

# 计算机视觉—图像配准

申抒含

中国科学院自动化研究所  
模式识别国家重点实验室

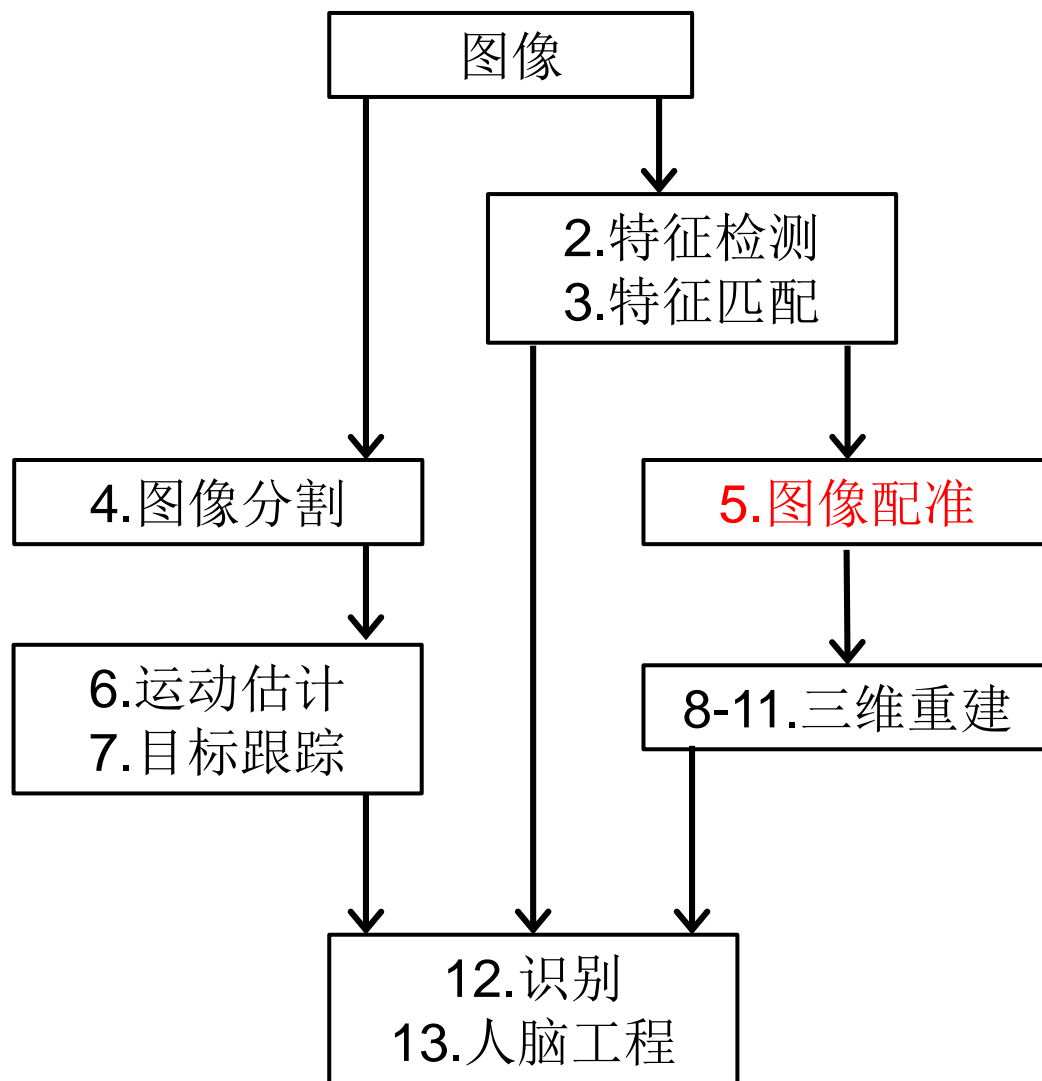


**Robot Vision Group**

National Laboratory of Pattern Recognition

Institute of Automation, Chinese Academy of Sciences

# 计算机视觉课程结构图

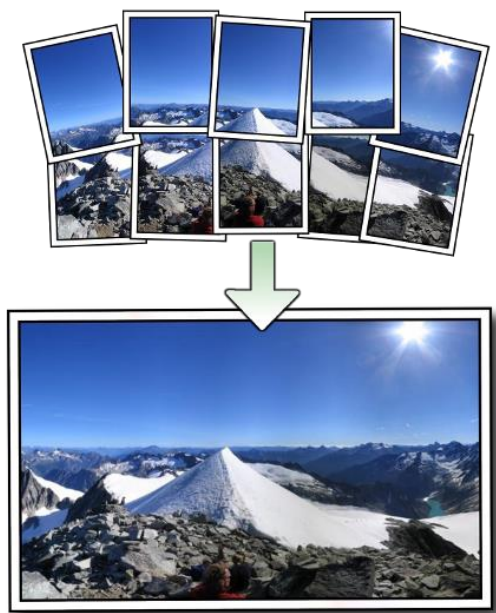


# 什么是图像配准

图像配准(Image registration)就是将不同时间、不同传感器或不同条件下获取的两幅或多幅图像进行匹配、叠加的过程。



# 图像配准的种类



刚体配准



非刚体配准



多模态配准



# 图像配准的应用



Panorama

Brown and Lowe, 2007

# 图像配准的应用



Photomontage

Agarwala, Dontcheva, Agrawala et al. 2004



# 图像配准的应用

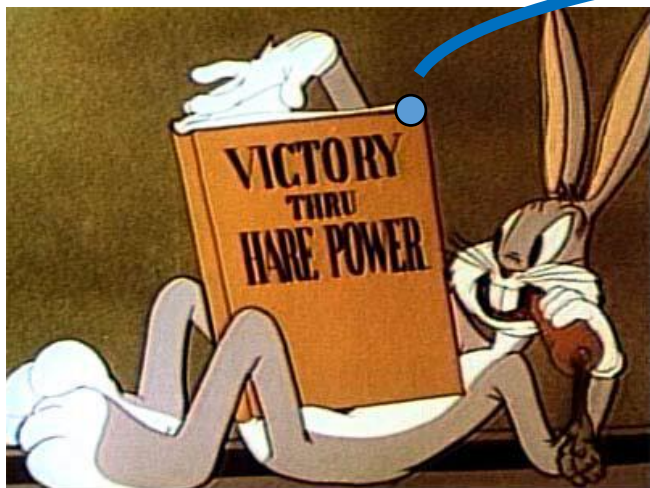


Poisson image editing

Perez, Gangnet, and Blake, 2003

# 图像配准的基本思想

图像变换  $W: \mathbb{R}^2 \mapsto \mathbb{R}^2$



源图像S



目标图像T

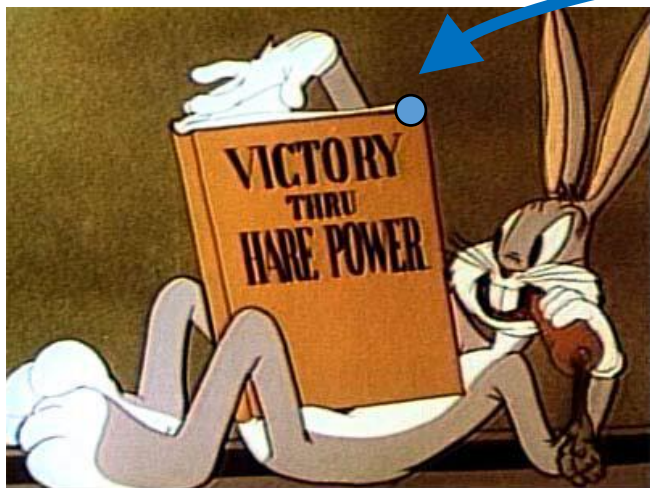
寻找变换  $W(u,v)$ ，将源图像S映射一个变换图像，使其与目标图像T一致。





# 图像配准的基本思想

图像逆变换  $W^{-1}: \mathbb{R}^2 \mapsto \mathbb{R}^2$



源图像S



目标图像T

为使变换图像完整，可以通过逆变换  $W^{-1}$  将每一个像素映射回源图像S来构造变换图像。

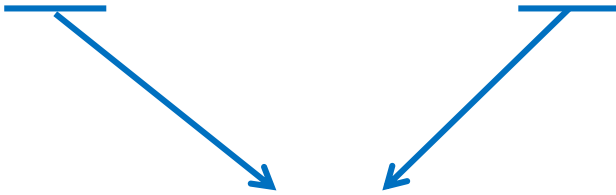
注意：并非所变换  $W$  都可逆



# 图像配准中的变换模型

## 平移 (2D Translation)



$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$


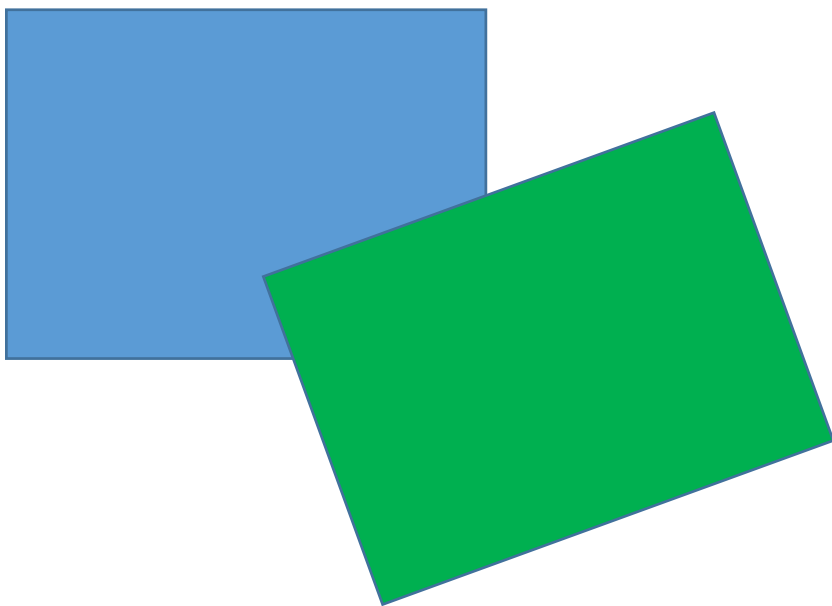
图像点的齐次坐标

自由度: 2

结构保持性: 姿态

# 图像配准中的变换模型

欧式 (2D Euclidean)



$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & t_x \\ \sin \theta & \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

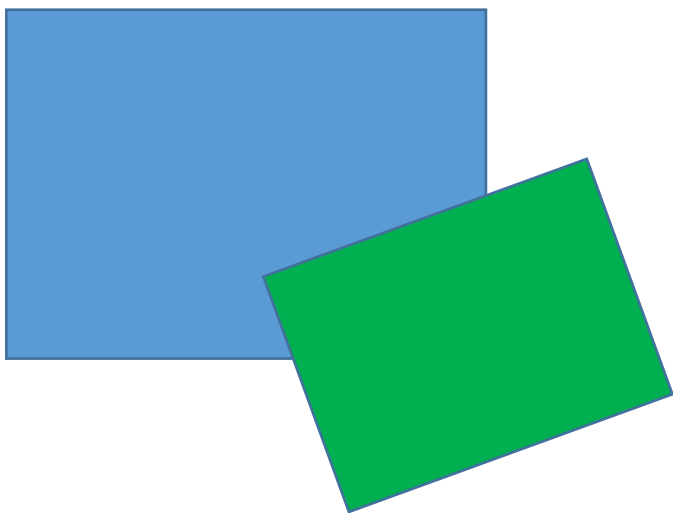
自由度: 3

结构保持性: 长度



# 图像配准中的变换模型

相似（2D Similarity）



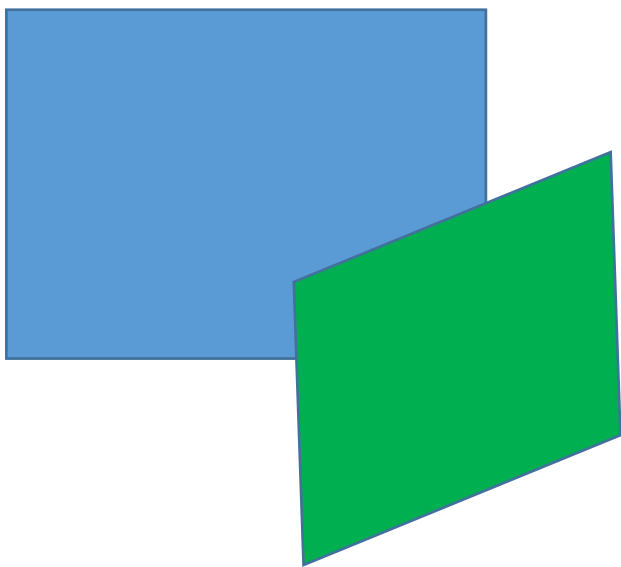
$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} s \cdot \cos \theta & -s \cdot \sin \theta & t_x \\ s \cdot \sin \theta & s \cdot \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

