计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2023,32(5):105−111 [doi: 10.15888/j.cnki.csa.009078]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

面向嵌入式平台多视图立体视觉深度感知①

单　兵[1,2,3], 胡益民[2,3], 张　龙[1], 李加东[2,3]

[1](南京理工大学 机械工程学院, 南京 210094)
[2](中国科学院 苏州纳米技术与纳米仿生研究所, 苏州 215123)
[3](中国科学院 多功能材料与轻巧系统重点实验室, 苏州 215123)
通信作者: 张　龙, E-mail: lzhang@njust.edu.cn; 李加东, E-mail: jdli2009@sinano.ac.cn

摘　要: 针对目前基于神经网络的多视图立体视觉 (multi-view stereo, MVS) 深度估计算法存在参数量大、内存消耗严重, 难以满足当下低算力嵌入式平台的需求. 提出基于 MVS2D 极线注意力机制与 MobileNetV3-Small 的 MVS 深度感知网络 (Mobile-MVS2D). 该网络采用编码器-解码器的结构, 使用 MobileNetV3-Small 网络进行编码特征提取, 对源图像与参考图像之间不同特征层的尺度信息耦合采用极线注意力机制, 解码阶段引入 SE-Net 与跳跃连接扩展解码特征细节, 提升预测精度. 实验结果表明, 提出的模型在 ScanNet 数据集中在深度图的评价指标中展现较高的精度. 在与视觉 SLAM 结合下可以展现出较准确的三维重建效果, 具有较好的鲁棒性. 在 Jeston Xavier NX 上推理精度为 Float16 尺寸为 640×480 的图片组, 仅需 0.17 s, GPU 消耗仅需 1 GB, 能够满足低算力嵌入式平台的需求.

关键词: 多视图立体视觉; 嵌入式; 注意力机制; 三维重建

引用格式: 单兵,胡益民,张龙,李加东.面向嵌入式平台多视图立体视觉深度感知.计算机系统应用,2023,32(5):105–111. http://www.c-s-a.org.cn/1003-3254/9078.html

Multi-view Stereo Depth Perception for Embedded Platform

SHAN Bing[1,2,3], HU Yi-Min[2,3], ZHANG Long[1], LI Jia-Dong[2,3]

[1](School of Mechanical Engineering, Nanjing University of Science & Technology, Nanjing 210094, China)
[2](Suzhou Institute of Nano-tech and Nano-bionics (SINANO), Chinese Academy of Sciences, Suzhou 215123, China)
[3](Key Laboratory of Multifunctional Nanomaterials and Smart Systems, Chinese Academy of Sciences, Suzhou 215123, China)

Abstract: The current multi-view stereo (MVS) depth estimation algorithms based on neural networks involve a large number of parameters and serious memory consumption, which is difficult to meet the needs of the current embedded platforms with low-computing power. Therefore, this study proposes an MVS depth perception network (Mobile-MVS2D) based on the MVS2D epipolar attention mechanism and MobileNetV3-Small. The network adopts the structure of encoder-decoder and uses MobileNetV3-Small network for encoding feature extraction. In addition, it adopts the epipolar attention mechanism for the coupling of scale information of different feature layers between the source image and the reference image and introduces SE-Net and jump connection to expand the decoding feature details in the decoding stage and improve the prediction accuracy. Experimental results show that the proposed model shows high accuracy in the evaluation index of depth maps in the ScanNet data set. By Combining with visual SLAM, the model can show a more accurate three-dimensional reconstruction effect and has excellent robustness. On the Jeston Xavier NX, the model only costs 0.17 s in inferring the image group with the accuracy of Float16 and the size of 640×480, and the GPU consumption is only 1 GB. Therefore, it can meet the needs of embedded platforms with low-computing power.

Key words: multi-view stereo (MVS); embedded; attention mechanism; 3D reconstruction

① 收稿时间: 2022-09-27; 修改时间: 2022-10-27; 采用时间: 2022-11-30; csa 在线出版时间: 2023-03-17
CNKI 网络首发时间: 2023-03-20

近年来, 微小型机器人广泛应用于环境监测、农业植保、灾害救援等领域, 其对导航、避障、巡检等自主化需求增加, 深度感知是实现这些功能必不可少的关键技术. 单目相机具有能耗低、体积小、易标定等特点, 出现了很多利用单目相机深度估计的研究. 基于单目相机相关的深度估计可分为传统的 MVS 深度估计、基于网络的单目深度估计、基于网络的 MVS 深度估计. 基于网络的单目深度估计会对数据集产生严重的依赖性, 泛化性较差. 基于网络的 MVS 深度感知再纹理缺失、光照变化等环境下比传统方法展现出更为精准的深度预测效果, 因此将重点对基于网络的轻量化 MVS 深度估计进行研究.

