A Deep Neural Network for Learning Hierarchical Multi-scale Point-Cluster Features Towards Large-Scale Point Cloud Classification

***Abstract***—室外大场景激光雷达点云的自动精确分类是遥感和计算机视觉等领域的难题。这主要是因为当前方法对点云数据的特征提取能力不够。本文利用了深度学习强大的特征表达能力提高分类精度，同时避免点云栅格化带来精度损失。本文提出了一种直接用于原始激光雷达点云分类的深度学习方法，该方法构建了一种具有特征结构性和空间结构性，且能进行多层次点云特征提取和分类的深度神经网络模型。该深度神经网络模型分别处理点云中不同类别的特征再进行特征融合，有效挖掘点特征的结构性；分开处理点云物体的边界点和主体部分点，并且设计了基于空间关系的pooling让神经网络自主决定边界点和主体部分点的贡献，使得点集特征既能突出地物表面主要结构，又能突出地物具有区分能的边界特征；把点集中所有点的特征整合成点集特征，通过同时学习所有多层次点集公共特征，但分开对不同层次进行分类的方法，既提高特征鲁棒性，又提高分类精度，最后联合各层次的分类结果，进行分类。该方法对复杂室外场景点云具有很好的性能，无需对点云进行栅格化，能对任意大小形状的点云进行分类。

***Index Terms***—Point cloud classification, deep neural network, point-cluster features.

**I. INTRODUCTION**

室外大场景激光雷达点云的自动精确分类是遥感和计算机视觉等领域难题，而特征提取优劣决定了分类结果好坏。当前已经有大量特征算子被用于点云分类并取得良好的分类效果，如spin图[1]、曲率、法向量等。而以这些特征算子为基础，进行进一步特征提取，能获取更有效的特征，进一步提高分类精度。当前特征提取方法有bag of words (BoW)，稀疏性约束等方法，这些特征提取方法能被看作单层神经网络，而这种单层结构限制了特征的表达能力。基于深度学习建立的多层神经网络结构，能学习到相对单层结构更好的特征表达形式。

深度学习用于点云分类的研究还较少，主要原因是点云数据缺乏组织性，无法像图像一样通过窗口移动在图像中获取大小相同且具有明确相邻关系的图像块进行训练。正因为此，当前基于深度学习进行点云分类方法先栅格化点云或三维目标，再进行下一步处理。但栅格化会带来三维信息丢失，并且栅格大小也会影响最后分类精度，尤其是在一个点云密度变化较大的场景中栅格大小无法很好的确定。

本文提出了一种具有特征结构性和空间结构性的深度学习网络，用于多层次点集特征学习，以实现大场景地物点云分类。通过分割原始无序的点云形成多层次点集，基于点的特征，学习点集特征并进行分类，避免栅格化带来精度损失的同时，又能利用深度学习强大的特征表达能力提高分类精度。虽然点云中点特征能直接利用深度学习进行特征提取，但是无法充分利用点与点之间的关系，且在大量点云中进行逐点分类效率很低。本文的深度神经网络也能进行基于点的分类，只需把每一点集看作该点一定邻域内点的集合即可。本文的主要贡献是：

*i*) 提出一种能够直接用于原始点云的多层次神经网络模型。该模型相比于其他处理点云的神经网络，无需对点云进行栅格化，该模型以每一个点集中所有点的特征作为一个样本输入，能够直接对原始点云进行分类，避免栅格化带来的信息损失。多层次分类能有效提高分类精度，但是点云中点密度变化大，不同层次的点云特征可能存在较大不同，所以类似迁移的思想，该模型学习所有层次的公共特征，保证特征鲁棒性，但同时为提高各层次分类精度，利用这个公共特征独立对每个层次点云进行分类，最后再联合各层次分类结果进行分类，而不是类似其他多层次模型在每个层次独立进行特征提取，再把各层次特征联合起来进行分类。

*ii*) 提出了点特征权值共享，但是不同点特征和不同点类别采用不同权值分开处理的策略。点特征权值共享保证了深度神经网络模型能够对原始点云分类，同时点云中点特征由多个特征算子构成，各算子提取特征的方式和目的不相同，点云中边界点和主体部分点也有各自的作用，这些都分开处理有利于提高模型的识别能力。

*iii*) 提出了基于DMST的pooling策略，有效挖掘了点云的空间结构性，描述了点云的邻域关系。这种策略通过pooling连接边界点和主体部分点的信息，使得边界点和主体部分点在神经网络中虽然有各自不同的权重，但是同时又能够把各自的信息相互传递，这样兼顾点云物体的边界点和主体点对分类的贡献，既保证了边界点，边界点往往也可能是特征点，能有效的区分相似的物体；又保证了主体点能有效区分不太相似的物体，使得整个分类不会受到噪声的影响。

**II. RELATED WORK**

**A. 点云分类**

在过去数年中点云分类得到广泛研究，大量特征提取以及分类方法被提出，主要采用特征算子或语义关系结合分类器进行分类。首先基于点的方法被提出，这些方法采用大量特征算子的组合进行分类。Chehata et al. [3]用随机森林对5类总计21种机载激光雷达点云特征进行分类，并通过迭代的特征选择获得6种最好的特征。Guo et al. [4]采用了26种特征以及JointBoost在复杂地面上对建筑，植被，地面，电线以及架线塔五种地物进行分类。Kragh et al. [5]采用13种特征以及svm对车载点云进行分类，并针对不同车载点云中点密度变化采用变邻域半径的方法提取特征。Brodu et al. [6]提出了一种多尺度特征提取方法描述地物特征，并对两个区域内植被，岩石，水和地面进行分类并取得很好的精度。其次一些基于点集的方法获得了研究，这些方法利用了点集特征以及点集之间的关系进行分类。Zhang et al. [7]对机载点云采用区域生长进行分割，然后提取分割后点集的几何，回波，辐射度以及拓扑特征，并用svm进行分类。Pu et al. [8]对车载激光雷达点云过分割后，通过位置，形状，方向等因素设计特征识别平面，柱状结构。Yang et al. [9]把车载激光雷达点云过分割成语义的超像元，然后建立了一系列规则对这些超像元进行合并，并对合并后物体设计规则进行识别。Fukano et al. [10]通过拉普拉斯收缩突出线状和面状特征，并根据线状和面状特征组合方式提取杆状地物。还有一些方法通过图像赋予点云一定空间语义关系进行分类。Zhang et al. [11]通过道路场景中地物在图像和点云中的一系列弱判别条件，然后通过一种过滤算法以及条件随机场的语义分析方法完成点云分类。

**B. 特征表达**

在直接使用特征算子或语义关系的基础上，分类过程中如果通过特征表达对已有特征进行重新组织和提取，能获得更优秀的特征，以此分类能够获得更好的分类精度。

字典学习在特征表达中获得了广泛应用。BoW[12]，Plsa[13]，Lda[14]和稀疏编码都是常用的字典学习方法。Yang *et al*. [15]提出了一种sparse coding spatial pyramid matching (ScSPM)的方法，结果显示采用稀疏编码以及max pooling要比矢量量化和averaging pooling识别效果更好。Yang et al.[16]提出了一种结构性的有监督的字典学习方法Fisher discrimination dictionary learning (FDDL) model，这种方法获得的字典具有类内差异小类间差异大的特点。在点云分类中，字典学习方法也得到了应用，Wang et al.[17]和Zhang et al.[18]分别采用了LDA和latent Dirichlet allocation and sparse coding(SCLDA)的方法进行了字典学习并进行分类。

流形学习方法在特征表达中也得到了广泛的应用，流形学习方法假设高维空间中存在低维流形，而通过寻找相应的嵌入映射完成特征提取。比如 ISOMAP [19]，KPCA[20]，LE [21]等。Weston et al.[22]采用了深度学习的框架进行了半监督的内嵌。

Bengio et al.[23]在综述中回顾了深度学习和稀疏编码，流形学习等表达方式之间的联系，并用深度学习的角度对这些表达方式进行了解释。Krizhevsky et al.[24]提出了一种7个特征提取层的卷积神经网络能够有效提取高层的视觉信息。Girshick et al.[25]通过high-capacity的卷积神经网络进行由下而上的特征提取并采用domain-specific fine-tuning，相对于VOC 2012提高了30%的平均精度。这些神经网络以及训练方法的提出，显示了深度学习强大的特征表达能力。

**C. 基于的深度学习的3D物体识别**

近年来深度学习在3D物体识别和检索中也得到了应用。Wu et al. [26]通过设置模型内部栅格值为0，外部值为1的方式栅格化3D模型，并设计了Convolutional Deep Belief Network (CDBN)分类栅格化后的3D模型。Zhu et al.[27]利用3d模型不同角度的深度图像，采用DBN预训练，自动编码机最后微调得到3d模型特征。Xie et al. [28]提取3D物体多尺度的形状表达，提出了一种用Fisher discrimination criterion约束隐层的自动编码机，提取了高层次的形状特征。Socher et al. [29]采用convolutional and recursive neural networks对RGB-D 图像进行目标识别。点云分类中，也有少量利用深度学习分类的研究。Guan et al. [30] 采用DBM对分割出的每一棵树点云的垂直廓线进行分类，分类了10个树种。Maturana et al. [31]采用体密度图的方式对机载点云进行栅格化，设计了3维CNN学习分类机载点云。

综上，深度学习进行3D物体识别和点云分类时都需要把3D物体进行栅格化，通过栅格化给予3D物体和点云空间结构性，获得规则的输入以及相邻关系，而我们的深度神经网络结构通过所有点特征的权值共享以及DMst建立点集内部结构，赋予了原始点云规则性以及相邻关系。

