

基于深度学习的平面位姿估计算法

许 鲜, 陈 宁, 陈玉鹏

(集美大学海洋装备与机械工程学院, 福建 厦门 361021)

[摘要] 为了在未知物体三维模型的情况下使用深度学习进行平面位姿估计, 采用编码器-解码器网络, 从单个 RGB 图像中检测平面实例分割及法线信息, 并利用这些信息进行位姿解算, 获得每个平面的实时位姿。实验结果显示, 平面召回率为 0.625, 平面法线召回率为 0.414, 实时性为 18.5 f/s, 验证了算法的可行性。

[关键词] 位姿估计; 平面检测; 实例分割; 法线估计

[中图分类号] TP 391

Research on Plane Pose Estimation Algorithm Based on Deep Learning

XU Xian, CHEN Ning, CHEN Yupeng

(School of Marine Equipment and Mechanical Engineering, Jimei University, Xiamen 361021, China)

Abstract: Plane pose estimation is widely used in robotics, augmented reality and other fields, but traditional visual pose estimation algorithms need to be added specific markers to the target, which is less robust and cannot be applied to any scene; pose estimation methods based on deep learning can effectively solve the above-mentioned problems, but the existing method requires a three-dimensional model of a known object. In order to use deep learning for plane pose estimation in the case of unknown object three-dimensional model, the encoder decoder network was used to detect plane instance segmentation and normal information from a single RGB image, which were then employed to perform pose calculation for obtaining each Real-time pose of each plane. The test results show that the plane recall rate is 0.625, the plane normal recall rate is 0.414, and the real-time performance is 18.5 f/s, which verifies the feasibility of the algorithm.

Keywords: pose estimation; plane detection; instance segmentation; normal estimation

0 引言

随着计算机视觉技术的发展, 平面位姿估计在机器人技术、增强现实和虚拟现实、船舶海上作业等领域的应用越来越广泛^[1-5]。尽管在过去几十年中, 已经出现了很多平面位姿估计的方法, 但是在快速准确和适应性方面还有很大的提升空间, 尤其是对缺少纹理表面的平面目标而言。

平面位姿估计的目的是检测平面并估计其平移和旋转。传统视觉平面位姿估计通常分为两类。

1) 基于特征的方法^[6-7], 计算一组三维点和二维投影之间的对应关系, 得到相机与平面之间的相对

[收稿日期] 2021-01-12

[基金项目] 集美大学国家基金培育计划项目 (ZP2020045)

[作者简介] 许鲜 (1992—), 女, 硕士, 从事机器视觉研究。通信作者: 陈宁 (1972—), 男, 副教授, 博士, 从事机器视觉、工业机器人研究。

<http://xuebaobangong.jmu.edu.cn/zkb>

位置和方向, 比如通过在平面物体上添加特定标志物, 使用相机对特定标志物进行定位和检测, 实现检测非接触式的平面位姿检测, 其中比较成熟的方法有 Aruco^[8], Apriltag^[9]等。然而这些方法精度易受场景限制, 依赖于特定的特征, 这些特征对图像变化和遮挡对不确定因素等的鲁棒性较差。2) 基于模板匹配的直接方法^[10], 将姿态估计问题简化为模板匹配问题, 通过优化参数来估计姿态, 观察目标图像的刚性变换, 但是这些模板匹配方法的主要缺点是, 当仿射或者单应变换空间与姿态空间之间未对准时, 容易产生额外的附加姿态误差。

随着深度学习对场景理解能力的增强, 基于深度学习的位姿估计方法实现了精度、实时性和鲁棒性更好的位姿估计结果。Rad 等^[11]提出 BB8 网络, 从单张 RGB 图像预测目标 3D 边框的 2D 投影, 并使用 PnP 算法解算出目标 3D 位姿; Kehl 等^[12]提出了 SSD-6D 网络, 使用合成数据集进行训练, 将 SSD 用于三维实例检测和 6D 位姿估计, 获得更好的实时性; Xiang 等^[13]提出 PoseCNN 网络, 可以在杂乱场景中实现精确的物体 6D 姿态估计; Sida 等^[14]提出 PVNet 网络, 从单张 RGB 图像预测目标的语义分割和物体关键点的向量场, 使用随机投票算法计算出目标的关键点, 并使用 PnP 算法解算出目标 6D 位姿; Chen 等^[15]提出 DenseFusion 网络, 在像素级别嵌入和融合颜色和深度信息, 以估计 RGB-D 图像中已知物体的 6D 位姿。然而现有大多数基于深度学习的位姿估计方法需要已知目标的三维模型, 且网络的泛化性较差, 很难实现未知模型的平面位姿估计。

