

中南大学地理信息系统专业系列实验教材

3 S 应用集成系统 实 验 指 导 书

(邓吉秋)

中南大学 GIS 研究中心
2003 年 9 月

目 录

1	课程地位、性质	1
2	课程基本要求	1
3	实验目的	1
4	实验内容	1
5	实验要求	2
6	时间安排	2
	实验一、GIS 与 RS 信息融合	3
	实验二、GPS 与 GIS 数据集成	9
	实验三、GPS 与 RS 数据集成	10
	实验三、GPS 与 RS 数据集成	11
	实验四、GIS 与 RS 数据集成	12
	实验五、GPS 与 GIS 应用集成系统的设计	15
	实验六、RS 与 GIS 应用集成系统的设计	16

1 课程地位、性质

《3S 应用集成系统》是地理信息系统专业教学计划中的一门专业必修课。

本课程是地理信息系统(GIS)、遥感(RS)、全球定位系统(GPS)的集成以及空间信息系统的实现主线，研究空间信息系统集成的特点、方法、过程和实现技术。

学习本课程，应具备地理信息系统、遥感、全球定位系统及地理信息系统分析、设计与开发等方面的基础知识。通过本课程的学习，学生应掌握和了解空间信息系统的集成与实现，3S 集成的内容、方法与关键技术，初步具备 3S 集成应用系统设计与开发的能力。

《3S 应用集成系统》包括地球空间信息科学的技术体系、空间信息系统集成的重要手段和方法、3S 集成的模式与关键技术、空间数据库及其组织、地球空间信息框架和空间信息系统功能、结构、核心技术及发展，概括了 3S 集成与空间信息系统的基础理论、关键技术、实现方法等。

2 课程基本要求

根据本课程在教学计划中的地位，通过本课程的学习，应达到如下基本要求：

1. 了解地球空间信息科学的基本功能、内容，以及与相关学科的关系；
2. 掌握空间信息系统的特点、功能、结构、核心技术；
3. 掌握空间信息系统集成与实现的重要方法和手段；
4. 了解 3S 集成的模式，各种模式的特点、内容与关键技术；
5. 掌握简单集成系统的设计与开发。

3 实验目的

通过 3S 不同集成方式的理解、分析、实例操作以及设计与开发，使学生具备简单的 3S 集成应用的能力，基本具备针对具体问题和应用进行基于 3S 的空间信息系统设计和开发的能力。

4 实验内容

本课程的实验操作为主，具体的实验内容包括：

- GIS 与 RS 信息融合：遥感信息融合、GIS 信息融合、GIS 与 RS 信息融合；
- 3S 不同集成方式：GPS 与 GIS 集成、GPS 与 RS 集成、GIS 与 RS 集成；
- 3S 应用集成系统的设计与开发：GPS 与 GIS 集成系统、RS 与 GIS 集成系统。

5 实验要求

1. 个人自主操作，独立完成；
2. 每次实验需提交实验报告，每次上机需检验中间性成果。

6 时间安排

课内实验时间纳入正常的教学计划，学生应在课外对实验内容加以巩固和熟练。

本课程的上机实验内容各种 3S 集成模式的操作处理以及相关空间信息系统的设计与开发，实验中安排 4 次操作性实验、2 次设计性编程性实验，具体内容与学时分配如下表。

序号	实验内容	实验学时
1	实验一、GIS 与 RS 信息融合	4
2	实验二、GPS 与 GIS 数据集成	2
3	实验三、GPS 与 RS 数据集成	2
4	实验四、GIS 与 RS 数据集成	4
5	实验五、GPS 与 GIS 应用集成系统设计	8
6	实验六、RS 与 GIS 应用集成系统设计	4

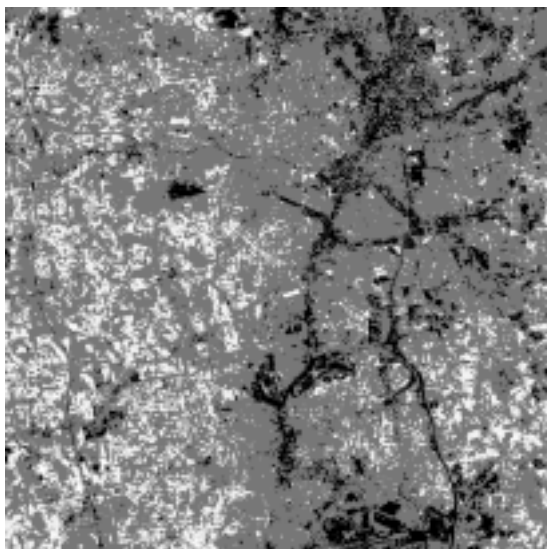
实验一、GIS 与 RS 信息融合

一、遥感数据的融合

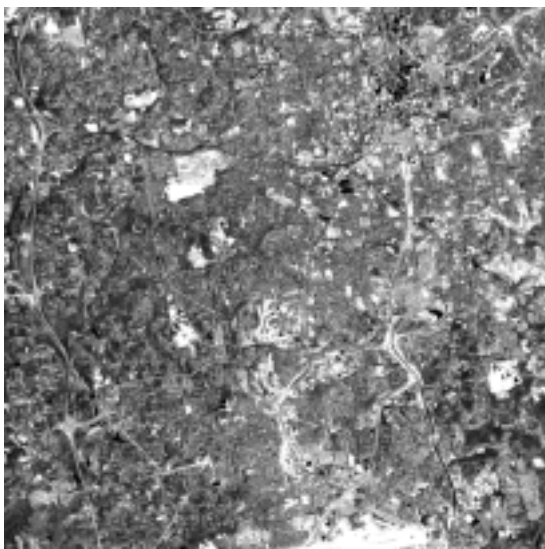
通过对遥感数据波段的算术运算（加、减、乘、除等）和函数运算（Sin、cos、tan、log 等）使得融合后的遥感数据更适合于分类与识别。

