

FENG Shuichun, WANG Zhipeng, YANG Jianjun, ZHOU Hai, BIAN Chunjiang, MENG Xin. An FPGA-implemented Method for Real-time Multi-dimensional Feature Extraction of Sequence Image Targets (in Chinese). *Chinese Journal of Space Science*, 2023, 43(3): 576–585. DOI:10.11728/cjss2023.03.2022–0014

一种 FPGA 实现的序列图像目标多维特征实时提取方法*

冯水春^{1,2,3} 王志鹏⁴ 杨建军^{1,2}
周海^{1,2} 卞春江^{1,2} 孟新^{1,2}

1(中国科学院国家空间科学中心 北京 100190)

2(中国科学院复杂航天系统电子信息技术重点实验室 北京 100190)

3(中国科学院大学 北京 100049)

4(哈尔滨工业大学航天学院 哈尔滨 150001)

摘要 目标区域多维特征的建模表征是实现目标跟踪搜索的前提。传统的特征提取方法通常先连通出目标区域,再计算目标特征,实时性不强。提出了一种基于像素的连通域标记及目标特征同步计算方法,建立特征传递机制,利用标号映射表关联标号表与特征属性表,并在区域邻接时进行标号合并和特征属性同步传递计算,保证了目标区域连通完成即可输出目标区域特征。给出一种基于 FPGA 硬件设计的实现方案。仿真测试结果显示,此方法具有以下几方面优势:连通域标记耗时少,仅为 $(L \times W + 2n/m) \times T_{\text{clk}}$, 接近理论最小值 $(L \times W) \times T_{\text{clk}}$; 利用循环缓存区存储图像,资源消耗小;标记与计算并行处理,提升了检测跟踪效率;多目标特征经测试验证结果准确,可有效支持后续的目标跟踪检测;具有理论与实用价值。

关键词 图像处理, 目标特征提取, 通域标记, 实时

中图分类号 TP75

An FPGA-implemented Method for Real-time Multi-dimensional Feature Extraction of Sequence Image Targets

FENG Shuichun^{1,2,3} WANG Zhipeng⁴ YANG Jianjun^{1,2}
ZHOU Hai^{1,2} BIAN Chunjiang^{1,2} MENG Xin^{1,2}

1(National Space Science Center, Chinese Academy of Sciences, Beijing 100190)

* 国防科技创新特区项目(1916318 ZT00900304)和中国科学院青年创新促进会项目(E0293401)共同资助

2022-04-29 收到原稿, 2022-12-16 收到修定稿

E-mail: fengshuichun@nssc.ac.cn

©The Author(s) 2023. This is an open access article under the CC-BY 4.0 License
(<https://creativecommons.org/licenses/by/4.0/>)

2(Laboratory of Electronic and Information Technology for Space Systems, Chinese Academy of Sciences, Beijing 100190)

3(University of Chinese Academy of Sciences, Beijing 100049)

4(School of Astronautics, Harbin Institute of Technology, Harbin 150001)

Abstract The prerequisite of target detection and tracking is to model and represent the target based on multi-dimensional features extracted from the target region. The traditional target feature extraction needs to connect the target region first and then must calculate the target feature, which has still room for improvement in real-time performance. An advantage of the new method based on the synchronous calculation of pixel-connected domain markers and target features is that the target features can be output when the target region is connected. This article proposes a method highlighting pixel-based marking of connection domains and synchronous computation of target features, which can output target features immediately after target areas are connected. By setting up a feature transfer mechanism, this method establishes a marker table, a marker mapping table and a feature attributes table while scanning images, and links the marker mapping table and the feature attributes table with the help of the marker table. When different areas are adjacent to one another, markers are consolidated while the features' attributes are synchronously transferred and computed, ensuring real-time extraction of the target features. Regarding real-time and multiple-dimensional extraction of the features of multiple targets in high-resolution distant sensing images, the article proposes an implementation plan based on FPGA hardware design. According to the results of a number of simulation tests, the method featuring high-speed marking of connection domains over one iteration of split pixels and real-time computing of target features boasts the following outstanding characteristics: The time consumed by marking connection domains is only $(L \times W + 2n/m) \times T_{\text{clk}}$, close to the theoretical minimum $(L \times W) \times T_{\text{clk}}$; saving images in cyclical buffers only takes up few resources; parallel pipeline processing of marking and computation improves the detection and tracking efficiency; the test results of multiple target features are accurate and consequently can effectively support subsequent target tracing detections; and the method also has both theoretical and practical values.

Key words Image processing, Target feature extraction, Connected domain marker, Real-time

0 引言

红外目标搜索跟踪系统由于具备远距离感知、被动探测、隐蔽性强等优点,被广泛应用于遥感成像、空间探测等领域。红外目标搜索跟踪系统通过对实时获取的红外序列图像进行目标检测,并对场景中多目标进行实时跟踪来实现对目标的精确定位,而红外图像的目标检测、分割与特征提取计算的精度与实时性关系到多目标跟踪的性能。可见,多目标特征的提取计算是目标检测跟踪中的重要一环。计算结果用于建立目标跟踪算法的目标特征模板,实现对目标的高精度实时跟踪。由于目标形态与目标辐射源的空间分布状态相关,导致像面目标的形态分布多样且不

确定,连通域标记方法要适应各种空间状态,往往需要消耗大量的硬件资源与运行时间。此外,特征提取应在目标区域明确后,基于目标分布区域计算目标的局部对比度特征。因此,目标区域标记及特征计算的实时性是影响后续目标跟踪时效性的关键。

