

# 改进游程编码算法的快速星点提取

章家保, 张剑阳, 刘 赫, 李 岩

(吉林大学 仪器科学与电气工程学院, 长春 130061)

**摘要:**为提高星点质心提取的实时性与准确性,提出了一种基于现场可编程逻辑门阵列(FPGA)的改进游程编码算法来实现快速星点提取。该算法结合星点目标特点和FPGA的并行处理结构,改进了传统游程编码算法在提取时需要设置等价表进行标号合并以及多次轮询游程编码的不足,只需扫描一次图像后延迟若干个时钟周期便可提取星点质心坐标。将本文算法在星敏感器的FPGA上进行测试验证,在时钟频率为50 MHz、输入图像分辨率为 $1280 \times 1024$ 像素的情况下,本文算法星点质心提取时间仅需 $2 \mu\text{s}$ 左右,提取星点质心位置完全正确。与用RISC微处理器(ARM)进行星点质心提取时消耗时间近19 ms相比,本文算法优势明显,具有显著的工程应用价值。

**关键词:**信号与信息处理;质心提取;现场可编程逻辑门阵列;连通域扫描;星敏感器;游程编码

**中图分类号:**V448.22 **文献标志码:**A **文章编号:**1671-5497(2025)04-1467-07

**DOI:**10.13229/j.cnki.jdxbgxb.20230730

## Fast star point extraction with improved run-length encoding algorithm

ZHANG Jia-bao, ZHANG Jian-yang, LIU He, LI Yan

(College of Instrumentation and Electrical Engineering, Jilin University, Changchun 130061, China)

**Abstract:** To enhance the real-time performance and accuracy of star point centroid extraction, an improved run-length encoding algorithm based on field programmable gate array (FPGA) was proposed for fast star point extraction. The algorithm combines the characteristics of star point targets with the parallel processing structure of field programmable gate array, addressing the shortcomings of the traditional run-length encoding algorithm that requires setting equivalence tables for label merging and multiple rounds of polling for run-length encoding during extraction. It only requires scanning the image once, and after a delay of several clock cycles, the centroid coordinate of the star point can be extracted. Finally, the algorithm is tested and validated on an field programmable gate array of a star sensor. With a clock frequency of 50 MHz and an input image resolution of  $1280 \times 1024$  pixels, the algorithm takes about 2  $\mu\text{s}$  to extract the centroid of the star point, and the position is completely accurate. Compared to the nearly 19 ms consumed by centroid extraction using an Advanced RISC Machine (ARM), the improved algorithm demonstrates significant advantages and holds notable engineering application value.

收稿日期:2023-07-12.

基金项目:国家自然科学基金项目(62073150,62227812).

作者简介:章家保(1981-),男,副教授,博士.研究方向:卫星星敏感器系统设计、仿真及测试技术;卫星姿控部件电子学设计与测试技术.E-mail:zhangjiabao@jlu.edu.cn

**Key words:** signal and information processing; centroid extraction; field programmable gate array (FPGA); connected domain scan; star sensor; run-length encoding

## 0 引言

航天器在太空中执行各种探测任务时,需要对航天器进行准确的姿态信息测量,其中测量信息的器件被称为姿态敏感器。目前航天器上最常用的高精度姿态敏感器为星敏感器<sup>[1,2]</sup>。它所用的参考系是与航天器运行轨道没有关系的天球坐标系,通过拍摄的星图来分析恒星所在的坐标位置,并以此来进行航天器的姿态确定。其工作过程主要包括4个阶段,分别为星图获取及预处理、星点坐标提取、星图识别与姿态确定<sup>[3]</sup>。其中,星点坐标提取的准确性与快速性会极大程度影响星敏感器的姿态信息精度与输出频率。因此,如何提高星点坐标提取的准确率和提取速率是提升星敏感器性能的关键因素之一。

