

A parallel pipeline connected-component labeling method for on-orbit space target monitoring

LI Zongling^{1,2}, ZHANG Qingjun³, LONG Teng¹, and ZHAO Baojun^{1,*}

1. School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China; 2. Institute of Spacecraft System Engineering, China Academy of Space Technology, Beijing 100094, China; 3. Institute of Remote Sensing Satellite, China Academy of Space Technology, Beijing 100094, China

Abstract: The paper designs a peripheral maximum gray difference (PMGD) image segmentation method, a connected-component labeling (CCL) algorithm based on dynamic run length (DRL), and a real-time implementation streaming processor for DRL-CCL. And it verifies the function and performance in space target monitoring scene by the carrying experiment of Tianzhou-3 cargo spacecraft (TZ-3). The PMGD image segmentation method can segment the image into highly discrete and simple point targets quickly, which reduces the generation of equivalences greatly and improves the real-time performance for DRL-CCL. Through parallel pipeline design, the storage of the streaming processor is optimized by 55% with no need for external memory, the logic is optimized by 60%, and the energy efficiency ratio is 12 times than that of the graphics processing unit, 62 times than that of the digital signal processing, and 147 times than that of personal computers. Analyzing the results of 8756 images completed on-orbit, the speed is up to 5.88 FPS and the target detection rate is 100%. Our algorithm and implementation method meet the requirements of lightweight, high real-time, strong robustness, full-time, and stable operation in space irradiation environment.

Keywords: Tianzhou-3 cargo spacecraft (TZ-3), connected-component labeling (CCL) algorithms, parallel pipeline processing, on-orbit space target detection, streaming processor.

DOI: [10.23919/JSEE.2022.000107](https://doi.org/10.23919/JSEE.2022.000107)

1. Introduction

Binary image has a strong expression on the spatial relationship of pixels because of its simple mode, and plays an important role in image analysis and recognition [1]. Much image information analysis is transformed into binary image analysis in practical application finally, and binarization plus morphology can solve the problems of target extraction in many scenes [2,3]. The most important method of binary image analysis is connected-com-

ponent labeling (CCL) by marking the pixel “1” in the binary image that each individually connected region forms an identified block, and further obtains the geometric parameters of these blocks, such as contour, circumscribed rectangle centroid, moment invariants and so on, so as to extract the interest targets [4–6].

As a basic graphic processing method, image segmentation includes threshold segmentation [7,8], region growth segmentation [9,10], edge segmentation [11], graph theory segmentation [12], and machine learning segmentation [13,14], for different methods have different segmentation effects. Therefore, it is necessary to select and optimize the method according to the specific scene and meeting the design requirements.

CCL is the basic algorithm for military target detection and tracking, industrial product monitoring, traffic intersection monitoring and other application scenarios [15–17]. After decades of development, scholars have conducted a lot of research and achieved rich findings at home and abroad. At present, the existing fast CCL algorithms can be roughly divided into two categories: (i) Methods based on equivalence. This kind of method needs to search the image at least twice from left to right and from top to bottom, which record and sort out the equal relationship between temporary labels. According to different scanning units, such methods can be divided into the run length scanning method [18,19], the pixel scanning method [20], and the block scanning method [21]. The improvement of such algorithms mainly focuses on improving the efficiency of label equal relationship process and image access [22–24]. (ii) Methods based on region growth [25,26]. This kind of method does not record and sort out the equal relationship between temporary labels, which only needs one scan to complete CCL and does not have regularity for image scanning, so it is difficult to realize in parallel or hardware acceleration [27–29].

Manuscript received March 03, 2022.

*Corresponding author.

In the aspect of algorithm engineering, it is mainly realized by embedded platforms such as graphic processing unit (GPU), digital signal processing (DSP), field program gate array (FPGA), application specific integrated circuit (ASIC), or system-on-a-chip (SoC). Papers [30,31] used GPU to realize the CCL algorithm and achieved good acceleration effect that is the friendly development environment based on compute unified device architecture (CUDA), which is convenient for algorithm implementation and parallel acceleration. However, GPU has high power consumption and poor energy efficiency, which is difficult to meet the requirements of full-time stable work in space irradiation environment. The TI DSP DM6437 was used [32] to realize the CCL algorithm, which has the characteristics of simple programming and high development efficiency that is a typical embedded implementation method. Limited by the computing power of DSP, it is difficult to ensure the real-time effect of realizing complex algorithms. ASIC or SoC has been used to realize the CCL algorithm [33–35], which has the characteristics of high operation efficiency and good energy efficiency ratio. However, high cost and high customization limit their application range.

The high flexibility and moderate energy efficiency ratio of FPGA is an important technical way for the implementation of complex algorithms, which has attracted extensive attention of scholars at home and abroad. Acceleration of the design of FPGA hardware for existing or new CCL algorithms has been proposed [36–40] but there exist some deficiencies and limitations. The algorithm in [36] and its FPGA design can label high-speed video images for specific scenes quickly but not universal. The traditional two scanning method was accelerated in [37] with FPGA hardware, but it requires large memory to buffer the intermediate image and twice the pixel clock to process the equal relationship. The One Scan CCL algorithm based on FPGA hardware acceleration proposed in [38] must use the line blanking time in the image scanning process to deal with the equal relationship of temporary labels, which is not applicable to the image scanning method without line blanking. A real-time embedded hardware CCL method based on linked list run was proposed in [39], which occupies less than 25% of hardware resources than similar methods. However, it is difficult for the algorithm to guarantee the calculation cycle and it can not meet strong real-time application scenarios. A CCL algorithm based on run length and FPGA implementation method was proposed in [40], which is not suitable for real-time processing by image transmission due to its strong serialization structure. Therefore, it is an effective way to solve the real-time problem to meet the application requirements of specific

scenarios, customize the optimization algorithm, and accelerate the real-time implementation through parallel and pipeline.

At present, there is little research on the application scenario of space target extraction in the scene of large size, huge amount of point targets, high complexity, strong dispersion, and low signal to noise ratio (SNR). The on-orbit deploy algorithm needs to meet the requirements of strong real-time, high reliability, and full-time work under the conditions of space irradiation environment and resource constraints, which poses a great challenge to the implementation method.

To solve the above mentioned problems, this paper starts from three aspects as follows: Firstly, a dynamic run length (DRL)-CCL algorithm customized and optimized for the scene of space target monitoring is proposed, which can reduce the reading, writing, and comparison operations of data greatly and achieve good real-time performance. Secondly, the peripheral maximum grey difference (PMGD) image segmentation method is optimized according to the imaging characteristics of space targets and the subsequent processing flow of space target detection and tracking. It reduces the generation of equivalences greatly, and further improves the real-time performance for DRL-CCL algorithm. Thirdly, the real-time implementation method based on stream processor is proposed according to the particularity of the on-orbit operation environment, and the functional correctness and performance advantages are proved by analyzing the on-orbit processing results from the on-orbit verification of Tianzhou-3 cargo spacecraft (TZ-3) carrying experiment.

