并支持 iPhone、iPad、华为、三星、小米等制造商的多个型号的设备。本文使用 Android Studio 4.0 开发环境，并借助 ARCore SDK 进行了华为 nova3 智能手机设备上的深度图获取实验，图 3 中展示了谷歌给出的官方效果图(a) 与自己创建的深度图(b) 对比:



(a) 官方效果图 (b) 本文结果

图 3 深度图获取实验

2.3 相机模型

深度图像中包含有被摄场景的三维空间信息，通过对相机成像模型的研究，可以从深度图中解算出每个像素点所对应的三维坐标，生成点云数据。深度图到点云的转换利用的是多视图几何的原理，其实质是图像坐标系到相机坐标系的变换，如图 4 所示。

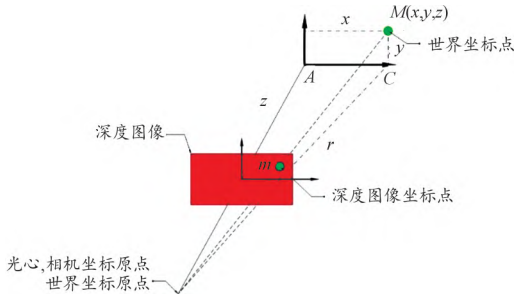


图 4 相机模型

转换过程中需要已知相机的内标定参数 c_x、c_y、f，其中 c_x、c_y 表示的是相机成像时光心在像素平面坐标系下的坐标，f 是相机焦距。另外记焦距 f 与单个像素在 x、y 方向实际距离 dx、dy 的比值为 f_x、f_y，由此可以得到相机的内参数矩阵:

$$\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

记像素平面坐标系中像素点的坐标为 u、v，相机坐标系中空间点的坐标为 X、Y、Z，如图 4 所示，根据相似三角形原理可知:

$$\begin{cases} u = f_x \dfrac{X}{Z} + c_x \\ v = f_y \dfrac{Y}{Z} + c_y \end{cases} \tag{2}$$

式(2) 实际上已经给出了像素平面坐标与相机空间坐标的转换公式，为了表示方便，将其写为矩阵的形式:

$$Z\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{3}$$

式(3) 给出的变换就是我们需要的转换公式，它表示了从深度图像上读取像素坐标 u、v 以及深度值 Z，利用相机内参数矩阵，就可以解算出对应的三维空间坐标 X、Y、Z。

3 深度学习模型压缩算法

深度学习在计算机视觉、自然语言处理等领域取得显著效果的同时，模型的结构也变得越来越复杂。对于移动设备来说，其内存空间有限、运算能力有限、续航时间短、散热条件不利，这都将导致深度学习模型在移动端设备上进行实际部署

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

时,将会遇到很大的限制。因此使用一定的方法对模型进行精简,使之更加轻量且具有相当的准确率是非常必要的,对于深度学习模型的移动端部署具有重要的意义。

目前主要的模型压缩方法有紧凑网络、参数剪枝、参数量化、知识蒸馏等,根据压缩策略的不同又可分为压缩结构与压缩参数两类[22]。两类模型压缩方法的技术描述如表1所示。

表1 深度学习模型压缩算法

类别	技术	描述
压缩结构	紧凑网络	从卷积核、特殊层、网络结构这3个级别重新设计网络
	知识蒸馏	将教师模型中的信息迁移到较小的学生模型
压缩参数	参数剪枝	选择衡量参数重要性的评价准则,基于该准则移除一部分重要性较低的神经元
	参数量化	将网络参数类型从float32量化为更低位数
	低秩分解	将高维参数矩阵分解为若干个小规模稀疏矩阵
	参数共享	利用矩阵或聚类等方法将全部网络参数映射到少量数据上,删除冗余参数

在对深度学习模型进行压缩的实践过程中,出现了几种具有代表性的轻量级网络,它们在减少网络参数量的同时保持了网络的性能,使移动终端、嵌入式设备运行神经网络模型成为可能,对其后的轻量级网络设计具有重要的启发与借鉴意义,表2对几种经典的轻量级网络进行了对比总结。

表2 几种轻量级网络对比

网络名称	TOP1准确率/%	参数量/M	CPU运行时间/ms	核心策略
SqueezeNet[13]	57.5	1.25	–	fire model的设计、延迟降采样
MobileNet[15]	70.6	4.2	113	深度可分离卷积
ShuffleNet(2×)[16]	73.7	–	108.8	深度可分离卷积、通道洗牌
Xception[17]	79.0	22.86	–	改进的深度可分离卷积

