

基于位姿状态的激光雷达点云帧间编码方法

李少良¹ 郑明魁¹ 石元龙¹ 张承琰¹

LI Shaoliang ZHENG Mingkui SHI Yuanlong ZHANG Chengyan

摘要

激光雷达动态获取点云压缩在智能驾驶领域具有重要应用。为了应对点云序列时域冗余问题,本文设计了基于位姿状态的激光雷达点云帧间编码方法。在动态获取点云场景中,点云在三维空间上分布广泛且稀疏。对此,将其映射到二维距离图上。并在此基础上,结合激光雷达位姿关系,提出了一种高效的帧间预测编码方法,以消除时域冗余。由于三维物体遮挡的原因,激光雷达运动时易导致预测距离图产生空洞现象,影响预测编码性能。采用空洞填补方法可提高算法预测精度,并对预测残差进行量化和压缩。实验结果表明,相较于 G-PCC 等编码方法,所提出的方法在编码性能方面表现更为优越。

关键词

激光雷达动态获取点云;点云压缩;距离图;空洞填补

doi: 10.3969/j.issn.1672-9528.2024.04.028

0 引言

点云是三维空间中由点组成的集合,其中每个点包含三个坐标信息 (x, y, z) 以及属性信息^[1]。三维点云在智能驾驶等领域有着广泛的应用,根据点云用途的不同,可分为三类^[2],其中动态获取点云适用于无人驾驶等,表示运动场景。点云应用广泛,但数据庞大,对存储和传输具有较大的挑战,因此设计高效的点云数据压缩方法对于点云相关应用是十分有必要的。

国际运动图像专家组(MPEG)发布了 TMC13 压缩测试模型,也称为基于几何的压缩框架(G-PCC)^[3]。G-PCC 在八叉树编码^[4]基础上进行了改进,借鉴了 Mekuria 等人的基于块的方法^[5],子立方体中的每个点坐标信息都表示为每个节点的二进制占用码,采用基于上下文的算术编码器实现几何压缩。针对激光雷达动态获取的点云序列,G-PCC 采用帧内编码方法进行编码。在连续的点云序列中,相邻帧点云在空间上重复的场景信息较多。若采用帧内编码方法,则算法会重复编码相同的场景信息,浪费编码资源。针对 G-PCC 编码动态点云序列时间复杂度较高、时域冗余较大的问题,MPEG 提出了具有运动补偿和运动估计的 Inter-EM^[6]点云帧间编码方案。InterEM 是基于 TMC13 进行改进的,整个过程可以分为全局运动估计与补偿、局部运动估计与补偿以及帧间预测几何编码。随着计算机算力的不断提高,也逐渐出

现在 3D 空间使用深度学习进行编码的算法。Fan 等人^[7]将点云表示为八叉树结构,分层构建八叉树熵模型,为点云分布建模提供了充分的上下文信息,同时提出了一个残差编码框架来压缩潜在变量。Cui 等人^[8]提出了一种基于八叉树的 Transformer 压缩方法,从点云中提取丰富的特征,可以有效地压缩点云。但深度学习的算法需要耗费大量时间进行点云数据集研究和训练,不符合车路协同领域的实时需求。

为应对基于八叉树的几何编码深层叶子节点需用更多比特表示的问题,一些方法考虑将点云降维,以降低编码的复杂度。Tu 等人^[9]利用雷达扫描获取点云的特点,将激光雷达的原始数据映射成 2D 图像的形式,随后将图像序列转化为视频,采用传统的视频压缩方法对点云数据进行压缩。Yu 等人^[10]将激光雷达的先验信息用于点云投影的预测中,提出了一种正则化点云投影算法,能够更好地利用点云的空间相关性。Wang 等人^[11]在二维上将原始的大规模点云分割成小而紧凑的区域,以实现空间冗余和显著区域分类,从而提升压缩效率。转换到 2D 空间中的点云序列,不论帧内还是帧间,相关性都有了很大的提升,使消除帧间冗余变成可能。