经典的连通域标记方法是 Rosenfeld 等^[1]提出的两遍扫描法。现有的连通域标记方法多衍生于此,主要有基于游程标记和基于像素标记两大类。经典方法要扫描两次图像,实时性不佳,近年提出的改进思路主要从减少扫描图像的次数入手。例如基于游程的方法有 Wang 等^[2]、Ling^[3]、Zhang 等^[4]、Dai 等^[5]提出的基于游程的一次扫描连通域标记方法,先按行扫描将连续像素合并为一个游程,再通过游程连通完

成区域标记。但基于游程的连通要先提取游程信息, 再对游程和像素做合并处理, 不利于目标特征量的迭代及加速计算, 因此不适用于复杂目标特征的实时计算。基于像素的一次遍历图像的连通域标记方法有 Zhao^[6] 提出的多行同时扫描法, Yu^[7]、Feng 等^[8] 及 Pandey 等^[9] 提出的利用标号表二次或多次遍历替代图像扫描的标记方法等。多行扫描的组合判断复杂, 而标号表多次遍历降低了连通时效性, 也不利于特征计算。在直接目标特征计算方面, Tan 等^[10]、Fan 等^[11] 提出了一次遍历同时统计参数信息或计算质心的方法, 但针对连通域形状与特征计算较简单, 且在图像遍历完成后才能输出结果, 实时性仍不能满足高性能系统的要求。

现有的连通域标记方法的瓶颈之一在于, 系统消耗的时间需先分配给目标区域连通计算, 然后再给目标特征计算, 实时建模表征的效率较低。对于空间应用的星上实时目标检测系统而言, 目标区域标定与特征提取计算需同时同步完成; 而且, 星载系统资源有限, 实现多目标特征的提取应尽量优化资源的利用。

本文提出一种基于分割像素一次遍历的目标特征实时提取计算方法, 并利用 FPGA 实现。此法主要解决了空间实时图像处理的 3 个关键问题: (1) 在图像逐行进入时, 沿行方向做连通域标记的同时完成多目标精细化特征迭代计算, 并在目标区域连通完成后, 即时输出目标区域和特征计算结果, 无需等待全图输入结束; (2) 采用并行策略对计算进行加速, 满足高分辨率高频图像的处理需求; (3) 减省星载系统的资源消耗。

1 算法原理

目标检测算法通过阈值分割后得到目标像元的

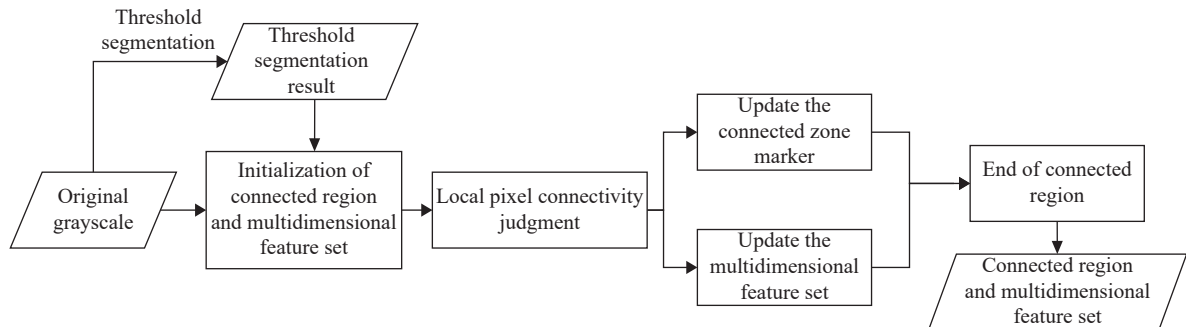


图 1 目标特征提取算法流程

Fig. 1 Algorithm flow of target feature extraction

坐标, 对于扩展尺寸目标, 需对分割后的像元簇进行形态连通获取扩展目标形态尺寸, 然后利用目标多维特征度量方法对目标多维特性进行表征, 以支持后续的目标跟踪任务。本文研究的目标形态连通特征提取方法总体流程如图 1 所示。具体流程为: 接收原始图并对其做阈值分割; 做连通标记, 建立目标区域的多维特征集; 依据连通关系更新连通区域标记, 同时更新多维特征集; 判断连通区域是否结束, 结束时输出最终的连通区域和多维特征集。

1.1 目标多维特征集

目标多维特征集包括目标局部峰值对比度特征 (L_{peak})、区域灰度重心位置 (R_c)、能量峰值 (E_{max})、区域能量和 (E_{area}) 以及区域覆盖像元数 (A_{num}) 等。

目标区域灰度重心的计算公式为

$$R_c(x_c, y_c) = \begin{cases} x_c = \frac{\sum p(x, y) \cdot x}{\sum p(x, y)}; \\ y_c = \frac{\sum p(x, y) \cdot y}{\sum p(x, y)}. \end{cases} \quad (1)$$

式中 $p(x, y)$ 表示当前目标像素的灰度值, x 和 y 为该像素对应的行列坐标, x_c 和 y_c 则为区域灰度重心的行列位置坐标。

目标局部峰值对比度特征 (L_{peak}) 计算公式如下:

$$\begin{aligned} G_{\text{max}} &= \frac{1}{9} \sum_{l=-1}^1 p(x_m + l, y_m + l); \\ G_{\text{area}} &= \sum_N p(x_i, y_i); \\ L_{\text{peak}} &= \frac{(N - 9) G_{\text{max}}}{N G_{\text{area}} - 9 G_{\text{max}}}. \end{aligned} \quad (2)$$

