

引文格式:邓非,颜青松,肖腾.多视密集匹配并行传播 GPU-PatchMatch 算法[J].测绘学报,2020,49(2):181-190. DOI:10.11947/j. AGCS.2020.20180459.  
DENG Fei, YAN Qingsong, XIAO Teng. A GPU-PatchMatch multi-view dense matching algorithm based on parallel propagation[J]. Acta Geodaetica et Cartographica Sinica, 2020, 49(2): 181-190. DOI: 10.11947/j. AGCS.2020.20180459.

## 多视密集匹配并行传播 GPU-PatchMatch 算法

邓 非<sup>1,2</sup>, 颜青松<sup>1</sup>, 肖 腾<sup>1</sup>

1. 武汉大学测绘学院, 湖北 武汉 430079; 2. 自然资源部城市国土资源监测与仿真重点实验室, 广东 深圳 518000

## A GPU-PatchMatch multi-view dense matching algorithm based on parallel propagation

DENG Fei<sup>1,2</sup>, YAN Qingsong<sup>1</sup>, XIAO Teng<sup>1</sup>

1. School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China; 2. Key Laboratory of Urban Land Resources Monitoring and Simulation, MNR, Shenzhen 518000, China

**Abstract:** Aiming at the problem of low efficiency of multi-view dense matching, a GPU-PatchMatch multi-view dense matching algorithm is proposed. The algorithm uses GPU to improve the computational efficiency of PatchMatch. At the same time, it also makes full use of sparse scene information to initialize the depth information. In addition, in order to improve the propagation efficiency, it uses the pyramid red-blackboard to propagate the depth information in parallel. Finally, the experiments are carried out on the DTU, Strecha and Vaihigen datasets, and compared with the commonly used multi-view dense matching algorithms. The results show that our algorithm has a significant improvement in reconstruction efficiency, which is 7 times higher than the CPU algorithm (such as PMVS, MVE and OpenMVS), and 2.5 times higher than the GPU algorithm (such as Gipuma), which proves the effectiveness of the proposed method.

**Key words:** 3D reconstruction; multi-view dense matching; PatchMatch; parallel computation

**Foundation support:** The Open Fund of Key Laboratory of Urban Land Resources Monitoring and Simulation, MNR (No. KF-2018-03-025)

**摘 要:** 针对多视密集匹配的效率较低的问题, 提出了 GPU-PatchMatch 多视密集匹配算法。该算法使用 GPU 提高 PatchMatch 的计算效率; 同时充分利用稀疏场景信息, 对深度信息进行规则初始化; 为提高传播效率, 使用了金字塔红黑板并行传播深度信息。最后在 DTU、Strecha 和 Vaihigen 数据集上进行了试验, 并与常用的多视密集匹配算法进行对比。试验结果表明, 本文算法在重建效率上有较大提高, 与 CPU 算法 (PMVS、MVE、OpenMVS) 相比有 7 倍以上提升, 与 GPU 算法相比也有 2.5 倍以上提升, 表明本文算法的有效性。

**关键词:** 三维重建; 多视密集匹; 块匹配; 并行计算

**中图分类号:** P237

**文献标识码:** A

**文章编号:** 1001-1595(2020)02-0181-10

**基金项目:** 自然资源部城市国土资源监测与仿真重点实验室开放基金 (KF-2018-03-025)

基于影像的三维重建方法具有便捷、低成本等优点, 已经被广泛地应用到生产实际中。多视密集匹配是其中的关键技术之一, 通过计算影像间像素的对应关系恢复出场景的密集点云。但随着重建范围和影像数量的增加, 多视密集匹配的效率急剧下降, 严重阻碍了三维重建在军事、抗震

救灾、城市规划中的应用。因此, 提升多视密集匹配效率是目前生产实际中的现实需求。

传统的多视密集匹配分为 4 种类型<sup>[1-2]</sup>, 其中较为常用的是特征点扩张法<sup>[3-5]</sup>和深度图融合法<sup>[6-12]</sup>。除此之外, 近年来还出现了基于深度学习的方法<sup>[13-15]</sup>。特征点扩张法首先从纹理丰富

区域提取较为稳健的特征点,然后逐步扩张,获取整个场景的密集点云。该方法虽然精度高,但是计算复杂,不适合用于大规模三维重建。深度图融合法将多视密集匹配问题简化成双目密集匹配,为每张影像计算深度信息,最后用约束条件融合深度信息得到密集点云。此方法具有较高的计算效率,能满足大规模重建的需求,但精度稍低。深度学习法则使用大量数据训练深度神经网络的参数,然后直接基于影像预测出深度信息。该方法在弱纹理区域能够得到较好的结果,但极度依赖训练数据,当训练数据不足时其重建质量会快速下降,并且该方法对硬件要求较高,目前仅用于处理较小分辨率的数据。

目前,国内也有大量多视密集匹配的研究结果。文献[4]结合特征尺度分布信息改进的对极几何约束,缩小匹配搜索范围,实现同名特征的高精度快速密集匹配。文献[5]以规则格网划分的空间平面作为基础,提出一种像方特征点和物方平面元集成的多视影像密集匹配方法。文献[8]提出一种基于 PatchMatch 算法的多视密集匹配算法,可以高效进行大规模场景的密集点云重建。文献[10]使用影像重叠关系构建立体相对,并引入导向中值滤波改进了 SGM 算法。文献[11]融合 SIFT 和 SGM 进行多视密集匹配,利用匹配的 SIFT 特征作为 SGM 优化计算的路径约束条件。文献[14]提出了一种端到端多视密集匹配框架,首先使用二维卷积网络用于生成初始深度图,然后使用三维卷积神经网络进行初始深度图优化。

PatchMatch 算法是一种简单高效的双目密集匹配算法,文献[8—9]基于该算法进行多视密集匹配。但文献[8—9]都是直接使用随机值初始化深度信息,没有充分利用稀疏场景信息。除此之外,文献[8]基于 CPU 实现 PatchMatch 算法,计算效率较低;文献[9]虽然使用 GPU 实现 PatchMatch 算法,但并没有完全解决并行深度信息传播带来的传播效率下降的问题,仅通过提高迭代次数和增加传播范围进行补偿。针对初始化和传播效率问题,本文提出了 GPU-PatchMatch 多视密集匹配算法。一方面使用稀疏信息构建的三角网进行规则初始化,提高初始化质量;另一方面使用金字塔红黑板进行深度信息传播,提升传播效率。试验结果表明,本文算法在保证重建精度和重建完整度与现有算法相当的前提下,可大幅度减少重建耗时,满足目前生产实践中对高效