The DRL-CCL algorithm designed in this paper is implemented in Virtex7-690T FPGA of Xilinx, with a running clock of 150 MHz. Through the carrying experiment of TZ-3, 8 756 images whose integration time is distributed in the range of 150–500 ms with a resolution of $4096 \times 4096 \times 16$ bit are processed, and the telemetry results are analyzed. The minimum processing time of single frame image is 115.448 ms, the maximum processing time is 169.872 ms, the processing speed is 5.88 FPS, and the data rate of load is 4 FPS, which meets the real-time requirements of data processing.

2. Algorithm principle

2.1 Basic method of CCL

In this paper, the basic flow of CCL algorithm is based on run length, which is mainly divided into three steps.

The first step is to record the runs, form a sequence of continuous white pixels in each line, which is called a run, and record its count (num_run), start pixel column coordinate position (start_run), end column coordinate

position (end_run), and row coordinate position (row_run).

As shown in Fig. 1, in the first row, get two runs that corresponding to (run,start_run,end_run,row_run) as (1,2,3,1) and (2,6,6,1). In the second row, get two runs (3,3,3,2) and (4,5,6,2). In the third row, get one run (5,4,4,3). In the fourth row, get three runs (6,1,2,4), (7,4,4,4), and (8,6,7,4). In the fifth row, get one run (9,5,5,5). After all the images are traversed, all runs are obtained.

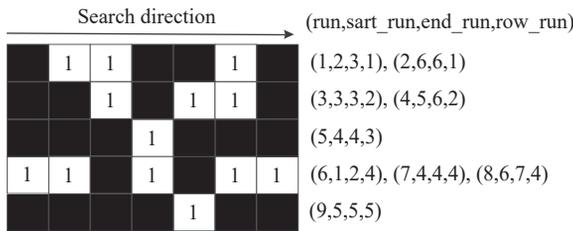


Fig. 1 Finding and recording runs

The second step is to label runs and generate equivalences. For the run in all rows except the first row, if it does not overlap with all the runs in the previous row, give it a new label; if it only overlaps with one run in the previous row, assign the label of that run in the previous row to it; if it overlaps with more than two runs in the previous line, assign the current run the minimum label of the connected run, and write the marks of these runs in the previous line into the equivalence, indicating that they belong to the same class.

As shown in Fig. 2, the two runs recorded in the first line are marked as 1 and 2. The two runs recorded in the second row have overlapping areas with the run in the previous line, so they are marked with the run in the previous row, i.e., 1 and 2. A group in the third row overlaps with the two groups in the previous row, so give it the smallest of the two, that is 1, and then write (1,2) into the equivalence. The fourth row records three runs, the first run has no overlapping area with the previous row, which is marked as 3. Similarly, the second run is marked as 1 and the third run is marked as 4. In the fifth row, a run of records overlaps with the two runs in the previous row, so give it the smallest of the two, that is 1, and then write (1,4) into the equivalence. After this step, mark the run number recorded in the first step to get a new label, and get a list of equivalences at the same time.

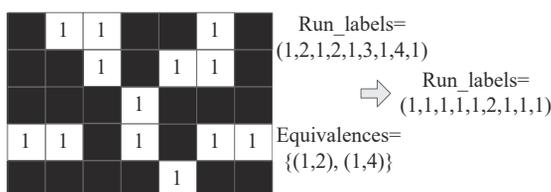


Fig. 2 Label runs and generate equivalences

The third step is to combine equivalences and generate the final label result. To convert equivalences into equivalent sequences, each sequence needs to be given the same label because they are equivalent. Starting from 1, give each equivalent sequence a label. Traverse the tags of the starting group, find the equivalent sequence, and give them new labels. Fill the tag of each run into the marking image to get the final label result.

The equivalences $\{(1,2),(1,4)\}$ are transformed into equivalent sequence: $1 \rightarrow 2 \rightarrow 4$ so that the maximum number of label tags, denoted as maxlabel, is 4. As shown in Fig. 3, points 1–4 are regarded as the nodes of the graph, and the equivalence (1,2) shows that there is a path between node 1 and node 2, and the resulting graph is an undirected graph, that is (1,2) actually contains (2,1). Therefore, we need to traverse the graph to find all connected graphs. The principle of image depth first traversal is adopted to find the equivalent sequence. Starting from node 1, it has two paths $1 \rightarrow 2$ and $1 \rightarrow 4$. There is no path after 2 and 4, and only 3 is left, which has never appeared in the equivalence that forms a separate sequence.

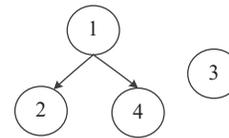


Fig. 3 Equivalences search graph (example)

Take the following equivalences as examples to better illustrate the depth first traversal algorithm (see Fig. 4): (1,2), (1,6), (3,7), (9,3), (8,1), (11,5), (10,8), (8,11), (12,11), (11,13). The above equivalences can be transformed into equivalent sequences:

- (i) List 1: $1 \rightarrow 2 \rightarrow 5 \rightarrow 6 \rightarrow 8 \rightarrow 10 \rightarrow 11 \rightarrow 12 \rightarrow 13$;
- (ii) List 2: $3 \rightarrow 7 \rightarrow 9$;
- (iii) List 3: 4.

Take $1 \rightarrow 13$ points as the nodes of the graph: Starting from node 1, which has three paths $1 \rightarrow 2$, $1 \rightarrow 6$, and $1 \rightarrow 8$. There are no paths behind 2 and 6, and 8 has two paths to 10 and 11, while 10 has no subsequent path, 11 has three paths to 5, 12, and 13. The equivalence sequence 1 is searched, the second equivalence sequence starts from 3, and then only two paths lead to 7 and 9. There is no path behind 7 and 9, and the equivalence sequence 2 is searched. Finally, only node 4 is left, which has not appeared in the equivalence so a separate sequence is formed (the maximum label of the clique in the second step is preset as 13), and the equivalence sequence is searched.

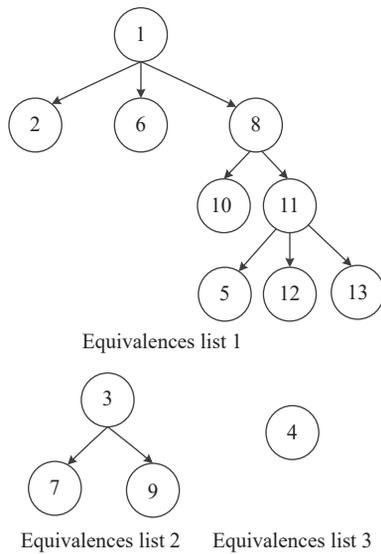


Fig. 4 Equivalences search graph (another example)

2.2 Optimization of CCL

In the engineering process of the CCL algorithm involved in this paper, the first two steps can form a fully parallel pipeline, and the processing time is fixed, which is not affected by the complexity of labeling graphics. In the third step, the image depth first traversal (DFS) method is used to realize the combined processing of equivalences. The processing time is related to depth and complexity of equivalences and the equivalent sequence. Moreover, this step is difficult to be accelerated by parallel and pipeline processing, which has become the bottleneck of the real-time performance of CCL algorithm.