其中, G_{max} 为能量峰值点 3×3 区域的灰度均值, G_{area} 为连通区域的灰度和, 峰值对比度由峰值点邻域

均值与剩余区域的均值之比确定。

区域能量和(E_{area})计算公式如下:

$$E_{\text{area}} = \sum p(x_i, y_i). \quad (3)$$

式中, 能量峰值 $E_{\text{max}}(x_i, y_i)$ 为区域中能量最大的像素点的能量值, 区域覆盖像元数 A_{num} 为区域内所有目标像素数量的统计值。

1.2 区域像素连通

目标特征集建立在连通区域基础上。在进行连通判断时, 需要对每一行图像进行逐像素遍历, 同时缓存当前行与上一行已标记的信息进行局部像素连通关系判断。传统连通域判别工作窗如图 2(a)所示, 其中 p 为当前遍历点(前景像素), a, b, c, d 为周边背景像素。在标记 p 的同时, 建立标号表记录相邻行连通区域的标号, 建立多维特征属性表记录对应的特征信息。在遍历比较的同时对相邻行标号表和多维特征表进行维护。遍历到连通区域结束条件满足时, 目标区域标定结束, 多维特征集更新完成, 为接下来获取目标区域特征提供了前提。

2 多维特征集计算方法

连通域标记主要基于二值图像进行, 目标特征提取应读取原图信息, 实现连通判断与特征计算的同步性, 需要依据二值图像坐标读取原始图像灰度信息用于特征计算, 二值用于连通域判断。为了对不同目标区域进行区分, 建立区域标号表, 每个标号同步建立标号映射表和目标特征属性集, 前者用于区域邻接关系表示, 后者用于区域特征参数迭代计算。不同标号映射后的标号如果相同, 则将对应的多个区域进行特征合并更新, 保证对像素邻接和区域邻接两种连通状态的特征计算和更新。

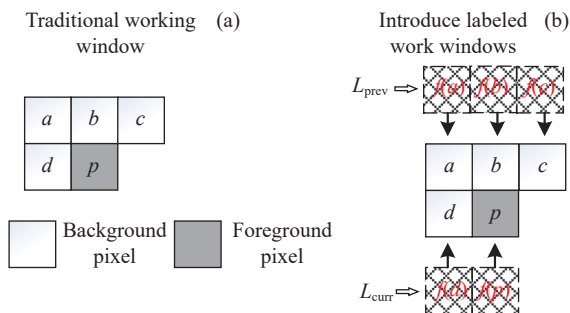


图 2 引入标号的工作窗

Fig. 2 Introduce labeled work windows

2.1 快速连通域标记

一次遍历快速连通域标记基于 8 邻域的连通判别方法, 不同于传统的 8 邻域标记只判断待标记点的周围 4 个像素, 该方法在像素判断时引入了标号表, 将判断与等价标号合并在一个流程中完成。

为待标记像素 p 建立如图 2(b)所示的工作窗, 设立上一行标号表 L_{prev} 和当前行标号表 L_{curr} 。在连通域标记时, 当前点的临时标号只与被扫描过的点有关, 即 p 的临时标号只取决于 $f(a), f(b), f(c)$ 和 $f(d)$, 同时根据邻域标号的分布情况在分配标号的同时做等价合并。标号分布情况有以下三种状态。

状态 1 邻域为背景像素, p 为新的连通域起点, 定义其标号为 $f(i+1)$ 。

状态 2 领域内有 1~4 个已标记点, 但标号相同, 则当前像素标号继承邻域标号 $f(i)$ 。

状态 3 邻域内两个点标号不同, 按连通域原则, 两不同标号位置只有两种, 分别为 a 和 c 不同, 或 d 和 c 不同。此时将 p 设为两标号中的小标号; 在遍历过程中, 仅在此种情况下, 使用图 2(b)建立的标号表 $f(x)$, 该表缓存上一行与当前行共 2 行的标号信息。当标号冲突时, 将前景像素确定为小标号, 同时将上一行的大标号也同步更新为小标号。

当一个目标区域底边任意一点的 8 连通区域内没有前景像素时, 没有新的标号出现, 同时不同区域的标号在连通域底部被更新为同一标号, 连通区域确认完成。

图 3 显示了状态 3 的两种连通域相遇时标号合并情况, 从图 3 可以看出, 两个不同区域在底部完成合并, 但上部标号并不相同。为保证两个区域目标特征合并计算的正确性, 本文提出了一种基于标号映射表的特征属性传递机制, 建立标号映射表, 将标号表与特征属性计算值连接到一起。在标号合并时, 将大标号区的特征属性传递到小标号区, 完成不同区域特征属性的合并计算, 避免标号的二次整理。

2.2 标号映射表与目标特征属性传递

根据不同的目标区域建立对应的目标特征集缓存区, 通过建立特征集缓存区与区域标号的标号映射表, 将二者一一对应起来。当两个区域没有合并时, 按照各自的标号计算各自的目标特征。当前景像素被确认为新标号时, 则为该标号开辟特征集缓存区, 存储当前区域各特征属性信息。同时为该存储区