率的要求。

## 1 PatchMatch 双目密集匹配

PatchMatch 算法由文献[16]提出,通过引入随机采样的方式快速在图像间寻找相似图像块。随后文献[17]提出了 PatchMatch 双目密集匹配算法。其核心步骤是:核线校正、深度信息随机初始化、深度信息全局传播和深度信息随机搜索,如图 1 所示。

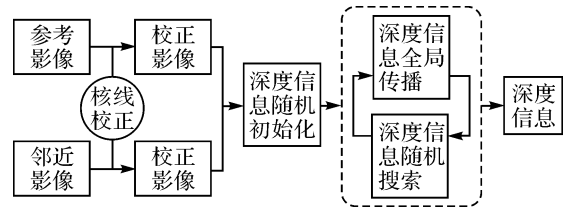


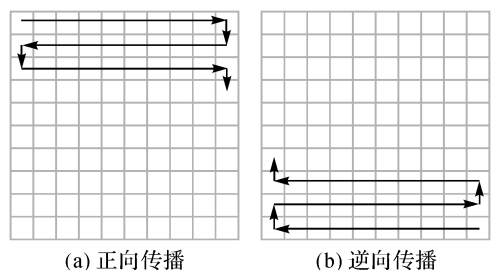
图 1 PatchMatch 双目密集匹配流程

Fig.1 PatchMatch stereo

虽然该算法使用视差描述深度信息,但其通过像素  $p$  的像素坐标  $(p_x, p_y)$  和一个三自由度的参数  $f_p(a_p, b_p, c_p)$  来刻画视差  $p_d$ ,如式(1)所示

$$p_d = a_p p_x + b_p p_y + c_p \quad (1)$$

因此在该算法中,深度信息随机初始化是指用随机数为  $f_p$  的 3 个参数设置初值。深度信息全局传播是指逐像素,将相邻像素的深度信息传播到当前像素,并根据匹配代价判断是否更新当前像素的深度信息。图 2 展示了全局传播时像素的传播次序,包括正向传播和逆向传播,即从影像的左上角或右下角的像素开始,直到图像的右下角或左上角像素结束。深度信息随机搜索是在全局传播的同时,通过随机搜索为当前像素寻找更优的深度信息。最后,循环迭代进行全局传播和随机搜索,经过一定迭代次数后即可恢复出场景的深度信息。



(a) 正向传播

(b) 逆向传播

图 2 深度信息全局传播

Fig.2 Global propagation of depth information

## 2 GPU-PatchMatch 多视密集匹配算法

文献[8]首次将 PatchMatch 算法引入多视密集匹配中,随后文献[9]对 PatchMatch 算法进行了 GPU 改进。本文在此基础上提出了 GPU-PatchMatch 多视密集匹配算法,通过使用深度信息规则初始化算法和金字塔红黑板深度信息并行传播算法提高计算效率。

为便于算法介绍,现作以下定义表述。记稀疏场景  $S = \{I, P\}$ ,其中  $I$  为  $m$  张影像的集合, $P$  为稀疏点的集合(即空三时的加密点); $I_i (1 \leq i \leq m) \in I$ ,其内参数矩阵为  $K_i$ ,外参数为  $R_i$  和  $C_i$ ,对应的像空间坐标系为  $C_i$ -XYZ;同时稀疏点  $P$

中  $I_i$  可见点记为  $P_{I_i}$ , $p \in P_{I_i}$  在  $I_i$  上的投影为  $I_i(p)$ ;另记  $D$  为待重建密集点云的集合, $D$  中  $I_i$  可见点记为  $D_{I_i}$ , $d \in D_{I_i}$  在  $I_i$  上的投影为  $I_i(d)$ 。

图 3 展示了本文算法的主要流程:首先通过场景划分为每张影像寻找一定数量的相邻影像;然后针对每张影像,利用稀疏场景完成深度信息初始化,并通过金字塔红黑板并行传播和深度信息随机搜索得到最终的深度信息;最后再融合所有影像的深度信息得到重建点云。本文将首先介绍深度信息及匹配代价,然后再逐步介绍场景划分、深度信息规则初始化、金字塔红黑板并行传播、深度信息随机搜索和深度信息融合。

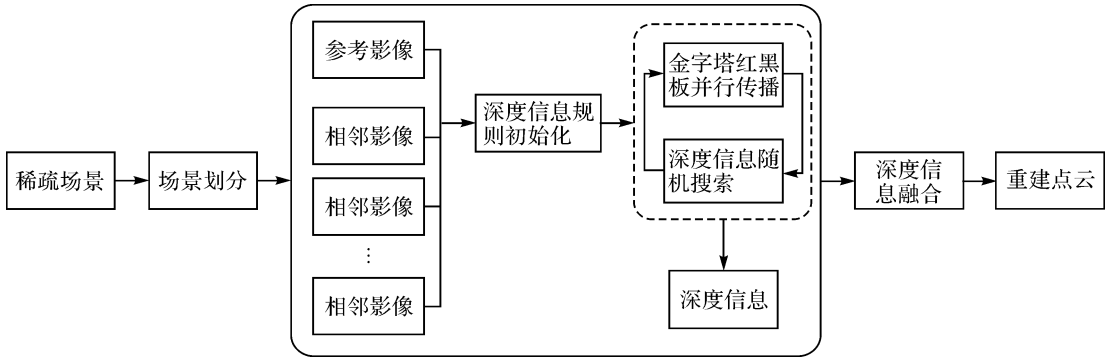


图 3 GPU-PatchMatch 多视密集匹配算法流程

Fig.3 GPU-PatchMatch multi-view dense matching algorithm

### 2.1 深度信息及匹配代价

密集匹配通常使用核线影像<sup>[7,10,17-18]</sup>,并利用视差描述深度信息。然而此类方法一方面需要耗费时间进行核线影像校正,另一方面还会由于计算精度引入误差降低匹配精度。本文参考文献[8]的深度信息表达方法,像素  $I_i(d)$  对应物方点  $d \in D_{I_i}$  的深度信息使用其在  $C_i$ -XYZ 下的深度  $d_{depth}$  和  $d$  与其相邻的物方点构成的平面  $S_d$  的法向量  $d_{normal}$  来表示,如图 4 所示。虽然这种方法与视差相比更复杂,但是其不需要对影像进行核线校正,能避免额外耗时,提高效率,并且该方法直接根据物方点和影像间的几何关系建模,更符合实际情况。

