目录 contents

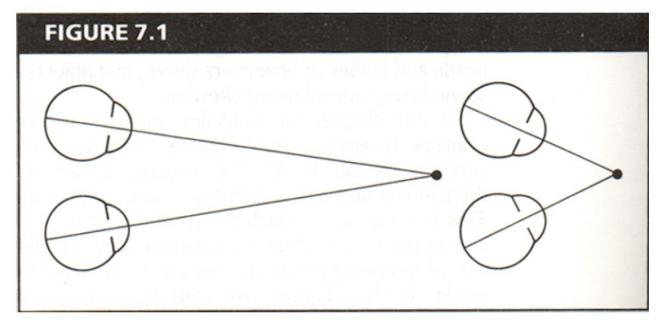
- 01 图像拼接与融合
- 02 运动与光流
- 03 立体视觉
- 04 运动恢复结构







模仿人眼的立体视觉过程,基于一帧或多帧具有共同拍摄场景的图像,由其中的 二维形状信息恢复原始场景中三维形状信息的视觉过程。

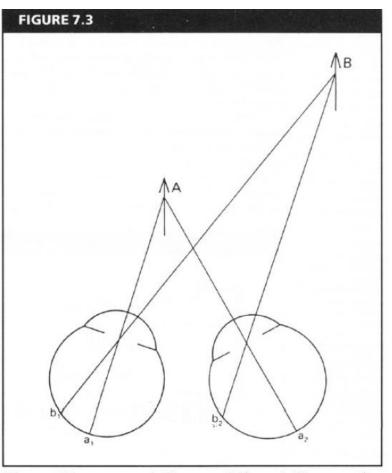


From Bruce and Green, Visual Perception, Physiology, Psychology and Ecology









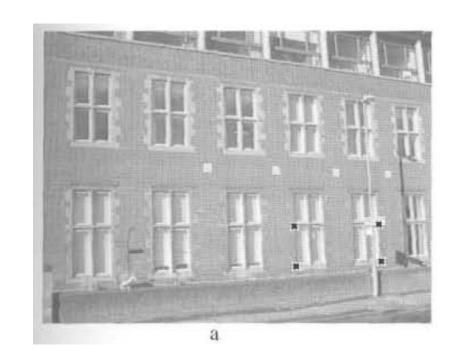
From Bruce and Green, Visual Perception, Physiology, Psychology and Ecology

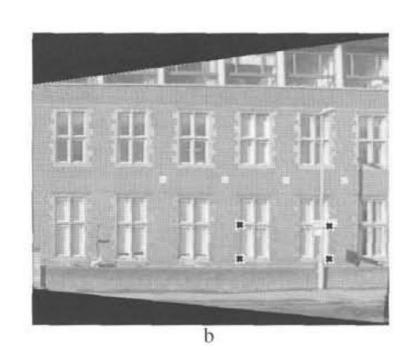
Disparity occurs when eyes fixate on one object; others appear at different visual angles





▶ 单视几何



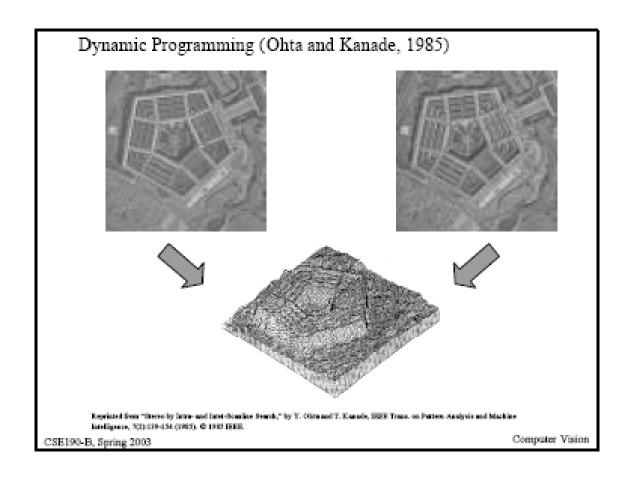


单视几何:针对平面物体的形状恢复





> 双视几何



双视几何: 基于标定摄像机的三维表面重建





> 多视几何



Figure 5. Images 1, 8, 14, and 22 of the castle sequence.



多视几何: 基于未标定摄像机的三维表面重建





- 现实存在的问题
 - · 一般的物体(Objects)都是三维的;
 - 图像(Images)却是有关灰度,颜色等信息的阵列;
 - 3D的深度(Depth)信息在一幅图像上不能明显的显示出来。
- 2D的分析需要3D的信息
 - 物体表面是连续,平滑(Smooth)的;
 - 物体都有特定的形状和边界。
- 3D的信息可以通过2D的图像计算出来
 - 视差(Disparity), 深度(Depth)信息等等。



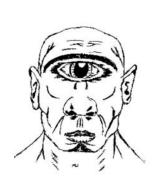




1. 概述

- > 为什么需要两只眼睛
 - 物体的深度信息不能通过单眼所获得。
 - 物体的深度信息可以通过双眼的观察得到。

$$w \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} m_0 & m_1 & m_2 & m_3 \\ m_4 & m_5 & m_6 & m_7 \\ m_8 & m_9 & m_{10} & m_{11} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$











多目







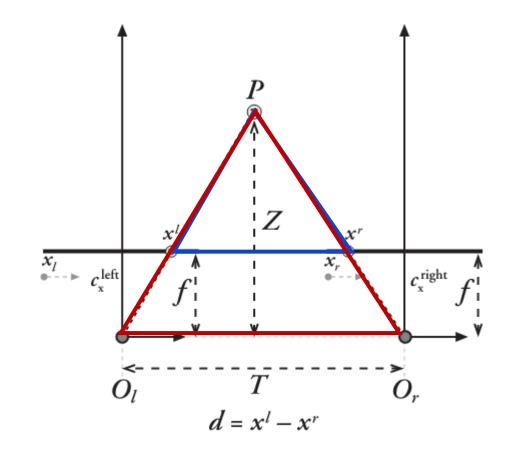


> 视差测量原理

- 双目立体视觉三维测量 是基于视差原理。
- 计算公式:

$$\frac{T - (x^{l} - x^{r})}{Z - f} = \frac{T}{Z} \implies Z = \frac{fT}{x^{l} - x^{r}}$$

T是基线距离



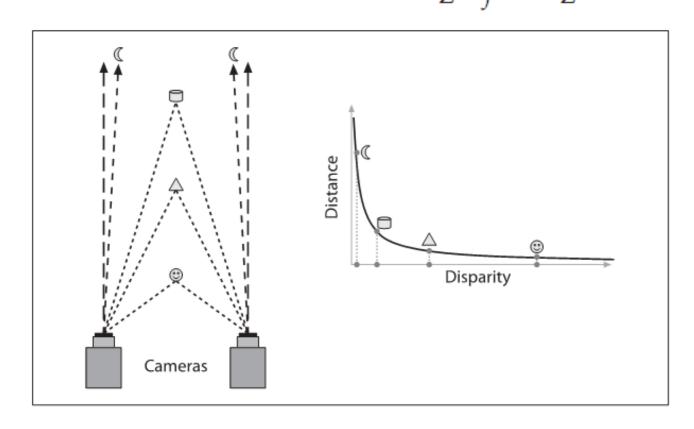






> 视差测量原理

• 视差和深度成反比关系: $\frac{T-(x^l-x^r)}{Z-f} = \frac{T}{Z} \Rightarrow Z = \frac{fT}{x^l-x^r}$



