⾼光谱图像处理之⽬标检测技术（CEM算法）（图像处理）

⾼光谱图像处理之⽬标检测技术

⼀、⾼光谱图像处理之⽬标检测

1、⾼光谱图像⽬标检测的发展趋势和研究现状：

20世纪80年代末，美国的⼀些研究机构开始利⽤⾼光谱图像数据进⾏⽬标检测⽅⾯的研究。⾃上世纪九⼗年代，国外出现了进⾏⾼光谱图像⽬标检测算法理论研究的研究组。由Reed和Yu提出了基于⼴义似然⽐检验的恒虚警RX 检测器(RXD)。Chang课题组提出了基于正交⼦空间投影的OSP检测⽅法， Harsanyi提出了基于约束能量最⼩化的CEM算法。未来⾼光谱⽬标检测的发展将会越来越重视实⽤性，算法的性能将会进⼀步提⾼，同时更也加适合使⽤FPGA硬件对其进⾏加速从⽽具有更⾼的实际应⽤价值。

2、⾼光谱⽬标检测技术的应⽤范围：

⾼光谱⽬标检测具有较强的实⽤性，可应⽤于公共安全、环境检测、城市规划、⾷品卫⽣、地质岩矿的识别等众多⽅⾯。

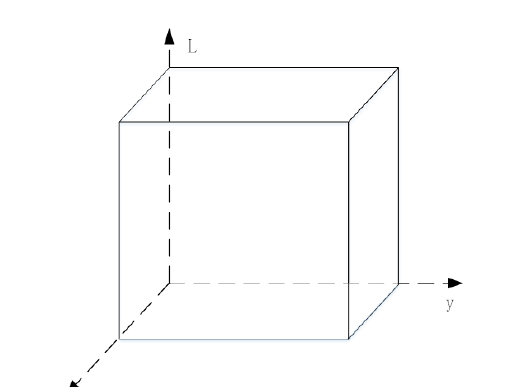


图1.1 ⾼光谱图像数据结构

3、⾼光谱图像数据的特点：

⾼光谱图像数据“图谱合⼀”,具有丰富的光谱维信息。⾼光谱图像数据共有三个维度，如图2.1所⽰，其中，图像空间维信息x、y⽤于表⽰物体的实际空间分布，⽽光谱波段L⽤于表⽰每个像素的光谱属性。

4、⾼光谱图像⽬标检测原理：

⾼光谱图像的各波段在成像范围内都是连续成像，因此⾼光谱的光谱曲线⼀般是平滑的、连续的曲线。⾼光谱图像的波段L中涵盖了物质的光谱信息，⽽每种物质的光谱信息都不⼀样，我们可以利⽤图像像素的光谱波段L所包含的特定的光谱信息来判断该像素所代表的特定的物质种类。

5、⾼光谱图像⽬标检测流程：



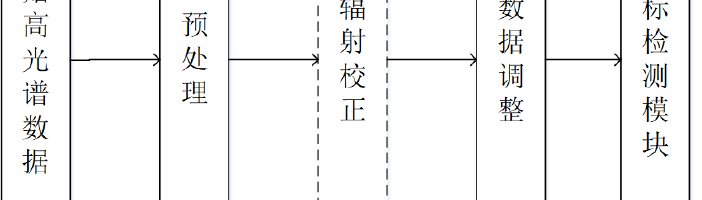


图1.2 ⾼光谱图像⽬标检测流程图

如图所⽰，对于拍摄得到的原始⾼光谱图像数据，需要先对数据进⾏预处理，包括数据格式化、⽆⽤数据剔除以及亮度到反射率的转化等。同时，对于遥感仪拍摄的⾼光谱图像还需要进⾏辐射校正，在⽬标检测前，应对数据进⾏调整，包括数据归⼀化等。最后根据已知的先验信息选择相应的⽬标检测算法进⾏检测。