匹配代价是衡量深度信息质量的标准。目前有很多计算匹配代价的方法,如 ANCC<sup>[19]</sup>,互信息<sup>[18]</sup>和 DAISY<sup>[20]</sup>等,但这些方法计算复杂,效率较低。本文平衡效率和质量,选用 NCC 计算匹配代价。NCC 计算简单,具有亮度不变性,在影

像质量较好时,其计算的结果与其他复杂方法的结果相当<sup>[8]</sup>。在计算时,首先要获取参考影像和相邻影像上对应的像素块。在参考影像  $I_i$  上,直接在像素  $I_i(d)$  附近取一个窗口内的所有像素即可;在相邻影像  $I_j$  上,基于物方点位于一个平面上的假设,可以通过单应变换  $H_{ij}(I_i(d))$  将参考影像上所取窗口内像素映射到相邻影像上。如图 5 所示,其中红色区域表示在参考影像和向量影像上选取用于计算  $I_i(d)$  深度信息匹配代价的像素。

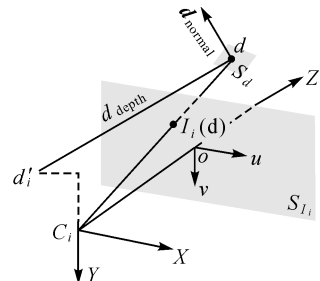


图 4 深度信息

Fig.4 Depth information

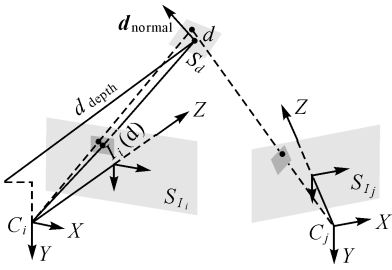


图5 影像间对应关系

Fig.5 The Correspondence between two images

根据多视图几何,可以得到  $H_{ij}(I_i(d))$  的计算公式,如式(2)所示,其中  $K$  是内参数, $R, C$  是外参数, $d_{normal}, d_{depth}$  是像素  $I_i(d)$  的深度信息<sup>[8]</sup>。由于在进行多视密集匹配时,参考影像会有多张相邻影像,根据不同的相邻影像,像素  $I_i(d)$  可计算出多个 NCC 值。本文中选取 NCC 值最大的作为像素  $I_i(d)$  的匹配代价

$$H_{ij}(I_i(d)) = K_j \left( R_j R_i^{-1} + \frac{R_j (C_i - C_j) \mathbf{d}_{normal}^T}{\mathbf{d}_{normal}^T [d_i; d_{depth}]} \right) K_i^{-1} \quad (2)$$

### 2.2 场景划分

在进行多视密集匹配时,每张影像都将依次

$$\omega_s(I_i, p, I_j) = \begin{cases} 2 \text{scale}(I_j, p) / \text{scale}(I_i, p) & 2 \leq \text{scale}(I_i, p) / \text{scale}(I_j, p) \\ 1 & 1 \leq \text{scale}(I_i, p) / \text{scale}(I_j, p) \leq 2 \\ \text{scale}(I_i, p) / \text{scale}(I_j, p) & \text{scale}(I_i, p) / \text{scale}(I_j, p) < 1 \end{cases} \quad (5)$$

式中,  $\text{scale}(I_i, p)$  表示  $p$  处的单位球投影在影像  $I_i$  上圆的像素直径,如果  $\text{scale}(I_i, p)$  和  $\text{scale}(I_j, p)$  差异较小,则意味着  $p$  在  $I_i$  和  $I_j$  上成像的尺度相似,更利于匹配代价的计算。

### 2.3 深度信息规则初始化

使用基于视觉算法恢复影像的外方位元素时,会同时得到稀疏场景  $S$ ,图 6 展示了一个稀疏场景。传统 PatchMatch 算法对稀疏场景的使用程度较低,文献[8—9,17]都仅使用稀疏场景的深度范围初始化深度信息。虽然 PatchMatch 算法并不需要较好的初值,但其代价是需要更多次迭代。除此之外,随机数初始还会降低算法的稳健性,尤其是当场景深度范围较大时。为此,本文提出了深度信息规则化初始化,利用稀疏点进行深度信息的初始化,主要流程包括稀疏场景规则化和规则初始化。

稀疏场景规则化是指利用影像  $I_i$  的可见稀疏点  $P_{I_i}$  获取  $I_i$  所见场景的大致形状,其步骤是首先将  $P_{I_i}$  投影到  $I_i$  上得到投影点  $I_i(P_{I_i})$ ,然后利用  $I_i(P_{I_i})$  在投影面上进行 Delaunay 构网<sup>[21]</sup>,

作为参考影像,并用场景划分为其选取相邻影像。场景划分的核心思路是通过分析参考影像与剩余影像间的共同可见的稀疏点来计算权重,最后根据权重从剩余影像中挑选出一定数量的影像作为参考影像的相邻影像<sup>[6]</sup>,如式(3)所示,其展示了  $I_i$  与  $I_j$  间权重的计算公式

$$W(I_i, I_j) = \sum_{p \in P_{I_i} \cap P_{I_j}} \omega_a(I_i, p, I_j) \omega_s(I_i, p, I_j) \quad (3)$$

式(3)中  $\omega_a(I_i, p, I_j)$  描述的是  $I_i$  和  $I_j$  在稀疏点  $p$  上的视角情况,如式(4)所示

$$\omega_a(I_i, p, I_j) = \min((\text{angle}(I_i, p, I_j) / \text{angle}_{\max})^2, 1) \quad (4)$$

式中,  $\text{angle}(I_i, p, I_j)$  表示  $p$  与  $I_i, I_j$  投影中心构成的夹角。当夹角过小时,成像几何条件较差;而当夹角过大时,视角差异较大,不利于匹配代价计算。因此在实际应用中取视角在一定角度的最佳,本文取  $\text{angle}_{\max} = 60^\circ$ 。

式(3)中  $\omega_s(I_i, p, I_j)$  描述的是  $I_i$  和  $I_j$  在  $p$  上的尺度的差异,如式(5)所示

最后结合投影点  $I_i(P_{I_i})$  的投影深度,将二维 Delaunay 网格转化成三维网格  $T_{I_i}$ 。图 7(a)展示了  $I_i$  可见点  $P_{I_i}$ ,图 7(b)展示对应的三维网格  $T_{I_i}$ 。与简单地使用深度范围相比,  $T_{I_i}$  能够对  $I_i$  的所见场景有更加细致的描述。