最近的一些使用深度学习的平面重建方法可以从单幅 RGB 图像中检测平面实例分割、深度及法线信息。Liu 等^[16]提出了一种端到端的深度神经网络 PlaneNet, 从单幅 RGB 图像中推算平面参数及其对应的平面分割掩码; Yang 等^[17]提出一种针对室外场景的非监督学习方法 PlaneRecover, 预测平面实例分割及其平面参数。但 PlaneNet 和 PlaneRecover 两者都需要事先给出单个图像中的最大平面数, 并且跨域的泛化能力较差。为了解决上述问题, Liu 等^[18]提出基于 MaskRCNN 的平面重建网络 PlaneRCNN, 可以从单个 RGB 图像中检测出小的平面区域并重建分段平面深度图; Yu 等^[19]提出了平面重建网络 PlanarReconstruction, 训练卷积神经网络得到每像素的嵌入映射, 再使用聚类算法对嵌入向量分组得到任意数量的平面实例, 通过平面参数预测分支得到平面参数。受这些工作的启发, 本文使用基于深度学习的方法进行平面分割和法线估计, 并经过位姿解算实现单个 RGB 图像的平面位姿估计, 解决传统视觉位姿估计需要添加特定标志物和现有深度学习位姿估计算法需要已知物体三维模型的问题。

1 平面位姿估计试验系统

1.1 平面位姿估计试验系统硬件设计

基于实验需求搭建一套平面位姿估计实验系统, 如图 1 所示, 包括六自由度运动模拟系统和视觉检测系统。其中: 六自由度运动模拟系统通过输入设定的运动方程模拟需要的六自由度运动; 视觉检测系统通过相机可以获得其安装视野下的彩色 RGB 图像, 通过单目视觉位姿检测算法检测出目标位姿, 可对位姿估计效果进行验证。

1.2 基于深度学习的单图像位姿估计算法

本文的目标是从单个 RGB 图像推断平面实例和平面位姿, 算法框架如图 2 所示。在第一个阶段, 使用多分枝的编码器-解码器网络共同预测单图像平面实例和法线。基于 ScanNet^[20]数据集生成平面实例数据集, 并进行训练, 输入单张 RGB 图像, 采用编码器-解码器网络获得每个像素的关联嵌入向量, 使用均值漂移算法对嵌入向量进行聚类, 得



图 1 平面位姿估计实验系统

Fig.1 Plane pose estimation test system

到平面实例分割; 共享编码特征进行平面参数分支解码, 得到每个像素的法线信息。第二个阶段, 对获得的平面掩码和法线进行位姿解算, 获得平面的位姿, 对实例分割和法线进行位姿解算, 得到每个平面实例的位姿信息, 最后通过实验对基于深度学习的平面位姿估计算法的可行性进行验证。

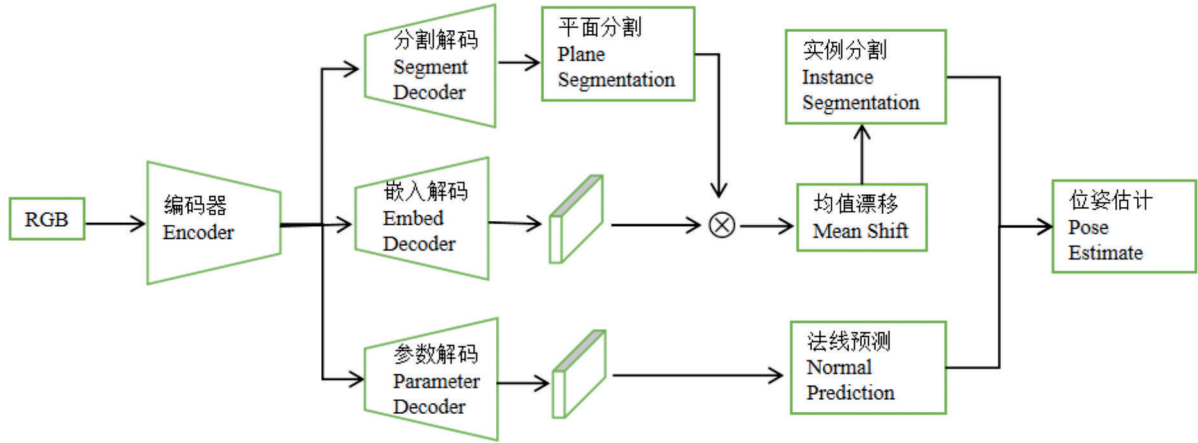


图2 基于深度学习的单图像平面位姿估计算法流程

Fig.2 Deep learning based estimation algorithm flowchart for single image plane pose