自由度：4

结构保持性：夹角

# 图像配准中的变换模型

仿射（2D Affine）



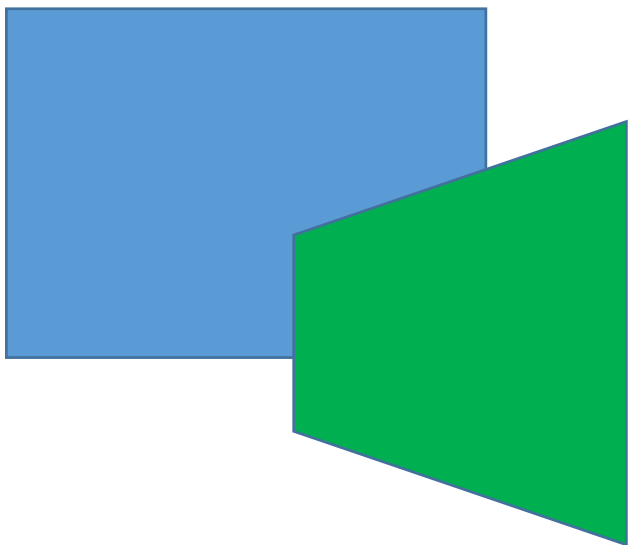
$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

自由度：6

结构保持性：平行性

# 图像配准中的变换模型

摄影（2D Projective），也称单应（Homography）



$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} p_{00} & p_{01} & p_{02} \\ p_{10} & p_{11} & p_{12} \\ p_{20} & p_{21} & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

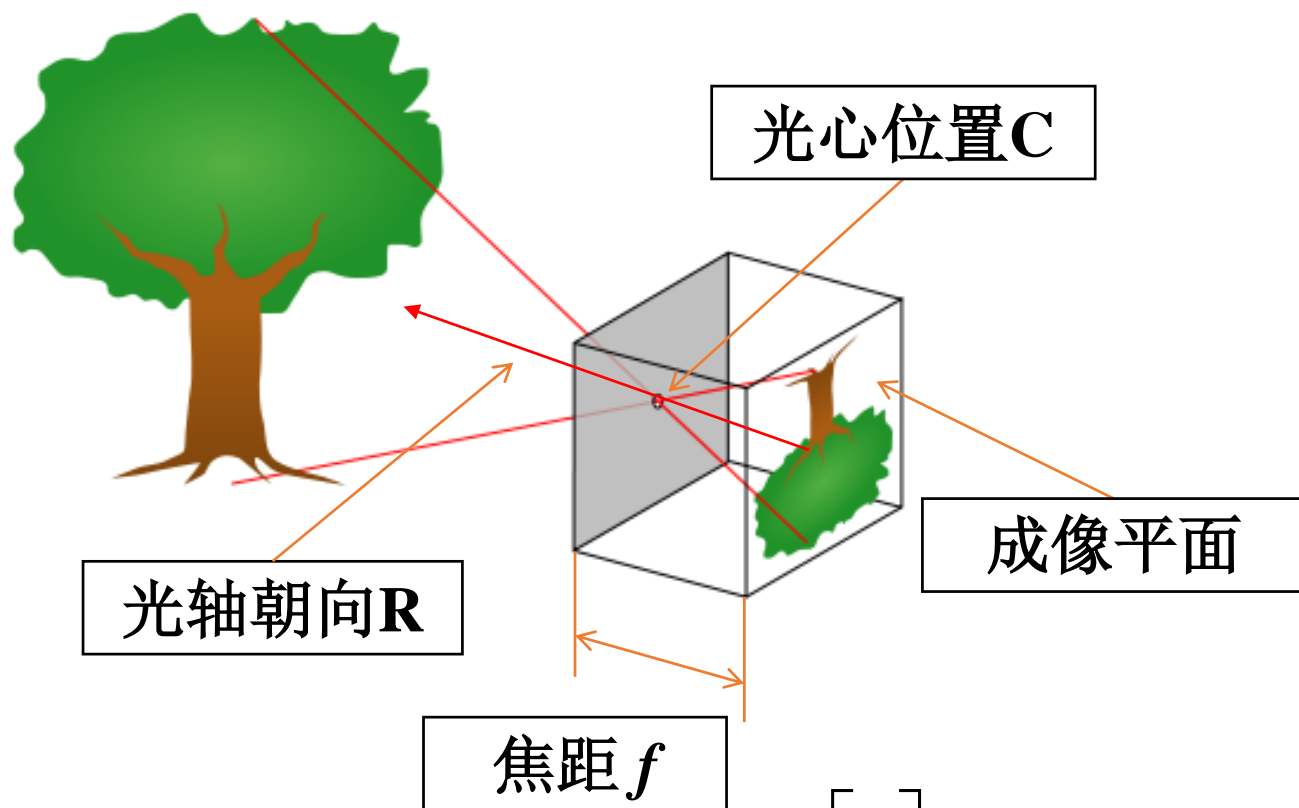
自由度：8

结构保持性：直线



# 图像配准中的变换模型

小孔相机模型



$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}_{3 \times 1} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}_{3 \times 3} \begin{bmatrix} \mathbf{R}_{3 \times 3} & | & -\mathbf{R}_{3 \times 3} \mathbf{C}_{3 \times 1} \end{bmatrix}_{3 \times 4} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}_{4 \times 1}$$

# 图像配准中的变换模型

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}_{3 \times 1} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}_{3 \times 3} \left[ \mathbf{R}_{3 \times 3} \mid -\mathbf{R}_{3 \times 3} \mathbf{C}_{3 \times 1} \right]_{3 \times 4} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}_{4 \times 1}$$

$$\mathbf{K} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{相机矩阵}$$

小孔相机模型由(K, R, C)完全决定（未考虑相机畸变参数）

# 图像配准中的变换模型

单应 (Homography)

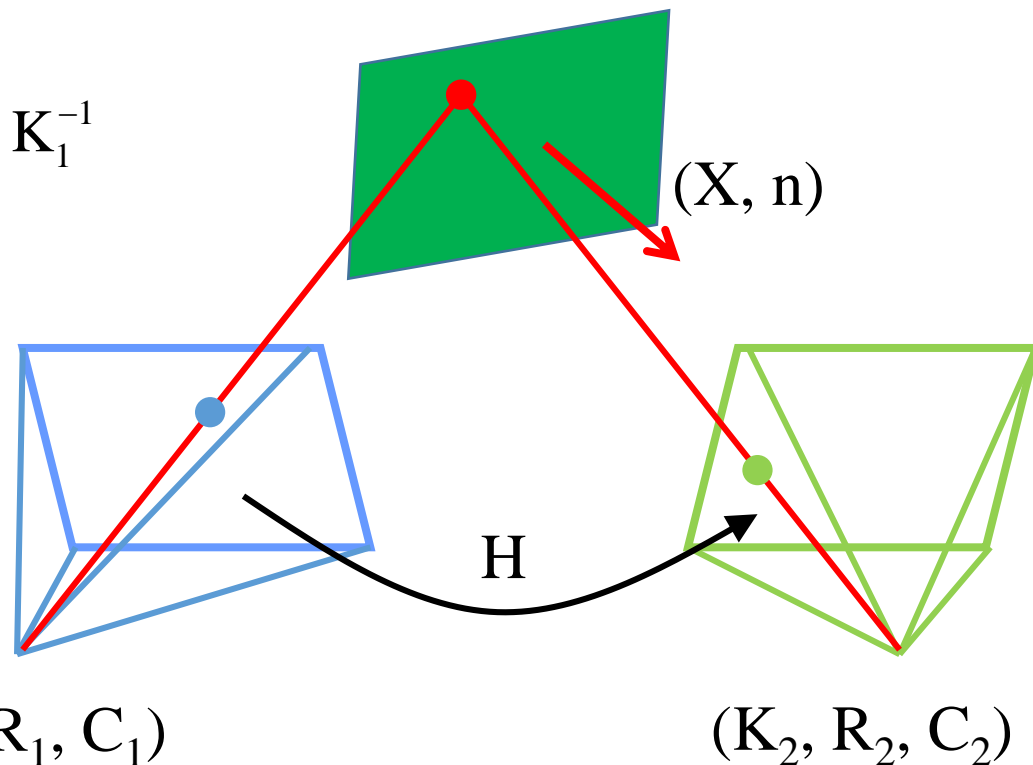
$$H = K_2(R_2R_1^{-1} + \frac{R_2(C_1 - C_2)n^T}{n^TX})K_1^{-1}$$

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} p_{00} & p_{01} & p_{02} \\ p_{10} & p_{11} & p_{12} \\ p_{20} & p_{21} & 1 \end{bmatrix}}_H \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

H

$(K_1, R_1, C_1)$

$(K_2, R_2, C_2)$



通过空间平面诱导的单应，自由度：8



# 图像配准中的变换模型

单应（Homography）特殊形式

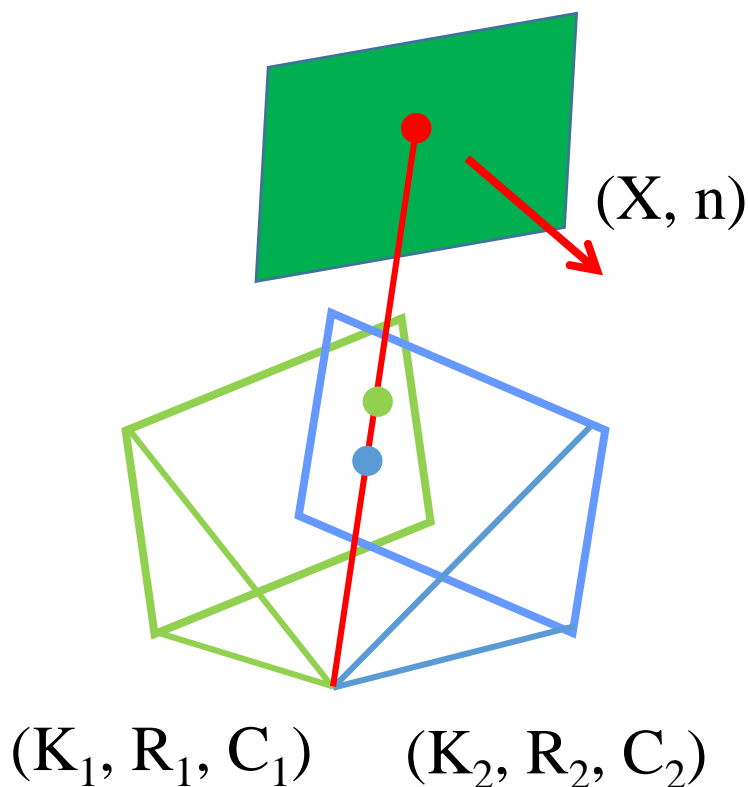
$$H = K_2 \left( R_2 R_1^{-1} + \frac{R_2 (C_1 - C_2) n^T}{n^T X} \right) K_1^{-1}$$