1. 同一数据不同波段的融合

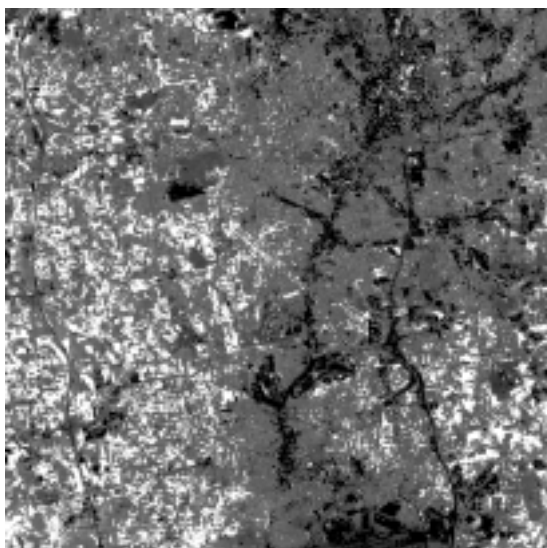
对于美国亚特兰大地区的 TM 数据，原始数据名称为 tmAtlanta，定义为 A_0 ，其波段分别定义为 T_1 、 T_2 、 T_3 、 T_4 、 T_5 、 T_6 ，对其进行算术运算，部分结果如下：