目前,国内外星点提取算法主要为两类:高斯拟合法与质心法<sup>[4,5]</sup>。高斯拟合法是将拍摄的星图近似看作高斯分布,然后通过高斯拟合法将灰度分布进行拟合得到高斯能量中心,该中心即为星点的中心位置。由于该算法复杂度高、计算量大,在质心提取效率上受到一定限制<sup>[6]</sup>。质心法及其各种改进算法具有精度高、算法简单且易于实现等优势,是目前星敏感器运用较为广泛的算法<sup>[7]</sup>。在计算星点质心位置之前需要将星点像素聚类提取,主要方法是利用连通域算法<sup>[8]</sup>确定星图中的邻域关系,并对星图中的连通区域星点进行标记。不同的连通域标记算法对星点质心提取的速度也会不同。星图中常用的连通域关系为四连通域或八连通域<sup>[9]</sup>。目前的算法一般需要对星图进行两次扫描,首先从上至下扫描所有像素来判断连通域位置,然后再将连通域通过等价表进行合并<sup>[10]</sup>。然而,上述连通域算法在星点质心提取速度上存在一定不足,会导致星点提取时间较长,并占用大量硬件存储资源以及无法保证星敏感器的实时性。

近年来,现场可编程逻辑门阵列(field programmable gate array, FPGA)的应用日益广泛。由于其工作机制,FPGA具备强大的并行处理能力,能够显著提升图像采集与处理的效率<sup>[11-13]</sup>。为此,本文提出了一种改进的游程编码算法来进行快速星点质心提取,可不用将星图进行缓存,只

需一次扫描即可完成星点提取,改进了标号合并与回收过程,在质心计算时减少游程编码寻址的时间。利用FPGA的并行处理优势实现阈值分割、星点目标提取、标号回收以及质心坐标计算等步骤,从而快速准确地得到星点质心的坐标位置。

## 1 改进的游程编码扫描算法

在传统的游程编码基础上提出本文改进算法<sup>[14]</sup>,以此来提升星敏感器的质心坐标提取速度。

### 1.1 阈值分割

当星敏感器在拍摄星图时,所得到的星图可能会存在背景噪声,其灰度值比星点的灰度值要低,因此需要一个合适的全局背景阈值,然后利用连通域分析算法将背景和星点分割开。确定阈值的大小可以有效将背景噪声消除,阈值过大可能会将有效星点剔除,阈值过小则可能将噪点视为星点提取出来。

因此,在本文算法中运用了动态背景阈值,当下一帧图像扫描之前会自动利用上一帧图像中计算的灰度平均值再加一定方差作为阈值,以此来减轻背景噪声对星点提取的影响。其中,像素灰度平均值如式(1)所示:

$$T = \frac{1}{1024 \times 1280} \sum_{y=1}^{1024} \sum_{x=1}^{1280} F(x, y) \quad (1)$$

式中: $T$ 为该帧星图的灰度平均值; $F(x, y)$ 为每个像素点的灰度值;图像分辨率为 $1280 \times 1024$ 像素。

### 1.2 星点目标提取

星图在进行去噪预处理后,需要进行星点像素聚类提取,常用的方法是通过连通域分析将星图中的星点进行标记,然后再计算出星点的质心位置。连通域分析算法主要有递归算法、逐行算法等,但这些算法多数先是将图像缓存然后进行两次扫描,但合并标号的过程比较复杂,且产生存储资源消耗过多,导致星点提取速率不高。因此,本文在传统游程编码的基础上对其进行改进,只扫描一次星图即可将星点目标提取。具体步骤如下:

(1)对去噪处理后的星图图像中的每个像素值进行由左向右、由上至下逐行扫描,当扫描到某

像素的灰度值大于所设的阈值,并且左侧像素点灰度值小于阈值时,存储该像素所在列的坐标为  $X_{S1}, X_{F1}$  被视为起始列;然后继续向右进行像素扫描,灰度大于阈值的像素均为与  $X_{S1}$  的像素相连通;直至扫描到某像素灰度值大于阈值且右侧像素灰度值小于阈值时,存储该像素所在列的坐标为  $X_{F1}$ , 被视为终止列。则该区域可记为一个连通区域并进行唯一标号。并将该连通域包含的信息存储在游程编码链表中,记录格式为该行连通域的标号+起始列+终止列。

