基于 FPGA 的数字航空影像特征提取及边缘检测算法实现

摘要：图像的边缘是图像最基本，也是最重要的特征之一。边缘检测一直是机器视觉和图像处理领域里面最经典的研究课题之一。这是对图像进行理解和分析的第一步。伴随着遥感技术的不断进步，尤其是数字航空图像的出现，边缘检测成为了用于数字航空图像中提取信息的关键的一步。边缘检测的目的在于去发掘在一幅图像中的形状和反射比或者透射比信息。检测结果的准确性和可靠性直接影响机器控制系统对客观世界的感知。本文中提出了一种基于 FPGA 的边缘检测算法。利用可编程逻辑门阵列（FPGA）作为平台来进行边缘检测算法的实施，好处在于 FPGA 具有大量的存储空间以及内嵌的乘法器。FPGAs 提供了一个在特定应用硬件上处理实时算法的平台，其性能显著优于数字信号处理器(DSPs)。本文提出的结构可作为航空图像系统用于导航或者用于模式识别。并给出了 Sobel 算子和 Prewitt 算子的硬件实现结果。

关键词：边缘检测算法；一阶梯度运算；FPGA；Prewitt 算子；Sobel 算子

Ⅰ.介绍

近些年，伴随着经济的快速发展，城市化进程的加快，以及土地利用问题逐渐成为全社会关注的热点问题，建设用地资源日趋紧张，面临压力越来越大。因此，从高分辨率图像中提取当前建筑分布状况信息，并从中获得当前土地使用信息，为村镇土地资源的合理利用提供提供了保障。得益于当前遥感技术的快速发展，我们可以获得高清晰度的遥感图像以及数字航空图像。

航空数码相机的出现解决了传统成像生产效率低、实时成像质量差的问题[1]。与航空扫描数字图像相比，数字航空图像具有空间分辨率高、集合精度高、纹理信息丰富、实时性好、效率高等优点。基于数字航空影像的边缘提取技术对村庄土

地动态监测具有重要意义。在图像处理和机器视觉中。边缘检测是一个旨在捕捉图像中的重要特性的过程。这些特性包括物体的光度、几何和物理特性的不连续性。这些信息引起灰度图像的变化;最常用的出现的变化是不连续(台阶边缘)，局部极值(线条边缘)，以及形成的至少两条边相交的二维特征(节点)[2]。边缘检测的目的是定位这些变化并识别产生这些变化的物理现象。边缘检测的有效性、效率和完成后续处理阶段的可能性都依赖于边缘检测的高效性和可靠性。为了满足这一要求，边缘检测提供了图像的所有重要信息。这份论文的组织方式如下：第二部分讨论了边缘检测算法，第三部分给出了该模型的硬件结构。紧跟第五节给出了结论，第六节给出了不同边缘检测方法的实现细节和结果。

Ⅱ.边缘检测算法

边缘检测可以检测区分出物体的轮廓以及物体与图像中的背景之间的边界。一种边缘检测滤波器也可以用于改善视频流的模糊或者用于抗锯齿。基本的边缘检测算子是一个矩阵区域梯度运算，它决定了不同像素之间的方差水平。基于梯度的边缘检测器的例子有 Roberts、Prewitt 和 Sobel 算子。所有基于梯度的算法都有核算子来计算相互垂直和水平方向上的边缘强度。然后将边缘各组成部分的贡献进行综合，得到边缘强度的总值。本文给出了 Prewitt 边缘检测算法和 Sobel 边缘检测算法的硬件实现。

该算法采用二维一阶导数算子对灰度图像进行二维一阶导数，以突出图像中具有高空间一阶导数的区域。在图像的梯度大小中，边缘被转换成脊。该算法沿着这些脊线的顶部跟踪，并将实际上不在脊线顶部的所有像素设置为零，并在输出中给出一条细线。图像的边缘检测是图像像素与不同掩膜的卷积，这就导致了对水平梯度和垂直梯度的计算。使用相邻像素之间的差异计算两个梯度。查找边的一

种方法是使用 Prewitt 内核。Prewitt 核函数基于中心差分的思想，通过以下一阶空间导数表示:

$$\frac{\partial I}{\partial x} \approx \frac{I(x+1,y)-I(x-1,y)}{2}$$

$$\frac{\partial I}{\partial y} = \frac{I(x,y+1)-I(x,y-1)}{2}$$

这两个导子对应于由{-1,0，+1}表示的水平卷积和{-1,0,1}表示的垂直卷积组成的卷积核。这些卷积应用于灰度图像，得到水平(Kx)和垂直梯度(Ky)。

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

(Kx)