4 实验及结果分析

4.1 数据集获取与处理

目前随着深度卷积神经网络在三维点云分割中的广泛应用,许多研究机构推出了一系列公开且可靠的三维数据集,如S3DIS、Semantic3D、SUN-RGB-D、KITTI、Apollo等。在综合考虑使用场景与数据集规模后,本文选择SUN RGB-D数据集进行模型训练与测试。

SUN RGB-D是普林斯顿大学Vision & Robotics实验室开发的用于室内场景理解的数据集,该数据集共包含10 335张经过密集标注的RGB-D图像,分别由Intel RealSense、Asus Xtion、Kinect v1、Kinect v2 4种不同的3D传感器采集图像和深度信息,共有146 617个2D多边形和58 657个3D边界框,包含了场景分类、语义分割、物体检测、物体朝向预测、房间布局预测等标注[23]。

针对本文的语义分割任务,使用SUN RGB-D数据集的子集NYUdata作为实验数据。NYUdata是由Kinect v1的RGB摄像机和深度摄像机同步拍摄的室内场景的视频连续帧中提取出来的,包含了1 449组具有像素级标注的彩色和深度图像对,每张图像的宽度为561像素、高度为427像素。

由式(3)可知,在确定了相机的内参数后,即可将二维像素坐标转换为三维空间坐标。查找资料可知,Kinect相机的固有内参数如下:

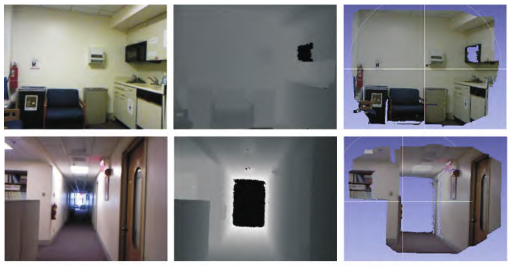
$$f_x = 525.0, f_y = 525.0$$

$$c_x = 319.5, c_y = 239.5$$

另外,Kinect在保存像素值时进行了比例缩放,比例因子factor为5 000,因此实际的Z坐标等于读取的深度值除以5 000。由于深度图像与彩色图像的对应关系,可以结合XYZ空间坐标与RGB颜色信息,得到每一组图像对所对应的RGB点云数据,其在三维处理软件MeshLab中的可视化效果如图5(c)所示:

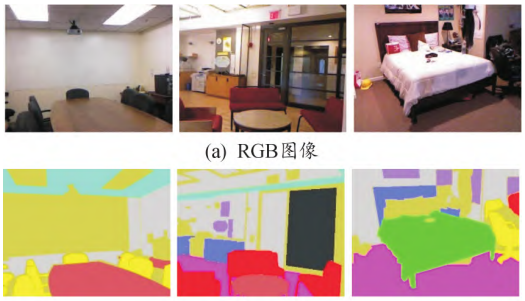
原始数据中的标签类别为894类,为便于模型训练与结果可视化,可以利用元数据中.mat格式的类别映射文件将标签类别转换为40类或13类,这里选择将894类映射到13类。RGB图像(a)与映射后的标签图像(b)如图6所示:

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net



(a) RGB图像 (b) 深度图像 (c) 点云数据

图 5 点云转换结果



(a) RGB图像

(b) 标签图像

图 6 标签类别映射

4.2 模型压缩

4.2.1 改进策略

为了使 PointNet 点云分割网络能够以较快速度、较高精度运行在移动端设备上，本文结合紧凑网络和参数剪枝 2 种模型压缩方法对 PointNet 进行改进。

紧凑网络当前比较成熟的做法是卷积核级别的重新设计，通常是用多个小的卷积核替代大的卷积核。本文借鉴 MobileNet [15] 等轻量级网络的设计思路，将 PointNet 中的全连接层替换为深度可分离卷积(depth-wise separable convolution)，在有效减少模型参数量的同时保持了推理的精度。深度可分离卷积的实质是分组卷积(Group Convolution) 和 1 ×1 卷积的结合，最大程度上实现了通道间的解耦。此外还使用减少网络宽度、深度的方式对 PointNet 进行了结构化的通道剪枝和层间剪枝。改进后的 PointNet 网络结构及参数设置如图 7 所示。