**III. 深度神经网络结构**

本文提出的深度神经网络模型总体结构如图1所示，该图是一个假设只有1个Net1和1个Net2，每个点特征都只包括两个不同类型的特征描述子，整个点集被分成三个层次的结构图，具体Net1,Net2以及采用的描述子多少和层次的多少可以根据实际需要改变，本文只是提供一个深度神经网络的基本框架。该网络包含一个输入层，每一个输入样本为一个点集中所有点的特征，然后连接多个Net1层，Net1层对每个点的各类不同特征分开处理，采用的特征类别比如基于特征值的特征，基于spin图的特征， Net1输出的每个点个类别特征合并后连接Net2层，在每个Net1和Net2后面紧跟着一个Net3层，Net3层是基于DMST结构的池化层进行基于空间结构信息的池化。在最后一个Net3层后连接一个Net4层对整个点集中每个点特征进行池化产生点集特征，接着连接一个Net5层进行多层次点集联合分类，以及最后输出层输出每个点集的类别。图1中彩色方框表示各个不同的功能层，每一个黑色圆圈表示一个特征向量，在Net2左边的黑色方框表示一个点的特征的整体，每个黑色方框包含两个特征向量表示一个点的特征中包含两类不同的点特征。

Net4平均池化

Net5多层次判别

基于DMST结构的池化

点云的各特征分开处理

合并

基于DMST结构的池化

Net1

Net2

softmax

输入

输出类别

综合处理各特征

Net3

Net3

图1. 神经网络总体结构图。

表1 整个神经网络的流程

|  |
| --- |
| 假设，只经过1个Net1和1个Net2，每个点特征都只包括两个不同类型的特征描述子，整个点集被分成三个层次。  输入: 一个点集的所有点特征，X1，这个点集对应的DMst结构  1，按照DMst结构，把点集中的点分成边界点X11和主体部分点X12  2，再按照每一个点特征的各特征算子分成X11=[X11,1, X11,2 ]，X12=[X22,1, X22,2]  3, 通过Net1 ， (1)  P用来表达这是边界点还是主体部分点，f表示是第几个特征描述子  4， 通过Net3  (2)  i表示点集中第i个点，表示在DMst上和i相连的点的集合，因为和i相连的点既有有边界点和主体部分点，所以p是由和i相连的点决定  5，把各特征描述子经过神经网络的特征合并起来准备输入Net2，  6，通过Net2  (3)  7，通过Net3  (4)  8，通过Net4  (5)  9，通过Net5  (6)  h表示点集属于第h个层次，  10，通过softmax  (7)  11，loss函数  (8)  表示真实的标签  \* 当Net3需要反向求导时，以公式5为例，  (9)  1[.]表示0/1的指示器。 |

**A. 点集中所有点的特征综合成点集特征**

在这一章中将介绍我们的神经网络结构能够直接处理点云无需栅格化的基本组成结构以及设计原因。如果直接把每个点的特征作为输入，这样能够直接利用深度神经网络进行特征提取，但是直接针对点的特征提取无法利用空间信息，分类精度始终无法进一步提高，所以除了栅格化以外一种基本的想法是首先把点云分割成点集，然后把点特征综合成点集特征。但是点集大小不一，难以用一个固定长度的向量表示点集中每个点的特征连接起来形成的特征向量，并且点集中点排列顺序是随机的，点排列顺序变化会改变输入的特征向量，会造成同一个点集，不同顺序的点排列形成不同的分类结果，所以采用对点集中所有点的特征进行池化，避免了点集中点的数量和顺序的影响。综上所述输入层输入的每一个样本是每一个点集中所有点的特征，经过Net4前各层对点进行权值共享的特征提取，再经过Net4的平均池化形成了每一个点集的点集特征，这构成了能够直接处理点云的基本框架。

输入层输入通过下述步骤得到。类似于文献[18]，首先去除输入点云中的地面点，对非地面点构建多层次点集。接着，提取点集中点的特征，即，每个点的特征由三个不同邻域范围内的基于特征值的特征和spin图特征构成，每一点在一个邻域内有7维基于特征值的特征和12维spin图特征，共19维特征，因此，三个不同的邻域范围共形成57维特征。每一点集中所有点特征合并起来形成57×n维的特征矩阵，n是点集中点数量。归一化所有点集特征矩阵的每一维后，每一个点集的特征矩阵对应输入层的一个输入。

经过最后一个Net3后所有单点特征通过Net4平均池化层后，得到与Net3输出的点特征同维的向量，作为点集特征。公式6显示了Net3中平均池化过程。

**B. 特征结构性的提取**

基于特征值的特征和spin图特征从不同角度对点集中的点进行描述，特征内部的关联性比不同特征间的关联性要密切。为了充分挖掘这两种特征，用两个独立的网络来分别处理这两种特征，表1第2，3步表示了这个过程。通过Net1的过程如公式1所示。

为了学习基于特征值的特征和spin图特征之间的关系，在Net2中，对Net1输出合并形成的向量进行处理，表1第5，6步表示了这个过程。通过Net2的过程如公式3所示。

该深度神经模型充分了考虑各种特征自身特点以及特征之间的关系，通过Net1和Net2实现对不同特征内部以及特征之间的表达。

**C. 空间结构性的提取**

深度神经网络模型通过Net3考虑点与点间空间结构性。良好的点集特征能很好的描述点集内点与点之间的空间和拓扑关系，但点云本身缺乏组织性，点与点之间没有明确的临近关系，需要根据点云中点空间分布的特点建立深度神经网络模型。

点云中存在较多噪声点和地物边界点，这类点分布比较散乱，且点特征不稳健，不能充分体现点集所表示地物的特点，因此需要减少这些点对整个点集特征的影响，突出点集中主体部分点的特征，主体部分点通常分布在物体表面，分布相对比较整齐和密集。但是同时当两个类别具有较大相似度时主体部分点将无法有效分类这两个类别，需要更加显著的特征点进行区分，而特征点往往是边界点。综上，对一个点集进行分类时，需要兼顾主体部分点和边界点，所以首先需要粗略的区分主体部分点和边界点，然后在神经网络中采用不同的权值进行特征提取，最后有效的把主体部分的特征和边界点的特征的信息进行融合，让神经网络自动的确定边界点的特征和主体部分特征的贡献。

本文通过the distance minimum spanning tree (DMst)算法[2]组织点云中点的结构，这种方法能够有效的区分边界点和主体部分点。由于DMst结合了MST算法，使得点集中主体部分点位于树主干而散乱的点位于叶子节点。DMst还结合了Dijkstra算法，以点集中心点作为根节点后，边界点更容易成为叶子节点。叶子节点指在DMst上只和一个其他节点相连的节点。值得注意的是当根节点也是只有一个节点和它相连时，它也作为叶子节点。

区分出边界点和主体部分点以后， 我们将在Net3进行基于DMst的池化，在每执行一次Net1或Net2以后都会紧接着一个Net3。表1第1，4，7步表示了这个过程。

具体的，对每一个节点和其所有相连接节点的特征进行max pooling，然后，把max pooling后的特征作为这个节点特征。采用max pooling是为了最大化的保留有可能对分类有帮助的信息。公式2和4表示了pooling过程。这样虽然边界点和叶子节点是采用的不同的权值，但是因为叶子节点和非叶子节点相互连接，随着pooling次数的增加，边界点和主体点的信息能够有效的在神经网络中传递。同时我们也能从正向公式2,4，以及反向公式9中发现，边界点和主体点以及和他们相关的权重经过pooling混合到了一起，这样神经网络能够自主的决定这两部分各自的比重。

图2显示了一个2维点集DMst的池化的过程。图2 (a) 显示了DMst中各节点，图2 (b) 显示了网络的连接方式。

1

2

4

3

5

1

2

3

4

5

(a) (b)

图2.剪枝过程。图2 (a) 显示了DMst中各节点，图2 (b) 显示了网络的连接方式。

**D. 多层次分类**

采用单一层次的点集难以获得良好的分类效果，进行多层次分类能有效提高分类精度及鲁棒性[17]，所以，深度神经网络模型采用多层次分类的结构。但是因为点云分布不均匀，尤其地面激光雷达的点云变化极大，所以和其他多层次模型在每个层次独立进行特征提取，再把各层次特征联合起来进行分类的过程不同，我们采用类似于迁移的思想，让其他不同层次的点集信息促进本层次点集分类效果，确保即使存在缺失或者点密度变化较大时点云分布情况也能被不同层次点集模拟到，所以所有的层次的点集都只学习一个共同的特征。这个过程通过Net5实现。不同层次的点集通过Net5前所有层都是采用的相同的权值，强制不同的层次采用相同的特征提取方式，保证特征的鲁棒性。但是同时为了提高了不同层次点集的判别能力，Net5又分开判别每个层次。具体的，多层次点集输入深度神经网络时，同时标记每一个点集属于哪个层次，经Net4变成点集特征后，在Net5中对不同层次的点集各连接一层独立的全连接网络，独立进行分类。如果一个点集属于第一个层次，就通过图1中Net4部分斜向上的全连接网络，如果属于第二个层次，那么通过图1中Net4部分中间的全连接网络，如果属于第三个层次，那么通过图1中Net4部分斜向下的全连接网络。每个层次的点集在Net5中都只连接自己对应的那个全连接网络，并连接对应的最后softmax层分类，而不连接其他两个全连接网络。整个Net5以及softmax过程由表1的9,10步以及公式6,7表示。最后loss函数采用交叉熵loss函数如公式8表示。

最后，对最细层次的点集，也就是，点集尺寸最小层次的点集进行分类，其分类结果为包含这个点集的多个层次点集概率的连乘，以概率最大的那一个类别作为分类结果，完成整个点云的分类。

