

遥感图像与视频处理技术

第四讲： 遥感图像配准技术 (Image registration)

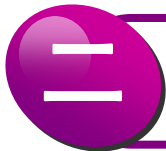
计算机学院
郑 锦



目 录



特征点提取算法



特征线提取算法

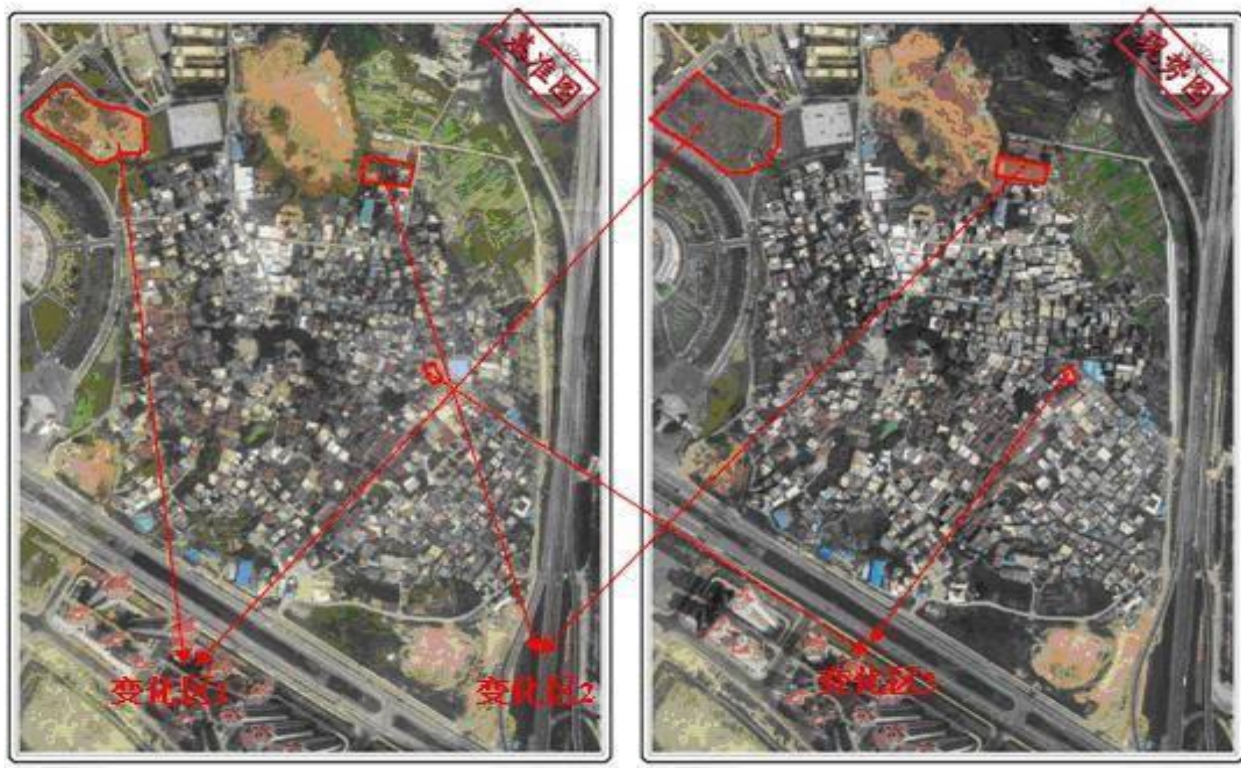


特征的定位算法



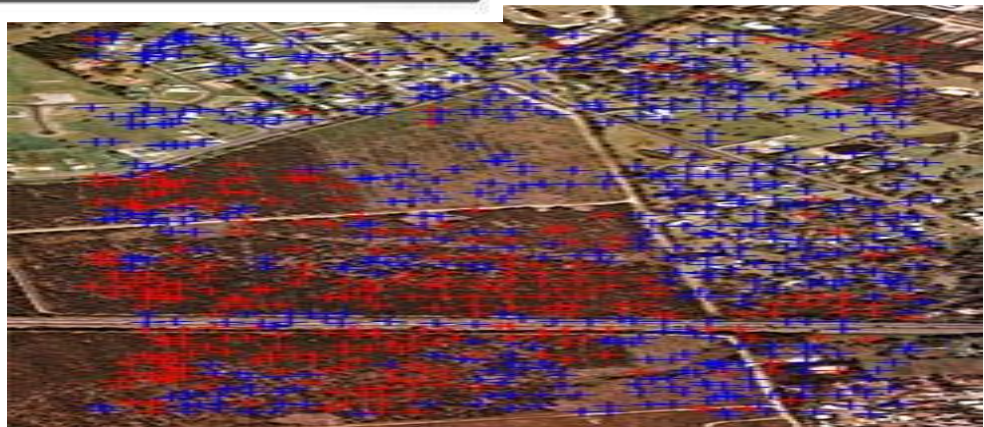
最小二乘匹配

1.1 特征点提取算法

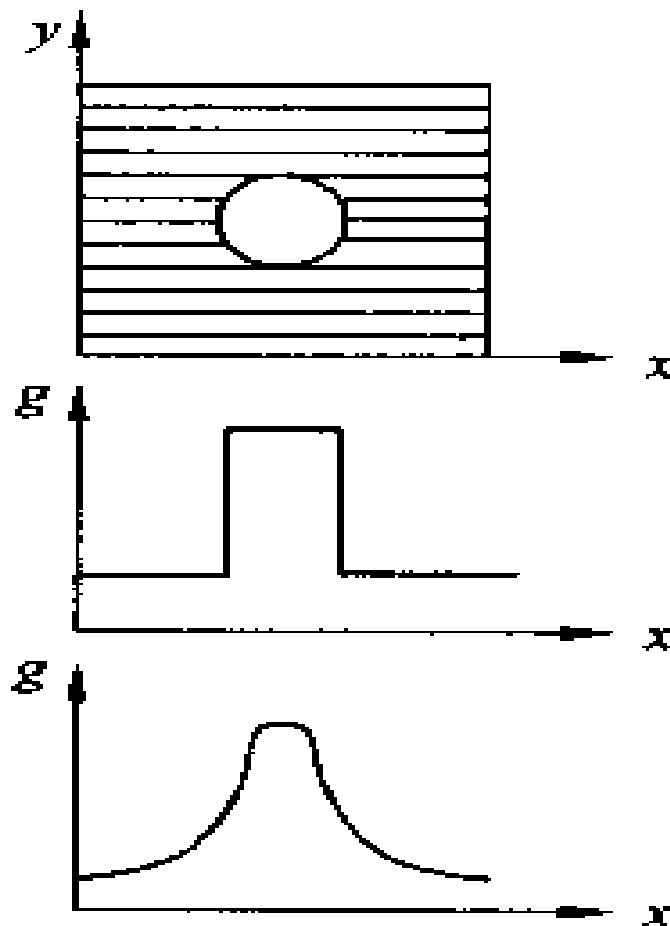


配准技术的流程如下：
首先对两幅图像进行特征提取得到**特征点**；
通过进行相似性度量找到匹配的特征图像**配准点对**；
然后通过匹配的特征点对得到图像**空间坐标变换参数**；
最后由坐标变换参数进行图像配准

