

基于图像的三维模型重建

稠密点云重建（上）



✓ 稠密点云的获取方式

- ✓ Lidar 扫描
- ✓ Kinect
- ✓ 结构光
- ✓ 基于图像的方法-Multi-view Stereo

✓ 基础知识

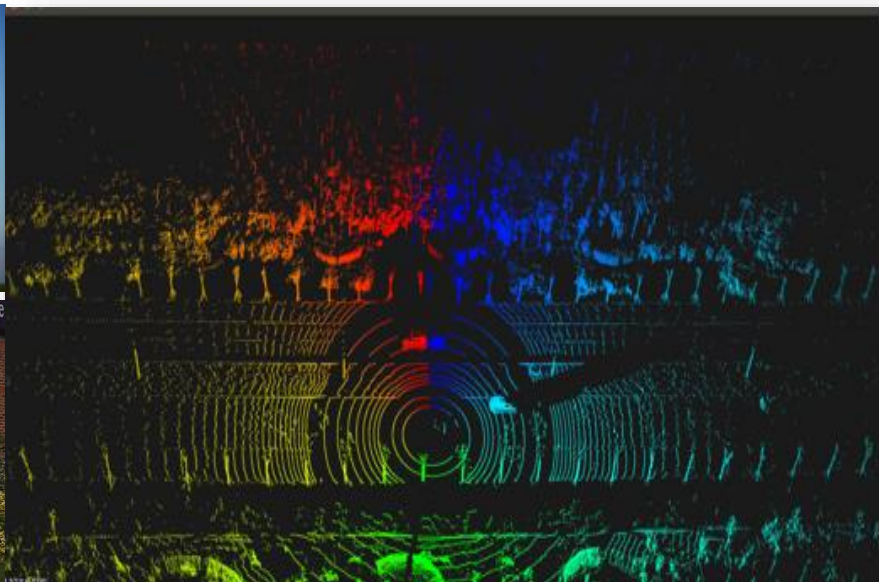
- ✓ 极线搜索
- ✓ 光度一致性约束
- ✓ 可视性约束

✓ 多视角立体技术

- ✓ 基于体素的方法
- ✓ 基于深度图融合的方法
- ✓ 基于空间patch 的方法

稠密点云的获取方式

LiDAR扫描

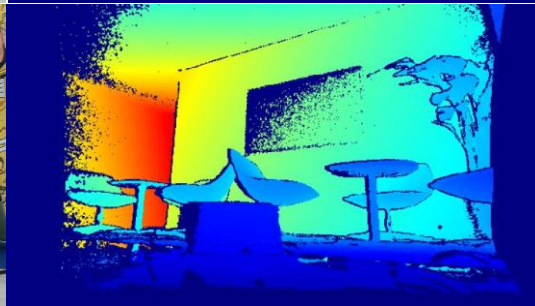
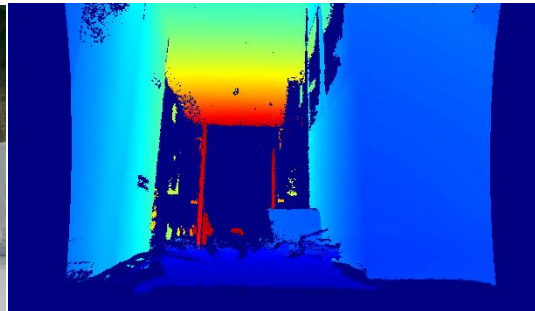


优势：精度高（毫米级别），效率高，有效范围几米到几百米；

劣势：价格昂贵，高反光，玻璃表面，吸收表面。

稠密点云的获取方式

Kinect

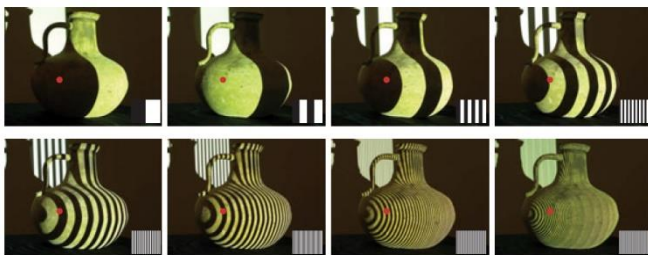
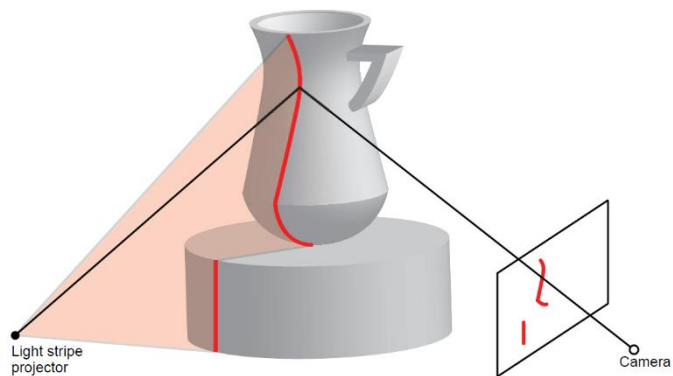


优势：使用方便，价格适中，速度较快；

劣势：精度较低，有效距离短。

稠密点云的获取方式

结构光(Structured Light)



基于灰度编码的二值样式



Hall-Holt and Rusinkiewicz's



Zhang et al 彩色结构光



口腔扫描仪

优势：高精度，高效率；

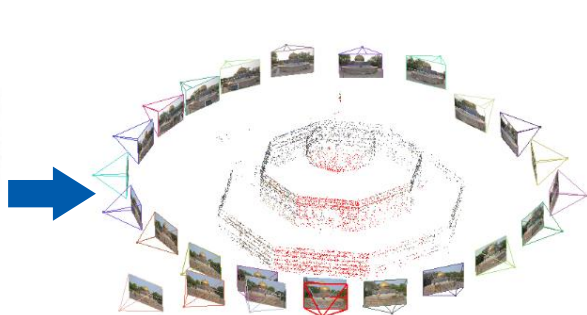
劣势：近距离数据获取。

稠密点云的获取方式

基于图像的方法-Multi-view Stereo



输入无序图像



运动恢复结构(SFM)



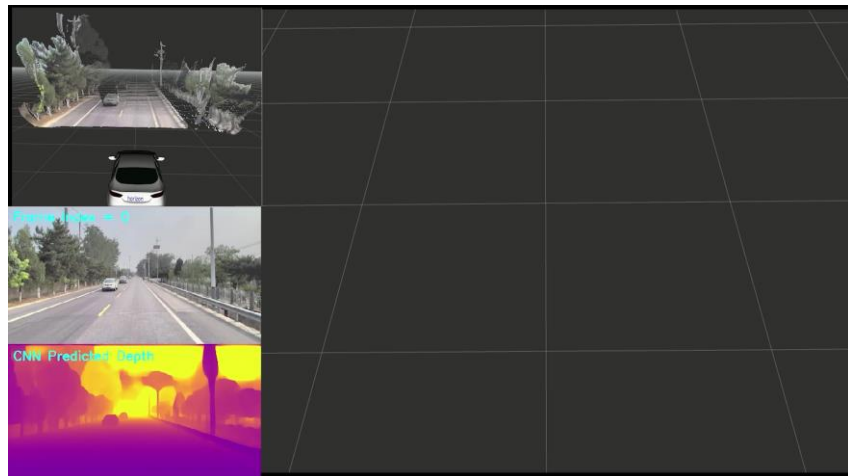
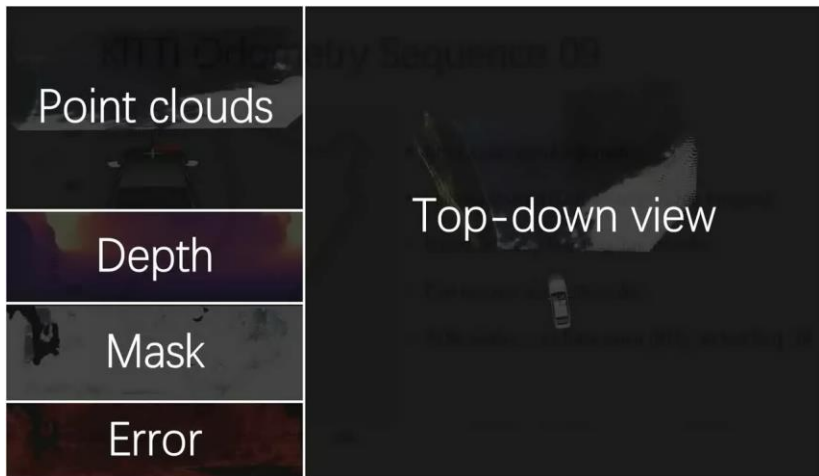
多视角立体重建(MVS)