MVS 旨在从多张图片估计出一个或多个深度图从而进行 3D 重建[1], 基于深度学习的 MVS 方法主要包括基于深度图、体素、点云的方法. 基于体素与点云需要大量内存不利于网络模型的整体运行, 难以进行部署. 基于深度图的方法相较于体素与点云的方法不会产生大量内存消耗同时可以轻松进行滤波处理. 早期基于深度图方法的 MVS-Net[2] 在数据集[3,4] 较传统 MVS 方法展现出卓越的性能, 但该模型在计算时消耗大量内存与时间. 为了限制内存和时间消耗, Badki 等人[5] 将深度估计转化为二进制分类问题, 虽然其精度并不是最先进的, 但其速度达到了较先进的水平. Yu 等人[6] 通过构造一个稀疏代价体来学习稀疏深度图, 然后使用高精度 RGB 图像和 2D 卷积对稠密化, 从而产生较为精准的深度图. 一些研究人员采样级联的方法[7–9] 从粗到细的优化深度图, 有效地在速度和精度间进行了权衡. Yang 等人[10] 通过采用轻量级的极线注意力机制有效地将单目和多视图进行结合, 在公开基于网络 MVS 的方法中达到了最为先进的精度与速度.

当前, 基于网络的 MVS 深度估计网络结构复杂, 冗余, 需要耗费大量的硬件资源, 而大多数嵌入式设备计算能力低, 无法满足使用要求, 想要在嵌入式平台上实现高效的深度感知, 主要难点就是如何平衡算法的精度与计算的耗时性. 为了平衡算法的精度与计算的耗时性出现了一系列的轻量级神经网络模型[11,12], 这类算法通过引入深度可分离卷积与 SE-Net[13] 注意力机制等方式在精度损失较小的情况下, 有效地减少了网络计算的吞吐量. 受上述启发, 提出基于 MVS2D 极线注意力机制与 MobileNetV3-Small 的 MVS 深度感知网络 (Mobile-MVS2D), 在精度损失有限情况下大幅降低网络参数以及计算量.

1 Mobile-MVS2D 模型构建与实现

所提出的 Mobile-MVS2D 模型结构如图 1 所示. 网络主要由 3 部分组成: 编码器 (特征提取)、基于 MVS2D 的极线注意力机制 (G)、解码器 (SE-upConv5). 其中编码器采用 MobileNetV3-Small, 图片和参考图片经过编码器进行特征提取, 在编码器的 Layer2、Layer3 通过基于 2D 卷积的轻量化的极线注意力机制将参考帧图片信息与源图片信息进行耦合. 在编码阶段, 高层的编码信息具有较高的分辨率, 但编码信息较少, 底层的编码信息分辨率较低, 但编码信息更多. 在解码阶段, 解码器采样双线性插值进行上采样、为了获取更多的特征信息在解码的 Layer3、Layer2 采用轻量级的跳跃连接.