**E. 实施细节**

整个网络中采用激活函数为min(5，elu(x))，设置5为上限是为了防止过大的神经元导致整个网络出错，并采用文献[32]的方法初始化。并在各层激活函数以后采用块归一化[33]。我们采用了最小块大小为148的随机梯度下降进行训练。网络的学习率为0.1，采用了0.5的动量。Net1，Net2，Net5都采用了10个神经元。Net1，Net2的层数详细见实验部分。

**III. EXPERIMENTAL RESULTS**

为了验证本文方法在室外场景点云分类的性能，我们用机载激光雷达和地面激光雷达点云对本文提出的深度神经网络进行了测试。

1. **Dataset**

本文采用三幅场景点云数据对本文的方法进行了验证。

场景一（Scene T）：天津市区局部区域的机载激光雷达点云，主要由建筑和植被构成，平均点密度是20–30 points/m2，屋顶外侧存在明显向外拓展的屋檐，屋檐处存在大量因散射而形成的噪声，且屋檐作为边界点和其他屋顶点的特征会有所不同，对分类建筑的分类精度造成影响。

场景二（Scene II）： ISPRS提供的Vaihingen数据集。由10个条带合并而成，区域点密度中值是6.7 points/m2，在条带重叠处点密度大，而在只有一个条带处平均只有4 points/m2 ，同时Vaihingen数据包含回波信息以及对应图像，由于本文专注于对点云的形状信息进行提取，所以只采用x，y，z的坐标信息。由于不同条带的重叠，点密度在整个场景不一致。其中建筑包括立面和屋顶，植被包括低植被，栅栏，灌木和树，多种类别混合在一起对分类造成影响。

Scenes I和II数据都是采用Leica ALS50 system，平均飞行高度是500m，视场角是45度，进行扫描得到的。

场景三（Scene III）：采用是瑞士理工大学（ETH）提供的地面激光雷达点云数据Semantic-8（<http://www.semantic3d.net/view_results.php?chl=2>）。场景点云密度变化较大，地物比较复杂，包括地面、高植被、低植被、建筑、hard scape、人、车。

下文中列出的测试数据分类精度采用了最后100次迭代的最佳结果。

**B．Classification results of Scenes I**

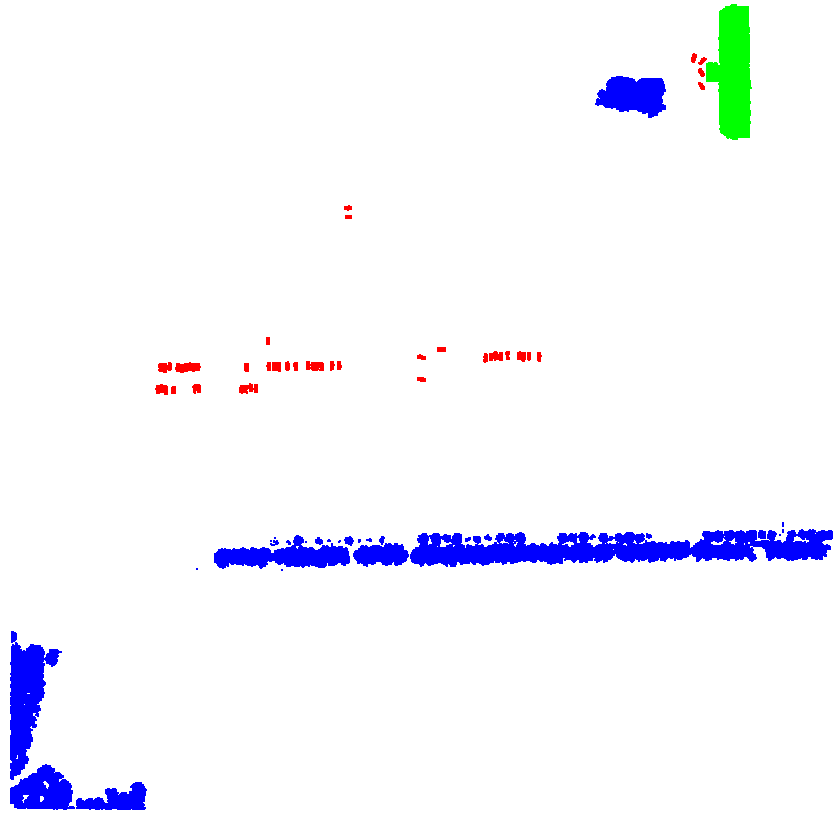
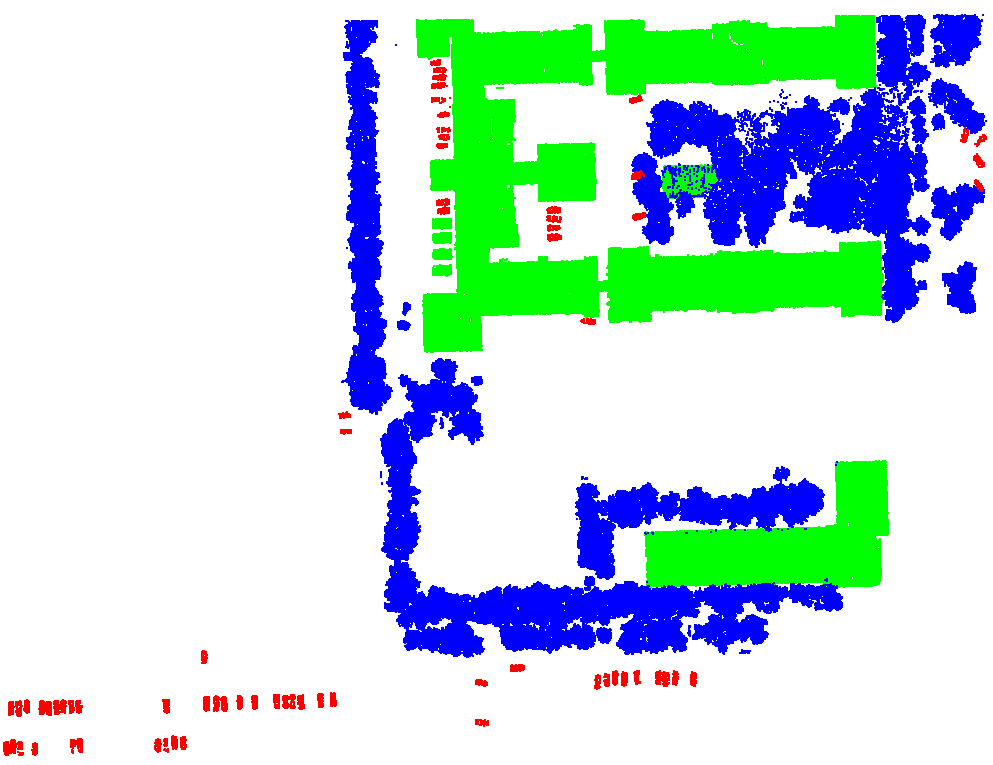
天津数据共有两个场景，采用和文献[18]相同训练和测试数据，以及相同的多层次点集和特征。训练和测试数据包含的点数量以及对应的多层次点集的总点集数如表1所示，括号中显示点集的数量。表2显示了测试时各类别分类精度。图3显示了训练样本，测试数据以及分类的结果，红色表示建筑，绿色表示植被。