优势：无源被动式，成本低，图像来源广，精度较高；

劣势：计算速度慢。

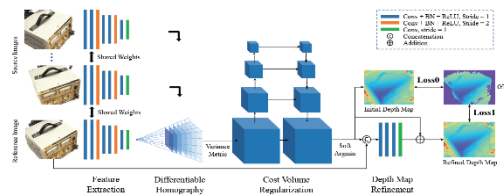
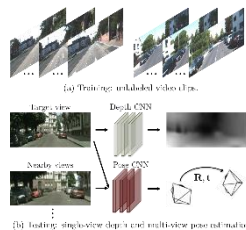
稠密点云的获取方式

基于图像的方法-Learning based methods



优势：数据驱动，算法流程简单，完全稠密的深度图

劣势：耗GPU, 泛化性较差



✓ 稠密点云的获取方式

- ✓ Lidar 扫描
- ✓ Kinect
- ✓ 结构光
- ✓ 基于图像的方法-Multi-view Stereo

✓ 基础知识

- ✓ 极线搜索
- ✓ 光度一致性约束
- ✓ 可视性约束

✓ 多视角立体技术

- ✓ 基于体素的方法
- ✓ 基于深度图融合的方法
- ✓ 基于空间patch 的方法

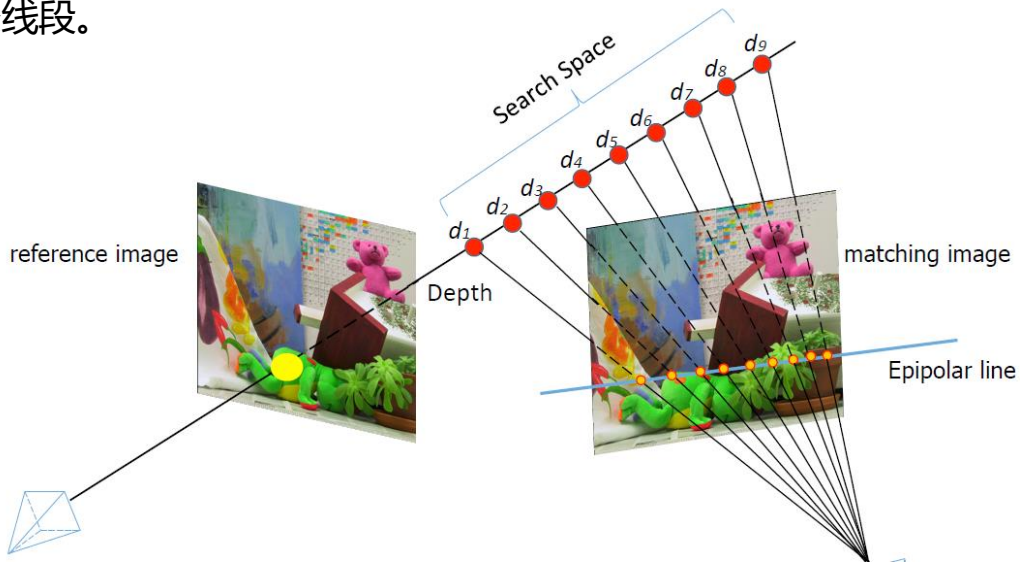
极线搜索

参考图像中的一点对应另一幅图像中的一条线段。

$$\mathbf{x}_2^T \mathbf{F} \mathbf{x}_1 = 0$$

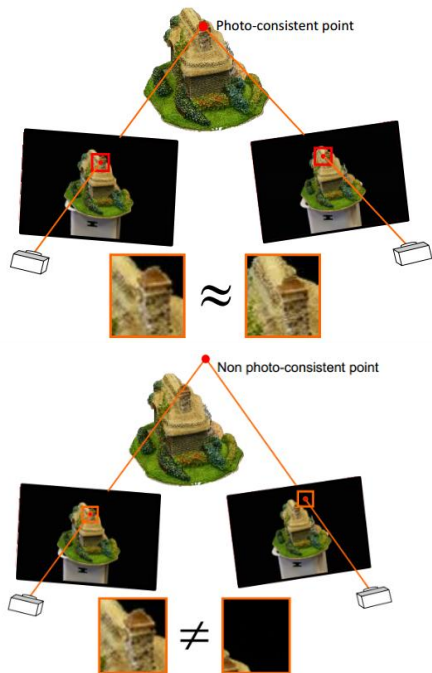
$$l_1 = \mathbf{F}^T \mathbf{x}_2, \quad l_2 = \mathbf{F} \mathbf{x}_1$$

$$\mathbf{F} = \mathbf{K}_2^{-T} [\mathbf{t}]_{\times} \mathbf{R} \mathbf{K}_1^{-1} \quad \text{是基础矩阵}$$



光度一致性假设(Photo-consistency)

同一空间的点在不同视角的投影，应当具有相同的光度，重建的核心在于恢复空间中具有光度一致性的点。



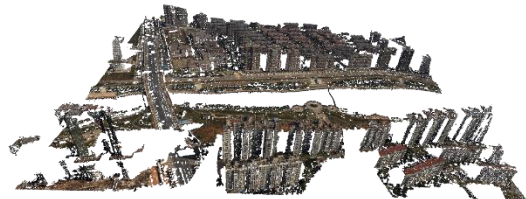
水面



白墙



玻璃



常见的非朗伯面

光度一致性假设的度量方式

常用的计算区域光度一致性的方法有：

- SSD (Sum of Squared Differences): $\rho_{SSD}(f, g) = \|f - g\|^2$
- SAD (Sum of Absolute Differences): $\rho_{SAD}(f, g) = \|f - g\|_1$
- NCC (Normalized Cross Correlation): $\rho_{NCC}(f, g) = \frac{(f - \bar{f}) \cdot (g - \bar{g})}{\delta_f \delta_g}$



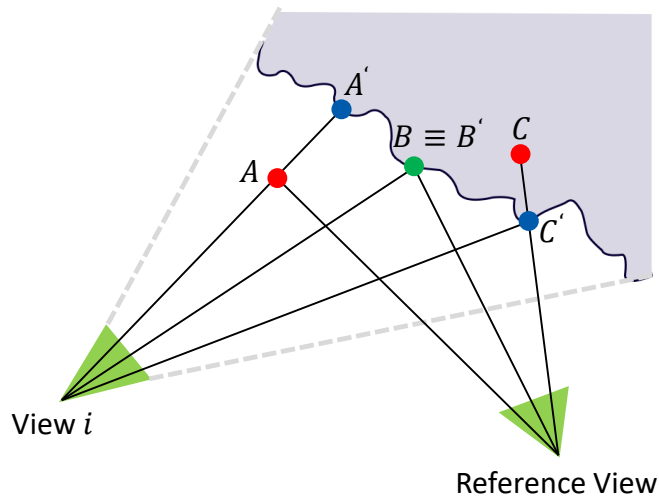
f, g : 特征描述子的特征向量

可视性约束

图像中出现的点不能被遮挡；

重建的点前面不能出现点；

不能出现在物体内部。



红色点：被遮挡的点

✓ 稠密点云的获取方式

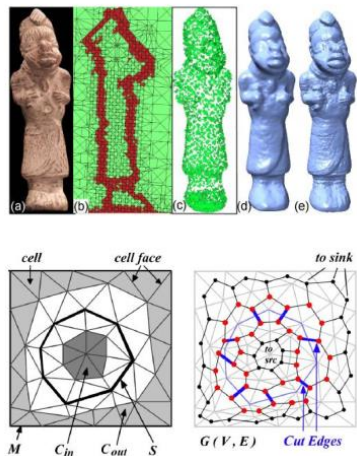
- ✓ Lidar 扫描
- ✓ Kinect
- ✓ 结构光
- ✓ 基于图像的方法-Multi-view Stereo

✓ 基础知识

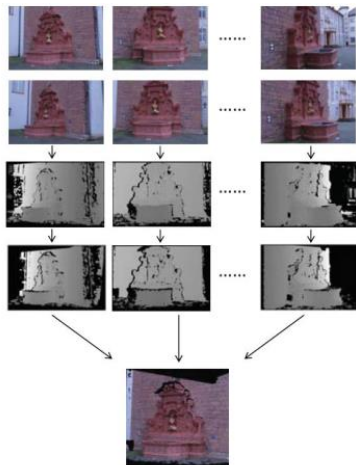
- ✓ 极线搜索
- ✓ 光度一致性约束
- ✓ 可视性约束

✓ 多视角立体技术

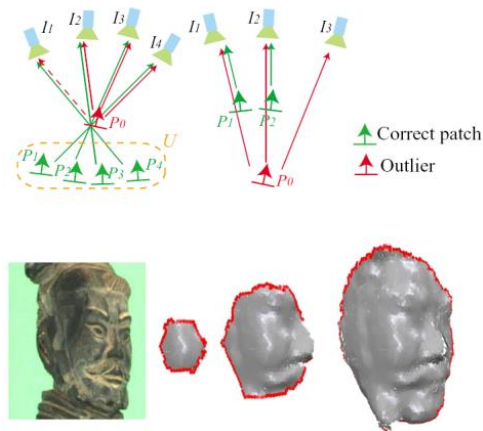
- ✓ 基于体素的方法
- ✓ 基于深度图融合的方法
- ✓ 基于空间patch 的方法