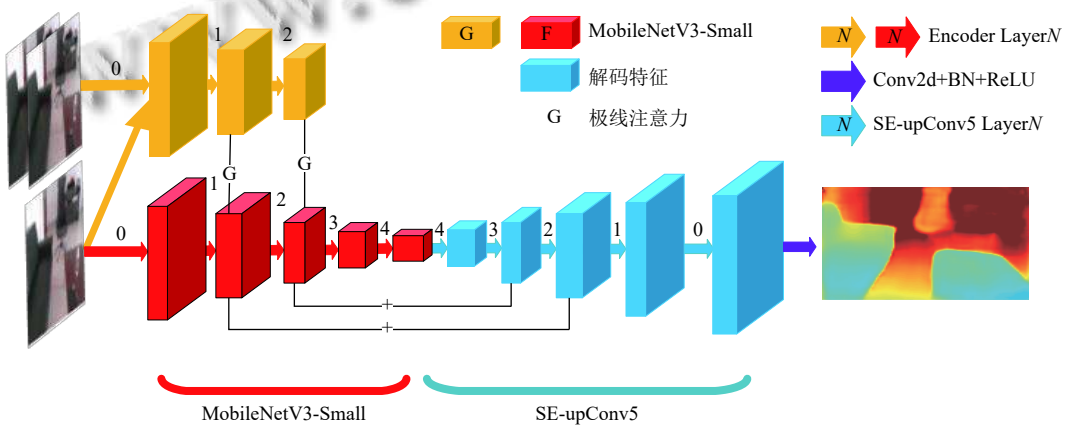

图 1 Mobile-MVS2D 结构

1.1 模型问题与编码层设置

模型的目的是在给定 n 张与源图像相同尺寸的参考图片$\{I_i\}_{i=1}^n$, 在已知相机参数$K\in R^{3\times3}$、源图像与参考图像之间的相对运动$T_i=(R_i|t_i)$, $R_i\in SO(3)$, $t_i\in R^3$下恢复源图像$I_0\in R^{h\times w\times3}$为每个像素$p_0$对应深度. 若源图像$I_0$的每个像素$p_0$对应的齐次坐标为$\overline{p}_0=\left(\overline{p}_{0,x},\overline{p}_{0,y},1\right)^{\mathrm{T}}$, p_0对应的深度d_0, 则根据成像原理在相机坐标系下的3D 点为$p_0(d_0)=d_0\left(K^{-1}p_0\right)$. 同样的, 根据相机运动方程可以将参考图片 i 坐标系中$p_0(d_0)$的 3D 坐标和齐次坐标可分别定义为$p_i(d_0)$和$\overline{p}_i(d_0)$, 如式 (1) 所示:

$$\begin{cases}p_i(d_0)=R_ip_0(d_0)+t_i\\ \overline{p}_i(d_0)=Kp_i(d_0)\end{cases}\tag{1}$$

网络模型的编码层主要是一个具有 L 层的编码前馈学习网络 F, 将$F_j\in R^{h_j\times w_j\times m_j}$表示第 j 层编码器的输出. m_j、h_j、w_j分别表示为 j 层输出特征维度、宽度、高度. 连续两层之间通过 MobileNetV3-Small 进行编码. 用通用的卷积$C_j:R^{h_j\times w_j\times m_j}\to R^{h_{j+1}\times w_{j+1}\times m_{j+1}}$每一个 Layer 总编码映射, 每一层的通用公式可表示为式 (2), 每一单层结构与 MobileNetV3-Small 一致, 每一 C_j 所包含的 MobileNetV3-Small 结构如表 1 所示, 表中$H\times W\times C_n$、$H\times W\times C_{\mathrm{out}}$分别为编码器特征输入输出的高、宽维度, Add为使用极线注意力. $A_j\in R^{h_j\times w_j\times m_j}$基于 MVS2D[10] 极线注意力机制, 在 Layer1、Layer2 层使用. 参考图的特征提取与源图的特征提取都采用训练好的 MobileNetV3-Small 进行迁移学习.

$$F_{j+1}=C_j\left(A_j+F_j\right)\tag{2}$$

表 1 编码层结构参数

C_i	$H\times W\times C_n$	$H\times W\times C_{\mathrm{out}}$	Add
C_0	640×480×3	320×240×16	
C_1	320×240×16	160×120×16	
C_2	160×120×16 80×60×24	80×60×24 80×60×24	+
C_3	80×60×24 40×30×40	40×30×40 40×30×40	+
C_4	40×30×40 40×30×40 40×30×48 40×30×48 20×15×96 20×15×96	40×30×40 40×30×48 40×30×48 20×15×96 20×15×96 20×15×96	

1.2 模型问题与编码层设置

极线注意力机制的目的是求参考帧I_n在 Layer1、Layer2 层的特征通过对极几何 (极线注意) 投影到源图像I_0对应的特征层编码信息进行融合, 根据 MVS2D 极线注意力机制, 可以将参考帧与源图像的耦合定义为$A_i(p_0)$ (式 (3)), 由 3 部分组成. $A_j^{ep}\left(p_0,\{I_i\}_{i=1}^n\right)$表示使用可训练代码 (卷积) 对$p_0$与对应参考图像之间的匹配结果进行编码. $A_j^0:R^{m_j}\to R^{m_j}$由 1×1 的卷积构成, 是对p_0在 j 层的特征进行可训练线性映射, 以增强源图像的特征信息灵活性.

$$A_j(p_0)=A_j^{ep}\left(p_0,\{I_i\}_{i=1}^n\right)+A_j^0\left(F_j(p_0)\right)+F_j(p_0)\tag{3}$$