The field of space target monitoring has the feature of wide imaging area, large size, huge amount of data, and the imaging characteristics of the target existing in the form of points. It shows the characteristics of high dispersion, complex edges, and low SNR, which is easy to form equivalence after segmentation. At the same time, the maximum number of clique tags $maxlabel$ is relatively large, which means that more cyclic comparisons and more time will be spent in the process of equivalences combine.

The image in the field of space target monitoring contains a large number of zeros, which indicates that there is no matching equivalences and does not need cyclic judgment. Therefore, a method of dynamically adjusting the priority traversal depth is proposed based on the characteristics of high dispersion and large number of point targets. Before the equivalences combine operation of each image, the priority traversal depth parameters are obtained in advance to dynamically adjust the number of cyclic comparisons, rather than making cyclic judgment according to the maximum number of labels, so as to

reduce the number of calculations and improve the real-time performance of the algorithm effectively.

2.3 PMGD image segmentation method

Image segmentation is the key step of target extraction, and a good segmentation method can remove the redundant information of the image greatly and facilitate the subsequent algorithm to extract effective information. Considering actual needs, the paper adopts the segmentation method based on threshold, which usually includes global threshold and regional threshold. According to the threshold acquisition method, it is divided into constant threshold processing and adaptive threshold processing, and adaptive threshold processing can be divided into empirical statistics, histogram statistical, maximum inter class variance segmentation, etc. Generally, appropriate image segmentation methods are selected according to different application scenarios.

The PMGD image segmentation method is proposed for the needs of space target detection, tracking, and positioning. Through the morphological plus mathematical processing method, the graphics complexity of the segmented binary image and the formation of equivalences are reduced, which is convenient for the subsequent target labeling.

The method flow is as follows: the original image S_{im} is segmented to obtain a binary image according to the following formula:

$$S_{binary} =$$

$$(S_{im} == S_{dilation}) \& [(S_{im} - S_{erosion}) \geq \alpha]. \quad (1)$$

$S_{dilation}$ in (1) is the dilation processing results of image S_{im} and obtained by the following formula:

$$S_{dilation} = \{z | [(B)_z \cap S_{im}] \subseteq S_{im}\} \quad (2)$$

where B is the structural element with the size of z , and the size of z is based on the pixel value correlation between the objects of interest. If z is too large, the adjacent objects will be merged, resulting in missed detection. If z is too small, it will affect the effect of dim target detection. Therefore, z is determined jointly according to the SNR of the camera image and the detection effect of the actual objects of interest, designed as seven in this paper.

$S_{erosion}$ in (1) is the erosion processing results of image S_{im} and obtained by the following formula:

$$S_{erosion} = \{z | [(B)_z \cap S_{im}^c] \subseteq S_{im}^c\} \quad (3)$$

where S_{im}^c is the complement of S_{im} .

In (1), α is the dynamic threshold and obtained by the following formula:

$$\alpha = \frac{\sqrt{\sum_{i=0}^{2^{\text{bit_width}}-1} (f_{\text{count}}(i) - f_{\text{mean}})^2}}{N \times M} + 1 \quad (4)$$

where bit_width is the effective bit width of the image with S pixels, generally ranging from 1 to 16. Therefore, $2^{\text{bit_width}}$ ranges from 2 to 65 536. $f_{\text{count}}(i)$ is the pixel i histogram statistical value of S_{im} , f_{mean} is the global mean of S_{im} , and N and M represent the size value of S_{im} .

The original image obtained by TZ-3 is shown in Fig. 5. The segmented histogram statistics segmentation result is shown in Fig. 6, where the number of connected component is 100, and the number of equivalences is 9.



Fig. 5 Original image

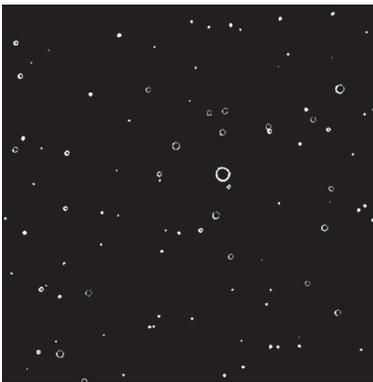


Fig. 6 Segmented histogram statistics image segmentation result

The segmentation result of PMGD is shown in Fig. 7 where the number of connected component is 70 and the number of equivalences is 3. The PMGD image segmentation method designed in the paper can greatly reduce the complexity of the segmented binary image and the generation of equivalences as can be seen from the comparison of the results, which retains a large amount of effective target information, and reduces the reading, writing, and judgment times of equivalences combine fundamentally.



Fig. 7 Image segmentation result of PMGD

3. Algorithm implementation

3.1 Streaming processor design

This paper proposes a design method of streaming processor based on parallel pipeline processing architecture in order to meet the requirements of space time-sensitive target detection and tracking, which adopts data-driven parallel computing mode and can realize the function of fast CCL in the space target monitoring image of large size, strong real time, high dispersion and complexity, and large scale point targets.

As shown in Fig. 8, the streaming processor is mainly composed of “binarization module” “run search and record module” “quad port random memory read-write control module” “run label and equivalences generation module” and “equivalences combine and label result output module”. According to the application requirements, different memories such as first input first output (FIFO), quad port random access memory (QPRAM), and double port random access memory (DPRAM) are used to cache and process data to meet the high-speed processing requirements.

The “binarization module” realizes the dynamic bit width adaptive adjustment of the input data and completes a 3×3 module median filter for the input image. The median filter of the template removes the noise in the image, then, completes the morphological expansion and corrosion processing on the denoised data in parallel, finally, compares the gray value with the pixel corresponding to the original data. When the pixel value at the corresponding position of the original image data and the expanded image data is equal, and the image data after subtracting the corrosion from the original image data is greater than or equal to the threshold, then the binary data corresponding to this position is “1”, otherwise the binary data is “0”.

The “run search and record module” mainly completes the preliminary positioning and recording of the run. The start line number and column number, and the end line number and column number with the value of “1” in the

input binary image are the preliminary position of the group.

The “run label and equivalences generation module” mainly completes the run merging between rows, records the run number that intersects with the two runs in the previous row as an equivalence, and determines the start line number and column number and the end line number

and column number of the run by row. If the run in row n intersects with the run in row $n-1$ ($n \geq 2$), then the run number in row $n-1$ is given to row n ; if the clique of row n intersects with the two cliques of row $n-1$, then the smallest clique number in row $n-1$ is assigned to row n , and the marks of the two cliques of row $n-1$ are written into the equivalences.

