基于 RGB-D 视频流的室内环境 3D 目标高效检测

缪永伟, 陈佳慧, 张新杰, 马文娟, 孙树森

(浙江理工大学信息学院 杭州 310018)
(ywmiao@zstu.edu.cn)

摘要: 由于室内复杂环境 RGB-D 扫描数据不完整、物体相互遮挡等缺陷, 以及表示部分场景的单帧数据输入的局限性, 导致难以通过一次操作高效检测室内场景中的所有 3D 目标物体. 为了克服难以感知获取室内场景中全部物体信息以及场景 3D 目标检测效率低等的难点, 提出一种基于 RGB-D 视频流的室内环境 3D 目标高效检测方法. 首先, 利用 Kinect 相机获取待检测的室内环境 RGB-D 视频流, 经预处理得到连续帧彩色图及其对应的扫描点云信息; 其次, 利用哈希算法从连续帧彩色图中提取内容敏感的视频关键帧, 并根据相邻关键帧所包含的物体种类及个数为其构建目标语义关系, 以确保各关键帧均出现不一样的目标物体; 然后, 利用神经网络 VoteNet 对视频关键帧点云数据进行 3D 目标检测, 并利用四元数球面线性插值算法插值相邻关键帧的相对位姿关系以估计其余帧检测结果, 最终实现 RGB-D 视频流中各帧数据的 3D 目标检测. 使用 SUN RGB-D 数据集对关键帧检测网络进行训练, 与基于 VoteNet 的视频流逐帧检测方法相比, 该方法的目标检测结果准确, 同时大大缩短了视频流整体检测耗时. 实验结果表明该方法的有效性和高效性.

关键词: 室内环境; RGB-D 视频流; 3D 目标检测; 哈希算法; 深度学习
中图法分类号: TP391.41 **DOI:** 10.3724/SP.J.1089.2021.18630

Efficient 3D Object Detection of Indoor Scenes Based on RGB-D Video Stream

Miao Yongwei, Chen Jiahui, Zhang Xinjie, Ma Wenjuan, and Sun Shusen

(College of Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018)

Abstract: For indoor object detection, the input complex scenes often have some defects such as incomplete RGB-D scanning data or mutual occlusion of its objects. Meanwhile, due to the limitations of single RGB-D data or point cloud data input of indoor scenes, it is always difficult to detect all of 3D objects simultaneously. In order to overcome this issue and also alleviate its low efficiency for indoor object detection, an efficient 3D object detection method is proposed which takes RGB-D video streams as input. First, the RGB-D video stream of different indoor environments can be obtained using Kinect camera, and also captured its continuous RGB frames and corresponding point cloud data. Secondly, the Hash function is adopted to extract the content-sensitive key frames from the continuous RGB frames, and the objects semantic relationship can also be constructed according to the type/number of 3D objects contained in adjacent key frames for ensuring that different objects will appear in each key frame. Then, 3D objects of the extracted key frames can be detected by using VoteNet, and the detection results of other frames can be estimated owing to relative posture relationship between adjacent frames by using the quaternion spherical linear interpolation algorithm. Finally, it can achieve efficient 3D object detection for each

收稿日期: 2020-08-27; 修回日期: 2020-12-07. 基金项目: 国家自然科学基金(61972458); 浙江理工大学科研基金(17032001-Y).
缪永伟(1971—), 男, 博士, 教授, 博士生导师, CCF 杰出会员, 主要研究方向为计算机图形学、数字几何处理、计算机视觉、机器学习; 陈佳慧(1995—), 女, 硕士研究生, 主要研究方向为计算机视觉、机器学习; 张新杰(1996—), 男, 硕士研究生, 主要研究方向为计算机图形学、机器学习; 马文娟(1982—), 女, 博士, 讲师, 主要研究方向为计算机视觉; 孙树森(1975—), 男, 博士, 副教授, 主要研究方向为计算机图形学、虚拟现实.

frame in the RGB-D video stream. Using SUN RGB-D dataset to train the object detection network of key frame, the detection result of proposed method is accurate, and the overall detection time is greatly reduced if comparing with the VoteNet based frame-by-frame detection scheme. Experimental results demonstrate that proposed method is effective and efficient.

Key words: indoor environment; RGB-D video stream; 3D object detection; Hash function; deep learning

针对室内复杂场景的 3D 目标检测是计算机视觉、计算机图形学和机器人研究领域中的重要问题,对于增强现实、室内机器人实现人机交互等应用具有现实意义^[1]. 机器人为了更好地实现在室内复杂环境中的智能抓取和智能交互,并完成特定任务,需要获取必要的 3D 目标信息,如目标姿势、目标大小和位置等,以更好地理解室内环境^[2]. 常见的室内场景具有物体杂乱、相互遮挡等情况,同时,在针对利用深度相机输入(如 RGB-D 数据)实现室内复杂场景的 3D 目标检测中,获取的扫描数据通常并不完整^[3]. 除此之外,由于单视角获取的扫描数据通常只能表示复杂场景的局部区域,部分物体的位置和姿态信息很难仅通过一次拍摄获取,这些给基于 RGB-D 数据的室内场景目标检测带来困难,导致难以一次性完整地检测室内复杂场景中的所有物体. 此外,利用 RGB-D 扫描数据进行场景目标检测时,通常需要进行特定处理,如 Zhang 等^[4]将深度数据体素化并作为神经网络的输入进行目标检测,基于体素表示的 3D 目标检测由于需要对输入数据进行预处理使得检测效率较低.

为了克服难以获取室内复杂场景中全部物体信息以及场景 3D 目标检测效率低等的难点,本文提出一种基于 RGB-D 视频流的室内环境 3D 目标的高效检测方法. 需要指出的是,为克服需要对室内复杂环境的多角度多次拍摄的局限性,本文直接利用 RGB-D 视频流作为输入数据,通过单次拍摄即可感知获取场景中的全部物体信息;为解决对视频流数据进行逐帧检测导致检测效率低的问题,本文对视频流先利用哈希值算法提取内容敏感的关键帧,以确保各关键帧均出现不一样的目标物体,从而减少冗余帧的出现;根据关键帧的目标检测结果,结合球面线性插值算法插值相邻关键帧检测得到的相对位姿关系,得到视频流各帧的目标检测结果,从而提高了视频流的 3D 目标检测效率. 以多段室内场景 RGB-D 视频流数据为例,利用本文方法能够有效检测视频流各帧场景中的 3D 目标;与传统的逐帧检测策略相比,本文方法

能够高效地检测室内场景中的 3D 目标对象.