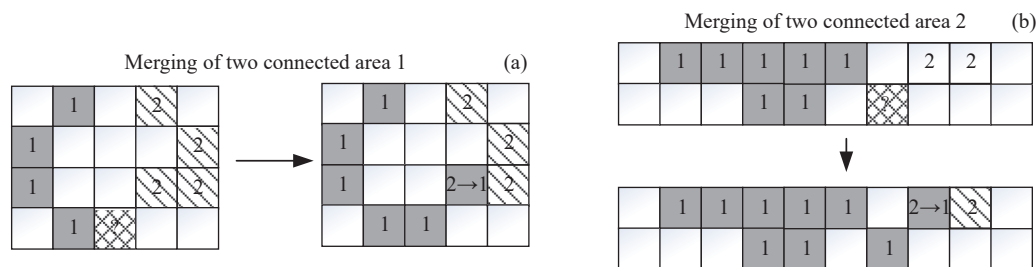


图3 标号更新

Fig. 3 Label update schematic diagram

分配标号映射表,记录下标号对应的缓存区位置。每个区域每连通一个点,即迭代计算当前区域的各项特征属性值,并实时更新该区域目标特征属性缓存区。

图4所示为映射表对应目标特征属性集在特征缓存中的存储位置。当两个区域相遇时,标号表将当前像素的标号更新为最小标号。同时,根据对应的映射表,可以找到缓存区中两个区域的各自特征。将大标号对应的存储区域里计算的特征参数读取出来,与小标号存储区域内的参数按特征计算要求进行合并运算,完成两个等价标号对应区域的特征合并计算,得到合并后的区域特征,存放在小标号特征属性区。大标号存储区域空出,大标号被释放,可循环利用。

2.3 多维特征计算

2.3.1 目标特征属性

在对目标区域做连通判断时,同步建立目标区域特征属性集,用于区域连通完成后多维特征的计算。特征属性信息列于表1。

每连通一个目标点,将特征属性集(见表1)内的数据同步更新,对每个目标点,分别计算其与 x 坐标的累积和,与 y 坐标的累积和,以及其周边 3×3 的区域能量累积和;与最大点相比,如果能量高,则更新最大点,同时更新最大点坐标,反之不更新;比较当前点坐标与区域左上、右下坐标,小于左上或大于右下,则替换对应区域边界,反之不替换;迭代计算区域能量和累加;迭代计算区域目标个数。

当两个连通区域碰到一起时,根据图4所示特征属性传递的方法,分别提取小、大标号区域的特征属性集,将对应属性进行合并运算,得到新的特征属性集。通过标号合并和属性传递,以及属性合并计算,当连通完成时,目标区域的特征属性统计、计算与合并全部完成,得到目标区域的特征属性。其中目标区域位置、目标区域总能量和、目标区域点个数可直接

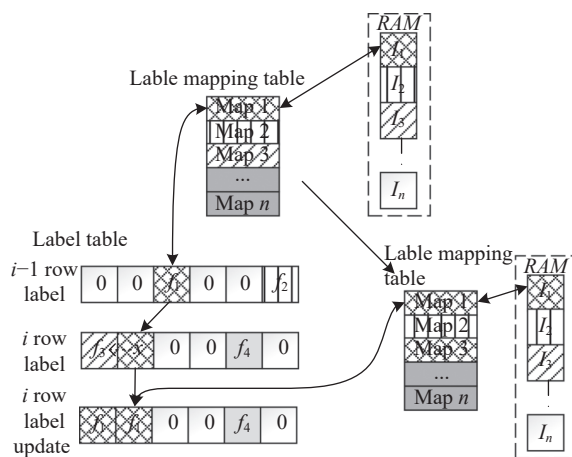


图4 标号映射表与特征属性传递

Fig. 4 Label mapping table and attribute passing

表1 目标区域多维特征属性信息

Table 1 Multi-dimensional feature attribute information of target region

行(x)最大坐标	列(y)最大坐标
行(x)最小坐标	列(y)最小坐标
能量峰值	区域点数和
区域能量和	峰值点局部灰度和
行(x)坐标灰度累积和	列(y)坐标灰度累积和

提取使用。但灰度重心和局部对比度还需进一步计算。

2.3.2 灰度重心计算

灰度重心在连通区域全部完成后,根据式(1)对灰度重心进行计算,先提取特征属性集中的行(x)方向坐标能量累积和 $\sum E_x$ 与列(y)方向坐标能量累积和 $\sum E_y$,分别计算上述两个值与区域能量总和 E_{area} 的比值,即可得到连通区域灰度重心。

2.3.3 局部对比度特征计算

将目标局部峰值对比度(L_{peak})的计算式(2)变换

为式(4)。提取属性集中的能量最大点周边 3×3 区域能量($E_{\max}$)和目标区域所有点能量(E_{area})即可完成特征计算,有

$$L_{\text{peak}} = \frac{(N-9)E_{\max}}{9NE_{\text{area}} - 9E_{\max}}. \quad (4)$$

3 FPGA 实时实现

FPGA 作为半定制可编程芯片,其内部开放的寄存器资源与丰富的缓存资源可方便地实现异步并行处理流水。对于高分辨率的遥感图像,视场内存在的目标点数量可能较多,待计算的目标特征也多,待处理图像与目标特征属性集的存储会占用较多的存储空间。在实现时采用循环缓存策略来降低资源消耗,分块并行流水策略以提升处理效率。

3.1 实现流程

采用 Xilinx V7 系列芯片作为主处理芯片,以 Vivado 设计套件作为开发工具,采用 Verilog 语言作为设计语言。具体实现的硬件流程如图 5 所示。

接收分割后带二值结果的图像数据为原始输入。图像数据存储在大量缓存中,通过缓存控制管理模块将图像从缓存中读出,分别提取二值数据和图像数据。二值数据用于目标区域连通;图像数据用于对应的特征属性计算。缓存数据读取后,判断二值图像是否为 1,若为 1 表明该像素为目标像素。二值数据进入连通域标记模块,同时提取该二值对应的图像数据送入特征属性提取计算模块,等待连通域判决结果。标记过程中建立标号表及标号映射表,按 8 连通准则进行判别,完成标号的更新与合并。通过标号映