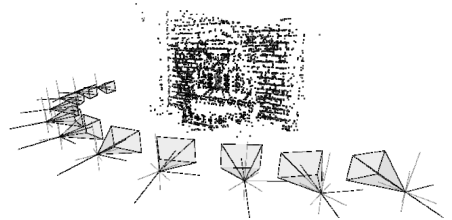


图6 稀疏场景示例

Fig.6 An example of sparse scene

规则初始化是利用  $T_{I_i}$  进行  $I_i$  深度信息的初始化。对于  $I_i$  上的一个像素  $I_i(p)$ ,首先判断其在  $T_{I_i}$  中可见的三角面  $F_p$ ,根据  $F_p$  的 3 个顶点可以求出其表达式  $ax + by + cz + 1 = 0$ ,则该像素的深度  $\text{depth}_{d_{I_i}(p)}$  和法向量  $\text{normal}_{d_{I_i}(p)}$  的计算如

式(6)和式(7)所示

$$\text{depth}_{d_{I_i}(p)} = -\frac{1+aX+bY}{c} \quad (6)$$

$$\text{normal}_{d_{I_i}(p)} = [a, b, c]^T \quad (7)$$

式中,  $(X, Y, \text{depth}_{d_{I_i}(p)})$  是  $I_i(p)$  与  $F_p$  的交点坐标。图 8 展示了  $I_i$  经过初始化得到的深度信息。对于富纹理丰富区域, 规则初始化能够提供接近真值的初值, 而对于弱纹理区域, 规则初始化则可以根据附近的稀疏点提供一个近似解。

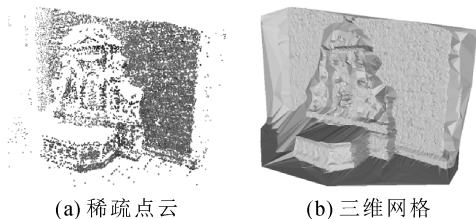


图 7 Delaunay 三角网  
Fig.7 Delaunay triangle mesh

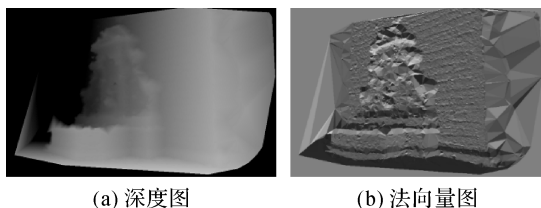


图 8 深度初始化图和法向量初始化图  
Fig.8 The initial depth map and normal map

经过深度信息规则初始化后, 能够给多视密集匹配算法提供更好的初始信息, 更利于 PatchMatch 计算深度信息。在本文中, 由于深度信息规则初始化计算量相对较小, 本文直接在 CPU 上实现了该算法。

### 2.4 金字塔红黑板并行传播

红黑板传播算法解决了并行传播问题, 但该算法仅能在小范围内传播深度信息, 传播效率较低。文献[9]通过增加迭代次数或相邻像素的数量来提高传播效率, 但该方法降低了运行效率。本文提出金字塔红黑板并行传播算法, 由粗到细地进行深度信息传播, 能在不增加计算量的同时, 提高并行传播效率。本文算法包括构造金字塔红黑板和并行传播。

在构造金字塔红黑板时, 首先在原始影像上构建  $L_0$  红黑板, 如图 9 中  $L_0$  所示, 像素按照相邻关系分为“红”像素和“黑”像素; 然后依次构建金

字塔红黑板, 偶数层红黑板使用上一层的“红”像素构成, 并根据上下左右 4 个方向上的相邻关系建立红黑板, 如图 9 中,  $L_2$  使用  $L_1$  中的“红”像素构建而成; 而奇数层红黑板则使用上一层的“黑”像素构成, 并根据对角线上的相邻关系建立红黑板, 如图 9 中,  $L_1$  使用  $L_0$  中的“黑”像素构建而成。在金字塔红黑板中, 每层的像素数量仅有上一层 1/2 的像素, 并且像素间的距离也越来越远。

在进行并行传播时, 从金字塔红黑板的高层到低层, 逐层进行深度信息传播; 在每一层中, 依次进行“红”像素和“黑”像素的深度信息并行传播, 即分别将“黑”像素的深度信息传播到“红”像素或将“红”像素的深度信息传播到“黑”像素; 在进行并行传播时, 偶数层红黑板使用上下左右 4 个相邻像素向当前像素传播深度信息, 奇数层红黑板使用对角线方向上 4 个相邻像素向当前像素传播深度信息。在图 10 中展示  $L_3 \rightarrow L_0$  进行深度信息更新时, 某个“黑”像素相邻的 4 个“红”像素向当前像素传播深度信息。在金字塔的底层深度信息传播较近, 而在金字塔高层深度信息传播较远。

假如记影像长宽为  $a$  和  $b$ 。对于文献[9]的算法, 2 次红黑板交替需要  $8ab$  次传播, 最多将一个像素的深度信息传播到 40 个不同的像素上, 如图 11(a) 所示。本文算法如仅使用 2 层金字塔, 一次完整的金字塔迭代需要  $6ab$  次传播, 最多能将一个像素的深度信息传播到 60 个不同的像素上, 如图 11(b) 所示。本文算法只需要文献[9]算法 2/3 的计算量, 却可以增加 50% 的传播范围。随着金字塔层数的增加, 本文算法的传播效率还能进一步提高。图 11 中展示了文献[9]算法的传播范围和本文的传播范围, 其中颜色的深到浅表示不同的传播时间, 颜色越浅被传播到的时间越晚。

在编程实现时, 本文充分利用了 CUDA 中的线程机制, 首先申请与参考影像像素数量相等的线程; 然后按照金字塔红黑板中的“红”像素或“黑”像素, 激活对应的线程; 最后在激活的线程上进行深度信息的并行传播。

### 2.5 深度信息随机搜索

深度信息随机搜索的核心思路是通过随机搜索方式, 在当前像素的深度信息附近寻找更优解; 如果新的深度信息降低了匹配代价, 则更新当前像素的深度信息。虽然单个像素找到更优解的概率较低, 但是由于目前影像的像素数量通常在百