6、现场可编程门阵列（FPGA）在⾼光谱图像处理中的应⽤：

⼀个FPGA可以⼤致定义为⼀系列互连的逻辑块。这些器件的主要优点之⼀是，为了实现不同的组合或时序逻辑功能，可以根据需要多次重新配置逻辑块及其互连。这⼀特性为FPGA提供了软件和硬件系统的优势。因为FPGA具有⽐专⽤集成电路（ASIC）更多的灵活性和更短的开发时间，更接近GPU提供的性能，但同时功耗要⽐GPU低得多。FPGA供应商通过改进FPGA体系结构（包括优化的硬件模块）并利⽤最新的硅技术不断提升FPGA的功耗和能效。可重构性、低功耗的特点以及FPGA对空间电离辐射耐受性的提⾼，这些因素已经使得FPGA成为⽬前板载⾼光谱遥感的最佳选择之⼀。

⼆、算法分类、⽐较和选择：

1、⽅法分类：

⾼光谱⽬标检测⽅法按照先验信息是否已知分为监督⽅法和⾮监督⽅法。前者⽤于⽬标光谱已知的情况下，利⽤⽬标光谱与图像像元光谱进⾏匹配，从⽽完成⽬标检测，⽐如CEM算法、OSP算法；后者多⽤于异常⽬标检测，⼀般不需要⽬标和背景的先验信息，根据⾼光谱图像数据获取⽬标检测所需要的数据，然后根据数据的⼤⼩来判断是否为异常⽬标，⽐如RXD算法。

2、CEM、OSP、RXD算法的区别：

（1）CEM（Constrained Energy Minimization）算法：

CEM算法主要思想是设计⼀个FIR线性滤波器，使得在满⾜约束条件式的情况下滤波器的输出能量最⼩。该算法不需要图像的背景信息，只需要知道要检测的先验光谱信息（⽬标向量）即可，具体⽅法是通过⾼光谱图像数据和先验已知的待检测⽬标确定⼀个滤波向量，让图像经过该滤波向量即可得到检测结果，其中滤波向量的作⽤是滤除图像中的⾮⽬标像素，让感兴趣的⽬标能够通过，同时抑制由其他信号带来的滤波器输出能量。

（2）OSP（Orthogonal Subspace Projection）算法：

OSP算法与CEM算法相⽐，最⼤区别在于不仅需要⽬标的先验知识，还需要图像中背景的先验知识，但在实际中中这些先验信息很难全部得到。在⾼光谱检测中我们⼀般⽤其来检测异常。该算法需要前提条件：图像信息、⽬标像元、⾮⽬标像元（异常⽬标）信息。

（3）RXD(Reed-Xiaoli Detector)算法：

RXD算法是异常⽬标检测领域中最基础的算法，不需要⽬标光谱的先验知识，⽽是基于背景服从多元正态分布的假设，通过检测与背景分布中⼼相⽐属于异常像元，并在这些感兴趣区域进⼀步查找可能存在的⽬标。该算法主要针对的是⼩⽬标检测问题。

（4）确定所采⽤的⽬标检测算法：

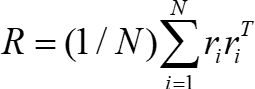
由于我们的应⽤场景多为有特定⽬标的⽬标检测，CEM正是针对未知场景中可能存在的特定⽬标的检测，只需要知道⽬标的光谱信息即可，⽽RXD算法适应于对特定场景的异常（未知⽬标）检测，⽽OSP算法除了需要已知⽬标光谱还需要背景信息。综上，我决定采⽤CEM算法进⾏⾼光谱⽬标检测的实现。