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

(Ky)

图 1Prewitt 卷积核

为了对整个图像进行卷积，这个想法包括构建一个 n x n(通常是 3x3)称为内掩码的数字矩阵，并将它与同一维度图像的一部分相乘。然后对所有产品进行求和，以确定中心像素值。使用 Prewitt 系数的内核掩模将是(Kx)中的结果用于垂直边缘检测的矩阵，以及(Ky)中的结果用于水平边缘检测的矩阵。卷积的结果将保持在由像素分辨率定义的位数范围内，因此，如果像素分辨率为 8 位，结果的范围将是[0:255]。如果生成的像素超出范围，则必须对其进行标准化。

Sobel 算法在单个核中提供了差分和噪声平滑运算。因此，该算法可以避免第一梯度运算的噪声敏感性。给出了 Sobel 算法的纵向和横向核函数。

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

(Kx)

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

(Ky)

图 2 Sobel 卷积核

在当前像素点上可以用核 Kx 计算垂直分量，用核 Ky 计算水平分量，用|Kx|+|Ky|给出梯度强度的指标。通过使用阈值，可以减少杂散噪声对特征提取的影响。由于增加了 Sobel 算子的平滑性，与 Roberts Cross 和 Prewitt 算子的输出相比，Sobel 算子的边缘线变得更加粗。 Sobel 和 Prewitt 算子在图像处理中得到了广泛的应用，例如在[4]Sobel 边缘检测是图像处理和立体渲染的一个重要应用。一个基于 0.18umCMOS 技术的 Sobel 处理器作为实时剪切翘曲因子分解/立体渲染系统的一部分被构建来计算梯度。在文献[5]中，提出了一种将 Sobel 技术与遗传算法相结合的图像版权保护方案。针对文献中基于 Sobel 版权保护方案，利用 Sobel 技术检测封面图像的边缘，即版权图像。提出了一种将 Sobel 边缘检测算子和软阈值小波去噪相结合的方法对含有高斯白噪声的图像进行边缘检测。在文献[7]中提出了一种基于区域生长的目标圆实时检测和跟踪算法。该算法包括边缘检测、区域生长、可能的圆心估计和依次精确定位的圆心。该算法利用 Prewitt 算子构造图像边缘，并采用区域生长的方法将形成的边缘划分为不同的连通子区域。 实际上，边缘检测算子是基本傅里叶滤波器的二维扩展。FPGA 结构允许用并行结构实现算法。基于 FPGA 的查找表 SRAM 可以方便地利用分布式算法(DA)技术实现数字滤波器或其他重要的数字信号处理。考虑到现场可编程门阵列(FPGA)的优点以及 Sobel 和 Prewitt 算

法的相关性，重点研究了上述两种算法的 FPGA 实现。

Ⅲ.边缘检测的硬件结构

硬件结构如图 3 所示。目标体系结构包括四个主要部分。首先，输入像素通过图像延迟线。图像延迟线通过行缓冲区移动传入图像像素来创建延迟线。缓冲区深度取决于每行的像素数。缓冲线的数量取决于卷积核的大小。像素被转发到垂直和水平的内核(图 1 和图 2)。内核对像素进行卷积运算，并将运算结果转发给组合块。这块结合了水平和垂直卷积的结果。这个块的最终输出是总和水平和垂直卷积结果的绝对值。