1 相关工作

根据输入数据的不同,3D 目标检测可以分为基于 RGB-D 数据的 3D 目标检测和基于离散点云数据的 3D 目标检测.

基于 RGB-D 数据的 3D 目标检测中, Deng 等^[2]提出利用卷积神经网络分别对彩色图像及深度图像进行特征提取的 2.5D 方法,其利用 Fast R-CNN 网络对图像进行目标检测,最终将 2D 检测结果转换到 3D 空间中,实现 3D 目标检测. 但其在目标检测中由于未能充分利用获取的深度数据中的场景几何结构信息,导致目标检测的准确度较低. 为了充分地利用场景物体的几何信息, Song 等^[5]将输入的深度图转化为点云数据,并利用支持向量机(support vector machine, SVM)分类器对 3D 滑动窗口提取的点云特征进行训练,以检测确定物体的位置. 在此基础上, Liu 等^[6]利用 3D 滑动窗口从 RGB-D 数据中提取图像特征及点云特征,并通过 R-CNN 从彩色图和深度图中分别获取预测的 3D 物体边界框,最后使用 SVM 分类器实现物体的 3D 目标检测. 此外,为了提升 3D 目标检测效率, Song 等^[7]采用 3D 卷积神经网络代替 SVM 分类器,并将彩色图以及由深度图获取的 3D 体素同时作为目标检测的输入,提出 3D 空间中的候选区域建议网络(region proposal networks, RPN),并利用 3D 卷积神经网络生成深度滑动窗口输出目标检测结果. Hou 等^[8]首先利用 2D 卷积网络对彩色图提取特征,并采用 3D 重建方法将彩色图特征投影到 3D 空间中,结合 3D 卷积神经网络提取几何及颜色特征,最后使用 3D 候选 RPN 进行目标检测. 然而,由于使用 3D 卷积神经网络通常导致计算复杂度高、目标检测时间长的问题, Lahoud 等^[9]提出利用 2D 目标检测网络 Faster R-CNN 限制在 2D 图像上的搜索空间的方法,首先定位可能目标周围的 2D 边界框,然后利用多层感知器回归物体的 3D 边界框,从而缩

短目标检测时间. Qi等^[10]提出利用F-PointNet结构实现3D目标检测,该网络在利用2D卷积神经网络提取彩色图特征的同时,使用PointNet对投影为斜锥体结构的点云数据进行3D特征提取,最终得到目标检测的结果.虽然此类方法的检测效率与准确度较之前有所提升,但由于采用简单的多层感知器预测物体位置和朝向,目标检测速度和准确度不高,且其仍主要依赖于2D目标检测技术,并未充分利用物体的3D几何信息.为此,研究者考虑直接将点云数据作为输入进行目标检测.

基于点云数据的3D目标检测中,传统方法通常将点云转变为2D图片或将其体素化,如Qi等^[11]将点云数据体素化为3D网格数据后采用3D卷积神经网络进行目标检测;Zhou等^[12]则将点云离散化为网格数据后利用3D卷积网络进行特征提取,然后利用候选RPN对物体进行检测与位置回归.由于鸟瞰视角下的点云数据在保留物体高度信息的同时数据缺失也较少,因此为了更有效地利用点云信息,Yang等^[13]提出利用卷积神经网络快速、有效地对点云数据进行3D目标检测的方法.首先将鸟瞰视角下的点云数据作为输入,并将点云数据转变为图像矩阵,最终利用全卷积网络完成3D目标检测.Engelcke等^[14]提出了利用一对中心点对称的投票机制的高效3D卷积网络,以避免遍历全部信息所导致的烦琐计算,同时提升了3D目标检测效率.然而,由于上述这些基于卷积网络的目标检测方法要求输入数据符合标准模式,而点云具有无序性,是一种不规则的数据,因此导致其适用范围受限、目标检测效率较低.最近,PointNet^[15]和PointNet++^[16]直接以原始点云数据作为输入,不需要对输入数据进行转化与处理,尽管该网络主要应用于点云数据的分类/分割任务中,其也为目标检测工作提供了借鉴.Lang等^[17]首先将点云数据网格化,并利用PointNet提取特征,其次将点云特征转换为2D伪图像,最后利用2D检测网络进行目标检测,其检测准确度主要依赖于2D检测器. Shi等^[18]提出采用PointRCNN进行目标检测的方法,首先提取高质量点云信息,将点云数据分割为在物体上的点和不在物体上的点;然后直接从点云数据中生成3D候选框,最后通过将候选框中的点云转化为规范坐标获取更多的局部信息,以提高3D目标检测的准确度.Chen等^[19]提出快速PointRCNN方法,通过将原始点云与体素化的点云相结合,保留准确的点云坐标位置,进一步提升了目标检测的效率.Yang等^[20]利用PointNet++直接从

原始点云中获取点云特征信息,并进行3D目标检测,该方法由于需要分2阶段进行,无法实现端到端的3D目标检测.此外,由于采用单目相机拍摄获取的点云数据通常无法像全景图一样能够表示大型室内复杂环境的全景数据,同时由于室内复杂场景点云数据分布杂乱、数据缺失等,使得该方法难以对室内场景视频流数据进行高效的目标检测.

为了有效地实现室内环境3D目标高效检测,本文将室内场景RGB-D视频流数据作为输入,先仅针对关键帧的点云数据进行目标检测,并利用插值方法根据相邻帧物体之间的相对位姿关系估计视频流其他各帧的检测结果,最终实现RGB-D视频流的高效检测.

2 本文方法流程

本文以室内场景RGB-D视频流数据作为输入,结合视频关键帧提取、基于神经网络的关键帧目标检测、基于视频相邻帧物体相对位姿关系的目标检测结果,最终实现RGB-D视频流的室内复杂场景3D目标检测.方法流程如图1所示,步骤如下.

Step1. RGB-D视频流的获取及表示.由于Azure Kinect DK相机能够提供待扫描场景深度数据和彩色RGB图像相匹配的数据流,故将使用该设备录制的RGB-D视频流作为输入数据.在手持单台Azure Kinect DK相机录制室内环境RGB-D视频流的过程中,需要保持手部移动速度较慢且均匀稳定、不出现大幅度的上下抖动以及不对已经录制过的场景进行重复录制.录制过程中可以利用相机软件开发包中提供的K4Arecorder工具将视频流保存到数据文件中,并相应获取连续的RGB图像帧以及相对应的场景点云数据.