2 原理及方法

2.1 数据集构建

用于平面位姿估计的深度学习算法需要大量数据集进行训练, 同时平面区域在图像中可能具有复杂的边界, 手动标注这些区域非常耗时, 并且不能保证对数据集提供精确的位姿标注。为了避免繁琐的手动标注过程, 可以使用模型拟合算法, 将现有 RGB-D 数据集集中的每像素深度图转换为平面实例标注, 使用 PlaneNet 生成的 ScanNet 数据集^[20]训练并评估该方法, 通过将平面拟合到 ScanNet 的合并网格并将其投影回各个框架, 可以获取真实值。生成过程还包含来自 ScanNet 的语义注释, 结果数据集包含 50 000 个训练图像和 760 张测试图像, 分辨率为 256 px × 192 px, 数据集含有深度信息, 并具有平面实例分割标记及平面法线信息。

2.2 算法网络结构

平面位姿估计算法采用一种编码器-解码器的网络结构, 如图3所示。

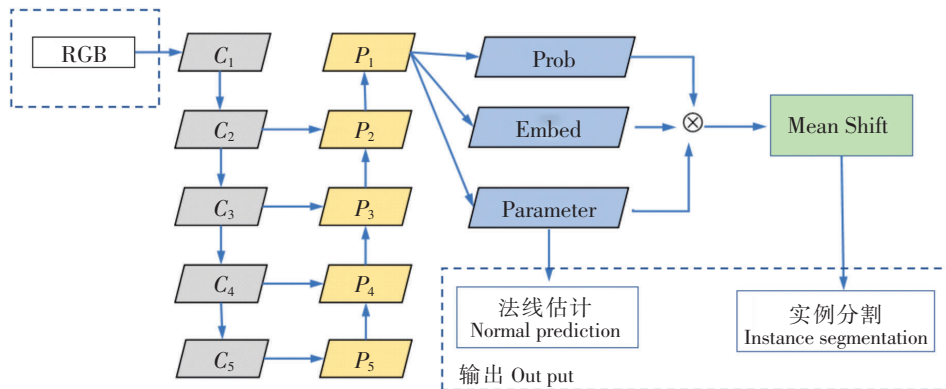


图3 平面位姿估计算法的网络结构

Fig.3 Network structure of plane pose estimation algorithm

其中: C 为卷积层; P 为池化层。输入单张 RGB 图像, 编码器使用 ResNet101-FPN 网络, 解码器分别为平面/非平面语义分割, 关联嵌入向量和平面参数解码, 三个分支共享相同的高级特征映射。

$C_1 \sim C_5$ 采用自下向上的通路, 即普通卷积特征自底向上逐层浓缩表达特征的过程。此过程低层表达边缘等较浅层次的图片信息特征, 较高层表达较深层次的图片特征, 如物体轮廓、类别等。 $P_1 \sim P_5$ 采用自上至下的通路, 处理每一层信息时会参考上一层的高维信息作为输入。上层特征输出的特征图尺寸小但感受野更大, 高维信息经实验证实能够对后续的目标检测、物体分类等任务发挥关键作用。解码器网络采用 1×1 卷积, 可有效降低中间层次的通道数量, 使输出不同维度的各个特征图有相同的通道数量。平面参数解码分支得到法线估计结果, 结合平面/非平面语义分割和平面参数解码, 使用均值漂移聚类算法对嵌入向量解码进行聚类, 得到平面实例分割结果。深度学习算法的网络输出为法线估计和实例分割。

2.3 损失函数

对平面/非平面分割解码器, 由于平面/非平面分割区域在人为环境下不平衡, 因此采用式 (1) 所示的平衡交叉熵损失函数。

$$L_S = -(1 - \omega) \sum_{i \in F} \lg p_i - \omega \sum_{i \in B} \lg(1 - p_i)。(1)$$

其中: F 和 B 分别是前景 (平面) 和后景 (非平面) 像素点集; p_i 是第 i 个像素属于前景的概率; ω 是前景和后景像素点数的比值。

平面嵌入解码器采用关联嵌入的思想, 为每个像素预测一个嵌入向量, 采用式 (2) 中的判别损失函数, 其中“拉”损失将每个嵌入拉到对应实例的平均嵌入, 而“推”损失将实例中心彼此推开, 使同一平面实例中的像素比不同平面中的像素更接近。由于图像中平面实例的数量不是先验已知的, 采用均值漂移聚类算法将得到的嵌入向量分组为平面实例。标准均值漂移算法在每次迭代中对所有像素嵌入向量进行两两距离计算, 计算复杂度较大。为了解决实时性, PlanarReconstruction 在聚类开始时过滤掉局部密度较低的锚点, 并提出了均值漂移聚类的一个快速版本, 只移动嵌入空间中的少量锚点, 并将每个像素分配给最近的锚点。聚类算法收敛后, 形成聚类, 每个聚类对应一个平面实例, 聚类中心是这个聚类中锚点的平均值。

$$L_E = L_{\text{pull}} + L_{\text{push}}。(2)$$

其中:

$$L_{\text{pull}} = (1/C) \sum_{C=1}^C (1/N_C) \sum_{i=1}^{N_C} \max(\|\mu_c - x_i\| - \delta_v, 0), (3)$$

$$L_{\text{push}} = [1/(C(C-1))] \sum_{C_i=1}^C \sum_{\substack{C_j=1 \\ C_i \neq C_j}}^C \max(\delta_d - \|\mu_{cA} - \mu_{cB}\|, 0)。(4)$$