组合块的输出是边缘检测算法的输出，不需要使用阈值。因此它由一些虚假的噪音组成。通过控制阈值块中的阈值，可以降低噪声的影响。最后从阈值块中提取输出像素。

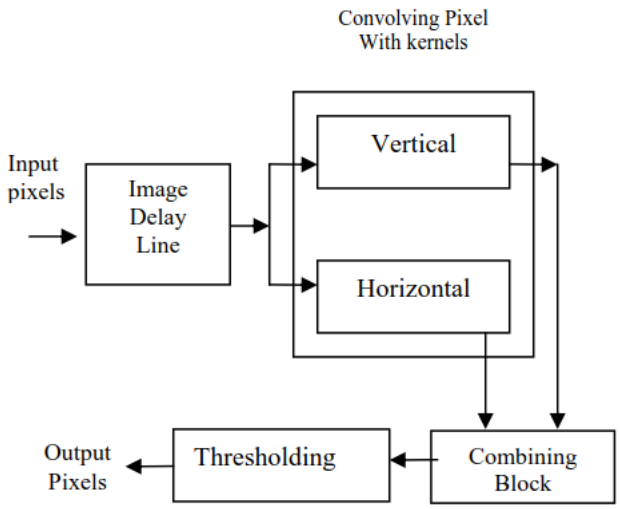

图 3 系统的硬件结构

Ⅳ.具体实验细节和结果分析

对于 FPGA 的实现，本文采用了高级综合的方法。利用系统生成器对该系统进行了设计、实现和验证。SystemGenerator 是一个来自 Xilinx 的 DSP 设计工具，它支持使用基于 MathWorks 模型的 Simulink 设计环境进行 FPGA 设

计 。使用系统生成器时，不 需 要 具 备 使 用 Xilinx FPGA 或 RTL 设计 方法的经验。在 DS 友好的 Simulink 建模环境中使用 Xilinx 特定的块集捕获设计。所有下游的 FPGA 实现步骤，包括综合、放置和路由都自动执行，以生成 FPGA 编程文件[9]。

首先利用 Matlab 将彩色航空图像转换为灰度图像，将二维图像转换为一维数组。此值从工作空间传递到系统生成器模型。图 4 显示了 640X480 分辨率图像的图像延迟线的系统生成器模型。在第一和第二以及第二和第三延迟线输出之间有 640 单元的延迟。第一和第三延迟线有两个附加的单位延迟，这使得 3X3 的像素矩阵可用于与 Prewitt 或 Sobel 核的卷积。

图 5 和图 6 显示 Prewitt 过滤器的垂直和水平内核。如图 1 所示，目前在 Prewitt 内核的值是 1,-1 和 0，因此这种结构不需要任何乘法器。Sobel 算子的核有 2 和-2 作为值之一。乘以二可以通过简单的右移操作来实现。可以看出，只有在这些内核中使用的数字块是加法器减树、比较器和多路复用器。这两个块的输出分别是：11 位的情况下 Prewitt 边缘检测和 12 位的情况下 Sobel 边缘检测。

输出的垂直和水平内核是绝对值。如图 6 所示。这些值传递给组合块，在 Prewitt 边缘检测的情况下是 11 位 ，在 Sobel 边缘检测的情况下是 12 位 。使 用 System generator 的 Slice 块，前 8 位 MSB 的输出被截断。阈值单位是如图所示的方框中所示。它由比较器和多路复用器组成。只要组合单元的输出大于阈值 255，就生成输出 else 0 作为输出。最终的结果存储在一个 Matlab 变量和脚本的帮助下在 matlab 环境中显示。本设计是在 Virtex-4 DSP 开发工具包上利用 xc4vsx35-10ff668FPGA 实现的。该设计的工作时钟