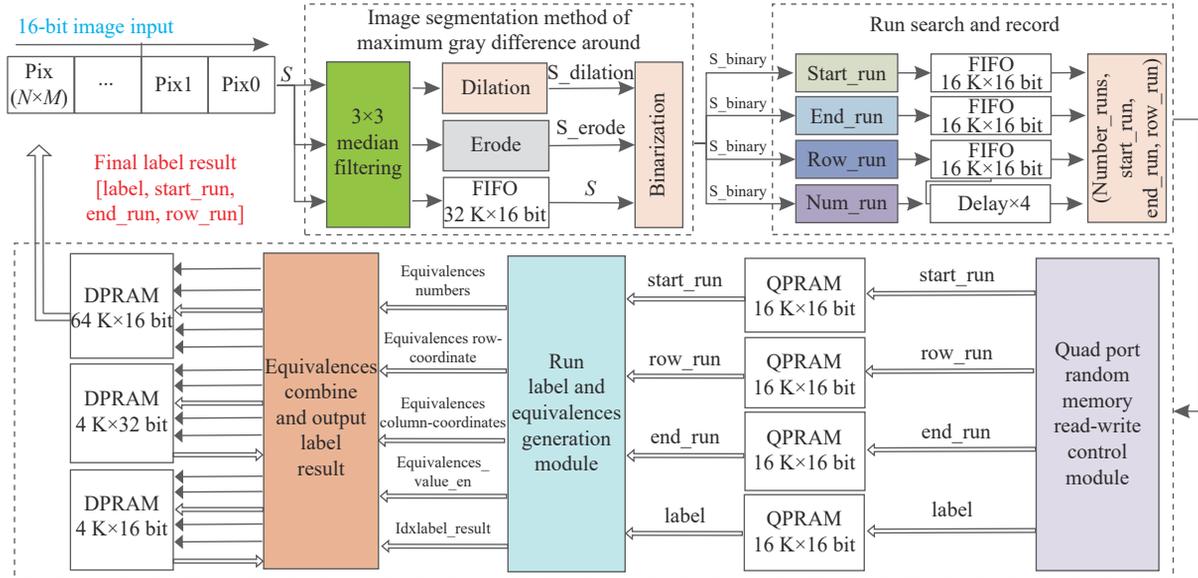


Fig. 8 Architecture of streaming processor

The “quad port random memory read-write control module” mainly completes the control of a large number of read-write operations involved in the generation of run tags and equivalences, simplifies the logic control, improves the parallelism of run tags and shortens the processing time.

The “equivalences combine and label result output module” is the key to label the connected component in the space target monitoring scene, which can reduce the tag retrieval times of sparse targets effectively and realize the combine of equivalences quickly by calculating the maximum value of the coordinate difference between the row and column of the equivalences as the depth value of the optimal traversal algorithm. Then the final label result is output.

3.2 Local pipeline optimization

It can be seen from the Subsection 3.1 that the streaming processor designed in this paper only has “run label and equivalences generation module” and “equivalences combine and label result output module”, which is not fully parallel pipelined architecture, and local pipelining optimization is needed to improve the degree of parallelism and efficiency of computing.

In the “run label and equivalences generation module”, four quad port read-write bus rams are used to store start_run, end_run, row_run, and label result. The write

operation of ram reuses one bus and the other write bus controls its invalidity at the enable end according to the characteristics of group label calculation, so as to save power consumption and ensure that there is no conflict between data write operations. The two read buses are controlled independently. They can read the data of two addresses at the same time in the same clock cycle. Therefore, they double the efficiency of read operation, simplify the control strategy, and optimize the calculation process, improving the parallelism of calculation and efficiency.

In the “equivalences combine and label result output module”, the core loop parameters are optimized by constructing the equivalence combine processing flow of local pipeline architecture. At the same time, the image binary segmentation algorithm is simplified according to the requirements of application scenarios and the number of equivalences is reduced at the source greatly, so as to improve the real-time processing performance.

3.3 Engineering realization

This paper designs a streaming processor, which compiles and synthesizes in the development platform Vivado2016.4. The hardware carrier is XC7VX690-TFFG1761-2, and the working frequency is 150 MHz.

As shown in Fig. 9, the core power consumption of the streaming processor combined with the application sce-

nario is no more than 1.538 W (excluding the power consumption of high-speed serial data transmission bus), occupies 2.89 Mbit of Block RAM (BRAM), and needs no external memory. It reduces the input/output (I/O) access of high power consumption greatly, and saves at least 5 W of power consumption (estimated according to two groups of 64 bit DDR3 memory and 800 MHz access clock). The mixed-mode clock manager (MMCM) power consumption is 0.333 W. The bit width of input image data adopts dynamic adaptive design, which can be up to 16 bit, and the maximum number of connected component is 65 536.

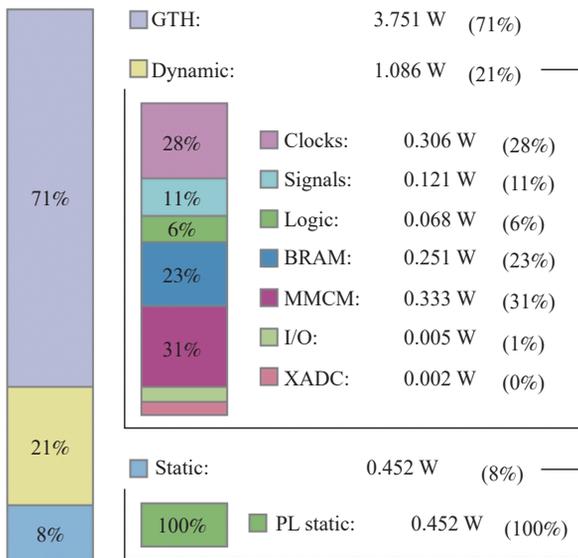


Fig. 9 Power consumption report

4. Experimental results

4.1 TZ-3 carrying experiment

The CCL streaming processor designed in this paper is

deployed on the space target monitor processing FPGA of TZ-3 carrying experiment. The processing architecture is shown in Fig. 10, and the streaming processor works in the FPGA that cooperates with TMS320C6678 to realize real-time detection and tracking of space targets, which obtains an excellent effect. V7-690T controls the camera through recommended standard 422 (RS422) interface, receives plane array camera data through camera link interface, and sends the processing results to DSPA and DSPB through serial bus and external bus after completing such as image segmentation and labeling by lable streaming processor. The DSPA and DSPB cooperate to complete target extraction and track association.