万级以上,影像中总会有像素得到更优的深度信息。

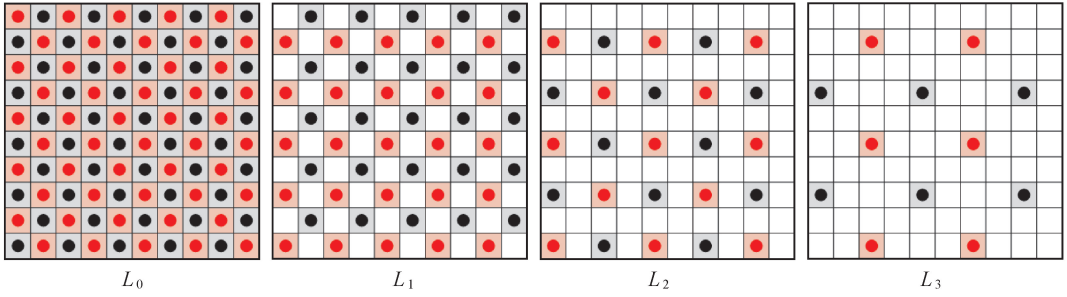


图9 金字塔红黑板

Fig.9 Pyramid red-black board

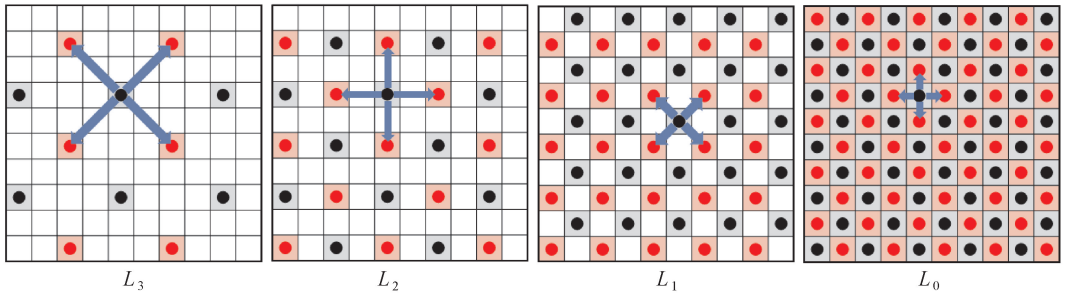
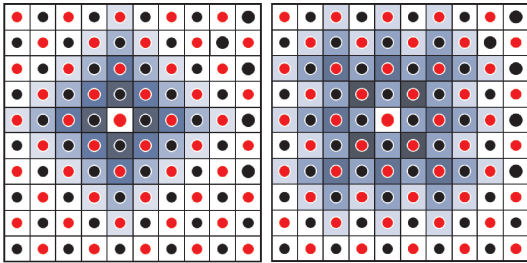


图10 深度信息传播

Fig.10 The propagation of depth information



(a) 文献[9]传播算法 (b) 本文传播算法

图11 传播效率比较

Fig.11 The comparison of propagations' efficiency

在编程实现时,本文首先使用 curandGenerate Uniform 函数,为当前像素生成 3 个 0~1 范围的随机数,得到深度信息的搜索结果为  $(\Delta d, \Delta \mathbf{normal})$ ;然后根据当前参考影像的深度信息范围  $(\text{depth}_{\text{range}}, \mathbf{normal}_{\text{range}})$  和当前像素的深度信息  $(d_{\text{depth}}, \mathbf{d}_{\text{normal}})$ ,就可以得到新的深度信息  $(d_{\text{depth}} + \Delta d \times \text{depth}_{\text{range}}, \mathbf{d}_{\text{normal}} + \Delta \mathbf{normal} \cdot \mathbf{normal}_{\text{range}})$ ;最后再使用匹配代价判断新的深度信息是否合理即可。

### 2.6 深度信息融合

深度信息融合的主要作用是将多个深度图融合成一个统一场景,从而得到整个场景的密集点云。本文直接使用了文献[4]的方法。首先将  $I_i$

上一点  $d \in D_{I_i} \cap D_{I_j}$  投影到的相邻影像  $I_j$  上  $d_{I_j}$ ,并计算其投影深度  $\text{depth}(I_j, d)$ ;然后获取  $I_j$  的深度图  $\text{depth}_{I_j}$  在投影点处的深度  $\text{depth}(I_j, d_{I_j})$ ;最后按式(8)比较两个深度的差异,若超出阈值则剔除该点

$$\frac{|\text{depth}(I_j, d) - \text{depth}(I_j, d_{I_j})|}{\text{depth}(I_j, d_{I_j})} \leq t \quad (8)$$

### 3 试验与分析

为测试本文算法的适用性,本文共使用了 DTU<sup>[22]</sup>、Strecha<sup>[23]</sup> 和 Vaihigen<sup>[24]</sup> 3 个数据集,每组数据都包含影像和对应场景的真实点云。DTU 是实验室数据,其采集环境稳定可控,目标到影像距离较近;Strecha 是地面近景数据,其采集环境为室外,目标到影像距离适中;Vaihigen 是航空数据,其使用搭载飞行平台上的设备采集,距离目标较远。除此之外,本文还使用 DJI 无人机在城区采集了一批实际数据,对本文算法进行测试。

为测试本文算法的性能,本文与 PMVS、MVE、OpenMVS 和 Gipuma 等开源框架进行了对比。PMVS 是一种基于稀疏点,通过区域生长来逐步获取整个场景的密集点云的算法<sup>[3]</sup>;MVE

是基于深度图的重建方法,但其通过稀疏点区域生长计算深度图,最后通过深度图融合得到密集点云<sup>[6]</sup>; OpenMVS 是文献[8]的方法,基于 PatchMatch 使用 CPU 多线程计算进行高效地多视密集匹配的算法; Gipuma 是文献[9]提出的算法,使用 GPU 对 PatchMatch 进行多视密集匹配。

为测试初始化对本文算法的影响,本文对比分析了使用 10% 稀疏点和使用 100% 稀疏点对本文算法进行初始化后的重建结果的差异。

最后,本文所涉及的试验均在 Ubuntu16.04 64 bit 平台上展开,使用的 CPU 为 Intel i7-6700, RAM 大小为 32 GB, GPU 为 NVIDIA GeForce GTX 1060。本文算法首先将参考影像和相邻影像都转换成灰度图,在运行时仅需要把相关影像和深度信息(深度图和向量图)拷贝到显存中即可,对显存大小需求较低。假如记参考影像和相邻影像的长宽为  $a$  和  $b$ , 相邻影像数目为  $n$ , 使用的显存为  $ab(n+1+2) \times 4/10^9$  GB。如果影像长宽为  $5000 \times 5000$  像素, 相邻影像数目为 4, 那么所需显存仅为 0.7 GB。