表2 样本内点数量以及对应的多层次点集的总点集数

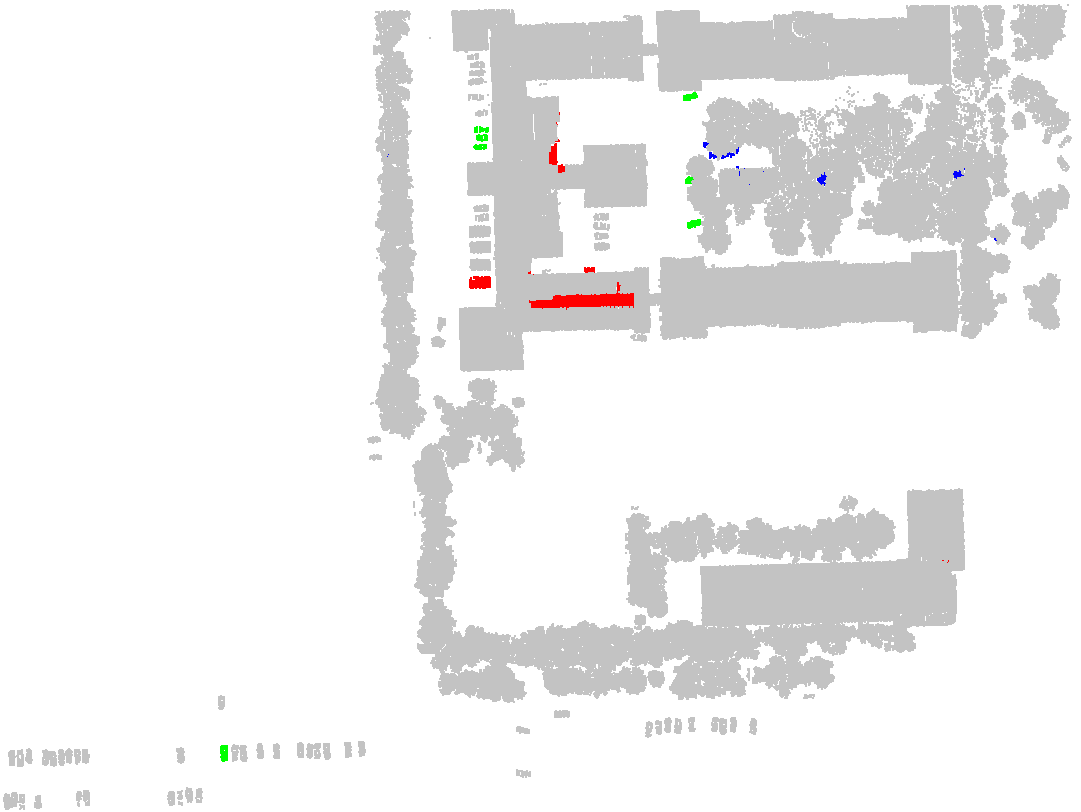
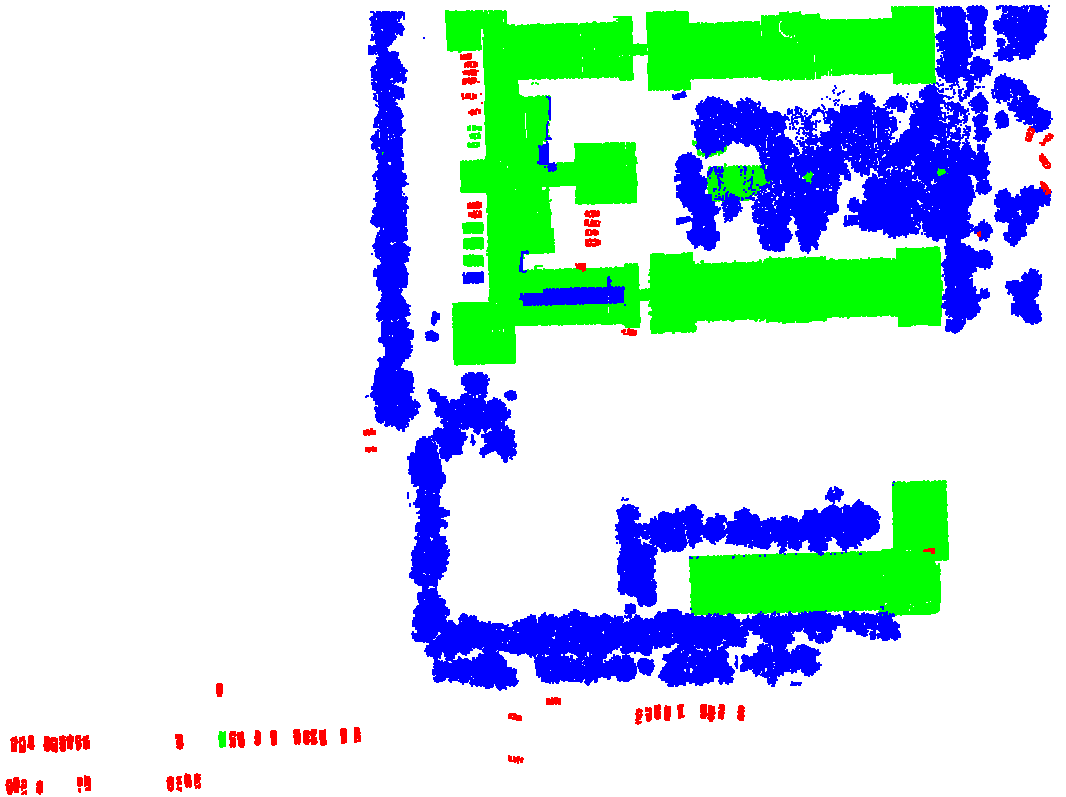
|  |  |  |  |
| --- | --- | --- | --- |
|  | Building  (场景1/场景2) | Tree  (场景1/场景2) | Car  (场景1/场景2) |
| Training data | 37847(1040)  /64952(1713) | 70540(2016)  /39743(1115) | 5410(173)  /4584(142) |
| Test data | 201674(5375)  /157447(4174) | 218110(6055)  /74264(2128) | 7987(249)  /7738(239) |

表3 分类精度

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 场景1 | Building(%) | Tree(%) | Car(%) | Accuracy(%) |
| our Method | 97.7/98.8 | 99.2/97.7 | 85.2/98.1 | 98.2 |
| Method in[17] | 94.0/95.4 | 95.0/94.3 | 79.1/60.8 | 94.5 |
| Method in[18] | 95.7/96.2 | 95.9/95.9 | 80.8/67.9 | 95.8 |
| Method in[4] | 89.7/98.1 | 97.9/89.1 | 65.2/46.6 | 92.9 |
| Method in[35] | 93.5/96.2 | 95.3/94.1 | 75.3/84.6 | 95.1 |
| 场景2 | Building(%) | Tree(%) | Car(%) | Accuracy(%) |
| our Method | 98.9/98.4 | 96.2/96.5 | 78.4/84.9 | 97.4 |
| Method in[17] | 90.3/93.9 | 97.6/96.5 | 49.4/42.0 | 94.1 |
| Method in[18] | 94.7/94.5 | 98.1/97.7 | 53.9/60.5 | 95.5 |
| Method in[4] | 86.8/91.2 | 96.8/95.5 | 44.1/34.8 | 92.2 |
| Method in[35] | 92.7/94.0 | 95.1/92.6 | 71.2/65.3 | 94.3 |