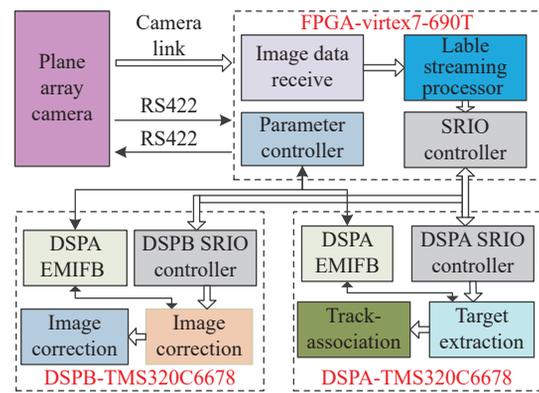


Fig. 10 TZ-3 experiment for space target detection

As shown in Table 1, the minimum processing time of single frame image is 115.448 ms, the maximum is 169.872 ms, the average processing speed reaches 5.88 FPS, and the maximum data rate of load is 4 FPS. The real-time performance meets the requirements of load data processing.

Table 1 Processing results of TZ-3 space target monitoring equipment

Data set (4096×4096)	CCD integral time/ms	Nums of runs			Nums of equivalences			Label time/ms		
		Maximum	Minimum	Average	Maximum	Minimum	Average	Maximum	Minimum	Average
Set 1 (2388 images)	150	18504	12567	14752	19	0	7	129.12	112.05	115.76
Set 2 (1934 images)	200	19531	13214	16321	35	0	12	141.21	112.05	127.85
Set 3 (1742 images)	250	25430	17654	22043	41	0	14	152.43	112.05	131.32
Set 4 (1543 images)	300	32567	21734	27621	55	1	23	167.95	119.05	142.92
Set 5 (1149 images)	500	45768	32165	35982	148	9	61	241.13	124.05	198.32

The target coordinates in the on-orbit processing result information are inversely labeled into the image data transmitted from TZ-3, and the effect is shown in Fig. 11.

The results show that the space target monitor has a good effect on moving target detection and tracking.

Compared with the stand results, the target coordinate position is consistent and the result is correct.

Among them, the integration time of CCD camera is 200 ms and the frame rate is 4 Hz. The image data of other integration times are analyzed and compared with

the ground processing results comprehensively. Although some single frame data is subject to missing detection and loss of dim targets, the detection rate of dim targets in images with low SNR reaches 100% after the accumulation of information in multiple frames (more than 16 frames), which proves that the PMGD image segmentation method has a good information retention rate.

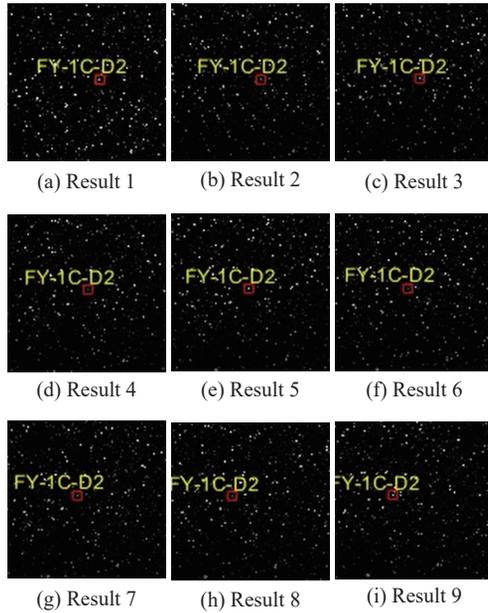


Fig. 11 TZ-3 process result

4.2 Performance analysis

Relying on the space target image data obtained by TZ-3 carrying equipment, the functions and performance of the algorithm and implementation method are analyzed and compared with the mainstream methods.

4.2.1 Performance of PMGD

As shown in Fig. 12, the image data (Fig. 12(a)–Fig. 12(c)) transmitted by the TZ-3 carrying equipment respectively uses the image segmentation method in this paper that the processing results are Fig. 12(d)–Fig. 12(f). Literature [7] proposed an image segmentation method based on *K*-means to calculate the segmentation threshold. The processing results are shown in Fig. 12(g). Literature [10] proposed an image segmentation method based on digital elevation region growth. The processing results are shown in Fig. 12(h). Literature [11] proposed an image segmentation method based on edge detection and peripheral coding. The processing results are shown in Fig. 12(i). As shown in Fig. 13, the image data transmitted by the TZ-3 carrying equipment (Fig. 13(a), Fig. 13(d), Fig. 13(g), Fig. 13(i)), uses the image segmentation method in this paper. The processing results of [7,10,11] are (Fig. 13(b), Fig. 13(e), Fig. 13(h), Fig. 13(k)). The label results are (Fig. 13(c), Fig. 13(f), Fig. 13(i), Fig. 13(l)).

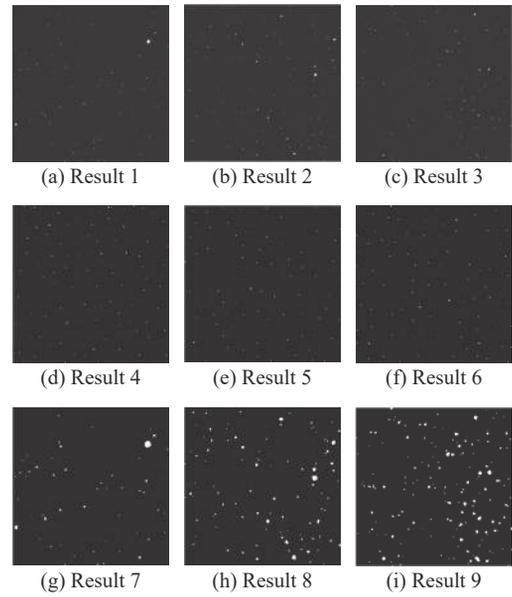


Fig. 12 Binary segmentation result

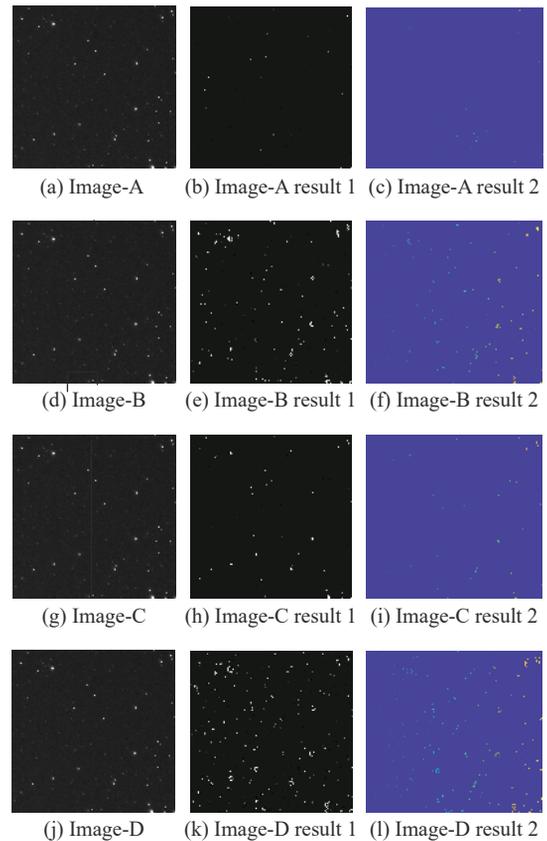


Fig. 13 Label result

For 1 000 images, the resolution is $4096 \times 4096 \times 16$ bit, and the integration time of CCD camera is in the range of 150–500 ms. The real data of TZ-3 carrying equipment is analyzed and compared, and the average values of key parameters such as the number of clusters and equiva-