(2)当扫描到下一行时,也同样取用上述判定方法。然后与上一行的连通域存储的信息进行对比,将该行连通区域的起始列  $X_{S2}$  和终止列  $X_{F2}$  与上一行  $X_{S1}$  与  $X_{F1}$  进行对比,满足  $X_{S2} \leq X_{F1}$  且  $X_{F2} \geq X_{S1}$  的条件时,则该区域与上一行连通区域是相连通的并标记上相同的标号,同时将存储在游程编码链表中的起始列、终止列以及标号更新为当前行的数据。若不满足上述的判定条件,则为一个新的连通区域,需要重新存储新的起始列与终止列,并标记一个新的标号。

(3)按照上述算法扫描至星图末尾像素并进行星点目标判定,如图 1 所示。上述方法一次扫描可将星图中的星点与背景分离开来,并对同一星点像素进行标记。传统的游程编码算法需要将每行的连通区域进行不同的标号标记,然后通过构建等价表进行标号合并。然而本文算法在进行连通区域判断时直接进行标号合并,标号作为链接地址并将同一标号的游程编码将按从左到右、从上到下的规则链接,有效地缩短了星点目标的提取时间,节省了硬件资源。

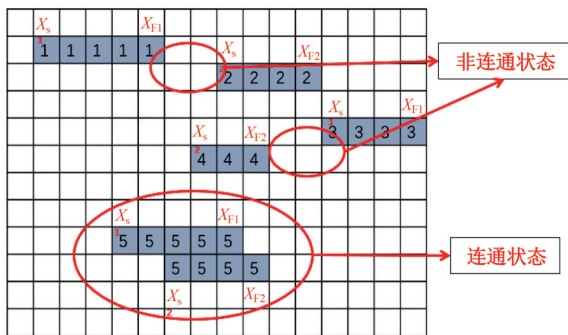


图 1 连通区域判定状态

Fig. 1 Connected area determines the status

### 1.3 标号回收

传统算法中标号回收是针对单个星点目标用多个标号情况时,在对该目标各部分进行标号

合并后,将剩余无效标号进行回收并重复利用。而本文算法不需要对每个星点目标都进行标号回收,只需针对特殊的畸变星点即可。

星敏感器拍摄的星图中星点可能会产生一些畸变。如图 2 所示为原始星点与畸变星点。

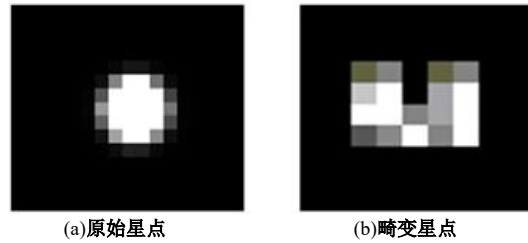


图 2 原始星点与畸变星点

Fig. 2 Primordial star points and distorted star points

当出现上述畸变星点时,传统的算法为第一行与第二行进行连通域判定时会出现两种标号 1 和 2,但在第三行连通域判断时,则将标号 2 合并到标号 1 中。此时,标号 2 的内容将会被清空。若再出现新的连通域时则需标号为 3,标号 2 则被丢弃,这样进行连通域标记会导致标号冗余,整幅星图扫描结束时标号过大,导致存储质心坐标信息的寄存器需要较大的存储空间。因此,本文算法提出了一种将标号回收的机制,在 FPGA 中利用 FIFO 将无用的大标号进行回收,然后释放出小标号进行标记,并在下一次新的连通域出现时在释放 FIFO 中最小的标号进行重新标记。这样会使星点标号不会产生冗余,减少寄存器的存储空间,在质心计算时加快了对游程编码的寻址速度,提高质心提取的效率。