射表对应标号和特征属性集在大量缓存中的对应位置,根据标号表的标号提取对应图像数据,计算各特征属性并完成属性的传递与更新,如图 4 所示。当目标连通到区域最下一行,邻域内再没有其他目标或区域可以连通时,目标区域连通完毕,发出连通结束标识。提取特征属性表中的各个属性值,同时根据公式计算灰度重心与区域对比度,计算完毕将全部特征结果输出,完成实时目标特征提取计算。

在硬件实现过程中,主要设计的重点在于资源和速度,下面对其进行详细说明。

3.2 循环缓存

特征提取计算涉及的主要存储包括输入图像数据的存储,标号表、标号映射表及多目标特征属性区等。其中标号表与当前行图像、下一行图像以及上一行标号表有关,连通域结束的判断条件是邻域内没有可关联像素,因此对于输入的图像,只需缓存 2 行即可满足计算要求,但后续处理时要采用分块并行的策略,此处设计时增加一倍余量,缓存 4 行图像。

特征属性缓存区的设置与图像局域点密集程度有关,本方法中由于特征是实时计算实时释放的,且特征属性集长度固定。在标号合并后,高标号特征属性集的存储空间将被释放可循环利用,因此特征属性集也设置为循环缓存区以减少存储空间的占用。

按连通域原则,如果每行有 m 个像素,则两行最多有 $m/2$ 个离散点。特征属性表中各个属性精度设置遵循如下原则。假设图像尺寸为 $L \times W$,分辨率为 P bit,则行列坐标位宽(l_x, l_y)满足 $2^{l_x} \geq L$ 和 $2^{l_y} \geq W$ 。能量峰值位宽为 P ,区域点数和位宽(l_{num})满足 $2^{l_{\text{num}}} \geq L \times W$ 。区域能量和位宽(l_E)满足

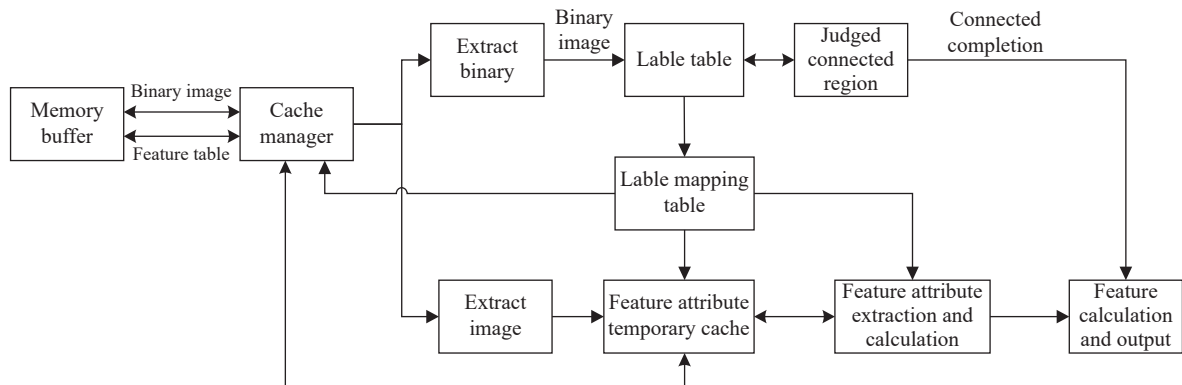


图 5 FPGA 硬件实现流程

Fig. 5 FPGA hardware function block diagram

$2^L \geq L \times W \times P$ 。考虑灰度重心计算时的精度设置,将行列坐标灰度累积和统一设为 double 型。

3.3 分块并行加速

在连通域标记与特征提取过程中,涉及的处理延时主要包括连通域标记、标号判别合并、映射表读写、特征属性计算、连通后灰度重心与局域对比度计算以及循环缓存区地址控制计算等。由于连通域是逐像素连通的,且特征可以区域间传递,因此为提高计算效率,在设计时采用分块并行处理策略,每块的运行流水线如图 6 所示。其中,特征属性计算、标号映射计算及循环缓存区的计算均包含在标号判别和

标号更新里,不增加新的计算延时,最终的处理延时只包含图像输入、标号判断与标号更新的延时,加最后的特征计算及结果输出延时。输入的图像在列方向根据实际需求可分为 m 块,同时开启连通处理,根据区域特征属性的传递原理,两块间有一个像素的重叠区,用于两分块区域标号的连接。分块并行的仿真结果如图 7(b) 所示。