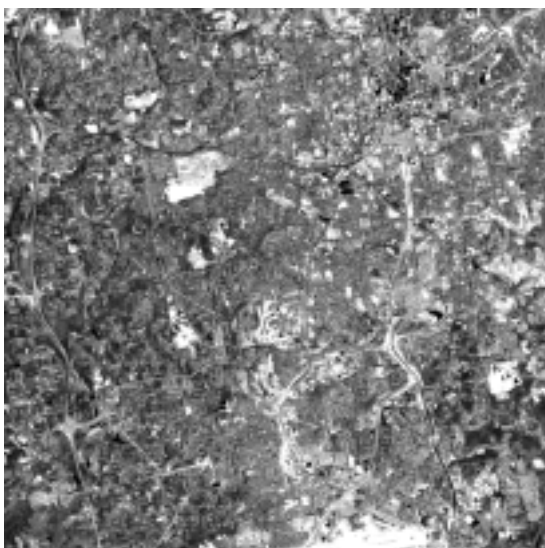
$$A_1 = T_4 / T_6$$



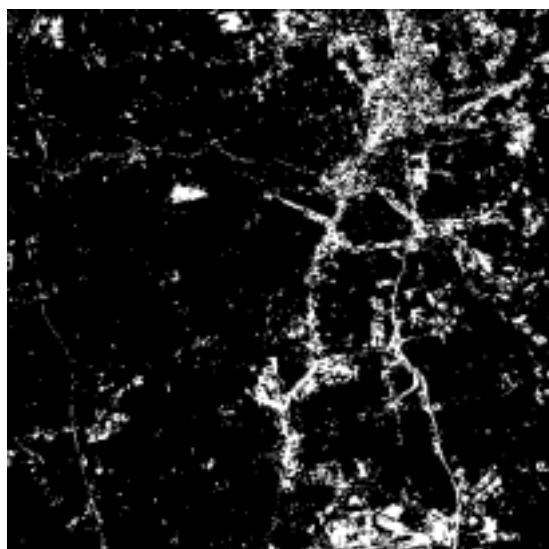
$$A_2 = T_4 + T_6$$



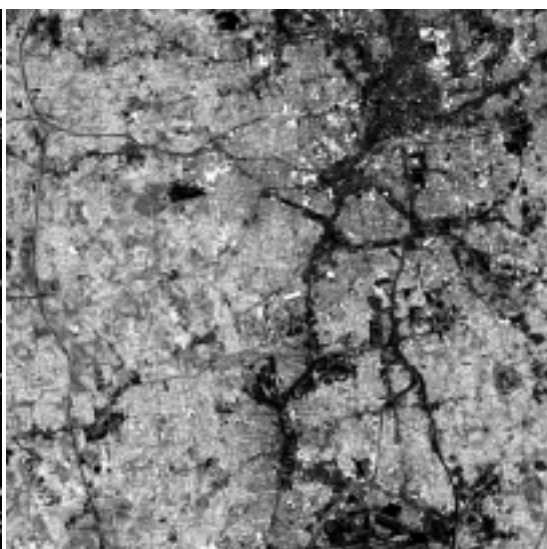
$$A_3 = A_1 / A_2$$



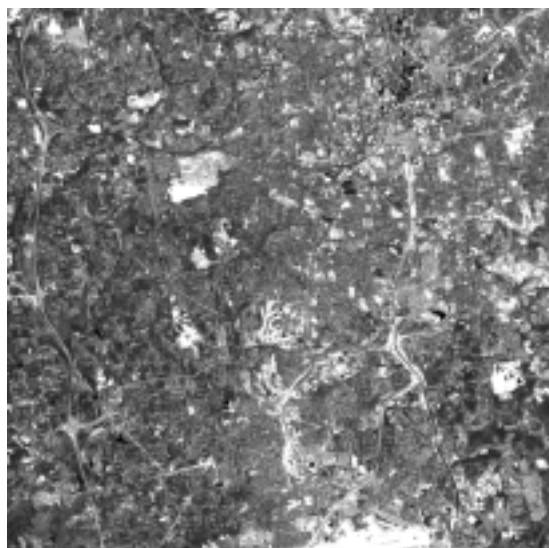
$$A_4 = A_2 - A_1$$



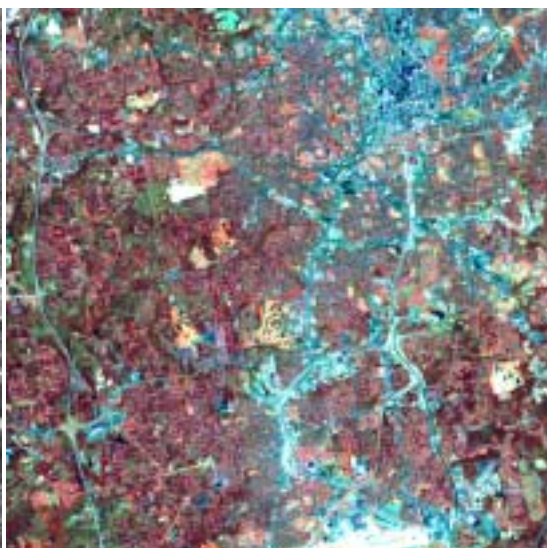
$$A_{11} = T_6 / T_4$$



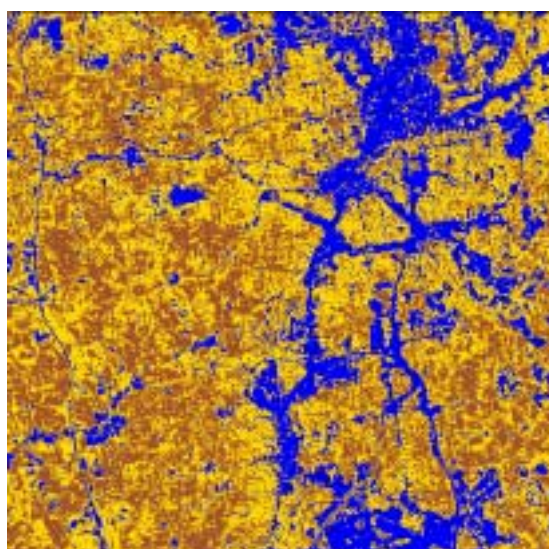
$$A_{12} = T_4 - T_6$$



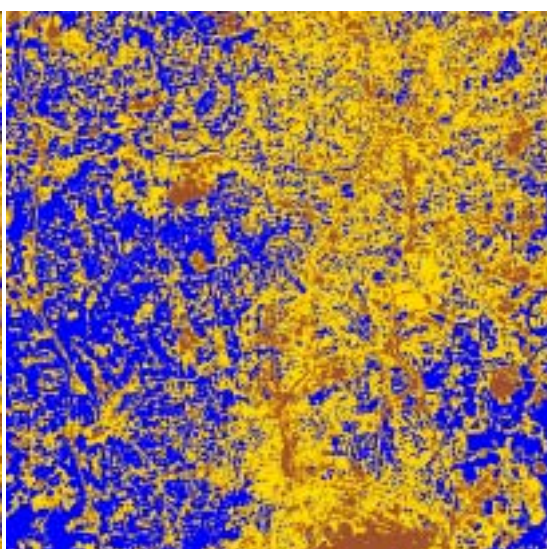
$$A_{1200} = T_4 \times T_6$$



$$A_0$$



利用 A_{12} 分 4 类的结果

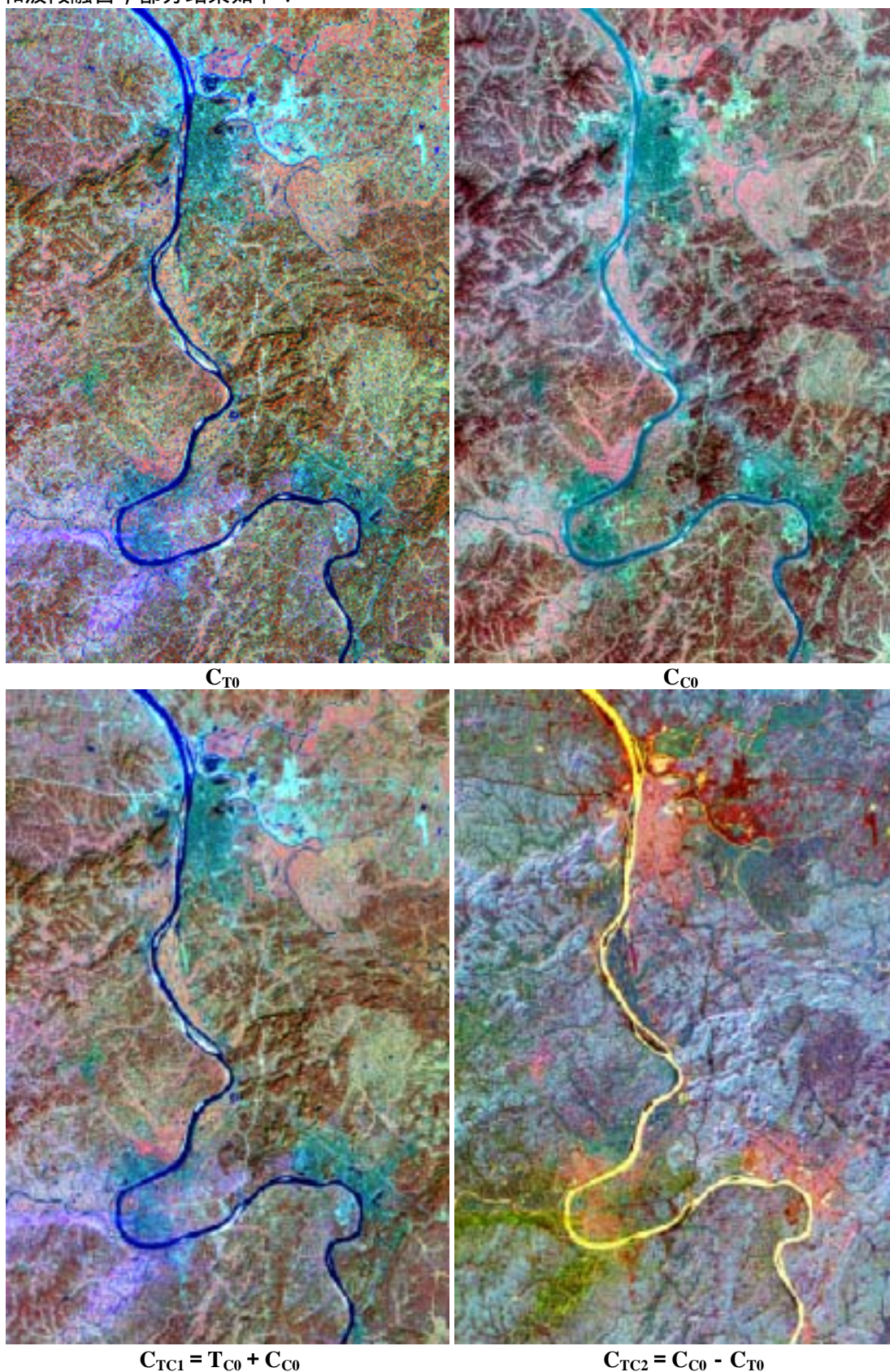


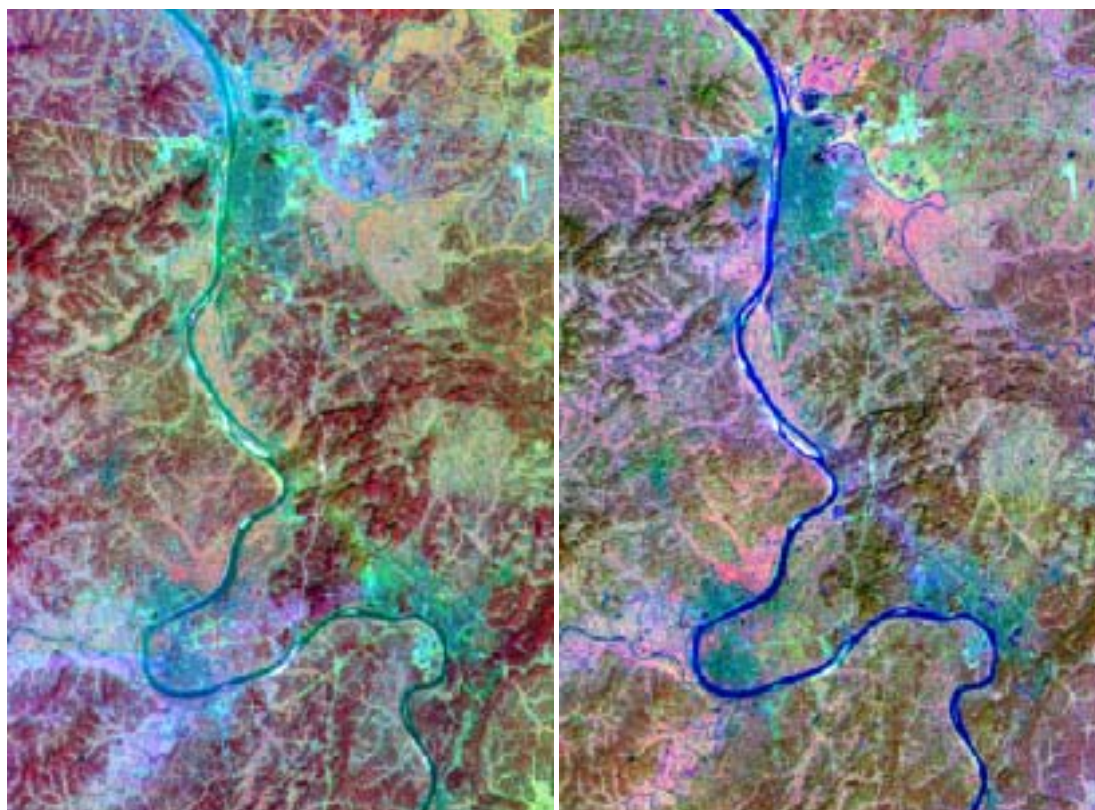
利用 A_0 分 4 类的结果

此结果表明：对分辨城区而言，经过波段融合的再分类结果比原始分类要好

2. 同一区域不同数据的融合

对于长株谭地区的 TM 数据、中巴卫星数据，原始数据名称分别为 tmCZT、cbCZT，分别定义为 C_{T0} 、 C_{C0} ，其波段分别定义为 T_1 、 T_2 、 T_3 和 C_1 、 C_2 、 C_3 ，对其进行算术运算和波段融合，部分结果如下：





$$C_{TC3} = \text{RGB} (T_1, C_2, T_3)$$

$$C_{TC4} = \text{RGB} (C_1, T_2, C_3)$$

二、GIS 数据的融合