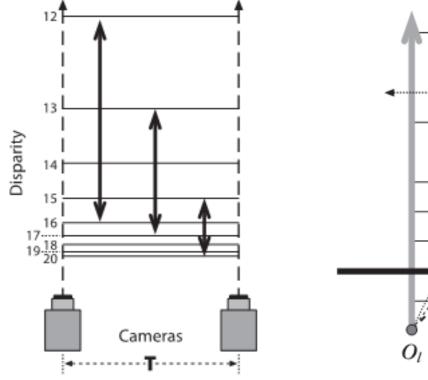


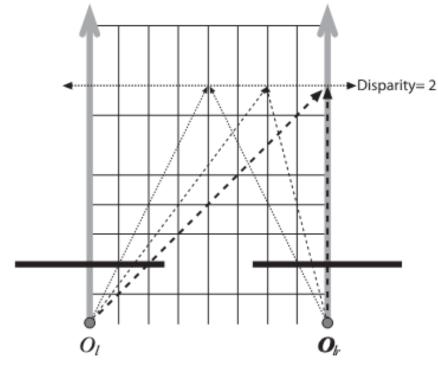




> 视差测量原理

• 同一深度下的视差一样







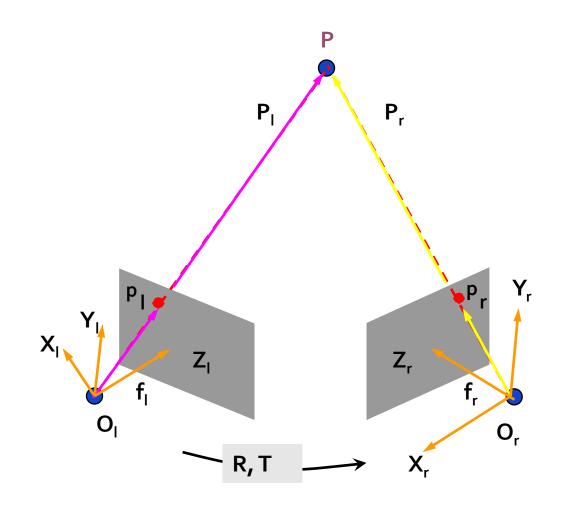




> 立体视觉的处理过程

入口数据: 单帧或多帧图像

为了最终恢复三维信息,需要基 于入口数据进一步获取哪些数据?







注 2. 基本原理

- > 立体视觉的处理过程
 - 如何知道不同图像中的匹配信息?
 - 如何知道不同拍摄方位的相对放置(外部参数)?
 - 如何知道摄像机的内部参数?

步骤:

- 图像配准
- (2) 摄像机标定 (确定内部参数)
- (3) 确定摄像机相对放置(确定外部参数)
- (4) 三维表面重建





2. 基本原理

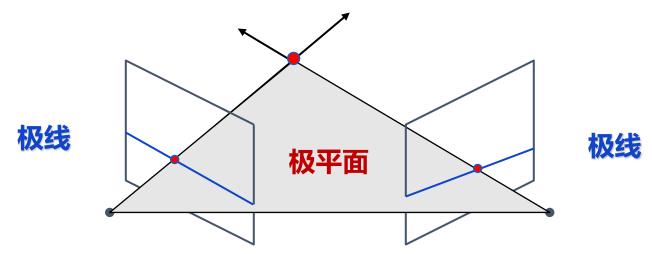
- > 图像配准
 - 1) 基于像素的图像配准方法 两帧图像中所有具有同一原像的像素对都应建立匹配关系。
 - 2) 基于特征的图像配准方法 仅针对两帧图像中的具有同一原像的点、线、区域特征对建立匹配关系。
 - 两类方法具有一定的联系,并且,基于特征的图像配准效率更高,在其基础上可简化像素级配准。
 - 由于各类特征均可以转换为点特征,因此,基于特征点的配准方法成为研究的重点。
 - 例如, 直线段特征可转换为直线段的两两交点, 区域特征可转换为区域的重心。





2. 基本原理

- > 极线几何
 - •Geometry of two views allows us to constrain where the corresponding pixel for some image point in the first view must occur in the second view.



为什么重要?

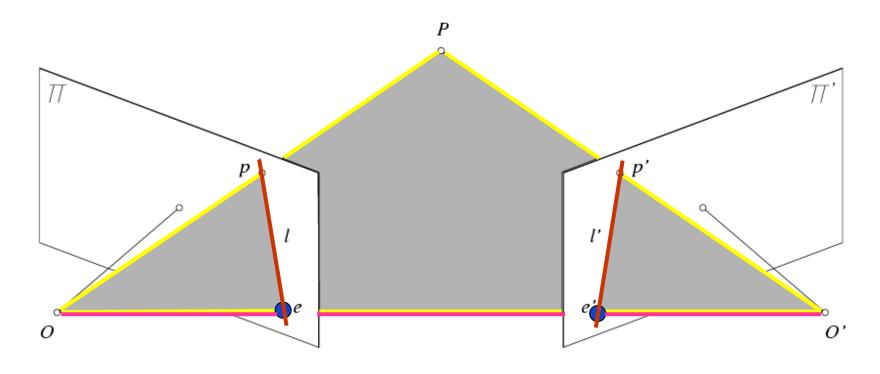
将二维的匹配搜索限制在一维的搜索。







▶ 极线几何



• 极平面

• 基线

•极点

• 极线





注 2. 基本原理

> 极线几何

· 基线: 两个光心的连线

• 极点: 基线和成像平面交点

• 极平面: 基线和物体点形成的平面

• 极线: 极平面与成像平面的交线

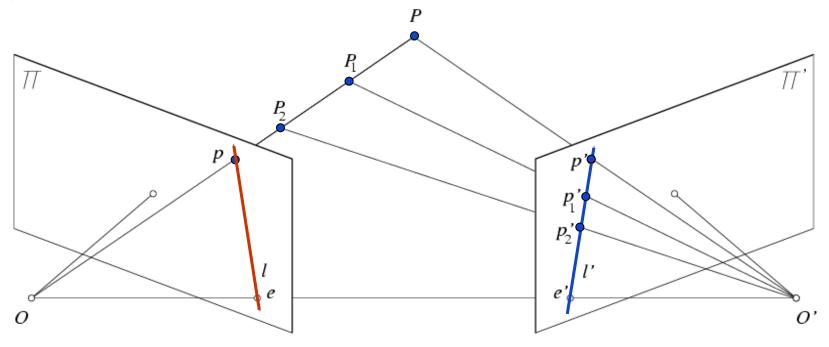
- 所有极线都过极点。
- 物体改变, 极平面会变, 极线会变, 但基线和极点不变。