如图 3 所示为本文算法畸变星点标号合并与回收的示意图。

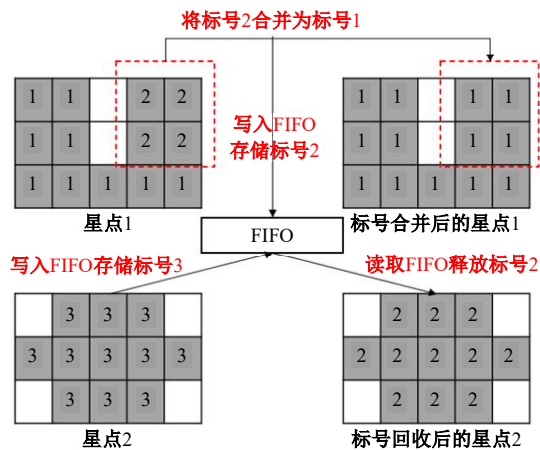


图 3 畸变星点标号合并与回收示意图

Fig. 3 Distortion star label merging and recovery diagram

### 1.4 异常星点剔除

星点目标与噪点或异常星点在大小方面有明显的不同,当星点目标出现拖尾现象或者月亮进入星敏感器视场范围内时,这些情况理论上均可视为一个异常星点目标,如图4所示。因此,本文算法利用目标的大小特征对异常星点目标进行剔除。首先,在进行连通域判定与标号标记时,同时将相同标号连通域的总像素个数、每行的像素个数以及总行数也存入游程编码链表中;然后通过本文算法对上述3个条件设置阈值,在每帧图像扫描结束后进行寻址判定。

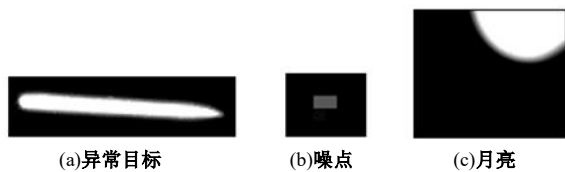


图4 异常星点目标

Fig. 4 Anomalous star target

根据星点目标自身的能量扩散程度,以及星敏感器视场角与卫星角速度对拖尾的影响进行阈值设置。随着角速度的增加,拖尾长度也会增加。经过计算,当角速度为 $2^\circ/\text{s}$ 时,拖尾长度约为30个像素。因此,当某行像素个数大于35个或者总行数大于35行时,可直接将该目标视为异常星点目标;若某星点目标总像素个数小于3个,则将其视为噪点进行剔除。阈值可根据目标识别效果进行动态调节。

### 1.5 质心信息计算与存储

在本算法中,利用FPGA并行处理的优势,在进行连通域判断时,将每个星点目标总像素个数、像素灰度值总和( $P$ )以及坐标 $x$ 、 $y$ 方向的灰度值与坐标相乘的累加和( $P_x, P_y$ )按照连通区域标号为地址进行寄存器存储。利用式(2)得到上述需要进行质心计算的信息。

$$\begin{cases} P = \sum_{x=1}^m \sum_{y=1}^n F(x, y) \\ P_x = \sum_{x=1}^m \sum_{y=1}^n F(x, y) \cdot x \\ P_y = \sum_{x=1}^m \sum_{y=1}^n F(x, y) \cdot y \end{cases} \quad (2)$$

式中: $F(x, y)$ 为每个像素点的灰度值; $x$ 、 $y$ 为每个像素点所在的行、列坐标; $m$ 、 $n$ 为星点目标总行数与总列数。

在扫描完一帧图像后,FPGA读取每个星点

目标的游程编码链表信息来判断是否为异常星点,若为正常星点则根据存储的星点信息进行质心计算,如式(3)所示。由于FPGA的逻辑资源受限,无法处理浮点数运算,本文算法是在扩大100万倍的情况下进行质心计算,以此在后续的处理中可将质心坐标保留6位小数。