通过 GIS 矢量数据的叠加融合生成具有新的涵义的数据。

对于湖南省的行政区与主干公路数据，原始数据名称分别为 region_shi、road_m，将两者进行叠置分析，可以得到每行政区的主干公路长度以及密度。



湖南省市级行政区

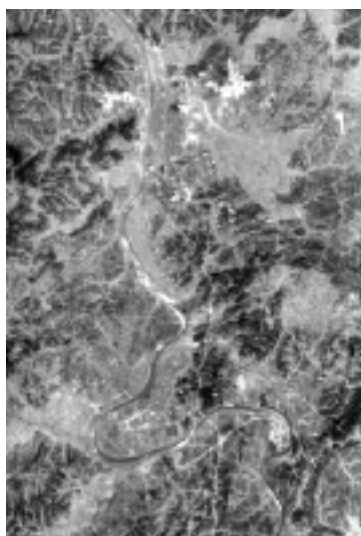


湖南省主干公路

三、GIS 与 RS 数据的融合

通过 GIS 数据与 RS 数据的叠加融合生成具有新的混合数据。

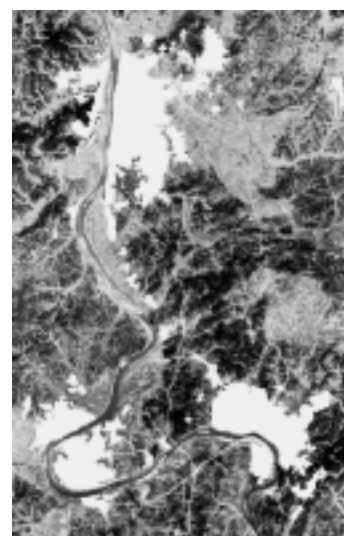
对于长株谭地区的遥感数据与城区范围数据，原始数据名称分别为 cbCZT2、N90，将两者进行混合叠置分析，取矢量多边形的 ID 值与遥感数据数据值的最大值作为重叠部分的新数据值，可以得到两者的混合数据。



原始影像

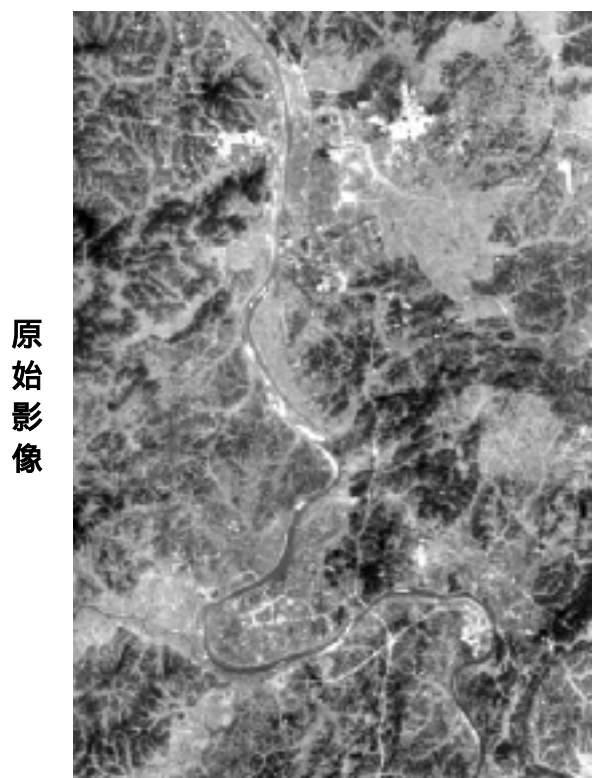


城区范围



混合数据

对于长株谭地区的遥感数据与行政区数据，原始数据名称分别为 cbCZT2、XT，将两者进行混合叠置分析，用矢量多边形去切遥感数据可以得到被矢量数据切边的遥感数据。



原始影像



行政区

切出来的影像

操作提示：

- 算术运算：**Interpreter -> Utilities -> Operators**
- 函数运算：**Interpreter -> Utilities -> Functions**
- 波段融合：**Interpreter -> Utilities -> Layer Stack**
- 矢量融合：用**MapInfo**切割**Split**，然后**SQL**查询统计
- 矢栅叠加：**Interpreter -> GIS Analysis -> Overlay**
- 矢切栅格：矢量转**AOI**，**Inquire Box -> Fit to AOI**
Interpreter -> Utilities -> Subset
(Form Inquire Box , AOI)