> 极线约束



- 左相机中点p 在右相机的匹配点必然在极线 I'上。
- 右相机中点p'在左相机的匹配点必然在极线 / 上。





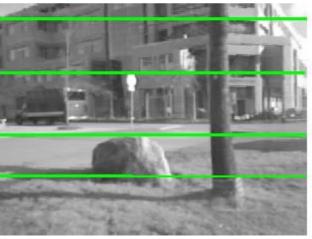


> 极线约束







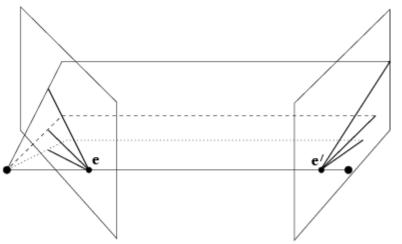






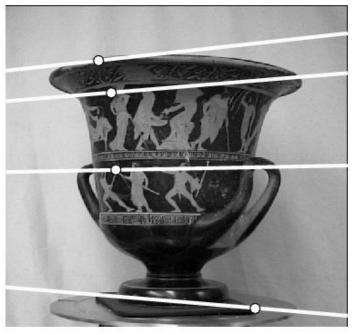


> 极线约束



随着物体变化, 极线绕 着基线旋转





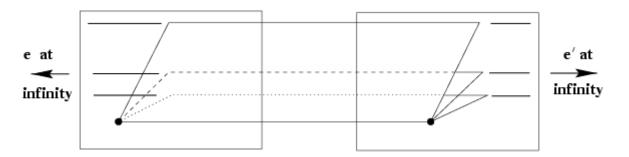


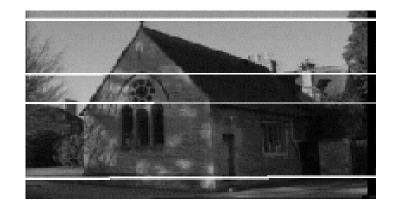


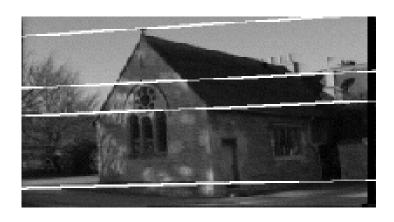


> 极线约束

物体平行X轴移动









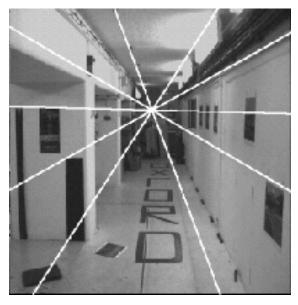


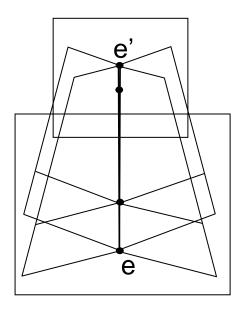


> 极线约束



相机前向运动





- 两相机极点在同一位置
- 所有像素点会以关注点为圆心发散

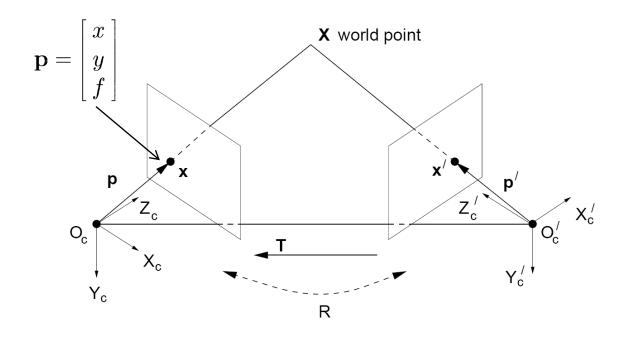






➤ Essential 矩阵

给定一个立体视觉装置,如何用代数的方法表达极线约束 如果两个相机是经过标定的:即知道两个相机之间的旋转和平移矩阵



两相机坐标 (光心坐标) 可由旋转矩阵R 和平移矩阵T 转化:

$$X' = RX + T$$







> Essential 矩阵

向量点乘 (标量)

$$\mathbf{A} \cdot \mathbf{B} = \sum a_i b_i$$

几何意义:投影

$$\mathbf{A} \cdot \mathbf{B} = |\mathbf{A}| |\mathbf{B}| \cos \theta$$

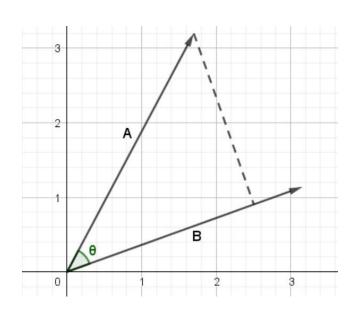
向量叉乘 (矢量)

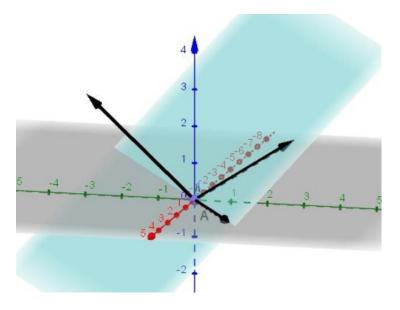
方向+模长:

$$a \times b = ((y1*z2 - y2*z1), -(x1*z2 - x2*z1), (x1*y2 - x2*y1))$$

几何意义:垂直平面

$$\|\mathbf{a} \times \mathbf{b}\| = \|\mathbf{a}\| \|\mathbf{b}\| \sin \theta.$$











注 2. 基本原理

➤ Essential 矩阵

向量叉乘 (矢量)

$$a \times b = ((y1*z2 - y2*z1), -(x1*z2 - x2*z1), (x1*y2 - x2*y1))$$

$$a'$$
 $b = ((y1*z2 - y2*z1), -(x1*z2 - x2*z1), (x1*y2 - x2*y1))$

其中 a,b 为1*3向量 a'为3*3矩阵

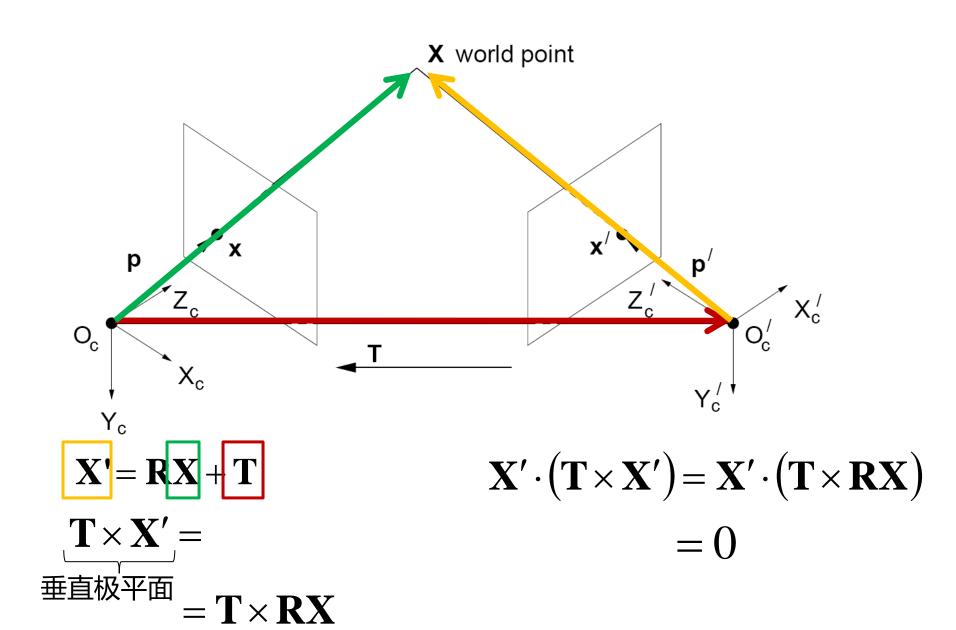
即叉乘可以转为普通矩阵的乘法







➤ Essential 矩阵







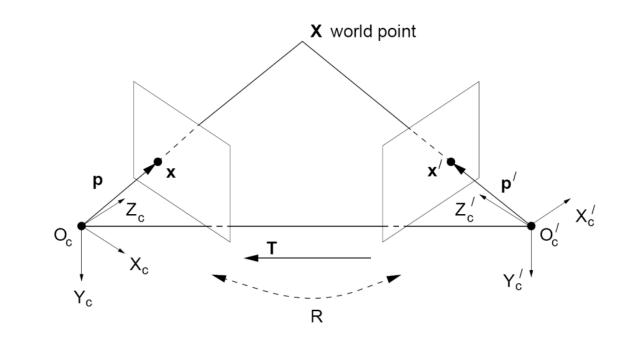
2. 基本原理

➤ Essential 矩阵

$$\mathbf{X'} \cdot (\mathbf{T} \times \mathbf{RX}) = 0$$
$$\mathbf{X'} \cdot (\mathbf{T}_x \ \mathbf{RX}) = 0$$