针对上述问题,本文设计了一种基于位姿状态的激光雷达点云帧间编码方法,实现了连续点云序列的编码。点云在进行帧间预测编码时,由于三维物体遮挡等原因,激光雷达运动时易导致预测距离图产生空洞现象,降低预测编码的性能。针对这个问题,本文在帧间编码框架的基础上,采用空洞填补方法以提高预测帧的质量,进而提高编码性能。

1 基于位姿状态的激光雷达点云帧间编码方法

1.1 编码框架简述

在连续的激光雷达点云序列中,物体的运动在空间上和

1. 福州大学物理与信息工程学院 福建福州 350116

[基金项目] 中国福建光电信息科学与技术创新实验室(闽都创新实验室)资助项目(2021ZR151);福州市科技重大项目(2022-ZD-002)

时域上都具有连续性。通常激光雷达的工作频率是在 10 Hz 左右, 相邻两帧点云的时间间隔为 0.1 s。因此, 相邻帧点云的时间间隔比较小。动态获取点云是车辆在运行过程中车载激光雷达采集得到的, 车辆很短的时间内只能前行较小的一段距离, 若一辆车辆以 36 km/h 的时速行驶, 那么在 0.1 s 内会前行 1 m 的距离。因此, 在连续的点云序列中, 相邻两帧点云具有较大范围的重复场景, 存在巨大的时域冗余。本文针对时域冗余的问题, 设计了基于位姿状态的帧间预测编码方案, 如图 1 所示。给定一个连续动态获取点云序列, 将点云区分为两种类型: 参考点云 (I 帧) 和待编码点云 (P 帧)。I 帧点云结合位姿信息预测 P 帧点云, 一帧参考点云预测的待编码点云帧数由参数 n 决定。如果 $n=3$, 那么一帧 I 帧便预测三帧 P 帧, 即该点云序列是以 “IPPPIPPP...” 的形式进行编解码。

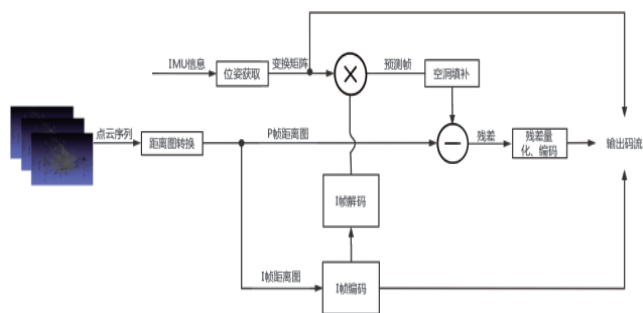


图 1 基于位姿状态的激光雷达点云帧间编码框架

激光雷达点云松散的空间分布不太适合八叉树编码。本文首先将稀疏分布的点云几何信息转化为二维距离图, 以便于后续编码。若测帧与 P 帧越相似, 所需要编码的残差信息越少, 则时域冗余减少得越多。KITTI 数据集包含惯性测量单元采集到的位姿变换信息, 本文称该信息为 IMU 信息。本文利用 IMU 信息计算变换矩阵, 结合 I 帧的信息预测 P 帧点云, 进行帧间预测编码。帧间预测编码时, 由于参考点云存在三维物体遮挡, 导致预测点云较待编码点云相比缺失一部分, 出现空洞区域, 降低了编码性能。针对这个问题, 本文结合三维视频编码技术中的空洞填补方法, 对预测图进行空洞填补。对于 P 帧, 最终只需要编码预测帧与原始帧的预测残差。本文对预测残差做量化编码操作, 进一步压缩了数据。而 I 帧作为参考帧, 信息非常重要, 因此选择 JPEG-LS 无损编码。最终传输的码流由三部分组成: 变换矩阵、预测残差和 I 帧编码信息。本文设计的基于位姿状态的激光雷达点云帧间编码方法能有效地降低连续点云序列的时域冗余, 编码性能较高。