实验二、GPS 与 GIS 数据集成

一、GPS 数据的转换

现有一 GPS 测点数据，名为 CCGS.TXT，包括每测点的 (x, y) 坐标，坐标采用经纬度投影，坐标单位为度。

1. 转换成 TAB 格式

在 Mapinfo 中直接打开 TXT 格式文件生成 CCGS.TAB。

2. 转换成 MIF 格式

在文本编辑器中将 TXT 格式文件补充生成 GS.MIF。

_COL1	_COL2
111.722	28.9202
111.756	28.8989
111.895	28.819
112.04	28.7481
112.062	28.7378
112.075	28.7346
112.1	28.733
112.118	28.7337
112.139	28.7351
112.151	28.736
112.162	28.7336
112.175	28.7301
112.195	28.7242
112.217	28.7182
112.246	28.7104
112.253	28.7075
112.275	28.6979
112.293	28.6865
112.311	28.6712
112.327	28.6559
112.34	28.6425
112.351	28.6295
112.352	28.6257
112.355	28.6212
112.361	28.6126
112.481	28.4765
112.479	28.4799

```

Version 300
Charset "WindowsSimpChinese"
Delimiter ","
CoordSys Earth Projection 1, 104
Columns 1
  ID Integer
Data

Pline 25
111.7224847 28.92018032
111.7555641 28.89888105
111.8949067 28.81903247
112.0403444 28.74812033
112.0615685 28.73778773
112.0748643 28.73455419
112.1000246 28.73304829
112.1183947 28.7336797
112.1388465 28.7350732
112.1507043 28.7360156
112.1622287 28.7335707
112.1746292 28.73007441
112.1947016 28.72416311
112.2168444 28.71823027
112.246365 28.71040117
112.2534367 28.70746814
112.2746399 28.6978849
112.2934487 28.68649642
112.3110207 28.67120694
112.3268121 28.65593271
112.3396634 28.6425107
112.3505755 28.62945445
112.3524681 28.62569893
112.3553694 28.62123811
112.3611748 28.61257696
  Pen (1,2,0)
Pline 26
112.4811501 28.4765418
112.4794249 28.4799496
112.4756934 28.48781097
112.4728042 28.49253479
112.4678719 28.49910549
112.4626435 28.50567921
  