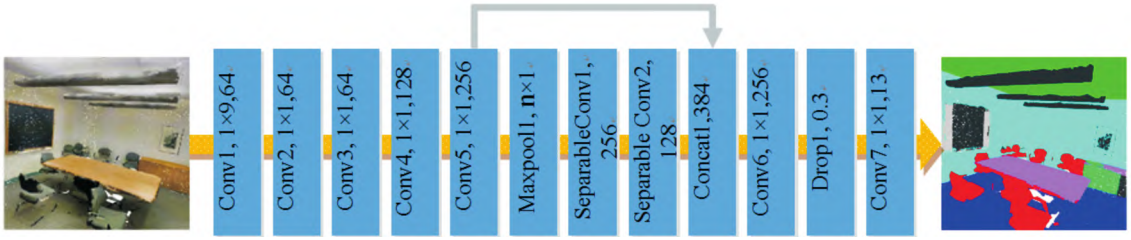


图 7 改进的 PointNet 网络

4.2.2 网络结构分析

改进后的 PointNet 网络共有 12 层，整体可分为 3 个部分。第一部分是前 5 层 Conv1 ~ Conv5，主要作用是逐级提取浅层特征，层名称后面的参数如 1 ×1、1 ×9 等表示的是卷积核尺寸，最后一个参数表示输出通道数，卷积层中的其他细节还包括高和宽 2 个方向上的滑动步长为 [1,1]，填充方式使用的是“valid”类型，激活函数为修正线性单元 ReLU，这些参数设置对于其他部分的卷积操作也是一致的; 第二部分是中间 3 层，主要作用是筛选全局特征和增加网络表达能力，该部分对于 Conv5 的输出先作一个最大池化，池化时的卷积核尺寸为 $n \times 1$，相当于每个输出通道保留一个特征，共得到 256 个全局特征，接着经过 2 个替代了全连接层的可分离卷积操作对全局特征进行非线性变换，输出 128 个特征值; 第三部分是最后 4 层，作用是对前两部分提取的浅层和全局特征进行跳跃连接并输出最终分割结果，得到每一个点属于 13 个类别中各个类别的概率。

4.3 模型训练

4.3.1 评价指标

语义分割算法的性能评价标准主要分为以下几个方面: 精确度、空间复杂度和执行时间。其中精确度的评价指标主要有总体精度(overall accuracy，OA) 、平均精度(average accuracy，AA) 、平均交并比(mean Intersection-over-Union，mIoU) 、Kappa系数等。本文中使用 OA 和 AA 作为评价指标，其定义分别如下:

$$OA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k} p_{ij}} \tag{4}$$

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

假设共有 $k+1$ 个语义类别(包括一个背景类),p_{ij}表示本属于 i 类实际预测结果为 j 类的点云数量,则 p_{ii}表示预测正确的点云数量。OA 指标反映了每一个随机样本的语义分割结果与真实标注类型的总体概率一致性。

$$AA = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k}p_{ij}} \tag{5}$$

AA 指标中 p_{ii}、p_{ij}的定义与 OA 中相同,它是总体精度的一种简单提升,反映了每个类别预测准确率的平均值,即平均类别精度。

4.3.2 训练过程

在 Windows 10 平台,编程环境选择 PyCharm,显卡设备是 NVIDIA-1080Ti,使用深度学习框架 TensorFlow 进行模型训练。网络超参数设置为批量数 batch_size = 24,初始学习率 learning_rate = 0.001,使用指数衰减法进行学习率调整,decay_step = 300 000、decay_rate = 0.7,迭代次数 max_epoch = 200。损失函数为 softmax 交叉熵,优化器选择 Adam 算法,训练集、验证集、测试集的比例分别为 60%、20%、20%。训练集(train)、验证集(val)上的总体精度分别为 0.77、0.74,验证集上的平均精度为 0.69,训练过程如图 8 所示。

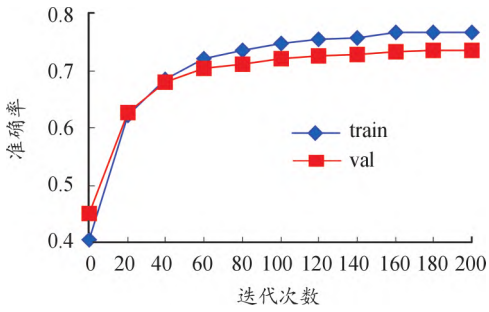


图 8 模型训练过程

4.4 移动端部署与加速

TensorFlow Lite 是专为 Android 和 iOS 等移动平台设计的深度学习解决方案,提供了转换 TensorFlow 模型并在移动端、嵌入式和物联网(IoT)设备上运行 TensorFlow 模型所需的所有工具。整个模型转换过程分为 2 个阶段,首先是对保存的 checkpoint 模型文件进行持久化操作,将其中的变量值固定,冻结为 pb 格式;再使用 TensorFlow Lite 转换器将 pb 文件转换生成最终的 tflite 文件。模型转换流程如图 9 所示。