$$C_1 = C_2$$

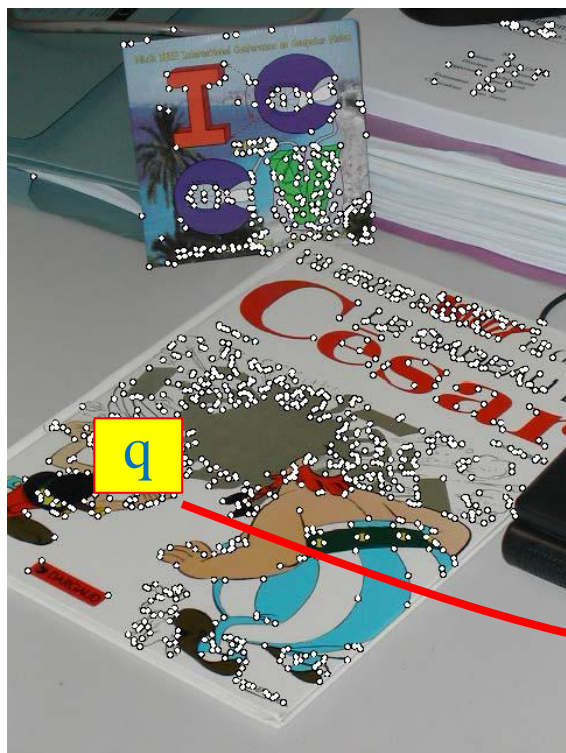
$$H = K_2 R_2 R_1^{-1} K_1^{-1}$$

与空间平面 $(X, n)$ 无关



相机纯旋转构成的单应，自由度：3-5（根据相机焦距是否已知），常用于全景图拼接。

# 变换模型的求解——基于特征



源图像S

变换 $W(.;x)$

$$\mathbb{R}^2 \mapsto \mathbb{R}^2$$



目标图像T

$$\varepsilon(x) = \sum_{(q \leftrightarrow q')} \|q' - W(q; x)\|^2$$

# 变换模型的求解——基于像素

源图像S



目标图像T



变换 $W(.;x)$

$$\mathbb{R}^2 \mapsto \mathbb{R}^2$$



变换图像

$$\mathcal{E}(x) = \sum_{q \in S} \|T(q) - S(W^{-1}(q; x))\|^2$$

SSD (Sum of Square Differences)

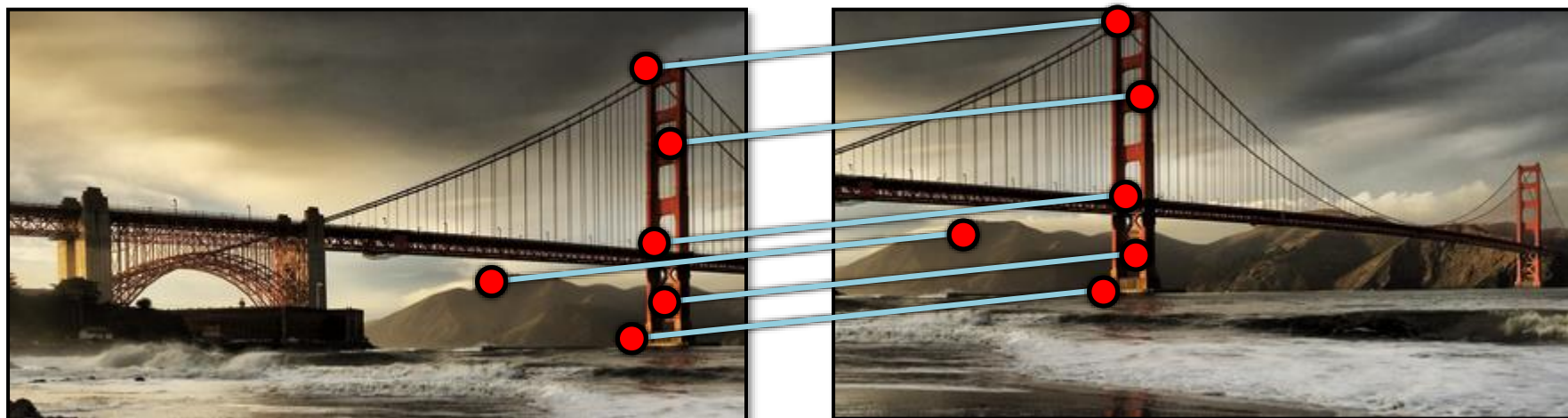
# 变换模型的求解

## 基于特征点 VS 基于像素

	基于特征点	基于像素
优点	<ul style="list-style-type: none"><li>变换模型计算只依赖特征点位置，对图像外观变化有一定鲁棒性；</li><li>能够计算宽基线时的变换；</li></ul>	<ul style="list-style-type: none"><li>使用了所有图像数据；</li><li>求解精度通常更高；</li><li>不需要特征检测和匹配；</li></ul>
缺点	<ul style="list-style-type: none"><li>没有使用所有图像数据；</li><li>需要进行特征检测和匹配。</li></ul>	<ul style="list-style-type: none"><li>容易受到图像外观变化影响；</li><li>一般只适用于窄基线时的变换。</li></ul>



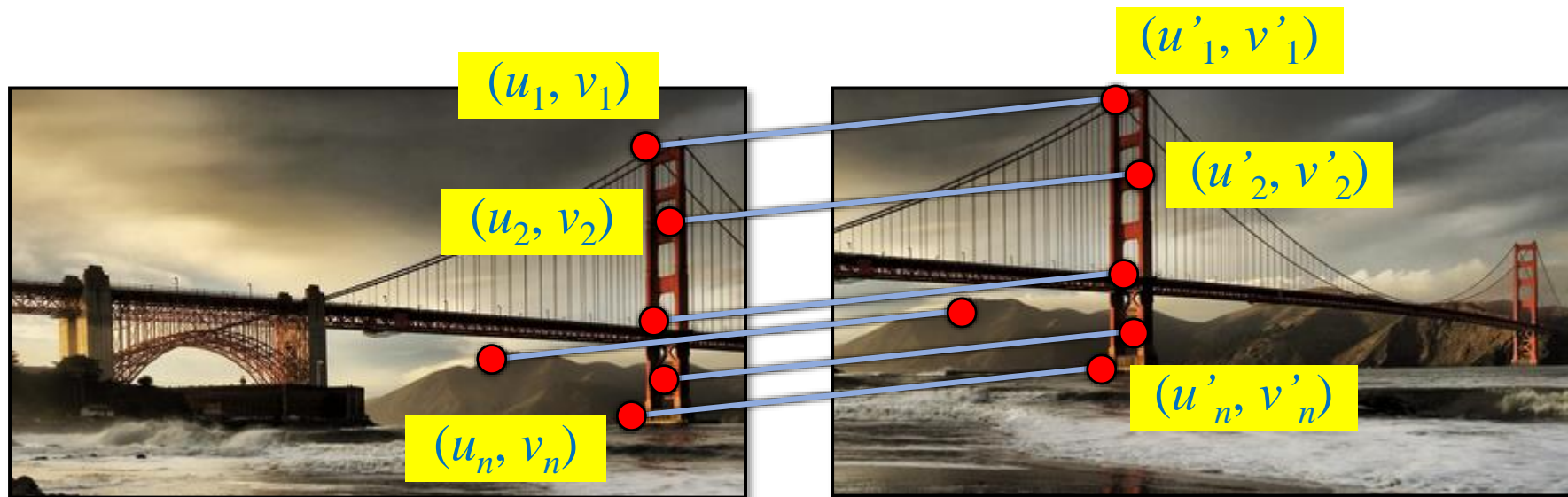
# 变换模型的求解——线性最小二乘



$(t_x, t_y)$

如何求解 $(t_x, t_y)$ ?

# 变换模型的求解——线性最小二乘



$$\begin{bmatrix} u'_i \\ v'_i \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \end{bmatrix} = \begin{bmatrix} u'_i - u_i \\ v'_i - v_i \end{bmatrix}$$

方程数量通常要远多于未知数数量， $(t_x, t_y)$ 为最小二乘解。

# 变换模型的求解——线性最小二乘

$\mathbf{x}=(t_x, t_y)$  为如下线性最小二乘问题的解:

$$\min_{t \in \mathbb{R}^2} \left\| \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \end{bmatrix} - \begin{bmatrix} u'_1 - u_1 \\ v'_1 - v_1 \\ \vdots \\ u'_n - u_n \\ v'_n - v_n \end{bmatrix} \right\|^2$$



$$\min_{\mathbf{x} \in \mathbb{R}^2} \left\| \mathbf{A}_{2n \times 2} \mathbf{x}_{2 \times 1} - \mathbf{b}_{2n \times 1} \right\|^2$$

# 变换模型的求解——线性最小二乘

$\mathbf{x}$ 为线性最小二乘问题的解： $\min_{\mathbf{x} \in \mathbb{R}^2} \|\mathbf{Ax} - \mathbf{b}\|^2$

定义目标函数： $\varepsilon(\mathbf{x}) = \|\mathbf{Ax} - \mathbf{b}\|^2$

$$= \|\mathbf{Ax}\|^2 + \|\mathbf{b}\|^2 - 2 \mathbf{x}^T \mathbf{A}^T \mathbf{b}$$

最优解 $\mathbf{x}$ 应满足： $\frac{\partial \varepsilon}{\partial \mathbf{x}} = 2 \mathbf{A}^T \mathbf{Ax} - 2 \mathbf{A}^T \mathbf{b} = 0 \quad \Rightarrow \quad \mathbf{Ax} = \mathbf{b}$

因此： $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b}$       其中： $\mathbf{A}^\dagger = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T$  称为 $\mathbf{A}$ 的伪逆

# 变换模型的求解——非线性最小二乘

线性最小二乘问题:  $\min_{\mathbf{x} \in \mathbb{R}^n} \sum_{i=1}^m \left\| \mathbf{a}_i^T \mathbf{x} - b_i \right\|^2$

x的线性函数

非线性最小二乘问题:  $\min_{\mathbf{x} \in \mathbb{R}^n} \sum_{i=1}^m \left\| f_i(\mathbf{x}) - b_i \right\|^2$

x的非线性函数



# 变换模型的求解——非线性最小二乘

$$\min_{\mathbf{x} \in \mathbb{R}^n} \sum_{i=1}^m \|f_i(\mathbf{x}) - b_i\|^2 \quad \text{with } f_i : \mathbb{R}^n \rightarrow \mathbb{R}^1, i = 1, \dots, m$$

写成紧凑形式:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{F}(\mathbf{x}) - \mathbf{b}\|^2 \quad \text{with } \mathbf{F} : \mathbb{R}^n \rightarrow \mathbb{R}^m$$

$$\mathbf{F}(\mathbf{x}) = \begin{pmatrix} \vdots \\ f_j(\mathbf{x}) \\ \vdots \end{pmatrix} \quad \mathbf{b}_{(m \times 1)} = \begin{pmatrix} \vdots \\ b_j \\ \vdots \end{pmatrix}$$

# 变换模型的求解——非线性最小二乘

$$\min_{\mathbf{x} \in \mathbb{R}^n} \varepsilon(\mathbf{x}) \quad \varepsilon(\mathbf{x}) = \|\mathbf{F}(\mathbf{x}) - \mathbf{b}\|^2$$

求解非线性最小二乘问题的方法：

- 非线性最小二乘是非线性优化的一类特殊形式；
- 针对一般非线性优化的梯度下降法（1阶）、牛顿法（2阶）；
- 针对非线性最小二乘的高斯牛顿法、LM法（Levenberg-Marquardt）。

# 变换模型的求解——非线性最小二乘

梯度下降法（Gradient Descent）

将 $\varepsilon(\mathbf{x})$ 在 $\mathbf{x}$ 处一阶Taylor展开：

$$\varepsilon(\mathbf{x}+\delta) = \varepsilon(\mathbf{x}) + \mathbf{g}(\mathbf{x})^T \delta + o(\delta^2) \quad \mathbf{g}(\mathbf{x}) = \nabla \varepsilon(\mathbf{x})$$