值得注意的是,所有算法在试验时都使用默认参数,并没有针对不同的数据集而进行相应调整。本文算法默认使用 4 张相邻影像和 4 层红黑板,并进行 3 次迭代。

### 3.1 评价标准

本文的评价指标有重建精度、重建完整度和重建耗时。重建精度通过计算重建点云到真实点

云的最近距离的平均值获取;重建完整度通过计算真实点云到重建点云最近距离小于阈值的点占整个真实点云的百分比获取;重建耗时通过记录开始重建到输出最终结果的时间间隔获取。需要说明的是,上述评价方法并不能准确刻画重建精度和重建完整度,只能提供一个近似解<sup>[25]</sup>。

## 3.2 试验结果及分析

### 3.2.1 DTU 数据

DTU 数据集为实验室数据,其影像数据是通过固定在机械臂上的相机在可控环境下拍摄得到的,分辨率为  $1200 \times 1600$  像素,其真实点云使用结构光设备采集。本文使用了 Scan1、Scan6 和 Scan9 这 3 组数据,每组数据都有 49 张影像。进行重建时,所有影像都进行了 1 次降采样;完成重建后,重建点云都进行了间隔 1 mm 的采样。评价重建精度和重建完整度时,距离大于 3 mm 的点被视为外点。

图 12 为各个算法重建的密集点云,表 1 为各个算法的重建结果评价指标。由于该数据质量较好,各个算法的重建精度和重建完整度相当。另外,虽然本文算法 GPU-PatchMatch(10%) 仅仅使用了 10% 的稀疏点进行初始化,但其重建效果与本文算法 GPU-PatchMatch(100%) 的效果十分接近,表明本文算法对稀疏点的依赖并不明显。在重建耗时上,本文算法与传统 CPU 方法相比至少有 7 倍以上效率提升,与 Gipuma 相比也至少有 4 倍效率提升。

表 1 重建性能

Tab.1 The performance of different algorithms

算法	Scan1			Scan6			Scan9		
	精度 /mm	完整度 /(%)	耗时 /s	精度 /mm	完整度 /(%)	耗时 s	精度 /mm	完整度 /(%)	耗时 /s
PMVS	0.80	91.11	462	0.87	77.89	489	0.99	71.61	504
MVE	0.98	90.88	1553	1.04	78.28	1450	1.15	72.18	1366
OpenMVS	0.73	90.85	396	0.86	79.10	405	0.95	71.97	296
Gipuma	0.90	90.22	195	0.97	77.57	195	1.05	72.61	194
本文算法(10%)	0.83	90.73	<b>41</b>	0.97	78.14	<b>41</b>	1.10	71.70	<b>38</b>
本文算法(100%)	0.81	91.10	<b>43</b>	0.96	78.31	<b>43</b>	1.09	72.52	<b>42</b>

### 3.2.2 Strecha 数据

Strecha 数据集为地面近景数据,其影像使用 Canon D60 采集,分辨率为  $3072 \times 2048$  像素;其真实点云使用 Zoller + Forhlich IMAGER 5003 采集。本文使用了该数据集中的 Fountain 数据和 Herzjesu 数据,其中 Fountain 有 11 张影像,

Herzjesu 有 8 张影像。进行重建前,所有影像都进行了 1 次降采样;完成重建后,重建点云都进行了间隔 0.02 m 的采样;评价重建精度和重建完整度时,距离大于 0.05 m 的点被视为外点。

图 13 为各个算法重建的密集点云,表 2 为各个算法的重建结果评价指标。整体而言,在重建

精度和重建完整度上各个算法间的差异并不大;但 Gipuma 算法的重建完整度较低,原因是其对参数十分敏感,不同的参数对重建质量有较大的

影响。在重建耗时上,本文算法效率最高,与传统 CPU 方法相比至少有 10 倍效率提升,与 Gipuma 相比也至少有 2.5 倍效率提升。

表 2 Fountain 与 Herzjesu 数据重建指标

Tab.2 The performance of different algorithms for Fountain and Herzjesu data sets

算法	Fountain			Herzjesu		
	精度/m	完整度/(%)	耗时/s	精度/m	完整度/(%)	耗时/s
PMVS	0.014	55.38	539	0.015	51.00	398
MVE	0.015	44.03	847	0.017	36.80	519
OpenMVS	0.015	61.24	325	0.015	54.11	224
Gipuma	0.013	41.21	105	0.015	30.21	53
本文算法(10%)	0.015	56.06	27	0.016	49.86	20
本文算法(100%)	0.016	59.24	28	0.016	52.43	21

### 3.2.3 Vaihingen 数据

Vaihingen 数据为航空影像,共有 20 张使用 Intergraph/ZI DMC 采集的影像,分辨率为  $7680 \times 13\ 824$  像素;真实点云使用 Leica ALS50 采集激光点云数据。进行重建前,所有影像都进行了 2 次降采样;完成重建后,重建点云都进行了间隔 0.2 m 的采样;评价重建精度和重建完整度时,距离大于 2 m 的点被视为外点。由于 Gipuma 无法处理航空影像,因此并没有获取其重建结果。

图 14 为各个算法重建的高程图,表 3 为各个算法的重建结果评价指标。在重建精度上,各个算法间的差异较小,均在 0.5 m 左右。在重建完整度上,各个算法间有一定差异,主要原因是像方与物方的距离较远,对算法的稳健性要求更高;本文算法基于 PatchMatch 算法,不需要设置参数,稳健性比 PMVS、MVE 更高。在重建耗时上,本文算法效率最高,在 4 min 左右就完成了重建,与传统 CPU 方法相比至少有 8 倍效率提升。

表 3 Vaihingen 数据重建指标

Tab.3 The performance of different algorithms for Vaihingen data set

算法	精度/m	完整度/(%)	耗时/min
PMVS	0.60	62.48	60.5
MVE	0.51	52.61	99.5
OpenMVS	0.65	64.97	33.5
本文算法(10%)	0.63	71.04	3.5
本文算法(100%)	0.60	72.31	3.9

### 3.2.4 实际数据

最后,为测试本文算法在实际场景下的重建效果和性能,使用 DJI 无人机采集了 188 张分辨率为  $5472 \times 3648$  像素的下视影像,样张见图 15(a)。通

过 SfM 获取所有影像的外方位元素后,再对原始影像进行 2 次降采样,本文算法仅需 5 min 即可完成整个场景的密集点云计算,如图 15(b)所示,在富纹理区域(建筑、操场)和低纹理区域(路面)都有较好的重建效果。这表明本文算法能够应用到生产实际,满足快速三维重建的需求。