本文算法减少了质心计算时的寻址时间,只需延时若干个时钟周期即可得到星点质心坐标。经过FPGA提取星点质心坐标位置后,通过总线将相关数据传输到ARM端,利用ARM端进行星图识别及星跟踪。

$$X = P_x/P, Y = P_y/P \quad (3)$$

式中: $X$ 、 $Y$ 为星点目标的质心坐标。

综上所述,本文提出的改进游程编码算法的快速星点质心提取算法的基本流程,如图5所示。

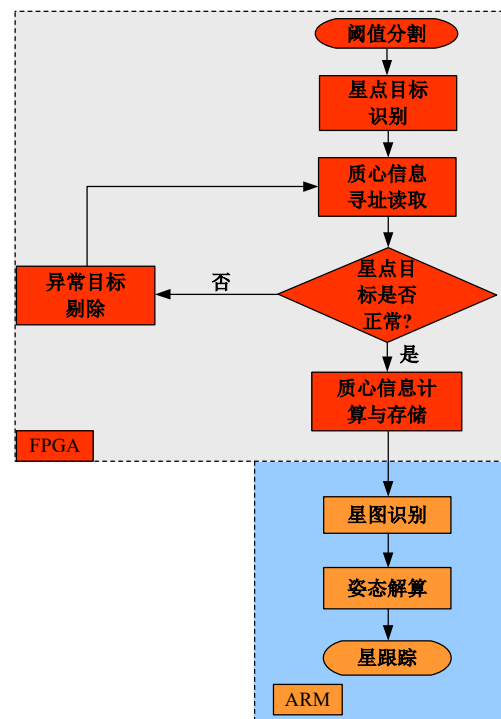


图5 本文算法流程图

Fig. 5 Flow chart of proposed algorithm

## 2 FPGA算法验证及结果分析

### 2.1 算法验证

为了验证本文算法在FPGA上的可行性,需要将本文算法在FPGA中进行实验测试。本文利用软件生成的模拟星图作为本算法的仿真输入,时钟频率为50 MHz进行验证。通过MATLAB软件的质心提取程序与本文算法进行对比,检验质心提取坐标是否一致。将本文算法与相同

表 1 FPGA 资源使用情况  
Table 1 FPGA resource usage

	4LUT	DFP	RAM1K18	User I/O
Used	1 017(1.18%)	555(0.64%)	2(1.83%)	41(15.36%)

表 2 模拟星图中有 5 个星点的提取结果

Table 2 Extraction results of 5 star points in simulated star map

目标 序号	MATLAB 提取质心坐标/像素		本算法提取质心坐标/像素		FPGA 中质心 提取时间/ $\mu$ s	ARM 中质心 提取时间/ $\mu$ s
	X	Y	X	Y		
1	1 053.760 416	45.715 460	1 053.760 416	45.715 460		
2	920.490 788	224.004 605	920.490 788	224.004 605		
3	1 010.325 242	364.962 783	1 010.325 242	364.962 783	0.6	16 464
4	602.477 215	761.500 000	602.477 215	761.500 000		
5	530.214 577	983.606 267	530.214 577	983.606 267		

表 3 模拟星图中有 10 个、15 个星点的对比结果

Table 3 Comparison results of 10 and 15 star points existing in simulated star map

星点个数	对比结果	FPGA 中质心 提取时间/ $\mu$ s	ARM 中质心 提取时间/ $\mu$ s
10	完全一致	1.0	17 544
15	完全一致	1.4	18 624

实验环境下使用 ARM 进行星点质心提取算法所消耗时间进行对比,分析对比提取时间。实验结果表 1~3 及图 6 所示。图 7 为模拟星图(为了显示效果星图进行反色处理)。

从上述对比数据中可以看出,本文算法提取的质心坐标与 MATLAB 软件提取质心坐标的坐标数据完全一致,验证了本文算法在 FPGA 中实现的可行性与可靠性。通过星点质心提取所消耗的时间进行对比,在接收完一帧图像时,利用 ARM 进行星点质心提取所需时间接近 19 ms,而本算法只需不到 2  $\mu$ s 即可得出星点质心坐标。