Step2. 基于哈希感知的内容敏感关键帧提取.为了克服逐帧检测所导致的检测效率低的问题,首先需要对RGB-D视频流数据进行内容敏感关键帧提取,根据相邻关键帧包含的物体种类及个数为其构建目标语义关系,以确保各关键帧均出现不一样的目标物体.由于不相似图像间的哈希值相差较大,首先利用哈希算法为每幅图像生成独特的哈希值,然后将视频第1帧确定为模板帧,其他帧与模板帧依次进行哈希值比较,将差值较大的视频帧确定为关键帧并更新为模板帧,重复上述过程,直到完成最后1帧的比较,最终确定连续帧彩色图中的内容敏感关键帧.

Step3. 视频关键帧的3D目标端到端检测.利用神经网络VoteNet^[21]对经过无效点去除、坐标系调整后的关键帧点云数据进行3D目标检测,从而获取室内场景物体的种类、位置、朝向等信息.同时,为了充分利用

物体几何信息以提高目标检测准确度, VoteNet 在结合 PointNet++ 提取点云特征信息后利用霍夫投票方法在物体质心周围生成集群, 最后利用 PointNet 生成物体 3D 边界框, 并获取物体的种类、位置等信息。

Step4. 基于 RGB-D 视频流的 3D 目标检测. 由于室内场景录制时相机保持平稳移动, 故可将 RGB-D 视频

流视为匀速运动下呈现的场景变化, 且所提取的内容敏感关键帧之间各帧并未出现新的目标物体. 因此, 可以利用四元数球面线性插值算法插值^[22]相邻关键帧检测得到的物体之间相对位姿关系, 并结合关键帧点云的 3D 目标检测结果估算其余各帧的目标检测结果, 最终实现 RGB-D 视频流的室内场景 3D 目标检测。

图 1 本文方法流程图

3 基于哈希感知的内容敏感关键帧提取

为了提高室内场景 3D 目标检测的效率, 避免对含有相同物体的视频帧进行重复检测, 同时利用相邻关键帧所含有的物体种类及个数为关键帧之间构建的目标语义关系, 这些均为后续准确估计中间帧的目标检测结果提供了可能, 以高效地实现目标检测. 本文对获取的室内场景 RGB-D 视频流数据首先提取其关键帧, 使各关键帧处均出现新的目标物体. 由于每帧 RGB-D 数据均包含一一对应的彩色图信息和深度信息, 为克服深度信息缺失较多且处理速度慢等难点, 本文利用哈希算法在 RGB 图像连续帧中获取关键帧。

研究者提出了若干视频关键帧提取方法, 如使用基于视频镜头分割的方法^[23]以及采用基于 C-模式的聚类方法^[24]分别进行了关键帧提取, 但此类方法大多应用于体育视频、新闻视频以及电影等时间长且内容变化明显的视频关键帧提取中. 本文结合室内复杂环境以及 RGB-D 视频流录制的特点, 提出利用图像哈希算法^[25]进行视频关键帧提取的方法。

图像哈希算法通常应用于图像相似性检索中,

其哈希值类似人类指纹, 不同的图像将对应不同的哈希值, 当 2 幅图像相似时, 其对应的哈希值较接近. 基于此原理, 本文提出基于哈希感知算法获取 RGB-D 视频流中的内容敏感关键帧, 选取候选帧中哈希值变化较大的, 即出现新物体的视频帧作为关键帧. 为准确地获取内容敏感关键帧, 在视频关键帧提取时, 本文采用均值哈希值与差值哈希值计算对不同视频帧的相似度, 以进行更为精准判断. 在均值哈希值计算生成中, 首先将图像缩小为 8×8 的尺寸, 并将其转变为灰度图以简化图像色彩; 其次计算图像灰度值平均值, 并根据平均值将每个像素二值化(大于平均值为 1, 小于平均值为 0), 最后生成均值哈希值. 差值哈希值计算与均值哈希值类似, 将图像灰度化后计算每行相邻像素的差异值, 前 1 个像素值大于后 1 个像素值为 1, 反之为 0, 最后生成差值哈希值. 在获取得到图像哈希值后, 可以采用汉明距离^[26]逐位比较计算 2 幅图像之间的哈希值差值. 若 2 幅图像之间的哈希值差值大于指定阈值(实验中均值哈希差值取阈值 $\delta_1 = 15$, 差值哈希差值取阈值 $\delta_2 = 20$), 表明 2 幅图像之间相似度较低, 即第 2 幅图像中出现了新物体. 为实现对完整视频流数据的有效检测, 本文

方法默认将视频流第 1 帧作为第 1 个关键帧及与后续帧进行比较的第 1 个模板帧; 同时考虑视频流相邻帧之间相似度较高, 为提高关键帧提取效率, 本文选取与模板帧间隔 4 帧的视频帧与模板帧进行哈希值的比较, 若差值小于实验所指定的阈值, 则继续选取与该视频帧间隔 4 帧的视频帧并与模板帧进行比较; 反之, 则在模板帧与视频帧之间确定第 1 个大于实验所指定阈值的帧, 将该帧作为关键帧输出的同时将其更新为模板帧. 假设室内场景 RGB 视频帧 $F = \{F_1, F_2, F_3, \dots, F_n\}$ (n 为帧数) 对应均值哈希值为 $A_H = \{a_{h1}, a_{h2}, a_{h3}, \dots, a_{hn}\}$, 差值哈希值为 $D_H = \{d_{h1}, d_{h2}, d_{h3}, \dots, d_{hn}\}$, 第 i 帧与第 j 帧对应的均值哈希值差值及差值哈希值差值分别为 A_j^i 和 D_j^i , 关键帧提取算法如下.

算法 1. 关键帧提取算法.

输入. 场景 RGB 视频帧 $F = \{F_1, F_2, F_3, \dots, F_n\}$.

输出. 关键帧 $F_k = \{F_{k_1}, F_{k_2}, F_{k_3}, \dots, F_{k_l}\}$.

Step1. 初始化 $i = 1$.

Step2. 计算模板帧 F_m ($1 \leq m \leq n-5$) 和视频帧 F_{m+5i} 的哈希值差值 A_{m+5i}^m 及 D_{m+5i}^m .

Step3. 若 $A_{m+5i}^m < \delta_1$ 且 $D_{m+5i}^m < \delta_2$, $i = i + 1$, 转 Step2; 否则, 执行 Step4.

Step4. 利用二分法在帧 $F_{m+5(i-1)}$ 与帧 F_{m+5i} 之间确定

关键帧.