1.2 变换矩阵获取

为了更加准确地预测 P 帧, 降低更多的时域冗余, 需要借助变换矩阵进行点云预测。变换矩阵是寻找两帧点云之

间最大重叠部分的旋转和平移信息, 也就是计算 6 个自由度 (6DoF) 的变换信息, 本文采用的 KITTI 数据集也携带有加速度等信息, 利用该信息能够计算得出相邻帧点云的变换矩阵。IMU 信息是设备传感器直接测量得到的数据, 最接近原始变换的信息, 故计算得到的变换矩阵较准确。

本文采用 I 帧点云的坐标系作为参考, 计算 I 帧与 P 帧之间对应的变换矩阵 M 。变换矩阵是利用 IMU 信息中的平移加速度 ($\hat{a}$) 和旋转速率 ($\hat{\omega}$) 计算得到。根据平移矩阵 $T_{3 \times 1}$ 和旋转矩阵 $R_{3 \times 3}$, 得到变换矩阵 M , 其公式为:

$$M = \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{bmatrix}_{4 \times 4} \quad (1)$$

设定 I 帧点云为 $I_{i,j} \in [1, \dots, N]$, i 是 I 帧点云中的任意一点, N 表示点云的点数。根据变换矩阵 M 和 I 帧点云便能预测得到 P' 帧点云, 设定 p' 是 P' 帧点云中的任意一点, 其公式为:

$$p'_{4 \times 1} = M \times i_{4 \times 1} = \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{bmatrix}_{4 \times 4} \times i_{4 \times 1} \quad (2)$$

经过上述步骤实现了 P 帧点云的预测。预测点云的重合部分是与 I 帧相同的点云, I 帧是已编码的点云, 故重合部分可以不用编码, 只需编码原始点云 P 与预测点云 P' 之间的差异部分。帧间预测编码不仅能减少编码的点云数据量, 而且可以大幅度地减少时域冗余, 有利于无人驾驶领域中点云数据的传输。

1.3 点云空洞填补算法

参考点云通过变换矩阵预测待编码点云, 结合帧间预测编码减少编码的点云数据。但是参考点云在三维空间存在遮挡问题, 导致点云产生空洞区域。帧间预测编码的目标是利用参考点云预测待编码点云, 但是缺失部分不参与预测变换, 故预测点云会缺失部分点云, 即产生空洞区域。

视频编码算法中的空洞填补方法包括帧内修复技术和多视点插值法, 其中帧内修复技术可分为基于深度学习修复、基于纹理修复和结构修复三种。基于深度学习修复的方法需要消耗大量时间, 不能满足无人驾驶领域高实时性的要求。距离图块与块之间的连续性不强, 不适合采用基于纹理修复的方法。距离图像内的像素值具有一定的连续性。因此, 本文选择伍能举等人^[12]提出的空洞填补算法, 该算法也是属于基于结构修复的方法, 对预测图进行空洞填补, 可以有效地改善空洞区域现象。

空洞区域的像素点是因为物体遮挡产生, 因此该位置的像素点与邻域周围的像素点有较大的关联性。本文选择空洞像素点周围的 10 个邻域像素点作为参考点, 根据像素点的远近关系设置不同的权重。如图 2 所示, 设定中间像素 B_3 为空洞像素, 将周围的 8 个近端邻域像素 $A_2 \sim A_4$ 、 B_2 、 B_4 、

$C_3 \sim C_4$ 和两个远端邻域像素 B_1 、 B_5 作为参考像素, 通过对不同权重的参考像素加权求和得到空洞像素值。8 个邻域像素离空洞像素最近, 因此相关性最高, 设置权值较大。距离图行内的像素值相关性较高, 因此多增加了 B_1 、 B_5 两个参考像素点。两个像素点距离相对较远, 称为远端邻域像素, 设置权值较小。