$A_j^{ep}\left(p_0,\{I_i\}_{i=1}^n\right)$由参考图像特征在$p_0$极线上的样本组成. 样本是对深度值$d_0$进行采样然后根据式 (1) 获得. 将用$p_i^k$表示第 i 个参考图像上的第 k 个样本. 为了保持源图像与参考图像特征一致性, 采用一个同样的 MobileNetV3-Small (G) 编码器进行特征提取, 对源图像与参考图像在第 j 层的特征分别用$G_j(I_0,p_0)\in R^j$和$G_j\left(I_i,p_i^k\right)\in R^j$进行表示, 为了遵循缩放点积注意力 (scaled-dot product attention)[14], 对提取的特征引入$f_0^j:R^{m_j}\to R^{m_j}$和$f_i^j:R^{m_j}\to R^{m_j}$进行可训练线性映射. 对于p_0与p_i^k之间的匹配分数定义为w_{ik}^j:

$$w_{ik}^j=\left(f_0^j\left(G_j(I_0,p_0)\right)\right)^{\mathrm{T}}\left(f_{ref}^j\left(G_j\left(I_i,p_i^k\right)\right)\right)\tag{4}$$

为了解决参考图像中部分特征映射到源图像边界外和对匹配分数w_{ik}^j进行桥接, 引入$c_{jk}\in R^{m_j}$表示第 k 个训练样本的掩码. 用$v_{\mathrm{in}}^j\in R^{m_j}$和$v_{\mathrm{out}}^j\in R^{m_j}$分别表示边界内与边界外的信息, 可定义为:

$$v_{ik}^j=\begin{cases}v_{\mathrm{in}}^j, & 0\leqslant\overline{p}_{i,x}\leqslant w,\ 0\leqslant\overline{p}_{i,y}\leqslant h,\ 0\leqslant\overline{p}_{i,z}\\ v_{\mathrm{out}}^j, & \text{otherwise}\end{cases}\tag{5}$$

其中, $\overline{p}_i^k=\left(\overline{p}_{i,x}^k,\overline{p}_{i,y}^k,1\right)^{\mathrm{T}}$, $p_i^k=\left(p_{i,x}^k,p_{i,y}^k,1\right)^{\mathrm{T}}$. 为了提高$G_j$的表现力引入$A_j^1:R^{m_j}\to R^{m_j}$对$p_0$进行线性映射. 结合式 (4) 与式 (5) 可得:

$$\begin{aligned}A_j^{ep}\left(p_0,\{I_i\}_{i=1}^n\right)=&A_j^1\left(G_j(p_0)\right)\\&+\sum_{i=1}^n\sum_{k=1}^K N\left(\frac{w_{ik}^j}{\sqrt{m_j}}\right)\left(v_{ik}^j\cdot c_{jk}\right)\end{aligned}\tag{6}$$

其中, N 为$\dfrac{w_{ik}^j}{\sqrt{m_j}}$的 Softmax 归一化函数, $1\leqslant k\leqslant K$. 由式 (3) 与式 (6) 可得$A_j(p_0)$:

$$\begin{aligned}A_j(p_0)=&A_j^1\left(G_j(p_0)\right)+\sum_{i=1}^n\sum_{k=1}^K N\left(\frac{w_{ik}^j}{\sqrt{m_j}}\right)\left(v_{ik}^j\cdot c_{jk}\right)\\&+A_j^0\left(G_j(p_0)\right)+A_j^0\left(F_j(p_0)\right)\end{aligned}\tag{7}$$

1.3 SE-upConv5 解码器模型

Wofk 等人[15] 基于深度可分离网络提出快速解码结构 upConv5, 虽然相对普通的 U-Net[16] 解码结构在内存消耗上与运行速度上有了极大的提升, 但基于深度可分离网络的 upConv5, 在处理低层编码信息会产生大部分卷积无效的情况. 对此将引入 SE-Net[13], 通过上采样层通道之间的非线性关系提高解码器的全局信息表现能力. 所提出的 SE-upConv5 如图 2 所示. 每一层的解码流程为上一次解码特征 $N\times C_n\times W\times H$, 经过 1×1 的卷积核进行维度升维 $N\times C_{\text{up}}\times W\times H$ 然后进行 DW 卷积操作到 $N\times C_{\text{dw}}\times W\times H$, 之后是 SE-Module 模块进行解码信息的全局提升, 再通过 1×1 的卷积进行降维到 $N\times C_{\text{down}}\times W\times H$, 然后进行双线性插值到 $N\times C_{n+1}\times W\times H$, 为了进一步提高不同层级编码器信息与特征的提取程度我们将双线性插值后的解码层 [Layer4, Layer3, Layer2] 与之相同的编码特征层进行加和计算从而得到最终的解码层为 $N\times C_{n+1}\times 2W\times 2H$. 其中每一层的各级维度如表 2 所示. 表 2 中 H、W、C_n、C_{up}、C_{dw}、C_{down}、Add 分别表示输入特征层的高、输入层宽、输入层维度、Conv1×1 升维后卷积维度、深度可分离卷积维度、Conv1×1 降维维度、是否与编码层信息融合.