```

CCGS.TXT GS.MIF

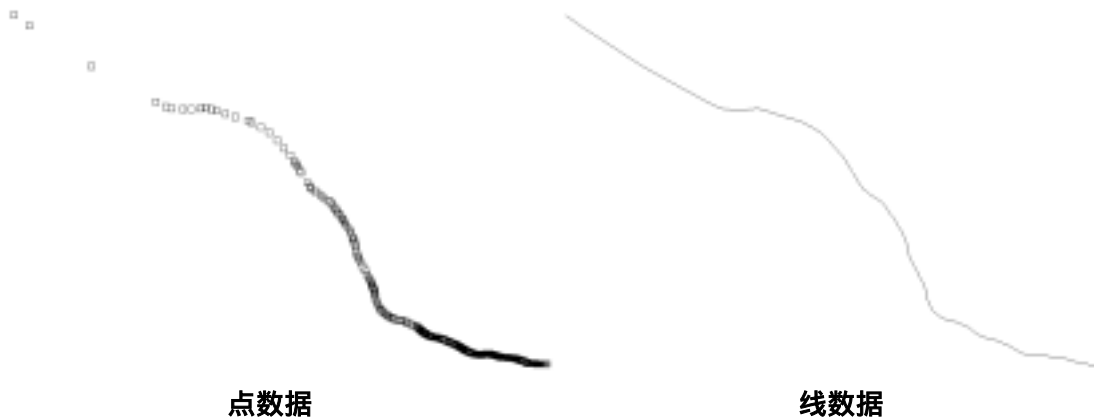
二、由 GPS 数据生成 GIS 数据

1. 生成点数据文件

在 Mapinfo 中为 CCGS.TAB 创建点。

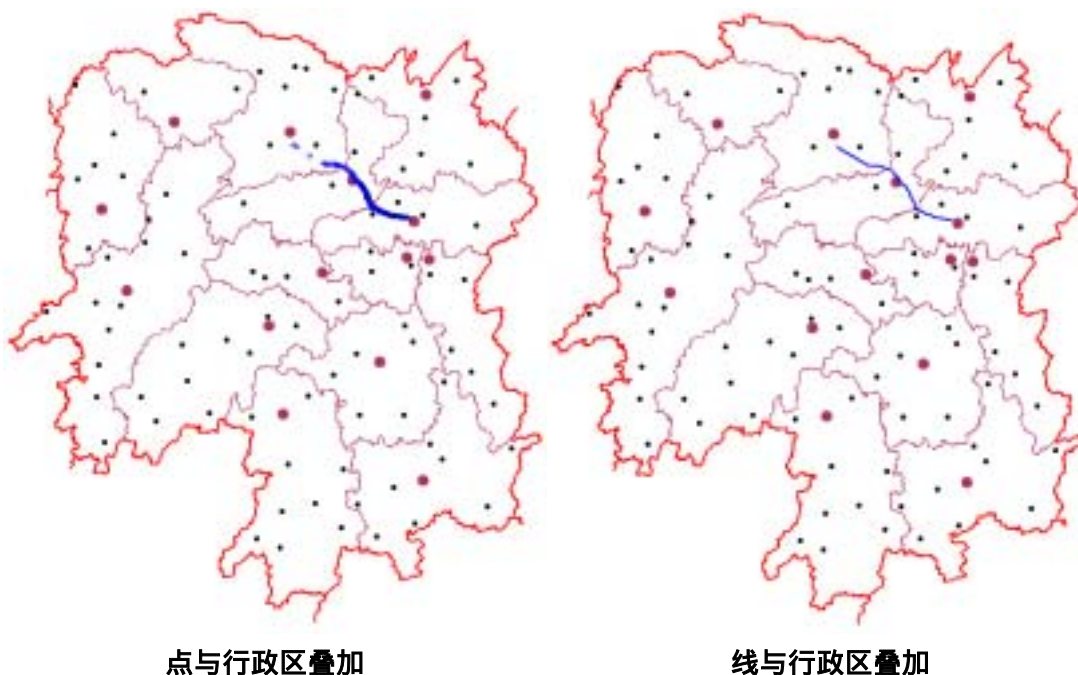
2. 生成线数据文件

在 Mapinfo 中为将 GS.MIF 转换成 TAB 格式。



三、GPS 数据与 GIS 数据的集成

将生成的点、线数据叠加在其它数据之上，这样由 GPS 获得的数据经转换后，完全可以与其它数据进行叠加显示与分析。



操作提示：

- 文本文件：File -> Open Table (Delimited ASCII)
Table -> Create Points
- MIF格式：以坐标的突变分段，线：Pline
Table -> Import
- 数据叠加：需统一投影，转入E00时选择

实验三、GPS 与 RS 数据集成

一、GPS 数据的转换

现有用于 RS 数据的分类训练区取样的 GPS 测点数据 PP.TXT、多边形数据 PL.TXT，包括每测点的 (x, y) 坐标，坐标采用经纬度投影，坐标单位为度。

1. 转换成 TAB 格式

在 Mapinfo 中直接打开 PP.TXT 格式文件生成 PP.TAB。

2. 转换成 MIF 格式

在文本编辑器中将 PL.TXT 格式文件补充生成 PL.MIF。

二、由 GPS 数据生成 RS 能接受的数据

1. 生成点数据文件

在 Mapinfo 中为 PP.TAB 创建点。

2. 生成线数据文件

在 Mapinfo 中为将 PL.MIF 转换成 TAB 格式。

3. 转换成 E00 格式

在 Mapinfo 中为将 PP.TAB、PL.TAB 转换成 E00 格式。

4. 矢量文件再生成

在 RS 软件中将 E00 格式转换成能接受的矢量格式。

三、GPS 数据与 RS 数据的集成

1. 获取点训练区 RS 数据值

将 PP 点文件与 tmCZT 遥感数据叠加，通过放大获得每个点所在的格网，从而获得每点所在格网不同波段的数据值，并以列表方式列出。

2. 获取区训练区 RS 数据值

将 PL 区文件与 tmCZT 遥感数据叠加，可以获得再多边形范围内 RS 数据各波段的最大值、最小值、均值、范围与标准偏差。

操作提示：

- 文本文件：File -> Open Table (Delimited ASCII)
Table -> Create Points
- MIF格式：区：Region，以首尾坐标重复为特征
Table -> Import
- 数据叠加：需统一投影，数据转入后添加
Viewer -> Utility -> Layer info
Edit -> Add coverage projections
- 点数据值：Inquire Cursor；Arrang Layers
- 区数据值：Vector -> Zonal Attributes，结果存于矢量