lences, the retention rate of effective information are compared and analyzed. The specific motion detection method is to calculate the effective rate of target information:

$$IRR = \frac{N_{dt}}{N_{num}} \quad (5)$$

where IRR is the effective information retention rate, N_{dt} is the number of moving targets after segmentation, and N_{num} is the number of moving targets before segmentation.

The effective information retention rate results of different integration time of CCD camera and different segmentation methods are shown in Fig. 14.

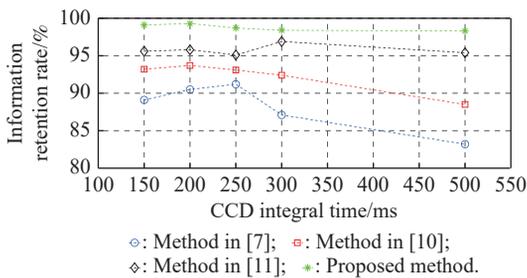


Fig. 14 IRR result of TZ-3 experiment

The number of runs with different integration time and different segmentation methods are shown in Fig. 15.

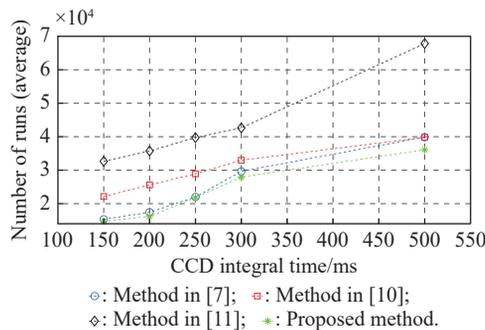


Fig. 15 Runs result of TZ-3 experiment

The equivalences results of different integration time and different segmentation methods are shown in Fig. 16.

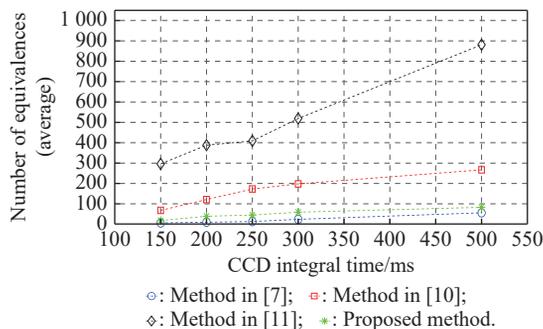


Fig. 16 Equivalences results of TZ-3 experiment

In this paper, the retention rate for effective information (time sensitive target) of PMGD image segmentation algorithm reaches more than 95% in images with different integration time, ranking first among the four methods, the number of runs ranked fourth, and the number of equivalences ranks third, and the comprehensive effect is best, which greatly reduces the consumption of resources such as calculation, memory and transmission of subsequent algorithms.

At the same time, the target missed in a single frame can be redetected through subsequent image frames. After calculation, after the accumulation of 16 frames of data, the detection rate of time-sensitive target can reach 100%.

4.2.2 Performance of the streaming processor

The performance analysis of streaming processor is mainly divided into two parts: on-orbit experiment and ground experiment.

Analyze the telemetry information of TZ-3, and obtain the internal temperature analysis of FPGA mainly through the X analog to digital converter (XADC) interface. The working environment temperature of FPGA is 10 °C, and the temperature data of continuous operation for 296 s is collected.

As shown in Fig. 17, Test 1 is the case that the load dose not work and there is no image data input, and the streaming processor is in the state of empty operation. The temperature curve of FPGA is balanced at about 20 °C finally that the temperature rise is 10 °C. Test 2 is shown that the load works normally and there is image data input that the input data is 1 FPS 4096×4096×16 bit data and the temperature curve of FPGA is balance at about 23 °C finally that the temperature rise is about 13 °C.

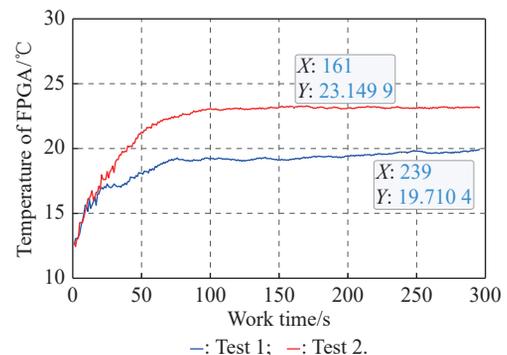


Fig. 17 Work temperature of TZ-3 experiment

Through the comparative analysis of experimental data, the streaming processor is a data-driven operation mode, and the dynamic power consumption is small, which is roughly estimated to be less than 300 mW. In the space environment, through the protection of space

heat dissipation measures, which can work stably for a long time and achieve hot balance.

The streaming processor is deployed on Virtex7 series FPGA for performance analysis. The experiment data is processed by the PMGD segmentation algorithm, and 1 000 binary images are obtained.

The methods of different platforms such as PC (Intel i7 3770), GPU (NVIDIA TX2) and DSP (TI TMS 320C6678) that compare with the main performance indicators include time, power and energy efficiency ratio (EER). The EER is obtained by comparing time with power.

As shown in Table 2, the EER of streaming processor method based on FPGA is 147 times than that of PC, 12 times than that of GPU and 62 times than that of DSP, which is suitable for on-orbit processing and other high EER requirements.

Table 2 Performance of different platform

Platform	Time/Hz	Power/W	EER/(Hz/W)
PC	1.94	77	0.025
GPU	4.54	15	0.302
DSP	0.89	15	0.059
Proposed method	5.88	1.6	3.675

Compared with FPGA implementation methods, the main performance indicators include the image size, the maximum number of connected components, the occupied logic and memory resources, and time.

Literature [35] proposed a method to realize CCL

based on SoC architecture and verified the performance on different types of FPGA. Literature [36] proposed a method to realize the CCL of 4K video stream based on FPGA parallel architecture, and verified the function and performance by face detection. Literature [40] used FPGA to realize a CCL method, which was applied to the field of remote sensing image processing. Literature [41] proposed a simplified architecture CCL for FPGA implementation, and analyzed the impact of image size and the number of connected component on hardware resource consumption. Literature [42] proposed a moving target detection and CCL algorithm, which realized the detection of people in a single FPGA.