Let
$$\mathbf{E} = \mathbf{T}_x \mathbf{R}$$

 $\mathbf{X'}^T \mathbf{E} \mathbf{X} = \mathbf{0}$



点 p 和 p'过光心的射线在X'和X射线上,因此有:

$$\mathbf{p'}^T \mathbf{E} \mathbf{p} = 0$$

E 被称为本征矩阵(essential matrix),它建立起两个像点之间的关系[Longuet-Higgins 1981]





🔆 2. 基本原理

- ➤ Essential 矩阵
 - 左右两幅图像相对应的点之间的关系可以通过Essential 矩阵或是 Fundamental 矩阵来表明。
 - Essential 矩阵是摄像机标定情况下用的。
 - 公式: $(p_{r})^{T} E p_{t} = 0$
 - p_r和p_i分别是齐次摄像机坐标向量。
 - · 公式描述了点p,位于与向量Ep,相关的外极线上。
 - Essential 矩阵是奇异矩阵,并有两个相等的非零奇异值,秩为2。







注 2. 基本原理

➤ Fundamental 矩阵

$$(p_r)^T E p_l = 0$$

本征矩阵反应的是两幅图像在相机物理坐标上的极线约束,而往往我们需 要的是在像素坐标上讲行搜索。

$$q_r = M_r p_r \qquad q_l = M_l p_l$$

 $M_r M_l$ 是相机内参矩阵

$$(M_r^{-1}q_r)^T E M_l^{-1} q_l = 0 \quad \Rightarrow \quad q_r^T M_r^{-T} E M_l^{-1} q_l = 0$$

$$F = M_r^{-T} E M_l^{-1}$$
 则: $q_r^T F q_l = 0$ 基本矩阵







注 2. 基本原理

> 弱标定

- 定义:对于内部参数未知的摄像机,通过两幅图 中的冗余点集合来估计外极几何。
- 方法:
 - 8点算法 (Longuet-Higgins 1981)
 - 最小二乘法
 - 最小二乘法的8点算法
 - 规范化线性8点算法 (Hartley 1995)

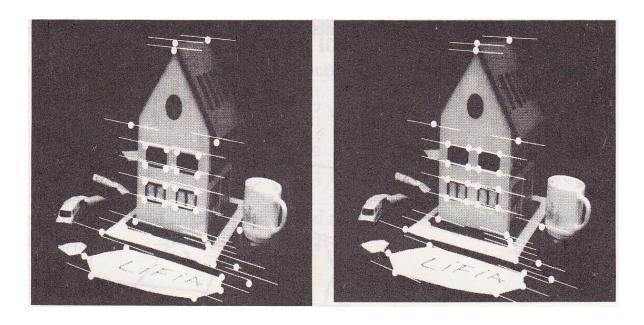
Reference: 《计算机视觉——一种现代方法》第10章







> 弱标定



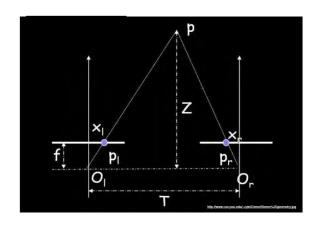
- 使用一个玩具房子的两幅图像上的37个点作为输入的弱标定实验。
- 数据点在图中用圆点表示, 所经过的外极线用短的直线段表示。
- 左图显示使用最小二乘法的普通8点算法得到的输出结果;右图为使用 Hartley变换后的该方法的输出结果。







> 举例



$$\mathbf{R} = \mathbf{T} = \mathbf{T}$$

 $\mathbf{E} = [\mathbf{T}_{\mathbf{x}}]\mathbf{R} =$

$$\mathbf{p'}^{\mathrm{T}}\mathbf{E}\mathbf{p} = 0$$

For the parallel cameras, image of any point must lie on same horizontal line in each image plane.





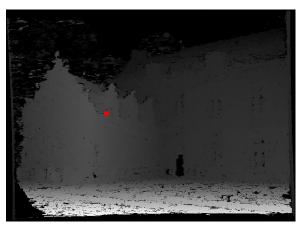


➤ 举例

左相机 I(x,y)



视差图 D(x,y)



(x',y')=(x+D(x,y),y)

右相机 I′(x′,y′)



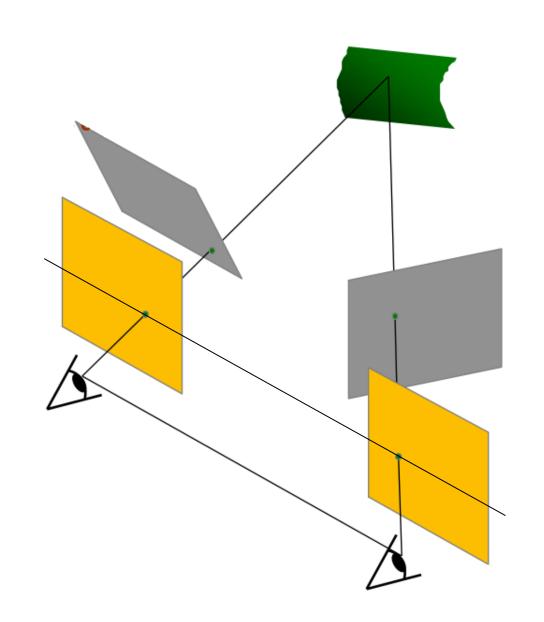
可是当两个相机光心不平行时如何处理?







- > 双目图像矫正
 - •目的:规范化极线约束中 的极线分布,使得匹配效 率得到进一步的提高。
 - 校正后的图像不需要求极 线方程,因为相对应的匹 配点在图像相对应的扫描 线(Scan-line)上。