Step4.1. 确定帧 $F_{m+5(i-1)}$ 与视频帧 F_{m+5i} 之间的中间帧 $F_{mid} = F_{m+5i-3}$;

Step4.2. 计算中间帧 F_{mid} 与模板帧 F_m 的哈希值差值 A_{mid}^m 及 D_{mid}^m . 若 $A_{mid}^m < \delta_1$ 或 $D_{mid}^m < \delta_2$, 执行 Step4.3; 否则, 转 Step4.4;

Step4.3. 令 $F_{mid} = F_{m+5i-1}$, 计算 A_{mid}^m 及 D_{mid}^m . 若 $A_{mid}^m < \delta_1$ 或 $D_{mid}^m < \delta_2$, 确定 F_{m+5i} 为关键帧, 同时令 $F_m = F_{m+5i}$; 否则 F_{mid} 为关键帧, 同时令 $F_m = F_{mid}$. 转 Step1;

Step4.4. 令 $F_{mid} = F_{m+5i-4}$, 计算 A_{mid}^m 及 D_{mid}^m . 若 $A_{mid}^m > \delta_1$ 且 $D_{mid}^m > \delta_2$, 确定 F_{mid} 为关键帧, 令 $F_m = F_{mid}$; 否则令 $F_{mid} = F_{m+5i-3}$, 计算 A_{mid}^m 及 D_{mid}^m . 若 $A_{mid}^m < \delta_1$ 或 $D_{mid}^m < \delta_2$, 确定 F_{m+5i-2} 为关键帧, 同时令 $F_m = F_{m+5i-2}$; 否则 F_{mid} 为关键帧, 同时令 $F_m = F_{mid}$, 转 Step1.

由于室内场景 RGB-D 视频流数据通常按照其拍摄过程先后进行顺序存储, 导致相邻关键帧之间的数据帧与模板帧的哈希差值依次增大, 因此, 本文方法采用二分法策略提取视频流关键帧. 首先确定与模板帧相差大于阈值的数据帧, 该过程的时间复杂度为 $O(1)$; 然后确定新的模板帧, 该过程的时间复杂度为 $O(\lg n)$; 由此关键帧提取算法的时间复杂度为 $O(\lg n)$.

图 2 所示为本文方法对室内场景 RGB-D 视频

图 2 室内场景 RGB-D 视频流关键帧提取

流的关键帧提取结果. 图 2a 所示为场景 1 视频流的关键帧提取结果, 其视频流共有 63 帧, 包含沙发、桌子及椅子等常见室内物体. 利用本文的关键帧提取方法可以确定第 1 帧、第 24 帧、第 45 帧以及第 60 帧为关键帧, 其中关键帧 2 较关键帧 1 相比完整展示了物体桌子. 图 2b 所示为场景 2 视频流的关键帧提取结果, 其视频流共有 60 帧, 包含 2 把相同的椅子. 利用本文方法可以确定第 1 帧、第 16 帧、第 30 帧及第 49 帧为关键帧, 而关键帧第 2 帧、第 3 帧、第 4 帧分别完整展示了椅子 2 出现及椅子 1 消失的过程. 图 2c 所示为场景 3 视频流的关键帧提取结果, 其视频流共有 90 帧, 包含沙发及 2 把相同的椅子. 利用本文方法可以确定第 1 帧、第 19 帧、第 31 帧、第 41 帧、第 53 帧、第 72 帧及第 86 帧为关键帧, 而关键帧第 2~4 帧分别展示了椅子右边的一小部分、大部分及全部, 并动态还原椅子 1 出现的完整过程. 从场景视频流的关键帧提取结果可以看出, 利用本文方法提取的关键帧中, 其每帧关键帧较前一帧关键帧均出现了不一样的目标物体, 从而能够有效地表示视频流中场景物体的变化过程.

4 视频关键帧的 3D 目标端到端检测

为了能够充分利用场景物体的 3D 几何信息克服 3D 目标检测中准确度较低等问题, 并利用深度神经网络有效地提高目标检测效率, 同时由于 Kinect 录制的 RGB-D 视频流中的每帧均包含相对应的点云信息与彩色图信息, 因此, 在提取场景视频流彩色图像关键帧后, 本文考虑将关键帧对应

的点云信息作为神经网络输入, 以进行高效的 3D 目标检测, 其网络结构如图 3 所示.

本文采用 Qi 等^[21]提出的针对场景 3D 目标检测的 VoteNet 结构. 与其他基于点云的处理方法不同, VoteNet 直接将原始点云作为输入, 不需要对其进行体素化或网格化处理, 能保留完整的点云几何结构信息, 并有效地提高目标检测的准确度. 以室内场景 SUN RGB-D 数据集^[27]中部分物体类别作为测试集, 将 VoteNet 与目前较为先进的深度学习方法进行比较, 其中, DSS 网络^[7]将经 Faster R-CNN 检测后的彩色 RGB 图像与点云数据作为检测网络的共同输入, 并利用 3D 卷积网络作为主要检测结构; COG 网络^[1]则采用滑动窗口机制及基于 3D 霍夫投票机制的检测器; 2D 驱动的目标检测方法^[9]以及 F-PointNet 结构^[10]则将 2D 目标检测与 3D 目标检测相结合, 但以 2D 检测器作为主导结构, 即只有被 2D 检测器检测到的物体才会被 3D 检测结构检测到. 这 4 种方法以及 VoteNet 的检测准确度如表 1 所示, VoteNet 表现良好的原因得益于其能充分利用物体的几何结构信息. 因此, 为准确高效地实现室内场景中全部物体的检测, 以场景 RGB-D 视频流作为输入, 本文使用 VoteNet 作为 3D 目标检测的网络结构. 该网络结构充分利用了原始点云的局部信息, 并利用 PointNet++ 对输入的关键帧点云进行采样及特征提取, 形成带有特征的种子点集. PointNet++ 模块中包含 4 个抽象层以及 2 个特征上采样层, 其中每个抽象层均包含采样层、聚合层以及 PointNet 网络层. 采样层利用最远点采样策略选取 N 个点对输入点进行采样, 选出若干个中心点; 而聚合层则将采样层得到的中

图 3 关键帧 3D 目标检测网络^[21]

表 1 VoteNet 与其他方法在 Sun RGB-D 数据集^[27]上的检测准确度对比

方法	输入	床	椅子	课桌	沙发	桌子	mAP(准确度均值)
COG ^[1]	RGB+点云	63.7	62.2	45.2	51.0	51.3	54.7
DSS ^[7]	RGB+点云	78.8	61.2	20.5	53.5	50.3	52.8
2D-driven ^[9]	RGB+点云	64.5	48.3	27.9	50.4	37.0	45.6
F-PointNet ^[10]	RGB+点云	81.1	64.2	24.7	61.1	51.1	56.4
VoteNet ^[21]	点云	83.0	75.3	22.0	64.0	47.3	58.3

心点划分为若干个区域. 其次, PointNet 网络层将每个区域进行编码并得到其特征向量. 最后, PointNet++ 中的特征上采样层对特征向量进行采样, 形成关键帧点云的种子点集, 将其输入 VoteNet 的投票层.