The detailed comparison between the performance of the streaming processor and other methods is shown in Table 3. CCL based on FPGA and other programmable logic devices is the current mainstream engineering method. The performance of CCL algorithm is related to the scene such as the size, target and background complexity of the input image strongly. When the number of connected component and the size of input image is small, the efficiency and resource utilization of different CCL algorithms are all excellent. With the growth of connected component number and the increase of image size, the number of maximum connected component will increase significantly, and the corresponding hardware resources such as computing and memory that need to be reserved will increase significantly. Different methods have different emphases and optimization points that should be needed to meet the application requirements.

Table 3 Performance of different algorithms

Algorithm	FPGA	Image size	Num	SLICE	RAM/kB	External memory	Time/ms
Algorithm in [35]	XC7Z020	640×480	64	481	0.375	DDR3	1.75
	XC7Z020	640×480	128	699	0.875	DDR3	1.74
	XC7Z020	640×480	256	1 072	2	DDR3	1.94
	XC7Z020	640×480	512	1 699	4.5	DDR3	2.16
	XC7Z020	640×480	1 024	3 226	10	DDR3	2.44
	XC7Z020	2 048×2 048	64	784	0.375	DDR3	21.55
	XC7Z020	2 048×2 048	256	1 439	2	DDR3	27.18
	XC7Z020	2 048×2 048	1 024	3 688	10	DDR3	33.01
	XC7VX1140T	2 048×1 536	8 192	28 886	104	DDR3	21.51
Algorithm in [36]	ZCU104	3 840×2 160	<1 024	55 040	70	DDR4	16.67
Algorithm in [40]	VIRTEXII	1 024×1 024	4 096	432	130	No need	305
Algorithm in [42]	XC7K325T	256×256	—	5 943	108	No need	—
	XC7K325T	640×480	—	8 243	156	No need	—
	XC7K325T	7 680×4 320	—	24 510	548	No need	—
Algorithm in [43]	XC4VLX160	640×480	<1 024	2 510	76	Synchronous dynamic random access memory	28.5
Proposed algorithm	XC7VX690T	4 096×4 096	65 536	25 877	361	No need	170.07

Different from other CCL processing scenarios, the streaming processor needs to meet the application scenarios of high real-time, strong dispersion, local high complexity and massive point target label processing, as well as low power consumption and fast processing speed. Therefore, it is optimized from the following aspects. No external memory is used to reduce power consumption. Pipeline processing architecture is adopted to reduce storage resources, which is only 45% of that in [35]. Optimize control logic and reduce logic resources, which is only 40% of that in [36]. Through comparative analysis, compared with other similar methods, it is suitable for the scene of spatial massive point target processing with superior performance and good effect.

5. Conclusions

Space-borne hardware resources are limited strongly, which requires space target oriented on-orbit detection and tracking algorithms to meet the requirements of lightweight, high real-time performance, and strong robustness. Space target imaging has the characteristics of large size, huge number of point targets, strong dispersion, local high complexity, and low SNR. The algorithm needs to meet the demanding requirements of full-time and stable operation in space irradiation environment, which poses a great challenge to the implementation method.

Firstly, the PMGD image segmentation method by studying the characteristics of space target imaging, detection, and tracking tasks is proposed, which is simple to be implemented and has a high effective information retention rate. And it reduces the pressure of calculation, storage, and transmission of subsequent algorithms greatly. Secondly, the CCL algorithm is optimized, and the cyclic comparison, read and write times are reduced greatly and the real-time performance is improved. Finally, an energy-efficient streaming processor that is deployed on FPGA based on multi-level cache architecture is constructed and the on-orbit verification is completed on TZ-3, which verifies the functional correctness and performance superiority of the algorithm effectively. By configuring the refresh function, FPGA verifies long-term stable operation of the streaming processor in space irradiation environment effectively.

The streaming processor can be designed as radiation resistant ASIC or SoC by the needs of future work, which will further improve the operation performance and can be used in the fields of space target monitoring.

References

- [1] GUL T, GUZIN U, BESTE U, et al. Ciratefi based copy move forgery detection on digital images. *Multimedia Tools and Applications*, 2022, 81: 22867–22902.
- [2] ONSHAUNJIT J, SRINONCHAT J. Algorithmic scheme for concurrent detection and classification of printed circuit board defects. *CMC-Computers, Materials & Continua*, 2022, 71(1): 355–367.
- [3] BARIS G, GURAY S. Recognition of dynamic objects from UGVs using interconnected neural network-based computer vision system. *Automatika*, 2022, 63(2): 244–258.
- [4] KH A, GDW B, KPY A. Informed component label algorithm for robust identification of connected components with volume-of-fluid method. *Computers & Fluids*, 2020, 197(1): 104373.
- [5] YANG G, ZHENG W P, CHE C H. Graph-based label propagation algorithm for community detection. *International Journal of Machine Learning and Cybernetics*, 2020, 11(6): 1319–1329.
- [6] YAN J, ZHOU L M, RAGHAVENDRA A. Observed changes in fire patterns and possible drivers over Central Africa. *Environmental Research Letters*, 2020, 15(9): 0940b8.
- [7] ZHENG X, LEI Q Y, YAO R, et al. Image segmentation based on adaptive k -means algorithm. *Journal on Image and Video Processing*, 2018, 2018: 68.
- [8] TOSUN A B, GUNDUZ-DEMIR C. Graph run-length matrices for histopathological image segmentation. *IEEE Trans. on Medical Imaging*, 2011, 30(3): 721–732.
- [9] GARCIA-UGARRIZA L, SABER E, AMUSO V, et al. Automatic color image segmentation by dynamic region growth and multimodal merging of color and texture information. *IEEE Trans. on Image Processing*, 2009, 18(10): 2275–2288.
- [10] ZHU H C, HUANG W, LIU H Y. Loess terrain segmentation from digital elevation models based on the region growth method. *Physical Geography*, 2017, 39(1): 51–66.
- [11] UEMURA T, KOUTAKI K, UCHIMURA K. Image segmentation based on edge detection using boundary code. *International Journal of Innovative Computing, Information & Control*, 2011, 7(10): 6073–6083.
- [12] LONG J W, FENG X, ZHU X F, et al. Efficient superpixel-guided interactive image segmentation based on graph theory. *Symmetry*, 2018, 10(5): 169–175.
- [13] VAN O A, IKRAM MA, VERNOOIJ M W, et al. Transfer learning improves supervised image segmentation across imaging protocols. *IEEE Trans. on Medical Imaging*, 2015, 34(5): 1018–1030.
- [14] CHEN L, BENTLEY P, MORI K, et al. DRINET for medical image segmentation. *IEEE Trans. on Medical Imaging*, 2018, 37(11): 2453–2462.
- [15] HE L F, CHAO Y Y, SUZUKI K. Two efficient label-equivalence-based connected-component labeling algorithms for 3-D binary images. *IEEE Trans. on Image Processing*, 2011, 20(8): 2122–2134.
- [16] LI Z L, ZHANG Q J, LONG T, et al. Ship target detection and recognition method on sea surface based on multi-level hybrid network. *Journal of Beijing Institute of Technology*, 2021, 30(S1): 1–10.
- [17] JASIEWICZ J, STEPINSKI T F. Geomorphons a pattern recognition approach to classification and mapping of landforms. *Geomorphology*, 2013, 182(1): 147–156.