基于体素的方法



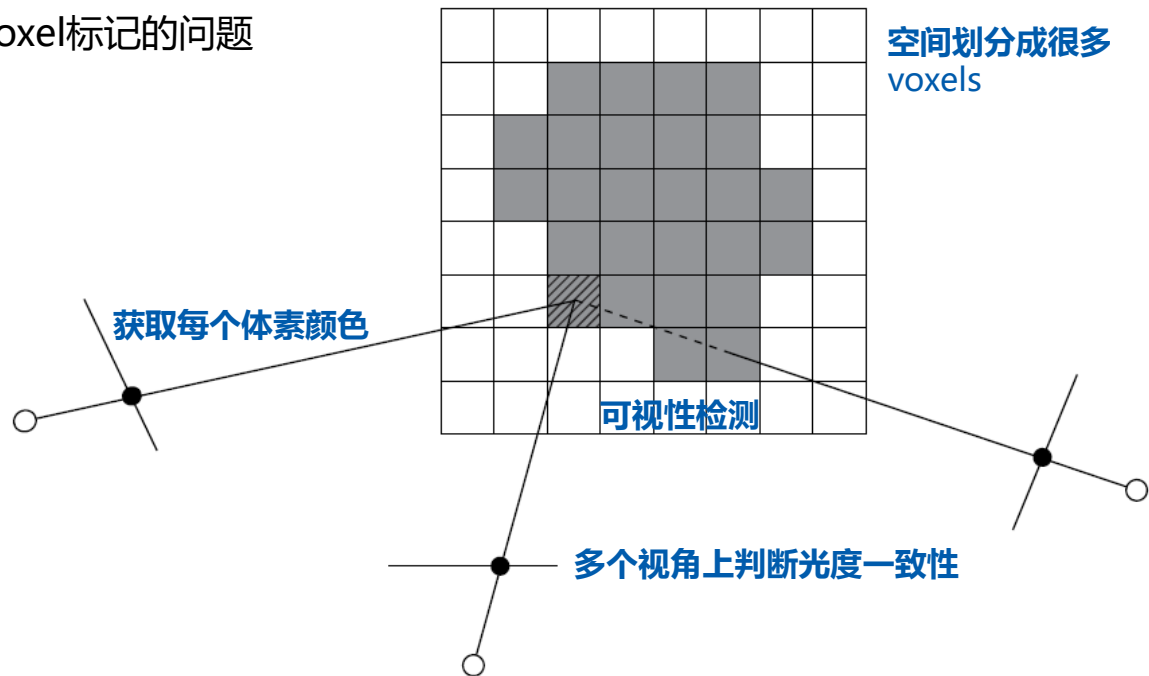
基于深度图融合的方法



基于3D patch扩张的方法

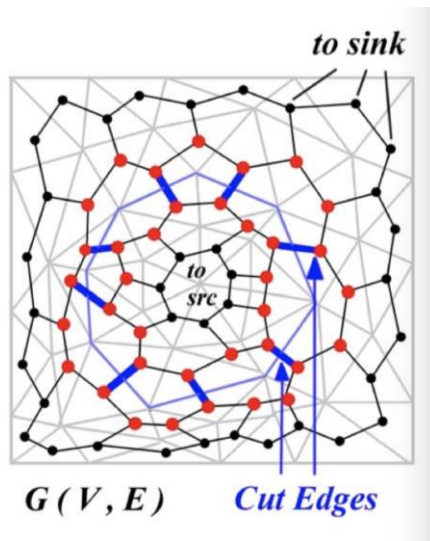
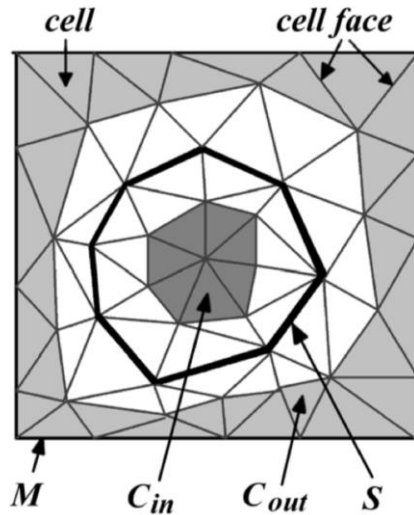
规则的划分