为克服点云数据无序性使其难以判断目标物体质心位置并导致检测准确度较低的问题, VoteNet 的投票层由多个多层感知器构成, 并利用霍夫投票功能在物体质心周围生成集群, 从而首先输出包含点坐标及特征的投票点集; 然后利用采样及群组模块, 并采用最远点采样策略, 以物体质心为中心点形成多个聚类; 最后通过多个感知器以及非最大抑制策略输出包含物体 3D 边界框的关键帧点云目标检测结果.

5 基于 RGB-D 视频流的 3D 目标检测

本文为了有效地克服 RGB-D 视频流逐帧目标检测所导致的检测时间长、效率低的问题, 对关键帧点云数据进行目标检测后, 仍需要根据关键帧之间的物体位姿关系及关键帧的目标检测结果估计视频流其余帧的检测结果. 根据本文提取的内容敏感关键帧的特点可知, 相邻关键帧之间由出现不一样的目标物体这一条件可以构建目标语义关系, 由此推断相邻关键帧之间其余帧的物体种类个数与第 1 帧关键帧保持一致, 即中间帧物体个数的检测结果与第 1 帧关键帧检测结果相同, 仅存在由于相机移动导致待检测物体位姿发生局部变化的情况. 为了得到由于相机移动导致物体相对位姿变化并获取其余帧的准确检测结果, 本文利用四元数球面线性插值法^[22]结合相邻关键帧的物体之间相对位姿关系及关键帧检测结果估计其余帧的目标检测结果, 最终实现 RGB-D 视频流的 3D 目标高效检测.

通常, 由于相机移动将导致相邻 RGB-D 视频帧中的物体发生相对平移、旋转等位姿变化; 又由于 RGB-D 视频流录制时保持平稳, 故将整段视频流中场景物体视为近似匀速运动, 可以认为相邻帧之间的平移及旋转变化量是相同的. 由此, 本文利用点云配准(iterative closest point, ICP)算法^[28], 通过迭代找到最近邻点之间误差最小的平移旋转变化矩阵, 进而获取相邻关键帧之间的空间位姿关系, 即相对平移及旋转变化量; 之后利用四元数球面线性插值计算得到相邻帧之间的物体位姿变化.

5.1 相邻帧间场景物体的旋转变化量及插值

常用的表示空间旋转变化量的方法有欧拉角、旋转矩阵以及四元数. 欧拉角表示旋转虽较为直观, 但是其角度值不可传递, 旋转顺序影响旋转角度的结果; 旋转矩阵表示法虽简单, 但是利用其进行插值运算难度极大, 而四元数兼具表示简单且易于插值的优点, 因此, 本文采用四元数表示旋转变化量, 并利用四元数球面线性插值获取相邻帧之间的旋转变化量^[22]. 四元数球面线性插值可用于四元数之间的平滑插值, 和普通的线性插值法相比, 利用球面四元数能更好地均分变化角度. 本文方法利用关键帧点云的目标检测结果及关键帧之间的物体相对平移/旋转变化关系估计其余帧检测结果.

假设由关键帧提取算法 1 提取的 RGB-D 视频关键帧对应的点云序列为 $P_F = \{P_{k_1}, P_{k_2}, P_{k_3}, \dots, P_{k_l}\}$; 相邻关键帧间隔帧数为 $S_F = \{s_1, s_2, s_3, \dots, s_{l-1}\}$, 集合 S_F 中可能存在元素值相同的情况, 则有如下过程.

Step1. 利用 ICP 算法^[28]获取相邻关键帧数据之间的相对旋转变化矩阵 $R_F = \{R_{k_1}, R_{k_2}, R_{k_3}, \dots, R_{k_l}\}$, 其中, R_{k_i} 为单位矩阵 I , 并记

$$R_{k_i} = \begin{bmatrix} m_{k_i}^{11} & m_{k_i}^{12} & m_{k_i}^{13} \\ m_{k_i}^{21} & m_{k_i}^{22} & m_{k_i}^{23} \\ m_{k_i}^{31} & m_{k_i}^{32} & m_{k_i}^{33} \end{bmatrix}, i = 1, 2, \dots, l.$$

Step2. 对 $i \neq 1$, 由于满足 $m_{k_i}^{11} + m_{k_i}^{22} + m_{k_i}^{33} > 0$, 并由四元数 $Q_{k_i} = W_{k_i} + X_{k_i}i + Y_{k_i}j + Z_{k_i}k$ 将相邻关键帧点云数据的相对旋转变化矩阵转变为对应的四元数 $Q_F = \{Q_{k_1}, Q_{k_2}, Q_{k_3}, \dots, Q_{k_l}\}$, 其中, Q_{k_i} 为单位四元数, 且

$$\begin{cases} W_{k_i} = \frac{1}{2} \sqrt{1 + m_{k_i}^{11} + m_{k_i}^{22} + m_{k_i}^{33}} \\ X_{k_i} = (m_{k_i}^{32} - m_{k_i}^{23}) / (4 \cdot W_{k_i}) \\ Y_{k_i} = (m_{k_i}^{13} - m_{k_i}^{31}) / (4 \cdot W_{k_i}) \\ Z_{k_i} = (m_{k_i}^{21} - m_{k_i}^{12}) / (4 \cdot W_{k_i}) \end{cases}$$

Step3. 根据文献[22]中对相邻关键帧点云数据 P_{k_i} 及 $P_{k_{i+1}}$ 之间进行球面线性插值, 并计算 2 个关键帧之间的第 j 帧点云数据 P_j 相对于 P_{k_i} 的旋转四元数为

$$Q_{k_i, j} = \frac{\sin(1-j \cdot t \cdot \theta_i) Q_{k_i} + \sin(j \cdot t \cdot \theta_i) Q_{k_{i+1}}}{\sin \theta_i} = W_{k_i, j} + X_{k_i, j}i + Y_{k_i, j}j + Z_{k_i, j}k.$$

其中, $t = 1/s_i = 1/(k_{i+1} - k_i)$; θ 为相邻四元数之间的夹角, $\cos \theta_i = Q_{k_{i+1}} \cdot Q_{k_i}$.