点特征主要指明显点，
提取点特征的算子称为
兴趣算子或有利算子

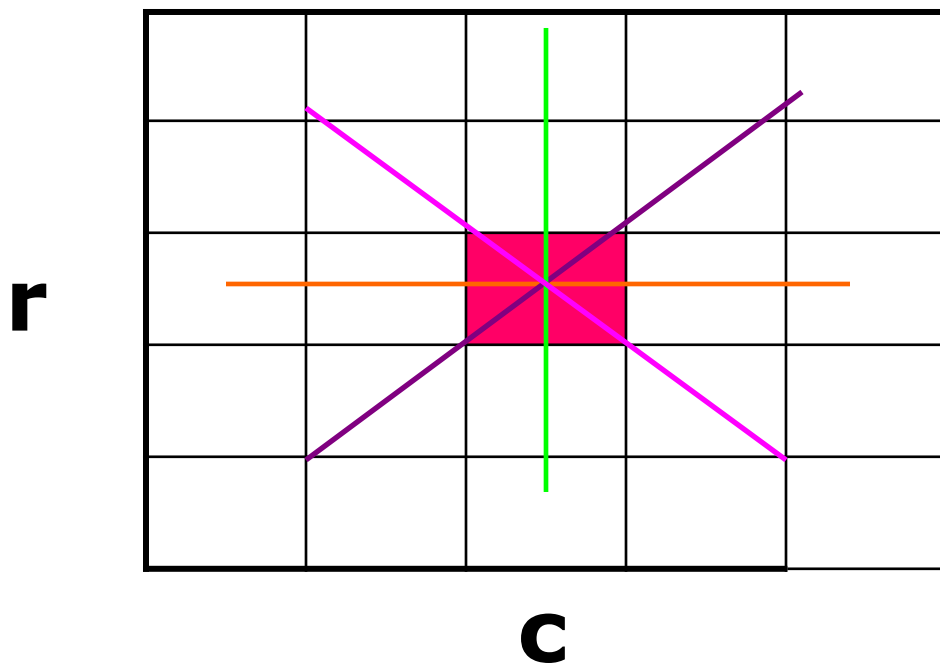


1.1 特征点的灰度特性



1.2 Moravec算子

Moravec于1977年提出利用灰度方差提取点特征的算子



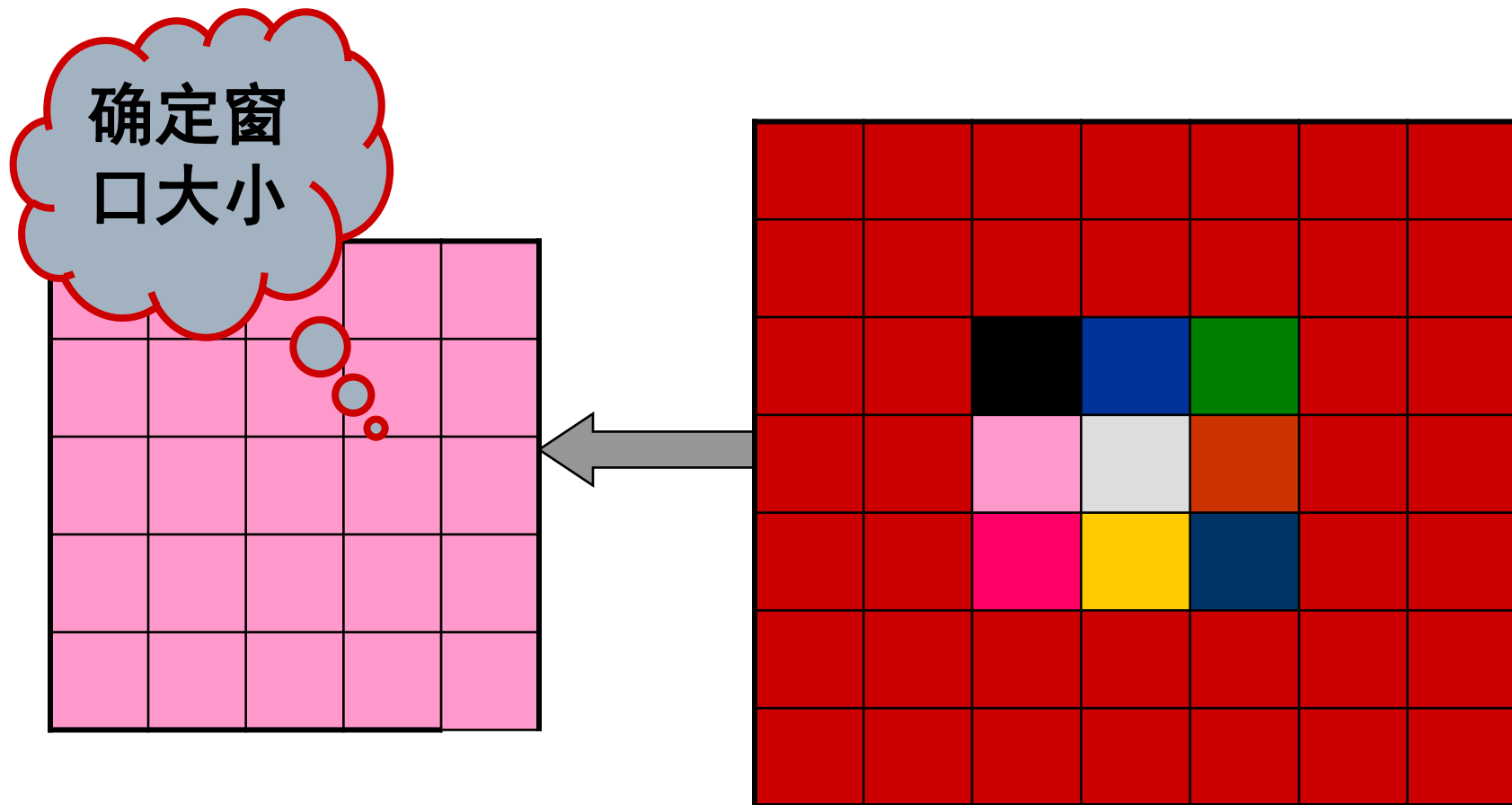
水平、垂直、对角线、反对角线四个方向的灰度方差

1.2 Moravec算子—计算各像元的兴趣值 $IV_{c,r}$

$$\left. \begin{aligned} V_1 &= \sum_{i=-k}^{k-1} (g_{c+i,r} - g_{c+i+1,r})^2 \\ V_2 &= \sum_{i=-k}^{k-1} (g_{c+i,r+i} - g_{c+i+1,r+i+1})^2 \\ V_3 &= \sum_{i=-k}^{k-1} (g_{c,r+i} - g_{c,r+i+1})^2 \\ V_4 &= \sum_{i=-k}^{k-1} (g_{c+i,r-i} - g_{c+i+1,r-i-1})^2 \end{aligned} \right\}$$

$$IV_{c,r} = \min \{V_1, V_2, V_3, V_4\}$$

1.2 Moravec算子—选取极值点



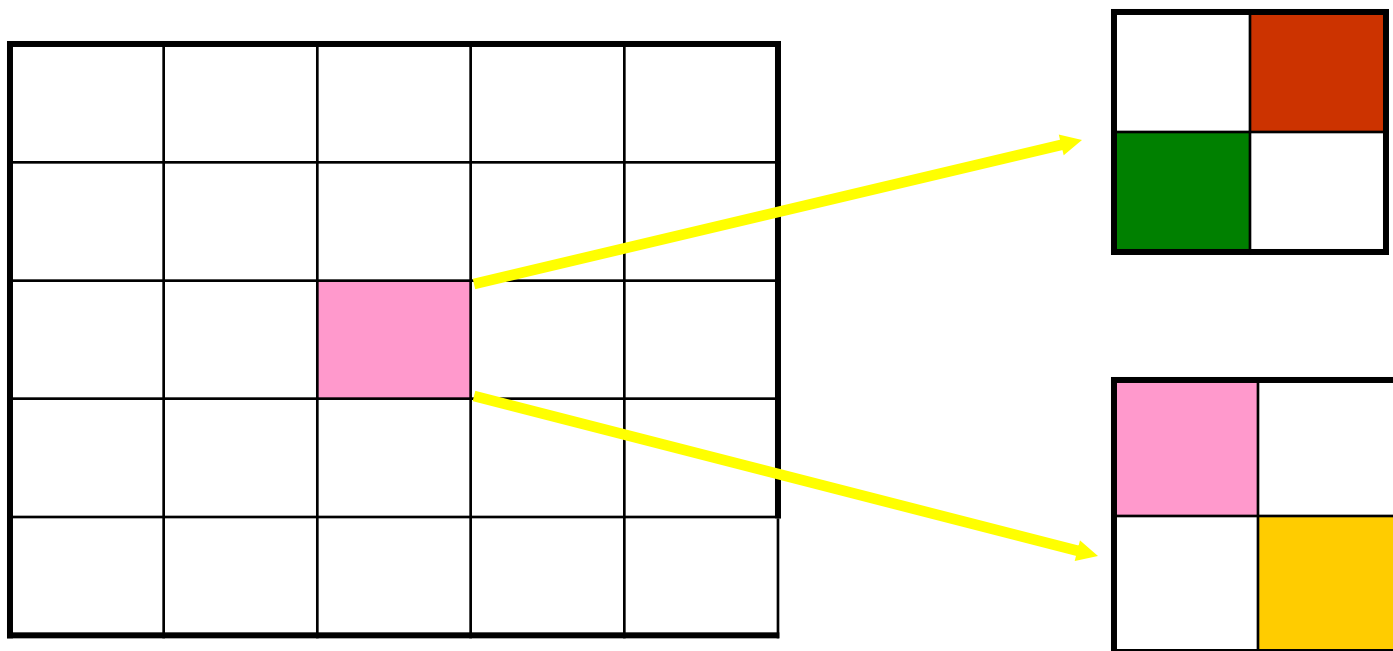
- (1) 给定经验阈值，将兴趣值大于阈值的点作为候选点
- (2) 选取候选点中的极值点作为特征点

1.2 Moravec算子——选取极值点

综上所述，Moravec算子是在四个主要方向上，选择具有最大——最小灰度方差的点作为特征点。

1.3 Forstner算子

计算各像素的**Robert's**梯度和像素 (c, r) 为中心的一个窗口的**灰度协方差矩阵**，在影像中寻找具有尽可能小而接近圆的误差椭圆的点作为特征点。



1.3 Forstner算子

(1) 计算各像素的Robert's梯度

$$\left. \begin{aligned} g_u &= \frac{\partial g}{\partial u} = g_{i+1,j+1} - g_{i,j} \\ g_v &= \frac{\partial g}{\partial v} = g_{i,j+1} - g_{i+1,j} \end{aligned} \right\}$$

1.3 Forstner算子

(2) 计算窗口中灰度的协方差矩阵

$$Q = N^{-1} = \begin{bmatrix} \sum g_u^2 & \sum g_u g_v \\ \sum g_v g_u & \sum g_v^2 \end{bmatrix}^{-1}$$

$$\sum g_u^2 = \sum_{i=c-k}^{c+k-1} \sum_{j=r-k}^{r+k-1} (g_{i+1,j+1} - g_{i,j})^2$$

$$\sum g_v^2 = \sum_{i=c-k}^{c+k-1} \sum_{j=r-k}^{r+k-1} (g_{i,j+1} - g_{i+1,j})^2$$

$$\sum g_u g_v = \sum_{i=c-k}^{c+k-1} \sum_{j=r-k}^{r+k-1} (g_{i+1,j+1} - g_{i,j})(g_{i,j+1} - g_{i+1,j})$$