其中: C 是标准的聚类 (平面) 的数量; N_C 是每个聚类中元素的数量; x_i 是像素嵌入; μ_c 是聚类 C 的平均嵌入; δ_v 和 δ_d 分别是拉损失和推损失边界。如果嵌入的像素是容易分离的, 即类内实例的距离大于 δ_d 或嵌入向量及其中心之间的距离小于 δ_v , 惩罚是零, 否则惩罚将大幅增加。

平面参数分支推断图像每个像素点的平面参数, 即点的法向量, 使用式 (5) 中的 L_1 损失即 L_1 范数损失函数, 也被称为最小绝对值偏差, 是目标值与估计的绝对差值的均值。

$$L_1 = (1/N) \sum_{i=1}^N \|n_i - n_i^*\|。(5)$$

其中: n_i 是预测的平面参数; n_i^* 是第 i 个像素的标准平面参数。

已知同一平面上单位法线相等且满足平面公式 (6)。

$$\mathbf{n}^T \cdot \mathbf{Q} = 1。(6)$$

其中: \mathbf{n} 为平面法线, 为单位法线矢量 (\mathbf{M}) 除以平面偏距 (d), 即 $\mathbf{n} = \mathbf{M}/d$; \mathbf{Q} 为平面像素点 3 维坐标。

由此先验知识, 使用如式 (7) 所示损失函数, 使同一平面实例上法线相等。

$$L_{\text{IP}} = [1/(N\tilde{C})] \left(\sum_{j=1}^C \sum_{i=1}^N S_{ij} \cdot \|n_j^T Q_i - 1\| \right). \quad (7)$$

其中: Q_i 是像素 i 结合真实深度值得到的 3 维点坐标; \tilde{C} 为形成的聚类; S_{ij} 为关联像素嵌入和聚类的 soft 分配; n_j 为由像素级参数聚合成的实例级参数。

2.4 位姿解算

与 PoseCNN 类似, 首先定义垂直平面向上为平面坐标系 z 轴, 然后结合平面法线信息解算得到物体与摄像机的距离, 从而估计物体的三维位移, 最终获得平面的姿态信息。由 2.2 节可知, 基于深度学习的方法已经求得平面实例分割及其对应的法线, 接下来进行平面位姿解算, 获得平面的实时位姿, 如图 4 所示。其中: $\{b_c\}$ 为相机坐标系; $\{b_p\}$ 为平面坐标系, 选平面质心为平面坐标系原点; T_p^c 为相机坐标系和平面坐标系的转换矩阵。求得平面的 2D 质心 c 为 (u_0, v_0) , 对应相机坐标系下坐标为 (x_0, y_0, z_0) 。以 c 为原点, 垂直平面向上为 z 轴建立平面坐标系。可以设置平面掩码最大内径方向为平面坐标系的第二轴, 又乘得到第三轴的方向来定义平面坐标系。也可以根据需要设置绕 z 轴的转角 γ , 此处忽略平面绕 z 轴的转动, 即设绕 z 轴转动角为 0° , 则相机的外参矩阵如式 (8) 所示。

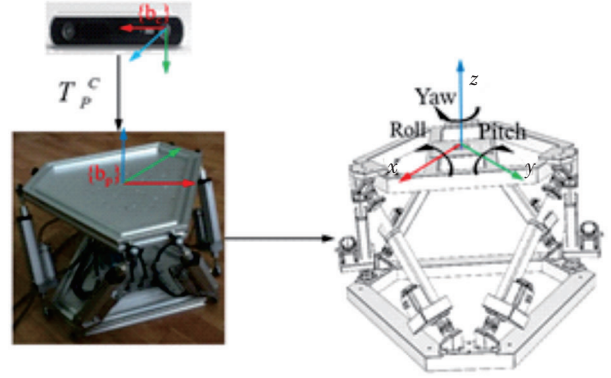


图 4 平面位姿解算示意图

Fig.4 Schematic diagram of plane pose solution

$$M_2 = \begin{bmatrix} \cos \beta & \sin \beta \sin \alpha & -\sin \beta \cos \alpha & t_x \\ 0 & \cos \alpha & \sin \alpha & t_y \\ -\sin \beta & -\sin \alpha \cos \beta & \cos \alpha \cos \beta & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (8)$$