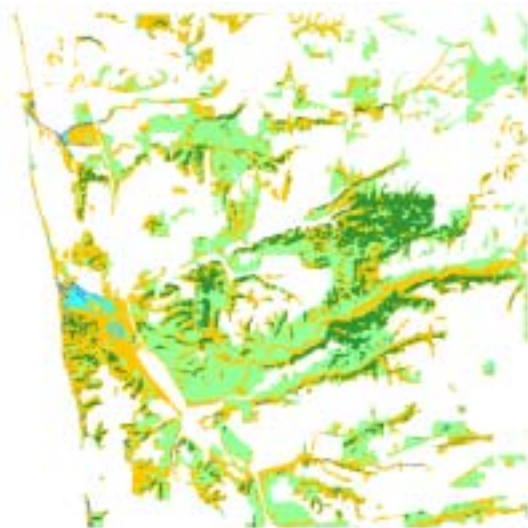
实验四、GIS 与 RS 数据集成

一、GIS 与 RS 数据的相互转换

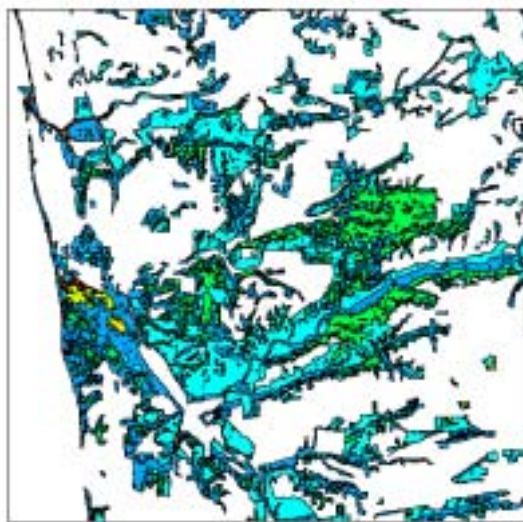
1. RS 数据转 GIS 数据

将 RS 分类或解译结果由栅格转换成矢量数据。

如将 LandCover 数据由栅格转换成矢量。



栅格数据

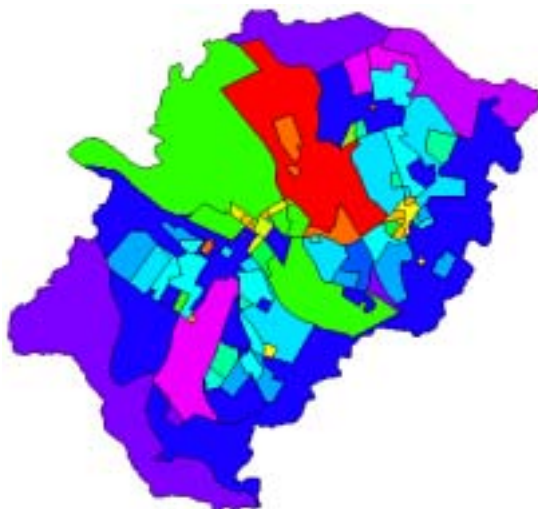


矢量数据

2. GIS 数据转 RS 数据

将 GIS 数据由矢量转换成栅格参与 RS 处理与分析。

如将 zone88 数据由矢量转换成栅格。



矢量数据



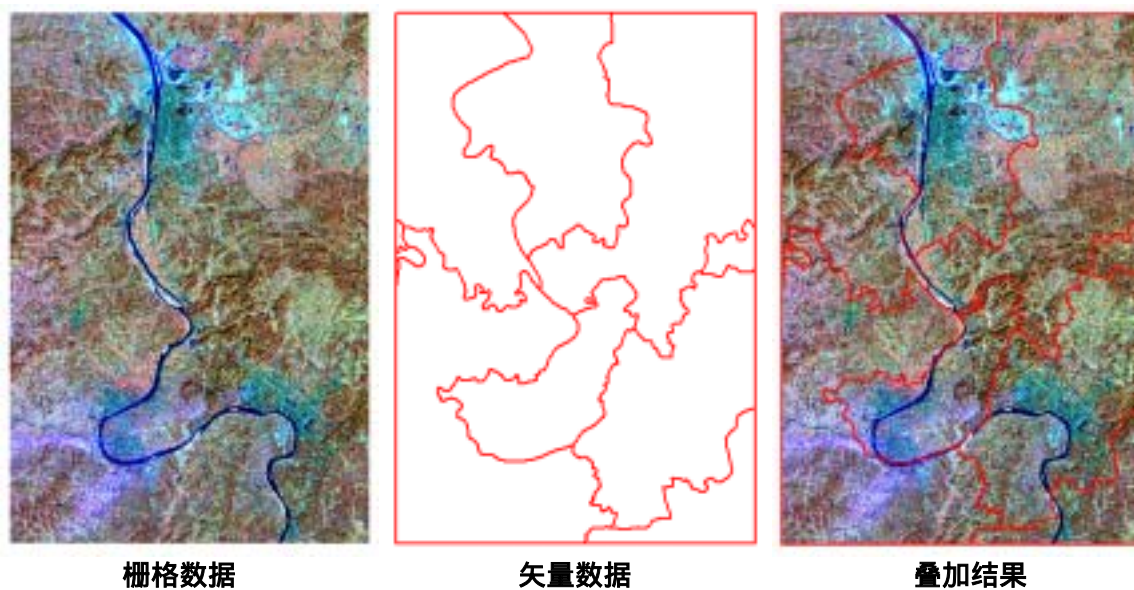
栅格数据

二、在 RS 中实现 GIS 与 RS 集成

1. 叠加显示

可以将具有投影的矢量与栅格数据在 RS 软件中叠加显示。

如将 tmCZT 与 Region_x 叠加显示。



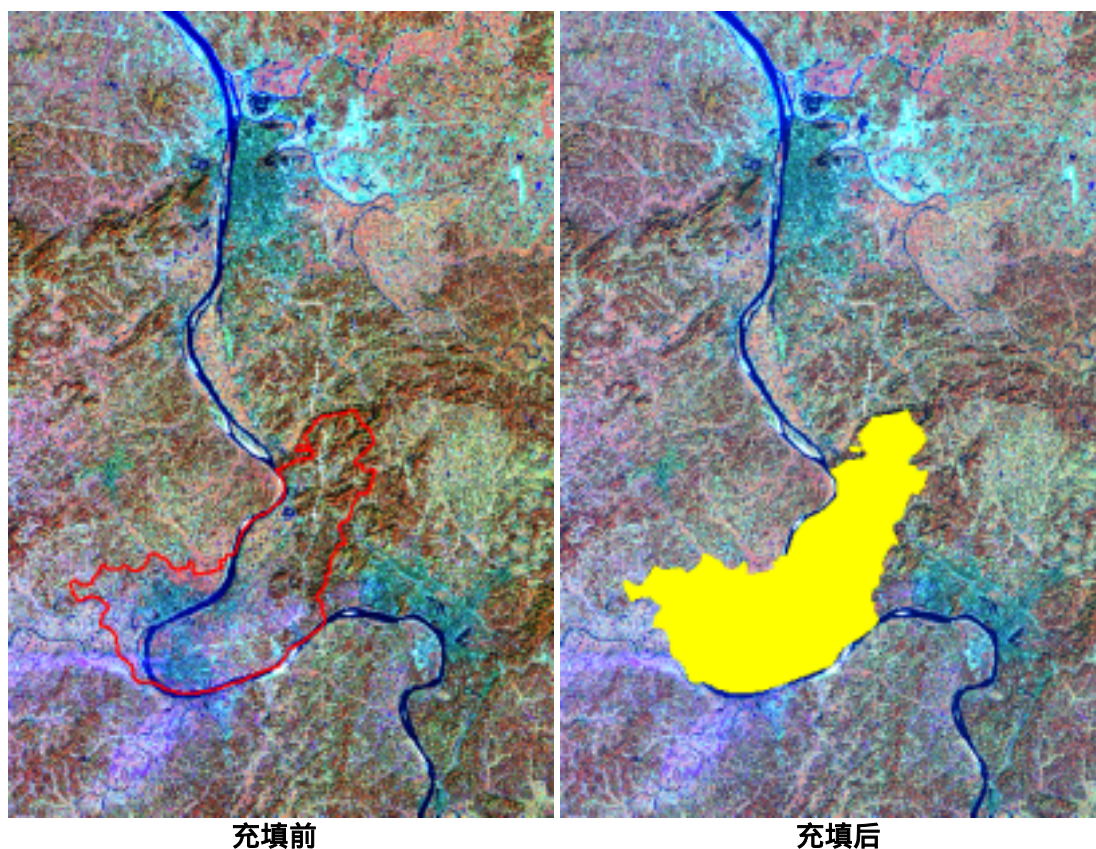
2. 利用 GIS 数据处理 RS 数据

a. 利用 GIS 数据切割 RS 数据

如用 XT 切 cbCZT2 切出不规则栅格图像（见实验一）。

b. 利用 GIS 数据对 RS 数据进行掩膜或充填处理

如用 XT 数据将 tmCZT 图像行政区边界内充填为黄色。

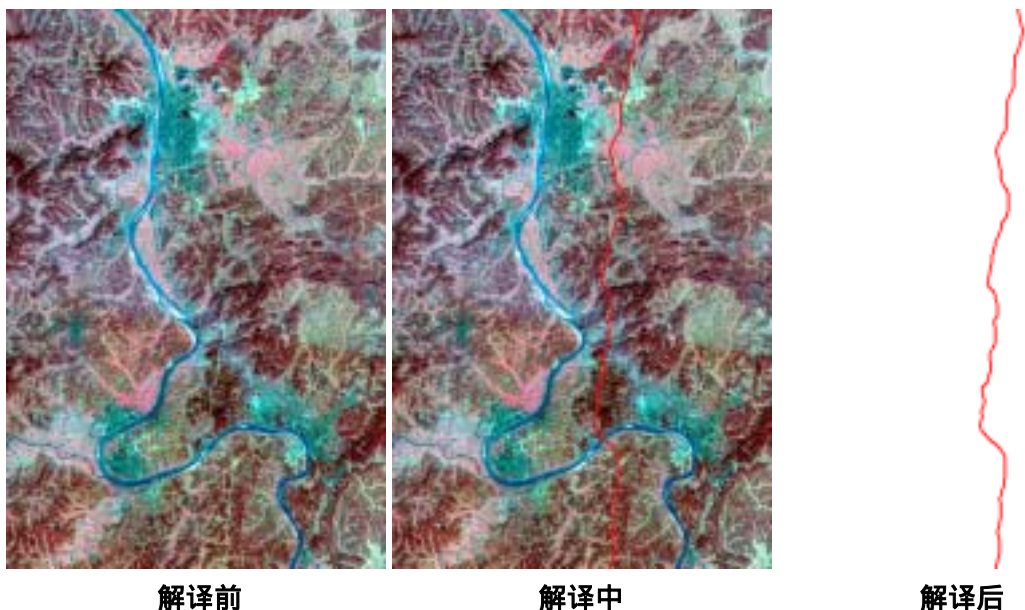


3. 直接由 RS 处理结果生成 GIS 数据

在 RS 数据解译或处理时直接将结果在集成环境中生成 GIS 数据。

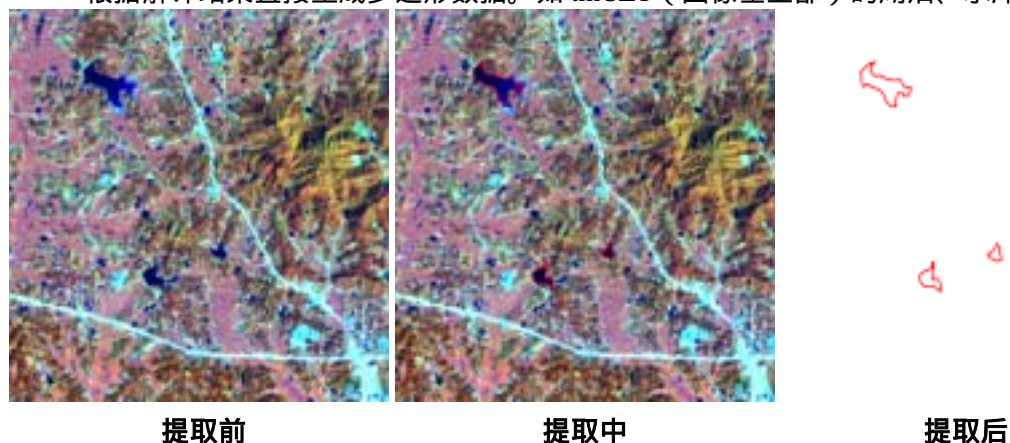
a. 直接绘制矢量数据

直接在集成环境中根据 RS 数据绘制所需的点、线数据。如 cbCZT 高速公路的解译。



b. 由处理结果生成矢量数据

根据解译结果直接生成多边形数据。如 tmCZT (图像左上部) 的湖泊、水库提取。



三、在 GIS 中实现 GIS 与 RS 集成

限于现在的技术条件，在 GIS 中一般只能实现 GIS 与 RS 数据的叠加显示。

如将 tmCZT 与 Region_x 叠加在 Mapinfo、Arcinfo 中显示。

操作指令：

- 栅格转矢量：Vector -> Raster To Vector
- 矢量转栅格：Vector -> Vector To Raster
- 矢量切栅格：打开矢量与栅格，矢量转AOI (AOI->Tools )
Inquire Box -> Fit to AOI