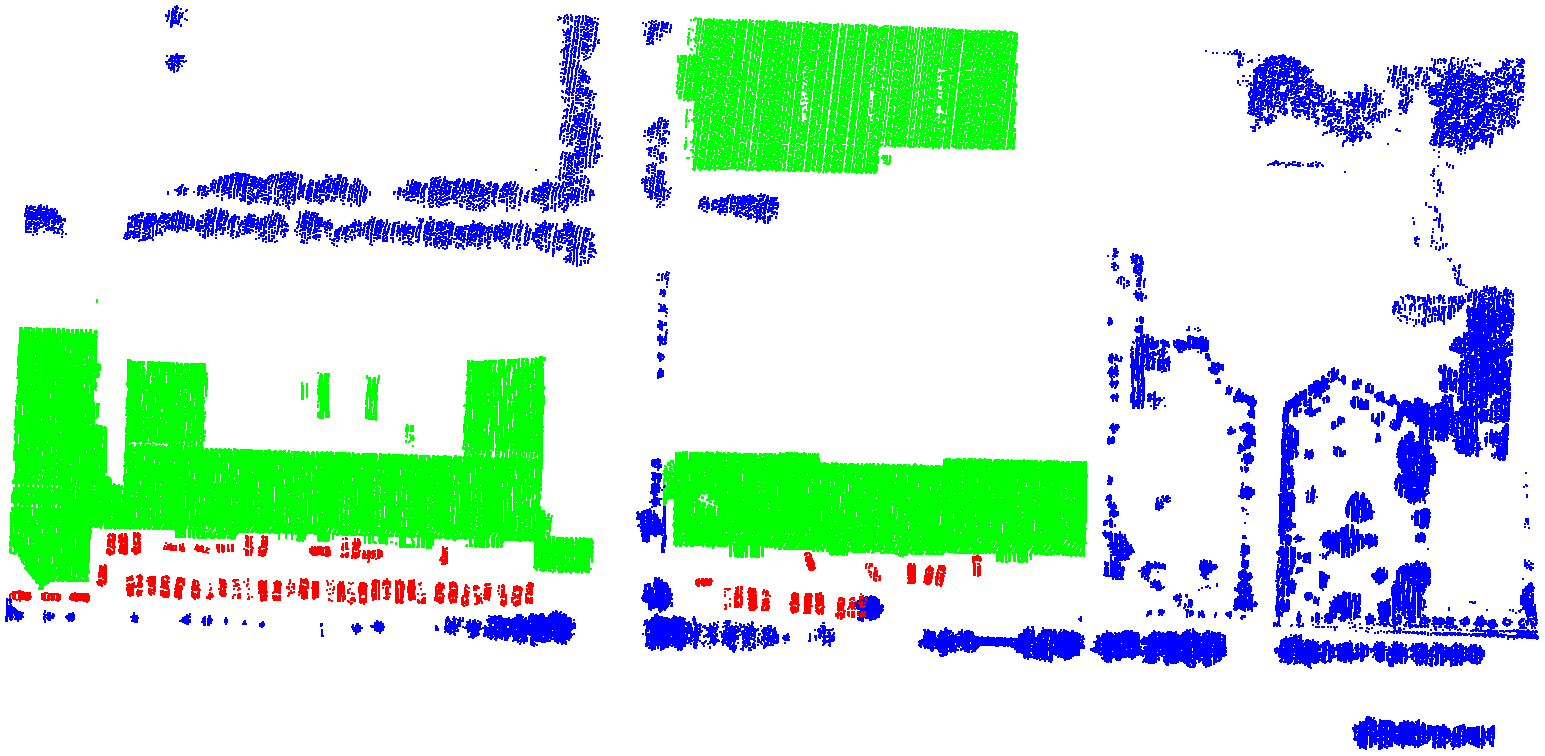
 

(a)训练样本 (b) 测试数据

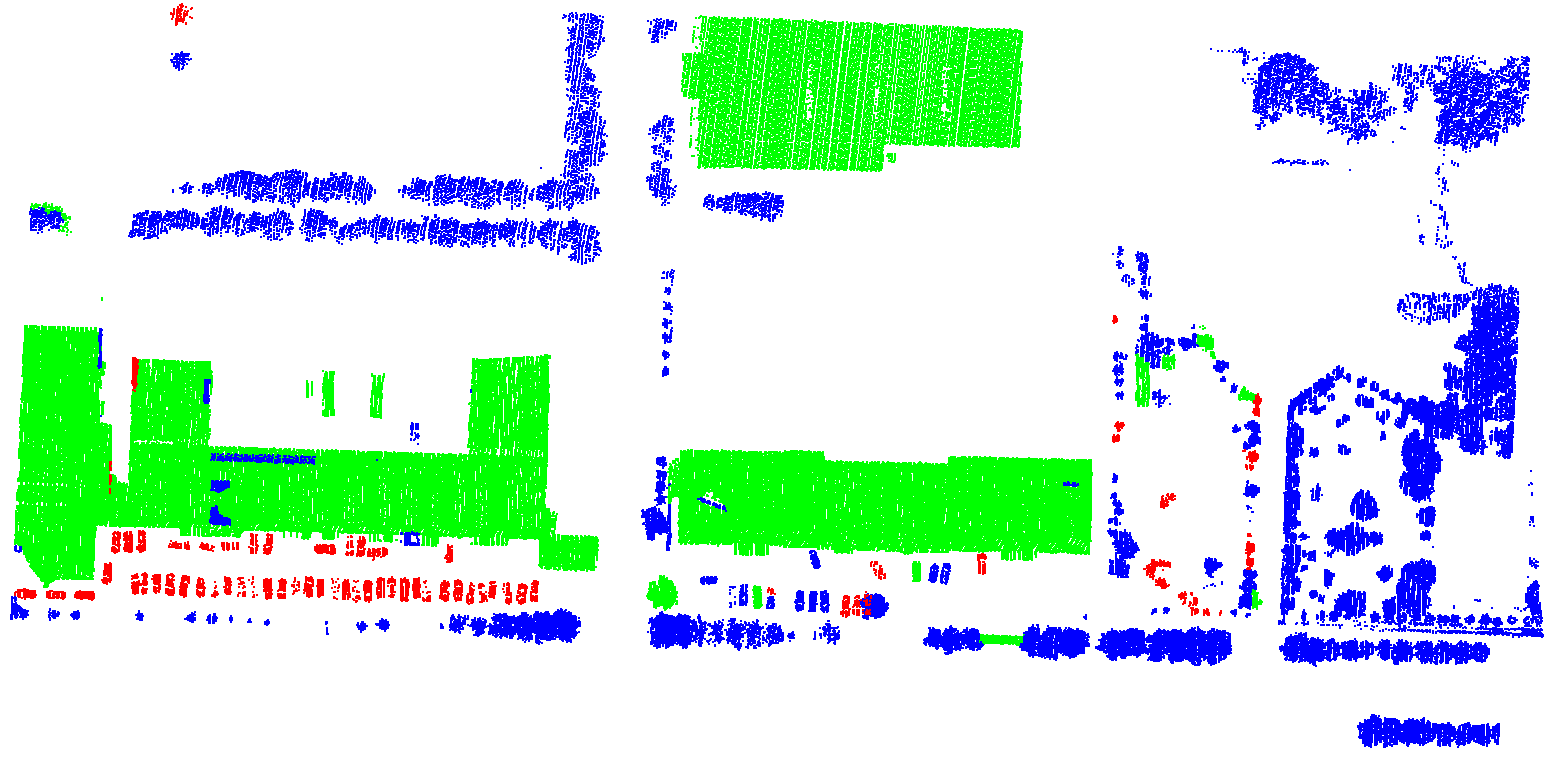
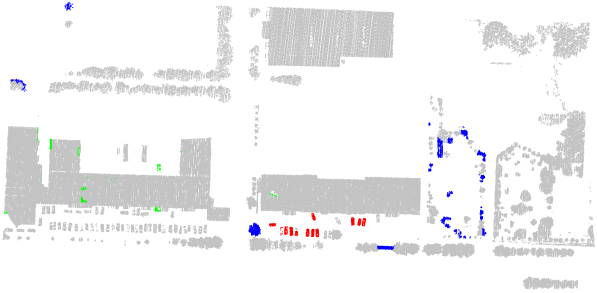


(c) 分类的结果 (d) 突出显示分类错误

图3场景1训练样本，测试数据以及分类的结果

(a)训练样本 (b) 测试数据

(c) 分类的结果 (d) 突出显示分类错误

图4场景2训练样本，测试数据以及分类的结果

**B．Classification results of Scenes II**

Scenes II范围较大，地物错综复杂。在对数据进行多层次分割的基础上，随机在最粗的那个层次的点集中挑选了70%的点集作为训练样本，剩余30%的点集作为测试样本。采用多层次分类的点集大小为60，120，240。训练样本包含的类别为屋顶，立面，低植被，树和灌木，其他类别因为点数量较少，而深度学习本身需要大量的数据，所以未进行分类。因为要分类屋顶，立面，低植被和树灌木，所以除了文献[18]的特征以外，参考文献[34]采用了每个点减去DEM的高度的特征以及每个点在三个邻域的法向量。训练样本包含的类别对应的点数量以及对应的多层次点集的总点集数如表3所示，括号中显示点集的数量。表4显示了测试时各类别的分类精度。图4显示了训练样本，测试数据以及分类的结果，其中红色为屋顶，粉色为立面，蓝色为低植被，绿色为树和灌木。为了清晰显示分类错误的点，图2(d)中灰色表示分类正确的点，彩色的点表示被分错的点。

表4 样本内点的数量以及对应的多层次点集的总点集数

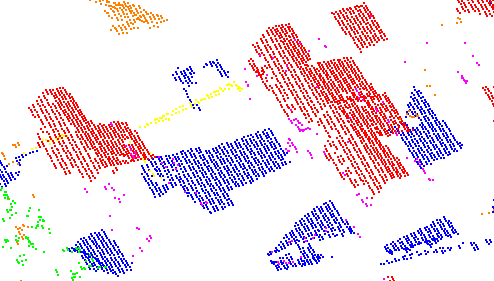
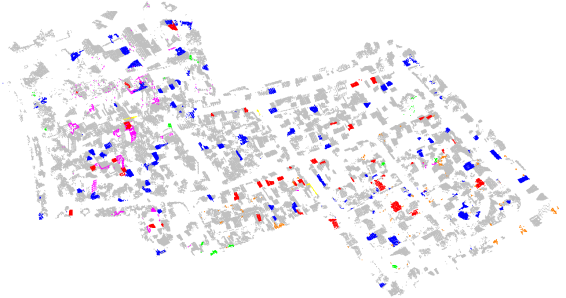
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 屋顶 | 树和灌木 | 低植被 | 立面 |
| Training data | 97631(4183) | 118526(11619) | 115484(5062) | 19566(932) |
| Test data | 61015(2303) | 61559(5873) | 50768(2815) | 8098(349) |

表5分类精度(precision/recall and Accuracy)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 屋顶(%) | 树和灌木(%) | 低植被(%) | 立面(%) | Accuracy(%) |
| Test data | 97.4/97.8 | 99.8%/97.3 | 97.0%/99.2 | 95.5/97.3 | 98.0 |
| Method in[17] | 79.1/85.5 | 89.4/86.4 | 89.6/86.3 | 53.8/59.6 | 85.1 |
| Method in[18] | 85.9/87.7 | 89.9/89.2 | 89.1/87.0 | 49.8/52.7 | 86.5 |
| Method in[4] | 87.5/81.8 | 81.9/85.2 | 79.3/82.8 | 37.4/35.9 | 81.1 |
| Method in[35] | 82.9/89.6 | 86.6/86.2 | 91.0/85.1 | 72.0/65.9 | 85.9 |



(a)训练样本 (b) 测试数据



(c) 分类的结果 (d) 低植被和屋顶的点云

图4训练样本，测试数据以及分类的结果

从上述不同分类方法比较中，可以发现我们的方法分类精度很好，几乎完全能够区分整个数据，说明深度神经网络结构能够很好的提取地物特征并进行分类。在场景1和场景2中，存在大量屋檐的结构，但是，只有图3和图4蓝色方框中存在一些建筑被误分成植被，其他部分几乎完全正确，而错分部分图3中主要是因为屋檐部分噪声很大，屋檐上有大量噪声点，而图4错分部分为独立的脱离屋顶的屋檐外延的点云，几乎形成一条孤立的线，丧失了建筑的形态。汽车也基本被识别出来了但识别率不是特别高，这是因为场景中汽车点云数量相对建筑和树的数量较少，且单个汽车本身包含点云较少，造成特征不明确，需要较多的样本才能发挥深度神经网络的特点。在图5中可以看到即使地物错综复杂，但是绝大部分点云都分类正确，只有一些零星的地物边角存在错分的现象。在这个场景中建筑立面的点数量相对其他3类也较少，但是可能因为建筑立面存在较为明确的特征足以区分它和其他地物，保证里它较高的分类精度。

同时把我们的方法和其他的方法进行比较，结果显示我们的方法要优于其他的方法，这显示了采用深度神经网络进行特征提取和分类的能力。场景1和2中，这几种方法对建筑和植被的分类精度都比较好，但是我们的方法显示了更加好的精度。并且令人惊讶的是采用了深度神经网络后，对汽车的分类精度也得到了一定的提高，这也许是因为建筑，植被以及汽车具有一定的共有特征，而这种共有特征被深度神经网络所提取，最后对汽车的分类有帮助。在场景3中其他方法中有大量建筑立面和屋顶被误分成其他类别，这主要是因为屋顶和低植被相似度较高，它们都相对比较平整，而立面点云虽然分布在平面上，但是靠近地面部分呈现散点状分布，有些会被分成植被，但是采用我们的方法以后这四类的分类精度都达到很高的精度。