A_1	A_2	A_3	A_4	A_5
B_1	B_2	B_3	B_4	B_5
C_1	C_2	C_3	C_4	C_5

图 2 空洞像素与邻域填补像素的关系

10 个参考像素点可能还存在空洞点, 并且通常不会存在近端邻域像素和远端邻域像素全都为空洞点。因此, 本文针对参考像素点的空洞点数量进行讨论, 根据不同情况, 设定不同的计算方法。设定 8 个近端邻域像素的空洞点数为 m , 2 个远端邻域像素的空洞点数为 n , 近端邻域像素点的权重为 ω_1 , 远端邻域像素点的权重为 ω_2 。

(1) 若 $m=0$ 且 $n=0$, 则空洞填补公式为:

$$B_3 = \omega_1(A_2 + A_3 + A_4 + B_2 + B_4 + C_2 + C_3 + C_4) + \omega_2(B_1 + B_5) \quad (3)$$

(2) 若 $0 < m < 8$ 且 $0 < n < 2$, 则空洞填补公式为:

$$B_3 = \frac{8\omega_1}{8-m}(A_2 + A_3 + A_4 + B_2 + B_4 + C_2 + C_3 + C_4) + \frac{2\omega_2}{2-n}(B_1 + B_5) \quad (4)$$

(3) 若 $m=8$ 且 $0 < n < 2$, 则空洞填补公式为:

$$B_3 = \frac{10\omega_2}{2-n}(B_1 + B_5) \quad (5)$$

(4) 若 $0 < m < 8$ 且 $n=2$, 则空洞填补公式为:

$$B_3 = \frac{10\omega_1}{8-m}(A_2 + A_3 + A_4 + B_2 + B_4 + C_2 + C_3 + C_4) \quad (6)$$

在上述方法中, 关键点是如何设置两个权重值 ω_1 和 ω_2 。由于 ω_1 对应近端邻域像素的权重, 故值较大, 而 ω_2 对应远端邻域像素, 值较小。其中两个权重值需满足公式 (7)。由于 $\omega_1 > \omega_2$, 故设 $0.100 < \omega_1 < 0.125$, $0 < \omega_2 < 0.1$ 。本文通过实验测试, 最终设定 $\omega_1 = 0.105$, $\omega_2 = 0.08$ 。

$$8\omega_1 + 2\omega_2 = 1 \quad (7)$$

1.4 预测残差的量化与编码

帧间预测编码分析了参考点云 I 帧和原始 P 帧的联系, 利用 I 帧的已知信息预测 P 帧, 减少 P 帧的编码数据, P 帧最终只需编码预测残差, 即原始 P 帧与预测帧 P' 的差异。残差先进行残差量化, 再利用 JPEG-LS 无损压缩残差。根据残差绝对值的大小, 设置两个阈值 $th1$ 和 $th2$, 将残差分为三部分, 不同的数值区间设置不同的量化步长。量化后, 原始数据范围从 $-13\ 000 \sim 13\ 000$ cm 映射为 $-4000/\alpha \sim 4000\alpha$ cm 的连续整数。参数 α 是控制量化的程度, 并且可用于调整压缩比和损耗。其中 $\alpha = (3/4)^i$, $i \in [0, 1, \dots, 10]$ 。残差的局部相关性较强,

因此采用 JPEG-LS 无损地压缩量化后的残差数据。

2 实验结果

2.1 实验条件和评价指标

本次实验是在 Windows10 系统上完成, 电脑配置是 AMD Ryzen 75800U @1.90 GHZ, 16 GB 运行内存。实验选用了 KITTI 数据集的不同场景, 包括校园、城市、居民区和道路四种场景。每个场景各选择一个序列, 每个序列选取 100 帧进行测试。本文采用场景名称代替各自的序列名。实验是以 “IPPPIPPI...” 的形式进行点云序列的编解码。