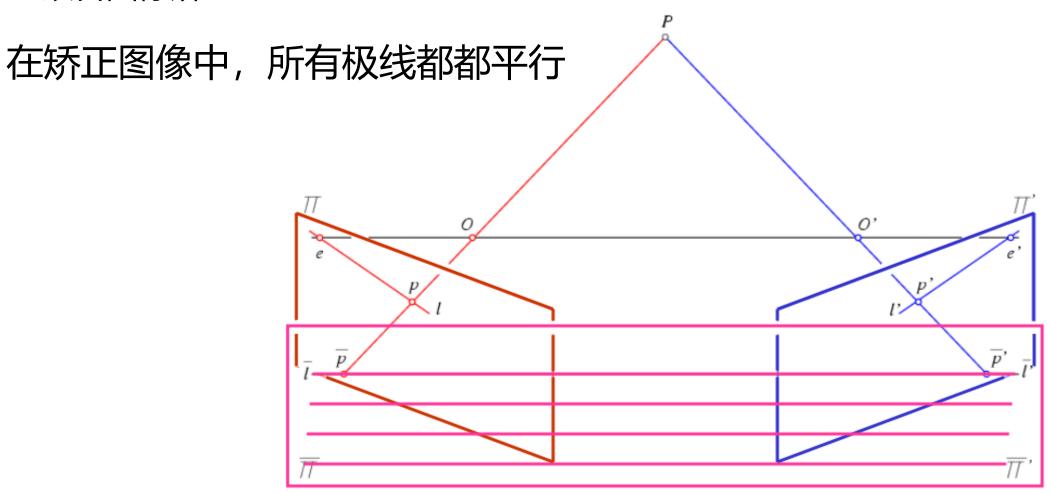








- **2.** 基本原理
- > 双目图像矫正



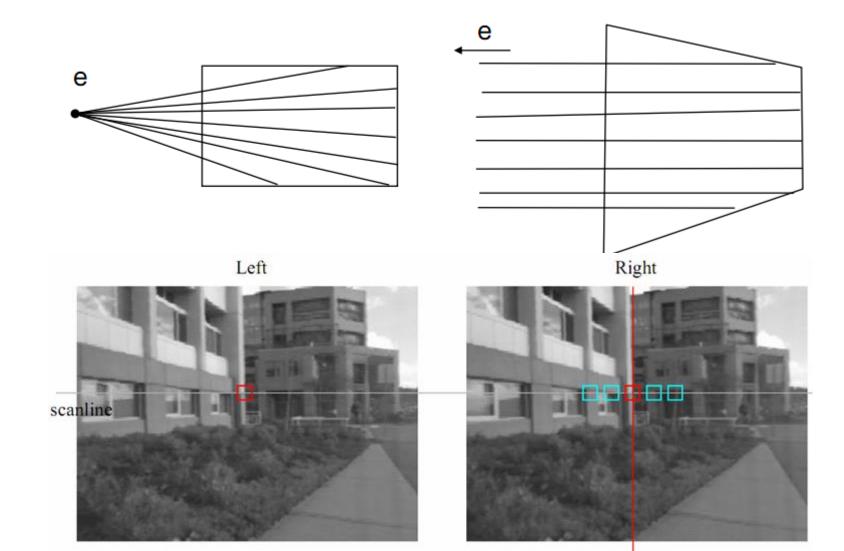






> 双目图像矫正

把极点拉向无穷远处。



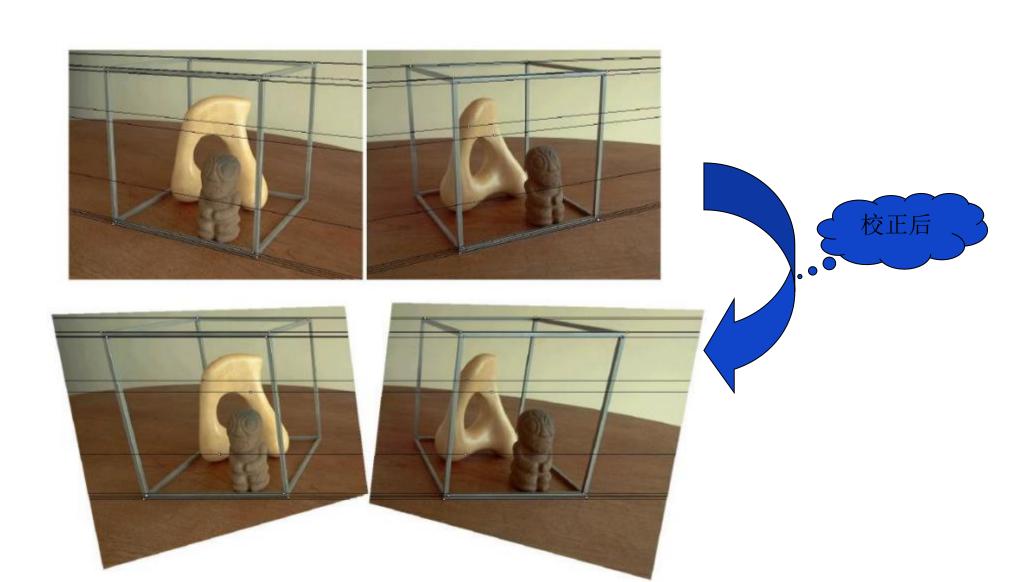






2. 基本原理

> 双目图像矫正









🔆 2. 基本原理

> 其他匹配约束

一致性约束

・对图像进行规范化处理(Normalization) 设参考摄象机和其它摄象机的图像函数分别为

 $f_0(i,j)$ 和 $f_k(i,j)$,则图像窗内规范化图像函数为:

$$\bar{f}_0(i,j) = (f_0(i,j) - \mu_0)/\sigma_0$$

$$\bar{f}_k(i,j) = (f_k(i,j) - \mu_k) / \sigma_k$$

 μ 是图像窗内光强的平均值, σ 是光强分布参数:

$$\sigma^{2} = \frac{1}{mn} \sum_{j=1}^{n} \sum_{i=1}^{m} (f(i, j) - \mu)^{2}$$

相似估价函数为差值绝对值之和 $\varepsilon_k = \sum_{i=1}^{n} \sum_{j=1}^{m} \left| \bar{f}_0(i,j) - \bar{f}_k(i,j) \right|$







- ▶ 2. 基本原理
- > 其他匹配约束

顺序约束

- ·如果在参考图像中点A在点B的 左边=> 在目标图像中点A的匹 配点也在点B的匹配点的左边
- 对细小物体不成立

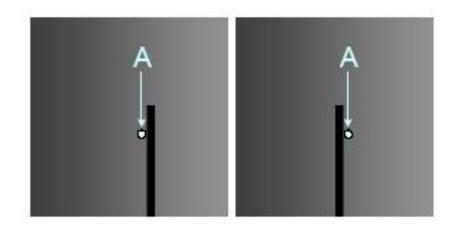


Image from Sun et al. CVPR05



- 🔆 2. 基本原理
 - > 其他匹配约束
 - ・唯一性约束
 - 一幅图像(左或右)上的每一个特征点只能与另一 幅图像上的唯一一个特征对应.
 - ・连续性约束

物体表面一般都是光滑的,因此物体表面上各点 在图像上的投影也是连续的,它们的视差也是连 续的.

在物体边界处,连续性约束不能成立.



- ☀3. 立体匹配
 - > 立体匹配难点
 - ·求解对应问题是立体成象系统的核心内容
 - · 求解对应问题极富有挑战性,可以说是立体视觉 最困难的一步.
 - ・方法
 - ・基于特征(点、线)的匹配(稀疏匹配)
 - ·基于区域的匹配(稠密匹配)

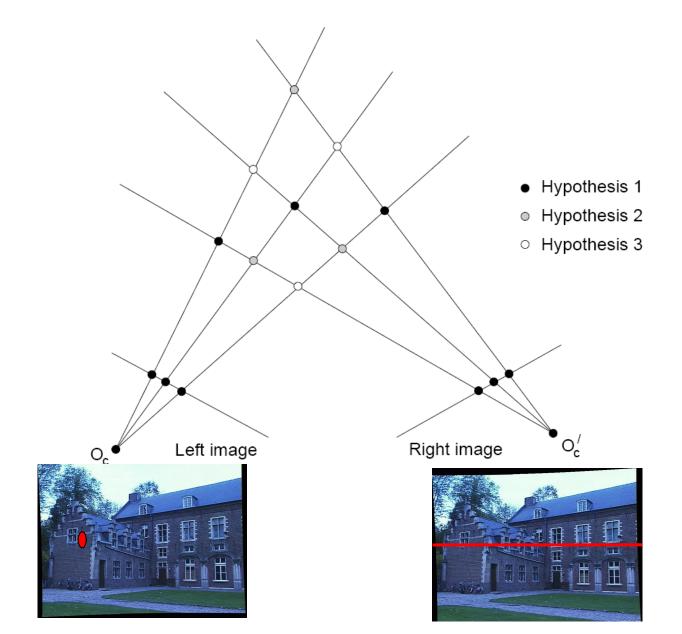






☀3. 立体匹配

> 立体匹配难点



有多个匹配假设满 足极线约束, 但哪 个是正确的?