等价于3D空间Voxel标记的问题

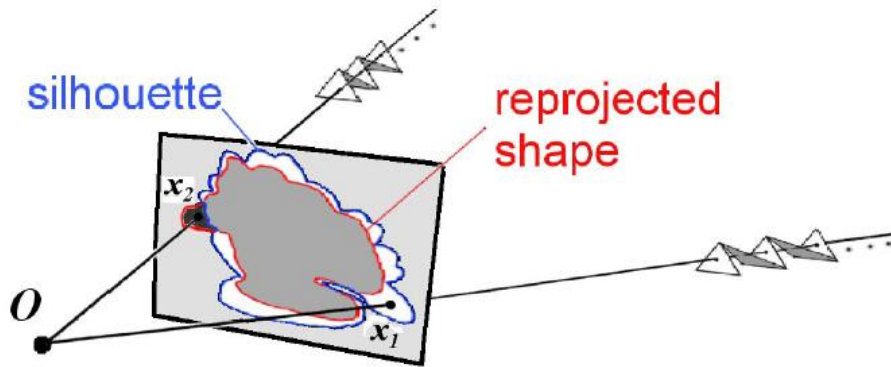


不规则的划分

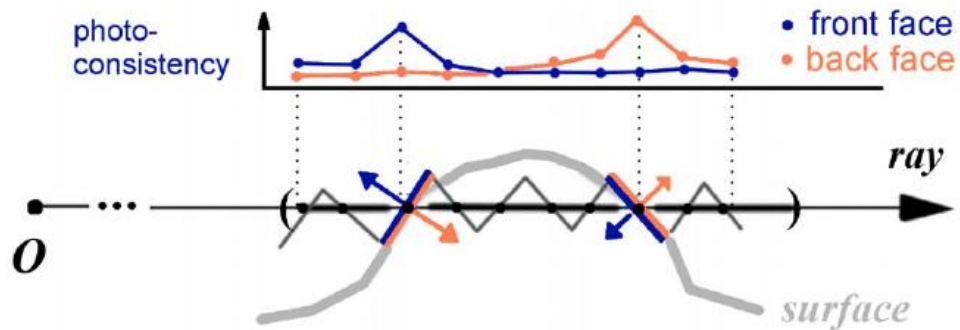
等价于3D空间四面体标记的问题



图像上的约束-剪影约束



图像上的约束-光度一致性约束

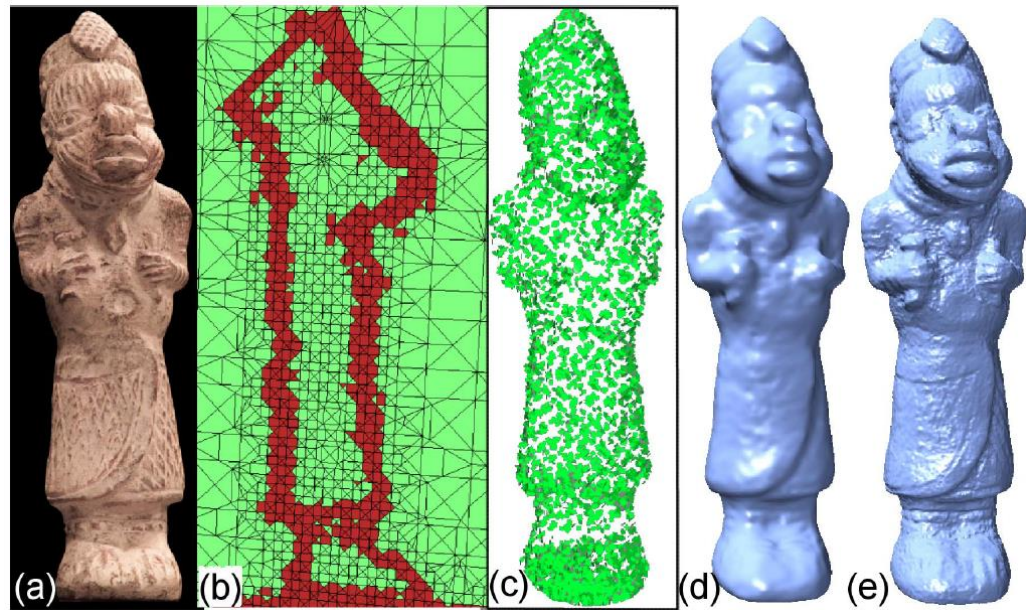


常用的优化模型-典型的MRF离散优化问题

参考图像上每个的像素分配一个标签（内部或者外部）

$$E(f) = E_{data}(f) + E_{smoothness}(f) + E_{visibility}(f)$$

- 数据项-光度一致性假设
- 平滑项-邻域假设
- 可视项-可视性约束



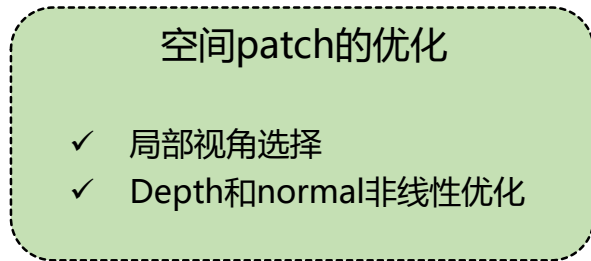
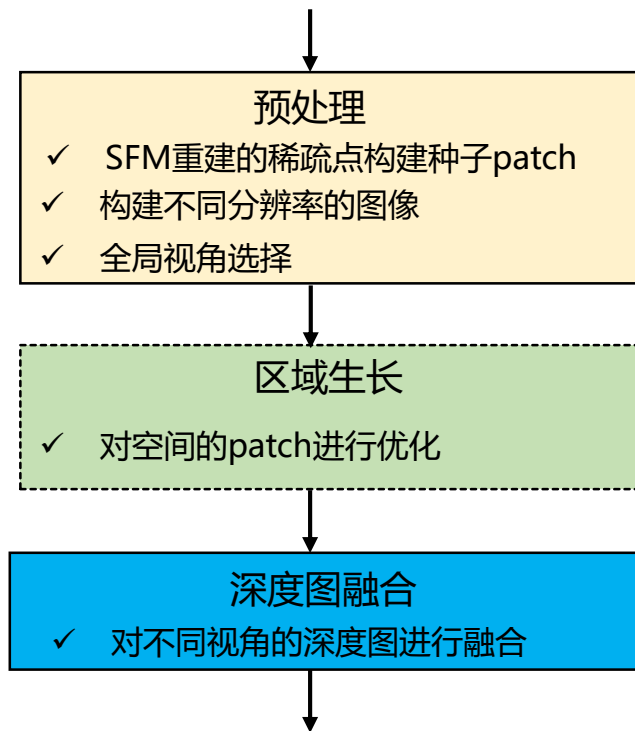
优点:

- 生成规则的点云
- 便于提取物体的平面

缺点:

- 精度受到空间划分分辨率的影响
- 难以处理精度高、规模大的场景仅适用于小场景，单个物体，遮挡较少的场景

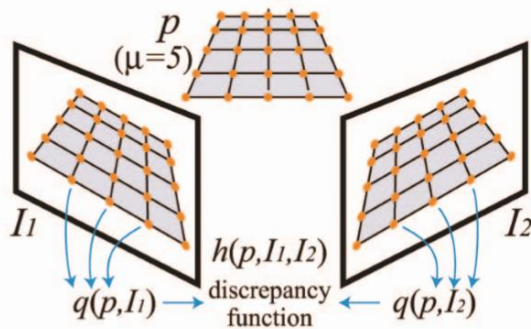
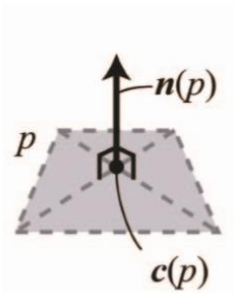
基于深度图融合的方法—整体流程



Patch

以空间中的3D点中心建立一个很小栅格，栅格的分辨率与图像分辨率以及3D点的位置相关

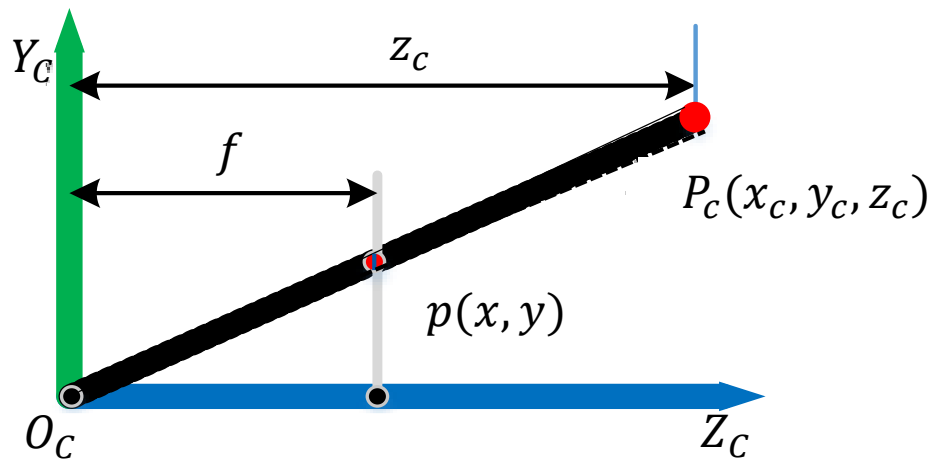
- ✓ Patch的中心点即为3D点的位置
- ✓ Patch的朝向即为3D点的法向量
- ✓ Patch上的每个点可以投影到不同视角中计算NCC用于度量光度一致性
- ✓ Patch实际上是增强了局部特征的描述能力



投影球半径-用于衡量图像的分辨率

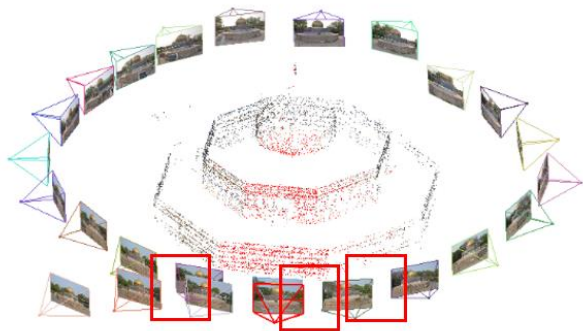
图像上1个像素的宽度对应的三维空间中的物体尺寸

$$r = \frac{z_c}{fa}, a = \max(w, h)$$



预处理

- ✓ SFM重建的稀疏点构建种子patch
- ✓ 构建不同分辨率的图像以处理图像分辨率不同的情况
- ✓ 全局视角选择



SFM稀疏重建的结果作为种子点



图像分辨率不同的情况，patch投影分辨率也不同

全局视角选择

1. 图像具有相同的内容, 外观—通过共享sift特征点的个数
2. 图像具有足够大的视差 (宽基线) --通过视线的夹角
3. 图像具有相似的分辨率-- 通过计算投影球半径
4. 考虑每个特征在参考图像与已选择图像的视线夹角—排它性

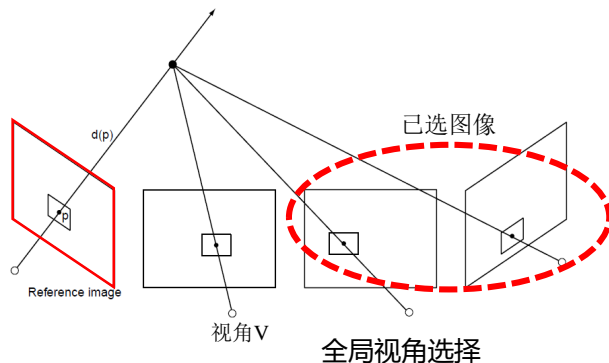
流程:

- A 对于每一个视角V, 计算和参考视角的 $g_R(V)$
- B 选择 $g_R(V)$ 最大的视角
- C 重复上述过程, 直到选择的视角个数满足预设

准则:

三角量测夹角适中

$$g_R(V) = \sum_{f \in \mathbf{F}_V \cap \mathbf{F}_R} \frac{w_N(f)}{\underbrace{\quad}_{\text{共有的特征点越多越好}}} \cdot \underbrace{w_s(f)}_{\text{图像尺度相似}},$$



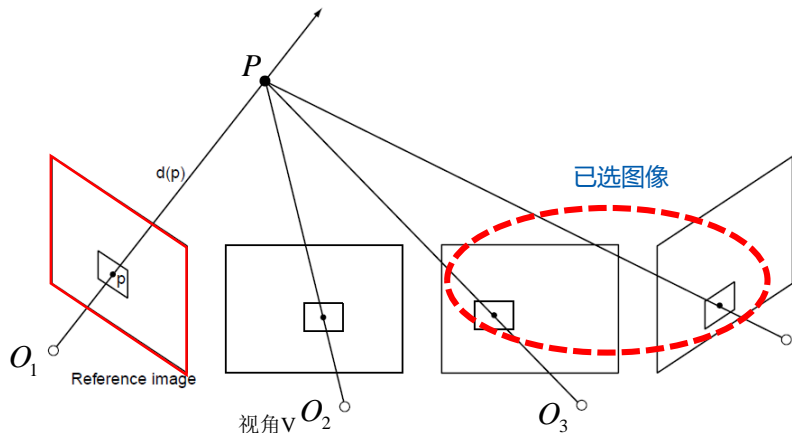
相关视角的选择-局部视角选择

1. NCC值大于一定的阈值
2. 和已经选择的视角的极平面 (PO_1O_3 和 PO_1O_2)
要足够分散(不共面)

$$l_R(V) = g_R(V) \cdot \prod_{V' \in A} \underbrace{w_e(V, V')}_{\text{和已选视角间的极平面夹角}}$$

流程:

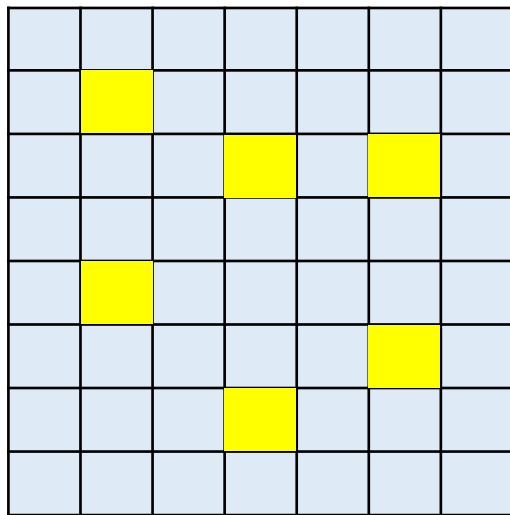
- A 对于每一个视角V,计算和参考视角的 $l_R(V)$
- B 选择 $l_R(V)$ 最大的视角
- C 重复上述过程, 直到选择的视角个数满足预设



从候选视角中选择active 视角

区域生长流程

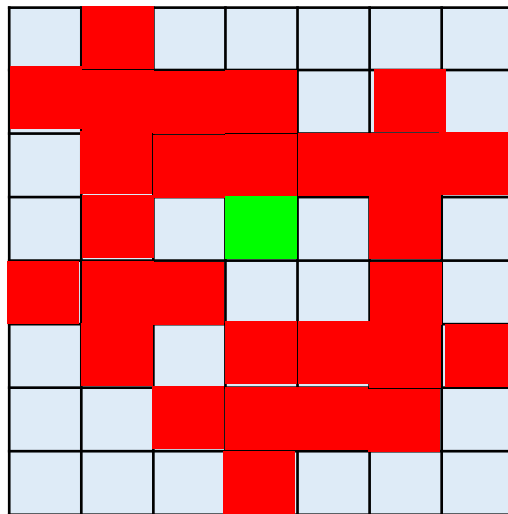
初始阶段的种子点选取



■ 初始的种子点

对种子点进行优化后添加到队列中，并按照置信度进行排序

中间过程的种子点选取



■ 当前优化完的种子点 ■ 已经重建了的种子点

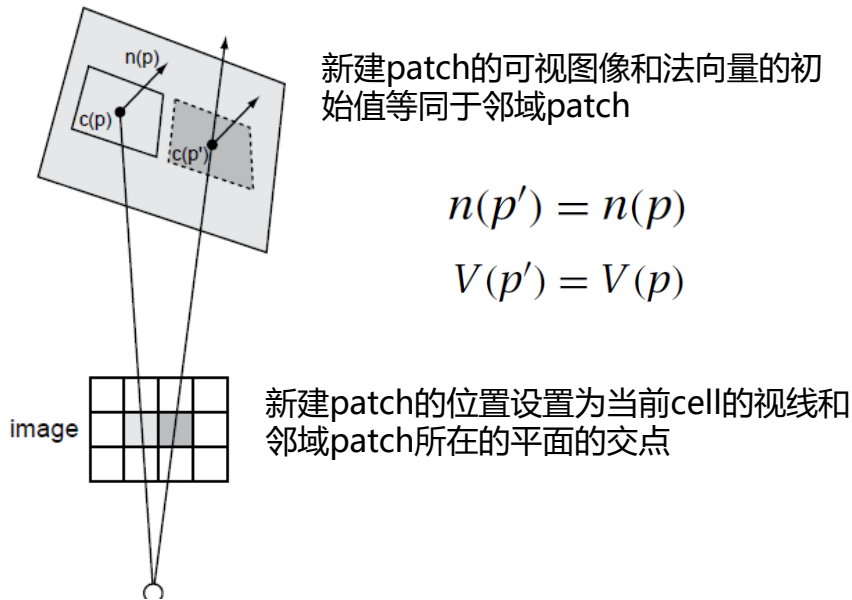
两类邻域点将被选为种子点：

- 1) 没有重建过的点
- 2) 置信度高于已经重建的点

区域生长法扩张

1. 重建的置信度建立优先级队列;
2. 从初始的稀疏特征点开始深度估计;
3. 对每个种子点进行非线性深度优化;
4. 每次优化完后判断以下两种情况, 将邻域像素添加到队列中:

- 1) 邻域没有深度值;
- 2) 当前像素的置信度值高于邻域像素一定范围 (0.05)



深度值非线性优化

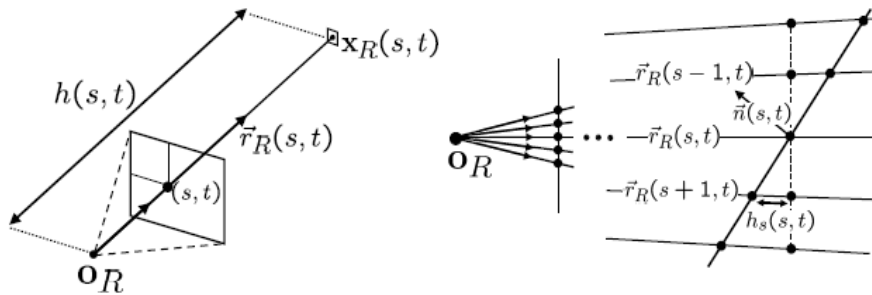
中心点的三维坐标

$$\mathbf{X}_R(s, t) = \mathbf{O}_R + h(s, t)\vec{\mathbf{r}}_R(s, t)$$

Patch中每个点的三维坐标

$$\mathbf{X}_R(\mathbf{s} + \mathbf{i}, t + j) = \mathbf{O}_R + \vec{\mathbf{r}}_R(s, t)(h(s, t) + i * h_s + j * h_t)$$

h_s 和 h_t 反应了patch的法向量。



深度值非线性优化

颜色模型

为邻域中每个视角 K 分配一个颜色尺度 $\mathbf{c}_k = [c_k^r, c_k^g, c_k^b]^T \in \mathbb{R}^{3 \times 1}$, 如果 $h(s, t)$ 估计准确, 那么应该有

$$I_R(s + i, t + j) = \mathbf{c}_k \cdot \mathbf{I}_k(\mathbf{P}_k(\mathbf{X}_R(s + i, t + j)))$$

为方便, 简写为

$$I_R(i, j) = \mathbf{c}_k \cdot \mathbf{I}_k(\mathbf{P}_k(\mathbf{X}_R(i, j))) = \mathbf{c}_k \cdot \mathbf{I}_k(i, j)$$

非线性优化的数学模型

$$E = \sum_{ijk} (I_R(i, j) - \mathbf{c}_k \cdot \mathbf{I}_k(i, j))^2$$

$i, j \in \left[-\frac{n-1}{2}, \frac{n-1}{2}\right]$ 是 *patch* 中的采样点, k 表示视角个数。

需要优化的变量: $h(s, t)$, h_s , h_t 以及 $\{\mathbf{c}_k\}$

颜色尺度的优化

求 E 关于 c_k 的导数

$$\frac{\partial E}{\partial c_k^r} = - \sum_{ij} \mathbf{I}_k^r(i, j) \mathbf{I}_R^r(i, j) + c_k^r \sum_{ij} (\mathbf{I}_k^r(i, j))^2$$

$$\frac{\partial E}{\partial c_k^g} = - \sum_{ij} \mathbf{I}_k^g(i, j) \mathbf{I}_R^g(i, j) + c_k^g \sum_{ij} (\mathbf{I}_k^g(i, j))^2$$

$$\frac{\partial E}{\partial c_k^b} = - \sum_{ij} \mathbf{I}_k^b(i, j) \mathbf{I}_R^b(i, j) + c_k^b \sum_{ij} (\mathbf{I}_k^b(i, j))^2$$

分别令上述三个偏导数为零，可以得到以下表达式

$$c_k^r = \frac{\sum_{ij} \mathbf{I}_k^r(i, j) \mathbf{I}_R^r(i, j)}{\sum_{ij} (\mathbf{I}_k^r(i, j))^2}$$

$$c_k^g = \frac{\sum_{ij} \mathbf{I}_k^g(i, j) \mathbf{I}_R^g(i, j)}{\sum_{ij} (\mathbf{I}_k^g(i, j))^2}$$

$$c_k^b = \frac{\sum_{ij} \mathbf{I}_k^b(i, j) \mathbf{I}_R^b(i, j)}{\sum_{ij} (\mathbf{I}_k^b(i, j))^2}$$

$h(s, t)$, h_s 和 h_t 的优化-梯度下降法

$$E = \sum_{ijk} \left(I_R(i, j) - \mathbf{c}_k \cdot \mathbf{I}_k(i, j) + \mathbf{c}_k \cdot \frac{\partial \mathbf{I}_k(i, j)}{\partial h(s, t)} (dh(s, t) + i * dh_s + j * dh_t) \right)^2$$

$$\text{令 } \mathbf{b}_{ijk} = I_R(i, j) - \mathbf{c}_k \cdot \mathbf{I}_k(i, j), \quad \nabla \mathbf{x} = [dh(s, t), dh_s, dh_t]^T, \quad \mathbf{A}_{ijk} = \left(\mathbf{c}_k \cdot \frac{\partial \mathbf{I}_k(i, j)}{\partial h(s, t)} \right) \begin{bmatrix} 1 \\ i \\ j \end{bmatrix}^T,$$