则当 $\|\delta\| \rightarrow 0$ 时， $\varepsilon(\mathbf{x})$ 的变化值可以表示为：

$$\begin{aligned} \Delta \varepsilon &= \varepsilon(\mathbf{x}+\delta) - \varepsilon(\mathbf{x}) \\ &= \mathbf{g}^T \delta \\ &= \|\mathbf{g}\| \|\delta\| \cos \theta \end{aligned}$$

$\theta$ 为 $\mathbf{x}$ 处梯度与向量 $\delta$ 间的夹角。显然， $\theta=\pi$ 时 $\varepsilon(\mathbf{x})$ 下降最快。

# 变换模型的求解——非线性最小二乘

## 梯度下降法（Gradient Descent）

梯度下降法的基本思想是将自变量向负梯度方向移动：

$$\mathbf{x} \leftarrow \mathbf{x} - \alpha \mathbf{g}(\mathbf{x})$$

$\alpha$ 为移动步长，最优的 $\alpha$ 可以通过一维搜索获得：

$$\min_{\alpha \geq 0} \varepsilon(\mathbf{x} - \alpha \mathbf{g}(\mathbf{x}))$$

寻找最优的 $\alpha$  计算代价很高，因此常使用一些简单的近似算法，如折半查找法（从 $\alpha = 1$ 开始不断将 $\alpha$  折半）。

# 变换模型的求解——非线性最小二乘

梯度下降法（Gradient Descent）

对于最小二乘问题： $\varepsilon(\mathbf{x}) = \|\mathbf{F}(\mathbf{x}) - \mathbf{b}\|^2$

梯度方向为： $\mathbf{g}(\mathbf{x}) = \mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x})$

Gradient Descent方向为： $\delta(\mathbf{x}) = -\mathbf{g}(\mathbf{x}) = -\mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x})$

其中： $\mathbf{J}(\mathbf{x})_{m \times n} = \frac{\partial \mathbf{F}(\mathbf{x})}{\partial \mathbf{x}}$  观测状态的Jacobian矩阵

$\mathbf{r}(\mathbf{x}) = \mathbf{F}(\mathbf{x}) - \mathbf{b}$  残差向量



# 变换模型的求解——非线性最小二乘

牛顿法（Newton Method）

梯度下降法是一阶收敛算法，收敛较慢。为加快收敛速度，可将 $\varepsilon(\mathbf{x})$ 在 $\mathbf{x}$ 处二阶Taylor展开：

$$\varepsilon(\mathbf{x}+\delta) \approx \varepsilon(\mathbf{x}) + \mathbf{g}(\mathbf{x})^T \delta + \frac{1}{2} \delta^T \mathbf{H}(\mathbf{x}) \delta$$

$\mathbf{g}(\mathbf{x})$  为 $n \times 1$ 的梯度向量， $\mathbf{H}(\mathbf{x})$  为 $n \times n$ 的Hessian矩阵。

将 $\varepsilon(\mathbf{x}+\delta)$ 对 $\delta$ 求导，并令导数为0，可得使 $\varepsilon(\mathbf{x}+\delta)$ 最小化的 $\delta$ ：

$$\mathbf{g}(\mathbf{x}) + \mathbf{H}(\mathbf{x})\delta = 0$$

$$\delta(\mathbf{x}) = -\mathbf{H}(\mathbf{x})^{-1} \mathbf{g}(\mathbf{x})$$

# 变换模型的求解——非线性最小二乘

牛顿法（Newton Method）

对于最小二乘问题： $\varepsilon(\mathbf{x}) = \|\mathbf{F}(\mathbf{x}) - \mathbf{b}\|^2$

梯度向量为： $\mathbf{g}(\mathbf{x}) = \mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x})$

Hessian矩阵为： $\mathbf{H}(\mathbf{x}) = \mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x}) + \mathbf{S}(\mathbf{x})$

$$\mathbf{J}(\mathbf{x}) = \frac{\partial \mathbf{F}(\mathbf{x})}{\partial \mathbf{x}}$$

$$\mathbf{r}(\mathbf{x}) = \mathbf{F}(\mathbf{x}) - \mathbf{b}$$

$\mathbf{S}(\mathbf{x})$ 包含 $\varepsilon(\mathbf{x})$ 的二阶导数

# 变换模型的求解——非线性最小二乘

牛顿法（Newton Method）

对于最小二乘问题： $\varepsilon(\mathbf{x}) = \|\mathbf{F}(\mathbf{x}) - \mathbf{b}\|^2$

Newton方向为： $\delta(\mathbf{x}) = -\left(\mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x}) + \mathbf{S}(\mathbf{x})\right)^{-1} \mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x})$

- 牛顿法是二阶收敛算法，收敛速度比梯度下降法快；
- 牛顿法需要求Hessian矩阵，即目标函数的二阶偏导数。

# 变换模型的求解——非线性最小二乘

## 高斯牛顿法（Gauss-Newton Method）

最小二乘问题的一种有效解法是高斯牛顿法，避免了牛顿法中对Hessian矩阵的计算。

对于最小二乘问题： $\varepsilon(\mathbf{x}) = \|\mathbf{F}(\mathbf{x}) - \mathbf{b}\|^2$

将 $\mathbf{F}(\mathbf{x})$ 在 $\mathbf{x}$ 处一阶Taylor展开：

$$\mathbf{F}(\mathbf{x} + \delta) \approx \mathbf{F}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\delta \quad \mathbf{J}(\mathbf{x}) = \nabla \mathbf{F}(\mathbf{x})$$

将上式代入 $\varepsilon(\mathbf{x})$ 可得：

$$\varepsilon(\mathbf{x} + \delta) \approx \|\mathbf{F}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\delta - \mathbf{b}\|^2$$

# 变换模型的求解——非线性最小二乘

高斯牛顿法（Gauss-Newton Method）

$$\varepsilon(\mathbf{x} + \delta) \approx \|\mathbf{F}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\delta - \mathbf{b}\|^2$$

由  $\mathbf{r}(\mathbf{x}) = \mathbf{F}(\mathbf{x}) - \mathbf{b}$  可得：

$$\varepsilon(\mathbf{x} + \delta) \approx \|\mathbf{r}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\delta\|^2$$

显然，如果要求 $\varepsilon(\mathbf{x} + \delta)$ 最小化，则应有：

$$\mathbf{r}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\delta = 0$$

则Gauss-Newton方向为： $\delta(\mathbf{x}) = -\mathbf{J}(\mathbf{x})^\dagger \mathbf{r}(\mathbf{x})$

$$= -\left(\mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x})\right)^{-1} \mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x})$$



# 变换模型的求解——非线性最小二乘

高斯牛顿法（Gauss-Newton Method）

高斯牛顿法与牛顿法的比较：

Newton方向为：
$$\delta(\mathbf{x}) = -\left(\mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x}) + \mathbf{S}(\mathbf{x})\right)^{-1} \mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x})$$

Gauss-Newton方向为：
$$\delta(\mathbf{x}) = -\left(\mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x})\right)^{-1} \mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x})$$

$\mathbf{S}(\mathbf{x})$ 可以忽略的条件：

- 最小二乘问题  $\varepsilon(\mathbf{x}) = \|\mathbf{F}(\mathbf{x}) - \mathbf{b}\|^2$  中 $\mathbf{F}(\mathbf{x})$ 适度非线性
- 残差 $\mathbf{r}(\mathbf{x})$ 比较小

大多数计算机视觉问题都满足这两个条件。

# 变换模型的求解——非线性最小二乘

## 阻尼高斯牛顿法（Damped Gauss-Newton Method）

高斯牛顿法的特点：

- 高斯牛顿法的收敛速度比梯度下降法快的多（超线性收敛，近似二阶收敛）；
- 高斯牛顿法不能保证收敛（梯度下降法和牛顿法收敛）。

阻尼高斯牛顿法： $\delta(\mathbf{x}, \lambda) = -\left(\mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x}) + \lambda \mathbf{I}\right)^{-1} \mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x})$

$\lim_{\lambda \rightarrow 0} \delta(\mathbf{x}, \lambda)$  则成为高斯牛顿法

$\lim_{\lambda \rightarrow \infty} \delta(\mathbf{x}, \lambda)$  则成为梯度下降法

# 变换模型的求解——非线性最小二乘

Levenberg-Marquardt Method (LM法)

LM法通过启发式方法在每一步动态调整 $\lambda$ :

$$\delta(\mathbf{x}, \lambda) = -\left(\mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x}) + \lambda \mathbf{I}\right)^{-1} \mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x})$$

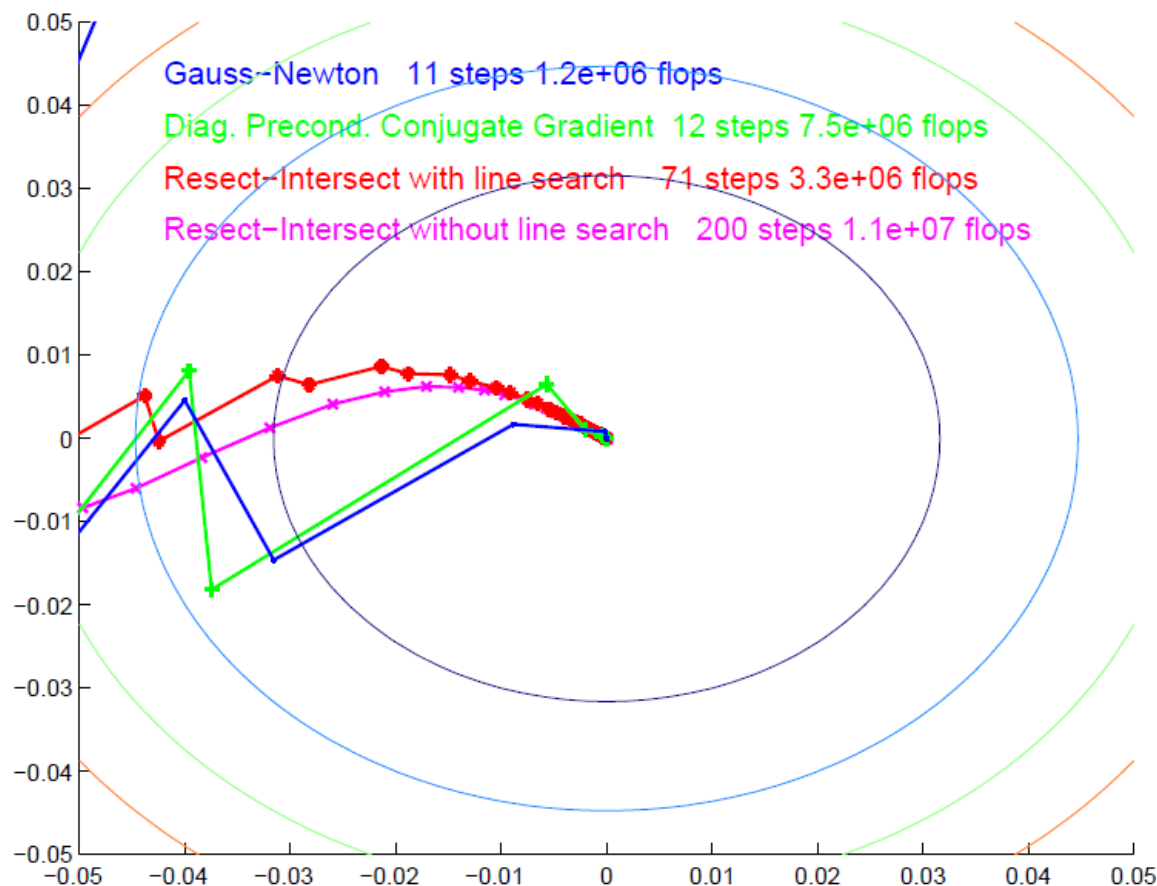
如果误差减少, 则令 $\lambda \leftarrow 0.1\lambda$

如果误差增大, 则令 $\lambda \leftarrow 10\lambda$

LM法是一种启发式的阻尼高斯牛顿法, 在计算机视觉中广泛使用。

# 变换模型的求解——非线性最小二乘

一阶和二阶收敛算法求解非线性最小二乘问题比较：



多数情况下我们会使用阻尼高斯牛顿法（如LM法）来处理计算机视觉问题中的最小二乘问题。

# 图像配准实例：全景图拼接

M. Brown and D. G. Lowe, [Automatic Panoramic Image Stitching using Invariant Features](#), International Journal of Computer Vision. 74(1): 59-73, 2007.