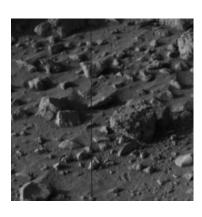






☀3. 立体匹配

- > 立体匹配难点
 - ・场景投影到两幅图像中并不总是一致的
 - ・摄像机相关
 - ・ 图像噪声、不同增益、不同对比度等等...
 - ・视点相关
 - ・透视畸变
 - ・遮挡
 - ・镜面反射
 - ・复杂场景因素
 - ・重复场景
 - ・无纹理区域





引入约束,减少搜索范围





🌺 3. 立体匹配

> 立体匹配分类

在立体匹配中,匹配问题可以看成是寻找两组数据相关程度的过程。立体匹配算法 由多种分类。

- ①根据算法运行时约束的作用范围:分为局部(local)匹配算法和全局(Global)匹配 算法。
 - ②基于生成的视差图:可分为稠密(Dense)匹配和稀疏(Sparse)匹配。

稠密匹配: 是基于生成的视差图,对于所有像素都能生成确定视差值,称为稠密匹配。

稀疏匹配:只选择关键像素点通常为角点或者边缘点1计算视差值的方法称为稀疏匹配, 该算法计算速度较快,但后续还需要通过插值算法计算缺失像素点的视差值,因此应用 场景上有很大限制。





3. 立体匹配

- ▶ 特征匹配——稀疏
 - ·在立体图像对中识别兴趣点(interesting point),而后在两幅图像中匹配相对应的点。
 - ・识别兴趣点(interesting point)

在图像中具有很大变化的区域内寻找兴趣点

在以某一点为中心的窗函数中,计算其在不同方向上的变化量

为避免将多个相邻点选为同一个特征对应的兴趣点,将特征点选在兴趣测度函数具有局部最大值的地方

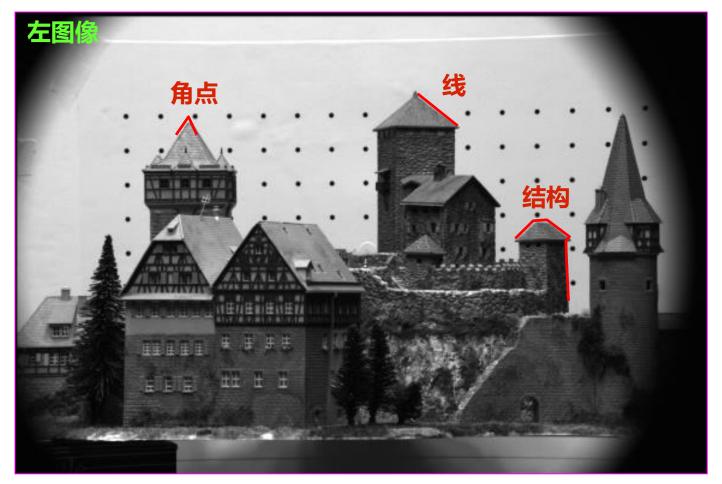






☀3. 立体匹配

▶ 特征匹配——稀疏



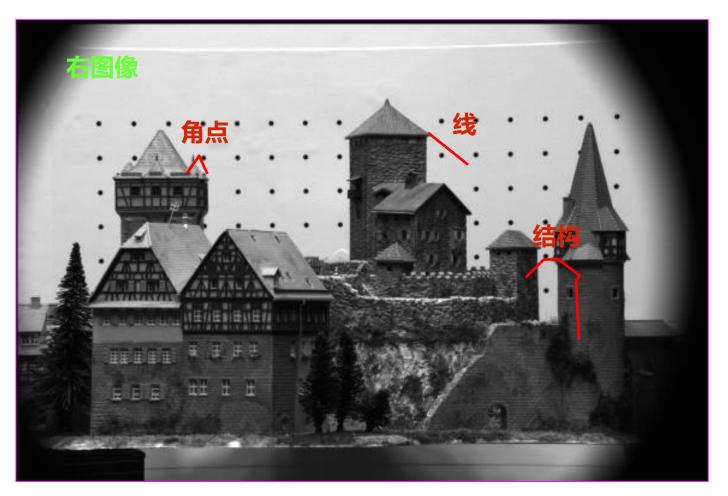
・对于左图像中的每一个特征...





☀3. 立体匹配

▶ 特征匹配——稀疏



· 在右图像中寻找... 当相似度达到最大时的偏移量就是视差





- 3. 立体匹配
- ▶ 特征匹配——稀疏
 - · 基于特征的立体匹配算法产生对应于图像特征点的场景稀疏深度图.
 - · 仅仅能恢复用于求解共轭对的像素子集对应的特征点深度.
 - ·要想得到其它点的深度值,必须通过使用有关计算方法来估算,如内插值技术.





3. 立体匹配

- ▶ 基于区域相关性的立体匹配 (稠密)
 - ・计算一幅图像的一个小窗函数内的像素与另一幅图像中具有同样的潜在对应特征的 小窗函数的像素之间的相关值.具有最大相关值的小窗区域就是对应区域.
 - · 只有满足外极线约束的区域才能是匹配区域. 考虑到垂直视差的存在, 应将外极线 邻近的像素点也包括在潜在的匹配特征集中.







· 对参考图像中每一点找到对应于场景中同一点的匹配像素, 得到稠密的视差图





3. 立体匹配

- > 立体匹配步骤
 - ・匹配代价计算

计算左图一个像素和右图一个像素之间的代价

・代价聚合

通常全局算法不需要代价聚合,而局部算法需要通过求和、求均值或其他方法对一个支持窗口内的匹配代价进行聚合而得到参考图像上一点p在视差d处的累积代价CA(p,d),这一过程称为代价聚合。通过匹配代价聚合,可以降低异常点的影响,提高信噪比(SNR,Signal Noise Ratio)进而提高匹配精度。代价聚合策略通常是局部匹配算法的核心,策略的好坏直接关系到最终视差图(Disparity maps)的质量。

- ・视差计算
- ・视差优化/后处理

常用的方法有插值(Interpolation)、亚像素增强(Subpixel Enhancement)、精细化(Refinement)、图像滤波(Image Filtering)等操





- 3. 立体匹配
- > 全局匹配与局部匹配
 - · 局部算法 (Local/window-based algorithms):
 - ・在匹配点的一个特定窗口中计算相似度。
 - SSD, SAD, MSE, MAD, etc.
 - ·全局算法 (Global algorithms):
 - ・能量方程:
 - 模拟退火(Simulated annealing), 动态规划 (Dynamic Programming), 最大流(Max-flow), 图像分割(graph-cut), etc.