1.3 Forstner算子

(3) 计算兴趣值q与w

$$w = \frac{1}{\text{tr}Q} = \frac{\text{Det}N}{\text{tr}N}$$

DetN代表矩阵N之行列式

$$q = \frac{4\text{Det}N}{(\text{tr}N)^2}$$

trN代表矩阵N之迹

1.3 Forstner算子

(4) 确定待选点

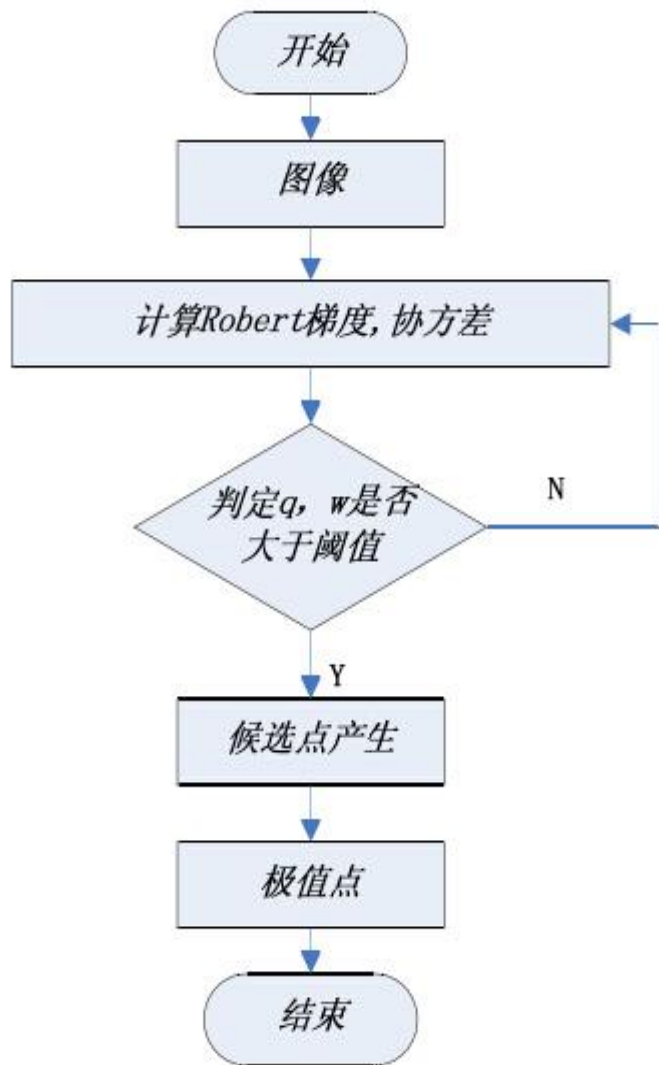
$$\left. \begin{aligned} T_q &= 0.5 \sim 0.75 \\ T_w &= \begin{cases} f\overline{w} & (f = 0.5 \sim 1.5) \\ c\omega_c & (c = 5) \end{cases} \end{aligned} \right\}$$

当 $q > T_q$ 同时 $w > T_w$ ，该像元为待选点

(5) 选取极值点

即在一个适当窗口中选择最大的待选点

1.3 Forstner算子



$$Q = N^{-1} = \begin{bmatrix} \sum g_u^2 & \sum g_u g_v \\ \sum g_v g_u & \sum g_v^2 \end{bmatrix}^{-1}$$

$$w = \frac{1}{\text{tr}Q} = \frac{\text{Det}N}{\text{tr}N}$$

$$q = \frac{4\text{Det}N}{(\text{tr}N)^2}$$

$$q > T_q \quad w > T_w$$

在一个适当窗口中选择最大的待选点

1.4 SIFT算子

□ 树冠

□ 波浪

□ 汽车

□ 飞机

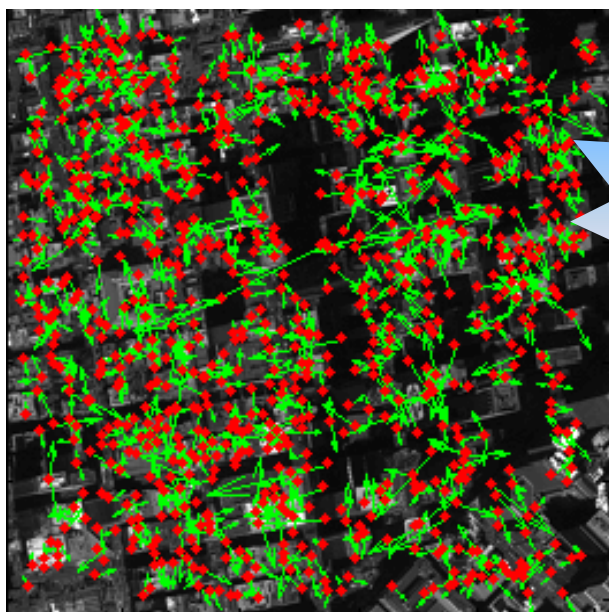
□ 舰船

□ 储油罐

□ 山丘

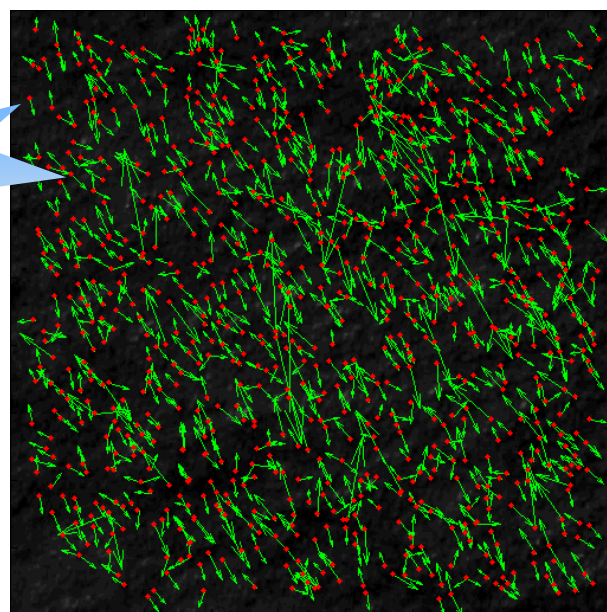
□ 海岛

□ ...



全色图像的城市区域

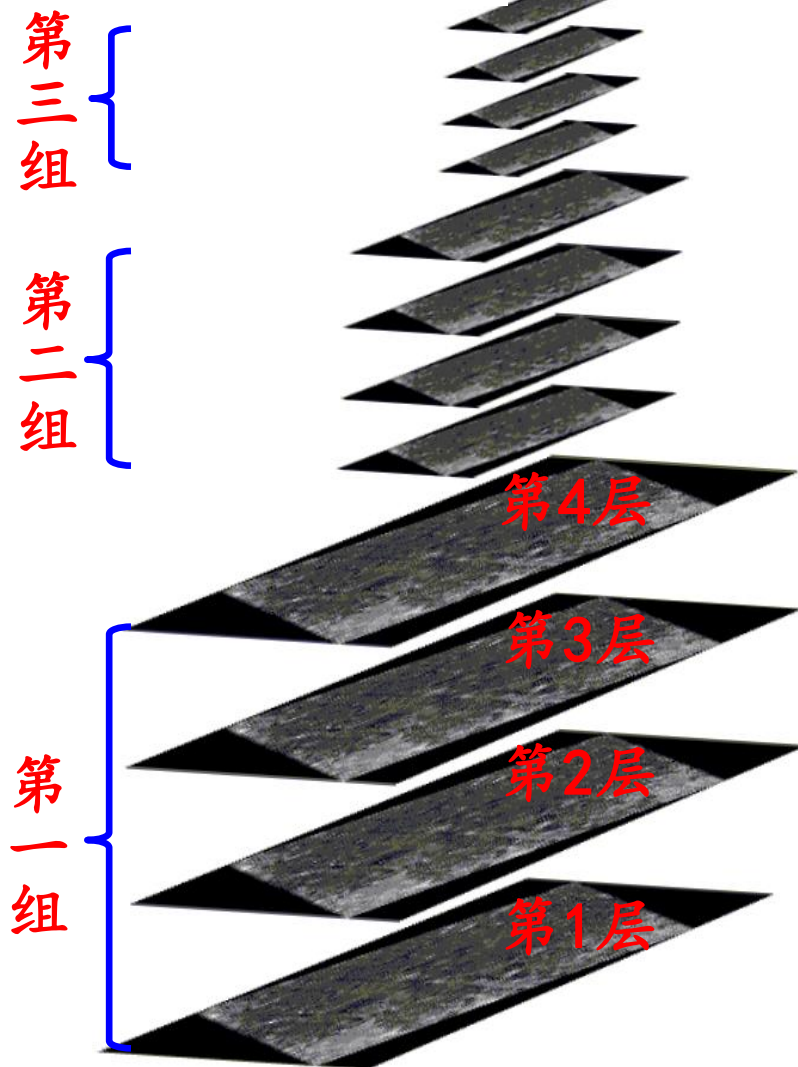
特征点
数量多



全色图像的海水区域

1.4 SIFT算子

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ = L(x, y, k\sigma) - L(x, y, \sigma)$$