本文只考虑点云的几何属性压缩, 采用码率作为压缩性能的评价标准。针对 KITTI 数据集, 点云的解码质量评价选择 MPEG 提供的基于点对点几何的峰值信噪比 (PSNR) [13]。

$$PSNR = 10 \times \log_{10} \left(\frac{3P^2}{MSE(p2p)} \right) \quad (8)$$

式中: P 为每个序列点云内部点与点之间距离的最大值, $MSE(P2P)$ 为重建点云到原始点云的均方误差。

2.2 本文算法与 G-PCC 的压缩性能对比

图 3 展示了在 KITTI 数据集上, 本文所提出的基于位姿状态的激光雷达点云帧间编码方法与 G-PCC 以及 Inter-EM 的 R-D 曲线比较结果。

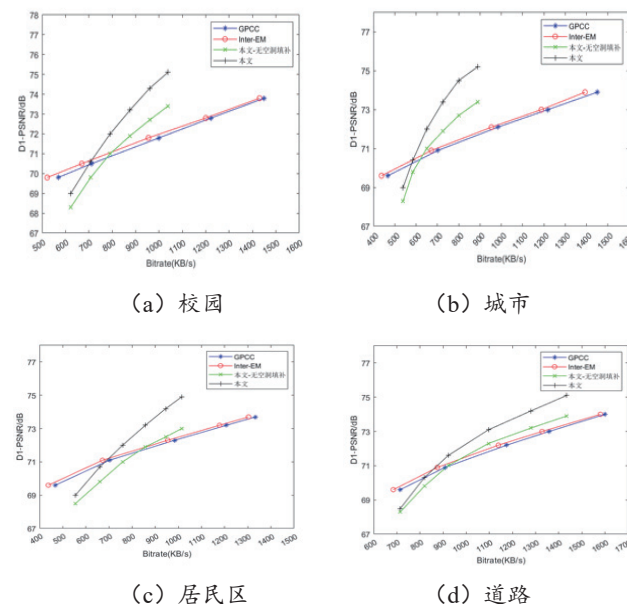


图 3 KITTI 数据集的 R-D 曲线图

从图 3 中可以看出, 在四个场景中, 本文方法在码率大于 800 kb/s 时, 在相同的码率下, 具有最高的 D1-PSNR, 相比于 MPEG G-PCC 和 Inter-EM 的方法, 性能有较大的提升, 且优于本文未加空洞填补的编码方法。而在码率较低时, G-PCC 和 Inter-EM 的编码性能略胜一筹, 这是由不同的量化方式导致的。G-PCC 的量化过程是通过合并周围点的坐标来

提高压缩率,在低码率时保持点云整体结构效果较好。相比之下,本文使用的量化方式在低码率时会导致距离值较大的点与密集区域的点丢失较严重,导致在重构后产生较大的误差。

本文算法对城市场景的编码效果相对较好。因为该场景数据结构单一,包含的动态物体较少,所以预测准确度更高,降低了更多的时域冗余,能够实现更优的编码性能。另外,道路场景中本文方法的编码性能在四个场景中较差,主要原因是道路两边都是树木丛林,导致点云数据分布杂乱无序,这会使得在点云的二维表示中边界不清晰,导致编码性能差于其他序列。

动态获取的点云具有扫描范围较大、分布不均的特点,靠近激光雷达的区域点云较为密集,远离激光雷达的区域点云则较为稀疏。在使用八叉树编码时会产生许多空白的分块,既需要用比特信息记录是否有点的存在,又需要额外比特来记录点的存在情况,因此八叉树的方法很难在低误差的情况下取得较低的码率。本文方法可以在低误差的情况下实现高于 G-PCC 与 Inter-EM 的压缩率,有效减少空间和时间冗余,因此可以实现更好的 R-D 性能。另外,在自动驾驶领域中所需的即时定位与地图构建(simultaneous localization and mapping, SLAM)任务中可容忍的误差通常在 0.1 m 之内,而在这区间内本文算法拥有最优的编码表现。