频率为 100 兆赫，能够在 0.003 秒内产生输出。

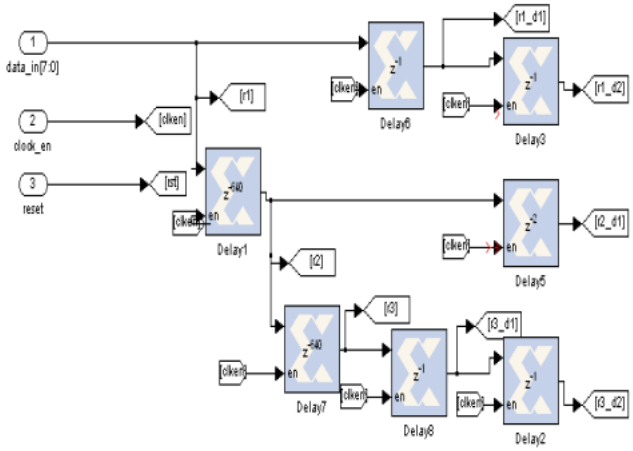

图 4 图像延迟线的实现

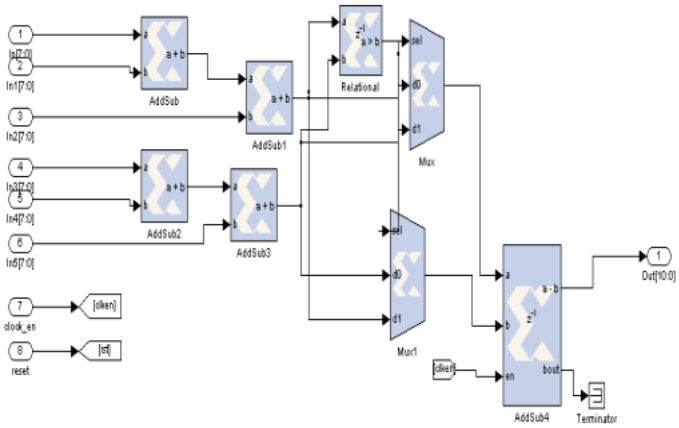

图 5 垂直内核的实现

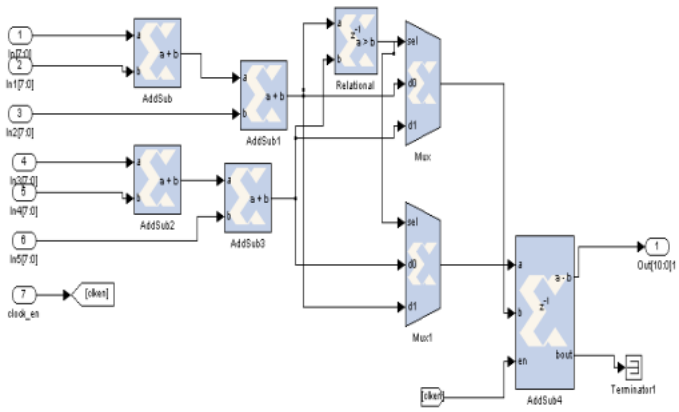

图 6 水平内核的实现

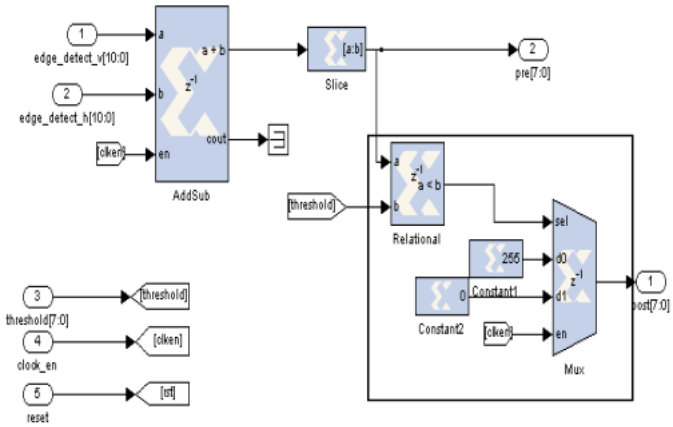

图 7 合成与阈值分割单元的实现

图 8 显示了原始彩色航空图像，选择其作为测试图像，其灰度转换图 9(a)显示了 prewitt 算子的输出前和 9(b)后的阈值应用。我们可以注意到，在后阈值输出没有杂散噪声。