基于SIFT特征的图像配准

□ 建立图像多尺度高斯差分金字塔

- 金字塔的组数多

- 每组含有的多层

□ 在每组金字塔内搜索极点

- 搜索范围—像素点的26邻域

□ 特征点的精确定位

- 特征点定位精度优于1个像素

□ 特征点的特征描述

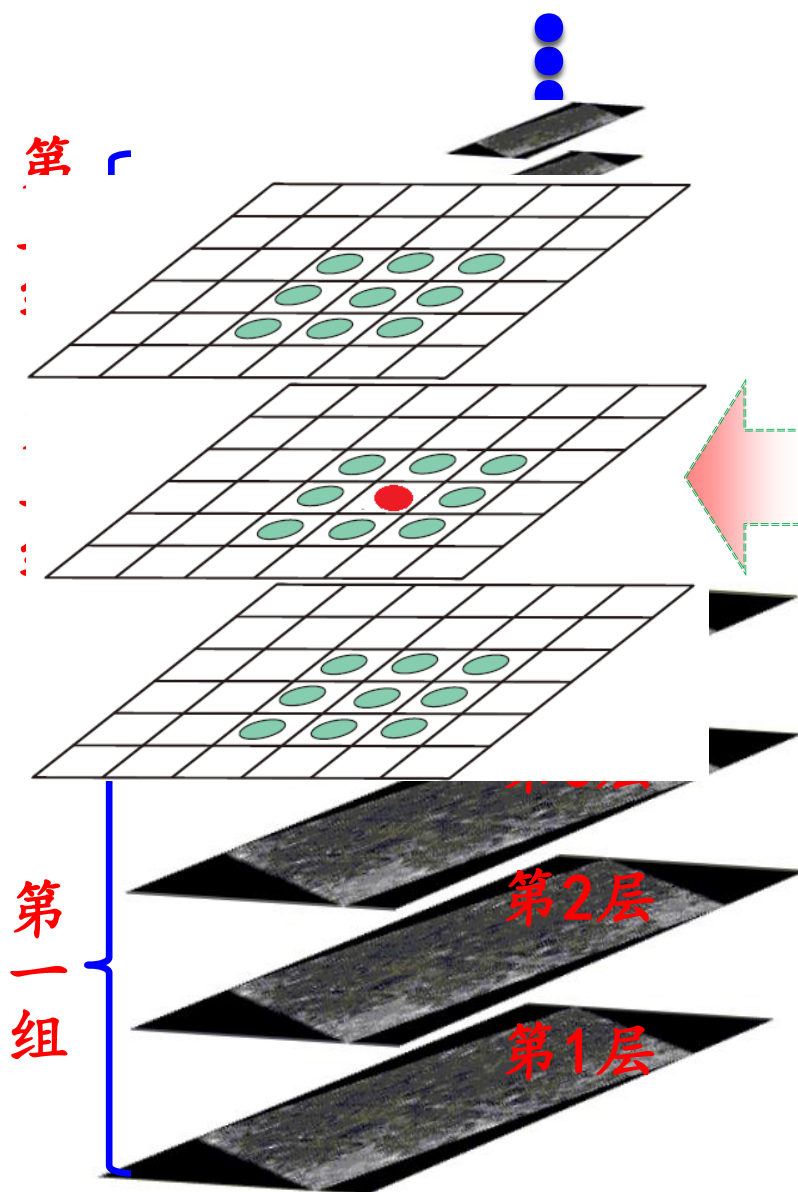
- 利用128维向量描述特征点

□ 特征点的匹配

- 特征点的数量多

- 计算128维向量之间的相似性

1.4 SIFT算子

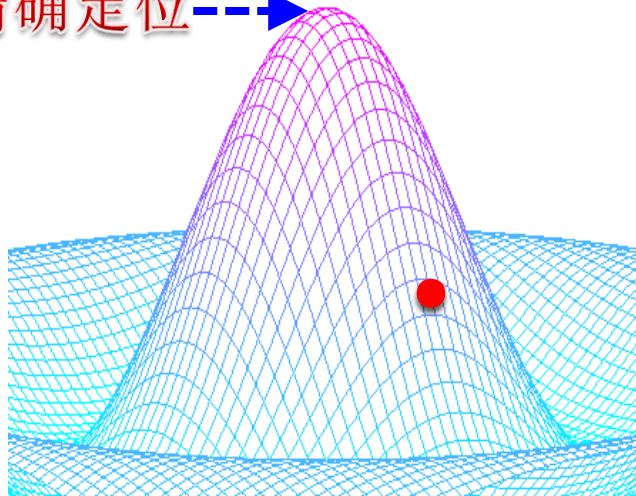


基于SIFT特征的图像配准

- 建立图像多尺度高斯差分金字塔
 - 金字塔的组数多
 - 每组含有的多层
- 在每组金字塔内搜索极点
 - 搜索范围—像素点的26邻域
- 特征点的精确定位
 - 特征点定位精度优于1个像素
- 特征点的特征描述
 - 利用128维向量描述特征点
- 特征点的匹配
 - 特征点的数量多
 - 计算128维向量之间的相似性

1.4 SIFT算子

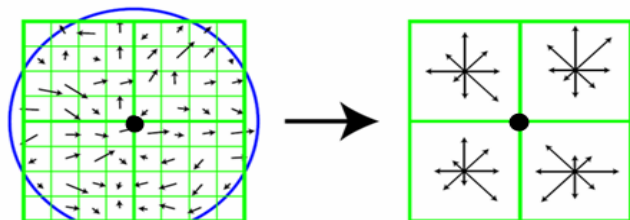
精确定位



基于SIFT特征的图像配准

- 建立图像多尺度高斯差分金字塔
 - 金字塔的组数多
 - 每组含有多层
- 在每组金字塔内搜索极点
 - 搜索范围—像素点的26邻域
- 特征点的精确定位
 - 特征点定位精度优于1个像素
- 特征点的特征描述
 - 利用128维向量描述特征点
- 特征点的匹配
 - 特征点的数量多
 - 计算128维向量之间的相似性

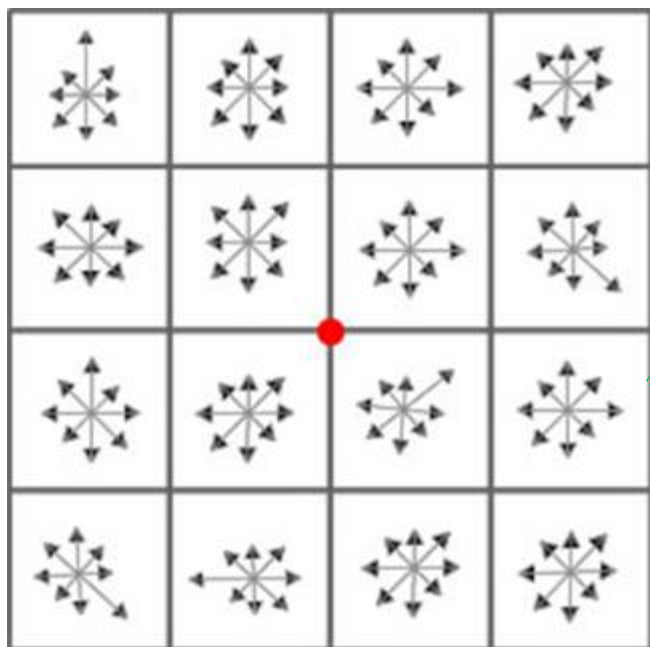
1.4 SIFT算子



领域梯度方向

关键点特征向量

8*8的邻域，**2*2**的种子，由关键点邻域梯度信息生成特征向量



4*4的种子，**4*4*8**(**8**个方向)=**128**维

基于SIFT特征的图像配准

- ❑ 建立图像多尺度高斯差分金字塔
 - ❑ 金字塔的组数多
 - ❑ 每组含有多层
- ❑ 在每组金字塔内搜索极点
 - ❑ 搜索范围—像素点的26邻域
- ❑ 特征点的精确定位
 - ❑ 特征点定位精度优于1个像素
- ❑ 特征点的特征描述
 - ❑ 利用128维向量描述特征点
- ❑ 特征点的匹配
 - ❑ 特征点的数量多
 - ❑ 计算128维向量之间的相似性

第四讲：遥感图像配准技术

一

特征点提取算法

二

特征线提取算法

三

特征的定位算法

四

最小二乘匹配

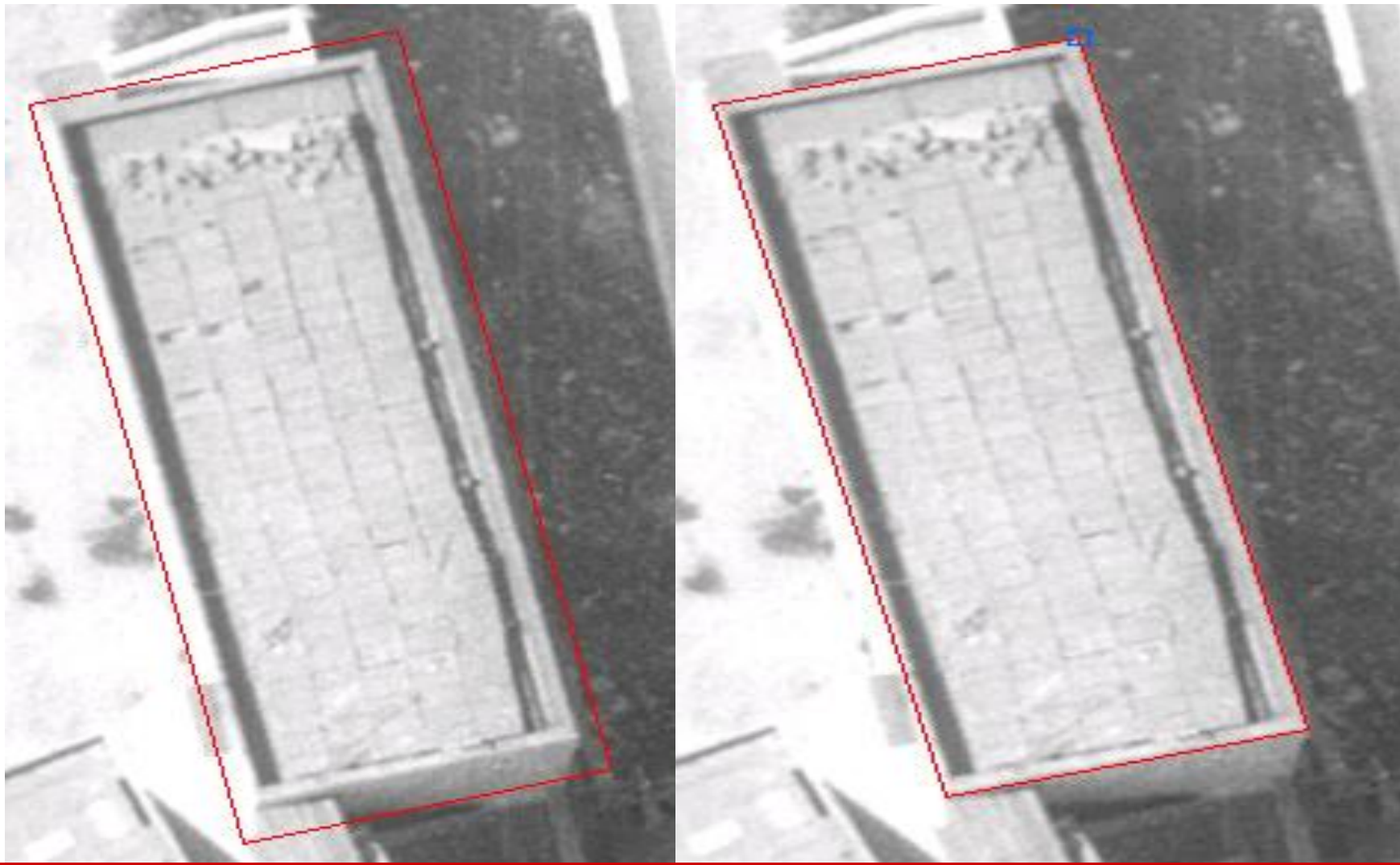
2.1 线特征提取算子

线特征是指影像的“边缘”与“线”

“边缘”可定义为影像局部区域特征不相同的那些区域间的分界线，而“线”则可以认为是具有很小宽度的其中间区域具有相同的影像特征边缘对

常用方法有差分算子、拉普拉斯算子、**LOG**算子

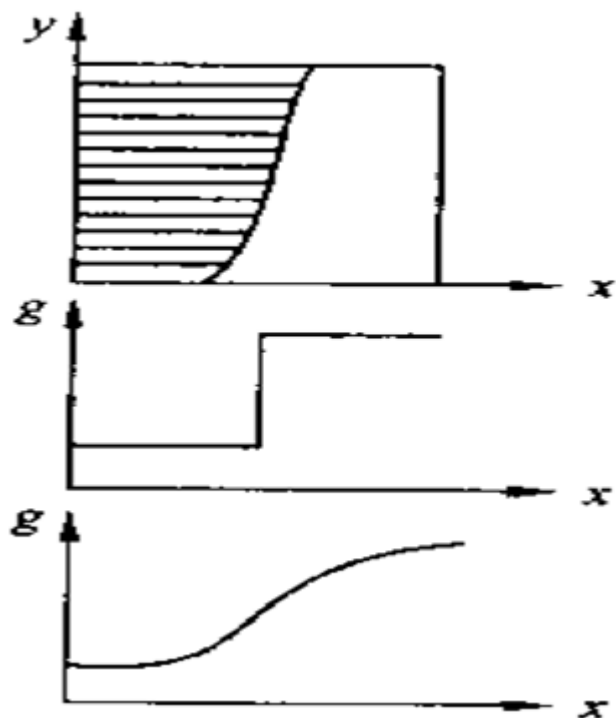
2.1 线特征提取算子



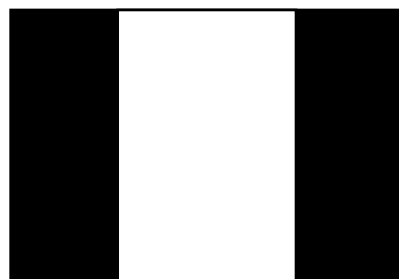
2.1 线特征提取算子



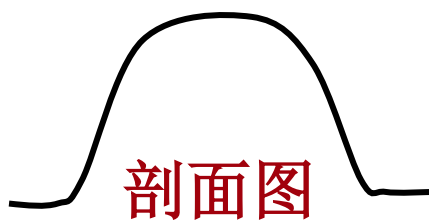
2.1 线特征提取算子



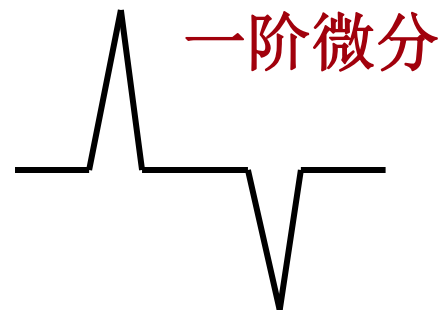
线的灰度特征



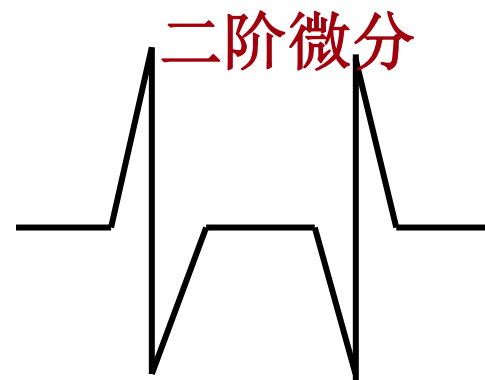
边界图像



剖面图



一阶微分

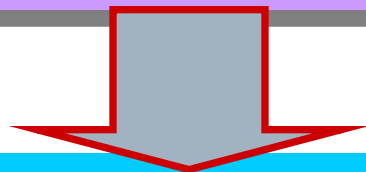


二阶微分

2.2 微分算子

1. 梯度算子

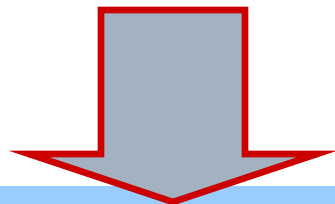
$$G[g(x, y)] = \begin{bmatrix} \frac{\partial g}{\partial x} \\ \frac{\partial g}{\partial y} \end{bmatrix}$$



$$G(x, y) = \text{mag}[G] = \left[\left(\frac{\partial g}{\partial x} \right)^2 + \left(\frac{\partial g}{\partial y} \right)^2 \right]^{\frac{1}{2}}$$

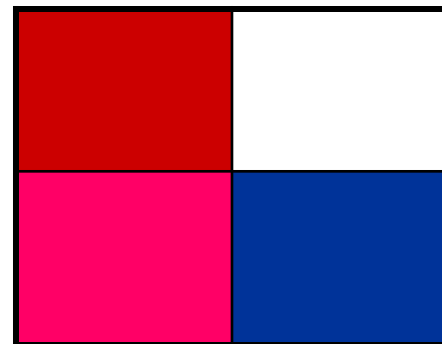
2.2 差分算子

$$G_{i,j} = \left[(g_{i,j} - g_{i+1,j})^2 + (g_{i,j} - g_{i,j+1})^2 \right]^{\frac{1}{2}}$$



近似

$$G_{i,j} = |g_{i,j} - g_{i+1,j}| + |g_{i,j} - g_{i,j+1}|$$



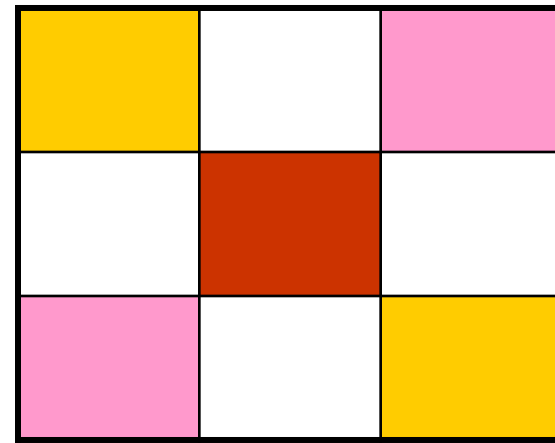
-1	1
----	---

-1
1

对于一给定的阈值**T**，当 **$G_{i,j} > T$** 时，则认为像素 **(*i*, *j*)** 是边缘上的点。

2.2 Robert梯度算子

$$G_r[g(x, y)] = \begin{bmatrix} \frac{\partial g}{\partial u} \\ \frac{\partial g}{\partial v} \end{bmatrix} = \begin{bmatrix} g_u \\ g_v \end{bmatrix}$$



$$G_r(x, y) = (g_u^2 + g_v^2)^{\frac{1}{2}}$$

-1	
	1

$$G_{i,j} = \left[(g_{i,j} - g_{i+1,j+1})^2 + (g_{i+1,j} - g_{i,j+1})^2 \right]^{\frac{1}{2}}$$

	-1
1	

2.2 方向差分算子

-1	-1	-1
2	2	2
-1	-1	-1

最佳响应水平线

-1	-1	2
-1	2	-1
2	-1	-1

最佳响应+45度线

-1	2	-1
-1	2	-1
-1	2	-1

最佳响应垂直线

2	-1	-1
-1	2	-1
-1	-1	2

最佳响应-45度线

每个掩模的系数之和为零，表明亮度不变区域中来自掩模的响应为零。

2.2 方向差分算子

1	1	1	1	1	1	1	1	1
5	5	5	5	5	5	5	5	5
1	1	1	1	1	1	1	1	1

-1	-1	-1
2	2	2
-1	-1	-1

$$R1 = -6 + 30 = 24$$

-1	-1	2
-1	2	-1
2	-1	-1

$$R2 = -14 + 14 = 0$$

-1	2	-1
-1	2	-1
-1	2	-1

$$R3 = -14 + 14 = 0$$

2	-1	-1
-1	2	-1
-1	-1	2

$$R4 = -14 + 14 = 0$$

2.2 方向差分算子

直线与边缘的方向

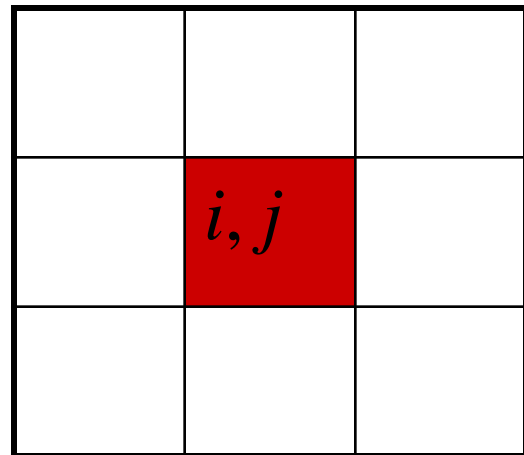
北	东北	东	东南
$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & 1 \\ -1 & -2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$
南	西南	西	西北
$\begin{bmatrix} -1 & -1 & -1 \\ 1 & -2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & -1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & -1 \\ 1 & -1 & -1 \end{bmatrix}$

2.2 Sobel算子

考察它上下、左右邻点灰度的加权差, 与之接近的邻点的权大:

$$S(i, j) = \left| g(i-1, j-1) + 2g(i-1, j) + g(i-1, j+1) - [g(i+1, j-1) + 2g(i+1, j) + g(i+1, j+1)] \right| \\ + \left| g(i-1, j-1) + 2g(i, j-1) + g(i+1, j-1) - [g(i-1, j+1) + 2g(i, j+1) + g(i+1, j+1)] \right|$$

$$\left. \begin{aligned} G_x &= \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \\ G_y &= \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \end{aligned} \right\}$$



2.2 Prewitt算子与Sobel算子

-1	0	1
-1	0	1
-1	0	1

-1	-1	-1
0	0	0
1	1	1

Prewitt算子

-1	0	1
-2	0	2
-1	0	1

-1	-2	1
-1	0	1
-1	2	1

Sobel 算子

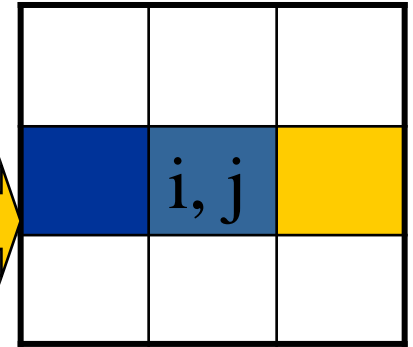
加大模板
抑制噪声

2.2 二阶差分算子

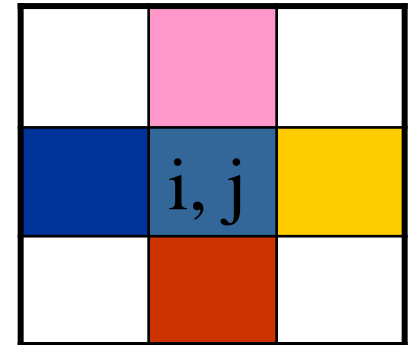
1. 方向二阶差分算子

$$g''_{ij} = (g_{i+1,j} - g_{i,j}) - (g_{i,j} - g_{i-1,j})$$

$$= -[g_{i-1,j} \quad g_{i,j} \quad g_{i+1,j}] \begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix} = -g_{ij} * [-1 \quad 2 \quad -1]$$



$$D = \begin{bmatrix} & & & \\ & -1 & 2 & -1 \\ & & & \end{bmatrix} + \begin{bmatrix} & -1 & \\ & 2 & \\ & -1 & \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$



$$D_1 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} + \begin{bmatrix} & & -1 \\ & 2 & \\ -1 & & \end{bmatrix} + \begin{bmatrix} -1 & & \\ & 2 & \\ & & -1 \end{bmatrix} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

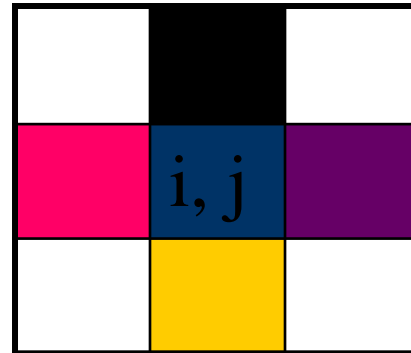
2.2 二阶差分算子

2. 拉普拉斯算子 (Laplace)

$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

$$\begin{aligned}\nabla^2 g_{ij} &= (g_{i+1,j} - g_{i,j}) - (g_{i,j} - g_{i-1,j}) + \\ &\quad (g_{i,j+1} - g_{i,j}) - (g_{i,j} - g_{i,j-1}) \\ &= g_{i+1,j} + g_{i-1,j} + g_{i,j+1} + g_{i,j-1} - 4g_{i,j}\end{aligned}$$

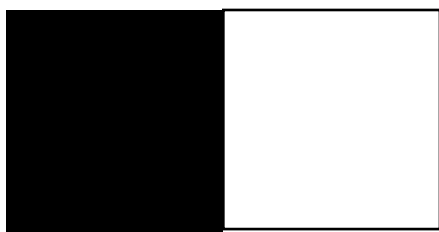
$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$



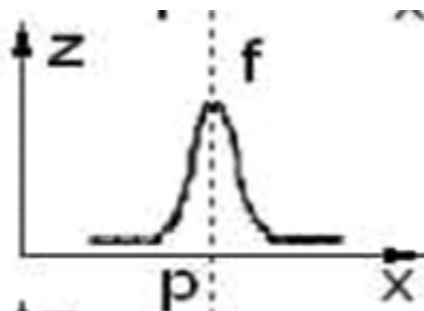
2.2 二阶差分算子

2. 拉普拉斯算子 (Laplace)

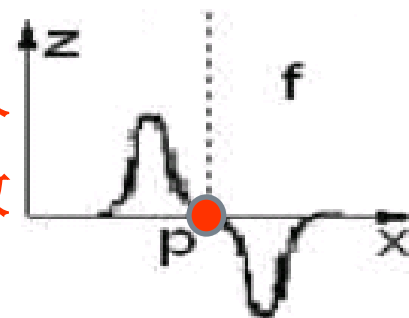
图像



一阶
导数

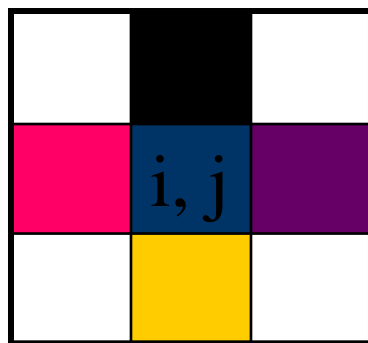


二阶
导数



准确确定边缘

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$



取其符号变化的点，即通过零的点为边缘点，因此通常也称其为零交叉 (zero-Crossing) 点

2.2 二阶差分算子

3. 高斯-拉普拉斯算子 (LOG)

首先用高斯函数进行低通滤波，然后利用拉普拉斯算子进行高通滤波并提取零交叉点

$$f(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

高斯函数

$$G(x, y) = f(x, y) * g(x, y)$$

低通滤波

$$G(x, y) = \nabla^2[f(x, y) * g(x, y)]$$

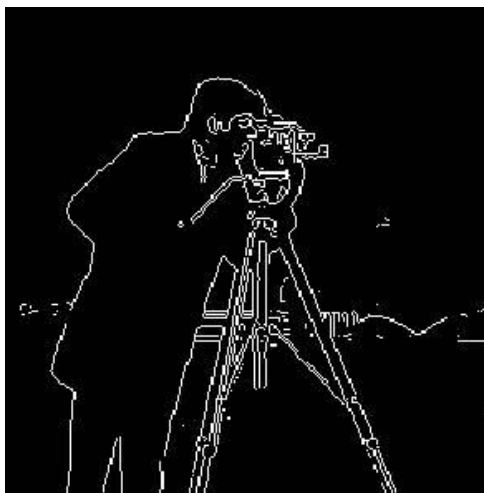
边缘提取

2.2 边缘检测算子比较结果

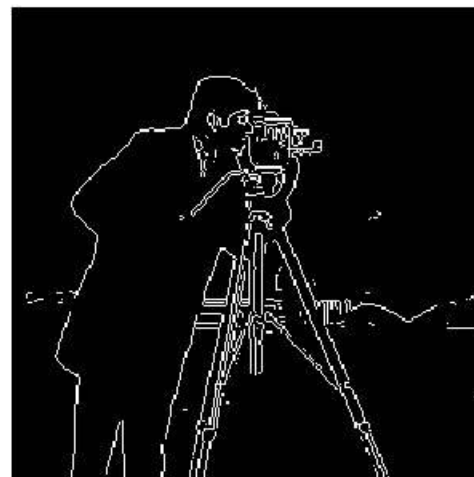
原始图像



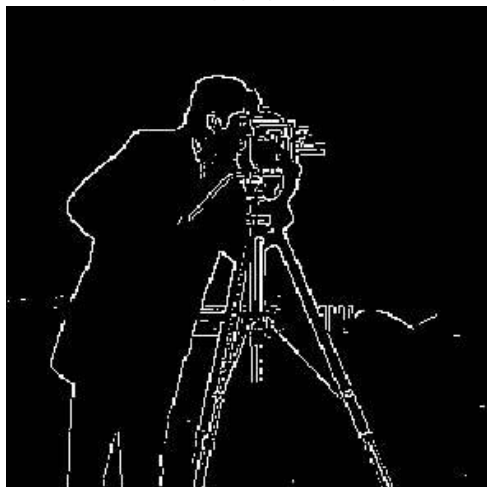
Sobel



Prewitt



Roberts



Laplacian of Gaussian



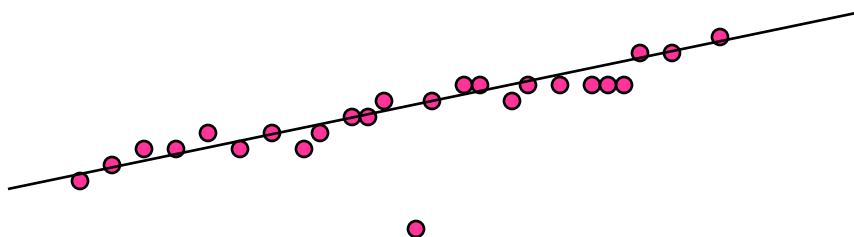
Canny



2.3 边缘提取

1. Hough变换

Hough变换问题的提出：在找出边界点集之后，需要连接，形成完整的边界图形描述

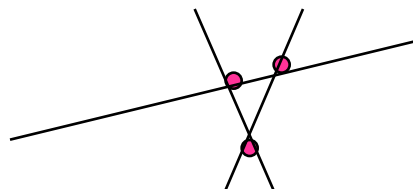


已检测出1条直线上的
若十个点，需要求它
们所在的直线。

■ 直接办法

- 先确定所有由任意2点决定的直线（需约 n^2 次运算以确定 $n(n-1)/2$ 条线），再找出接近具体直线的点的集合（需约 n^3 次运算以比较 n 个点中的每1个与 $n(n-1)/2$ 条直线中的每一条）

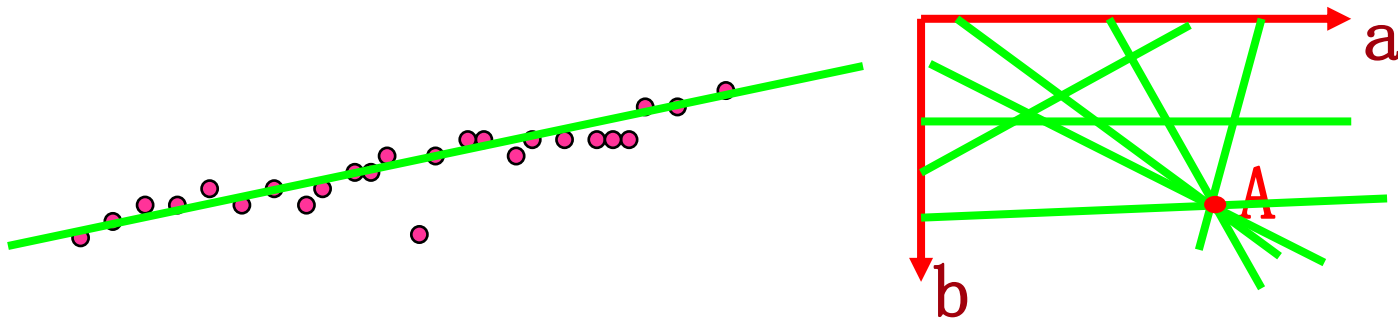
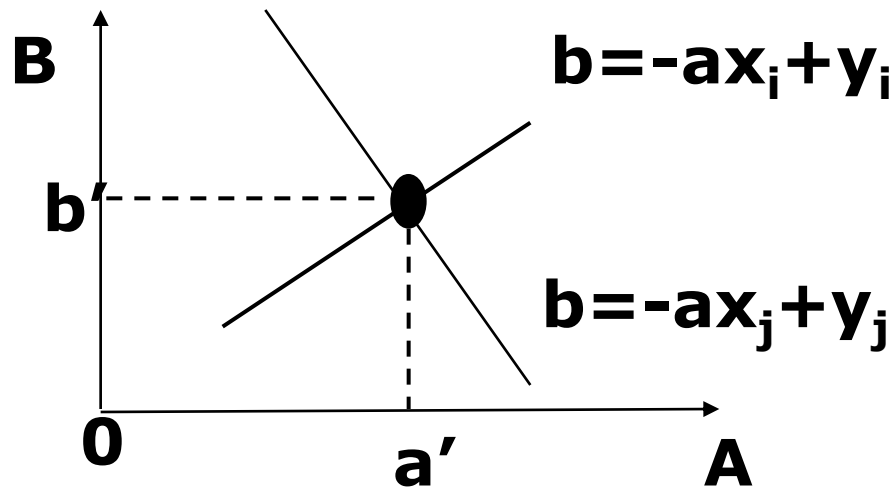
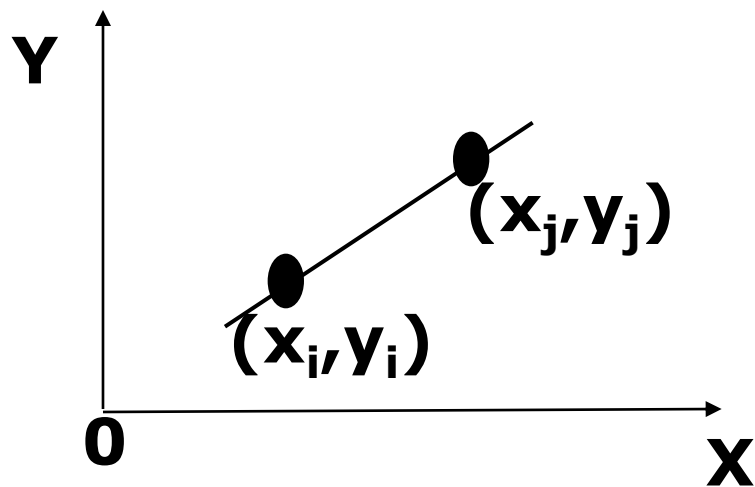
- 计算量大！



2.3 边缘提取

1. Hough变换

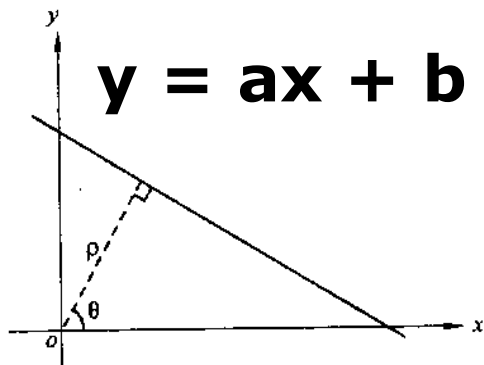
■ 点一线对偶性



2.3 边缘提取

1. Hough变换

图像空间



用于检测图像中直线、圆、抛物线、椭圆等

$$\rho = x \cos \theta + y \sin \theta$$

对于影像空间直线上任一点 (x, y) 变换将其映射到参数空间 (θ, ρ) 的一条正弦曲线上

Hough变换步骤:

- 对影像进行预处理提取特征点
- 将 (θ, ρ) 参数平面量化, 设置二维累计矩阵 $H(\theta_i, \rho_j)$ 作为交点累加器, 找出相交线段最多的参数空间的点, 确定直线线段

第四讲： 遥感图像配准技术

一

特征点提取算法

二

特征线提取算法

三

特征的定位算法

四

最小二乘匹配

3 定位算子

数字影像上**明显目标**主要指：地面上明显地物在影像上的反映，或者是数字影像自身的明显标志，例如道路、河流的交叉口、田角、房角、建筑物上的明显标志、影像四角上的框标、地面人工标志点等

3.1 Wong-Trinder 圆点定位算子

利用二值图像重心对圆点进行定位

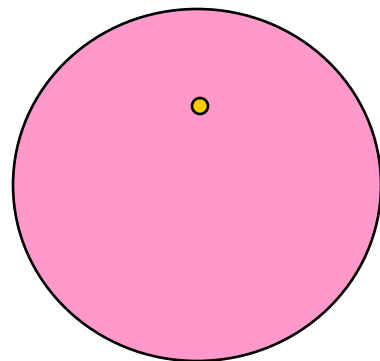
- 利用阈值 $T = (\text{最小灰度值} + \text{平均灰度值}) / 2$ 将窗口中的影像二值化 .
- 计算目标重心坐标 (x, y) 与圆度 r .

$$m_{pq} = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} i^p j^q g_{ij} \quad (p, q = 0, 1, 2 \dots)$$

$p+q$ 阶原点矩与中心矩

$$M_{pq} = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (i-x)^p (j-y)^q g_{ij} \quad (p, q = 0, 1, 2 \dots)$$

$$\left. \begin{aligned} x &= m_{10} / m_{00} \\ y &= m_{01} / m_{00} \\ \gamma &= M_x' / M_y' \\ M_x' &= \frac{M_{20} + M_{02}}{2} + \sqrt{\left(\frac{M_{20} - M_{02}}{2}\right)^2 + M_{11}^2} \\ M_y' &= \frac{M_{20} + M_{02}}{2} - \sqrt{\left(\frac{M_{20} - M_{02}}{2}\right)^2 + M_{11}^2} \end{aligned} \right\}$$



当 r 小于阈值时，目标不是圆；否则圆心为 (x, y)

3.1 Trinder 改进算子

算子受二值化影响，误差可达**0.5**像素。

$$\left. \begin{aligned} x &= \frac{1}{M} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} i g_{ij} W_{ij} \\ y &= \frac{1}{M} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} j g_{ij} W_{ij} \end{aligned} \right\}$$

$$M = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} g_{ij} W_{ij}$$

原始灰度

定位精度可达0.01像素，这种算法只对圆点定位

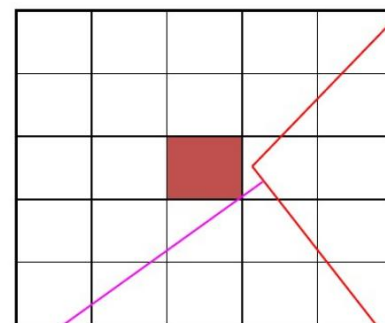
3.3 Forstner定位算子

Forstner定位算子是摄影测量界著名定位算子

- 最佳窗口由Forstner特征提取算子确定
- 以原点到窗口内边缘直线的距离为观测值，梯度模之平方为权，在点 (x, y) 处可列误差方程

$$\left. \begin{aligned} v &= x_0 \cos \theta + y_0 \sin \theta - (x \cos \theta + y \sin \theta) \\ \omega(x, y) &= |\nabla g|^2 = g_x^2 + g_y^2 \end{aligned} \right\}$$

在最佳窗口内加权重心化，由上述误差方程通过最小二乘法得到角点坐标，即窗口内像元的加权重心



第四讲：遥感图像配准技术

一

特征点提取算法

二

特征线提取算法

三

特征的定位算法

四

最小二乘匹配

4 最小二乘影像匹配

德国Ackermann教授提出了一种新的影像匹配方法——最小二乘影像匹配
(Least Squares Image Matching)

影像匹配可以达到 $1/10$ 甚至 $1/100$ 像素的高精度

4 最小二乘影像匹配

给定模板图像 $f(x,y)$ ，对应点为 (x_f, y_f)

测量图像的匹配窗口灰度 $g(x,y)$ ，对应点为 (x_g, y_g)

仿射变换
$$\begin{cases} x_g = a_0 + a_1 x_f + a_2 y_f \\ y_g = b_0 + b_1 x_f + b_2 y_f \end{cases} \quad \mathbf{AF} = \mathbf{G}$$

求解变换参数 $(a_0, a_1, a_2, b_0, b_1, b_2)$

最小二乘是一个迭代过程：

- 第一步的粗提取结果作为变换参数的迭代初值，代入矩阵方程求变形参数；
- 利用改正后的参数对测量图像重采样，计算模板与匹配子图的相关系数。若大于预定阈值，迭代结束。

4 最小二乘影像匹配

最小二乘影像匹配中可以非常灵活地引入各种已知参数和条件，从而可以进行整体平差

解决“单点”的影像匹配问题，以求其“视差”；也可以直接求解其空间坐标

同时解决“多点”影像匹配或“多片”影像匹配

引入“粗差检测”，从而大大地提高影像匹配可靠性

4.1 最小二乘影像匹配原理

“灰度差的平方和最小”

$$\sum v^2 = \min$$

仅仅认为影像灰度只存在偶然误差

$$v = g_1(x, y) - g_2(x, y)$$

4.1 最小二乘影像匹配原理

按 $\sum vv = \min$ 原则进行影像匹配的数字模型。若在此系统中引入系统变形的参数，按 $\sum vv = \min$ 的原则，求解变形参数，就构成了最小二乘影像匹配系统。

灵活, 可靠和高精度是优点, 缺点是, 如当初始值不太准时, 系统的收敛性等问题有待解决。

4.2 影响匹配的因素

辐射畸变

- ♠ 照明及被摄影物体辐射面的方向
- ♠ 大气与摄影机物镜所产生的衰减
- ♠ 摄影处理条件的差异，以及影像数字化过程中所产生的误差等等

4.2 影响匹配的因素

几何畸变

- 摄影机方位不同所产生的影像的透视畸变
- 由于地形坡度所产生的影像畸变等竖直航空摄影的情况下，地形高差则是几何畸变的主要因素。

4.3 最小二乘模型—线性辐射畸变

在影像匹配中引入这些变形参数，同时按最小二乘的原则，求解这些参数，就是最小二乘影像匹配的基本思想。

考虑辐射线性畸变的最小二乘匹配—相关系数

$$g_1(x, y) + n_1 = h_0 + h_1 g_2(x, y) + g_2 + n_2$$

误差方程：

$$v = h_0 + h_1 g_2 - (g_1 - g_2)$$

h_0, h_1 是辐射畸变参数

4.3 最小二乘模型—线性辐射畸变

$$v = h_0 + h_1 g_2 - (g_1 - g_2)$$

按 $\sum vv = \min$ 的原理, 可得法方程式

$$nh_0 + (\sum g_2)h_1 = \sum g_1 - \sum g_2$$

$$(\sum g_2)h_0 + (\sum g_2^2)h_1 = \sum g_1 g_2 - \sum g_2^2$$

$$h_1 = \frac{\sum g_1 \sum g_2 - n \sum g_2 g_1}{(\sum g_2)^2 - n \sum g_2^2} - 1$$

$$h_0 = \frac{1}{n} (\sum g_1 - \sum g_2 - (\sum g_2)h_1)$$

4.3 最小二乘模型—线性辐射畸变

消除了两个灰度分布的系统的辐射畸变后，其残余的灰度差的平方和为

$$\begin{aligned}\sum v v &= \sum \left(g_2 \cdot \frac{\sum g_2 g_1}{\sum g_2} - g_1 \right)^2 \\ &= \left(\frac{\sum g_2 g_1}{\sum g_2} \right)^2 \sum g_2^2 - 2 \frac{\sum g_2 g_1}{\sum g_2} \sum g_2 g_1 + \sum g_1^2 \\ \sum v v &= \sum g_1^2 - \frac{(\sum g_2 g_1)^2}{\sum g_2^2}\end{aligned}$$

4.3 最小二乘模型—线性辐射畸变

相关系数



$$\rho^2 = \frac{(\sum g_2 g_1)^2}{\sum g_1^2 \sum g_2^2}$$

相关系数与 $\sum vv$ 的关系

$$\sum vv = \sum g_1^2 (1 - \rho^2)$$

$$\sum vv / \sum g_1^2 = 1 - \rho^2$$

$\sum \mathbf{v}\mathbf{v}$ 是噪声的功率
 $\sum \mathbf{g}_1^2$ 为信号的功率

4.3 最小二乘模型—线性辐射畸变

信噪比



$$(SNR)^2 = \frac{\sum g_1^2}{\sum v v}$$

相关系数与信噪比之间的关系

$$(SNR)^2 = \frac{1}{(1 - \rho^2)}$$

以“相关系数最大”作为影像匹配搜索同名点的准则，其实质是搜索“信噪比为最大”的灰度序列

4.3 最小二乘模型—直接解算相对位移

影像匹配的主要目的是确定影像相对移位，传统的算法采用目标区相对于搜索区不断地移动一个整像素，搜索最大相关系数的影像区中心作为同名像点

在最小二乘影像匹配算法中，可引入几何变形参数，直接解算影像移位，这是此算法的特点

4.3 最小二乘模型——直接解算相对位移

假设两个一维灰度函数 $g_1(x)$, $g_2(x)$, 除随机噪声外, $g_2(x)$ 相对于 $g_1(x)$ 只存在零次几何变形——移位量 Δx 。

$$g_1(x) + n_1(x) = g_2(x + \Delta x) + n_2(x)$$

误差方程式

$$v(x) = g_2(x + \Delta x) - g_1(x)$$

4.3 最小二乘模型—直接解算相对位移

$$v(x) = g_2(x + \Delta x) - g_1(x)$$

为求解相对位移量 Δx ，需对上式进行线性化

$$v(x) = g'_2(x) \cdot \Delta x - [g_1(x) - g_2(x)]$$

对离散的数字影像，灰度函数的导数 $g'_2(x)$ 可由差分代替

$$\dot{g}_2(x) = \frac{g_2(x + \Delta) - g_2(x - \Delta)}{2\Delta}$$

误差方程式可写为

$$v = \dot{g}_2 \cdot \Delta x - \Delta g$$

4.3 最小二乘模型—直接解算相对位移

误差方程式可写为

$$v = \dot{g}_2 \cdot \Delta x - \Delta g$$

解得影像的相对移位

$$\Delta x = \sum \dot{g}_2 \cdot \Delta g / \sum \dot{g}_2^2$$

最小二影像匹配是非线性系统，必须进行迭代。迭代过程收敛的速度取决于初值。

4.3 最小二乘模型—直接解算相对位移

补充

考虑到图像子块间的相对旋转与缩放很小，因此利用相位相关法计算二者之间的相对偏移：

□ 计算全色和多光谱图像子块的功率谱 E

$$E = \frac{F_1(u, v)F_2^*(u, v)}{|F_1(u, v)F_2^*(u, v)|}$$

□ 计算功率谱的逆傅立叶变换 $F^{-1}(E)$

□ 搜索 $F^{-1}(E)$ 峰值点的坐标，该坐标代表了二者的相对偏移