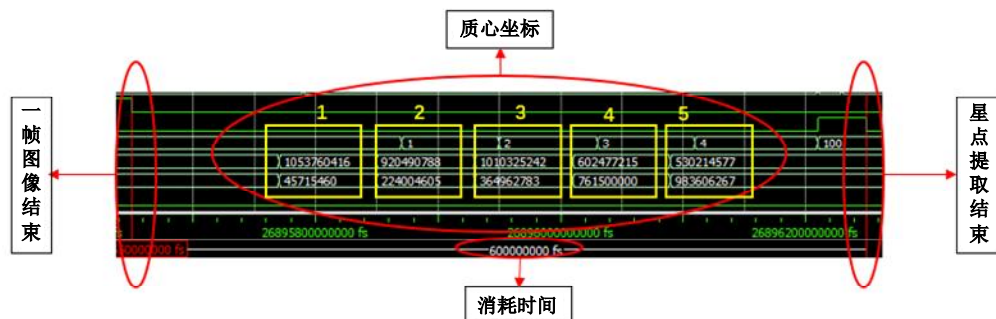


图 6 仿真数据

Fig. 6 Simulation data

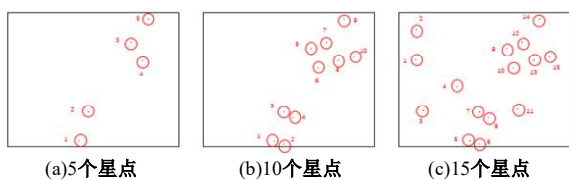


图 7 模拟星图(反色处理)

Fig. 7 Simulated star map(reverse color processing)

最后,为了验证本文算法中的异常目标剔除功能与畸变星点的准确性,本文利用静态星模中的星图进行实拍测试,如图 8、图 9 所示(为了显示效果星图进行反色处理),其中设置了 5 个异常目标。使用本文算法与 MATLAB 软件分别进行质心坐标提取,实验结果如表 4 所示。

通过表 4 可以看出,本文算法在提取质心坐标时,将星图中设置的 3 个异常目标均进行了剔

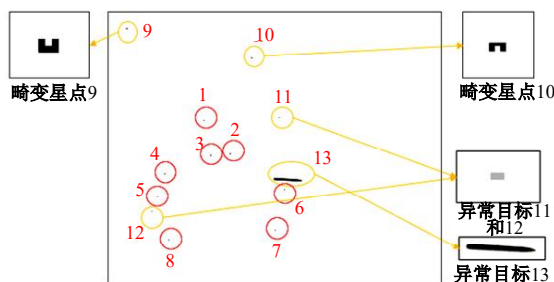


图 8 模拟特殊星图(反色处理)

Fig. 8 Simulate special star maps(reverse color processing)

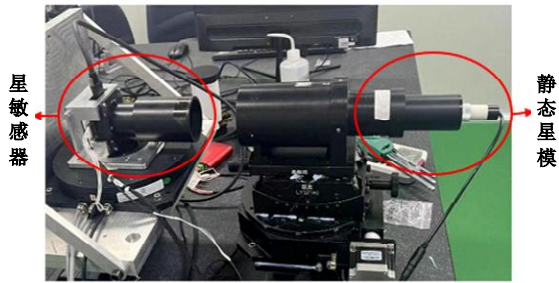


图 9 实际拍摄测试

Fig. 9 Actual shot test

表 4 特殊星图星点提取结果

Table 4 Special star map star extraction results

目标 序号	MATLAB提取 质心坐标/像素		本文算法提取 质心坐标/像素	
	X	Y	X	Y
1	367.919 469	409.935 398	367.919 469	409.935 398
2	470.379 032	537.242 943	470.379 032	537.242 943
3	393.917 574	543.201 525	393.917 574	543.201 525
4	217.110 975	613.310 975	217.110 975	613.310 975
5	676.900 000	671.700 000	676.900 000	671.700 000
6	188.531 914	701.067 375	188.531 914	701.067 375
7	642.077 777	824.655 555	642.077 777	824.655 555
8	231.578 651	862.460 674	231.578 651	862.460 674
9	71.000 000	60.200 0000	71.000 000	60.200 000
10	552.000 000	173.357 142	552.000 000	173.357 142
11	166.000 000	755.500 000	异常星点目标	
12	656.000 000	400.500 000	异常星点目标	
13	687.769 328	635.372 269	异常星点目标	