其中: α 、 β 分别为绕 x 、 y 轴的转动角; $[t_x, t_y, t_z]^T$ 为平移矢量。

在相机坐标系中, 平面公式满足式 (9)。

$$n_x \cdot x_c + n_y \cdot y_c + n_z \cdot z_c = 1. \quad (9)$$

在理想针孔模型下, 摄像机的成像模型齐次坐标投影关系为:

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = M_1 \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = M_1 M_2 \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}. \quad (10)$$

其中: f_x, f_y 为焦距; c_x, c_y 为图像中心与像素坐标系原点像素偏差; R, T 分别为旋转矩阵和平移矩阵, M_1, M_2 分别为摄像头的内参矩阵和外参矩阵; $[u \ v]$ 为像素坐标系下坐标点; $[x_c \ y_c \ z_c]$ 为相机坐标系下坐标点; $[x_w \ y_w \ z_w]$ 为世界坐标系下坐标点。

由式 (10) 和式 (11) 可得相机坐标系下坐标点为 $[x_c \ y_c \ z_c]$ 。

$$\begin{cases} x_c = z_c \cdot (u - c_x) / f_x; \\ y_c = z_c \cdot (v - c_y) / f_y; \\ z_c = 1 / (M \cdot q_c). \end{cases} \quad (11)$$

$$\text{由已知条件可得} \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} = M_2 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} x_0 + n_x \\ y_0 + n_y \\ z_0 + n_z \\ 1 \end{bmatrix} = M_2 \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}。$$

联立两式解得: $t_x = x_0$; $t_y = y_0$; $t_z = z_0$; $\alpha = \arcsin(n_y)$; $\beta = \arctan(-n_x/n_z)$ 。代入式 (8) 可解得 M_2 , 得到平面的位姿。

3 实验

首先对平面重建网络进行复现和对比, 在此基础上选择泛化性更好的平面重建网络架构。然后使用选定的平面重建网络进行平面实例分割和法线估计, 在 ScanNet 数据集上对平面实例分割和法线估计的结果进行量化分析和评估。最后在 ScanNet 数据集上对平面位姿估计的结果进行对比分析, 并在 NYU 数据集和平面位姿估计实验系统上进行验证。使用 PyTorch 来实现本文的模型。服务器平台配置: Intel (R) Core (TM) i7 - 7700K CPU@4.2GHz 处理器, 16 GB 内存, 1T 盘, 8GB Geforce GTX 1070 GPU。

3.1 平面实例分割结果

平面重建网络 PlaneRCNN 和 PlanarReconstruction 均使用 ScanNet 数据集进行训练, 并使用 NYU 数据集和用户图片对两者的训练结果进行测试, 测后对比结果如图 5 所示。由图 5 可见, PlaneRCNN 对小平面的检测结果较好, 但会误识别多余的小平面。相比之下, PlanarReconstruction 对小平面的检测结果较差, 但能够检测出较完整的平面, 该网络对平面实例的检测具有更好的泛化性。因此本文尝试在平面重建网络 PlanarReconstruction 基础上进行平面位姿估计。

图 6 为在 ScanNet 数据集上的平面实例分割结果, 与真实平面实例分割相比, 基本能得到数量和形状一致的平面分割结果。

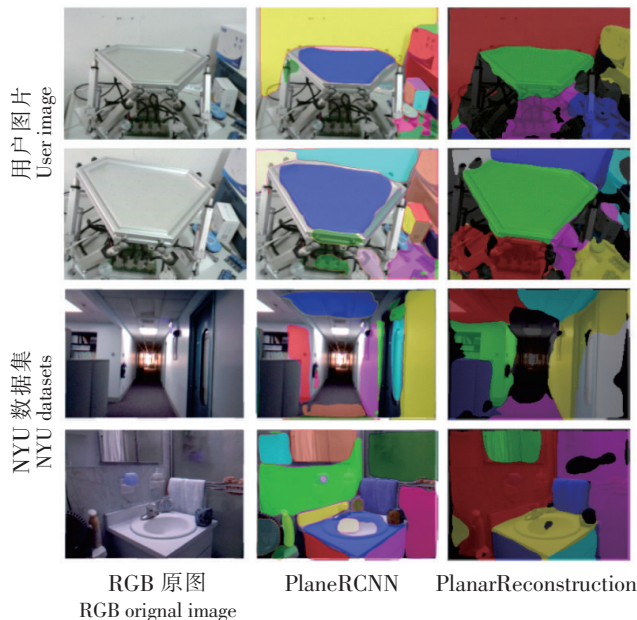


图 5 PlaneRCNN 和 PlanarReconstruction 在 NYU 数据集和用户图片的平面检测结果对比
Fig.5 Comparison of plane detection results of PlaneRCNN and PlanarReconstruction in NYU dataset and user picture