Step4. 将点云对 P_{k_i} 和 $P_{k_{i+1}}$ 之间第 j 帧 P_j 相对 P_{k_i}

的旋转变换四元数转换为矩阵

$$\mathbf{R}_{k_i,j} = \begin{pmatrix} 1-2Y_{k_i,j}^2-2Z_{k_i,j}^2 & 2X_{k_i,j}Y_{k_i,j}+2W_{k_i,j}Z_{k_i,j} & 2X_{k_i,j}Z_{k_i,j}-2W_{k_i,j}Y_{k_i,j} \\ 2X_{k_i,j}Y_{k_i,j}-2W_{k_i,j}Z_{k_i,j} & 1-2X_{k_i,j}^2-2Z_{k_i,j}^2 & 2Y_{k_i,j}Z_{k_i,j}+2W_{k_i,j}X_{k_i,j} \\ 2X_{k_i,j}Z_{k_i,j}+2W_{k_i,j}Y_{k_i,j} & 2Y_{k_i,j}Z_{k_i,j}-2W_{k_i,j}X_{k_i,j} & 1-2X_{k_i,j}^2-2Y_{k_i,j}^2 \end{pmatrix}.$$

Step5. 根据 Step3 和 Step4 可得相邻关键帧点云 $\mathbf{P}_{k_i}$ 和 $\mathbf{P}_{k_{i+1}}$ 之间第 j 帧 $\mathbf{P}_j$ 相对于 $\mathbf{P}_{k_i}$ 的旋转矩阵 $\mathbf{R}_{k_i,j}$, 同理可得相邻关键帧点云 $\mathbf{P}_{k_m}$ 和 $\mathbf{P}_{k_{m+1}}$ 之间第 j' 帧 $\mathbf{P}_{j'}$ 相对于 $\mathbf{P}_{k_m}$ 的旋转矩阵 $\mathbf{R}_{k_m,j'}$, 则进一步可得视频流中任意 2 帧点云数据 $\mathbf{P}_j$ 和 $\mathbf{P}_{j'}$ 之间的关系如下.

当 $m=i$ 时, 位于同一关键帧段中 2 帧数据间关系为

$$\mathbf{P}_{j'} = \mathbf{P}_j \cdot \mathbf{R}_{k_i,j}^{-1} \cdot \mathbf{R}_{k_i,j'} \quad (1)$$

当 $m>i$ 时, 位于不同关键帧段中 2 帧数据间关系为

$$\mathbf{P}_{j'} = \mathbf{P}_j \cdot \mathbf{R}_{k_i,j}^{-1} \cdot \mathbf{R}_{k_i,k_{i+1}} \cdot \mathbf{R}_{k_{i+1},k_{i+2}} \cdots \mathbf{R}_{k_m,k_{m+1}} \cdot \mathbf{R}_{k_m,j'} \quad (2)$$

由此即可获取视频流中 2 帧场景间的旋转变化关系, 但拍摄过程中由于相机移动将不可避免地导致场景物体之间除旋转变化外还存在平移变化, 故需要进一步计算相邻帧间场景物体的平移变化量.

5.2 相邻帧间场景物体的平移变化量及插值

如前所述, 通过点云配准 ICP 算法^[28]可以容易计算相邻关键帧点云数据 $\mathbf{P}_{k_i}$ 和 $\mathbf{P}_{k_{i+1}}$ 之间的相对平移变化矩阵 $\mathbf{T}_{k_i,k_{i+1}} = [t_{k_i}^x \ t_{k_i}^y \ t_{k_i}^z]^T$, 且 $\mathbf{P}_{k_i}$ 和 $\mathbf{P}_{k_{i+1}}$ 间相隔帧数 $s_i = k_{i+1} - k_i$, 则该关键帧段中第 j 帧点云 $\mathbf{P}_j$ 相对于 $\mathbf{P}_{k_i}$ 的平移变化量为 $\mathbf{T}_{k_i,j} = \frac{j-k_i}{s_i} \mathbf{T}_{k_i,k_{i+1}}$.

与式(1)(2)类似, 假设 $\mathbf{P}_{j'}$ 位于相邻关键帧点云 $\mathbf{P}_{k_m}$ 和 $\mathbf{P}_{k_{m+1}}$ 之间, 则可计算得到视频流中任意 2 帧点云数据 $\mathbf{P}_j$ 和 $\mathbf{P}_{j'}$ 之间关系如下.

当 $m=i$ 时, 同一关键帧段中 2 帧数据间关系为

$$\mathbf{P}_{j'} = \mathbf{P}_j \cdot \mathbf{T}_{k_i,j}^{-1} \cdot \mathbf{T}_{k_i,j'} \quad (3)$$

当 $m>i$ 时, 不同关键帧段中 2 帧数据间关系为

$$\mathbf{P}_{j'} = \mathbf{P}_j \cdot \mathbf{T}_{k_i,j}^{-1} \cdot \mathbf{T}_{k_i,k_{i+1}} \cdot \mathbf{T}_{k_{i+1},k_{i+2}} \cdots \mathbf{T}_{k_m,k_{m+1}} \cdot \mathbf{T}_{k_m,j'} \quad (4)$$

联合式(1)(3)以及式(2)(4), 可以确定 2 帧数据之间场景物体的位姿变化关系如下.

当 $m=i$ 时, 同一关键帧段中 2 帧数据间关系为

$$\mathbf{P}_{j'} = \mathbf{P}_j \cdot (\mathbf{T}_{k_i,j}^{-1} + \mathbf{R}_{k_i,j}^{-1}) \cdot (\mathbf{T}_{k_i,j'} + \mathbf{R}_{k_i,j'}) \quad (5)$$

当 $m>i$ 时, 不同关键帧段中 2 帧数据间关系为

$$\begin{aligned} \mathbf{P}_{j'} = & \mathbf{P}_j \cdot (\mathbf{T}_{k_i,j}^{-1} + \mathbf{R}_{k_i,j}^{-1}) \cdot (\mathbf{T}_{k_i,k_{i+1}} + \mathbf{R}_{k_i,k_{i+1}}) \cdot \\ & (\mathbf{T}_{k_{i+1},k_{i+2}} + \mathbf{R}_{k_{i+1},k_{i+2}}) \cdots (\mathbf{T}_{k_m,k_{m+1}} + \mathbf{R}_{k_m,k_{m+1}}) \cdot \\ & (\mathbf{T}_{k_m,j'} + \mathbf{R}_{k_m,j'}) \end{aligned} \quad (6)$$

由于相机运动通常导致视频相邻帧之间物体位姿发生变化, 利用式(5)(6)可以确定 2 帧视频数据之间的物体位姿变化关系. 同时, 根据内容敏感关键帧提取结果可知, 相邻关键帧之间并未出现新的物体, 其他帧物体种类及个数均与前 1 帧关键帧保持语义一致. 由此, 根据第 1 个关键帧点云数据的检测结果及场景物体位姿变化可得到其余视频帧点云数据的检测结果, 最终输出经完整检测的视频流.