4 结果分析

4.1 仿真验证

图 7(a) 为仿真模拟序列图像(局域有密集点)连

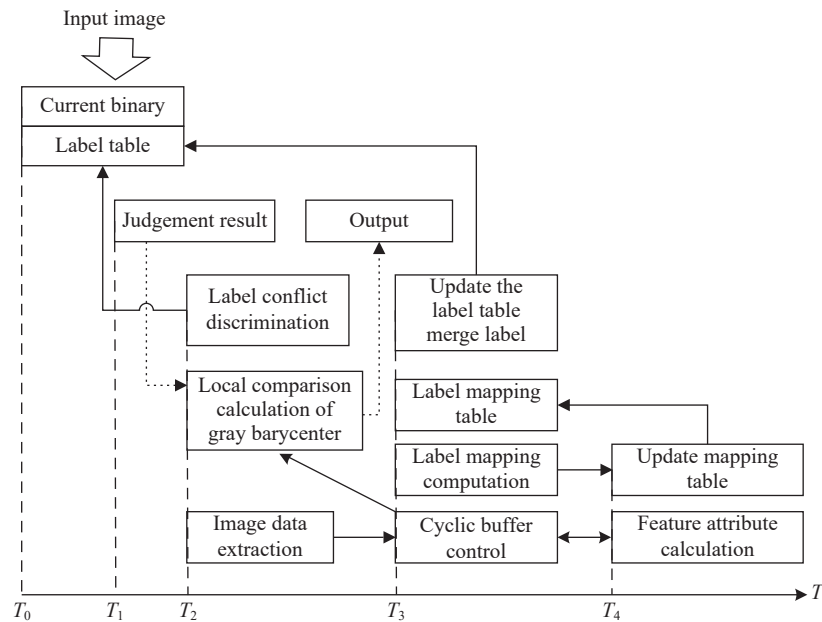


图 6 单块特征提取运行流水

Fig. 6 Feature extraction running flow

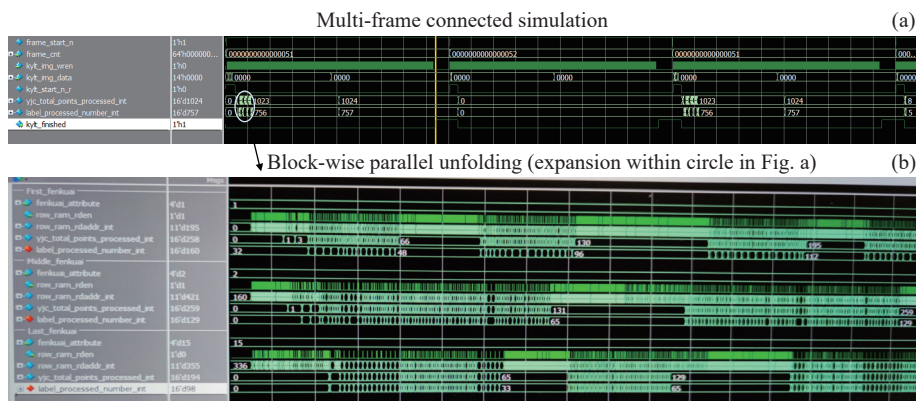


图 7 连通域仿真结果

Fig. 7 Simulation results of connected domain

通后的结果,可以看出分割后离散目标有 1024 个点,连通后变为 757 个点;图 7(b)是图 7(a)分块并行的局部详细仿真结果,按列方向分为三块,分别连通,连通后的结果点数累加到一起,去掉重叠部分,即为图 7(a)中连通后的结果点。图 8 为特征实时计算结果仿真,特征属性分别计算完成后,在连通结束时计算灰度重心和局域对比度并输出。

4.2 性能分析

以下分别从计算时间、资源消耗两方面对特征提取方法的性能进行分析。

4.2.1 计算时间

计算时间包括两部分,连通域生成时间和特征计算时间。假设图像尺寸为 $L \times W$, 每行内包含的目标不超过 n 个, 电路工作的时钟周期为 T_{clk} 。目标在连通时有三种基本情况(见图 9)。图 9 中排列 1 总耗时 $7 T_{\text{clk}}$, 排列 2 总耗时 $3 T_{\text{clk}}$, 排列 3 总耗时 $5 T_{\text{clk}}$, 按 n 个点平均处理时间为 $5n/3 T_{\text{clk}}$, 由于排列 2 和 3 的情况居多, 因此最终计算的总耗时接近 $2n T_{\text{clk}}$ 。

连通标记模块从扫描图像的第二行进入时就判断输出, 若第二行没有新点与第一行的目标区连通, 表明第一行目标区连通完成, 提取区域特征属性集进行特征属性计算, 输出目标区域。后续进入行以此类推, 直到全图扫描完成。如果每行内包含目标数量过多, 每行图像按列向根据实际需求可分为 m 块并行处

理, 计算总耗时降为 $2n/m T_{\text{clk}}$ 。因此, 全图扫描后, 如图像输入时间为 $(L \times W) \times T_{\text{clk}}$, 则整幅图像的全部连通域处理完成时间不超过 $(L \times W + 2n/m) \times T_{\text{clk}}$ 。根据文献 [2] 的计算, 其基于游程的连通处理能力达到 $(L \times W + 2n - 1) \times T_{\text{clk}}$, 本文的设计时间比其快近 m 倍, 更接近理论极限时间 $L \times W \times T_{\text{clk}}$ 。

特征计算时间包含特征属性计算时间和连通完成后目标特征计算时间。其中特征属性计算时间在连通过程中实时读取图像数据进行计算, 包含在连通域流水内, 不影响整体结果的计算延时。目标特征的计算延时是连通域完成后的整合计算时间, 灰度重心和局部对比度计算涉及除法器的效率, 可根据资源和实时要求的具体情况调整延时, 计算延时通常在 $10 \sim 60 T_{\text{clk}}$ 左右。因此从处理图像进入到特征输出的最长耗时不超过 $(L \times W + 2n/m + 60) T_{\text{clk}}$ 。

4.2.2 资源消耗

以上各步流程不额外消耗系统内存, 对接口的速率等也没有额外要求, 仅占用 FPGA 内部的逻辑资源、计算资源以及内部存储器(Block RAM)资源。

由于连通及目标特征计算需要缓存标号表、特征属性集及部分原始图像数据, 对 FPGA 内部存储资源的消耗不能忽略。以下主要分析存储资源使用情况。表 2 给出了对于 1024×1024 图像, 按 4 行图像循