纯旋转拍摄无序图像



全景图像

# 图像配准实例：全景图拼接

算法基本流程：

1. 特征点检测与匹配；
2. 图像匹配与验证；
3. 变换模型参数计算；
4. 全景图矫正；
5. 增益补偿；
6. 多频段光度融合。

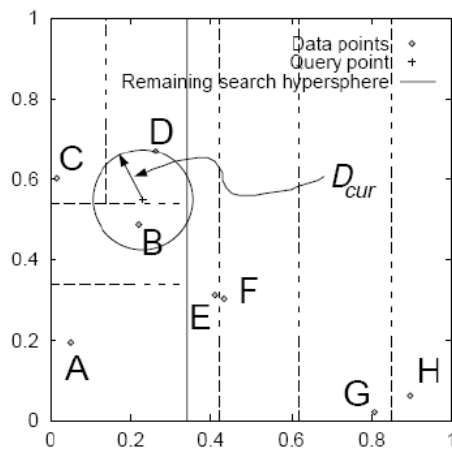


# 图像配准实例：全景图拼接

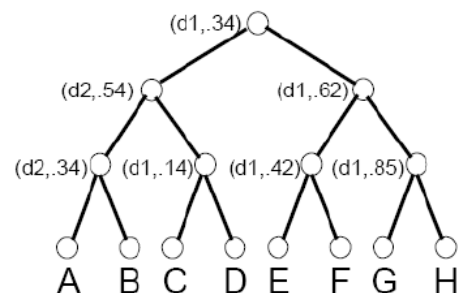
## 步骤1：特征点检测与匹配

- 检测每幅图像中的SIFT特征点（共 $n$ 幅图像）；
- 在每幅图像上建立k-d树；
- 每幅图像中每个特征点与其他图像匹配（只和每幅图像k-d树中4个最近邻特征点计算匹配值）。

回忆一下k-d树：



(a)



(b)

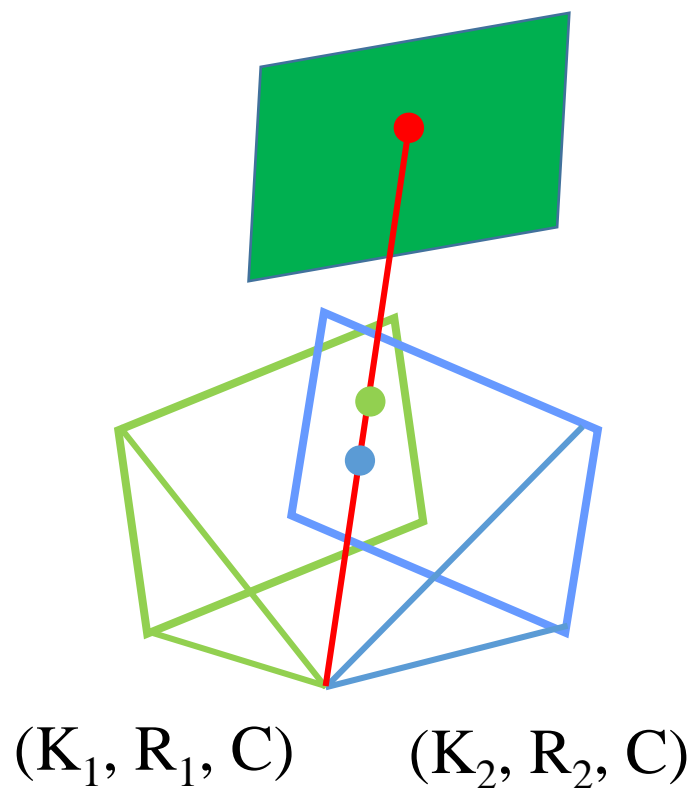
# 图像配准实例：全景图拼接

## 步骤2：图像匹配与验证

相机纯旋转拍摄，因此图像间变换为一个5自由度的单应变换  
（一般单应变换为8自由度）

$$H = K_2 R_2 R_1^{-1} K_1^{-1}$$

5自由度的单应变换构成一个非线性最小二乘问题，为了简化操作，放松为一个普通单应（8自由度）。



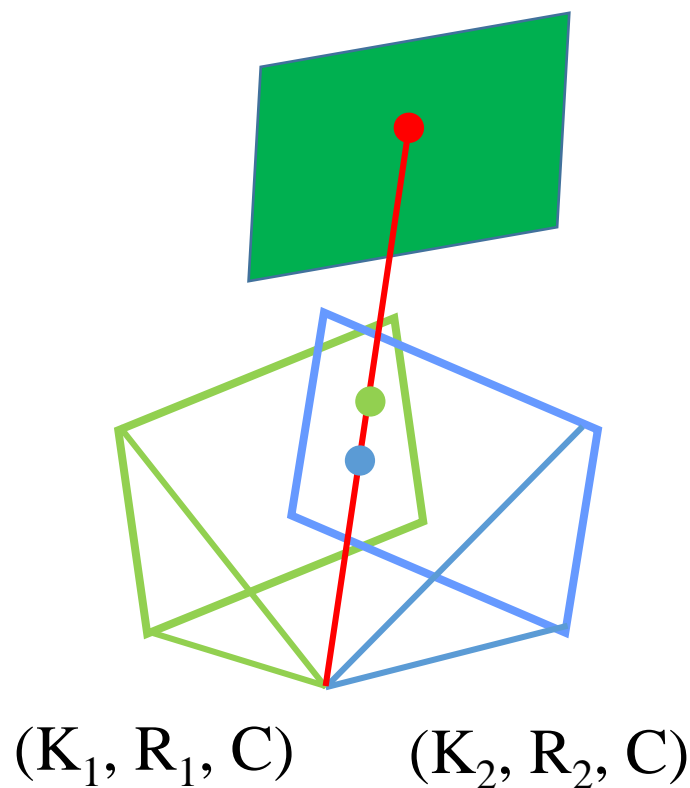
# 图像配准实例：全景图拼接

## 步骤2：图像匹配与验证

相机纯旋转拍摄，因此图像间变换为一个5自由度的单应变换  
(一般单应变换为8自由度)

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} p_{00} & p_{01} & p_{02} \\ p_{10} & p_{11} & p_{12} \\ p_{20} & p_{21} & 1 \end{bmatrix}}_{\mathbf{H}} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

$\mathbf{H}$ 包含8个未知量，可以通过  
4组对应点求得。



# 图像配准实例：全景图拼接

## 步骤2：图像匹配与验证

对每一幅图像，不需要和其他所有图像计算单应 $H$ ，只需选择匹配点最多的6幅图像进行计算。匹配点中不可避免的存在外点，因此使用RANSAC估计 $H$ 。

RANSAC计算 $H$ 流程：

1. 随机选4对匹配点构成8个线性方程组求 $H$ ；
2. 计算 $H$ 的一致集（所有符合 $H$ 的匹配点构成的集合）；
3. 步骤1-2循环500次；
4. 最大一致集中的所有匹配点构成超定线性方程组，通过线性最小二乘法求解 $H$ 。

# 图像配准实例：全景图拼接

## 步骤2：图像匹配与验证

假定两幅图像中正确匹配点所占比例为 $p=0.5$ ，则经过 $n$ 次RANSAC后找到正确单应H的概率为（ $r$ 为最小数据点个数，这里 $r=4$ ）：

$$p(\text{H is correct}) = 1 - (1 - (p)^r)^n$$

当 $n=500$ 时，正确的H未被找到的概率为 $1 \times 10^{-14}$

# 图像配准实例：全景图拼接

## 步骤2：图像匹配与验证

对每一对匹配图像，我们获得了它们之间的单应变换 $H$ ，下面需要验证这对图像是否确实匹配？

验证的基本依据：源图像经过单应变换后与目标图像产生一定面积的重叠区域，记这一区域内特征点总数为 $n_f$ ，记这一区域中符合 $H$ 的特征点（内点）数量为 $n_i$ ，如果 $n_i/n_f$ 足够大，则这两幅图像确实匹配。



# 图像配准实例：全景图拼接

## 步骤2：图像匹配与验证

- 定义指标 $m$ ， $m=1$ 表示图像匹配， $m=0$ 表示图像不匹配；
- 定义指标 $f^{(i)}$ ， $f^{(i)}=1$ 表示第 $i$ 个特征点为内点， $f^{(i)}=0$ 表示第 $i$ 个特征点为外点；

则总内点数符合伯努利分布（Bernoulli distribution）：

$$p(f^{(1:n_f)} | m = 1) = B(n_i; n_f, p_1) = \frac{n_f!}{n_i!(n_f - n_i)!} p_1^{n_i} (1 - p_1)^{(n_f - n_i)}$$

$$p(f^{(1:n_f)} | m = 0) = B(n_i; n_f, p_0) = \frac{n_f!}{n_i!(n_f - n_i)!} p_0^{n_i} (1 - p_0)^{(n_f - n_i)}$$

$p_1$ 和 $p_0$ 分别为图像匹配和不匹配时特征点为内点的概率

# 图像配准实例：全景图拼接

## 步骤2：图像匹配与验证

根据贝叶斯公式，图像正确匹配的后验概率为：

$$\begin{aligned} p(m=1 | f^{(1:n_f)}) &= \frac{p(f^{(1:n_f)} | m=1) p(m=1)}{p(f^{(1:n_f)})} \\ &= \frac{1}{1 + \frac{p(f^{(1:n_f)} | m=0) p(m=0)}{p(f^{(1:n_f)} | m=1) p(m=1)}} \end{aligned}$$

当后验概率  $p(m=1 | f^{(1:n_f)}) > p_{\min}$  时认为两幅图像正确匹配。

# 图像配准实例：全景图拼接

## 步骤2：图像匹配与验证

等价于当下面不等式成立时，两幅图像正确匹配：

$$\frac{B(n_i; n_f, p_1) p(m=1)}{B(n_i; n_f, p_0) p(m=0)} > \frac{1}{\frac{1}{p_{\min}} - 1}$$

给定参数：  $p_1=0.6$ ，  $p_0=0.1$ ，  $p(m=1)=10^{-6}$ ，  $p_{\min}=0.999$ ， 上述判定条件可以简化为：