## 4 结论和展望

针对目前多视密集匹配效率较低的问题,本文提出了 GPU-PatchMatch 多视密集匹配算法。该算法首先使用稀疏场景进行深度信息规则初始化,然后使用金字塔红黑板进行深度信息并行传播。本文在 3 组不同类型的数据上进行了多方面的测试。试验结果表明,本文算法在重建精度和重建完整度与传统算法相当,并且在重建效率上有较大提高。与 CPU 方法(PMVS、OpenMVS、MVE)相比至少有 7 倍以上效率提升,与 GPU 方法(Gipuma)相比也有 2.5 倍效率提高。同时,本文还在试验中测试了稀疏点数量的影响,分别选用 10% 和 100% 稀疏点进行多视密集匹配,两者的重建结果并没有出现较大差异,表明本文算法对稀疏场景并没有强依赖。在后续研究中,将考虑引入更多先验信息对 PatchMatch 进行初始化,例如线特征或特殊目标(水面、玻璃表面等),更加充分且合理地使用场景信息;另一方面则考虑将深度学习的方法引入多视密集匹配中,使用更加合理的匹配代价计算方法,从而进一步提高场景重建的质量。

**致谢:**感谢 ISPRS/EuroSDR 项目及斯图加特大学摄影测量研究所提供的倾斜影像试验数据。



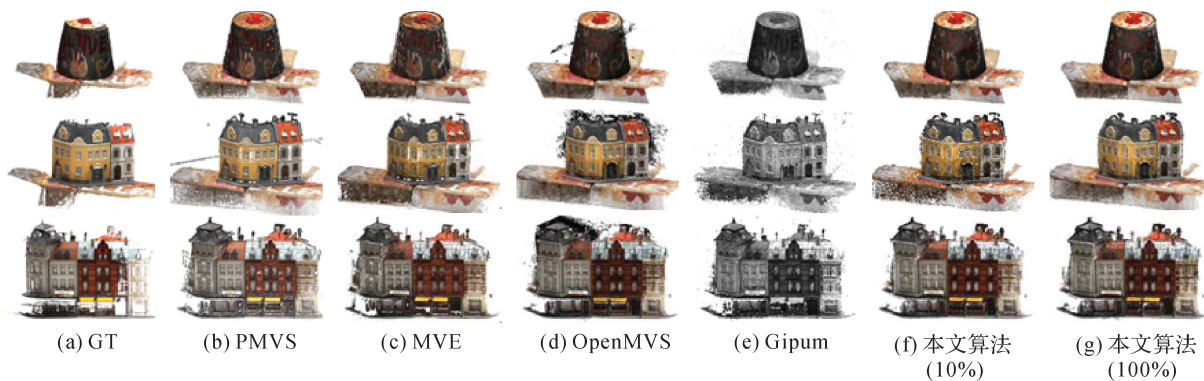


图 12 重建结果

Fig.12 Reconstruction result

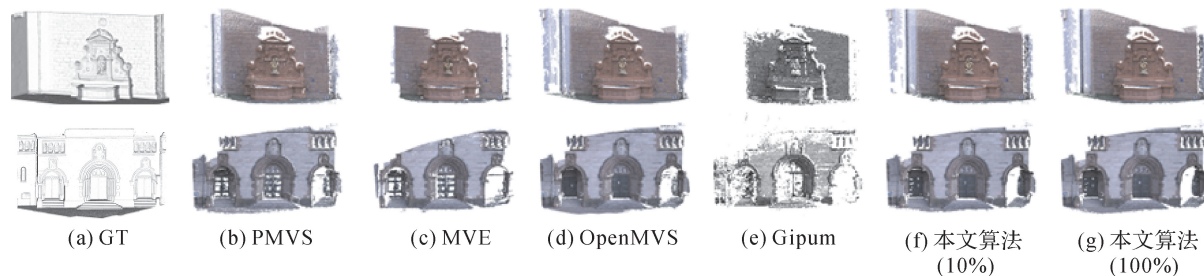


图 13 Fountain 与 Herzjesu 数据重建结果

Fig.13 Reconstruction result for Fountain and Herzjesu data sets

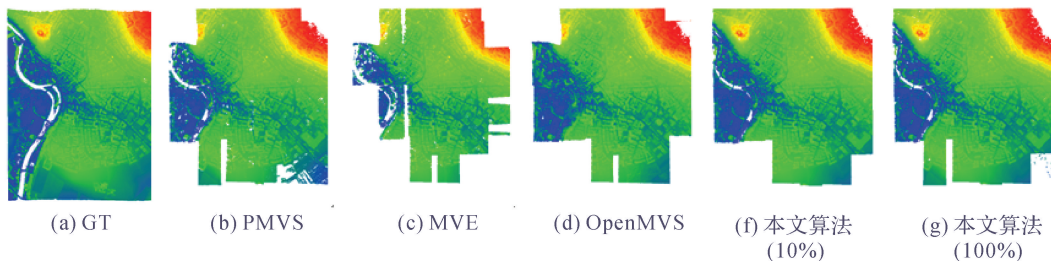


图 14 Vaihingen 数据重建结果

Fig.14 Reconstruction result for Vaihingen data set

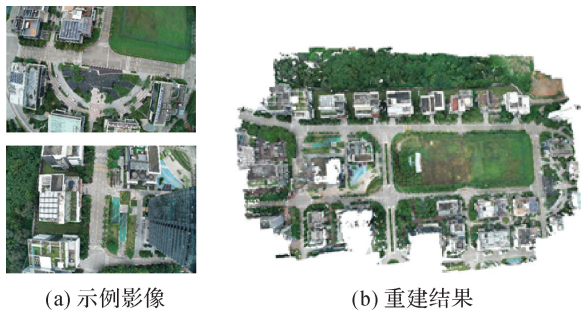


图 15 样张与重建结果

Fig.15 Sample picture and reconstruction result

参考文献:

[1] SEITZ S M, CURLLESS B, DIEBEL J, et al. A comparison and evaluation of multi-view stereo reconstruction algorithms[C]//Proceedings of 2006 IEEE Computer Society

Conference on Computer Vision and Pattern Recognition. New York; IEEE, 2006; 519-528.

[2] REMONDINO F, SPERA M G, NOCERINO E, et al. State of the art in high density image matching[J]. Photogrammetric Record, 2014, 29(146): 144-166.