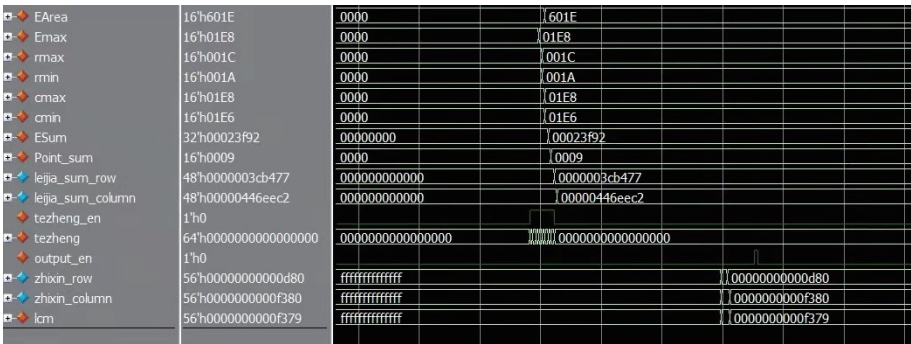


图 8 目标特征实时计算结果

Fig. 8 Real-time calculation results of target features

表 2 存储资源消耗

Table 2 Storage resource

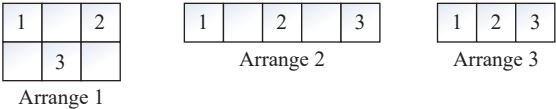


图 9 连通域分布排列

Fig. 9 Connected domain distribution permutation

内容	存储区域	数量
二值+原图	Block RAM	4
目标特征属性集	Block RAM	14
标号表	Block RAM	2
标号映射表	Block RAM	1

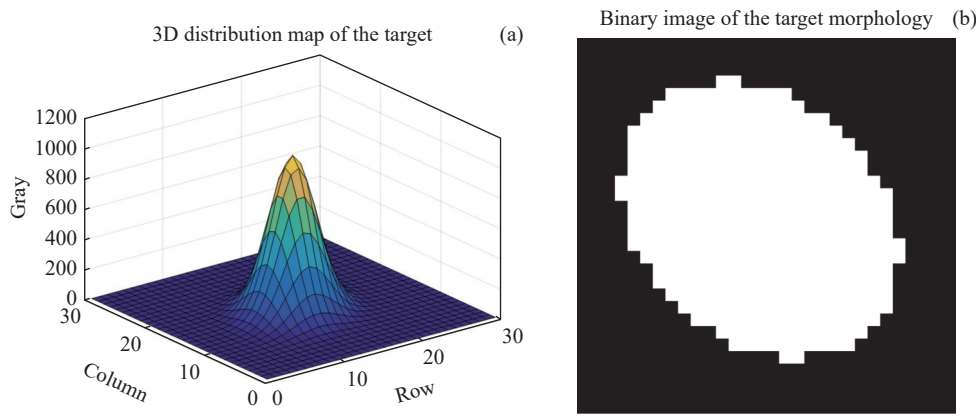


图 10 单目标灰度及形态分布
Fig. 10 Gray scale and morphology distribution of single target

环缓存, 1000 个目标点特征属性循环缓存条件下的存储资源消耗情况。V7 系列中最小的 XC7 V585 T 的内部缓存资源也有 1590 个 18 K BRAM, 因此资源消耗量很小, 完全可以满足实时存储计算要求。

4.3 特征提取效果验证

4.3.1 单目标连通实验

利用式(5)高斯函数仿真的目标灰度分布见图 10(a), 形态分布见图 10(b), 本文算法输出结果如图 11 所示, 可以看出本文算法可以正确地提取目标形态外接矩形, 同时可以计算能量峰值点(星号)以及灰度重心坐标(圆圈), 有

$$z = [D \times \exp(-0.5 \times [(x - \mu_x)^2 + (y - \mu_y)^2] / C)] \quad (5)$$

式中, D 为最大量化灰度值, μ_x 和 μ_y 分别为峰值点的图像坐标, C 为灰度协方差矩阵, 相应设置值见表 3。

4.3.2 多目标连通实验

为对多目标连通能力进行验证, 利用上述目标仿真方法仿真多个不同尺寸和形态的目标斑块, 对算法输出结果的正确性进行验证, 实验结果选取 512 图像(左上区域)目标集中分布的 128×128 区域, 目标仿真参数见表 4。多目标灰度与形态分布如图 12 所示, 连通结果如图 13 所示, 特征提取结果见表 5。

5 结语

针对高分辨率遥感图像内多目标多维特征实时提取问题, 基于分割像素一次遍历的快速连通域标记及目标特征实时计算方法, 能够在 FPGA 系统上得以很好地实现。该方法具有如下特点。

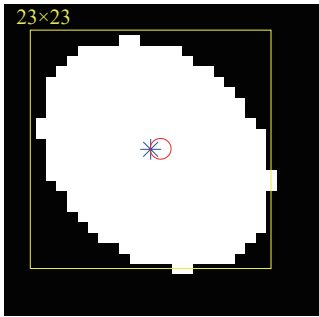


图 11 目标光斑连通结果(星号为能量峰值点)
Fig. 11 Result of target spot connectivity (The asterisk is the peak energy point)

表 3 单目标仿真参数
Table 3 Single objective simulation parameters

序号	参数名称	含义	设置值
1	峰值点坐标	目标能量峰值点图像坐标	$\mu_x = 15$ $\mu_y = 15$
2	协方差矩阵 C	用于描述形态分布	[8, 2; 2, 8]