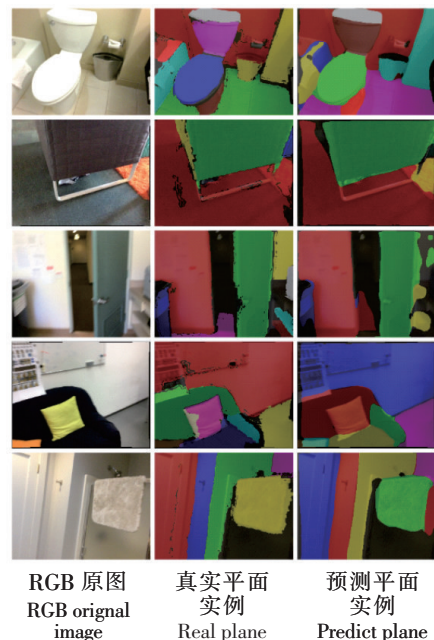


图 6 在 ScanNet 数据集上的平面实例分割结果
Fig.6 Plane instance segmentation results on ScanNet dataset

平面召回率是正确预测标准平面的百分比, 像素召回率是正确预测平面内像素的百分比, 因此, 本文使用平面和像素召回率作为评价指标。标准平面预测指标: i) 其中一个预测平面交并比 (IOU, Intersection over Union) 得分大于 0.5; ii) 重叠区域平均深度差小于阈值, 该阈值从 0.05 m 到 0.60 m 不等, 增量为 0.05 m, 则认为标准平面预测正确。实验中, 通过该指标进行判定得到算法的平面和像素召回率, 图 7 为在 ScanNet 测试数据集上的平面召回率 (蓝色) 和像素召回率 (黑色), 可以看出测试集上平面召回率平均值为 0.625, 像素召回率平均值为 0.773。

3.2 平面位姿检测结果

法线估计的准确性是平面位姿估计准确率的关键因素。根据研究, 平面法线召回率是正确预测标准平面法线的百分比, 像素召回率是正确预测法线的平面内像素的百分比。其中平面法线预测指标为: i) 其中一个预测平面交并比 (IOU) 得分大于 0.5; ii) 预测法线与真实法线夹角小于阈值, 该阈值从 0 到 30° 不等, 增量为 2.5°, 则认为平面法线预测正确。实验中, 通过该指标进行判定, 得到算法的平面和像素法线召回率。

如图 8 为在 ScanNet 测试数据集上的平面法线召回率 (蓝色) 和像素法线召回率 (黑色), 可以看出平面法线召回率平均值为 0.414, 像素法线召回率为 0.328。

表 1 为在 ScanNet 测试数据集上的深度预测准确率, 其均方根误差为 0.039 m。由图 8 和表 1 可以看出, 相比平面实例分割效果, 基于单图像的平面法线预测精度还有较大提升空间。

表 1 在 ScanNet 测试集上的深度预测准确率

Tab. 1 Accuracy of depth prediction on ScanNet test set

Abs_Rel	Sq_Rel	RMSE	RMSE_log
0.0963	10.6402	0.0393	0.1239

使用位姿解算算法对平面实例分割和法线估计进行位姿解算, 在 ScanNet 数据集上的平面位姿估计结果如图 9 所示。

其中: 第 2 列为使用深度学习预测的平面实例分割和法线估计解算的平面位姿, 第 3 列为使用数据集中真实平面实例分割和法线信息解算的平面位姿。图 9 中, 每个平面实例中心处均标记解算得到的平面坐标系, 红绿蓝颜色的坐标轴分别为 x , y , z 轴。本文采用的平面位姿估计算法在简单场景下的位姿估计