本文为解决空洞现象设计了空洞填补的方法,使得预测结果更加准确,重建质量有所提高。从图 3 可以看出,在相同的码率下,本文的方法比无空洞填补的方案重建质量更高,表明了空洞填补方法的有效性。

3 结语

针对连续的点云序列中存在的时域冗余问题,本文借助相邻帧点云的位姿关系,设计了一种基于位姿状态的激光雷达点云帧间编码方法。本文利用 IMU 信息计算参考点云与待编码点云之间的变换矩阵,结合参考点云的信息预测待编码点云。针对预测图的空洞现象,本文进一步使用一种空洞填补的方法,提高了算法的预测精度和编码性能。实验结果显示,本文设计的方法具有较优的编码性能,并且编解码速度较快。

参考文献:

- [1] 李厚强,李礼,李竹.点云编码综述[J].中兴通讯技术,2021,27(1):5-9.
- [2] SCGWARZ S, PREDA MARIUS, BARONCINI V, et al. Emerging MPEG standards for point cloud compression[J]. IEEE journal on emerging and selected topics in circuits and systems, 2019, 9(1):133-148.
- [3] Point Cloud Compression Category 13 Reference Software, TMC13-v14.0[EB/OL].htx.int-evry.fr/software/MPEG/PCC/

TM/mpeg-pcc-tmc13/tree/release-v14.0.

- [4] ELSEBERG J, BORRMANN D, NUECHTER A. One billion points in the cloud – an octree for efficient processing of 3D laser scans[J]. ISPRS journal of photogrammetry and remote sensing, 2013, 76:76-88.
- [5] MEKURIA R, BLOM K, CESAR P. Design, implementation, and evaluation of a point cloud codec for tele-immersive video[J]. IEEE transactions on circuits and systems for video technology, 2017, 27(4):828-842.
- [6] MPEG-3DG. Inter-prediction Exploration Model (Inter-EM) v4.0, document N00241[R]. Online: ISO/IEC JTC1/SC29/WG7 MPEG, 2021.
- [7] FAN T, GAO L, XU Y, et al. Multiscale latent-guided entropy model for LiDAR point cloud compression[J]. IEEE transactions on circuits and systems for video technology, 2023, 33:7857-7869.
- [8] CUI M, LONG J, FENG M, et al. OctFormer: efficient octree-based transformer for point cloud compression with local enhancement[J]. Proceedings of the AAAI conference on artificial intelligence, 2023, 37(1):470-478.
- [9] TU C, TAKEUCHI E, MIYAJIMA C, et al. Compressing continuous point cloud data using image compression methods[C]// IEEE International Conference on Intelligent Transportation Systems. Piscataway: IEEE, 2016:1712-1719.
- [10] YU Y, ZHANG W, LI G, et al. A regularized projection-based geometry compression scheme for LiDAR point cloud[J]. IEEE transactions on circuits and systems for video technology, 2022, 1:99.
- [11] WANG S, JIAO J, CAI P, et al. R-pcc: a baseline for range image-based point cloud compression[C]// 2022 International Conference on Robotics and Automation (ICRA). Piscataway: IEEE, 2022:10055-10061.
- [12] 伍能举,徐栋,杨依忠.图像空洞填补算法及其 FPGA 硬件实现[J].合肥工业大学学报(自然科学版),2021,44(9):1216-1220.
- [13] MPEG-3DG. Common test conditions for point cloud compression, document N19324[R]. Alpbach: ISO/IEC JTC1/SC29/WG11 MPEG, 2020.

【作者简介】

李少良(1999—),男,福建莆田人,硕士研究生,研究方向:点云编码。

郑明魁(1976—),男,福建福州人,副教授,博士,研究方向:点云编码与视频编码。

(收稿日期:2024-02-09)