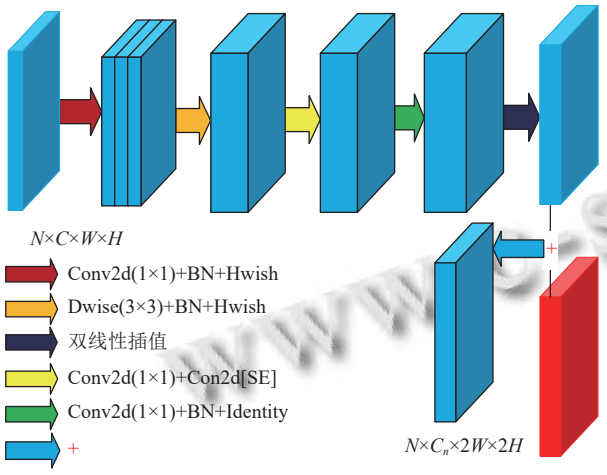

图 2 SE-upConv5 解码器模型

1.4 损失函数

基于神经网络的 MVS 深度估计任务中标准的损失函数为深度图的有效真实值与对应预测深度值之间的差. 不同类型的损失函数对 MVS 网络估计的精度和训练速度有很大的影响, 为了有效恢复图片的精度, 将采用的损失函数为文献 [17] 所提出的损失函数, 其由 3 部分组成, 其整体定义为$L(d,d^*)$, $L_{\text{depth}}(d,d^*)$表示预测深度值和真实深度值之间均差, $L_{\text{grad}}(d,d^*)$为像素梯度损失对深度图的深度边缘进行梯度约束, $L_{SSIM}(d,d^*)$利用结构相似性对图像进行约束. 其中d_p表示预测深度, d_p^*表示真实深度值, N 表示所有带真实标签的像素.

$$L(d,d^*)=\lambda L_{\text{depth}}(d,d^*)+L_{\text{grad}}(d,d^*)+L_{SSIM}(d,d^*) \quad (8)$$

$$L_{\text{depth}}(d,d^*)=\frac{1}{N}\sum_p^n\left|d_p-d_p^*\right| \quad (9)$$

$$L_{\text{grad}}(d,d^*)=\frac{1}{N}\sum_p^n\left|g_x\left(d_p-d_p^*\right)+g_y\left(d_p-d_p^*\right)\right| \quad (10)$$

$$L_{SSIM}(d,d^*)=\frac{1-SSIM\left(d_p-d_p^*\right)}{2} \quad (11)$$

表 2 SE-upConv5 结构参数

参数	H	W	C_n	C_{up}	C_{dw}	C_{down}	Add
Layer5	15	20	96	288	288	40	
Layer4	30	40	40	120	120	24	
Layer3	60	80	24	96	96	16	+
Layer2	120	160	16	72	72	16	+
Layer1	240	320	16	32	32	16	

2 实验过程与结果分析

本节将展现所提出的 Mobile-MVS2D 网络模型在公开数据集 ScanNet[18] 与现实场景两部分进行实验测试. 在 ScanNet 数据集上, 与现有部分开源方法[2,6,9,10] 进行对比. 在现实场景中, 将基于四旋翼所采集的室内数据在与世界 SLAM 的 VINS 扩展框架上进行深度估计与三维重建, 嵌入式设备 Jeston Xavier NX 上进行了位姿估计与深度感知测试.

2.1 实验环境

实验训练环境: 硬件为 CPU 为 Intel Xeon(R) Silver 4214, 48 核, 单核 2.20 GHz, 显卡为 RTX 5000, 内存为 16 GB, 运行内存为 8 GB. 软件环境为 PyTorch 1.7.0, Python 3.7, Ubuntu 18.04, CUDA 10.2, cuDNN 8.2.1.

嵌入式设备: Jeston Xavier NX, 实际运行内存 6 GB, GPU 为 384-core NVIDIA Volta GPU 和 48 Tensor Cores, 算力为 6T (Float16), CPU 为 6 核, 单核 1.4 GHz. 软件环境为 PyTorch 1.9.0, Python 3.6, Ubuntu 18.04, CUDA 10.2.

实验设置: 输入图像的分辨率为 640×480; 输入图像个数为 3; 采用 Adam 优化器, 其中$\epsilon=10^{-8}$, $\beta=$

(0.9,0.999), 对于 ScanNet 数据集, 采用的初始学习率为2E−3, 在第 5 轮和第 10 轮的时候学习率分别降低 0.05; 训练 batch 为 12; Batch_Size 为 80, 训练时长 4–5 天.