三、CEM算法分析：

1、算法步骤：

（1）对⾼光谱图像进⾏预处理，得到⼆维化和归⼀化后的数据r（L\*N）；

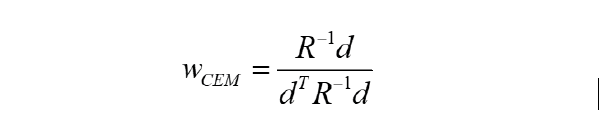
（2）根据图像数据r，求得图像的⾃相关矩阵：



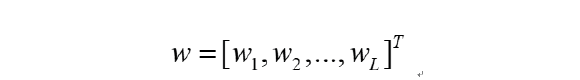
（3）确定⽬标光谱向量d（d⼤⼩为L\*1）；

（4）根据公式：

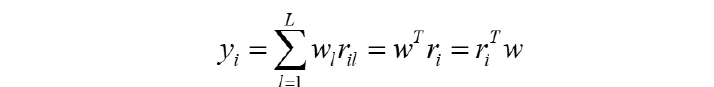
（4）根据公式：



设计FIR线性滤波器：



（5）将归⼀化后的数据经过FIR滤波器，得到输出信号y。



2、问题分解：

CEM算法的实现可分为三部分：⾃相关矩阵、矩阵求逆、线性FIR滤波器。

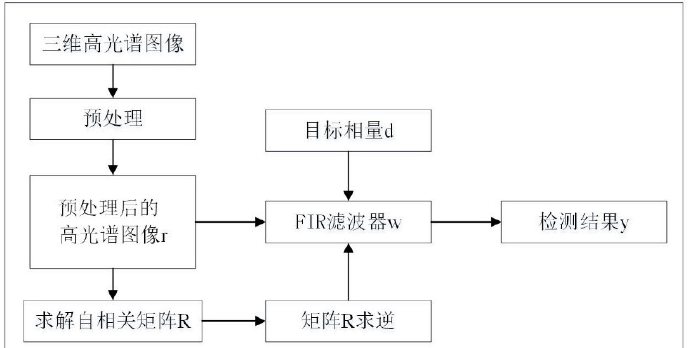


图 1.3 CEM算法分解流程图

如图所⽰，⾸先根据⾼光谱图像r求得⾃相关矩阵，再利⽤矩阵求逆模块求得⾃相关矩阵的逆矩阵，结合从光谱库获取到的⽬标向量的先验信息求得FIR滤波器的滤波向量，最后将⾼光谱图像r通过FIR滤波器即可得到最终的检测结果。

四、CEM算法实现：

1、算法流程：

在MATLAB和C语⾔中实现CEM算法的具体流程如图所⽰，因为语⾔特性是串⾏执⾏命令，所以在编写程序时与硬件设计⽐较更加直接明了。



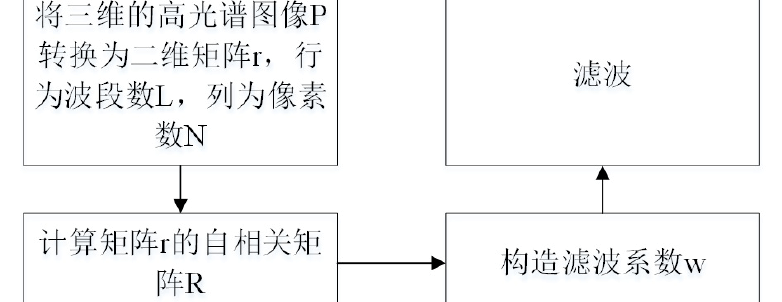


图1.4 CEM算法流程图

2、数据预处理：

对前期得到的⾼光谱图像在MATLAB平台上进⾏预处理。这⼀过程主要对原始的200\*200\*189⼤⼩的⾼光谱图像进⾏操作：

（1）⼆维化：通过调⽤MATLAB⾥⾯的reshape（）函数实现。

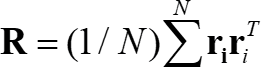
（2）归⼀化：采⽤“min-max”⽅法。具体步骤是先找到数据的最⼤值

（maxA）和最⼩值（minA），通过 y = (x-minA) /(maxA-minA) 计算的 y 即为归⼀化后的数据。⾼光谱数据量⼤，也造成了数据的存储⽐较困难，数据在程序中的存储等处理