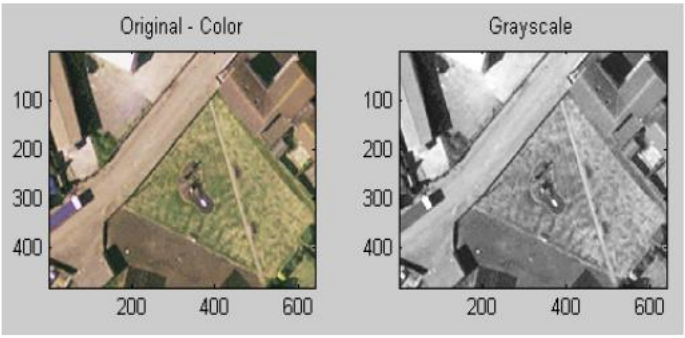

图 8 航空原始图像及其灰度化版本

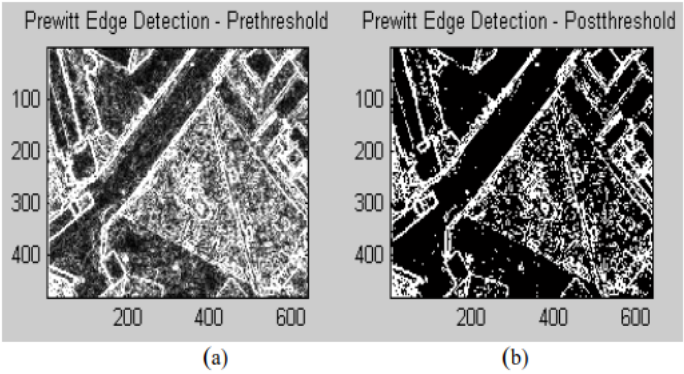

图 9 Prewitt 边缘检测器输出

图 10(a)显示了应用阈值前和应用阈值后 Prewitt 算子的输出。由于 Sobel 边缘检测器的平滑效果，我们可以注意到阈值后输出没有杂散噪声，而且边缘的厚

度比 Prewitt 边缘检测器更大。

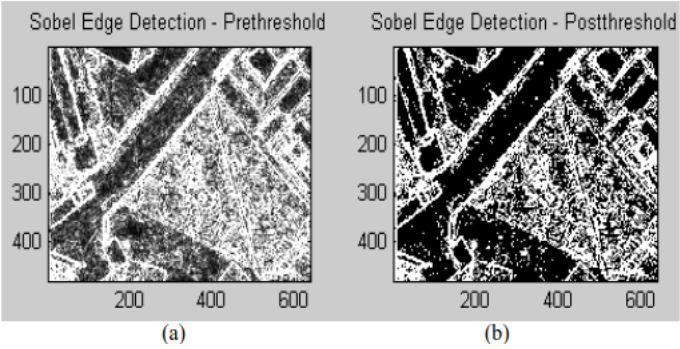

图 10 Sobel 边缘检测器输出

表一显示了系统资源利用率的概况。如图所示，该方法是无乘法器的，因此它消耗更少的面积和功率。查找表使用率较高的原因是因为延迟线的实现。

SN	Resource	Numbers
1.	Slice	451
2.	Flip flops	718
3.	LUT	850
4.	IOBs	58

表 1 系统资源利用率概况

Ⅴ.结论

本文在现场可编程门阵列(FPGA)上实现了两种不同的航空图像边缘检测方法。该方法对 FPGA 资源的要求较低，易于与航空监视仪器和自动导航设备集成。

参考文献

[1] X.L. Liu, F.Z. Duan 和 H.L. Gong，航空影像技术的成就与展望，前沿科学，2007(03):10~14

[2] D. Ziou 和 S. Tabbone，边缘检测技术-概 述，TR-195，数学与信息部，舍布鲁克大学，魁北克，1997 年，第 1-41 页

[3] J. F. Canny，一种边缘检测的计算方法，IEEE 模式识别与人工智能领域，8(6)：769-798，November 1986

[4] Kazakova, N.和 Margala, M.和 Durdle, N.G.,Sobel 边缘检测处理器的实时立体渲染系统”, 2004 年国际电路与系统研讨会论文集, 2004 年, ISCAS’ 04 卷:2 出版年份:2004，页:II-913-16

[5] Mn-Ta Lee 和 Shih-Syong Chen，应用 Sobel 技术和遗传算法的图像版权保护方案，2010 年计算机通信控制与自动化国际研讨会(3CA)，第一卷:出版年份:2010，第 201-204 页

[6] Wenshuo Gao，Xiaoguang Zhang，Lei Yang 和 Huizhong Liu，一个改进的索贝尔边缘检测，IEEE 第三届国际计算机科学与信息技术会议，2010 年卷:5 出版年份:2010 年，第 67-71 页

[7] Wang Yigang，Liu Yangguang，Wang Zhuoyuan，Fan Shengli 和 Cui Jialin，一种低复杂度、高性能的实时检测和跟踪孔内圆形的算法，第三届智能系统与知识工程国际会议，2008 年，ISKE 2008 年,卷:1 出版年份:2008，第 604-608 页

[8] Li Xue，Zhao Rongchun 和 Wang Qing,VCAS 中基于 FPGA 的 Sobel 算法作为车辆边缘检测器,《2003 年国际学报》神经网络与信号处理会议, 2003 年.卷:2 出版年份:2003 年,第二卷 第 1139-1142 页

[9] 系统生成器介绍 http://www.xilinx.com/tools/dsp.htm