6 实验结果与分析

本文方法在 Centos 系统上借助 Python 3.5.3, g++ 5.4.0 以及 CUDA 9.0 和 Pytorch 1.1 深度学习平台得以实现, 其硬件配置为 Intel core i5-7th CPU, 运行内存 4 GB. 本文方法以 Kinect 相机采集得到的不同室内场景 RGB-D 视频流作为输入, 结合基于神经网络的关键帧点云目标检测方法和基于四元数球面线性插值方法插值视频相邻帧点云位姿变化关系, 从而实现了 RGB-D 视频流的 3D 目标高效检测.

6.1 实验数据及预处理

首先, 本文方法使用 SUN RGB-D 数据集^[27]对神经网络进行预训练. SUN RGB-D 数据集是广泛应用于室内场景理解问题中的单视角 RGB-D 数据集, 其中包含约 5 000 幅对 37 类物体进行非模态 3D 边界框标注的训练图像. 本文在对场景目标检测的神经网络进行预训练之前, 需要先根据实际相机参数将 RGB-D 数据转变为点云, 并将录制的 RGB-D 视频流处理为连续帧彩色图像及相对应的点云数据作为方法输入. 具体地说, 为获取实验所需的数据, 本文利用 Kinect Azure DK 相机连接一台计算机在不同室内环境中进行视频流的录制, 录制过程中保持相机平稳, 避免出现明显抖动, 以及不对场景进行中的物体进行重复录制. 同时, 利用相机软件开发工具包中提供的 K4Arecorder 工具将视频流保存到数据文件中, 并将 RGB-D 视频流读取为连续的图像帧以及相对应的点云数据. 此外, 为满足检测网络对于输入点云数据格式的要求, 本文去除点云信息中的无效点后, 将点云沿 X 轴旋转 -90° , 使其 Z 轴垂直于地面. 然后, 对 RGB-D 视频流提取关键帧并根据所录制的视频流对检测网络参数进行调整后对关键帧点云进行 3D 目标检测, 获取关键帧检测结果. 考虑到利用点云配准 ICP 算法^[28]获取相邻关键帧点云间的位姿关系耗时较长, 本文采用 CUDA 对 ICP 算法进行加

速, 并构建相邻关键帧之间的目标语义关系, 进而利用四元数球面线性插值法插值相邻关键帧检测得到的物体之间相对位姿关系估计其余帧的检测结果, 最终实现视频流中各帧数据的 3D 目标检测。

6.2 多场景 RGB-D 视频流目标检测结果

对录制得到的多段室内场景 RGB-D 视频流进行实验, 使用本文方法对视频流中的物体进行 3D 目标检测时, 不同类别物体采用不同框进行标注。作为本文实验输入的多段视频流中包含室内常见的家具, 如沙发、桌子、椅子等分别进行标注。本文实验中首先将使用 VoteNet 对 RGB-D 视频逐帧检测的结果与利用本文方法得到的 3D 目标检测结果进行对比, 体现了其有效性; 其次, 将使用 VoteNet 对视频流进行逐帧检测所需时间与利用本文方法所需时间进行比较, 体现了其高效性。

6.2.1 本文方法的检测有效性

图 4 所示为场景 3 视频流的目标检测效果及比较。图 4a 为视频流第 31~41 帧, 展示了物体沙发右侧从镜头中逐渐消失的过程及视频左侧新的目标物体从进入视频至展示一半的过程, 采用灰色框对视频关键帧进行了标注, 分别为第 31 帧及第 41 帧, 即新的目标物体刚刚进入视频的画面和该目标物体进入视频一半的画面, 符合内容敏感关键

帧的提取要求。利用本文方法对该段视频流进行检测的结果及利用 VoteNet 对该段视频流进行逐帧检测的结果分别如图 4b 和图 4c 所示。可以看出, 利用本文方法的目标检测结果与经 VoteNet 逐帧检测的结果基本一致, 且本文方法获取的目标检测结果边界框清晰, 具有前后帧之间的检测连贯性。

表 2 所示为场景 1 视频流从第 1~32 帧中部分帧的检测结果, 包含桌子及沙发 2 类物体。其中第 1 帧与第 24 帧为关键帧, 动态展示了左侧物体桌子从出现至展示全部及沙发右侧边缘消失的过程。第 1 行所示为采用本文方法对视频流进行目标检测的结果, 第 2 行所示为采用 VoteNet 逐帧检测对应帧的结果, 可以看出利用本文方法检测得到的目标物体边界框位置准确, 与 VoteNet 逐帧检测结果基本一致, 本文方法具有有效性。表 3 所示为场景 2 视频流从第 1~22 帧中部分帧的检测结果, 视频流中包含 2 把相同的椅子, 第 1 帧及第 16 帧为关键帧, 动态展示了左侧椅子 2 出现及椅子 1 右侧边缘消失的过程。与表 2 类似, 通过与 VoteNet 逐帧检测的结果对比可以看出, 利用本文方法检测得到的结果对目标物体的位置标注准确, 且检测得到的物体语义信息与实际情况保持一致, 可见本文方法的有效性。

图 4 不同方法对场景 3 视频流的目标检测效果及比较

表 2 场景 1 视频流第 1~32 帧检测结果

方法	第 1 帧(关键帧)	第 8 帧	第 16 帧	第 24 帧(关键帧)	第 32 帧
本文					
VoteNet ^[21]					

表 3 场景 2 视频流第 1~22 帧检测结果

方法	第 1 帧(关键帧)	第 8 帧	第 16 帧(关键帧)	第 22 帧
本文
VoteNet ^[21]

6.2.2 本文方法的检测效率

表 4 给出本文方法所需检测时间与 VoteNet^[21] 逐帧检测相同视频流所需时间的对比. 本文方法将室内场景点云数据及 RGB 图像数据作为输入, 而 VoteNet 仅将点云数据作为输入数据. 统计 VoteNet 对视频流进行逐帧检测所需时间时, 首先计算每段视频流检测 10 帧点云数据所需时间, 然后估算平均检测每帧点云所需的耗时, 最后根据该时间计算出逐帧检测整段视频流所需的时间. 如表 4 中第 1 行所示, 视频流 1 共有 63 帧, 采用本文方法进行视频流关键帧提取耗时 11.45s, 检测关键帧点云耗时 4.50s, 对其余帧利用插值检测耗时 20.00s, 视频流目标检测耗时总计 35.95s; 而采用神经网络 VoteNet 对该段视频流进行逐帧检测用时 75.60s. 本文方法所需的检测时间比利用 VoteNet 进行逐帧检测所需时间减少近 40s, 仅为利用 VoteNet 进行逐帧检测用时的一半, 体现出本文方法的高效性.