则上式可以表达为

$$E = \sum_{ijk} (\mathbf{A}_{ijk} \nabla \mathbf{x} + \mathbf{b}_{ijk})^2$$

$h(s, t)$, h_s 和 h_t 的优化

求 E 关于 x 的梯度, 可以得到

$$\frac{\partial E}{\partial \nabla x} = \sum_{ijk} A_{ijk}^T (A_{ijk} x + b_{ijk})$$

令 $\frac{\partial E}{\partial \nabla x} = 0$, 可以得到

$$\left(\sum_{ijk} A_{ijk}^T A_{ijk} \right) \nabla x + \sum_{ijk} A_{ijk}^T b_{ijk} = 0$$

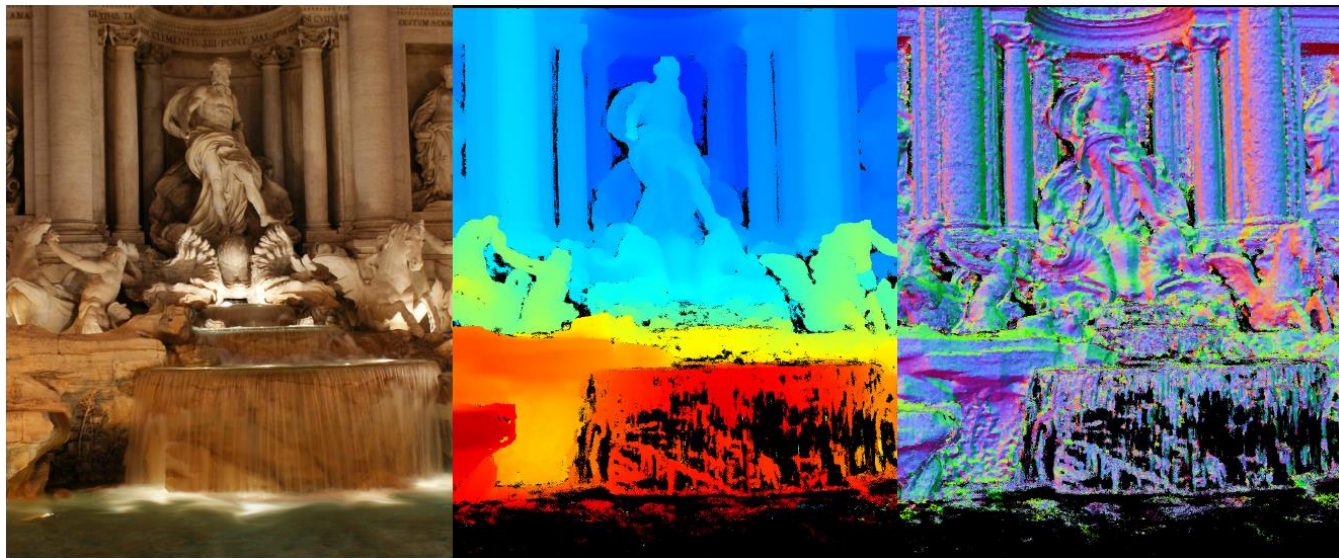
令 $A = \sum_{ijk} A_{ijk}^T A_{ijk}$, $b = \sum_{ijk} A_{ijk}^T b_{ijk}$, 则有

$$\nabla x = A^{-1} b$$

深度值非线性优化—整体框架

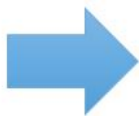
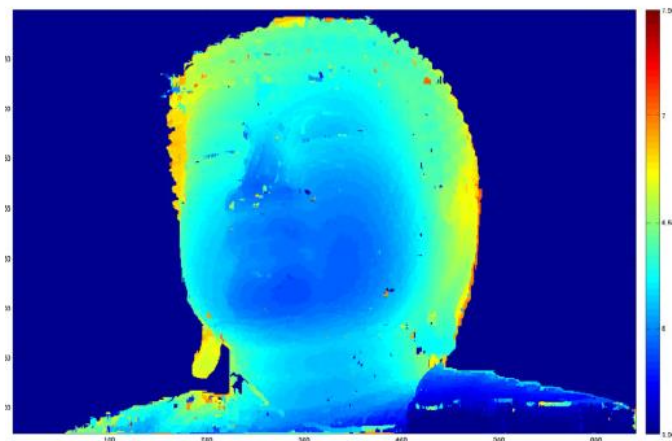
1. 只进行深度 $h(s, t)$ 的优化 (迭代4次)
2. While (迭代次数 < 20)
3. 每间隔5次迭代, 进行 $h(s, t)$, h_s, h_t 的优化, 否则仅优化深度
4. 优化完成后跟踪判断每个视角的置信度
5. 如果视角的置信度太小, 或者超过迭代14次尚未收敛, 则从局部视角中移除该视角, 并重新进行局部视角选择

深度估计结果



M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. Seitz. Multi-view stereo for community photo collections. In *IEEE International Conference on Computer Vision (ICCV)*, 2007.

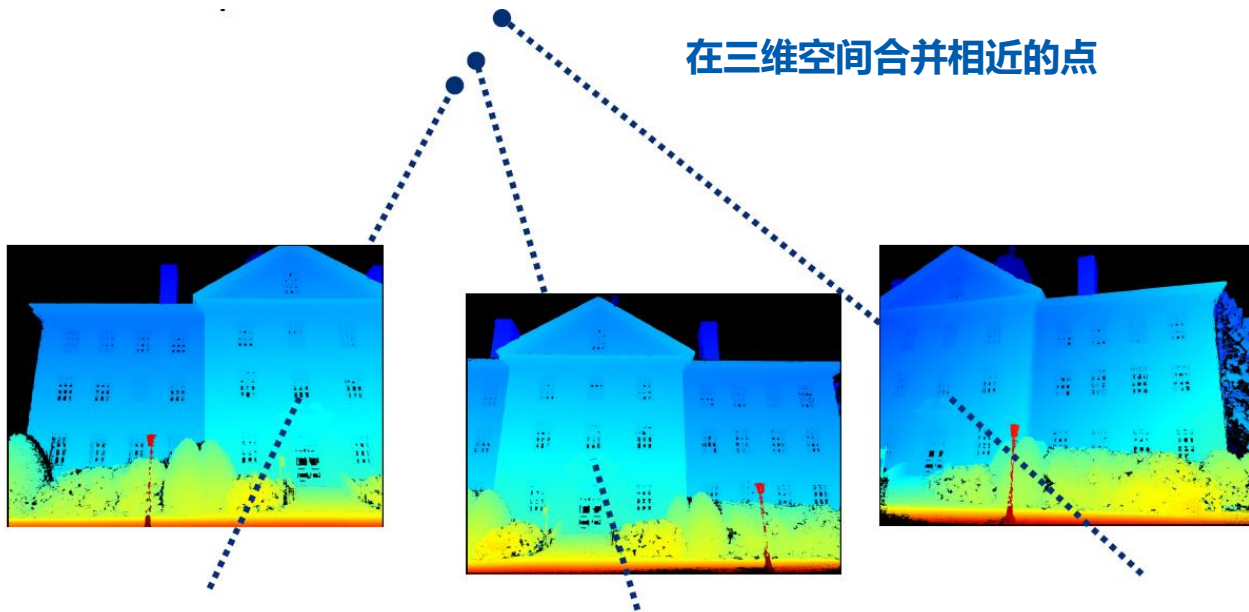
深度融合



直接聚合？ 一致性，可视性

M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. Seitz. Multi-view stereo for community photo collections. In *IEEE International Conference on Computer Vision (ICCV)*, 2007.