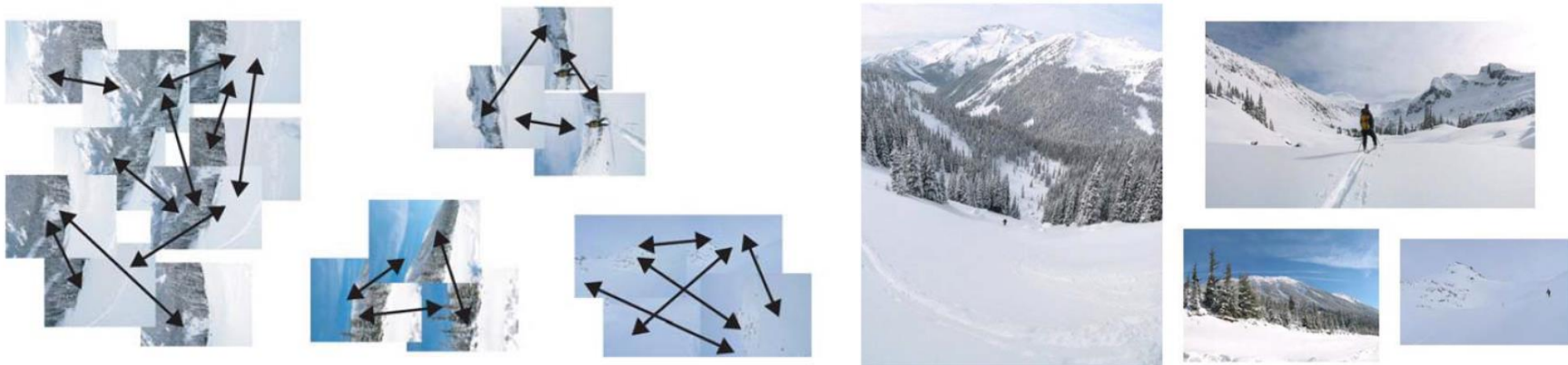
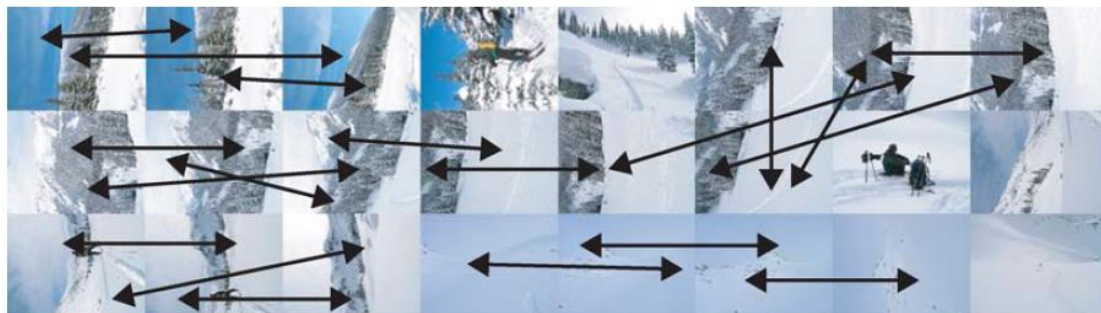
$$n_i > \alpha + \beta n_f$$

其中：  $\alpha=8.0$ ，  $\beta=0.3$

# 图像配准实例：全景图拼接

## 步骤2：图像匹配与验证

使用匹配验证条件剔除掉错误匹配图像对后，可以获得图像匹配的连接集，即判断输入图像可以分成几幅全景图。



# 图像配准实例：全景图拼接

## 步骤3：变换模型参数计算

特征点匹配中使用RANSAC得到的单应H是根据两两图像计算获得的，为保证匹配点的全局一致性，在变换模型参数计算中使用捆绑调整算法（Bundle Adjustment）。

相机纯旋转拍摄时对应一个5自由度的单应变换，令 $\mathbf{u}_i^k$ 和 $\mathbf{u}_j^k$ 为图像*i*和*j*中第*k*组匹配特征点，则将 $\mathbf{u}_j^k$ 通过单应变换到图像*i*上的位置为：

$$\mathbf{p}_{ij}^k = \mathbf{K}_i \mathbf{R}_i \mathbf{R}_j^{-1} \mathbf{K}_j^{-1} \mathbf{u}_j^k$$

变换误差为：

$$\mathbf{r}_{ij}^k = \mathbf{u}_i^k - \mathbf{p}_{ij}^k$$

# 图像配准实例：全景图拼接

## 步骤3：变换模型参数计算

误差函数定义为所有图像中所有匹配点的变换误差平方和：

$$\mathcal{E} = \sum_{i=1}^n \sum_{j \in I(i)} \sum_{k \in F(i,j)} h(\mathbf{r}_{ij}^k)$$

其中 $I(i)$ 表示所有与图像 $i$ 匹配的图像， $F(i,j)$ 表示图像 $i$ 与图像 $j$ 中的特征匹配点集合， $h()$ 为Huber误差函数(Huber, 1981)：

$$h(\mathbf{x}) = \begin{cases} |\mathbf{x}|^2, & \text{if } |\mathbf{x}| < \sigma \\ 2\sigma |\mathbf{x}| - \sigma^2, & \text{if } |\mathbf{x}| \geq \sigma \end{cases}$$

在优化循环初期使用 $\sigma=\infty$ ，在后期使用 $\sigma=2$ 。

# 图像配准实例：全景图拼接

## 步骤3：变换模型参数计算

误差函数定义为所有图像中所有匹配点的变换误差平方和：

$$\mathcal{E} = \sum_{i=1}^n \sum_{j \in I(i)} \sum_{k \in F(i,j)} h(\mathbf{r}_{ij}^k)$$

误差函数 $\mathcal{E}$ 的自变量包含每个相机的焦距和旋转矩阵，共 $4n$ 个参数（ $n$ 为图像数量），是一个非线性最小二乘问题，使用Levenberg-Marquardt (LM)算法对其进行求解。



# 图像配准实例：全景图拼接

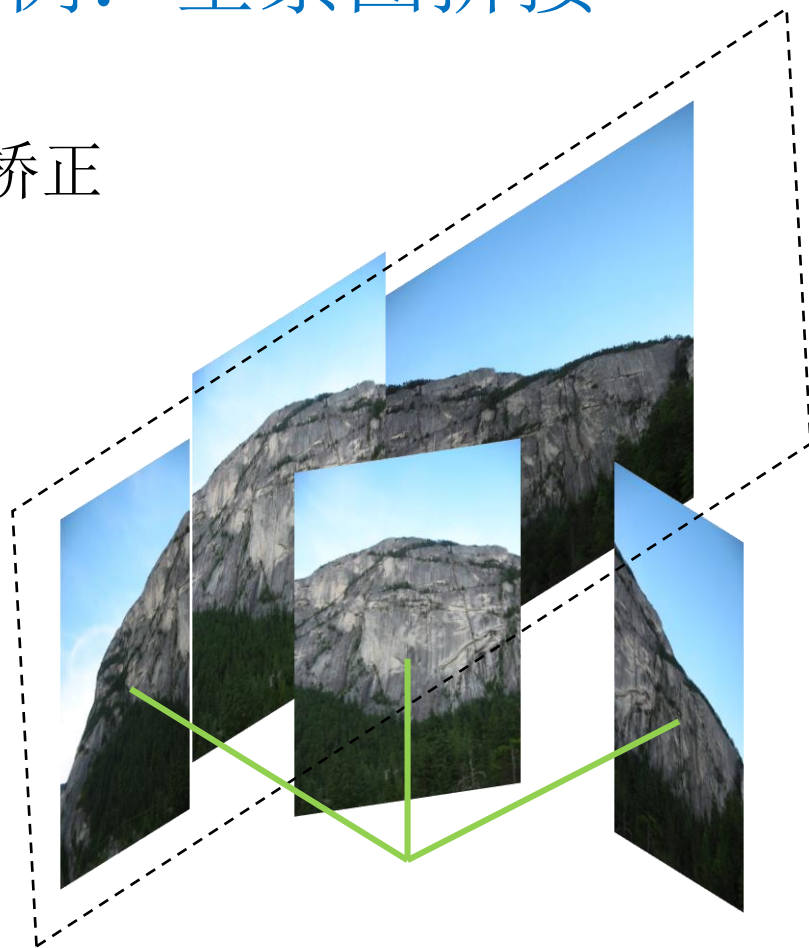
## 步骤4：全景图矫正

在上一步骤中，我们已经计算了每个相机的参数（焦距 $f$ 和姿态 $\mathbf{R}$ ），下面需要考虑如何把它们拼接起来。

平面坐标：最简单的拼接方法是将某一副图像作为参考图像（如图像 $i$ ），通过 $\mathbf{H}_{ij} = \mathbf{K}_i \mathbf{R}_i \mathbf{R}_j^{-1} \mathbf{K}_j^{-1}$  ( $j=1, \dots, n$ ) 计算单应，并将图像 $j$  ( $j=1, \dots, n$ ) 都转到图像 $i$ 所在平面完成拼接。