3. 立体匹配

▶ 全局匹配

全局(半全局)匹配算法主要是采用了全局的优化理论方法估计视差,建立一个全局能量函数,其包含一个数据项和平滑项,通过最小化全局能量函数得到最优的视差值。其中,图割(Graph cuts, GC)、置信传播(Belief Propagation, BP)、动态规划(Dynamic Programming,DP),粒子群算法(Particle Swarm Optimization,PSO)、遗传算法(Genetic Algorithm, GA)等优化算法都是常用的求解能量最小化的方法。

$$E(d) = E_{data}(d) + E_{smooth}(d) = \sum_{\mathbf{p} \in R} C(\mathbf{p}, d) + \sum_{\mathbf{q}, \mathbf{p} \in R} P(d_{\mathbf{q}} - d_{\mathbf{p}})$$

 匹配程度 定义的约束 匹配代价 \mathbf{p}, \mathbf{q} 连续两点的视差,惩罚项

由于全局匹配算法在数学上是一个能量函数的优化问题,因此可以找到最优解。 这个问题被证明在二维空间是NP困难的。因此,即使全局算法具有准确性较高的优点, 其计算速度确非常慢,在实时性要求高的场合不适合使用全局立体匹配算法。





🔆 3. 立体匹配

▶ 全局匹配

考虑到能量优化问题在一维空间的复杂度是多项式级的,因此一些研究试图做一 些近似来降低算法的复杂度。例如,半全局算法(SGM)就利用了这一特性将二维问 题简化为8到16个一维问题,以实现一种较好的近似。其在各个方向上计算累积代价后, 将各方向代价相加得到总代价,这样就模拟了二维的优化问题。SGM是立体匹配逐渐 取代激光雷达生成视差图的技术关键,同时也是商业软件中应用最多的立体匹配算法

$$L_{\mathbf{r}}(\mathbf{p},d) = C(\mathbf{p},d) + \min(L_{\mathbf{r}}(\mathbf{p} - \mathbf{r},d),$$

$$L_{\mathbf{r}}(\mathbf{p} - \mathbf{r},d-1) + P_{1}, L_{\mathbf{r}}(\mathbf{p} - \mathbf{r},d+1) + P_{1},$$

$$\min_{i} L_{\mathbf{r}}(\mathbf{p} - \mathbf{r},i) + P_{2}) - \min_{k} L_{\mathbf{r}}(\mathbf{p} - \mathbf{r},k)$$





3. 立体匹配

> 局部匹配

- 局部立体匹配算法又称基于窗口的方法或基于支持区域的方法。
- 算法对参考图像中的每个像素计算一个合适大小、形状和权重的窗口,然后对这个 窗口内的视差值进行加权平均。
- 理想的支持窗口应该完全覆盖弱纹理区域,并在窗口内深度连续。
- 与全局立体匹配算法相似,通过优化一个代价函数的方法计算最佳视差。但是,在 局部立体匹配算法的能量函数中,只有基于局部区域的约束数据项,没有平滑项。
- 局部匹配算法仅利用某一点邻域的灰度、颜色、梯度等信息进行计算匹配代价,计算复杂度较低,大多实时的立体匹配算法都属于局部立体匹配的范畴,但局部立体匹配算法对低纹理区域、重复纹理区域、视差不连续和遮挡区域匹配效果不理想。
- 基于区域相关性的匹配就属于最传统的局部匹配







3. 立体匹配

- > 局部匹配
 - ◆ **固定窗口代价聚合**使用固定大小和形状的窗口作为代价聚合的基元,通常是一个矩形,并假设支持窗口内的其它像素点与待匹配点具有相同的视差。固定窗口法精度不高,但易实现、耗时短,在一些对实时性要求极高的场合得到了应用。
 - ◆基于双边滤波的代价聚合算法仍然使用固定大小和形状的窗口,但窗口内的元素权重不同,权重由目标图像在该窗口内像素与窗口中心的灰度差和距离计算。基于双边滤波的代价聚合算法精度高,但计算复杂,实时性差,算法性能随窗口尺寸指数

增加。

自适应权重:颜色+空间距离。影响广泛,Google引用1200+次。

空间距离项 颜色距离项
$$C_d^A(p) = \frac{\sum_q exp(-\frac{|p-q|}{\sigma_S})exp(-\frac{|I(p)-I(q)|}{\sigma_R})C_d(q)}{\sum_q exp(-\frac{|p-q|}{\sigma_S})exp(-\frac{|I(p)-I(q)|}{\sigma_R})}$$
 [归一化项可以省略掉]
$$C_d^A(p) = \sum_q exp(-\frac{|p-q|}{\sigma_S})exp(-\frac{|I(p)-I(q)|}{\sigma_R})C_d(q)$$







3. 立体匹配

- > 局部匹配
 - ◆ 基于分割的代价聚合算法的主要思想是: 预先将作为参考图像的左图进行分割, 对于支持窗口内与窗口中心处于同一分割的像素, 对应的权值取1, 否则为一个远小于1的正数。但是图像分割是一个非常耗时的操作, 同样无法在实时性要求较高的场合使用。
 - ◆ 基于十字的代价聚合算法(Cross-based Cost Aggregation,CBCA)的支持窗口形状并不确定,会根据匹配点邻域的灰度值而改变,具体实现后面的更新中会介绍。该方法可以使用GPU并行计算,具有较好的实时性,现广泛应用于各种算法的代价聚合步骤。





4. 多基线立体视觉

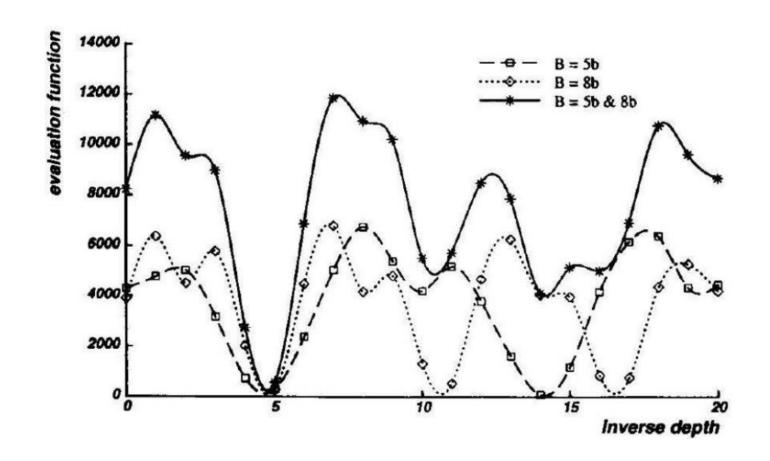
- 增加第三个摄像机可以消除(大部分)由双目图像点造成的不确定性。
- 本质上,第三幅图像可以用来检查前两幅图像中假定的匹配:和前两幅图像中 匹配点对应的三维空间点首先被重建,然后再投影到第三幅图。如果在第三幅 图像的再投影点周围没有相容的点,那么这个匹配一定是错误的匹配。
- Okutami 和 Kanade (1993) 提出一个多摄像机的算法,其中同时利用所有图像 来搜索匹配。
- 基本想法:假设所有图像都是被校正过的,讲搜索正确的视差的操作转换为搜 索正确的深度或者深度的倒数。
- 选择第一幅图像作为参考,将与所有其他摄像机相关的平方差加到一个全局评 价函数E中。
- 评价函数E是深度倒数的函数





4. 多基线立体视觉

• 下图是不同数量的摄像机评价函数E的函数值:





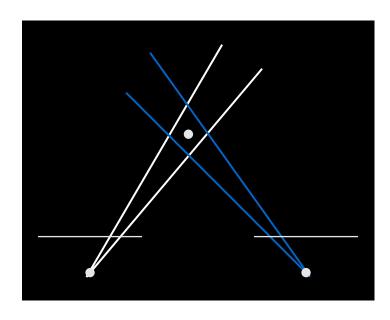




4. 多基线立体视觉

> 基线选择

选择最优基线



Large Baseline

Small Baseline

- •What's the optimal baseline?
 - Too small: large depth error
 - Too large: difficult search problem



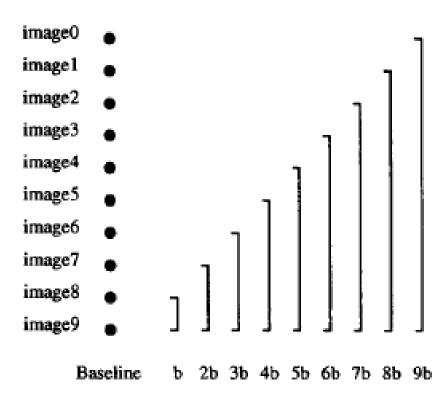




★4. 多基线立体视觉



Figure 2: An example scene. The grid pattern in the background has ambiguity of matching.