表 4 2 种方法的检测时间对比 s

视频流	本文				VoteNet ^[21] 逐帧检测总 时间
	关键帧 提取	关键帧 检测	其他帧 检测	合计 时间	
1 (63 帧)	11.45	4.50	20.00	35.95	75.60
2 (60 帧)	11.10	4.20	19.00	34.30	70.00
3 (90 帧)	16.50	7.80	34.00	58.30	103.00

7 结 语

为了克服需经多角度拍摄获取室内场景中所有物体信息并解决场景 3D 目标检测效率较低的难点, 本文以单次拍摄的 RGB-D 视频流作为输入, 利用哈希值算法预先提取视频流的内容敏感关键帧, 再对关键帧数据进行 3D 目标端到端检测, 并利用四元数球面线性插值法确定相邻 2 帧间物体位姿变化关系, 从而估计其余帧的目标检测结果.

与利用神经网络 VoteNet 逐帧检测相比, 本文方法大大缩短了视频流检测耗时; 同时, 本文方法得到的检测结果与利用 VoteNet 逐帧检测得到的检测结果基本一致, 表明本文方法的有效性和高效性.

需要指出的是, 由于在实际应用中寻求针对 RGB-D 视频流数据的实时检测, 未来将考虑结合 GPU 加速和轻量级神经网络缩短目标检测耗时, 实现场景视频流数据的实时目标检测.

参考文献(References):

- [1] Ren Z L, Sudderth E B. Three-dimensional object detection and layout prediction using clouds of oriented gradients[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2016: 1525-1533
- [2] Deng Z, Latecki L J. Amodal detection of 3D objects: Inferring 3D bounding boxes from 2D ones in RGB-depth images[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2017: 5762-5770
- [3] Shen Le, Li Guiqing, Xian Chuhua, *et al.* Door and window detection in 3D point cloud of indoor scenes[J]. Journal of Computer-Aided Design & Computer Graphics, 2019, 31(9): 1494-1501(in Chinese)
(沈乐, 李桂清, 冼楚华, 等. 室内 3D 点云模型的门窗检测[J]. 计算机辅助设计与图形学学报, 2019, 31(9): 1494-1501)
- [4] Zhang Y D, Bai M R, Kohli P, *et al.* DeepContext: context-encoding neural pathways for 3D holistic scene understanding[C] //Proceedings of the IEEE International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2017: 1201-1210
- [5] Song S R, Xiao J X. Sliding shapes for 3D object detection in depth images[C] //Proceedings of the European Conference on Computer Vision. Heidelberg: Springer, 2014: 634-651
- [6] Liu W, Ji R R, Li S Z. Towards 3D object detection with bi-modal deep Boltzmann machines over RGBD imagery[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2015: 3013-3021
- [7] Song S R, Xiao J X. Deep sliding shapes for amodal 3D object

- detection in RGB-D images[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2016: 808-816
- [8] Hou J, Dai A, Nießner M, *et al.* 3D-SIS: 3D semantic instance segmentation of RGB-D scans[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2019: 4416-4425
- [9] Lahoud J, Ghanem B. 2D-driven 3D object detection in RGB-D images[C] //Proceedings of the IEEE International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2017: 4632-4640
- [10] Qi C R, Liu W, Wu C X, *et al.* Frustum PointNets for 3D object detection from RGB-D data[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2018: 918-927
- [11] Qi C R, Su H, Nießner M, *et al.* Volumetric and multi-view CNNs for object classification on 3D data[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2016: 5648-5656
- [12] Zhou Y, Tuzel O. VoxelNet: end-to-end learning for point cloud based 3D object detection[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2018: 4490-4499
- [13] Yang B, Luo W J, Urtasun R. PIXOR: real-time 3D object detection from point clouds[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2018: 7652-7660
- [14] Engelcke M, Rao D, Wang D Z, *et al.* Vote3Deep: fast object detection in 3D point clouds using efficient convolutional neural networks[C] //Proceedings of the IEEE International Conference on Robotics and Automation. Los Alamitos: IEEE Computer Society Press, 2017: 1355-1361
- [15] 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. Los Alamitos: IEEE Computer Society Press, 2017: 77-85
- [16] Qi C R, Yi L, Su H, *et al.* PointNet++: deep hierarchical feature learning on point sets in a metric space[C] //Proceedings of the 31st International Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2017: 5105-5114
- [17] Lang A H, Vora S, Caesar H, *et al.* PointPillars: Fast encoders for object detection from point clouds[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2019: 12689-12697
- [18] Shi S S, Wang X G, Li H S, *et al.* PointRCNN: 3D object proposal generation and detection from point cloud[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2019: 770-779
- [19] Chen Y L, Liu S, Shen X Y, *et al.* Fast point R-CNN[C] //Proceedings of the International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2019: 9774-9783
- [20] Yang Z T, Sun Y N, Liu S, *et al.* STD: sparse-to-dense 3D object detector for point cloud[C] //Proceedings of the International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2019: 1951-1960
- [21] Qi C R, Litany O, He K M, *et al.* Deep Hough voting for 3D object detection in point clouds[C] //Proceedings of the IEEE International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2019: 9276-9285
- [22] Shoemake K. Animating rotation with quaternion curves[C] //Proceedings of International Conference on ACM SIGGRAPH Computer Graphics and Interactive Techniques. New York: ACM Press, 1985: 245-254
- [23] Liu L J, Fan G L. Combined key-frame extraction and object-based video segmentation[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2005, 15(7): 869-884
- [24] Zhang Chan, Gao Xinbo, Ji Hongbing. Video key frame extraction algorithm based on possibilistic C-patterns clustering[J]. Journal of Computer-Aided Design & Computer Graphics, 2005, 17(9): 2040-2045(in Chinese)
(张婵, 高新波, 姬红兵. 视频关键帧提取的可能性 C-模式聚类算法[J]. 计算机辅助设计与图形学学报, 2005, 17(9): 2040-2045)
- [25] Sarohi H K, Khan F U. Image retrieval using perceptual hashing[J]. IOSR Journal of Computer Engineering, 2013, 9(1): 38-40
- [26] Strecha C, Bronstein A M, Bronstein M M, *et al.* LDAHash: improved matching with smaller descriptors[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(1): 66-78
- [27] Song S R, Lichtenberg S P, Xiao J X. SUN RGB-D: a RGB-D scene understanding benchmark suite[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2015: 567-576
- [28] Besl P J, McKay N D. A method for registration of 3-D shapes[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992, 14(2): 239-256