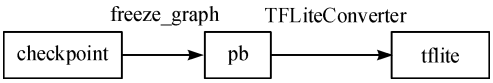


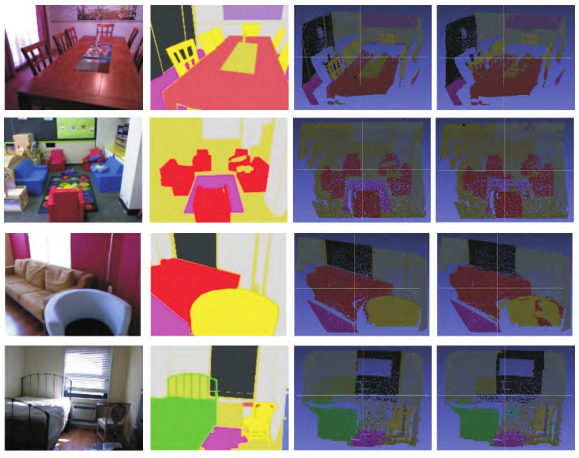
图 9 模型转换流程

模型部署的最后一步是使用 tflite 文件在安卓设备上进行推理。本文基于 Android Studio 4.0 搭建运行环境,使用 Kotlin 作为开发语言,在工程的 build.gradle 配置文件中添加 tensorflow-lite 相关依赖后就可以调用 TFLite 解释器运行输入数据并获得预测结果。

实测时使用的是华为 nova 3 智能手机,CPU 为海思麒麟 970,GPU 为 Mali G72 MP12,运行内存 6 GB,存储空间 128 GB,运行过程中发热情况一般,未启用 GPU 加速时推断一幅图像约需 1.8 s。将配置文件中的相关依赖替换为 GPU 版本后即可实现运行加速,启用 GPU 代理后的推理速度约为 0.7 s。

4.5 实验结果分析

如图 10 所示,展示了 4 个不同场景下的点云分割结果,每行代表一个场景。使用本文改进的 PointNet 网络对深度图转换的点云进行分割,取得了不错的效果,总体精度为 73.3%,对于场景中的主要类别,均能得到较好的分割结果。PointNet 网络与本文改进后的模型在参数量、精确度、推理速度方面的定量性能对比如表 3 中所示。



(a) RGB图像 (b) 标签图像 (c) ground truth (d) 预测结果

图 10 点云分割结果

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

表 3 实验结果对比

模型	参数量/M	OA/%	AA/%	推理速度/ms
PointNet	3.52	76.0	71.1	84.6
本文方法	0.76	73.3	67.6	23.5

从模型压缩的角度看,参数量减少了约 80%,总体精度仅降低了 2.7%,网络轻量化的同时也保持了较高的分割精度,模型训练和推断的速度进一步加快,证明了本文提出的改进算法的有效性;另外,从移动端部署的情况来看,将 checkpoint 模型文件大小从 13.4 MB 减少到了 2.9 MB,经过参数量化并转为移动端的 tflite 文件后仅有 268 kB;在启用 GPU 加速后,模型在华为智能手机上能够以准实时级速度进行推断,可以满足一般的即时查看需求。

5 结论

通过对深度图获取原理及点云分割方法的研究,提出了一种仅凭借移动端设备实现三维数据采集和准实时语义分割的技术方案。利用谷歌 AR Core SDK,可以在 Android、iOS 等设备上很方便地获取 16 位深度图像,在相机内参已知的情况下可以转换生成密集点云数据。考虑到移动端设备的性能受限情况,对 PointNet 网络进行改进,将模型参数量压缩为原来的 1/5,再利用 TensorFlow Lite 将 checkpoint 模型转换为可以运行在安卓手机上的 tflite 模型。实验结果表明,在启用 GPU 加速的情况下,部署的模型能以较高的精度和速度进行点云分割,证明了本文方法的可行性。

参考文献:

[1] SU H,MAJI S,KALOGERAKIS E,et al. Multi-view convolutional neural networks for 3D shape recognition [C] //Proceedings of the IEEE international conference on computer vision. 2015: 945 - 953.

[2] FENG Y,ZHANG Z,ZHAO X,et al. GVCNN: Group-view convolutional neural networks for 3d shape recognition [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 264 - 272.

[3] ZENG A,YU K T,SONG S,et al. Multi-view self-supervised deep learning for 6d pose estimation in the amazon picking challenge [C] //2017 IEEE international conference on robotics and automation (ICRA). IEEE,2017: 1386 - 1383.