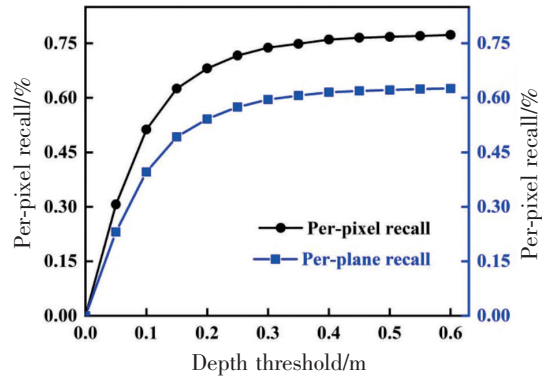


图 7 在 ScanNet 测试集上的平面召回率和像素召回率

Fig.7 Plane recall rate and pixel recall rate on ScanNet test set

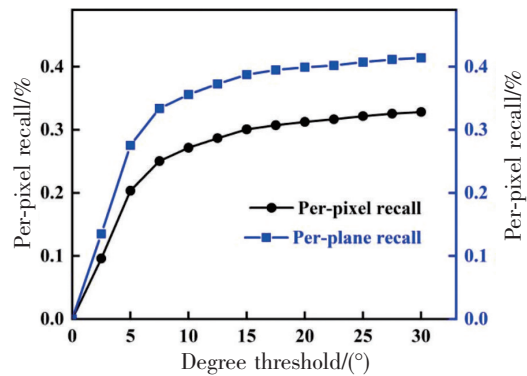
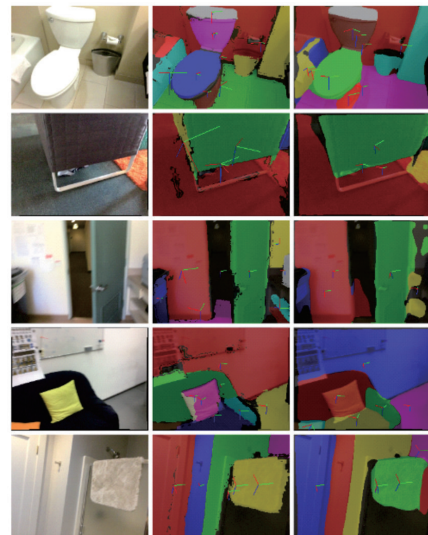