# 图像配准实例：全景图拼接

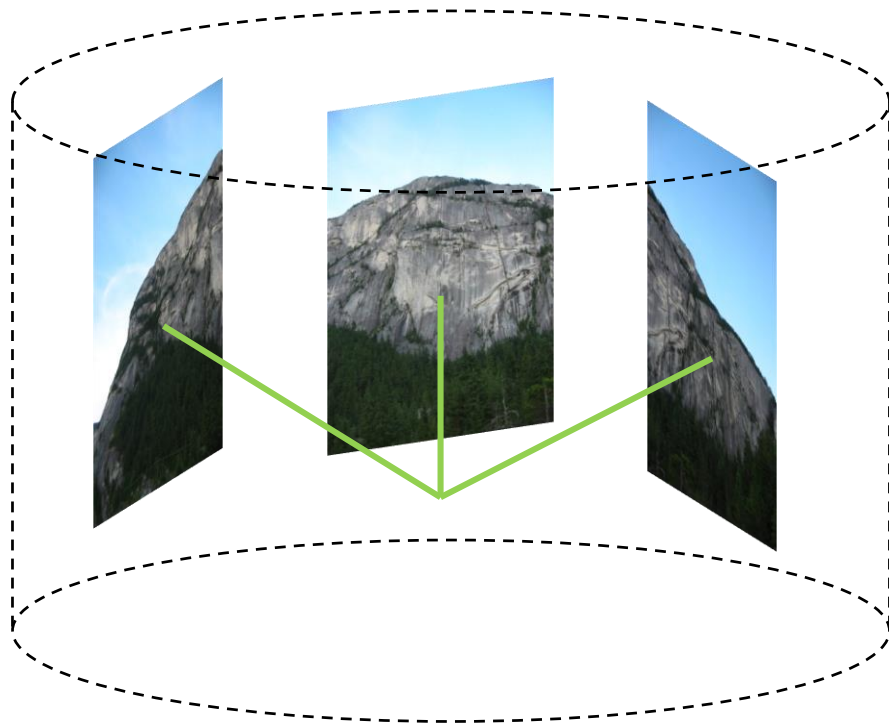
## 步骤4：全景图矫正



平面坐标拼接在源图像和目标图像间旋转角度较大时，图像会被严重拉伸，影响拼接效果。

# 图像配准实例：全景图拼接

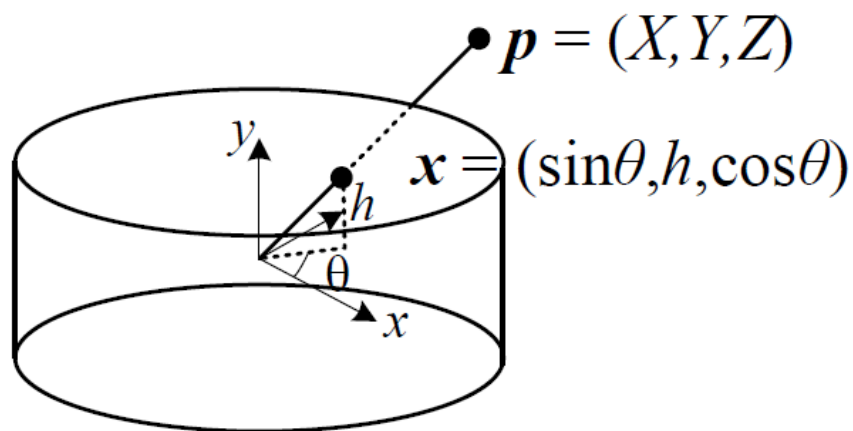
## 步骤4：全景图矫正



解决图像过度拉伸问题的方法是将图像变换到一个圆柱面上，这个圆柱中轴线通过相机光心，底面为单位圆。

# 图像配准实例：全景图拼接

## 步骤4：全景图矫正



$$u' = s \tan^{-1} \frac{u}{f}$$
$$v' = s \frac{v}{\sqrt{u^2 + f^2}}$$

$(u, v)$ : 图像平面坐标

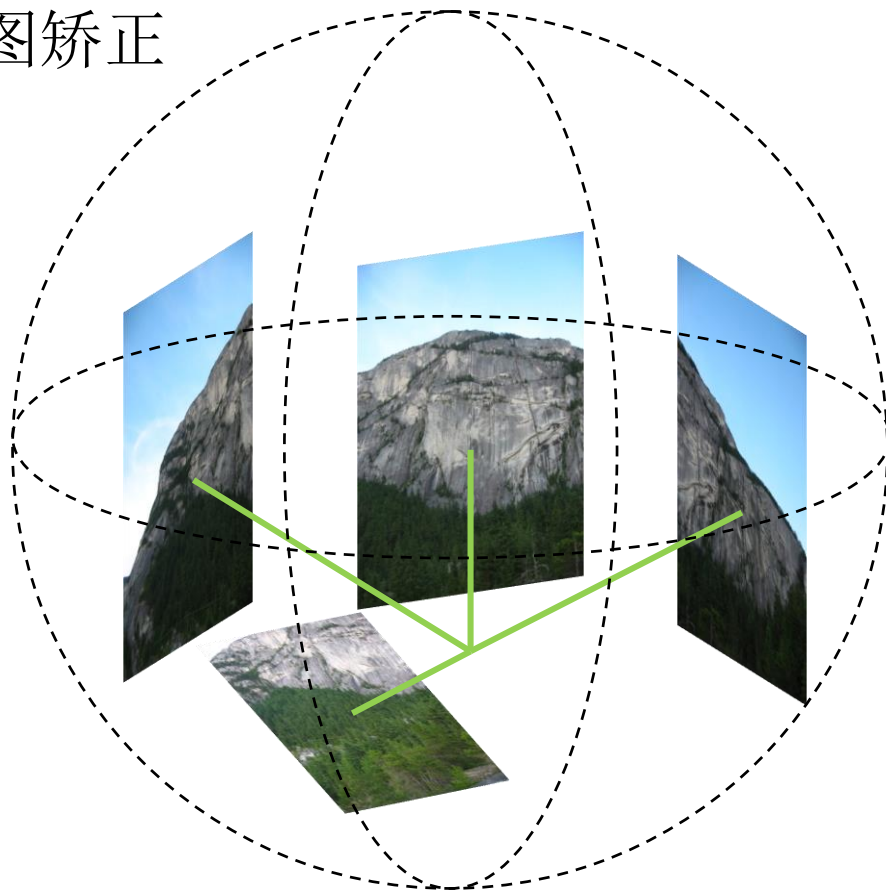
$(u', v')$ : 柱面展开平面坐标

柱面坐标：使用柱面坐标系进行拼接时，每个图像点变换为圆柱面上一点： $I(u, v) \rightarrow I(\theta, h) \rightarrow I(u', v')$

柱面坐标适合纯水平方向旋转的全景图拼接，如果在竖直方向也有旋转，则可能导致竖直方向旋转角大的图像被过度拉伸。

# 图像配准实例：全景图拼接

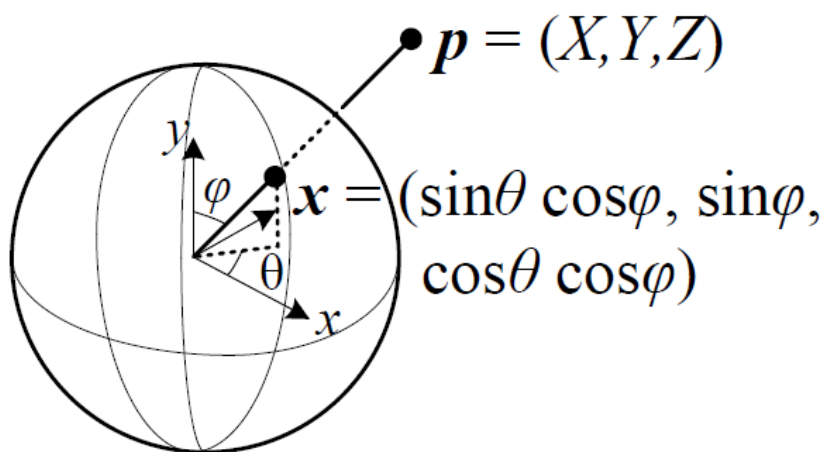
## 步骤4：全景图矫正



解决竖直方向图像过度拉伸问题的方法是将图像变换到一个单位球面上，这个球的球心和相机光心重合。

# 图像配准实例：全景图拼接

## 步骤4：全景图矫正



$$u' = s \tan^{-1} \frac{u}{f}$$

$$v' = s \tan^{-1} \frac{v}{\sqrt{u^2 + f^2}}$$

$(u, v)$ : 图像平面坐标

$(u', v')$ : 球面展开平面坐标

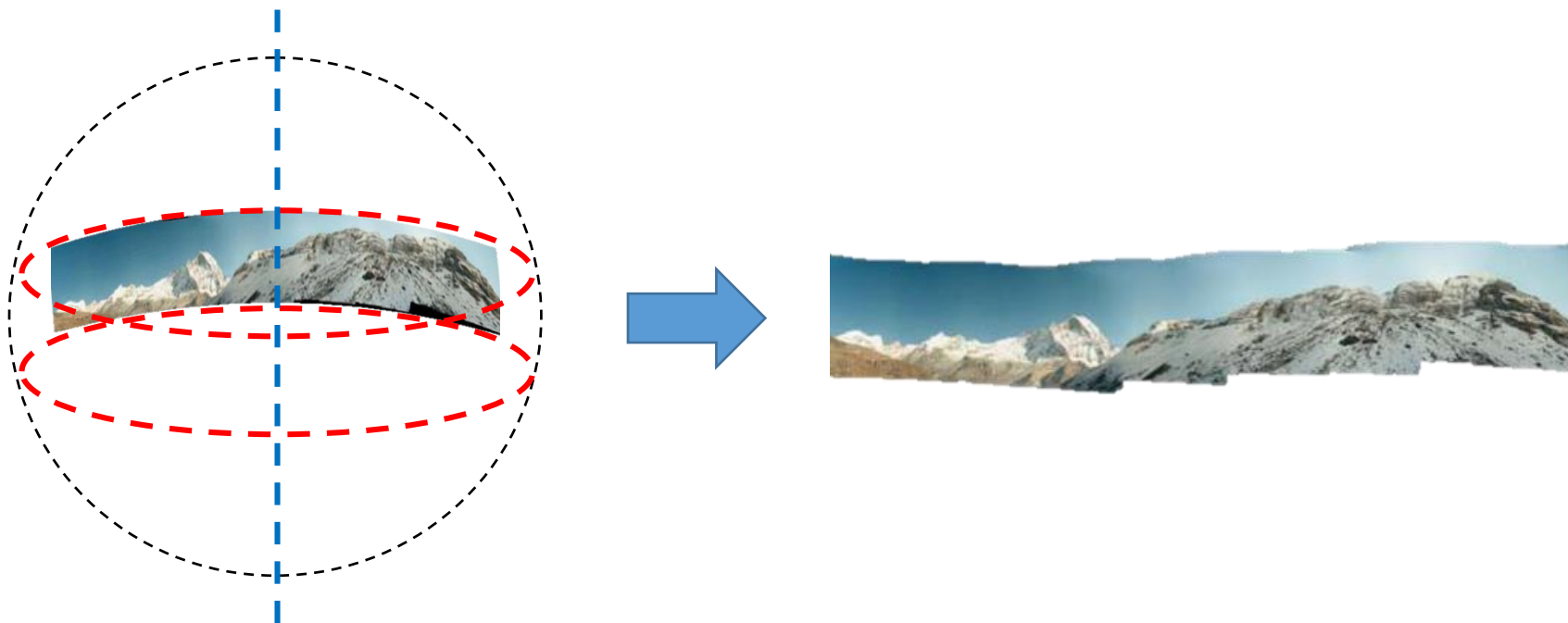
球面坐标：使用球面坐标系进行拼接时，每个图像点变换为球面上一点： $I(u, v) \rightarrow I(\theta, \phi) \rightarrow I(u', v')$

在后续步骤中，全景图的拼接均使用球面坐标。

# 图像配准实例：全景图拼接

## 步骤4：全景图矫正

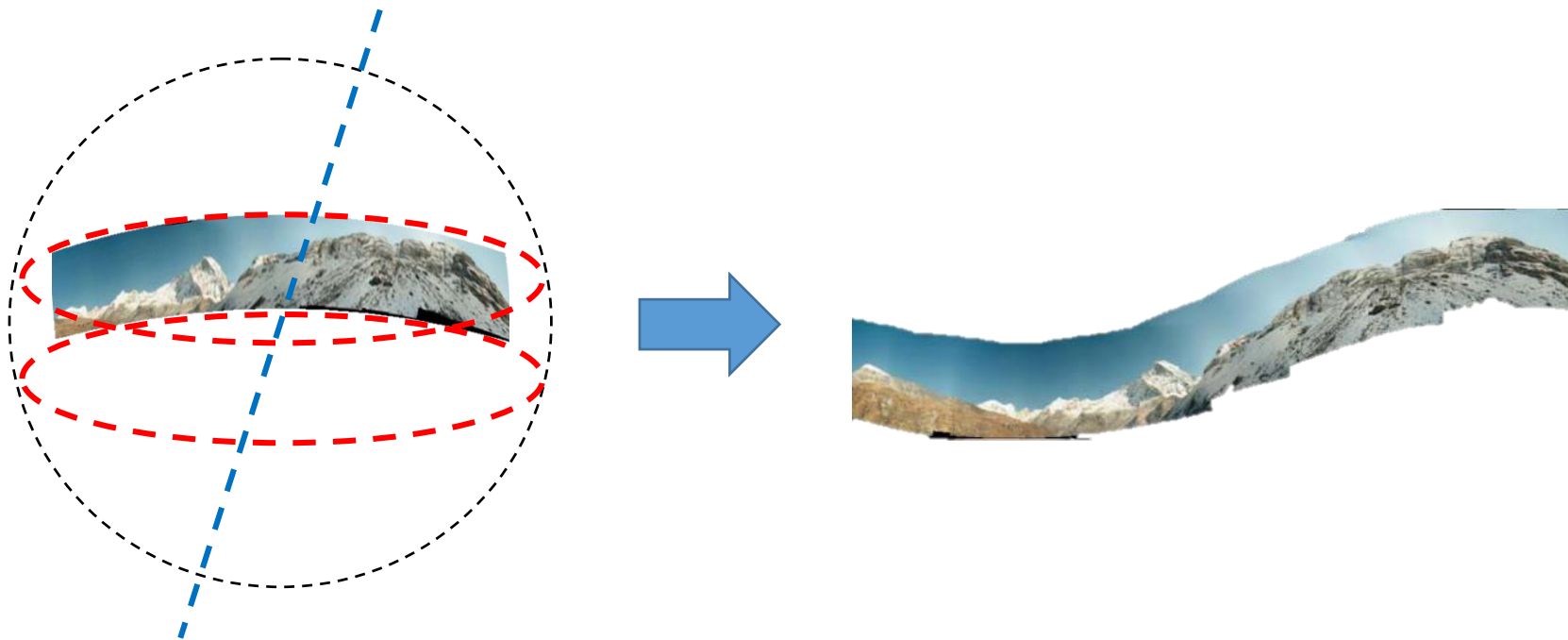
图像变换到球面坐标系时，根据球面坐标系主轴方向不同，最终获得的全景图会有很大不同。