除,且仍能准确地将其余正常星点与畸变星点的质心坐标提取出来,验证了本文算法在保证质心提取精度的情况下可以有效地剔除异常星点目标。

### 2.2 外场观星实验

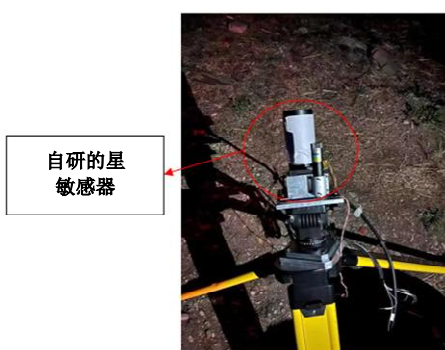
上述实验验证了本文算法的完整功能,然后将本算法烧录到自研星敏感器的Microsemi某一型号的FPGA硬件系统中,并用实际外场的观星实验进行实际验证。外场观星实验见图10。

由表5可知,本文算法可在该星敏感器上进行准确的星点质心坐标提取,并且将星图中两个异常目标剔除,证明本文算法可以在星敏感器中可靠地工作。通过外场观星实验可知,该星敏感器可以准确地进行星图识别与姿态解算,数据更新频率可以达到20 Hz左右,进一步证明了本算法的准确性与快速性。

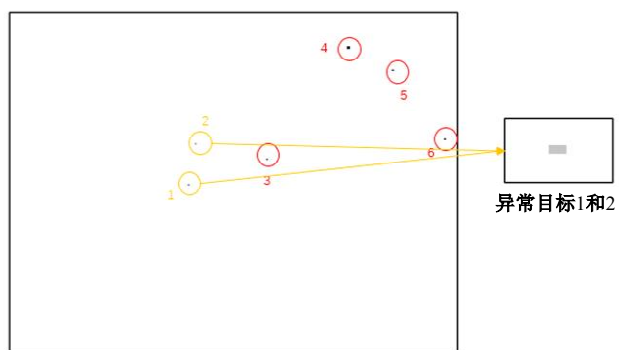
表 5 外场观星星点提取结果

Table 5 Extraction of star points from field observation

目标 序号	MATLAB提取 质心坐标/像素		本文算法提取 质心坐标/像素	
	X	Y	X	Y
1	510.000 000	517.500 000	异常星点目标	
2	530.500 000	395.000 000	异常星点目标	
3	736.535 874	439.867 713	736.535 874	439.867 713
4	970.006 906	104.754 143	970.006 906	104.754 143
5	1 097.141 509	172.858 490	1 097.141 509	172.858 490
6	1 245.315 420	380.996 495	1 245.315 420	380.996 495



(a)星敏感器



(b)拍摄的星图(反色处理)

图 10 外场观星实验

Fig. 10 Field stargazing experiment

### 3 结束语

在星图中取用不同星点目标数量、异常星点目标的条件下,对本文算法进行了验证实验。实验结果证明本文算法可以准确地提取星点质心坐

标,有效地剔除异常目标。在提取速度方面,当FPGA接收完一帧图像后,只需延时 $2\ \mu\text{s}$ 左右即可完成星点质心提取。该算法通过FPGA并行处理的优势,不需要将星图缓存,在FPGA中只需扫描一次图像后,延迟若干个时钟周期即可直

接输出星点质心的坐标信息,处理速度相比于ARM中进行星点质心提取提升显著;改进了标号合并与回收过程,在星点质心计算时加快了寻址速度,减少寄存器的资源占用率。利用外场观星实验,将本文算法编程到自研的星敏感器中验证了可行性与可靠性,表明本文算法可应用于实际工程中。后面的研究会进一步优化算法的组合逻辑,减少资源的消耗并提高算法的处理速度。