2.2 ScanNet 数据集实验对比

2.2.1 数据集介绍

实验选用的 ScanNet 数据集包含 807 个独立场景, 其中包含不同相机轨迹捕获的图像序列, 为了保证训练速度, 将参考 MVS2D 中采样方式进行采样, 对 807 个独立场景进行采样 1:20 的采样, 选其中 700 个数据集进行为训练数据集, 对剩余数据集再进行 1:40 的采样. 最终选用 72 574 组图片 (一幅源图像和两幅参考图像) 进行训练和 666 组图片 (包含 107 个独立场景) 用于测试.

2.2.2 算法评估标准及结果对比

将根据深度估计中公认的评价指标对相关网络进行评估, 其中d_i表示预测深度, d_i^*表示真实深度值, N 表示所有带真实标签的像素. 评价公式为绝对值相对偏差 (*AbsRel*)、线性均方根误差 (*RMSE*)、线性均方根误差 (*RMSE*)、相对平方差 (*SqRel*), 不同阈值下的准确度 δ_i: 相对误差在阈值内预测像素的百分比.

$$AbsRel = \frac{1}{N}\sum\left|\frac{d_i^* - d_i}{d_i}\right| \tag{12}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_i^n \left|d_i - d_i^*\right|^2} \tag{13}$$

$$SqRel = \frac{1}{N}\sum\left(\frac{d_i^* - d_i}{d_i^*}\right)^2 \tag{14}$$

$$\max\left(\frac{d^*}{d}, \frac{d}{d^*}\right) \leqslant \delta_i,\ \delta_i = 1.25, 1.25^2, 1.25^3 \tag{15}$$

Time (ms): 为了更加适配嵌入式设备 Jeston NX, 本实验中将模型设置为半径度 (Float16), 图片的尺寸为 640×480, 在 Jeston NX 上进行实时的模型速率的验证. 为了保证数据的准确性, 将进行 300 次的推理然后选其平均值.

表 3 给出了在公开数据集 ScanNet 上公开算法[2,6,9,10]的对比结果 (部分数据来源于 MVS2D), 其中 MVS2D 基于多层极线注意力机制, 其在精度与速度在公开算法中为最优结果. 同时, 可视化了所提出的 MVS 算法在 ScanNet 数据集上的预测结果, 如图 3 所示, 为了更好说明算法的效果, 同样可视化了 MVS2D 预测结果进行对比, 可视化工具为 Python 中 matplotlib 工具包中热力图显示模块, 可以看出所提出的算法在 ScanNet 数据上展现出了较高的精度, 对于低纹理与被遮挡的地方有较好的补充, 同时在 Jeston Xavier NX 上的推理速度仅为 0.17 s.

表 3 ScanNet 数据集实验对比

Method	*AbsRel* ↓	*SqRel* ↓	*RMSE* ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑	Time (ms)
MVSNet[2]	0.094	0.042	0.253	0.897	0.975	0.993	—
FastMVS[6]	0.089	0.038	0.231	0.912	0.978	0.993	—
PatchmatchNet[9]	0.133	0.075	0.320	0.834	0.955	0.987	334
MVS2D[10]	0.059	0.016	0.159	0.965	0.996	0.999	218
Ours	0.079	0.024	0.189	0.943	0.991	0.998	**168**

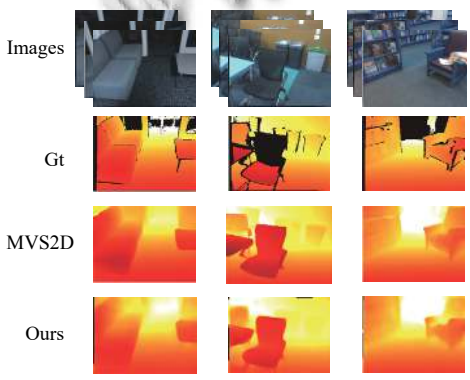

图 3 数据效果展示

2.3 真实环境实验展示

2.3.1 VINS 扩展方法

为了验证所提出方法的扩展性与简化现实环境下 MVS 数据收集流程, 同时满足机器人多元化需求与多设备之间的高效通信, 将基于 ROS 通信机制, 采用一种级联的方法对当下流行的嵌入式视觉 SLAM 框架 VINS[19] 进行扩展. 其流程如图 4 所示.

所提出的 VINS 扩展框架主要包含两个部分: 数据收集、位姿估计与深度感知. 位姿估计感知. 位姿估计与深度感知处理流程为: 图片与 IMU 数据经过

VINS 获取图片与对应的位姿, 通过 ROS 特有的自定义话题传入到 Mobile-MVS 中进行处理, 在处理过程中为了防止数据冗余与造成内存拥挤, 采用一种基于滑动窗口的形式, 对之前图片的信息进行有效利用, 经 Mobile-MVS 进行深度预测获得图片对应的深度图, 利用针孔成像原理对深度图与 RGB 图片进行拼接, 为了进一步保证点云输出的实时性, 将对点云进行八叉树 (OctoMap) 算法压缩, 生成轻量化可用于机器人导航、避障的点云地图.