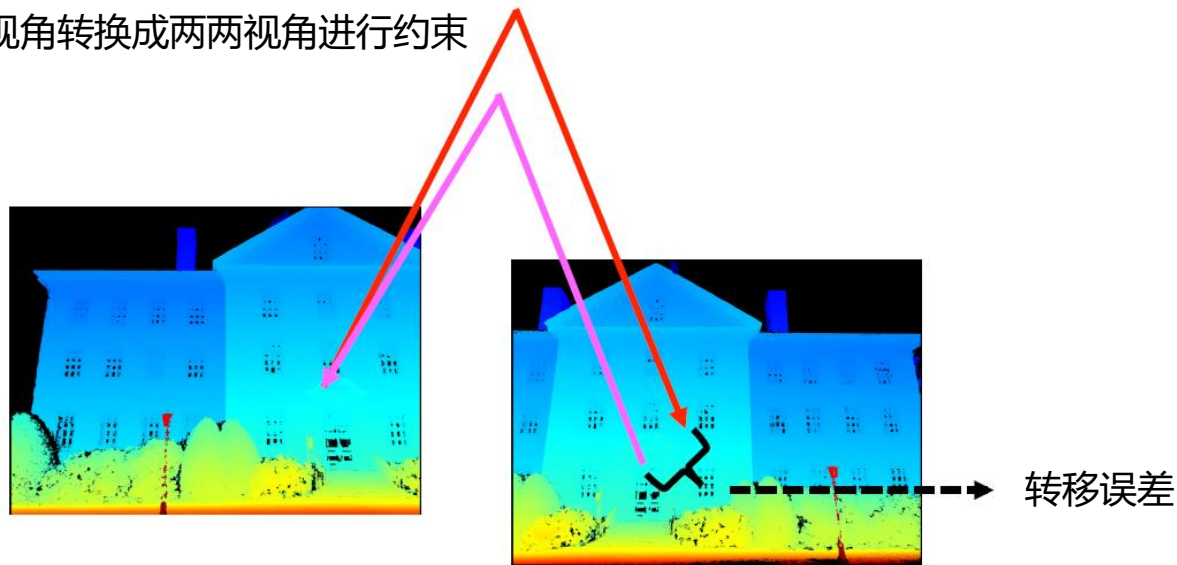
深度融合-一致性约束



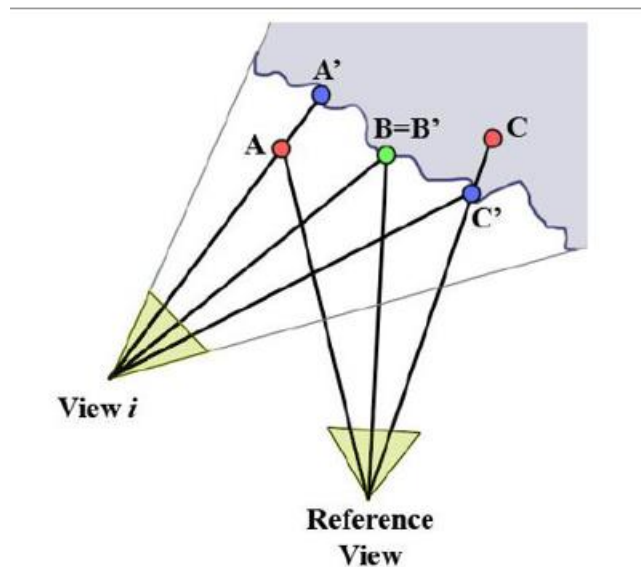
M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. Seitz. Multi-view stereo for community photo collections. In *IEEE International Conference on Computer Vision (ICCV)*, 2007.

深度融合-一致性约束

二维空间的邻域约束-将多视角转换成两两视角进行约束



深度融合-可视性约束



P. Merrell, A. Akbarzadeh, L. Wang, P. Mordohai, J.-M. Frahm, R. Yang, D. Nistér, and M. Pollefeys, Real-Time Visibility-Based Fusion of Depth Maps. In *IEEE International Conference on Computer Vision (ICCV)*, 2007.

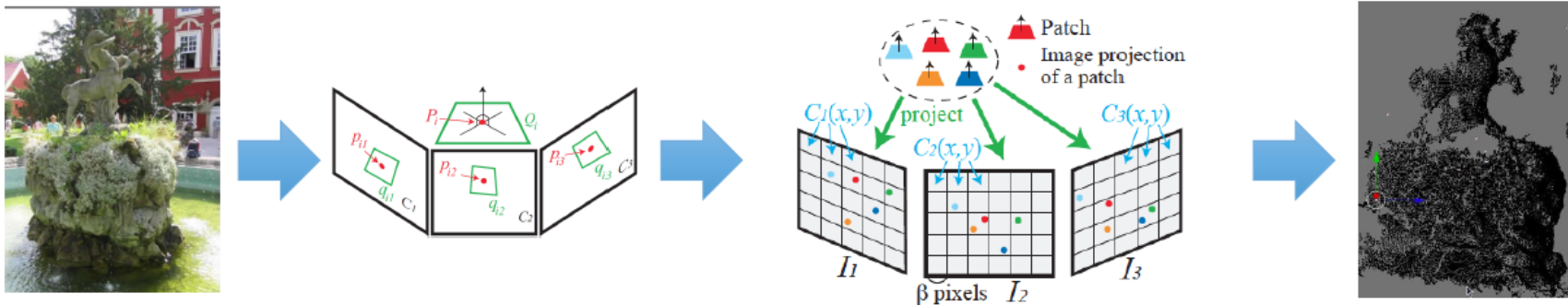
基于深度图融合的方法

- 邻域视角选择使得深度估计准确度提升,能够处理一些遮挡, 障碍物较多的场景
- 原理简单, 只用到光度一致性约束和可视性约束, 适用的场景广泛

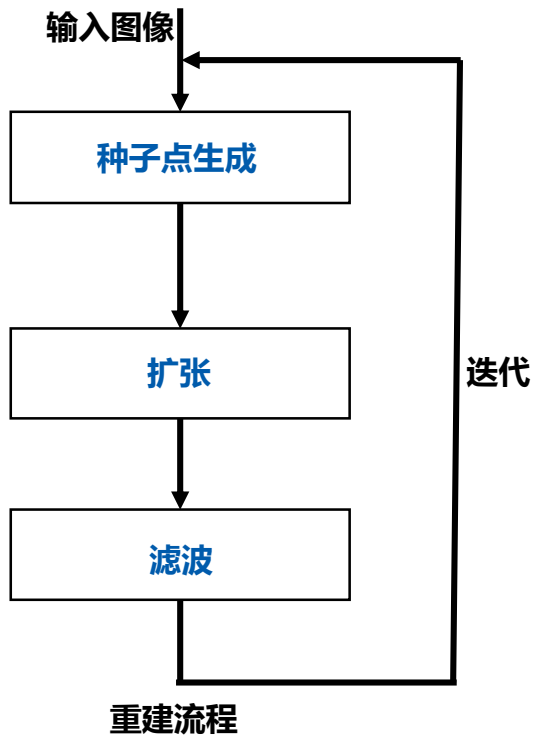


基于空间patch扩散的方法

- 假设空间中的3D 矩形patch
- 通过一定规则的扩张方法，使得patch覆盖物体表面



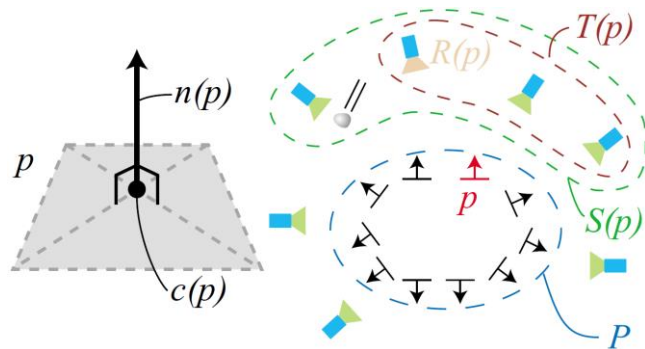
基于空间patch扩散的方法



- 初始种子点生成采用SIFT,HOG等特征点
- 扩张过程对已重建三维点的邻域进行匹配
- 滤波过程采用两种约束去除噪声点
 - ❑ 光度一致性约束
 - ❑ 可视性约束

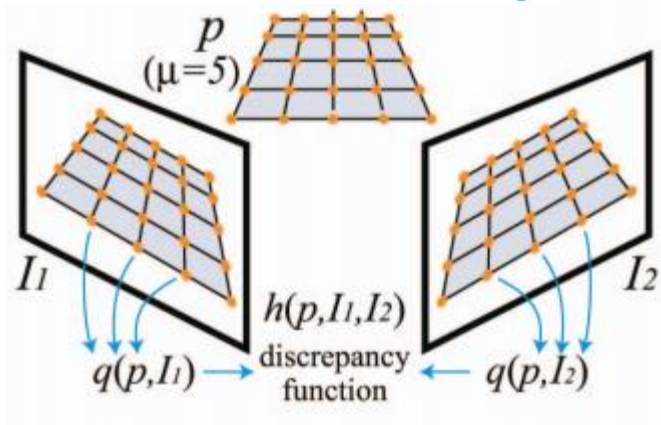
3D Patch定义

位置, 法向量
大小 5×5



光度一致性约束 $h(p, I_1, I_2)$

Patch投影到不同图像上, 计算NCC



PMVS <http://www.di.ens.fr/pmvs/>

参考博客: <https://blog.csdn.net/lhanchao/article/details/51885998>

初始3D patch 的生成

1. 在图像上均匀计算HOG/Harris特征;
2. 沿极线进行搜索找到匹配特征点;
3. 对匹配对, 通过三角化建立patch;