#### 参考文献:

- [1] Liebe C C. Star trackers for attitude determination [J]. Aerospace and Electronic Systems Magazine, IEEE, 1995, 10(6): 10-16.
- [2] 张刘, 何金航, 刘赫, 等. 基于等面积圆环和伴星夹角的星图识别算法[J]. 吉林大学学报: 工学版, 2024, 54(3): 821-827.  
Zhang Liu, He Jin-hang, Liu He, et al. Star map recognition algorithm based on equal area ring and companion star angle[J]. Journal of Jilin University (Engineering and Technology Edition), 2024, 54(3): 821-827.
- [3] 韩金良. 面向动态星图的星点匹配和端到端识别技术研究[D]. 长春: 中国科学院长春光学精密机械与物理研究所, 2022.  
Han Jin-liang. Research on star point matching and end-to-end recognition technology for dynamic star maps[D]. Changchun: Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, 2022.
- [4] 魏新国, 张广军, 江洁. 星敏感器中星图图像的星体细分定位方法研究[J]. 北京航空航天大学学报, 2003(9): 812-815.  
Wei Xin-guo, Zhang Guang-jun, Jiang Jie. Research on star subdivision localization method of star map image in star sensor[J]. Journal of Beijing University of Aeronautics and Astronautics, 2003(9): 812-815.
- [5] 卫昕. 高精度星敏感器星图识别算法研究[D]. 西安: 中国科学院西安光学精密机械研究所, 2021.  
Wei Xin. Research on star map recognition algorithm of high precision star sensor[D]. Xi'an: Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, 2021.
- [6] 于文波, 江洁. 低存储资源开销的多路快速星点质心提取方法[J]. 北京航空航天大学学报, 2018, 44(12): 2586-2594.  
Yu Wen-bo, Jiang Jie. Fast multi-channel star centroid extraction method with low storage resource overhead[J]. Journal of Beijing University of Aeronautics and Astronautics, 2018, 44(12): 2586-2594.
- [7] 郝雪涛, 江洁, 张广军. CMOS星敏感器图像驱动及实时星点定位算法[J]. 北京航空航天大学学报, 2005, 31(4): 381-384.  
Hao Xue-tao, Jiang Jie, Zhang Guang-jun. CMOS star sensor image driver and real-time star location algorithm[J]. Journal of Beijing University of Aeronautics and Astronautics, 2005, 31(4): 381-384.
- [8] 王静. 二值图像连通域的分段标记算法及实现[J]. 红外与激光工程, 2010, 39(4): 761-765.  
Wang Jing. Segmentation labeling algorithm and implementation in connected domain of binary image [J]. Infrared and Laser Engineering, 2010, 39(4): 761-765.
- [9] 邢飞, 尤政, 孙婷, 等. APS CMOS星敏感器系统原理及实现方法[M]. 北京: 国防工业出版社, 2017.
- [10] Azizabadi M, Behrad A, Ghaznavi-Ghouschi M B. VLSI implementation of star detection and centroid calculation algorithms for star tracking applications [J]. Journal of Real-time Image Processing, 2014, 9(1): 127-140.
- [11] 李葆华, 曾庆双, 张同双. 一种基于FPGA实现实时提取恒星星像坐标的方法[J]. 导航定位与授时, 2015, 2(1): 46-51.  
Li Bao-hua, Zeng Qing-shuang, Zhang Tong-shuang. An obtaining star locations method with FPGA for star sensor[J]. Navigation Positioning and Timing, 2015, 2(1): 46-51.
- [12] 樊巧云, 张广军. 离散噪声图像的光斑质心算法及其硬件实现[J]. 光学精密工程, 2011, 19(12): 2992-2998.  
Fan Qiao-yun, Zhang Guang-jun. Spot centroiding algorithm for discrete noise image and its hardware implementation[J]. Optics and Precision Engineering, 2011, 19(12): 2992-2998.
- [13] 鹿瑞, 武延鹏. 动态拖尾星图模拟算法研究[J]. 空间控制技术与应用, 2016, 42(4): 57-62.  
Lu Rui, Wu Yan-peng. Research on dynamic trailing star map simulation algorithm[J]. Space Control Technology and Application, 2016, 42(4): 57-62.
- [14] 提舒雯. 基于星敏感器的星图预处理与星点提取技术研究[D]. 哈尔滨: 哈尔滨工程大学智能科学与工程学院, 2020.  
Ti Shu-wen. Research on star map preprocessing and star point extraction technology based on star sensor [D]. Harbin: College of Intelligent Systems Science and Engineering, Harbin Engineering University, 2020.