**C．Classification results of Scenes III**

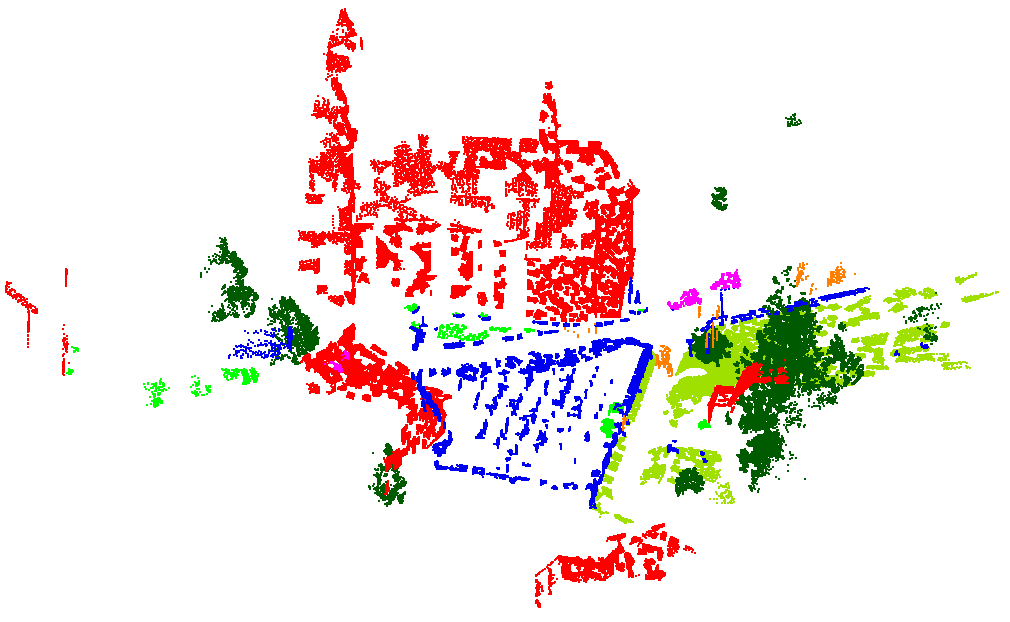
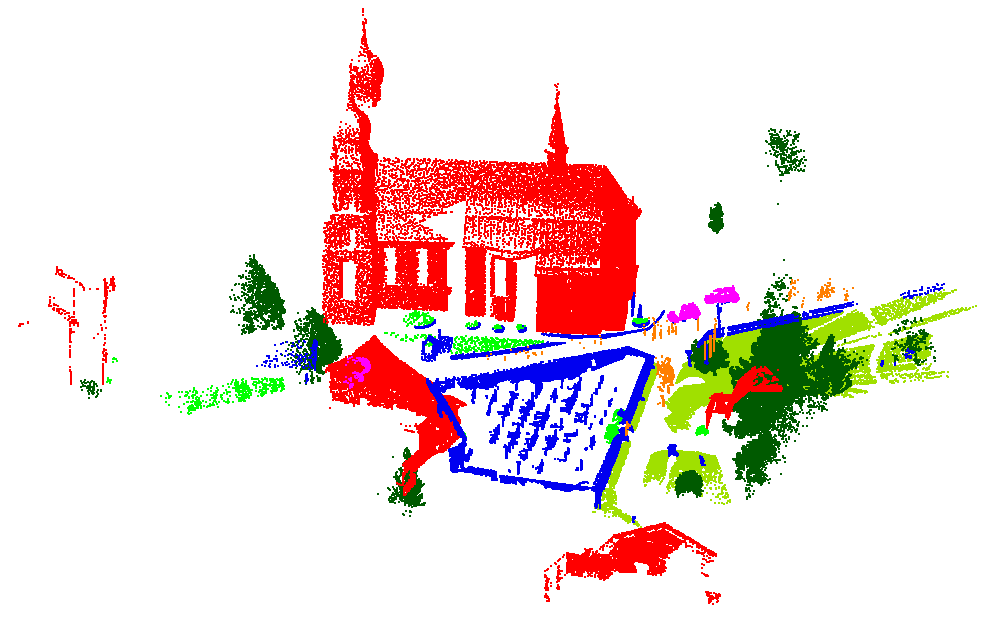
从网站下载了三幅数据，采用其中一幅数据最粗的那个层次的点集中随机挑选的40%的点集作为训练样本，另两幅数据作为测试样本。表5显示了三个场景中点的数量以及对应的多层次点集的总点集数。其中场景1是训练数据。表6显示了各类别的分类精度。图5显示了三个场景的点云数据，以及场景2和场景3分类后的结果。其中浅绿色为自然地面，暗绿色为高植被，亮绿色为低植被，红色为建筑，紫色为hard scape，橙色为人，粉色为车。

表6样本内点的数量以及对应的多层次点集的总点集数

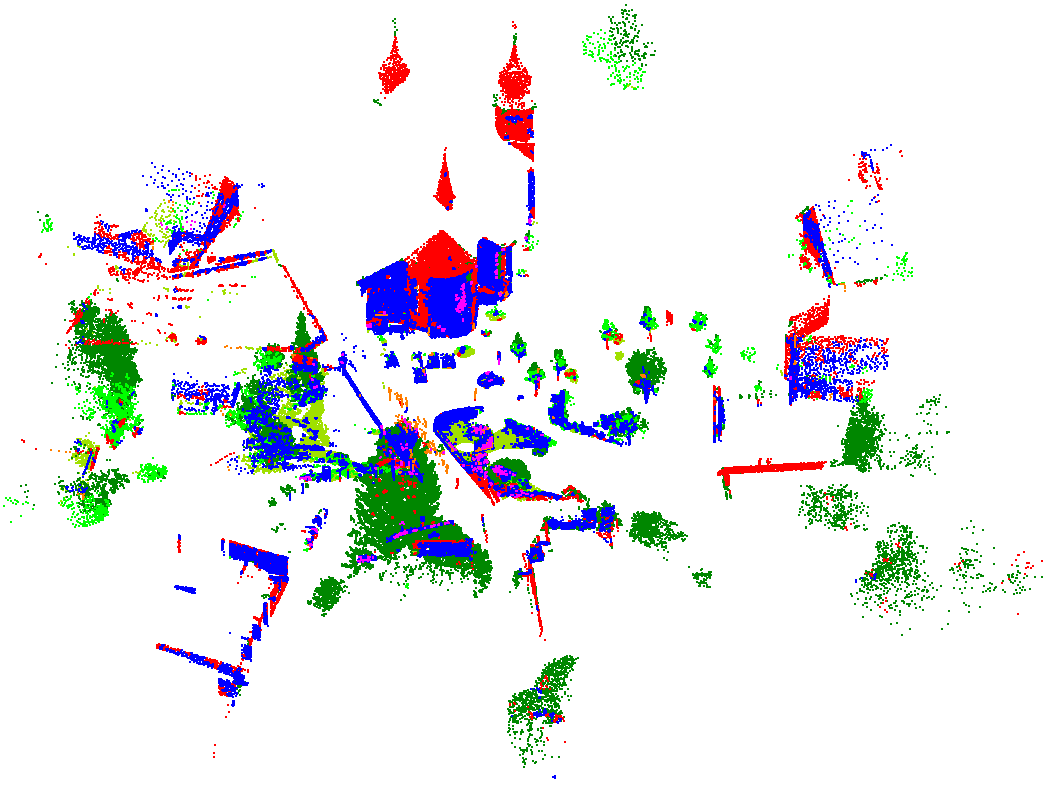
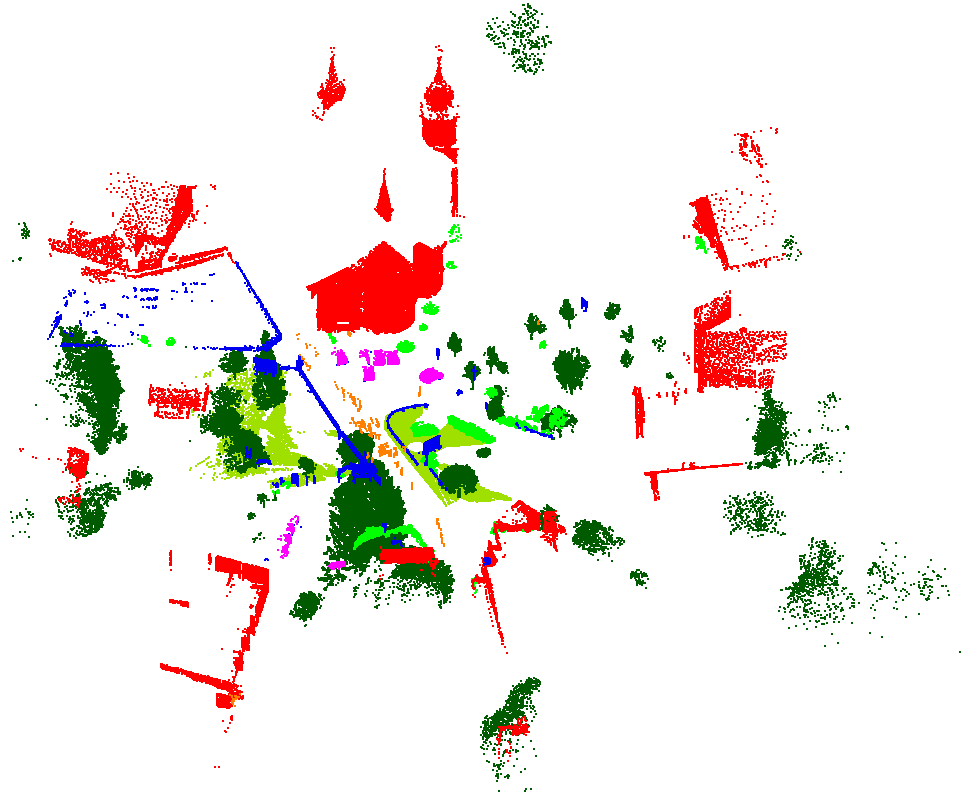
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| scene | natural terrain | high vegetation | low vegetation | buildings | hard scape | scanning artefacts | cars |
| 1 | 4924691  (113311) | 352455  (8252) | 172081  (3944) | 1611908  (36433) | 46140  (1090) | 751  (22) | 403970  (9049) |
| 2 | 3174149  (72081) | 1027837  (24161) | 592309  (13756) | 539935  (12233) | 1260888  (28367) | 7040  (176) | 92873  (2097) |
| 3 | 3507576  (80268) | 2537763  (59086) | 49680  (1170) | 1241838  (28049) | 762982  (17190) | 13899  (318) | 65636  (1471) |