4. 多基线立体视觉

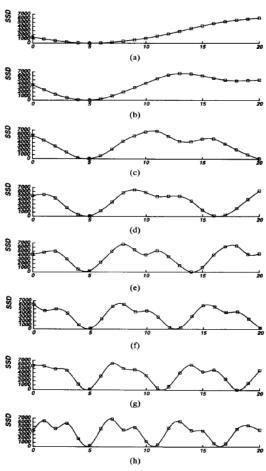


Fig. 5. SSD values versus inverse distance: (a) B=b; (b) B=2b; (c) B=3b; (d) B=4b; (e) B=5b; (f) B=6b; (g) B=7b; (h) B=8b. The horizontal axis is normalized such that 8bF=1.

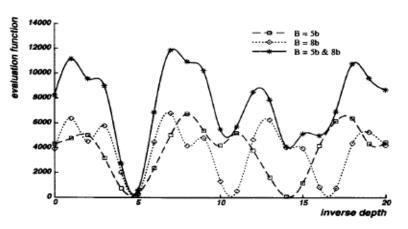


Fig. 6. Combining two stereo pairs with different baselines.

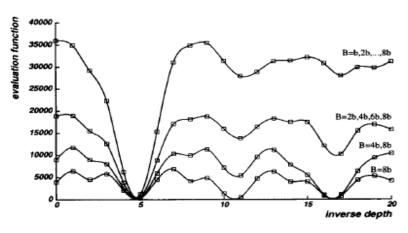


Fig. 7. Combining multiple baseline stereo pairs.

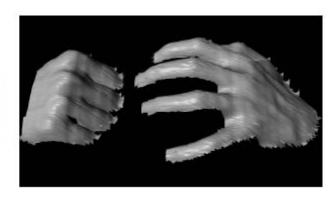




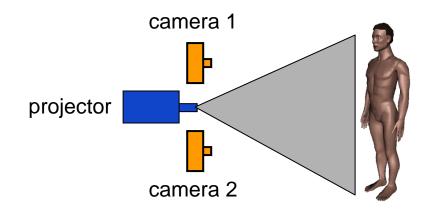


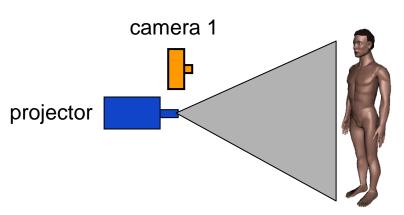






Li Zhang's one-shot stereo





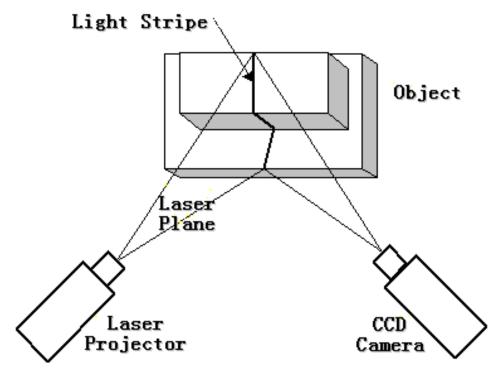
• 投射结构光到物体表面, 简化相关匹配过程





5. 基于结构光的立体视觉

- 光学投影器将一定模式的 结构光投射于物体表面, 在表面上形成由被测物体 表面形状所调制的光条三 维图。
- 结构光的目的就是增加三 维物体的纹理信息。



两个整数之间的汉明距离则是,求两个整数对应的二进制 相应位置的不同字符的个数

结构光要求:编码要唯一;汉明距离为1;满足抽样定理。





5. 基于结构光的立体视觉

- 二进制编码 (binary code)
- 格雷码 (Gray code)
- 改进格雷码
- 灰度编码:锯齿状条纹 (Saw-tooth pattern)、正弦状条纹、etc.
- 金字塔子光栅条纹投射法 (Pyramid sub-grating projecting method)
- 伪随机编码阵列 (Pseudo-random encoded array)

Ref:

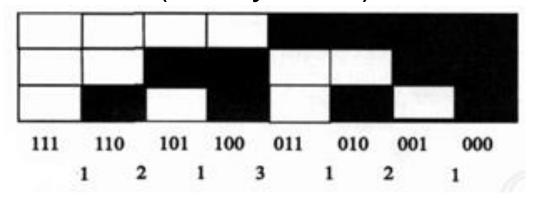
Salvi et al., Pattern codification strategies in structured light system, Pattern Recognition, 37, 827-849, 2004.



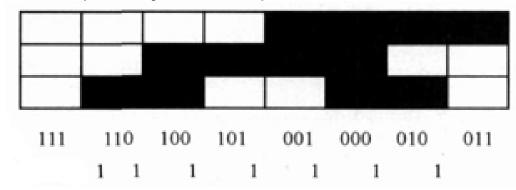




• 二进制编码 (binary code)



• 格雷码 (Gray code)



任意两个相邻的代码只有一位二进 制数不同;格雷编码不仅能降低解 码的出错率还能降低投射图案的制 造误差

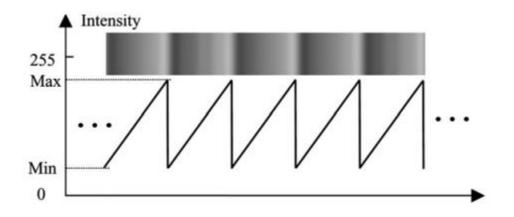






• 灰度编码 (n-array codes)



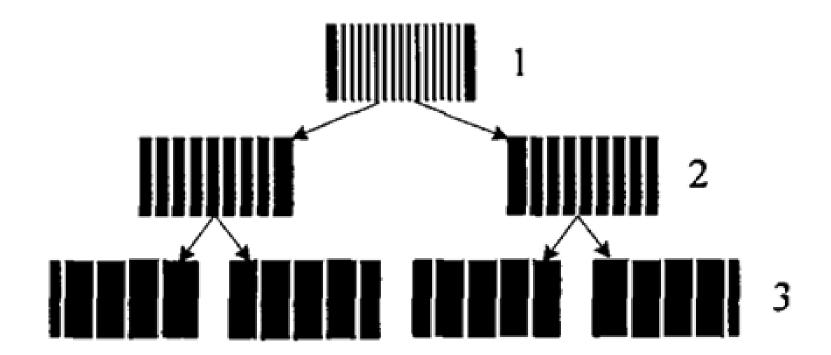






5. 基于结构光的立体视觉

• 金字塔子光栅条纹投射法 (Pyramid sub-grating projecting method)









- 伪随机编码阵列 (Pseudo-random encoded array)
- 63*65 的伪随机阵列:







