

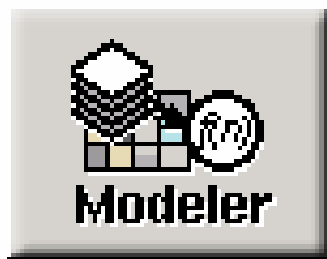
ERDAS建模

ESRI 中国北京

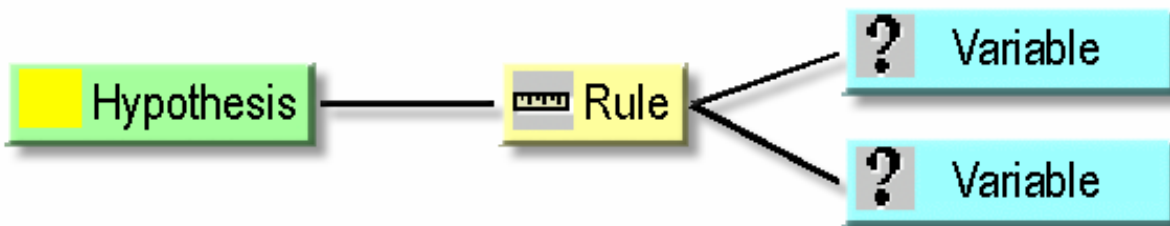
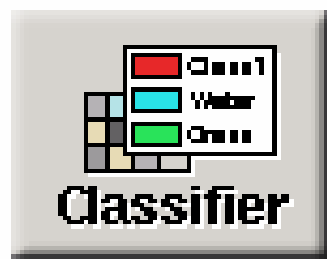
工程师：刘卓颖

Modeling and Expert Systems

- ERDAS IMAGE 空间建模工具



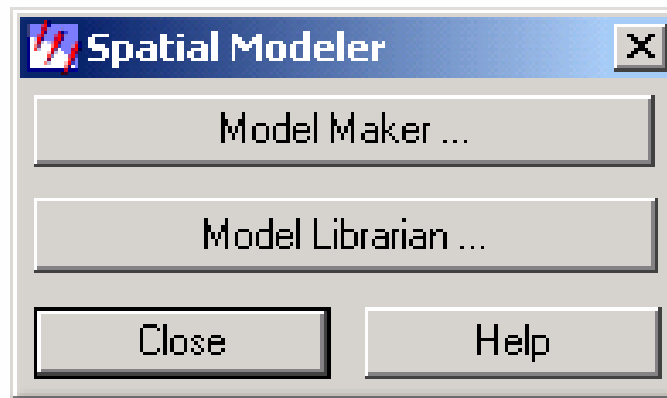
- ERDAS IMAGE 专家分类器



高级空间建模工具

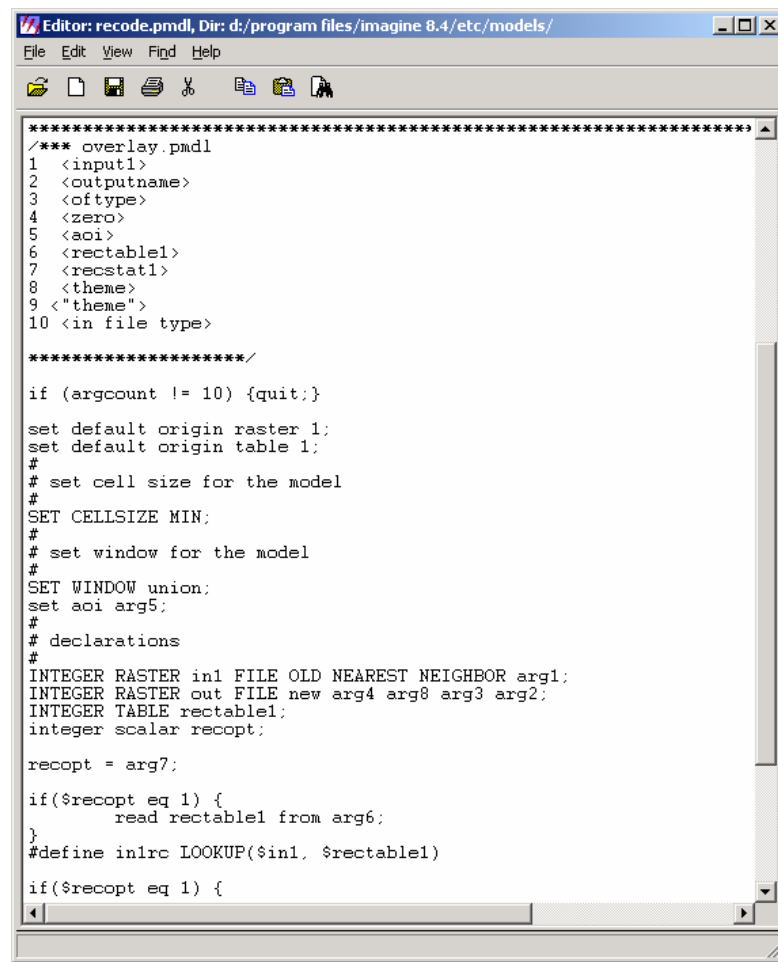


- 使用 **Spatial Modeler Language (SML)**
- 提供一个称为**Model Maker**的图形建模工具



Spatial Modeler Language (SML)

- 用于Model Maker 执行操作
- 用于写你自己的script模型
- 在模型执行的时候，会创建一个临时的SML script



```
Editor: recode.pmdl, Dir: d:/program files/imagine 8.4/etc/models/
File Edit View Find Help

**** overlay.pmdl ****
1 <input1>
2 <outputname>
3 <oftype>
4 <zero>
5 <aoi>
6 <rectable1>
7 <recstat1>
8 <theme>
9 <"theme">
10 <in file type>

****/

if (argcount != 10) {quit;}

set default origin raster 1;
set default origin table 1;
#
# set cell size for the model
#
SET CELLSIZE MIN;
#
# set window for the model
#
SET WINDOW union;
set aoi arg5;
#
# declarations
#
INTEGER RASTER in1 FILE OLD NEAREST NEIGHBOR arg1;
INTEGER RASTER out FILE new arg4 arg8 arg3 arg2;
INTEGER TABLE rectable1;
integer scalar recopt;

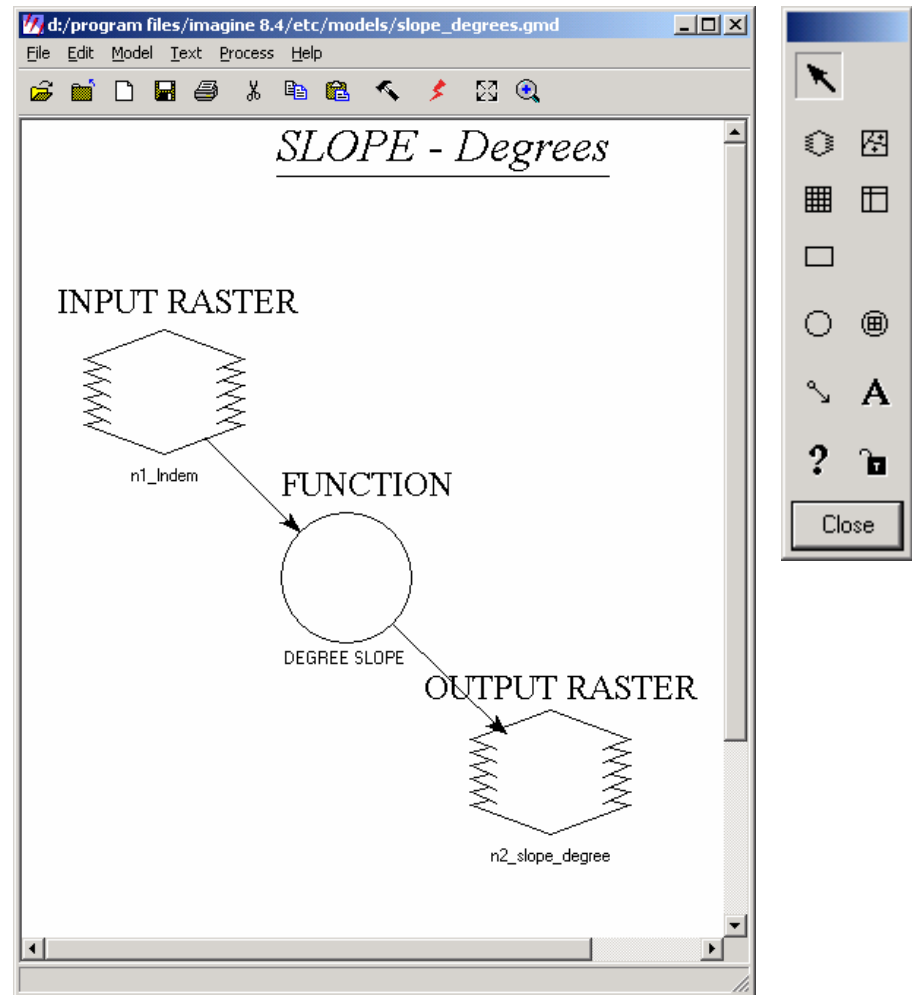
recopt = arg7;

if($recopt eq 1) {
    read rectable1 from arg6;
}
#define in1rc LOOKUP($in1, $rectable1)

if($recopt eq 1) {
```

Model Maker

- 用户可以使用模板工具设计模型



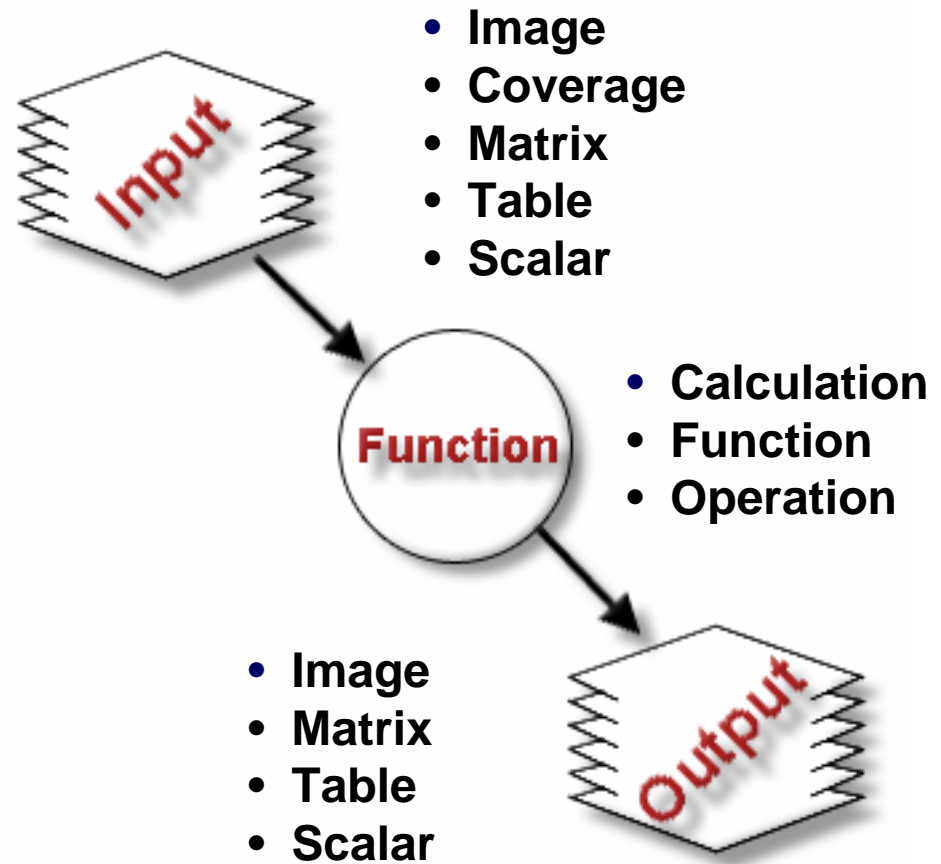
Inputs, Functions, Outputs

- 模型定义流程图:

- 输入

- 函数

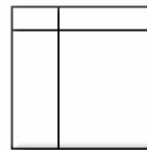
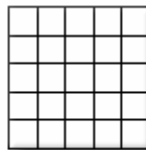
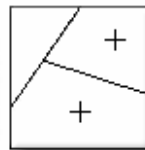
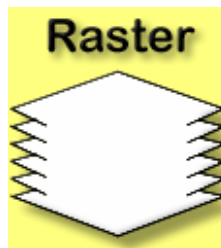
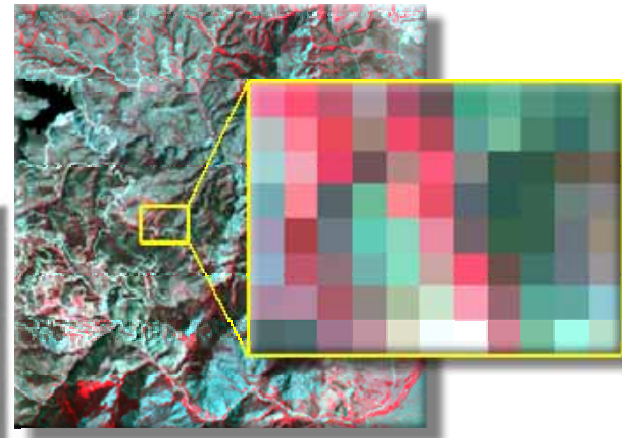
- 输出



Object Types

• 栅格

- 栅格数据层或是层集
- 包含操作影像文件中的数据



Object Types

• 矢量

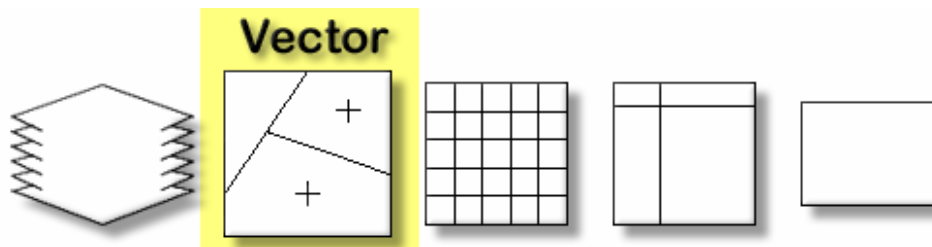
- 一个Arc/Info Coverage 或者是一个注记层



Attributes for e:/temp/vector_fulton

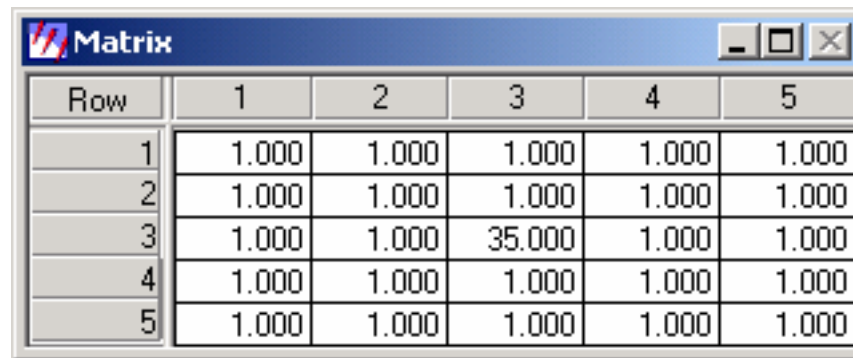
File Edit View Help

Record	RPOLY	FULTON	FULTON ID	FEATURE	UR	FEATURE TY
1	2	1.	477162	Road Segment		Narrow Road
2	2	2.	474209	Road Segment		Narrow Road
3	2	3.	477165	Road Segment		Narrow Road
4	2	4.	477168	Road Segment		Narrow Road
5	2	5.	477164	Road Segment		Wide Road
6	2	6.	477169	Road Segment		Narrow Road
7	2	7.	477140	Road Segment		Wide Road
8	2	8.	477171	Road Segment		Narrow Road
9	2	9.	477119	Road Segment		Narrow Road
10	2	10.	477174	Road Segment		Narrow Road
11	2	11.	477172	Road Segment		Wide Road

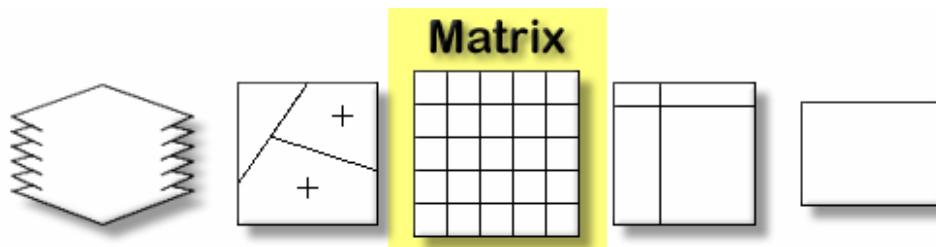


Object Types

- 矩阵
 - 排列在2D队列 中的一些列数据
 - 有固定的行列数



Row	1	2	3	4	5
1	1.000	1.000	1.000	1.000	1.000
2	1.000	1.000	1.000	1.000	1.000
3	1.000	1.000	35.000	1.000	1.000
4	1.000	1.000	1.000	1.000	1.000
5	1.000	1.000	1.000	1.000	1.000



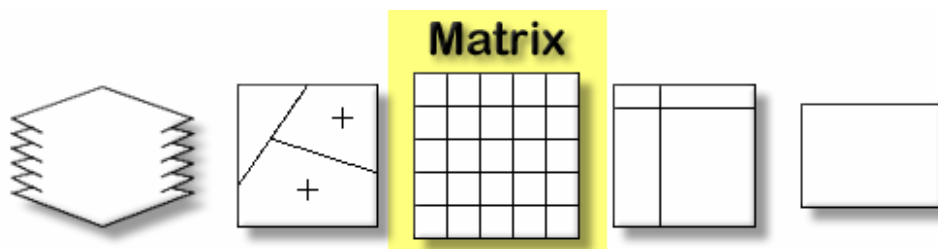
Object Types

- 矩阵可以存储:
 - 卷积核
 - 在临近函数中使用的临近定义
 - 协方差矩阵
 - 特征向量矩阵
 - 线性组合系数矩阵

-1.000	-1.000	-1.000	-1.000	-1.000
-1.000	-2.000	-2.000	-2.000	-1.000
-1.000	-2.000	60.000	-2.000	-1.000
-1.000	-2.000	-2.000	-2.000	-1.000
-1.000	-1.000	-1.000	-1.000	-1.000

0.257	-0.126	-0.213	-0.126	0.257
-0.126	-0.627	0.352	-0.627	-0.126
-0.213	0.352	2.928	0.352	-0.213
-0.126	-0.627	0.352	-0.627	-0.126
0.257	-0.126	-0.213	-0.126	0.257

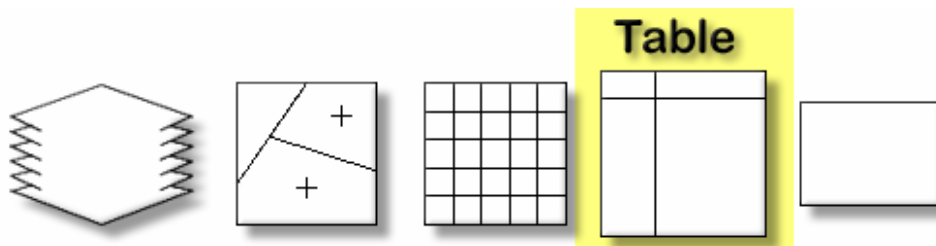
-1	-1	-1
-1	-1	-1
-1	-1	-1
-1	-1	-1
-1	56	-1
-1	-1	-1
-1	-1	-1
-1	-1	-1
-1	-1	-1



Object Types

- 表
 - 一系列数值或字符串
 - 一行或者一个固定的行数

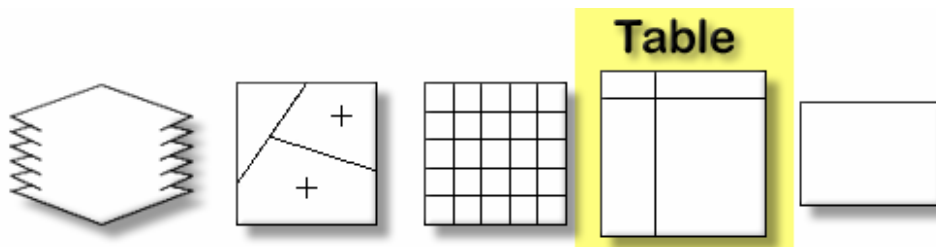
Row	LENGTH
0	253.729
1	79.199
2	21.886
3	21.886
4	70.204
5	81.045
6	31.782
7	34.002
8	41.144
9	41.144
10	109.579
11	53.662
12	142.379
13	52.014
14	63.622



Object Types

- 表可以存储:
 - 属性表中的列
 - 栅格层附带的一栏数值
- 表中的信息可以是:
 - 属性
 - 计算结果(如直方图)
 - 用户定义的信息

Row	Histogram
0	136.000
1	2.000
2	2.000
3	8.000
4	37.000
5	95.000
6	143.000
7	253.000
8	305.000
9	362.000
10	443.000
11	496.000
12	580.000
13	822.000
14	1195.000
15	1764.000



Object Types

- 标量
 - 一个单一常数值

Value: 1.00000

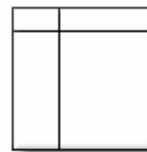
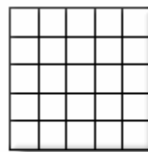
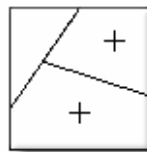
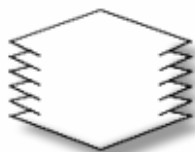
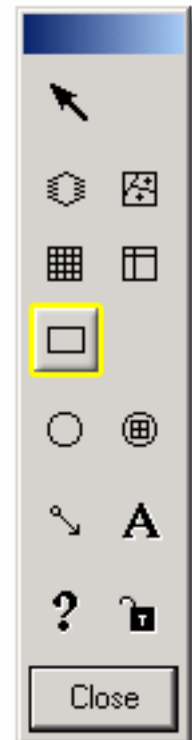
Value: 1.00000 Type: Binary

☐ Read From: ☐ Prompt

☐ Write To: ☐ Prompt

☐ Show Scalar in Session Log

OK Cancel Help



Scalar



Data Types

- 二进制

- 0 (假) 或 1 (真)

1
0

- 整数

- 整数值从 -2,147,483,648 到 2,147,483,647 (32位有符号型整数)

5688946
-996658

- 浮点型

- 一个数值:
 - 包含一个小数点
 - 一个科学计数法
 - 在32位有符号型整数之外

33.220
68.240

- 字符串

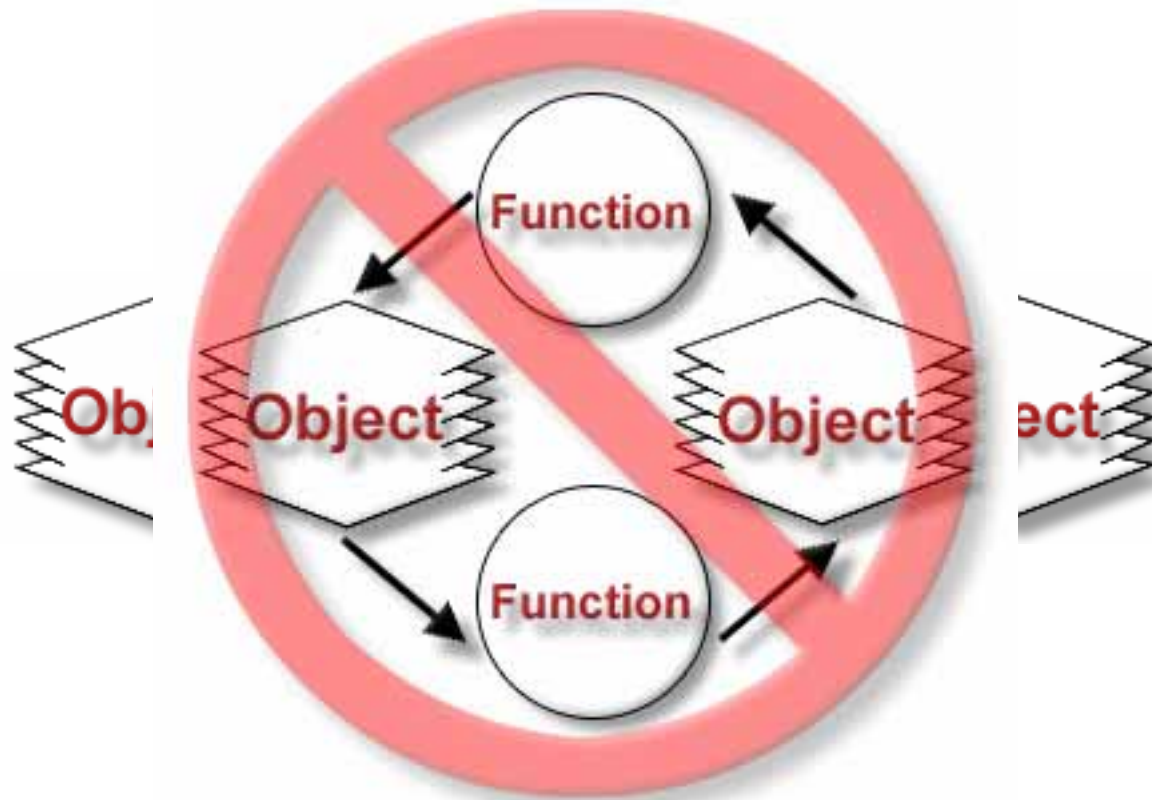
- 一个字符串 (只能在表对象中使用)

"water"
"forest"

Connections

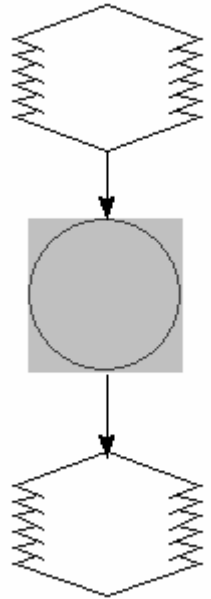


- 连接对象到函数
- 连接函数到对象



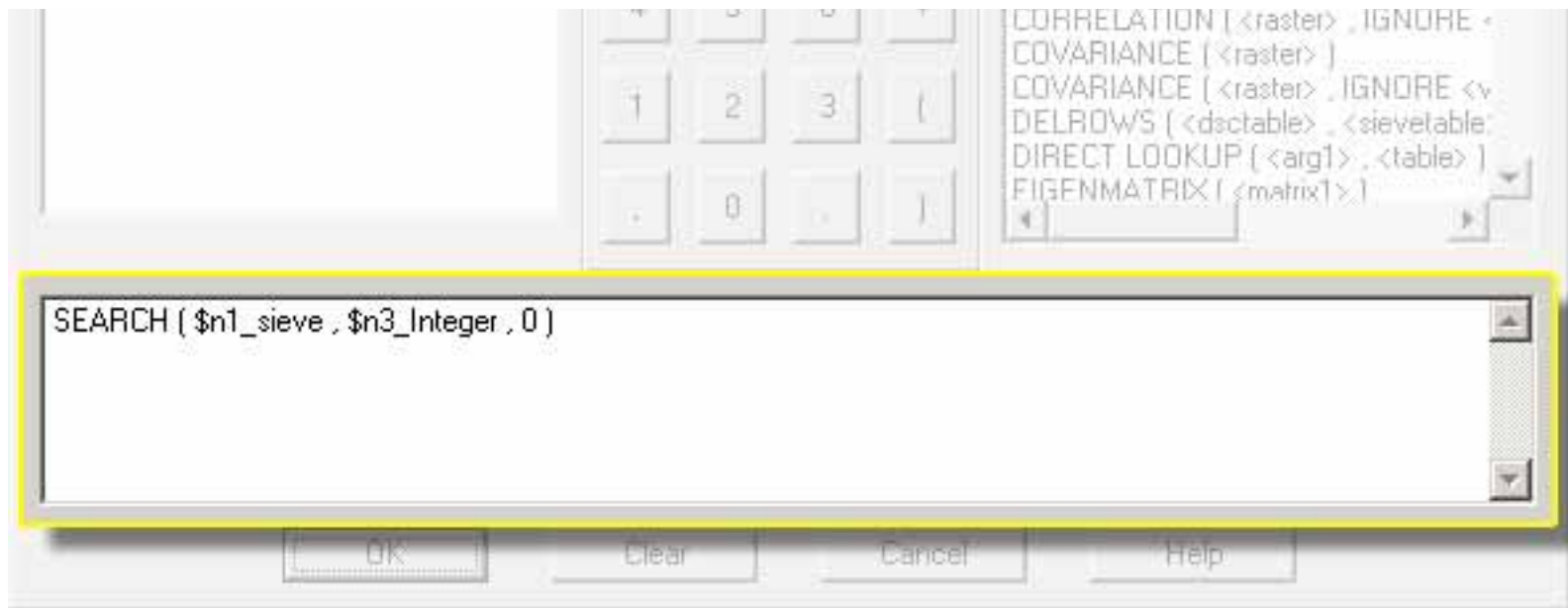
Functions

- 函数定义对对象执行的操作。
- 你的输出是用表达式（例如" $a + b + c$ "）定义的。
- 创建函数定义的主要步骤是：
 - 定义输入对象和对输入对象到函数进行连接
 - 在函数图上鼠标左键双击，用函数定义对话框来编写函数定义。

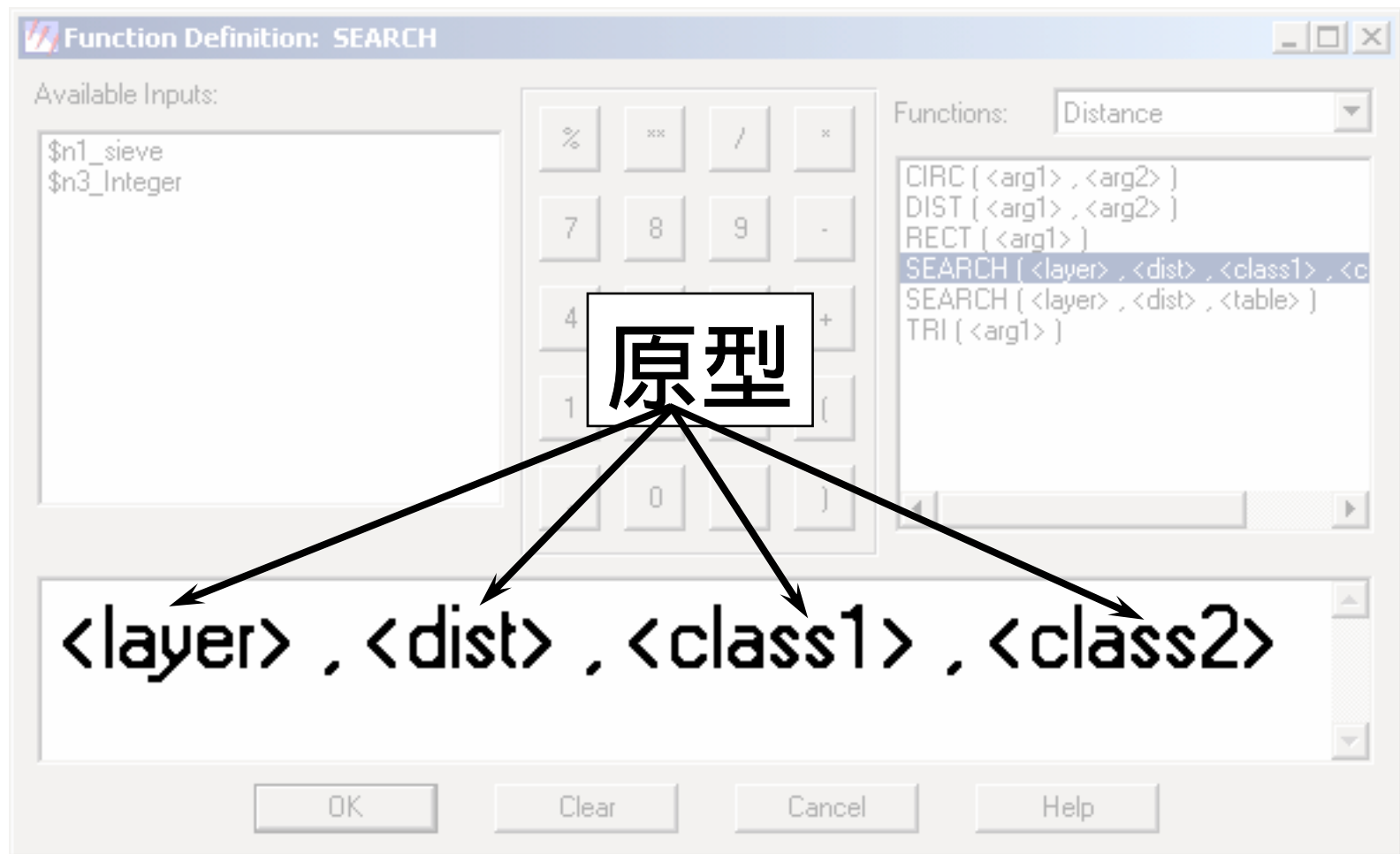


The Function Definition Dialog

- 文本区
- 用鼠标左键点击该区域任何地方，就可以在光标位置进行输入了。

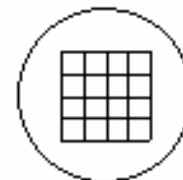


Prototypes < >



The Criteria Function Dialog

- 在该表格中创建复杂的条件语言



Criteria

Available Layers: <none defined>

Criteria Table

Row	Output		
1	1		
2	2		
3	3		

Rows: 3 Columns: 2

Satisfy: ☒ All Criteria ☐ Any Criteria Otherwise: 0

Descriptor: [cell value]

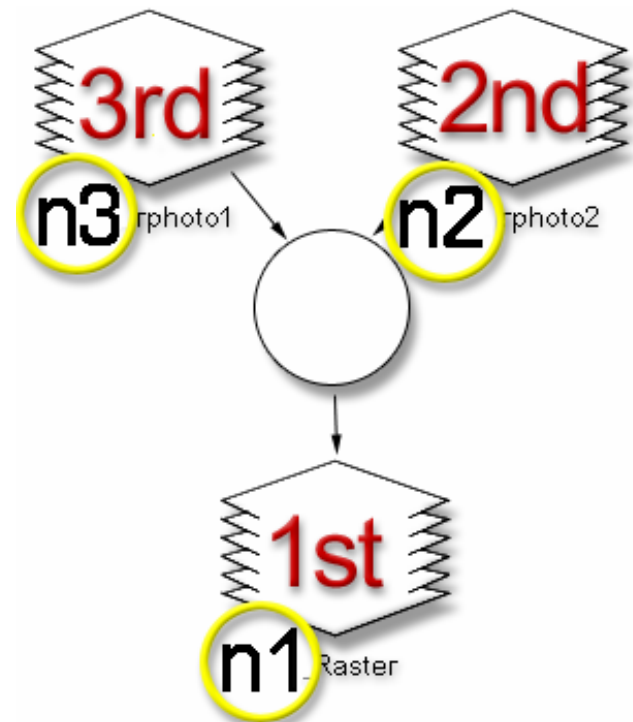
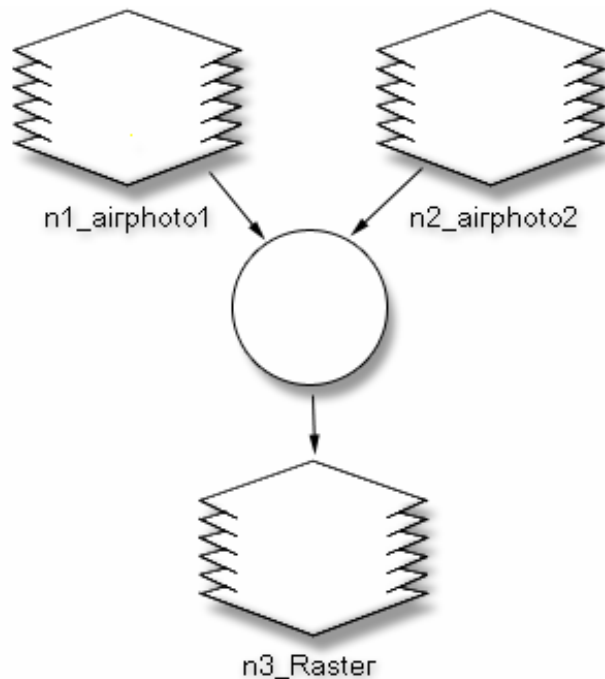
OK Add Column Cancel Help

Function Categories

- 分析
- 算术
- 位运算
- 布尔
- 颜色
- 条件
- 数据产生
- 描述符
- 距离
- 指数
- 焦点(扫描)
- 焦点 (使用 Opts)
- 焦点 (运用 Opts)
- 全局
- 矩阵
- 其它
- 相关
- 尺寸
- 叠加统计
- 统计
- 字符串
- 地表
- 三角法
- 带状

Variable object numbers

- 面对现存的所有对象
- 由对象放置的顺序决定
- 用于识别对象

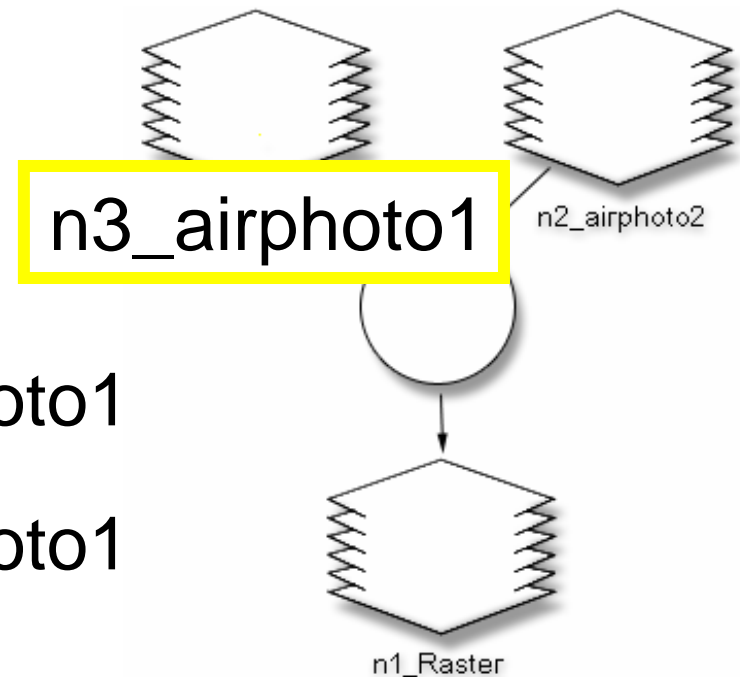


Variable * Explanation

- 根据变量位置顺序编号
- 用星号代替变量数

$n^*_\text{airphoto1} = n1_\text{airphoto1}$

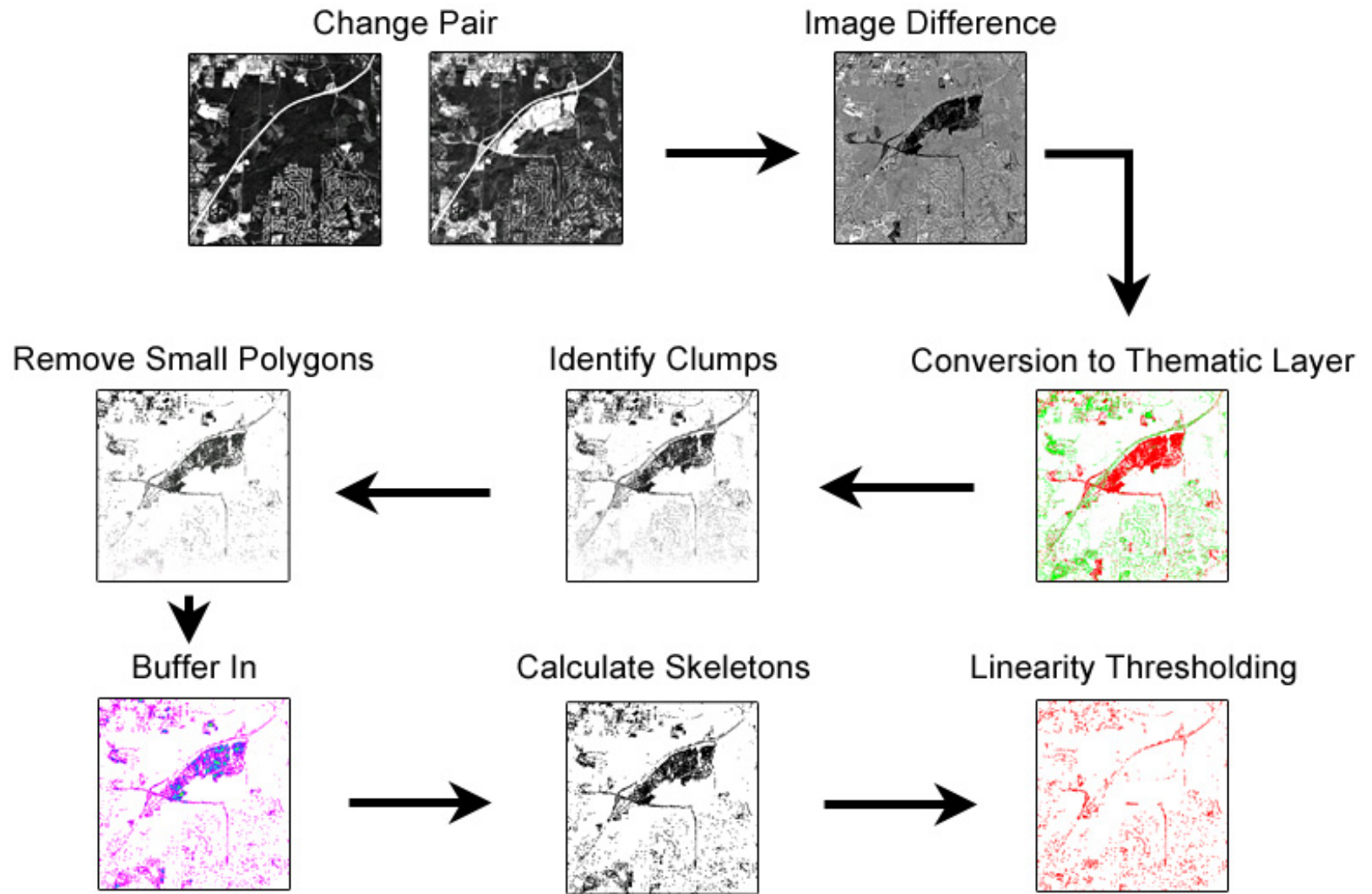
$n^*_\text{airphoto1} = n3_\text{airphoto1}$



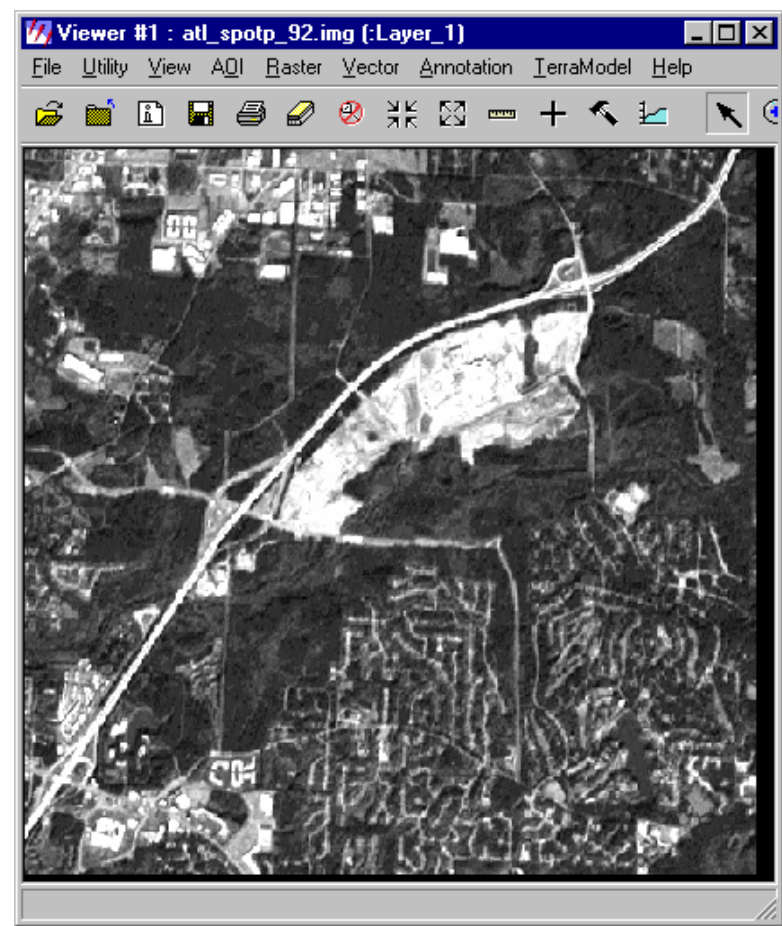
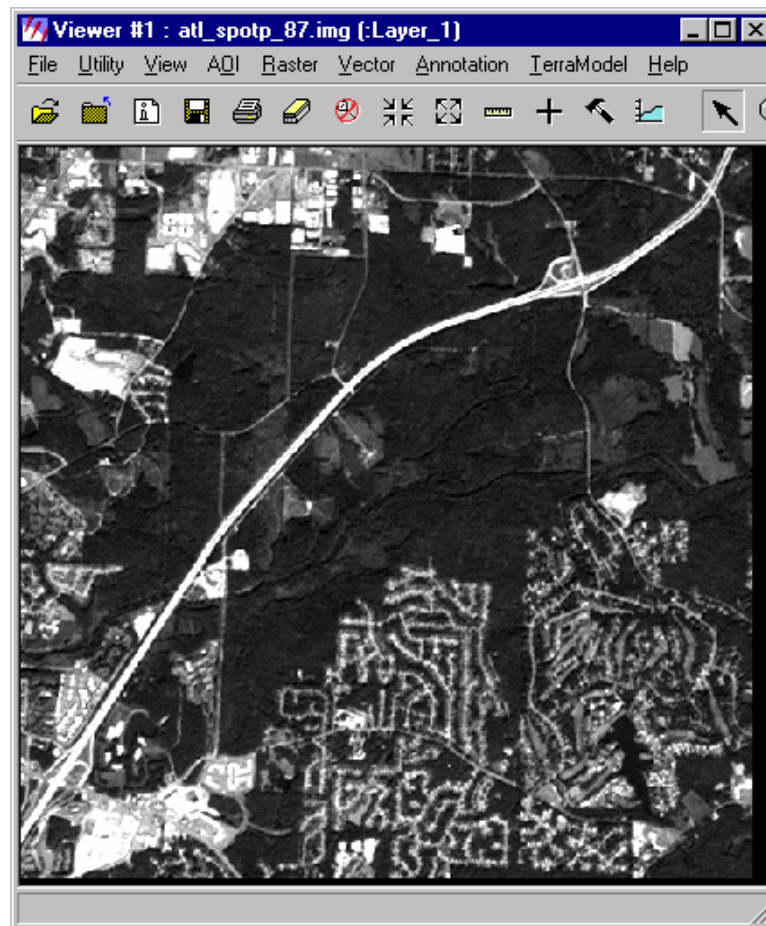
Application

- 对两张影像进行增强和变化监测
- 对差值影像进行分类
- 对于特定大小的多边形进行聚类 and 筛选
- 对于公路网：
 - 在聚类的多边形中作缓冲分析，以帮助识别线性图斑
 - 在缓冲后运行矩阵以辨识多边形中的轮廓线数量，如果数目过大，则多边形不是线性的
- 在最后的文件中加上颜色和类名属性
- 创建一个大的工作流，包括所有的模型并可以同时运行，适用于任何一对需要变化监测的影像
- 利用道路矢量图来判断道路是否是新的线性变换，并编辑矢量图

Application

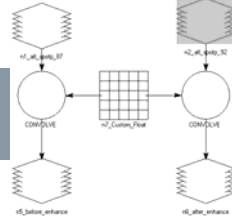


Analyzing a change pair

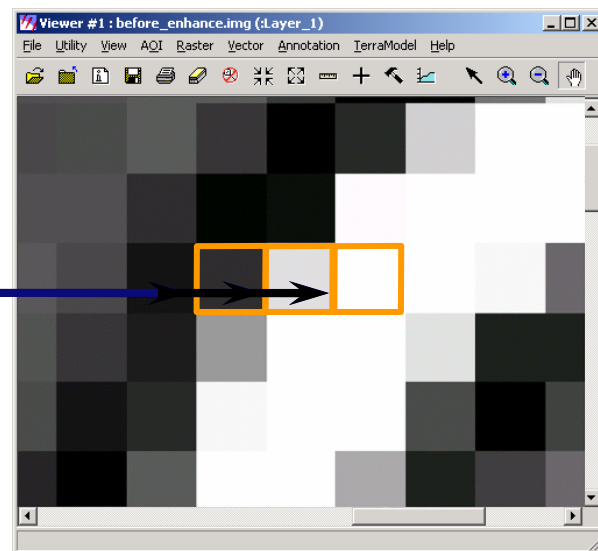
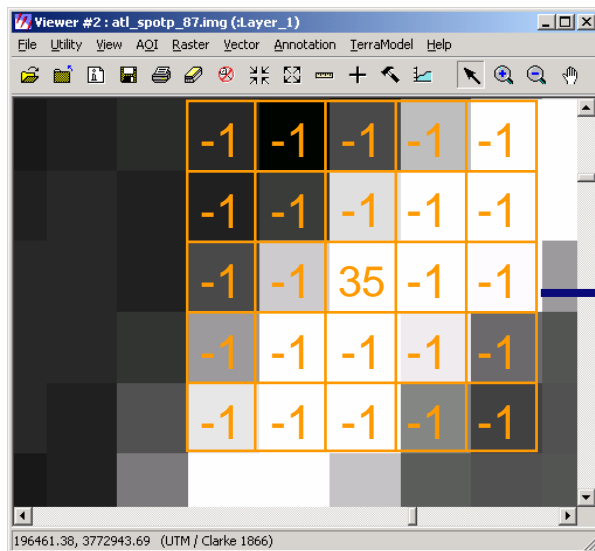


练习 3: 图像增强

Enhancement



- 用边缘提取滤波 (卷积滤波) 增强道路
- 卷积滤波是对整幅影像进行小单元像素平均化的处理过程



Normalize the Matrix

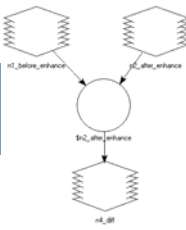
- 用原值除以矩阵中所有像素值之和后得到的值代替原来的值

归一化
3x3 矩阵 =

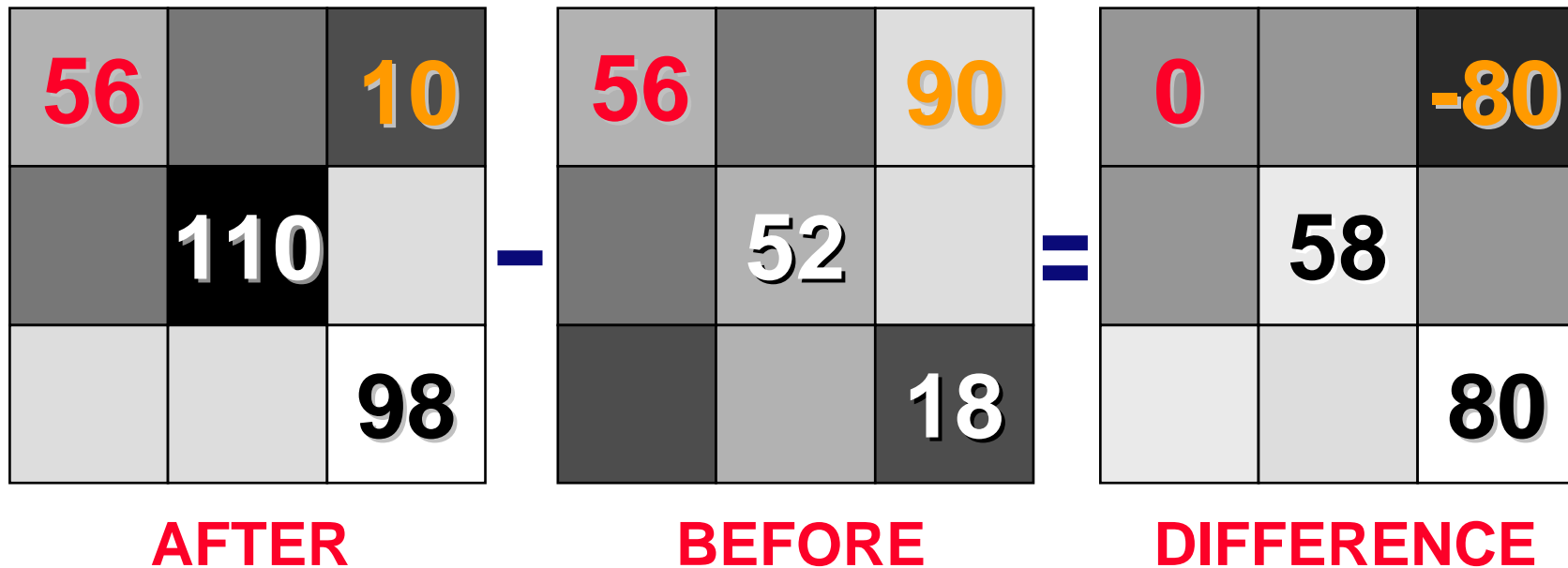
1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

练习 4: 使用变化检测

Image Differencing

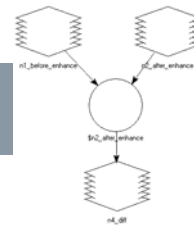


- 从**AFTER** 影像减去**BEFORE** 影像



•

Image Differencing



- 亮区和暗区表示变化区域

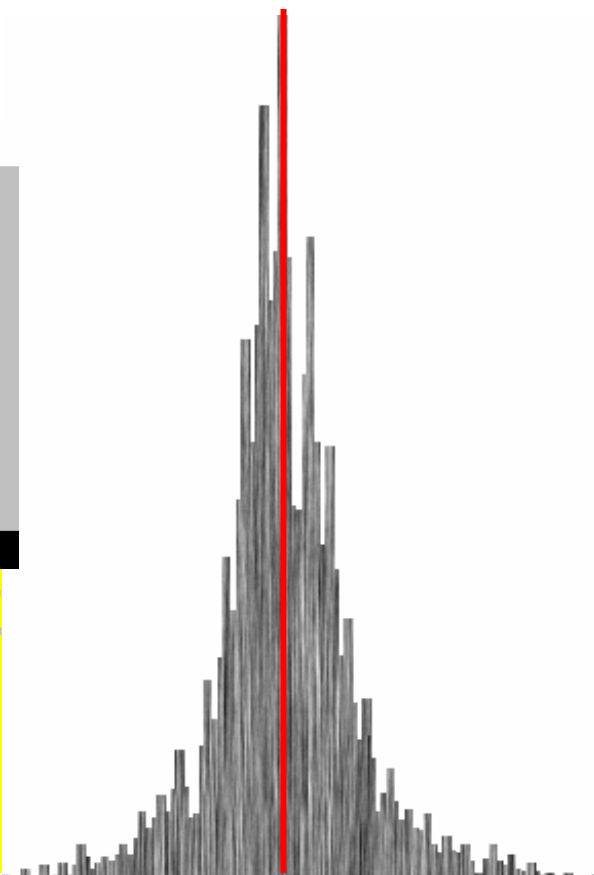
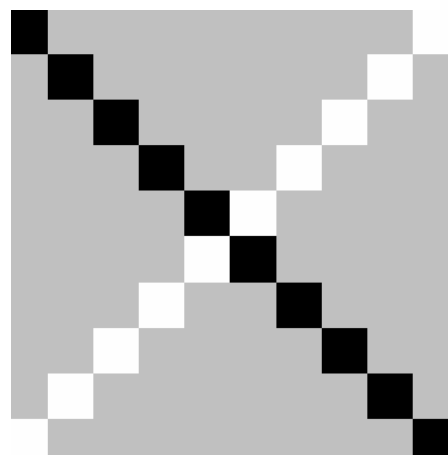
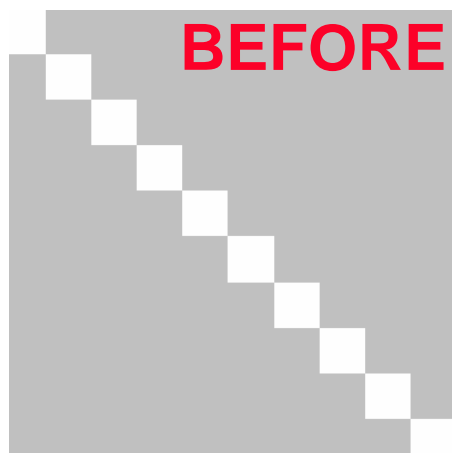
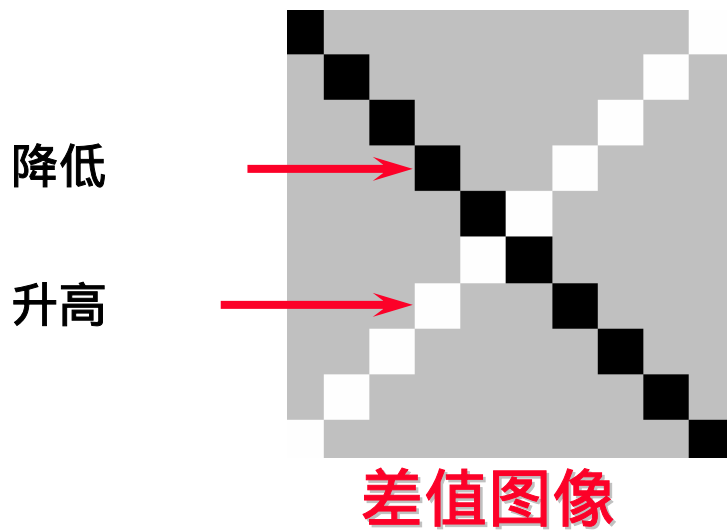


Image Differencing



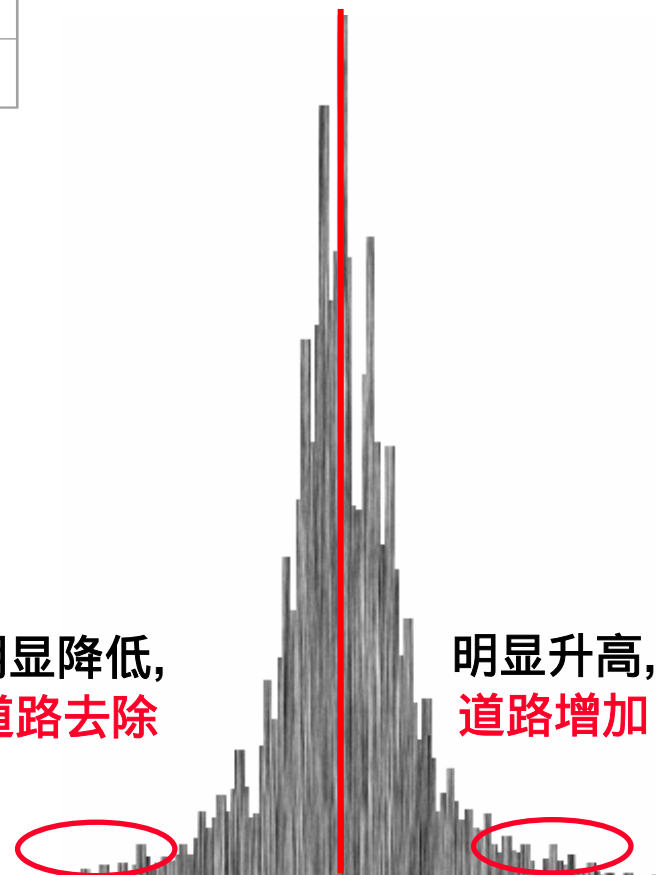
AFTER		BEFORE		diff.img
100	-	20	=	80
20	-	100	=	-80
50	-	50	=	0

接近平均值的区域
= 无变化



明显降低,
道路去除

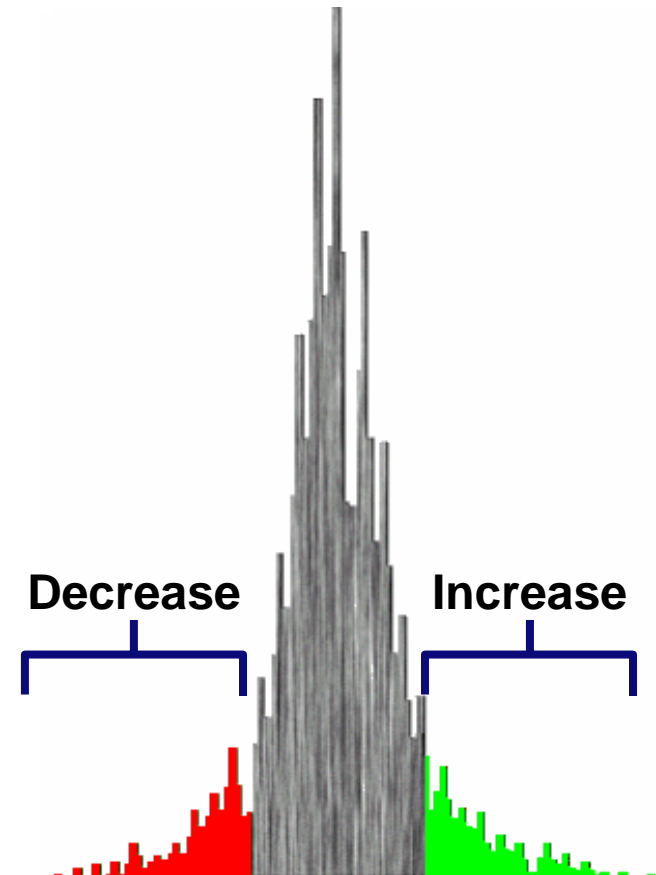
明显升高,
道路增加



练习5:

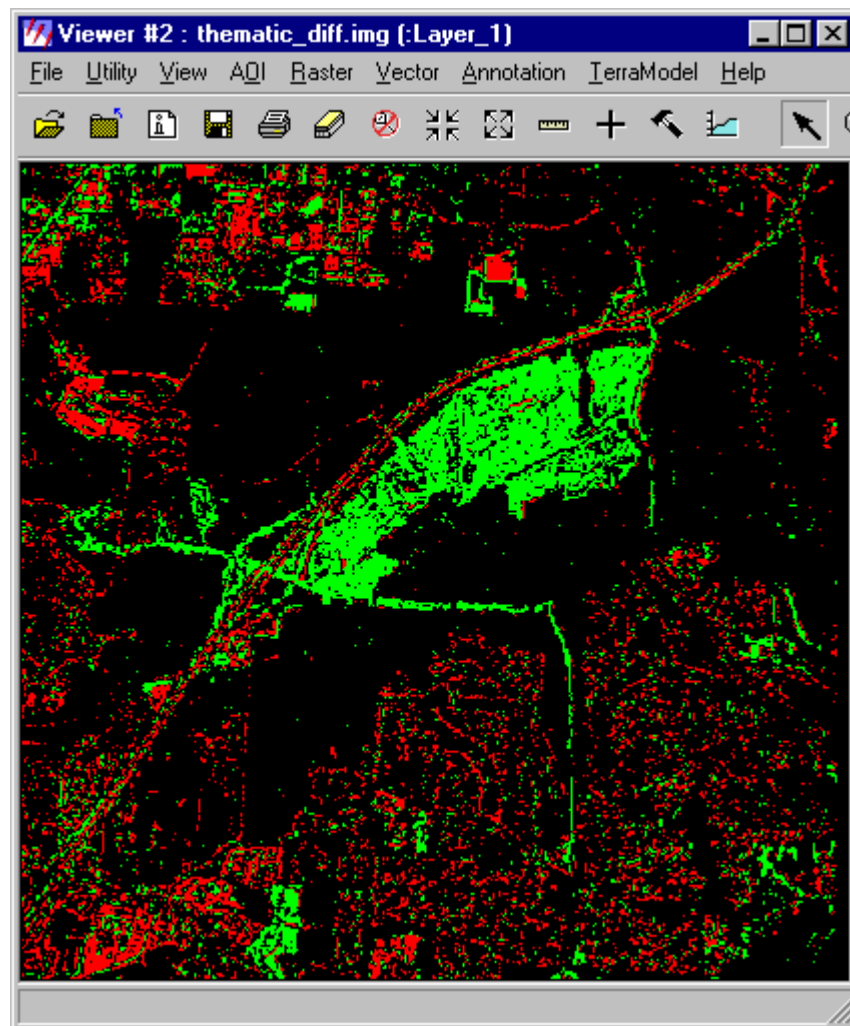
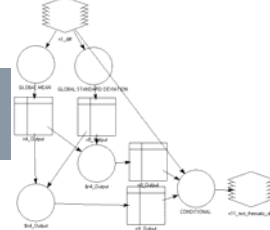
差值图像的专题分类

- ## The Tails



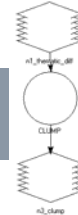
Classify the change

- 将差值图像简单化为只有较大变化的区域
- 集中在“尾巴”上
 - 明显升高的值
 - 明显降低的值

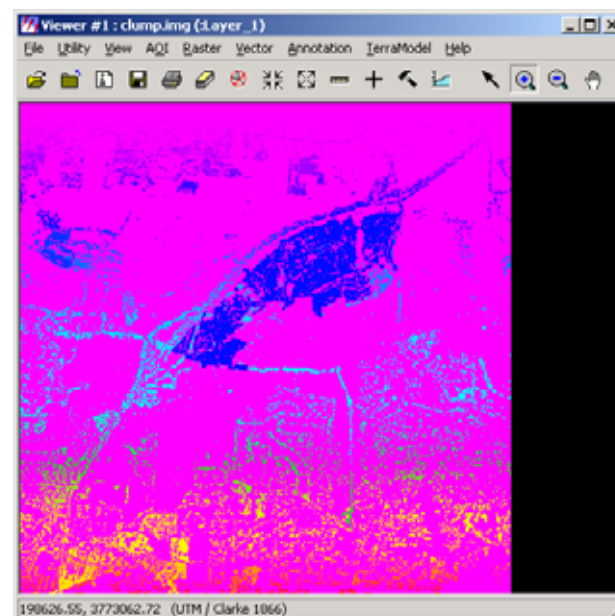
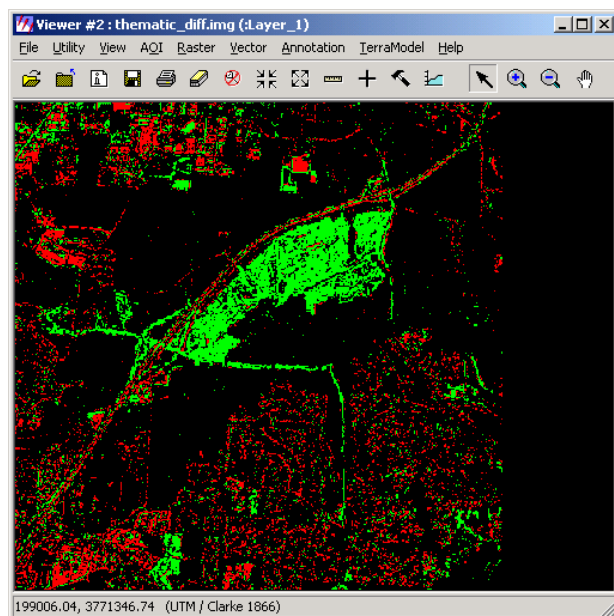


练习6: 检查图斑

Clump

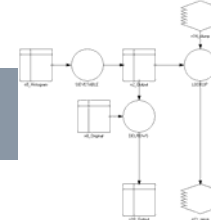


- 辨别临近像元群
 - 可以考虑邻近的4或者8个像元
- 多边形需要一个独立的识别符



练习 7: 去除小的图斑

Eliminating Small Clumps



- 关注道路网的变化
- 10 米数据
 - 小路 = 1-2 像元
 - 主路 = 3-5 像元
- Sieve函数去除比最小尺寸小的块

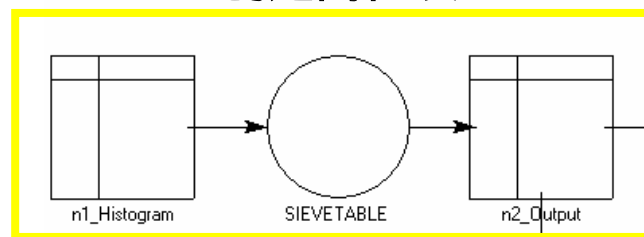
sieve.img

Row	Histogram	Original Value	Color
0	229610	0	
1	6	2	
2	23	2	
3	26	2	
4	29	2	
5	12	1	
6	29	2	
7	136	1	
8	10	2	
9	13	2	
10	9	1	
11	23	2	
12	6	2	
13	6	1	

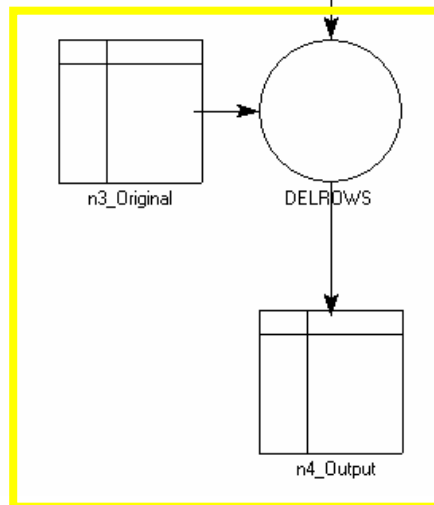
1262 Records

Eliminating Small Clumps

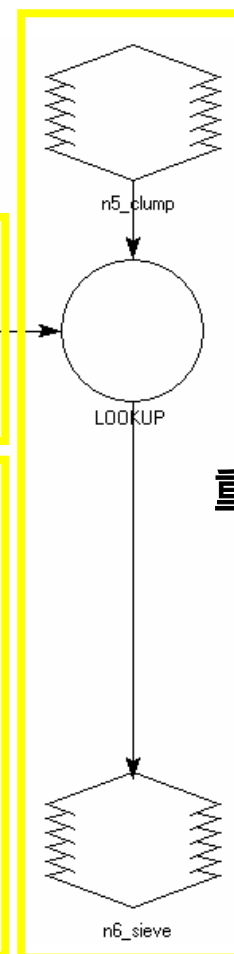
任务 1 创建阈值表



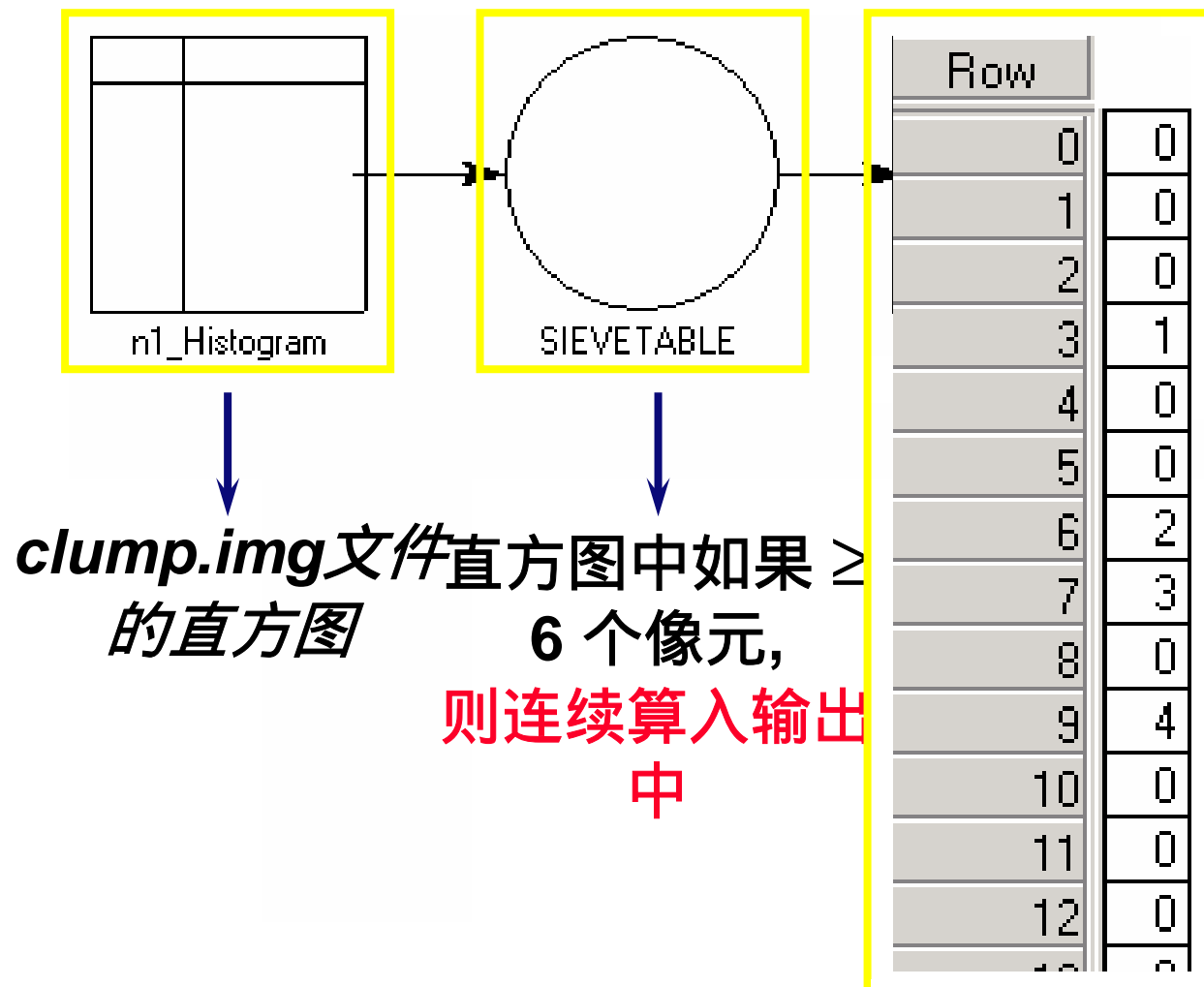
任务 3 返回原始值信息



任务 2 重新计算二进制数

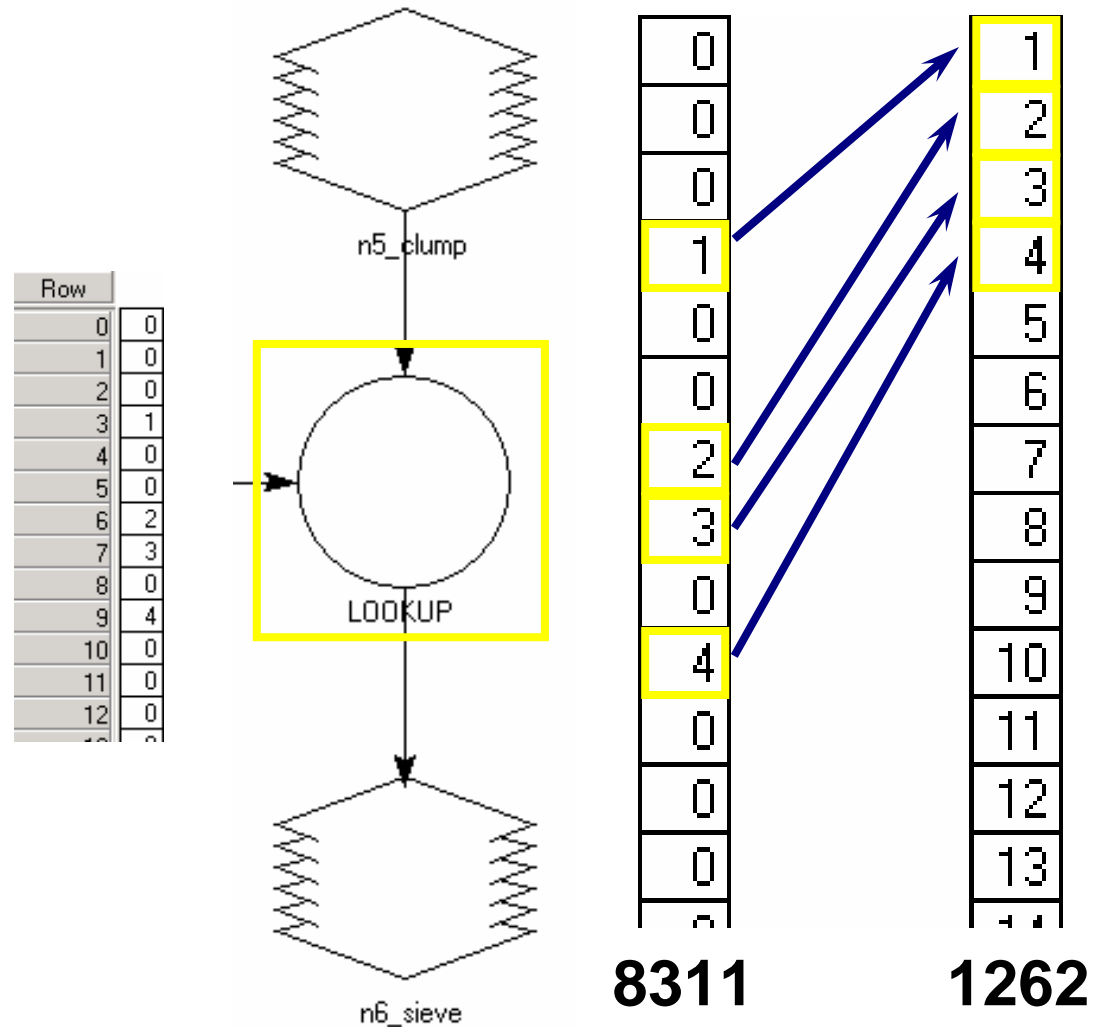


Eliminating Small Clumps – Task 1



Eliminating Small Clumps – Task 2

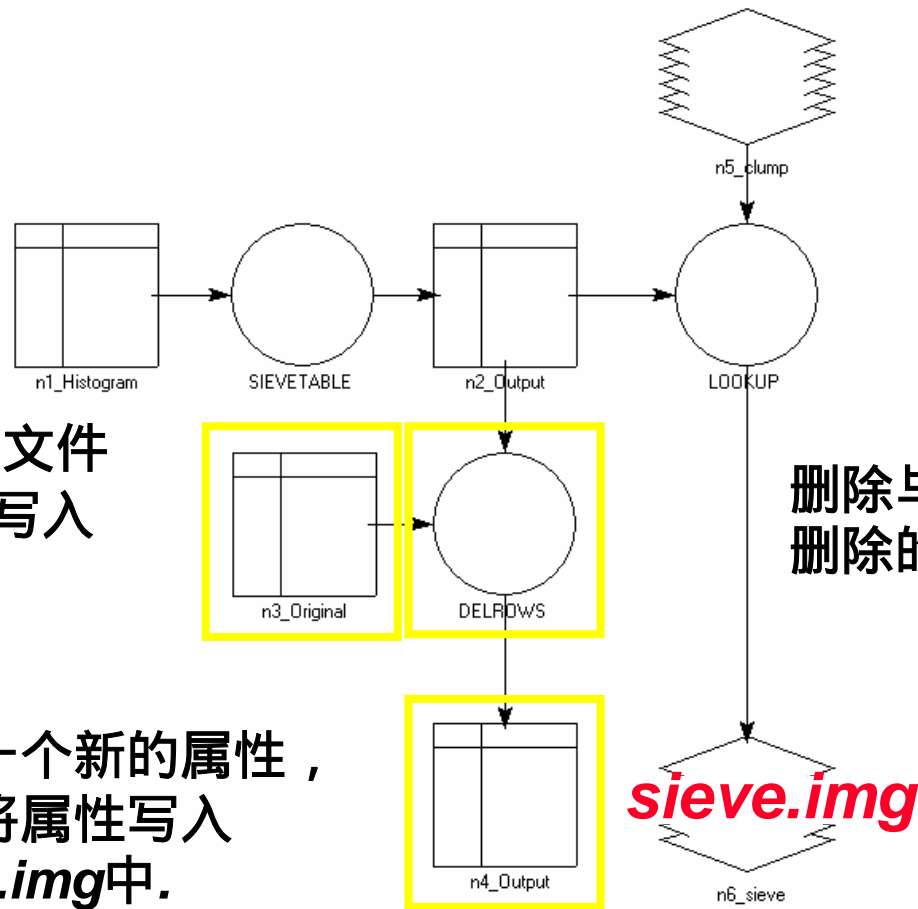
- 值大于0的块被放入输出栅格文件中
- 总的块从8311减少到1262



Eliminating Small Clumps – Task 3

将 *clump.img* 文件的原始属性值写入一个表中。

创建一个新的属性，然后将属性写入 *sieve.img* 中。

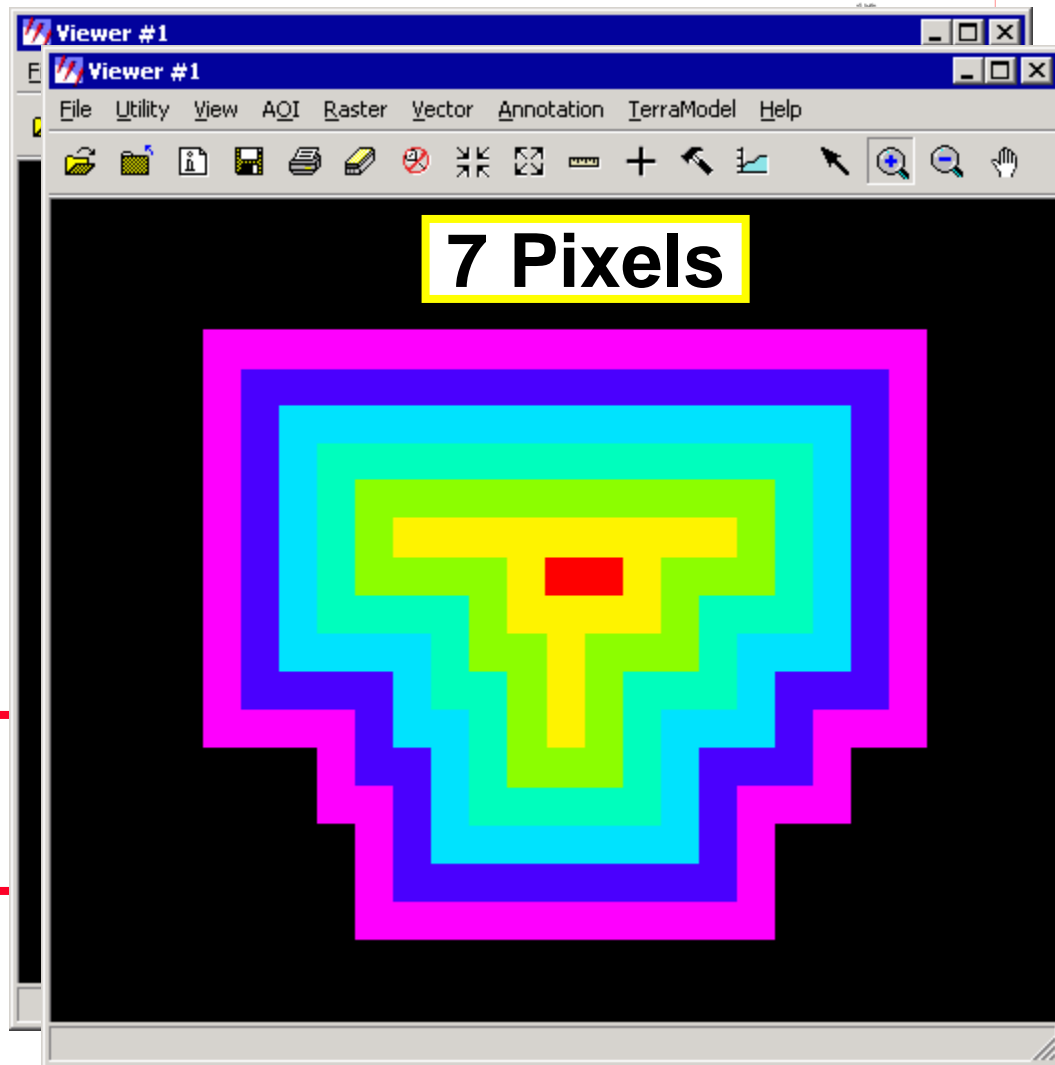
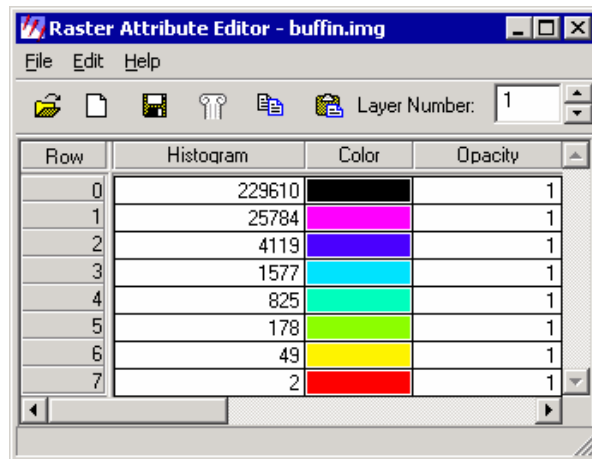


删除与从直方图列中删除的行相关的行。

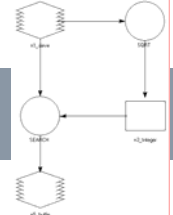
练习 8: 块缓冲分析

Buffering In

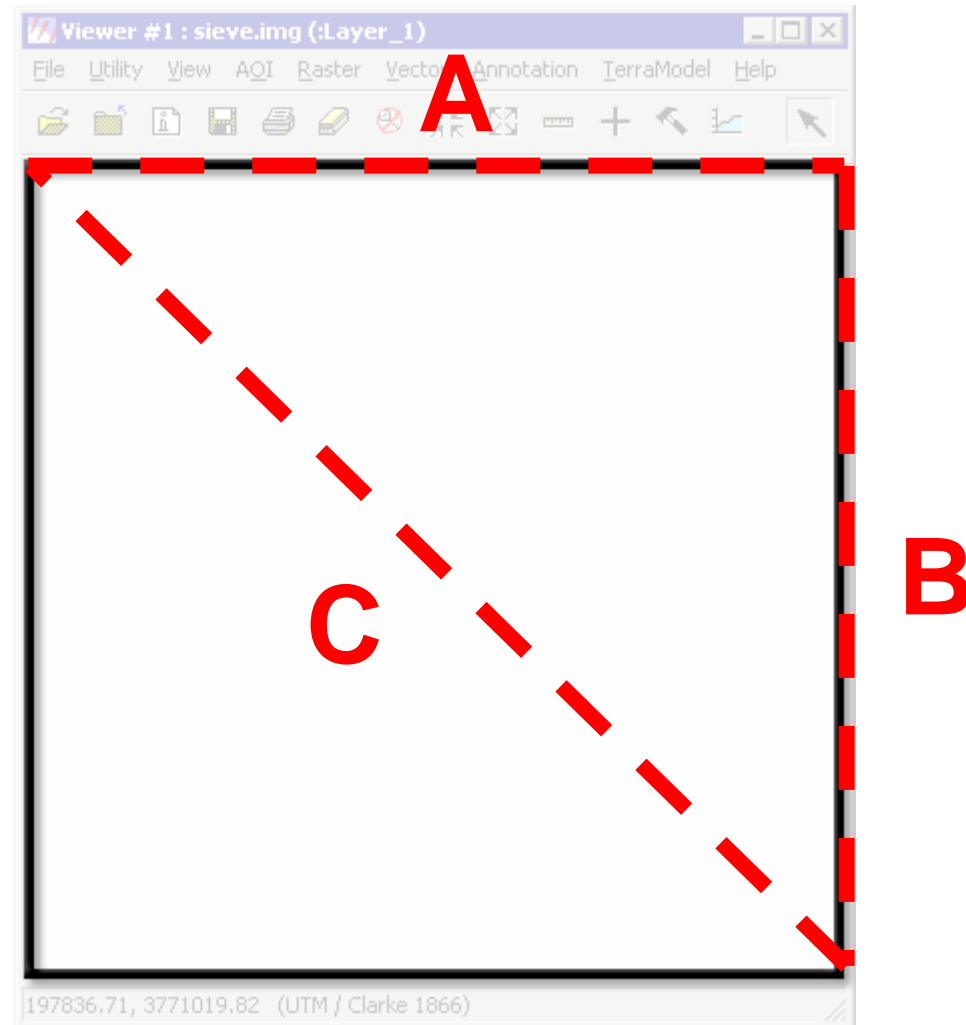
- 通过无数据区域的缓冲区分析创建这个区域多边形的等高线



Buffer Search Size

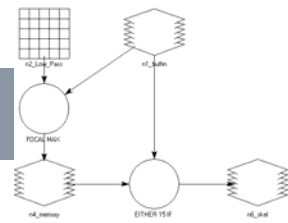


- 由勾股定理决定
- $C^2 = A^2 + B^2$
- C是最可能的缓冲数值



练习 9: 寻找轮廓线

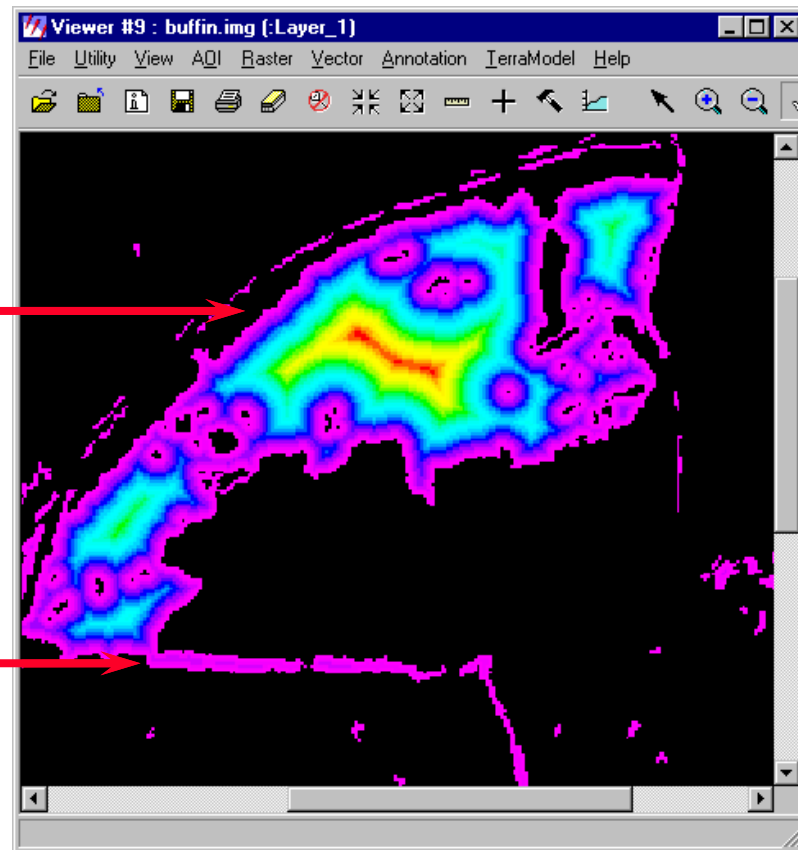
Finding Skeletons



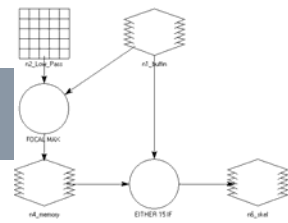
- 在多边形缓冲区中使用一个矩阵，可以提取出每个多边形的轮廓线。

多条轮廓线 != 线性特征

一条轮廓线 = 线性特征

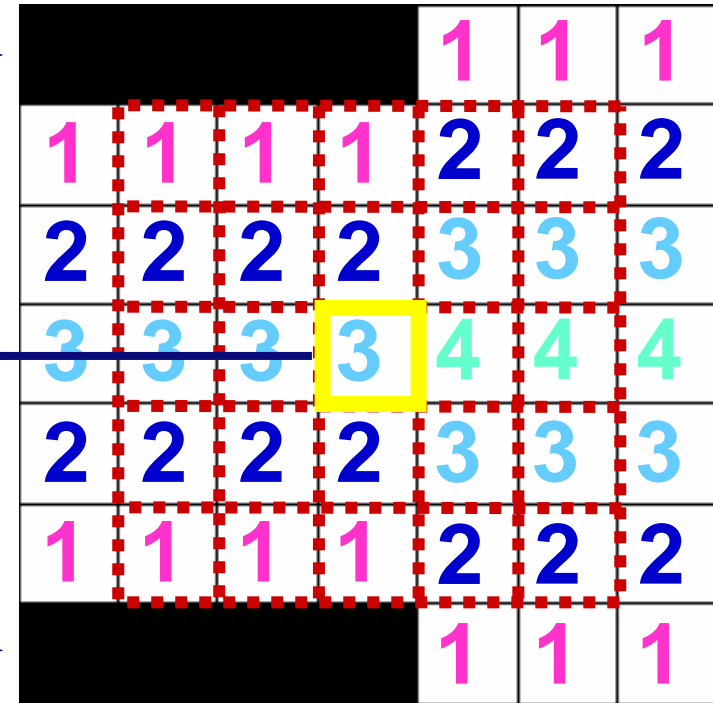
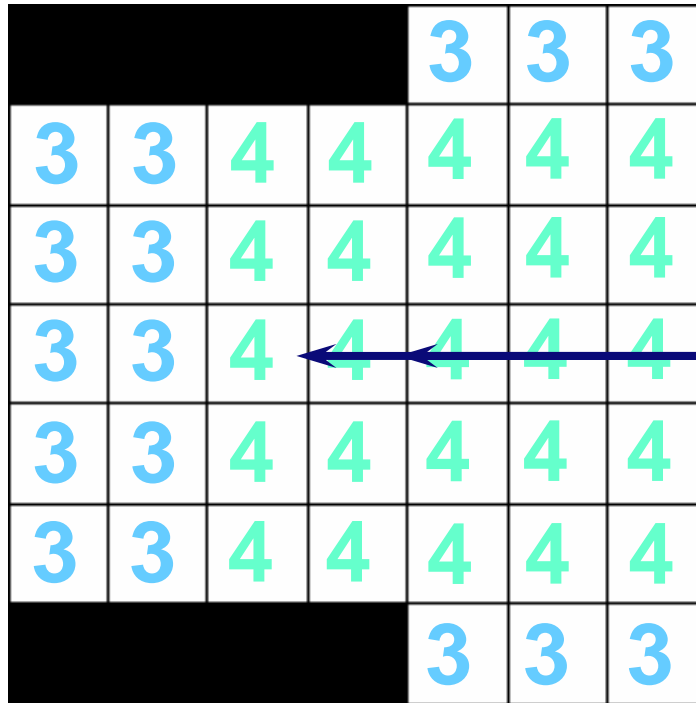


Finding Skeletons

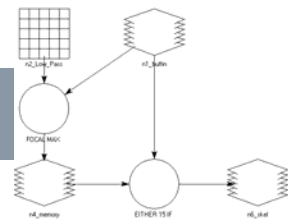


- **Skeletons**是对象内部的中心点

焦点最大值 - 返回焦点窗口
区域中的最大值



Finding Skeletons



Skeleton

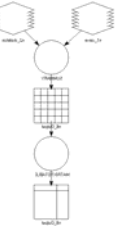


				2	2	2
2	2	3	3	2	2	2
1	1	2	2	1	1	1
0	0	1	1	0	0	0
1	1	2	2	1	1	1
2	2	3	3	2	2	2
				2	2	2

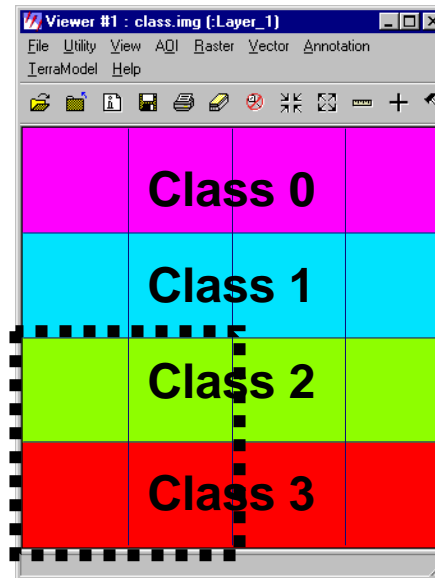
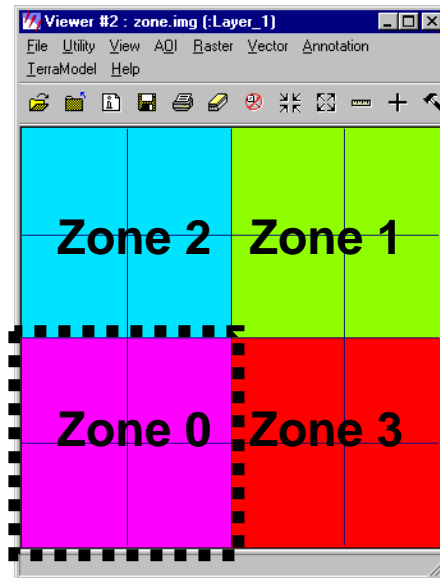
练习 10:

了解函数 Summary Functions

Summary Functions



- 通过两个输入栅格文件的交叉列表，形成一个矩阵



		Classes			
		0	1	2	3
Zones	0	0	0	2	2
	1	2	2	0	0
	2	2	2	0	0
	3	0	0	2	2

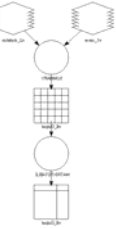
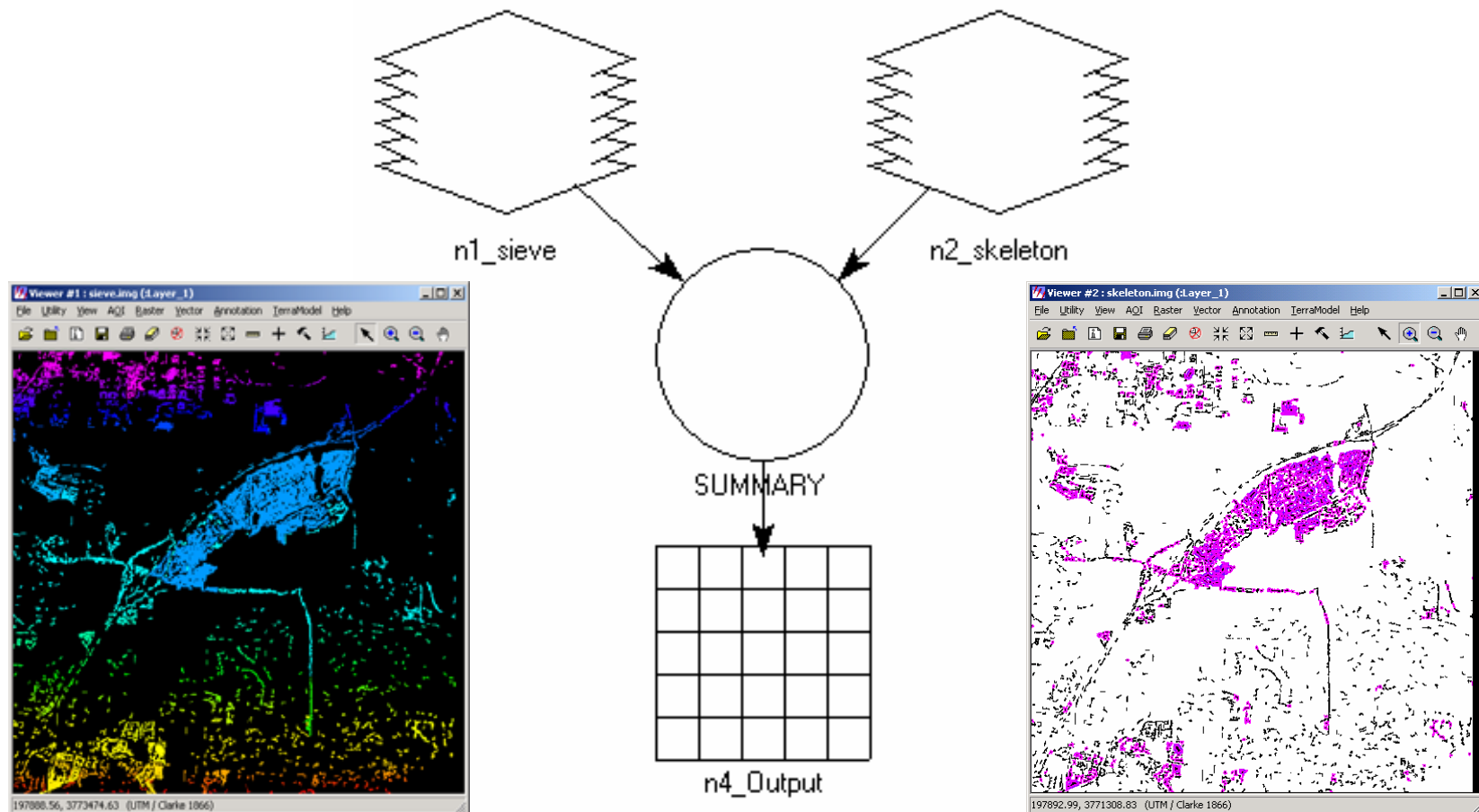
Zones = 3 + 1 = 4 rows

Classes = 3 + 1 = 4 columns

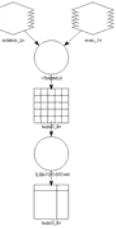
Summary Functions

Zones = sieve.img

Classes = skeleton.img



Summary Functions



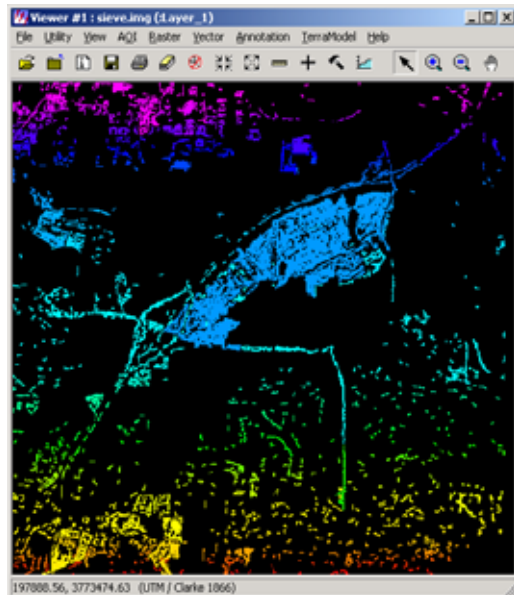
- 全局最大值 – 计算图像每一层的最大值

Global Max

1262

Zones

1262 + 1 = 1263 rows



Classes

0 1 . . . 15 16

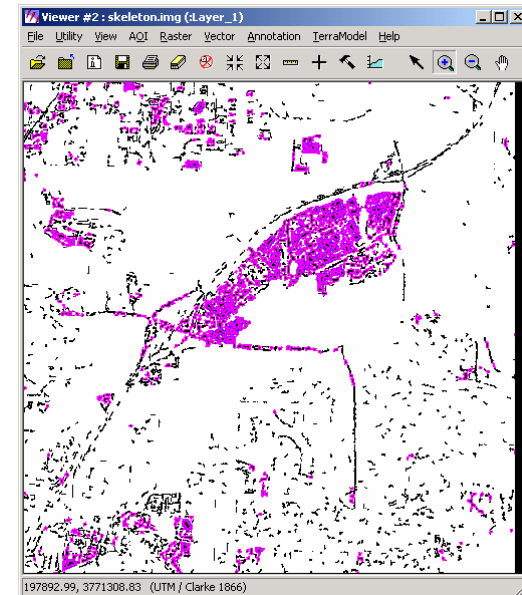
0						
1						
.						
.						
.						
.						
1261						
1262						

Global Max

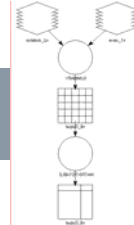
15

Classes

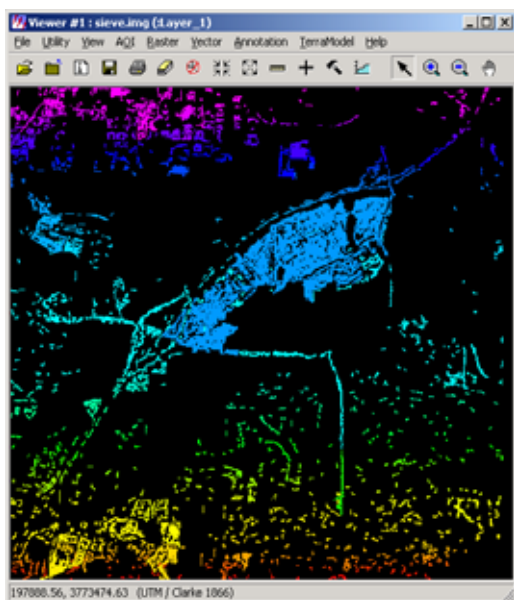
15 + 1 = 16 columns



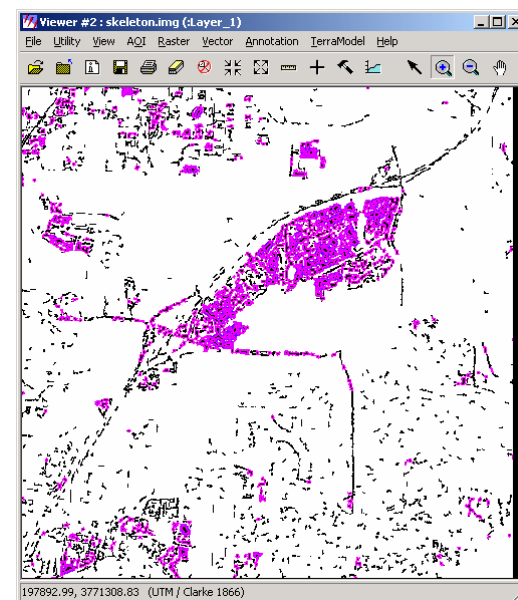
Summary Functions



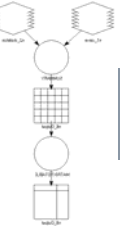
- **skeleton.img** 文件的0类定义Skeleton
- 0列描述在每一个zone中的Skeleton像元数 (过滤多边形)
- 将0列输出到一个表中



		Classes						
		0	1	.	.	.	15	16
Zones	0							
	1							
	.							
	.							
	.							
	1261							
	1262							



Summary – Matrix Subexpressions



- 从矩阵中复制一部分定义成新的对象

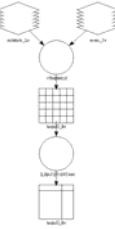
- **Skeleton**

<matrix-expression> [<row>, <column>]

- **Matrix**

- <matrix-expression> [<startrow> , <startcolumn> : <endrow> , <endcolumn>]

Summary – Matrix Subexpressions

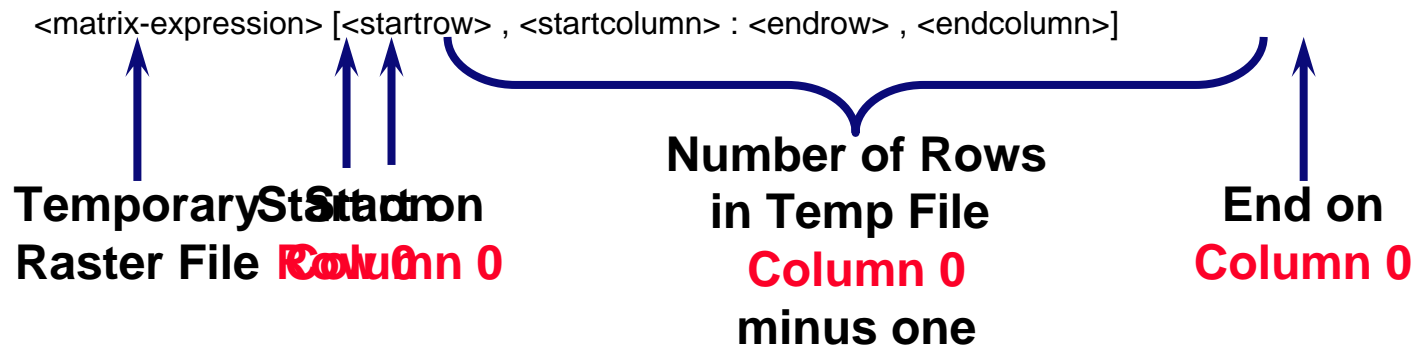


■ Matrix

<matrix-expression> [<startrow> , <startcolumn> : <endrow> , <endcolumn>]

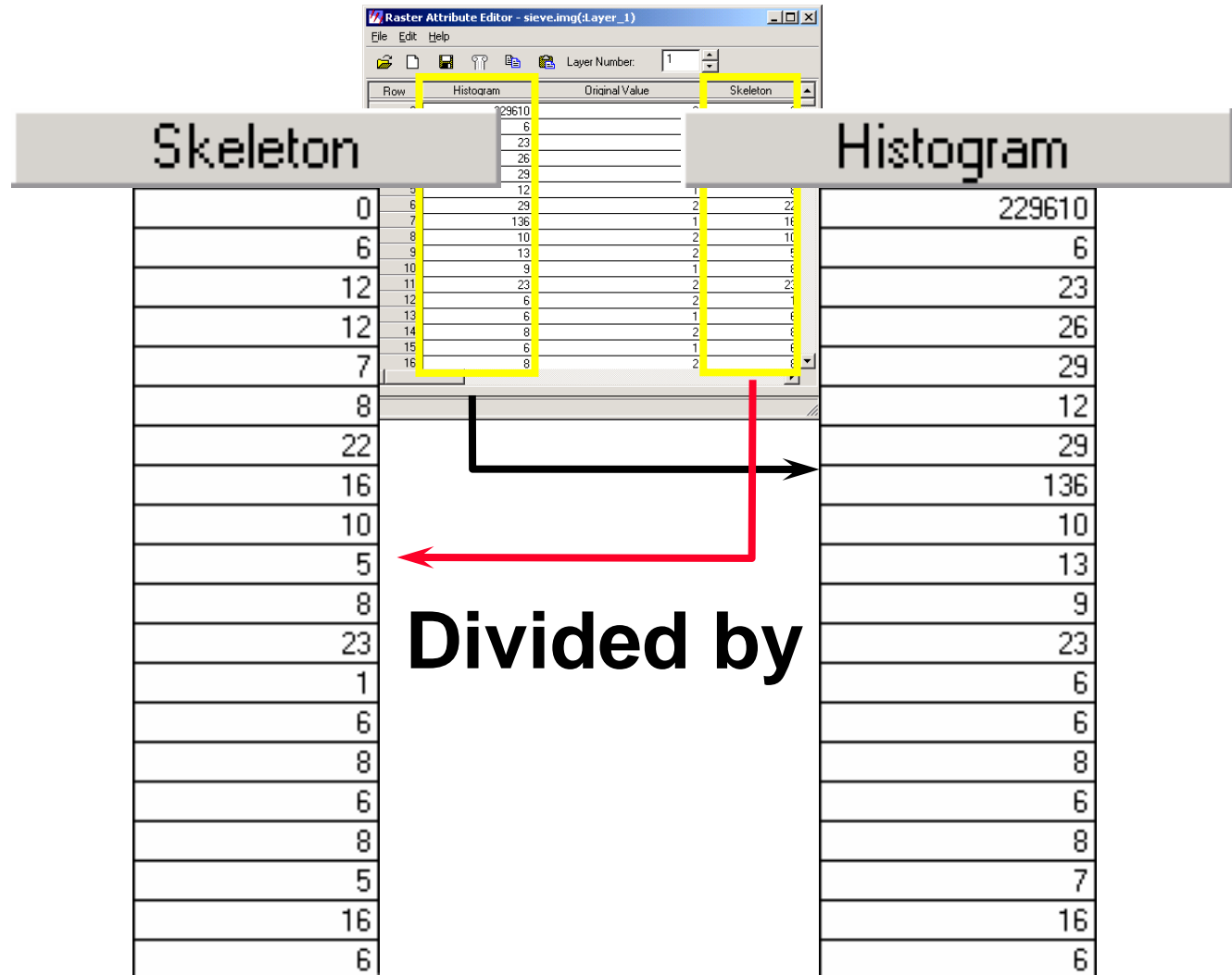
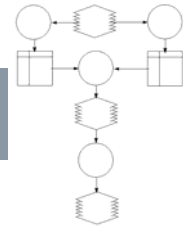
■ 矩阵到表

(\$n6_Output [0,0: NUMROWS (\$n6_Output)-1,0])

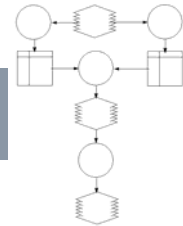


练习 11: 检查线性关系

Object Linearity



Object Linearity

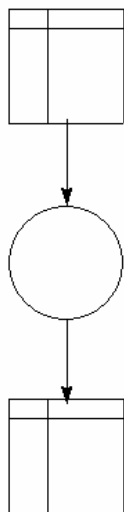


- 如果接近 1, 则是线性的
- 如果接近 0, 则是非线性的
- 阈值
 - 选出来用于描绘线性关系的数值
 - 例如阈值 = 0.5, 那么
 - 如果 > 0.5 , 那么是线性的, 则加到输出文件中
 - 如果 < 0.5 , 那么是非线性的, 则不加到输出文件中

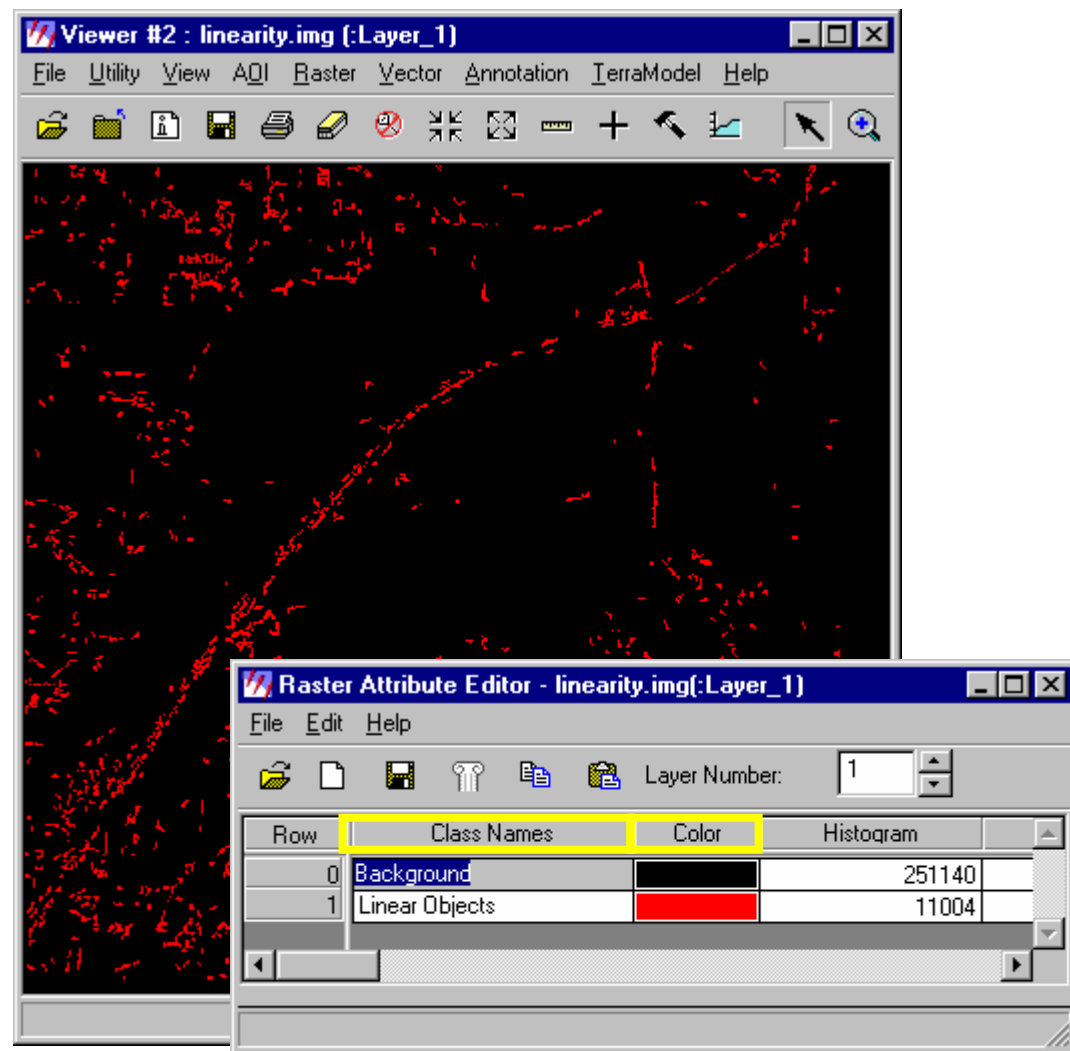
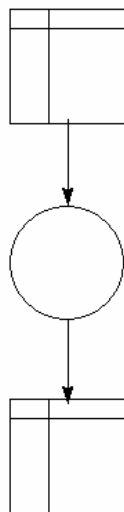
Adding Attributes

- 加到Final Linearity 文件中的值

Class Names



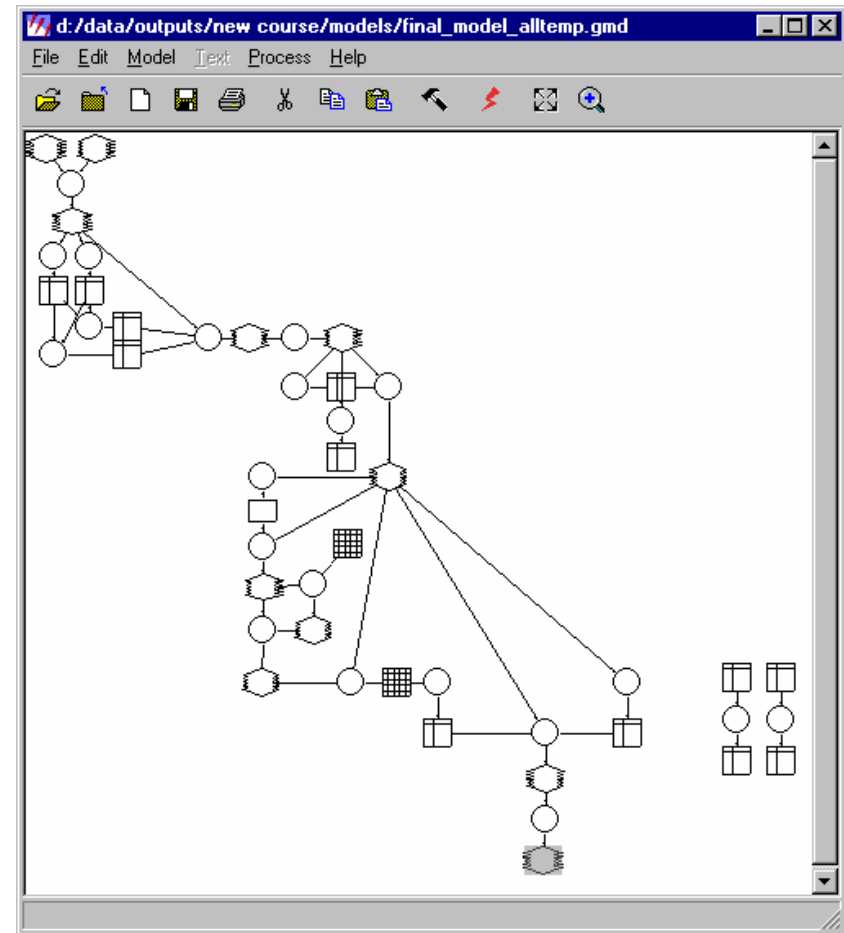
Color



练习 12: 集成

Simplifying the Workflow

- 创建一个整体模型来运行所有的文件
- 利用临时文件节省存储空间
- 增加灵活性，任何变化监测的影像都可以使用



练习 13: 更新矢量

Vector Updating

- 新识别的道路可以用来更新已有的矢量图层，或者创建新的矢量数据