表7分类精度(precision/recall and Accuracy)

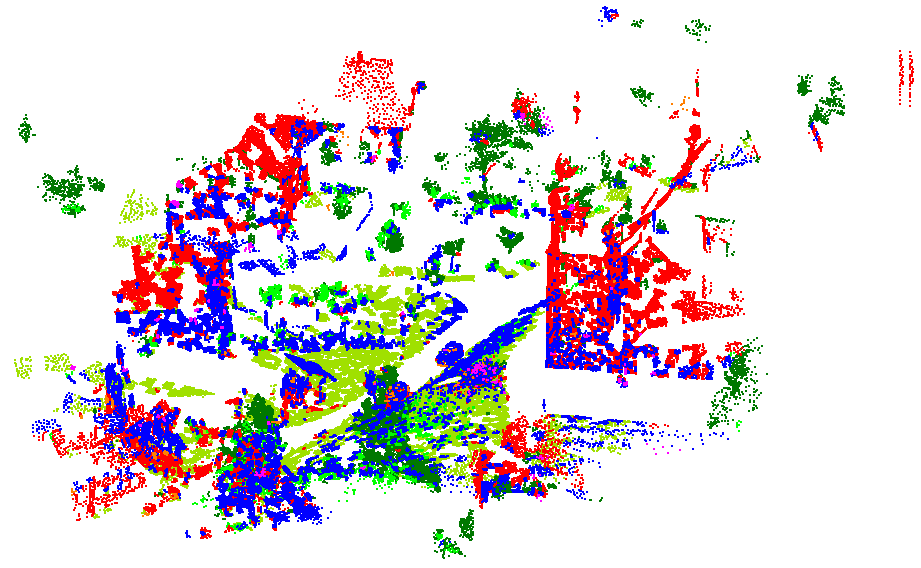
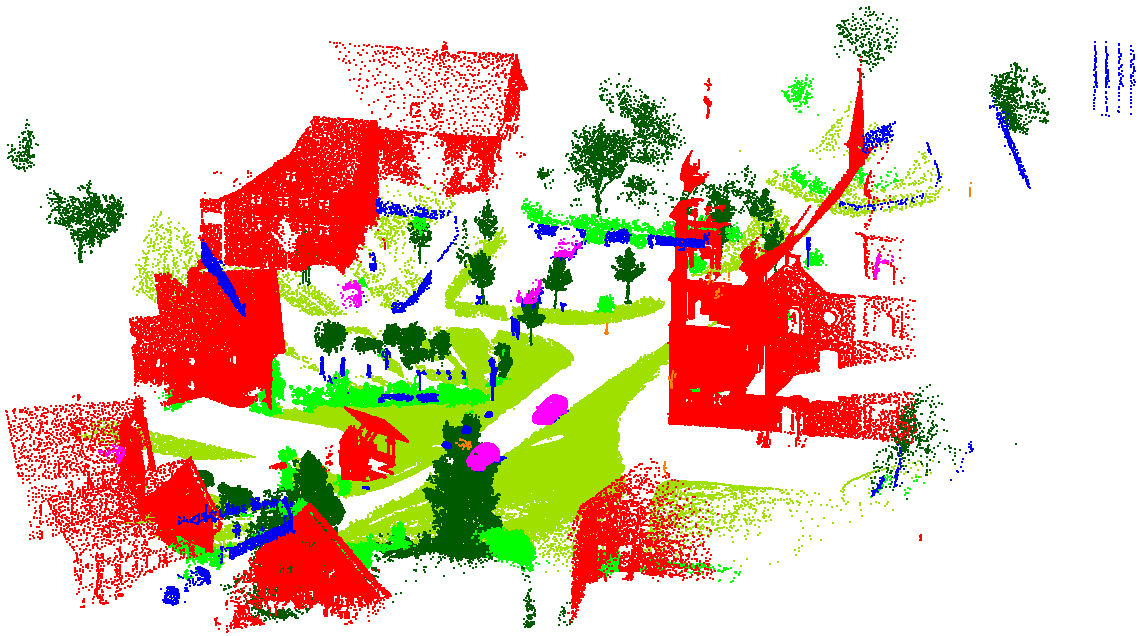
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Scene2 | natural terrain | high vegetation | low vegetation | buildings | hard scape | scanning artefacts | cars | Accuracy |
| Test data | 94.3/87.9 | 81.3/47.1 | 10.8/38.5 | 41.9/67.0 | 27.9/36.6 | 1.3/3.2 | 0/0 | 66.9 |
| Method in[17] | 74.1/64.3 | 58.9/23.6 | 0/19.4 | 20/33.0 | 1.2/23.4 | 0/0 | 0/0 | 46.1 |
| Method in[18] | 82.7/76.7 | 50.3/29.1 | 7.8/27.2 | 27.2/44.6 | 0/15.4 | 0/0 | 0/0 | 49.7 |
| Method in[4] | 74.8 /69.5 | 48.5/27.9 | 6.0/23.7 | 23.2/38.3 | 1.3/0 | 0/2 | 0/0 | 45.2 |
| Method in[35] | 77.8/72.0 | 48.9/27.9 | 8.9/30.8 | 27.1/44.6 | 0/2.9 | 4.5/2.6 | 5/1.3 | 47.1 |
| Scene2 | natural terrain | high vegetation | low vegetation | buildings | hard scape | scanning artefacts | cars | Accuracy |
| Test data | 88.6/92.8 | 59.1/38.6 | 21.6/11.7 | 68.4/94.5 | 49.4/3.0 | 0/0 | 0/3 | 76.7 |
| Method in[17] | 76.9/84.5 | 85.2/13.5 | 8.5/37.8 | 22.7/79.3 | 19.3/12.6 | 0/0 | 0/0 | 59.6 |
| Method in[18] | 76.7/80.3 | 60.3/39.0 | 5.1/3.0 | 40.5/55.0 | 30.6/1.9 | 0/0 | 0/4 | 62.0 |
| Method in[4] | 69.3/72.9 | 53.8/34.5 | 6.7/3.4 | 42.6/57.7 | 25.1/15.6 | 0/0 | 0/0 | 0.575 |
| Method in[35] | 65.6/68.7 | 56.1/36.4 | 30.6/16.5 | 58.4/79.4 | 20.4/1.2 | 4.0/3.7 | 10.2/5.7 | 59.5 |

****

(a)场景1 (b)训练样本

****

(c)场景2 (d) 场景2分类结果

****

(e)场景3 (f) 场景**3**分类结果

图5显示了三个场景的点云数据，采样后形成的训练样本以及场景2和场景3分类结果

地面激光雷达的数据分类结果不如机载激光雷达分类精度高，这主要是因为，这个场景中的类别之间相似度较高，且有些分类在形状上并没有明显的区别，尤其是 hard scape本身更加接近于一种概念性的地物和其他各类的区分度极低，一块类似汽车车头的石头和汽车从外形上很难区分。这导致虽然在训练数据中分类精度很高但是在测试数据中泛化性能很差。从图5中可以发现大量的点被分成了hard scape。同时自然地面和高低植被之间也存在大量的混淆尤其是和低植被之间，可以看到大量低植被被分成自然地面。而建筑和植被之间很少有互相误分的情况。类别之间的相似度高就更需要有区分度的特征，但是地面激光点云变化相对机载激光点云要剧烈的多，这加剧了分类的困难程度，同时因为我们的神经网络还是需要依赖于输入特征，所以输入特征本身的表达能力会影响最终的分类结果。在以后的工作中不再直接依赖于输入特征，而是利用深度学习直接从点中提取特征并分类。同时因为训练数据中人的样本较少，人的类别没有泛化能力。但是也发现采用同样的特征，我们的方法相对于其他方法精度有较大的提高。同时结合机载点云的分类结果，当场景越复杂，采用我们的方法精度提升的越多。这说明虽然我们的方法受到初始输入特征的影响，但是对初始特征的利用能力是最好的，即使初始特征不好，但是经过我们的方法以后能够产生较大的性能提升。

**D．模型分析**

因为Net3是跟在Net1和Net2后面，无法单独测试Net3的效果，所以 为了测试各个层次的层数对最后分类结果的影响，我们采用不加Net3的不同层数的Net1和Net2以及添加了Net3的不同层次的Net1和Net2来说明个层次的效果。



表8不添加Net3时各场景采用不同层次的Net1和Net2的分类精度

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Net1  Net2 | 零个Net1 | 一个Net1 | 两个Net1 | 三个Net1 | 四个Net1 |
| 零个Net2 |  |  |  |  |  |
| 一个Net2 |  |  |  |  |  |
| 两个Net2 |  |  |  |  |  |
| 三个Net2 |  |  |  |  |  |

表9添加Net3时各场景采用不同层次的Net1和Net2的分类精度

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Net1  Net2 | 零个Net1 | 一个Net1 | 两个Net1 | 三个Net1 | 四个Net1 |
| 零个Net2 |  |  |  |  |  |
| 一个Net2 |  |  |  |  |  |
| 两个Net2 |  |  |  |  |  |
| 三个Net2 |  |  |  |  |  |