2.3.2 VINS 扩展框架的实验结果对比

为了进一步展示提出方法的鲁棒性, 将以四旋翼机器人搭载 D435I 相机对室内真实环境的 RGB 图片、IMU 数据、深度图进行收集. RGB 和深度图片收集的像素为 640×480、帧率为 30 f/s. IMU 数据收集的帧率为 200 Hz. 所有的数据均在无人机绕室内正常飞行, 最终以 ROS 的数据格式 bag 进行存储. 利用所采集的数据在基于 VINS 扩展框架在 CPU 为锐龙 R7-5800H、16 GB 运行内存、显卡为 RTX3050Ti 的电脑端实现了实时的位姿估计和深度感知与三维重建, 经所提出鲁棒的训练集收集方法训练后, 其深度感知效果与三维重建效果分别如图 5 和图 6 所示. 为了进一步展现所提出方法的精度, 同样方法对 D435i 所采集的深度图、RGB 图、IMU 数据在 VINS 下进行三维重建, 由可视化地图可看出所提出方法具有较好的可扩展性与精度. 为了体现所提出算法在嵌入式设备上 Jeston Xavier NX 实现了 10 Hz 的位姿估计与 5 Hz 的像素为 480×384 的深度图感知, 可以满足当下嵌入式机器人对于避障等算法实时性要求.

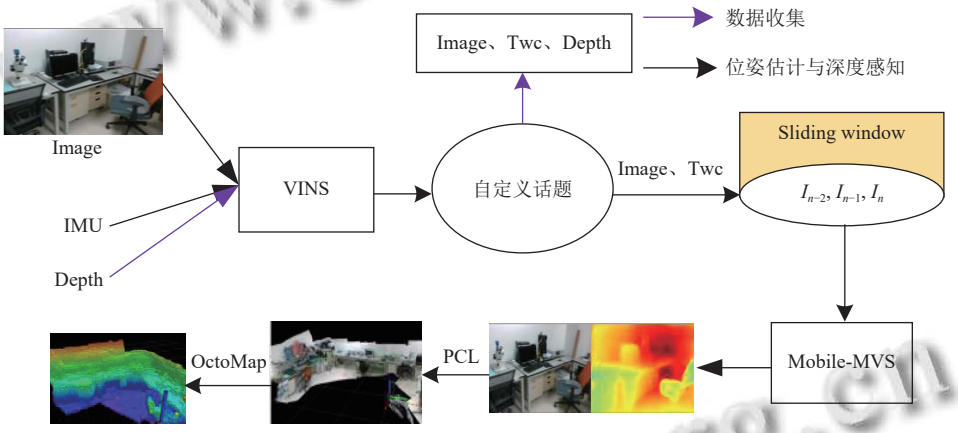

图 4 VINS 扩展框架

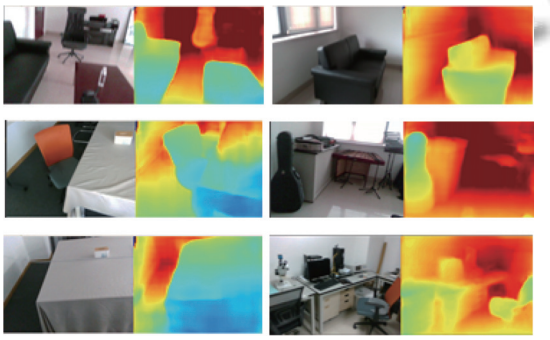

图 5 真实环境下深度感知

3 结束语

为了解决低算力嵌入式平台算力吞吐量低的问题, 本文基于 MVS2D 深度感知网络的极线注意力机制, 引入高效的编码 MobileNetV3-Small 与 SE-Net 网络, 设计了一款轻量级的 MVS 深度感知网络 Mobile-MVS2D. 本文研究的内容主要从以下方面进行展开: (1) 为了加速模型的推理速度与训练速度, 采样 Mobile-NetV3-Small 进行信息编码 (特征提取) 与迁移学习, 为了增强解码信息在解码层引入 SE 注意力机制, 提出一种新的 SE-upConv5 结构. (2) 为了增强模型在现实环境的表达能力, 基于 VINS 提出一种简易的数据集收集、深度估计与三维重建框架. 结果如下: 1) 在 ScanNet 数据集可以达到较高的精度, 相对于当下公开算法精度最高与速度最快的 MVS 网络架构 MVS2D 在图像尺寸为 640×480、精度为 Float16、数量为 3 的图片下, 在精度损失有限情况下, 速度提升了 20% 左右, 预测一张图片在嵌入式设备 Jeston NX 上仅需 168 ms,

GPU 仅消耗 1 GB. 2) 在所提出的 VINS 三维重建框架下实现了较高质量的点云地图, 同时在嵌入式设备上 Jeston Xavier NX 实现了 10 Hz 的位姿估计与 5 Hz 的像素为 480×384 的深度图感知.