3、CEM算法的MATLAB实现：

⾸先，根据 2 对⾼光谱图像数据进⾏预处理，⽽后求其⾃相关矩阵。⾼光谱图像数据的⼤⼩为200\*200\*189，其中200\*200为像素

数，189为波段数。



⾸先⽤reshape()函数和transpose()函数将200\*200\*189的三维⾼光谱图像数据转为189\*40000的⼆维矩阵，再⽤⾃⼰写的Normalize()函数对矩阵数据⽤“min-max”⽅法进⾏归⼀化，然后根据公式式利⽤MATLAB中的矩阵求转置函数transpose()和矩阵相乘操作得到⼀个189\*189的矩阵，最后对189\*189的矩阵除以像素数N即可得到⾃相关矩阵。

最后，根据先验已知的⽬标向量和⾃相关矩阵R的逆矩阵（在MATLAB中矩阵R求逆即为（1/R））求得FIR滤波器，将⾼光谱图像模拟数据通过FIR滤波器，即可得到最终的检测结果。（本次测试中选取的⽬标向量 d 是模拟图像中的 C 物质）

4、CEM算法的C语⾔实现：

在Visual Studio平台上完成CEM算法C程序的编写。相⽐于MATLAB实现，CEM算法的C语⾔实现主要难点在于矩阵运算。因为在C语⾔中，矩阵的转置、相乘、求逆等操作均没有现有的函数，需要编写相应的函数。在实现矩阵基本运算过程中,通过动态分配内存运⽤⼆维指针传递参数,完成矩阵加减法运算、矩阵转置运算、矩阵相乘运算以及矩阵求逆运算，这样可以节省存储空间，使⽤完后释放空间即可。

CEM算法的C语⾔实现主要包括如下步骤：

（1）在MATLAB中，将归⼀化后的⾼光谱图像数据转为189\*40000的⼆维形式，保存为CEM.mat；

（2）编写矩阵初始化、矩阵转置、矩阵相乘、矩阵求逆、内存释放函数；

（3）编写main函数，读CEM.mat，调⽤上述函数，求得FIR滤波器，进⽽获得输出信号y，将其写⼊CEM.txt；

（4）在MATLAB中显⽰CEM.txt中数据所代表的图像。

5、难点解决：

（1）矩阵转置

矩阵转置函数的输⼊是⼤⼩为m\*n的矩阵A，输出是⼤⼩n\*m为的矩阵B。采⽤嵌套的for循环分别遍历矩阵的⾏和列，将输⼊的⼆维矩阵A按列读出，重新按⾏写⼊新矩阵B中，即可实现矩阵的转置。

（2）矩阵相乘

矩阵相乘函数的输⼊是⼤⼩m\*n为的矩阵A和n\*k的矩阵B，输出是⼤⼩为m\*k的矩阵C。采⽤嵌套的for循环分别遍历矩阵的⾏和列，⾸先将矩阵A按⾏读出，矩阵B按列读出，然后将读出的矩阵A的第i⾏和矩阵B的第j列对应位相乘求和，将计算的结果写⼊矩阵C第i⾏第j列元素中。

（3）矩阵求逆——QR分解求逆

QR分解求逆的原理是：对于可逆矩阵A，⾸先利⽤QR分解将A矩阵分解为Q矩阵和R矩阵。即A=QR，其中Q是正交矩阵，R是上三⾓矩

阵。然后将公式左右同时求逆，可以求得A的逆矩阵，其中Q矩阵的逆矩阵和转置矩阵相同，R求逆有特定的公式。QR分解求逆的运算较为简单，且数据稳定度较⾼，可以得到误差较⼩的求逆结果。

6、算法实现结果和对⽐分析：

1）MATLAB和C语⾔实现CEM算法的检测结果如图所⽰：







（a）（b）（c）

图1.6 原图CEM算法检测结果：（a）原图（每⼀⾏是⼀种物质）（b）MATLAB检测C物质（c）C语⾔检测C物质

通过对MATLAB和C语⾔实现结果进⾏对⽐，发现⼆者均成功实现了CEM算法并完成了对⽬标的检测，观察检测结果基本⽆差异。