采用不同层组合方式的分类精度如表6所示，表中较高的分类精度用红色标出。对Scene I-II实验了0-4个 Net1和0-3个Net2；因为更多的层数反而导致精度降低，所以对场景3实验了0-3个 Net1和0-3个Net2。总体来看每个场景都自己最佳的Net1和Net2的层数，而当达到最佳的层数以后，随着Net层数的增加精度会下降，这说明不同地物特征需要通过不同层数的神经网络进行提取，而层数过多以后反而造成过拟合会降低精度。三个场景最佳的几个结果都是采用了Net1和Net2组合的方式得到的，单独采用Net1或Net2分类精度都不高，说明了采用先分开处理各特征再综合处理的方式的有效性。Scene I-II在Net1和Net2的总层数为4层左右时，Scene III在Net1和Net2的总层数为2-3层时，分类精度都相对较高，这也许是因为Scene I-II中汽车类别和其他地物的区分度较低，所以这两个场景需要的层数比场Scene III要多。

下面对Net4和Net5结构的测试都是基于最佳的分类结果所采用的Net1和Net2的层数进行的。

Net4层利用了点云中的空间结构信息，为了说明点云分类中边界点，主体部分点对分类结果的影响，我们试验了只采用边界点，只采用主体部分点两种方式；同时为了显示针对边界点和主体部分点采用不同权重的效果，我们试验了采用所有点但是采用同样的权重和采用不同的权重两种方式；更进一步的为了验证Net4的有效性，又试验了对边界点和主体部分点采用不同权重后，但是没有Net4的结果。表6显示了这些结果。为了方便我们用测试点1-6分别表示上述的6中情况。

表10添加Net3时各场景采用不同层次的Net1和Net2的分类精度

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 测试点1 | 测试点2 | 测试点3 | 测试点4 | 测试点5 | 测试点6 |
| Scene I |  |  |  |  |  |  |
| Scene II |  |  |  |  |  |  |
| Scene III |  |  |  |  |  |  |
| Scene IV |  |  |  |  |  |  |
| Scene V |  |  |  |  |  |  |

Net5的结构保证了不同层次的点云能够在同一个网络框架中进行分类，既能提取多层次的特征，又能独立的对各层次进行判别。如果不采用Net5的结构而是分别对每一个层次以当前的网络结构进行训练并得到对应的网络，最后再同样采用包含最细层次点集的多个层次点集概率的连乘最为分类结果的分类精度为94.2%，93.5%，92.3%，整个分类精度大大下降，这说明采用Net5结构能较大幅度提升分类精度。

**IV. CONCLUSION**

本文提出了一种从原始激光雷达点云中进行多层次点集特征提取的深度神经网络结构。该结构能将大小不一的点集中的单点特征构整合成点集特征，并能提取多层次点集公共特征，同时进行多层次点集的分类，增加分类精度。该结构利用点云的特征结构性，先分别处理点云特征中不同部分，然后，再综合处理各部分输出，能够充分挖掘各特征内部及特征间的关联关系，使提取的点集特征更显著，同时该结构利用点云的空间结构性，基于DMst构建的点与点之间联系，通过剪枝的pooling层去除DMst中叶子节点，并且强化保留下来的点，保证特征不受不良点特征影响，突出主要结构的单点特征，提高分类鲁棒性。

当前的模型中我们还是需要先从单点特征开始进行深度网络的特征提取，今后我们会尝试直接利用点云的坐标作为输入来提取特征，达到自动特征提取的效果。同时我们会尝试采用一些贝叶斯网络和RNN的结构对我们网络结构进行改进，因为在Net3中直接采用平均池化有可能会损失大量的信息。随着后续样本的增加及计算效率的改进，我们也将进行多类别的分类测试，同时尝试点云进行目标识别和检索工作。

REFERENCES

[1] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 21, pp. 433-449, 1999.

[2] Z. Wang, L. Zhang, T. Fang, P. T. Mathiopoulos, H. Qu, D. Chen, et al., "A structure-aware global optimization method for reconstructing 3-D tree models from terrestrial laser scanning data," Geoscience and Remote Sensing, IEEE Transactions on, vol. 52, pp. 5653-5669, 2014.

[3] N. Chehata, L. Guo, and C. Mallet, "Airborne lidar feature selection for urban classification using random forests," International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 38, p. W8, 2009.

[4] B. Guo, X. Huang, F. Zhang, and G. Sohn, "Classification of airborne laser scanning data using JointBoost," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 100, pp. 71-83, 2015.

[5] M. Kragh, R. N. Jørgensen, and H. Pedersen, "Object Detection and Terrain Classification in Agricultural Fields Using 3D Lidar Data," in Computer Vision Systems, ed: Springer, 2015, pp. 188-197.

[6] N. Brodu and D. Lague, "3D terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: Applications in geomorphology," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 68, pp. 121-134, 2012.

[7] J. Zhang, X. Lin, and X. Ning, "SVM-based classification of segmented airborne LiDAR point clouds in urban areas," Remote Sensing, vol. 5, pp. 3749-3775, 2013.

[8] S. Pu, M. Rutzinger, G. Vosselman, and S. O. Elberink, "Recognizing basic structures from mobile laser scanning data for road inventory studies," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 66, pp. S28-S39, 2011.

[9] B. Yang, Z. Dong, G. Zhao, and W. Dai, "Hierarchical extraction of urban objects from mobile laser scanning data," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 99, pp. 45-57, 2015.

[10] K. Fukano and H. Masuda, "Detection and Classification of Pole-Like Objects from Mobile Mapping Data," ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 1, pp. 57-64, 2015.

[11] H. Zhang, J. Wang, T. Fang, and L. Quan, "Joint Segmentation of Images and Scanned Point Cloud in Large-Scale Street Scenes With Low-Annotation Cost," Image Processing, IEEE Transactions on, vol. 23, pp. 4763-4772, 2014.

[12] E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-of-features image classification," in Computer Vision–ECCV 2006, ed: Springer, 2006, pp. 490-503.

[13] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," Machine learning, vol. 42, pp. 177-196, 2001.

[14] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," the Journal of machine Learning research, vol. 3, pp. 993-1022, 2003.

[15] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, 2009, pp. 1794-1801.

[16] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Sparse representation based fisher discrimination dictionary learning for image classification," International Journal of Computer Vision, vol. 109, pp. 209-232, 2014.

[17] Z. Wang, L. Zhang, T. Fang, P. T. Mathiopoulos, X. Tong, H. Qu, et al., "A multiscale and hierarchical feature extraction method for terrestrial laser scanning point cloud classification," Geoscience and Remote Sensing, IEEE Transactions on, vol. 53, pp. 2409-2425, 2015.

[18] Z. Zhang, L. Zhang, X. Tong, Z. Wang, B. Guo, X. Huang, et al., "A Multilevel Point-Cluster-Based Discriminative Feature for ALS Point Cloud Classification."

[19] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," science, vol. 290, pp. 2319-2323, 2000.

[20] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," Neural computation, vol. 10, pp. 1299-1319, 1998.

[21] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," Neural computation, vol. 15, pp. 1373-1396, 2003.

[22] J. Weston, F. Ratle, H. Mobahi, and R. Collobert, "Deep learning via semi-supervised embedding," in Neural Networks: Tricks of the Trade, ed: Springer, 2012, pp. 639-655.

[23] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 35, pp. 1798-1828, 2013.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012, pp. 1097-1105.

[25] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580-587.

[26] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, et al., "3d shapenets: A deep representation for volumetric shapes," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1912-1920.

[27] Z. Zhu, X. Wang, S. Bai, C. Yao, and X. Bai, "Deep learning representation using autoencoder for 3d shape retrieval," in Security, Pattern Analysis, and Cybernetics (SPAC), 2014 International Conference on, 2014, pp. 279-284.

[28] J. Xie, Y. Fang, F. Zhu, and E. Wong, "Deepshape: Deep learned shape descriptor for 3d shape matching and retrieval," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1275-1283.

[29] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Y. Ng, "Convolutional-recursive deep learning for 3d object classification," in Advances in Neural Information Processing Systems, 2012, pp. 665-673.

[30] H. Guan, Y. Yu, Z. Ji, J. Li, and Q. Zhang, "Deep learning-based tree classification using mobile LiDAR data," Remote Sensing Letters, vol. 6, pp. 864-873, 2015.

[31] D. Maturana and S. Scherer, "3D convolutional neural networks for landing zone detection from lidar," in Robotics and Automation (ICRA), 2015 IEEE International Conference on, 2015, pp. 3471-3478.

[32] He K, Zhang X, Ren S, et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification[C] Proceedings of the IEEE International Conference on Computer Vision. 2015: 1026-1034.

[33] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In ICML, 2015.

[34] Niemeyer J, Rottensteiner F, Soergel U. Contextual classification of lidar data and building object detection in urban areas. ISPRS journal of photogrammetry and remote sensing, 2014, 87: 152-165.

[35] Z. Li, L. Zhang, X. Tong, B. Du, Y. Wang, L. Zhang and Z. Zhang, “A Three -step Approach for TLS Point Cloud Classification,” *IEEE Trans. Geosci. Remote Sens*., vol. 54, no. 9, pp. 5412-5424, Sep. 2016.

### [36] Zhenxin Zhang, Liqiang Zhang, Xiaohua Tong, Bo Guo, Liang Zhang, Xiaoyue Xing. Discriminative Dictionary Learning-Based Multi-Level Point-Cluster Features for Urban Point Cloud Classification. IEEE Transactions on Geoscience and Remote Sensing, DOI: 10.1109/TGRS.2016.2599163.