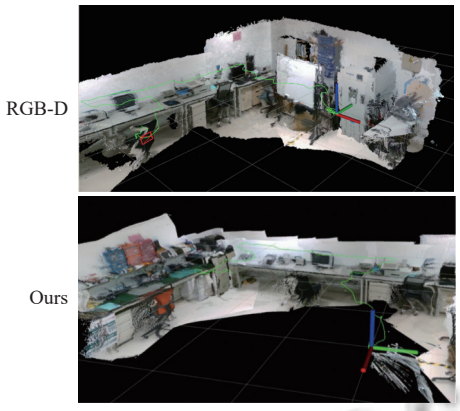

图 6 三维重建效果展示

参考文献

1 Laga H, Jospin LV, Boussaid F, *et al*. A survey on deep learning techniques for stereo-based depth estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(4): 1738–1764. [doi: 10.1109/TPAMI.2020.3032602]

2 Yao Y, Luo ZX, Li SW, *et al*. MVSNet: Depth inference for unstructured multi-view stereo. Proceedings of the 15th European Conference on Computer Vision. Munich: Springer, 2018. 785–801.

3 Knapitsch A, Jaesik P, Qian YZ, *et al*. Tanks and temples: Benchmarking large-scale scene reconstruction. ACM Transactions on Graphics, 2017, 36(4): 78.

4 Schöps T, Schönberger JL, Galliani S, *et al*. A multi-view stereo benchmark with high-resolution images and multi-camera videos. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 3260–3269.

5 Badki A, Troccoli A, Kim K, *et al*. Bi3D: Stereo depth estimation via binary classifications. Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 1597–1605.

6 Yu ZH, Gao SH. Fast-MVSNet: Sparse-to-dense multi-view stereo with learned propagation and Gauss-Newton refinement. Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 1946–1955.

7 刘会杰, 柏正尧, 程威, 等. 融合注意力机制和多层 U-Net 的多视图立体重建. 中国图象图形学报, 2022, 27(2): 475–485. [doi: 10.11834/jig.210516]

8 Yang JY, Mao W, Alvarez JM, *et al*. Cost volume pyramid based depth inference for multi-view stereo. Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE , 2020. 4876–4885.

9 Wang FJH, Galliani S, Vogel C, *et al*. Patchmatchnet: Learned multi-view patchmatch stereo. Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE , 2021. 14189–14198.

10 Yang ZP, Ren ZL, Shan Q, *et al*. MVS2D: Efficient multiview stereo via attention-driven 2D convolutions. Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022. 8564–8574.

11 Howard A, Sandler M, Chen B, *et al*. Searching for MobileNetV3. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 1314–1324.

12 Tan MX, Le Q. EfficientNetV2: Smaller models and faster training. Proceedings of the 38th International Conference on Machine Learning. PMLR, 2021. 10096–10106.

13 Hu J, Shen L, Sun G. Squeeze-and-excitation networks. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 7132–7141.

14 Vaswani A, Shazeer N, Parmar N, *et al*. Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6000–6010.

15 Wofk D, Ma FC, Yang TJ, *et al*. FastDepth: Fast monocular depth estimation on embedded systems. Proceedings of the 2019 International Conference on Robotics and Automation. Montreal: IEEE , 2019. 6101–6108.

16 Ronneberger O, Fischer P, Brox T. U-Net: Convolutional networks for biomedical image segmentation. Proceedings of the 18th International Conference on Medical Image Computing and Computer-assisted Intervention. Munich: Springer, 2015. 234–241.

17 Qin T, Li PL, Shen SJ. VINS-Mono: A robust and versatile monocular visual-inertial state estimator. IEEE Transactions on Robotics, 2018, 34(4): 1004–1020. [doi: 10.1109/TRO.2018.2853729]

18 Alhashim I, Wonka P. High quality monocular depth estimation via transfer learning. arXiv:1812.11941, 2018.

19 Dai A, Chang AX, Savva M, *et al*. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 2432–2443.

(校对责编: 牛欣悦)