中心: 三角化确立

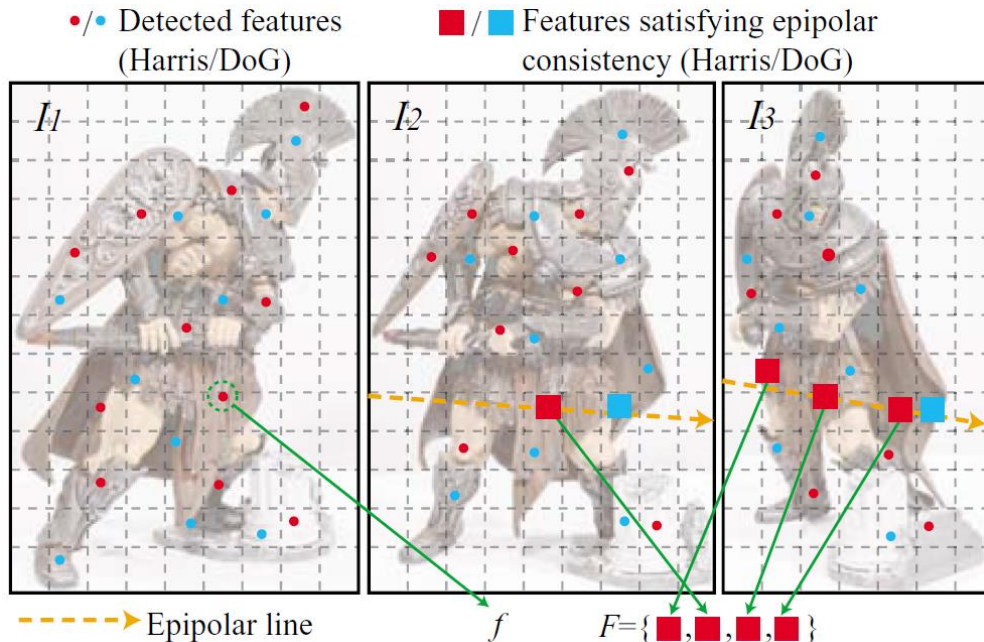
法向量: 指向参考图像

可视图像 $V(p)$: 法向量和视线夹角足够小,
通过光度一致性约束对可视图像进行筛选

$$V^*(p) = \{I | I \in V(p), h(p, I, R(p)) \leq \alpha\}$$

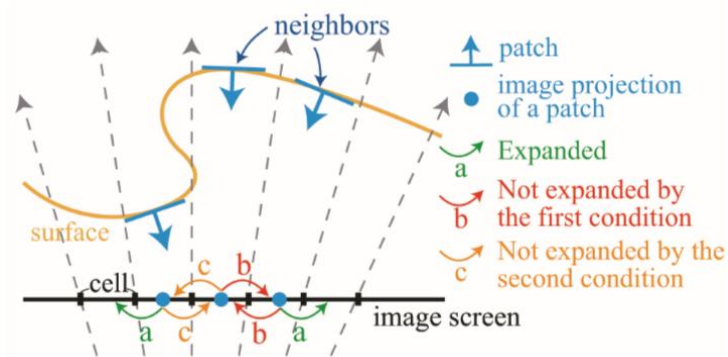
4. 对patch 位置和法向量进行优化。

$$\min_{c(p), n(p)} g^*(p) = \frac{1}{|V^*(p) \setminus R(p)|} \sum_{I \in V^*(p) \setminus R(p)} h(p, I, R(p)).$$

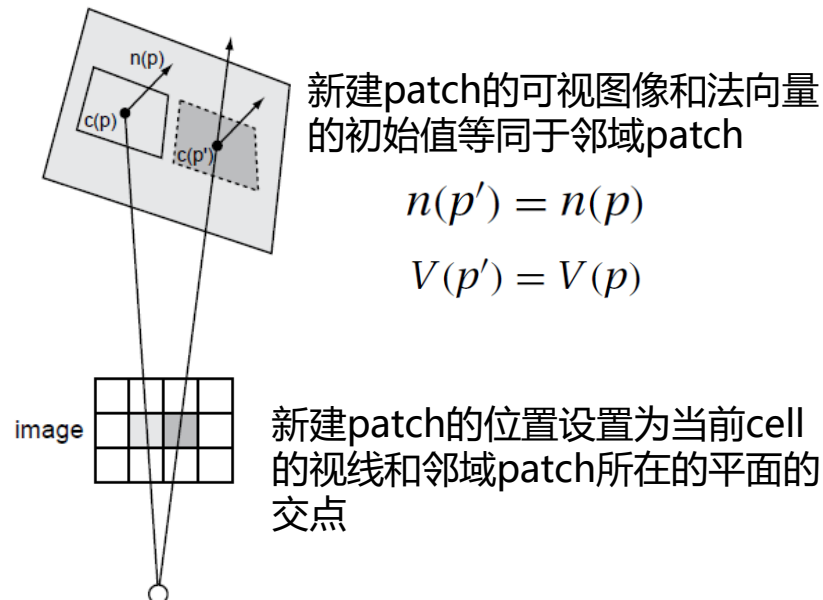


Patch扩张

1. 将三维patch投影到图像上;
2. 如果相邻cell没有patch且深度连续, 则建立初始patch;



3. 计算初始patch的可视图像, 并进行优化。



Patch滤波

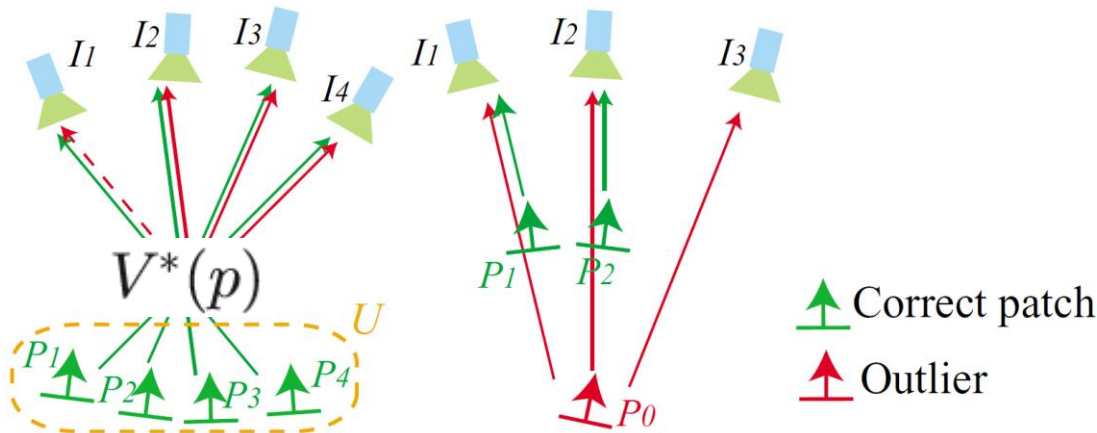
1. 可视性约束

$$|V^*(p)|(1 - g^*(p)) < \sum_{p_i \in U(p)} 1 - g^*(p_i)$$

2. 可视图像个数

$$|V^*(p)| < \gamma$$

3. 图像邻域中的cell同时也是空间邻域的比例小于0.25。



空间邻域: $|(\mathbf{c}(p) - \mathbf{c}(p')) \cdot \mathbf{n}(p)| + |(\mathbf{c}(p) - \mathbf{c}(p')) \cdot \mathbf{n}(p')| < 2\rho_1$

$U(p)$: 三维上不是邻域, 但投影到图像上同一个cell

优点

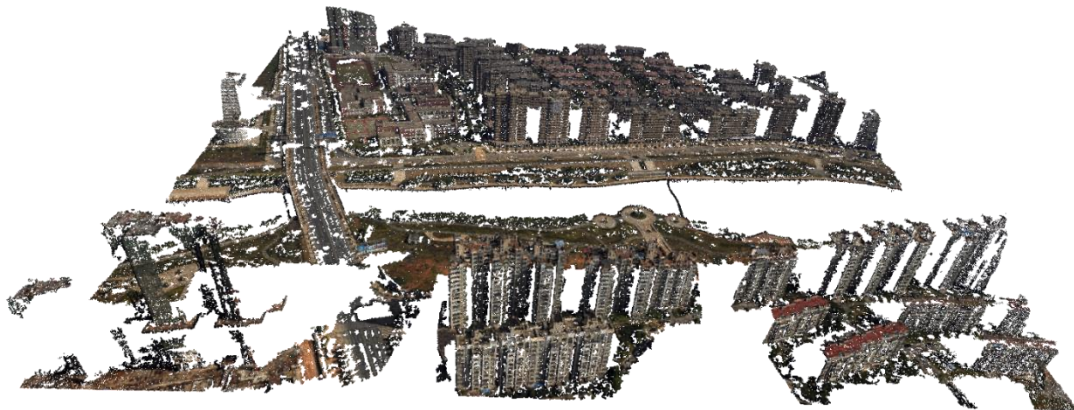
- 算法适用性强
- 使用于各种形状的物体



缺点



- 朗伯面假设
- 容易产生空洞



缺点



影像集	影像数目	影像尺寸	SFM (Hour)	稠密匹配 (Hour)
鲁甸	407	5616*3744	0.253	4.752
北川	273	5910*3854	0.351	3.545
连云港	782	5616*3744	0.643	7.414

- 计算量非常大

感谢各位聆听!
Thanks for Listening