Interpreter -> Utilities -> Subset
(Form Inquire Box , AOI)
- 矢量掩栅格：打开矢量与栅格，矢量转AOI，Raster->Tools 
- 栅格绘矢量：打开栅格，新建矢量，绘折线
- 栅格提矢量：打开栅格，新建矢量，Raster->Tools  
Vector->Tools 

实验五、GPS 与 GIS 应用集成系统的设计

一、总体目标

以编程实现实验二中 GPS 数据在 GIS 中的集成。

二、技术思路

1. 实现背景数据 (region_s) 的显示；
2. 读入点数据至内存数组 (或链表)；
3. 从一端开始，对点数据进行排序，使点坐标具有顺序；
4. 从一端开始动态演示 GPS 移动的轨迹；
5. 最后以点坐标生成新的线数据，并保存。

三、编程要求

1. 采用组件式开发；
2. 以功能实现为主要目的；
3. GIS 平台不限、开发工具不限。

编程提示：

- 读入文件：定义文件变量，读入浮点数据，存入数组
- 投影转换：将数组中的地理坐标转换成投影坐标
- 坐标排序：将数组中的坐标 (x, y) 从小到大排列
- 轨迹跟踪：依次闪烁各点，并添加地理事件
- 数据存储：创建点集 (Points) 记录各点坐标
将点集赋给一个Parts
将Parts赋给一个Line
新建DataConnection，再AddGeoDataset
将GeoDataset赋给一个新建的MapLayer
获得MapLayer的Recordset，再AddNew
将Line赋给Recordset的Shape字段
Update将Recordset保存

实验六、RS 与 GIS 应用集成系统的设计

一、总体目标

以编程实现 RS 数据 tmCZT 在 GIS 中的集成显示。

二、技术思路

1. 使 RS、GIS 数据都应具有相同的投影、坐标体系；
2. 每一 RS 数据作为集成系统一层；
3. RS 数据置于所有数据的底层；
4. 当要显示底层的 RS 数据时，上层的 GIS 数据应透明或关闭；
5. 动态获得影像层上鼠标位置的数据值（RGB）。（选做）

三、编程要求

1. 采用组件式开发；
2. 以功能实现为主要目的；
3. GIS 平台不限、开发工具不限。

编程提示：

- 读入影像：定义 ImageLayer，设置 File
- 数据透明：将影像置于底层
上层多边形数据关闭
上层多边形数据设为透明（moTransparentFill）
- 点数据值：影像转为位图，并加入内存
根据影像层范围建立坐标与行列的对应关系
根据鼠标点的投影坐标确定对应的行列
根据行列从位图获得对应点的颜色值
将颜色值分解为 RGB