# 图像配准实例：全景图拼接

## 步骤4：全景图矫正

图像变换到球面坐标系时，根据球面坐标系主轴方向不同，最终获得的全景图会有很大不同。





# 图像配准实例：全景图拼接

## 步骤4：全景图矫正

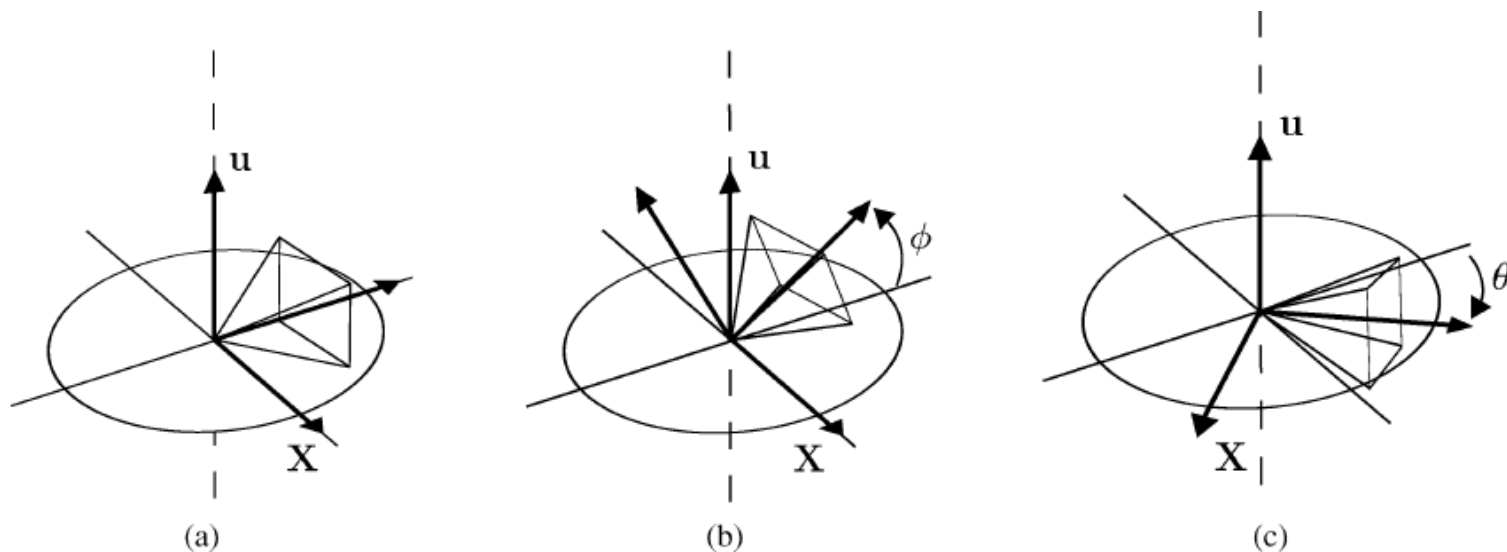
图像变换到球面坐标系时，根据球面坐标系主轴方向不同，最终获得的全景图会有很大不同。

因此，将图像变换到球面坐标系时需要确定合适的主轴方向，主轴方向与图像拍摄时的相机姿态直接相关。

# 图像配准实例：全景图拼接

## 步骤4：全景图矫正

全景图拍摄时，相机可能发生水平旋转、竖直旋转，但一般极少发生绕光轴旋转，因此相机 $X$ 轴通常会保持在一个平面，这一平面的法向 $u$ 可以作为一个理想的主轴方向。



# 图像配准实例：全景图拼接

## 步骤4：全景图矫正

全景图拍摄时，相机可能发生水平旋转、竖直旋转，但一般极少发生绕光轴旋转，因此相机 $\mathbf{X}$ 轴通常会保持在一个平面，这一平面的法向 $\mathbf{u}$ 可以作为一个理想的主轴方向。

我们期望得到的主轴方向 $\mathbf{u}$ 与所有图像坐标系 $\mathbf{X}$ 轴垂直，因此：

$$\mathbf{X}_i^T \mathbf{u} = 0 \quad i=1, \dots, n$$

写成最小二乘形式：

$$\min \sum_{i=1}^n \|\mathbf{X}_i^T \mathbf{u}\|^2$$

# 图像配准实例：全景图拼接

## 步骤4：全景图矫正

这是一个线性最小二乘问题，最优解 $\mathbf{u}$ 满足：

$$\nabla \sum_{i=1}^n \|\mathbf{X}_i^T \mathbf{u}\|^2 = 0 \quad \longrightarrow \quad \left( \sum_{i=1}^n \mathbf{X}_i \mathbf{X}_i^T \right) \mathbf{u} = 0$$

$\mathbf{u}$ 为 $3 \times 3$ 矩阵  $\sum_{i=1}^n \mathbf{X}_i \mathbf{X}_i^T$  最小特征值对应的特征向量。

以 $\mathbf{u}$ 作为主轴方向确定球面坐标系，根据相机参数将所有图像变换到球面上 $\mathbf{I}(u, v) \rightarrow \mathbf{I}(\theta, h) \rightarrow \mathbf{I}(u', v')$ ，获得全景图。

# 图像配准实例：全景图拼接

## 步骤4：全景图矫正



未校正全景图



校正后全景图

# 图像配准实例：全景图拼接

## 步骤5：增益补偿

在前面的步骤中已经确定了全景图拼接的几何结构参数（相机焦距、相机姿态、全景图坐标），已经可以将图像拼接为全景图像，但这样直接拼接时可能产生光度不一致的现象。



直接拼接结果

# 图像配准实例：全景图拼接

## 步骤5：增益补偿

为保证拼接后重叠区域的光度一致性，需要对每幅图像进行增益补偿。



$$\times g =$$



# 图像配准实例：全景图拼接

## 步骤5：增益补偿

为使全景图全局光度一致，我们在所有图像上定义全局光度误差函数：

$$e(g_1, \dots, g_n) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n N_{ij} \left( \frac{(g_i \bar{I}_{ij} - g_j \bar{I}_{ji})^2}{\sigma_N^2} + \frac{(1 - g_i)^2}{\sigma_g^2} \right)$$

$n$ 为图像数量， $g_i$ 为第 $i$ 幅图像的增益， $N_{ij}$ 为图像 $i$ 和 $j$ 重叠区域像素数量， $\bar{I}_{ij}$ 为图像 $i$ 在图像 $i$ 与 $j$ 重叠区域的亮度均值， $\sigma_N$ 为归一化光度误差方差（设为10）， $\sigma_g$ 为增益方差（设为0.1）。

最小化 $e(\cdot)$ 是一个二次优化问题，可通过超定线性方程组求得最优增益 $g_1 \sim g_n$



# 图像配准实例：全景图拼接

## 步骤5：增益补偿



直接拼接结果



增益补偿结果

# 图像配准实例：全景图拼接

## 步骤6：多频段光度融合

由于光晕、光心移动、配准误差、镜头畸变等因素的影响，即使经过增益补偿后，全景图中任然可能存在某些光度不一致的区域，需要进一步进行修正。

图像重叠区域可能包含多幅图像，一种有效的光度修正方式是将重叠区域像素用多幅图像进行加权融合。

# 图像配准实例：全景图拼接

## 步骤6：多频段光度融合

令 $I^i(u,v)$ 表示第 $i$ 幅图像上的像素点 $(u,v)$ ， $W^i(u,v)=\omega(u)\omega(v)$ 表示像素点 $(u,v)$ 的融合权重， $\omega(\cdot)$ 的值在图像中心为1，在图像边缘为0，中心和边缘之间线性递减。

将 $I^i(u,v)$ 和 $W^i(u,v)$ 均转换到球面坐标系 $I^i(\theta,\phi)$ 和 $W^i(\theta,\phi)$ ，则全景图像中每一个像素点可以通过下式进行光度融合：

$$I^{\text{linear}}(\theta,\phi) = \frac{\sum_{i=1}^n I^i(\theta,\phi) W^i(\theta,\phi)}{\sum_{i=1}^n W^i(\theta,\phi)}$$

这一融合称为线性光度融合。

# 图像配准实例：全景图拼接

## 步骤6：多频段光度融合

如果在图像配准过程中出现微小误差，那么线性光度融合在图像高频部分可能出现模糊，因此一种更鲁棒的融合方法是多频段光度融合。



线性光度融合



多频段光度融合

# 图像配准实例：全景图拼接

## 步骤6：多频段光度融合

首先将每个像素点的初始融合权重设为如下形式：

$$W_{\max}^i(\theta, \phi) = \begin{cases} 1 & \text{if } W^i(\theta, \phi) = \arg \max_{j=1, \dots, n} W^j(\theta, \phi) \\ 0 & \text{otherwise} \end{cases}$$

对于第*i*幅图像上的像素点 $(\theta, \phi)$ ，只有当它与图像*i*的中心距离比其他图像都小时权重才为1，否则权重为0

# 图像配准实例：全景图拼接

## 步骤6：多频段光度融合

定义频段图和频段权重如下：

$$B_{\sigma}^i(\theta, \phi) = I^i(\theta, \phi) - I_{\sigma}^i(\theta, \phi) \quad I_{\sigma}^i(\theta, \phi) = I^i(\theta, \phi) * g_{\sigma}(\theta, \phi)$$

$$W_{\sigma}^i(\theta, \phi) = W_{\max}^i(\theta, \phi) * g_{\sigma}(\theta, \phi)$$

$B_{\sigma}^i$ 为频段图， $W_{\sigma}^i$ 为频段权重， $g_{\sigma}$ 为为方差为 $\sigma$ 的高斯卷积核

$B_{\sigma}^i$ 为图像 $I^i$ 的高斯差分图，对应频段 $[0, \sigma]$

# 图像配准实例：全景图拼接

## 步骤6：多频段光度融合

接着可以定义一系列不同的频段图和频段权重：

$$\mathbf{B}_{(k+1)\sigma}^i = \mathbf{I}_{k\sigma}^i - \mathbf{I}_{(k+1)\sigma}^i \quad \mathbf{I}_{(k+1)\sigma}^i = \mathbf{I}_{k\sigma}^i * \mathbf{g}_{\sigma'}$$

$$\mathbf{W}_{(k+1)\sigma}^i = \mathbf{W}_{k\sigma}^i * \mathbf{g}_{\sigma'} \quad \sigma' = \sqrt{2k+1}\sigma$$

$\mathbf{B}_{(k+1)\sigma}^i$ 为图像 $\mathbf{I}^i$ 在频段 $[k\sigma, (k+1)\sigma]$ 的高斯差分图。则多频段光度融合定义为：

$$\mathbf{I}_{k\sigma}^{\text{multi}}(\theta, \phi) = \frac{\sum_{i=1}^n \mathbf{B}_{k\sigma}^i(\theta, \phi) \mathbf{W}_{k\sigma}^i(\theta, \phi)}{\sum_{i=1}^n \mathbf{W}_{k\sigma}^i(\theta, \phi)}$$



# 图像配准实例：全景图拼接

## 步骤6：多频段光度融合

多频段光度融合的核心思想：

- 对图像中的低频区域在较大的空间范围融合以保证光度一致；
- 对图像中的高频区域在较小的空间范围融合以保证细节。



# 图像配准实例：全景图拼接

## 步骤6：多频段光度融合



增益补偿结果



多频段光度融合结果

# 图像配准实例：全景图拼接

全景图拼接算法完整流程：

1. 提取每幅图像上的SIFT特征点；
2. 使用k-d树寻找每个特征点的 $k$ 个最近邻匹配点进行匹配；
3. 对每一幅图像：
  - (1) 选择匹配点最多的 $m$ 幅图像作为匹配备选图像；
  - (2) 使用RANSAC计算两幅匹配图像间的单应；
  - (3) 使用贝叶斯后验概率验证图像匹配结果；
4. 寻找图像匹配的连接集，确定全景图数量；
5. 对每一个连接集：
  - (1) 使用捆绑调整算法计算相机参数（焦距 $f$ 和旋转 $R$ ）；
  - (2) 对全景图进行增益补偿和多频段光度融合。

# 总结—本节课内容回顾

图像配准的变换模型：平移、欧式、相似、仿射、单应

图像配准的求解方法：线性/非线性最小二乘

一个完整的全景图拼接实例

# 总结—参考文献

- Brown, M. and Lowe, D. [Automatic panoramic image stitching using invariant features](#). International Journal of Computer Vision, 74(1):59–73, 2007.
- Szeliski, R. [Image Alignment and Stitching: A Tutorial](#). Foundations and Trends in Computer Graphics and Vision, 2(1): 1–104, 2007.