图 8 在 ScanNet 测试集上的平面法线召回率和像素法线召回率

Fig.8 Plane normal recall rate and pixel normal recall rate on ScanNet test set



RGB 原图 真实位姿检测 预测位姿检测
RGB original image Real pose Predict pose

图 9 在 ScanNet 数据集上的平面位姿估计结果

Fig.9 Plane pose estimation results on ScanNet dataset

结果基本与真实值一致。

图 10 为在用户图片和 NYU 数据集上的平面位姿检测结果, 可以看出位姿估计结果仍具有一定参考性。

实验中在 ScanNet 数据集上预测每张图片平面位姿平均用时 0.0545 s, 实时性为 18.5 f/s, 基本满足实时性要求。

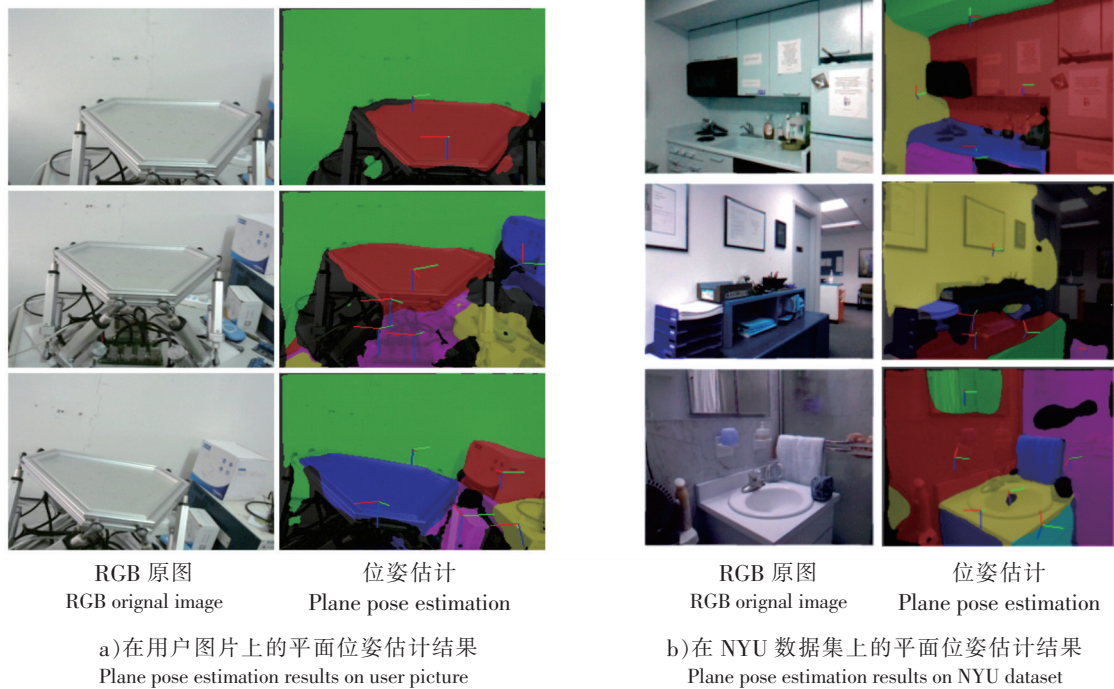


图 10 在用户图片和 NYU 数据集上的平面位姿估计结果

Fig.10 Plane pose estimation results on user picture and NYU dataset

4 结论

本文研究了基于深度学习的平面位姿检测算法, 采用编码器-解码器网络从单个 RGB 图像中检测平面实例分割及法线信息, 对每个平面进行位姿解算, 从而获得其实时位姿。实验结果表明, 在 ScanNet 测试数据集上的平面召回率为 0.625, 平面法线召回率为 0.414, 实时性为 18.5 f/s。同样在 NYU 和用户自制的 RGB 图上平面分割效果较好, 但基于单图像的平面法线预测准确率较差, 平面位姿估计效果有待提高。同时还表明, 基于深度学习的 RGB 图像平面位姿检测方法是可行的, 但平面检测效果尤其是单图像法线和深度估计的准确率有待进一步提升。未来改进平面位姿检测的工作应该集中在研究更鲁棒的实例分割和法线估计算法, 尝试多视图、多传感器融合等方法来提高平面位姿估计的准确率。

[参考文献]

- [1] MARCHAND E, UCHIYAMA H, SPINDLER F. Pose estimation for augmented reality: a hands-on survey [J]. IEEE Transactions on Visualization & Computer Graphics, 2016, 22(12): 2633-2651.
- [2] 丁小虎, 陈宁. 港口集装箱自主识别定位技术 [J]. 集美大学学报 (自然科学版), 2019, 24(4): 290-298.
- [3] 褚冠宜. 面向移动增强现实的多平面物体检测与跟踪 [D]. 杭州: 浙江大学, 2019.
- [4] SONDRE S T, HOVLAND G. Relative vessel motion tracking using sensor fusion, aruco markers, and MRU sensors [J]. Modeling, 2017, 38(2): 79-93.
- [5] 魏振忠. 舰载机着舰位姿视觉测量技术概述 [J]. 测控技术, 2020, 39(8): 2-6.

- [6] 李海博. 船模六自由度的实时测量方法与系统实现研究 [D]. 哈尔滨: 哈尔滨工程大学, 2017.
- [7] 崔伟. 基于双目视觉的六自由度波浪补偿检测技术研究 [D]. 长沙: 国防科学技术大学, 2016.
- [8] KAM H C, YU Y K, WONG K H. An improvement on ArUco marker for pose tracking using kalman filter [C] //19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). Busan: IEEE Computer Society, 2018.
- [9] WANG J, OLSON E. AprilTag 2: efficient and robust fiducial detection [C] //IEEE RSJ International Conference on Intelligent Robots & Systems. New York: IEEE, 2016.
- [10] TSENG H Y, WU P C, YANG M H, et al. Direct 3D pose estimation of a planar target [C] //Applications of Computer Vision. New York: IEEE, 2016.
- [11] RAD M, LEPETIT V. BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth [C]. IEEE International Conference on Computer Vision. Venice: IEEE, 2017.
- [12] WADIM K, FABIAN M, FEDERICO T, et al. SSD-6D: Making RGB-based 3d detection and 6d pose estimation great again [C] //2017 IEEE International Conference on Computer Vision (ICCV). New York: IEEE, 2017.
- [13] XIANG Y, SCHMIDT T, NARAYANAN V. Posecnn: a convolutional neural network for 6d object pose estimation in cluttered scenes [J]. Arxiv preprint arxiv, 1711.00199, 2018.
- [14] PENG S, LIU Y, HUANG Q, et al. PVNet: Pixel-wise voting network for 6DoF pose estimation [C] //IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). New York: IEEE, 2019.
- [15] WANG C, XU D, ZHU Y, et al. DenseFusion: 6D object pose estimation by iterative dense fusion [C] //IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). New York: IEEE, 2020.
- [16] LIU C, YANG J, CEYLAN D, et al. PlaneNet: piece-wise planar reconstruction from a single RGB image [C] //CVF Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2018.
- [17] YANG F, ZHOU Z. Recovering 3D planes from a single image via convolutional neural networks [C] //Proceedings of the European Conference on Computer Vision. Munich: 2018: 85-100.
- [18] LIU C, KIM K, GU J, et al. PlaneRCNN: 3D plane detection and reconstruction from a single image [C] //IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). New York: IEEE, 2020.
- [19] YU Z, ZHENG J, LIAN D, et al. Single-image piece-wise planar 3D reconstruction via associative embedding [C] //IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). New York: IEEE, 2020.
- [20] ANGELA D, ANGEL X. C, MANOLIS S, et al. ScanNet: richly annotated 3D reconstructions of indoor scenes [C] //IEEE Conference on Computer Vision and Pattern Recognition (CVPR). New York: IEEE, 2017.

(责任编辑 陈 敏 英文审校 郑青榕)