[4] LONG J,SHELHAMER E,DARRELL T. Fully convolutional networks for semantic segmentation [C] //Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431 - 3440.

[5] WU B,WAN A,YUE X,et al. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud [C] //2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE,2018: 1887 - 1893.

[6] IANDOLA F N,HAN S,MOSKEWICZ M W,et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size [J]. arXiv preprint arXiv: 1602.07360,2016.

[7] MATURANA D,SCHERER S. Voxnet: A 3d convolutional neural network for real-time object recognition [C] //2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE,2015: 922 - 928.

[8] TCHAPMI L,CHOY C,ARMENI I,et al. Segcloud: Semantic segmentation of 3D point clouds [C] //2017 International Conference on 3D Vision (3DV). IEEE,2017: 537 - 547.

[9] RIEGLER G,OSMAN ULUSOY A,GEIGER A. Octnet: Learning deep 3d representations at high resolutions [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 3577 - 3586.

[10] KLOKOV R,LEMPITSKY V. Escape from cells: Deep kd-networks for the recognition of 3d point cloud models [C] //Proceedings of the IEEE International Conference on Computer Vision. 2017: 863 - 872.

[11] QI C R,SU H,MO K,et al. Pointnet: Deep learning on point sets for 3d classification and segmentation [C] //Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 652 - 660.

[12] QI C R,SU H,MO K,et al. Pointnet ++: Deep hierarchical feature learning on point sets in a metric space [J]. arXiv preprint arXiv: 1706.02413,2017.

[13] VALENTIN J,KOWDLE A,BARRON J T,et al. Depth from motion for smartphone AR [J]. ACM Transactions on Graphics (ToG),2018,37(6): 1 - 19.

[14] YANG X,ZHOU L,JIANG H,et al. Mobile3DRecon: Real-time monocular 3D reconstruction on a mobile phone [J]. IEEE Transactions on Visualization and Computer Graphics,2020,26(12): 3446 - 3456.

[15] HOWARD A G,ZHU M,CHEN B,et al. Mobilenets: Effi-

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

cient convolutional neural networks for mobile vision applications [J]. arXiv Preprint arXiv:1704.04861,2017.

[16] ZHANG X,ZHOU X,LIN M,et al. Shufflenet: An extremely efficient convolutional neural network for mobile devices [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018:6848-6856.

[17] CHOLLET F. Xception: Deep learning with depthwise separable convolutions [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017:1251-1258.

[18] 景庄伟,管海燕,臧玉府,等. 基于深度学习的点云语义分割研究综述 [J]. 计算机科学与探索,2020,15(1):1-26.

[19] RONNEBERGER O,FISCHER P,BROX T. U-net: Convolutional networks for biomedical image segmentation [C]//International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, Cham,2015:234-241.

[20] BADRINARAYANAN V,KENDALL A,CIPOLLA R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation [J]. IEEE transactions on pattern analysis and machine intelligence,2017,39(12):2481-2495.

[21] CHEN L C,PAPANDREOU G,KOKKINOS I,et al. Deep-lab: Semantic image segmentation with deep convolutional nets,atrous convolution,and fully connected crfs [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2017,40(4):834-848.

[22] 高晗,田育龙,许封元,等. 深度学习模型压缩与加速综述 [J]. 软件学报,2021,32(1):68-92.

[23] SONG S,LICHTENBERG S P,XIAO J. Sun RGB-D: A rgb-d scene understanding benchmark suite [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015:567-576.

Research on point cloud segmentation method of mobile devices based on RGB-D image

YU Fangjie[1,2], WANG Bin[1]

(1. Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China;
2. University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: In recent years, the popularity of depth sensors and 3D laser scanners has promoted the rapid development of 3D point cloud processing methods. Aiming at the traditional point cloud segmentation mode with front-end and back-end separation, an integrated technical solution for 3D data collection and processing using mobile devices is proposed. Based on Google's AR Core development platform, the depth map acquisition experiment on Android devices is carried out, the depth map can be further converted into point cloud data; through the research on the light-weight method of the model, the PointNet network is improved, the model parameters are reduced to 1/5 of the original, while it had a segmentation accuracy of about 73%. Finally, using the TensorFlow Lite mobile terminal deep learning framework, the improved PointNet network is successfully deployed on Android smartphone, and the quantized tflite model is only 268 kB in size. After enabled GPU acceleration, the inference speed of single scene point cloud data is about 0.7 s. The experimental results show the feasibility of the proposed method.

Key words: point cloud segmentation; the depth map acquisition; deep learning framework

(责任编辑 王 欢)

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net