[3] FURUKAWA Y, PONCE J. Accurate, dense, and robust multiview stereopsis [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, 32(8): 1362-1376.

[4] 赵红蕊, 陆胜寒. 基于特征尺度分布与对极几何约束的高清影像快速密集匹配方法[J]. 测绘学报, 2018, 47(6): 790-798. DOI: 10.11947/j.AGCS.2018.20170630.

ZHAO Hongrui, LU Shenghan. Dense high-definition image matching strategy based on scale distribution of feature and geometric constrain [J]. Acta Geodaetica et Cartographica Sinica, 2018, 47(6): 790-798. DOI: 10.11947/j.AGCS.2018.20170630.

[5] 王竞雪, 朱庆, 王伟玺. 多匹配基元集成的多视影像密集

- 匹配方法[J]. 测绘学报, 2013, 42(5): 691-698.
- WANG Jingxue, ZHU Qing, WANG Weixi. A dense matching algorithm of multi-view image based on the integrated multiple matching primitives[J]. Acta Geodaetica et Cartographica Sinica, 2013, 42(5): 691-698.
- [6] GOESELE M, SNAVELY N, CURLESS B, et al. Multi-view stereo for community photo collections[C]// Proceedings of 2007 IEEE 11th International Conference on Computer Vision. Rio de Janeiro: IEEE, 2007: 1-8.
- [7] ROTHERMEL M, WENZEL K, FRITSCH D, et al. SURE: photogrammetric surface reconstruction from imagery[C]// Proceedings of LC3D Workshop. Berlin: ISPRS, 2012: 1-9.
- [8] SHEN Shuhan. Accurate multiple view 3D reconstruction using patch-based stereo for large-scale scenes[J]. IEEE Transactions on Image Processing, 2013, 22(5): 1901-1914.
- [9] GALLIANI S, LASINGER K, SCHINDLER K. Massively parallel multiview stereopsis by surface normal diffusion [C]// Proceedings of 2015 IEEE International Conference on Computer Vision. Santiago: IEEE, 2015: 873-881.
- [10] 闫利, 费亮, 陈长海, 等. 利用网络图进行高分辨率航空多视影像密集匹配[J]. 测绘学报, 2016, 45(10): 1171-1181. DOI: 10.11947/j.AGCS.2016.20160068.
- YAN Li, FEI Liang, CHEN Changhai, et al. A multi-view dense matching algorithm of high resolution aerial images based on graph network[J]. Acta Geodaetica et Cartographica Sinica, 2016, 45(10): 1171-1181. DOI: 10.11947/j.AGCS.2016.20160068.
- [11] 吴军, 姚泽鑫, 程门门. 融合 SIFT 与 SGM 的倾斜航空影像密集匹配[J]. 遥感学报, 2015, 19(3): 431-442.
- WU Jun, YAO Zexin, CHENG Menmen. Airborne oblique stereo image dense matching by integrating SIFT and SGM algorithm[J]. Journal of Remote Sensing, 2015, 19(3): 431-442.
- [12] TOLDO R, FANTINI F, GIONA L, et al. Accurate multiview stereo reconstruction with fast visibility integration and tight disparity bounding [C]// Proceedings of 3D Virtual Reconstruction and Visualization of Complex Architectures. Trento: ISPRS, 2013: 243-249.
- [13] KAR A, HÄNE C, MALIK J. Learning a multi-view stereo machine[C]// Proceedings of the 30st Conference on Neural Information Processing Systems. Long Beach: Neural Information Processing Systems Foundation, 2017: 365-376.
- [14] YAO Yao, LUO Zixin, LI Shiwei, et al. MVSNet: depth inference for unstructured multi-view stereo [C]// Proceedings of the 15th European Conference on Computer Vision. Munich: Springer, 2018: 785-801.
- [15] JI Mengqi, GALL J, ZHENG Haitian, et al. SurfaceNet: an end-to-end 3D neural network for multiview stereopsis [C]// Proceedings of 2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017: 2326-2334.
- [16] BARNES C, SHECHTMAN E, FINKELSTEIN A, et al. PatchMatch: a randomized correspondence algorithm for structural image editing [J]. ACM Transactions on Graphics (TOG), 2009, 28(3): 24.
- [17] BLEYER M, RHEMANN C, ROTHER C. PatchMatch stereo-stereo matching with slanted support windows[C]// Proceedings of 2011 22nd British Machine Vision Conference. Dundee: BMVA, 2011: 1-11.
- [18] HIRSCHMULLER H. Stereo processing by semiglobal matching and mutual information[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 30(2): 328-341.
- [19] YOON K J, KWEON I S. Adaptive support-weight approach for correspondence search [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(4): 650-656.
- [20] TOLA E, LEPETIT V, FUA P. DAISY: an efficient dense descriptor applied to wide-baseline stereo[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, 32(5): 815-830.
- [21] BOISSONNAT J D, DEVILLERS O, PION S, et al. Triangulations in CGAL[C]// Proceedings of the 16th Annual Symposium on Computational Geometry. Hong Kong, China: ACM, 2000: 11-18.
- [22] AANAES H, JENSEN R R, VOGIATZIS G, et al. Large-scale data for multiple-view stereopsis [J]. International Journal of Computer Vision, 2016, 120(2): 153-168.
- [23] STRECHA C, VON HANSEN W, VAN GOOL L, et al. On benchmarking camera calibration and multi-view stereo for high resolution imagery [C]// Proceedings of 2008 IEEE Conference on Computer Vision and Pattern Recognition. Anchorage: IEEE, 2008: 1-8.
- [24] CRAMER M. The DGPF-test on digital airborne camera evaluation-overview and test design[J]. Photogrammetrie-Fernerkundung-Geoinformation, 2010, 2010(2): 73-82.
- [25] ALTMAN S, XIAO Wen, GRAYSON B. Evaluation of LOW-COST terrestrial Photogrammetry for 3D reconstruction of complex buildings[C]// Proceedings of ISPRS Geospatial Week 2017. Wuhan: Copernicus GmbH, 2017: 199-206.

(责任编辑:丛树平)

收稿日期: 2018-10-10

修回日期: 2019-06-15

第一作者简介: 邓非(1976—),男,博士,教授,研究方向为摄影测量与计算机视觉。

First author: DENG Fei(1976—), male, PhD, professor, majors in photogrammetry and computer vision.

E-mail: fdeng@sgg.whu.edu.cn

通信作者: 颜青松

Corresponding author: YAN Qingsong

E-mail: yanqs\_whu@whu.edu.cn