表 4 多目标仿真参数
Table 4 Multi-objective simulation parameters

序号	参数名称	含义	设置值
目标1	峰值点坐标	目标能量峰值点图像坐标	(40, 65)
	协方差矩阵 C	用于描述形态分布	[8, 2; 2, 8]
目标2	峰值点坐标	目标能量峰值点图像坐标	(70, 85)
	协方差矩阵 C	用于描述形态分布	[5, 2; 2, 5]
目标3	峰值点坐标	目标能量峰值点图像坐标	(100, 115)
	协方差矩阵 C	用于描述形态分布	[8, 2; 2, 8]

(1)通过建立标号表、标号映射表及特征属性传递表, 将连通域标记、标号合并与特征提取计算结合,

实现边连通、边计算、边输出,实时性强,可并行处理。

(2)采用分块并行流水策略进一步提升处理能力。本方法的设计时间比其他方案快,更接近理论极限时间 $L \times W \times T_{\text{clk}}$ 。

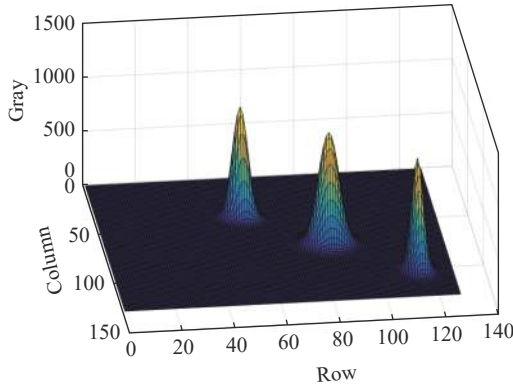


图 12 多目标灰度及形态分布
Fig. 12 Multi-target gray scale and morphology distribution

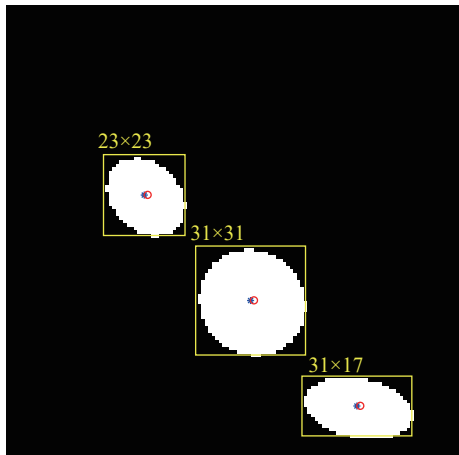


图 13 多目标光斑连通结果
Fig. 13 Multiple target light spot connectivity results

表 5 多目标特征提取结果
Table 5 Multi-target feature extraction results

序号	特征名称	目标1输出值	目标2输出值	目标3输出值
1	能量峰值	1023	800	500
2	能量峰值点坐标	(55, 40)	(85, 70)	(115, 100)
3	区域行列边界	[44 66 29 51]	[71 99 56 84]	[107 123 86 114]
4	能量和	49783	75204	26830
5	区域目标像素数	373	689	371
6	局部对比度	8.25	7.62	7.27
7	灰度重心坐标	(55, 41)	(85, 71)	(115, 101)

(3)利用循环缓存区缓存图像及特征属性,降低了系统的资源消耗,且不受图像尺寸的限制,输入图像存储够 4 行即可进行计算。

仿真测试结果表明,此特征提取方法正确有效,实时性好,在红外目标实时检测等领域具有很强的理论与实用价值。

参考文献

- [1] ROSENFELD A, PFALTZ J L. Sequential operations in digital picture processing[J]. *Journal of the ACM*, 1966, **13**(4): 471-494
- [2] WANG Fan, ZHOU Guoqing, ZHANG Rongting, et al. FPGA-oriented fast connected component labeling method[J]. *Computer Engineering and Applications*, 2020, **56**(22): 230-235
- [3] LING Luxiang. Target Localization Method Based on Embedded Vision with FPGAs and Its Application[D]. Guangzhou: South China University of Technology, 2017
- [4] ZHANG Guohe, XU Kuai, DUAN Guodong, et al. A fast labeling algorithm of connected components applicable for hardware implementation[J]. *Journal of Xi'an Jiaotong University*, 2018, **52**(8): 95-101
- [5] DAI Huadong, HU Moufa, LU Huanzhang, et al. Design and implementation of target detection algorithm based on connected-domain labeling[J]. *Modern Electronics Technique*, 2015, **38**(20): 71-74
- [6] ZHAO Xiao. Speed-up of Connected Component Labeling Algorithm and Its Application in Paper Defect Detection[D]. Xi'an: Shaanxi University of Science and Technology, 2019
- [7] YU Xiaoyu. Research on the Technologies of High Speed Vision Measurement System[D]. Harbin: Harbin Institute of Technology, 2014
- [8] FENG Haiwen, NIU Lianqiang, LIU Xiaoming. Efficient one-scan algorithm for labeling connected component[J]. *Computer Engineering and Applications*, 2014, **50**(23): 31-35
- [9] PANDEY J G, KARMAKAR A, MISHRA A K, et al. Implementation of an improved connected component labeling algorithm using FPGA-based platform[C]//Proceedings of 2014 International Conference on Signal Processing and Communications. Bangalore: IEEE, 2014: 1-6
- [10] TAN Xubin, XIE Yizhuang, CHEN He, et al. Design and implementation of the connected component labeling based on FPGA[J]. *Signal Processing*, 2011, **27**(11): 1729-1733
- [11] FAN Qiaoyun, ZHANG Guangjun. Spot centroiding algorithm for discrete noise image and its hardware implementation[J]. *Optics and Precision Engineering*, 2011, **19**(12): 2992-2998