- [18] WU K S, OTOO E, SUZUKI K. Optimizing two-pass connected-component labeling algorithms. *Pattern Analysis & Applications*, 2009, 12(2): 117–135.
- [19] DIAZ-DEL-RIO F, SANCHEZ-CUEVAS P, MOLINA-ABRIL H, et al. Parallel connected-component-labeling based on homotopy trees. *Pattern Recognition Letters*, 2020, 131(4): 71–78.
- [20] BATAINEH B. A fast and memory-efficient two-pass connected-component labeling algorithm for binary images. *Turkish Journal of Electrical Engineering and Computer Sciences*, 2019, 27(2): 1243–1259.
- [21] ZHAO X, HE L F, YAO B, et al. A new connected-component labeling algorithm. *IEEE Trans. on Information & Systems*, 2015, 98(11): 2013–2016.
- [22] HE L F, REN X W, GAO Q H, et al. The connected-component labeling problem: a review of state-of-the-art algorithms. *Pattern Recognition*, 2017, 70(1): 25–43.
- [23] DOUBE M. Multithreaded two-pass connected components labelling and particle analysis in Image. *Royal Society Open Science*, 2021, 8(1): 201784.
- [24] HE L F, CHAO Y Y, SUZUKI K. A run-based two-scan labeling algorithm. *IEEE Trans. on Image Processing*, 2008, 17(5): 749–756.
- [25] JEONG J W, LEE G B, LEE M J, et al. A single-pass connected component labeler without label merging period. *Journal of Signal Processing Systems for Signal, Image, and Video Technology*, 2016, 84(2): 211–223.
- [26] YANG Y, ZHANG D P. A novel line scan clustering algorithm for identifying connected components in digital images. *Image and Vision Computing*, 2003, 21(5): 459–472.
- [27] CHANG F, CHEN C J, LU C J. A linear-time component-labeling algorithm using contour tracing technique. *Computer Vision and Image Understanding*, 2004, 93(2): 206–220.
- [28] HU Q M, QIAN G Y, NOWINSKI W L. Fast connected-component labeling in three-dimensional binary images based on iterative recursion. *Computer Vision & Image Understanding*, 2005, 99(3): 414–434.
- [29] KLAIBER M J, BAILEY D G, SIMON S. Comparative study and proof of single-pass connected components algorithms. *Journal of Mathematical Imaging and Vision*, 2019, 61(8): 1112–1134.
- [30] ASAD P, MARROQUIM R, SOUZA A L E L. On GPU connected components and properties: a systematic evaluation of connected component labeling algorithms and their extension for property extraction. *IEEE Trans. on Image Processing*, 2019, 28(1): 17–31.
- [31] ALLEGRETTI S, BOLELLI F, GRANA C. Optimized block-based algorithms to label connected components on GPUs. *IEEE Trans. on Parallel and Distributed Systems*, 2020, 31(2): 423–438.
- [32] LIN D T, LIN M C, HUANG K Y. Real-time automatic recognition of omnidirectional multiple bar-codes and DSP implementation. *Machine Vision and Applications*, 2010, 22(2): 409–419.
- [33] TSAI T H, CHEN S W. Single-chip design for intelligent surveillance system. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 2018, 26(9): 1637–1646.
- [34] AZIZABADI M, BEHRAD A, GHAZNAVI-GHOUSHCHI M B. VLSI implementation of star detection and centroid calculation algorithms for star tracking applications. *Journal of Real-Time Image Processing*, 2014, 9(1): 127–140.
- [35] PERRI S, SPAGNOLO F, CORSONELLO P. A parallel connected component labeling architecture for heterogeneous systems-on-chip. *Electronics*, 2020, 9(2): 292.
- [36] KOWALCZYK M, CIARACH P, PRZEWLOCKA-RUS D, et al. Real-time FPGA implementation of parallel connected component labeling for a 4k video stream. *Journal of Signal Processing Systems*, 2021, 93(5): 481–498.
- [37] ITO Y, NAKANO K. Low-latency connected component labeling using an FPGA. *Journal of Foundations of Computer Science*, 2010, 21(3): 405–425.
- [38] KLAIBER M J, BAILEY D G, SIMON S. A single-cycle parallel multi-slice connected components analysis hardware architecture. *Journal of Real-Time Image Processing*, 2019, 16(4): 1165–1175.
- [39] TANG J W, SHAIKH-HUSIN N, SHEIKH U U, et al. A linked list run-length-based single-pass connected component analysis for real-time embedded hardware. *Journal of Real-time Image Processing*, 2018, 15(1): 197–215.
- [40] XIE Y Z, TAN X B, CHEN H. A new algorithm for connected components labeling. *Transactions of Beijing Institute of Technology*, 2012, 32(12): 1273–1278.
- [41] GONZLEZ C G, RAFALEW W. *Digital image processing*. 3rd ed. London: Pearson Education, 2011.
- [42] KLAIBER M J, BAILEY D G, BAROUD Y O, et al. A resource-efficient hardware architecture for connected component analysis. *IEEE Trans. on Circuits and Systems for Video Technology*, 2016, 26(7): 1334–1349.
- [43] APPIAH K, HUNTER A, DICKINSON P, et al. Accelerated hardware video object segmentation: from foreground detection to connected components labelling. *Computer Vision and Image Understanding*, 2010, 114(11): 1282–1291.

Biographies



LI Zongling was born in 1985. He is currently pursuing his Ph.D. degree in the School of Information and Electronics, Beijing Institute of Technology, Beijing, China. He is currently a senior engineer with the Institute of Spacecraft System Engineering. His research interests include satellite system design, space-borne computer design, and high-speed radar signal processing.

E-mail: leezl0519@163.com



ZHANG Qingjun was born in 1969. He received his Ph.D. degree from Huazhong University of Science and Technology from 1993 to 2005, he designed spacecraft surveying and control subsystem and designed spacecrafts' rendezvous and docking scheme as a chief designer with China Aerospace Science and Technology Corporation. Since 2012, he has been a chief designer with the

Beijing Institute of Spacecraft System Engineering, for Landsat Series Satellites and Gaofen-3. His research interest includes synthetic aperture radar satellite system design and application.

E-mail: ztzhangqj@163.com



LONG Teng was born in 1968. He received his B.S. degree from the University of Science and Technology of China, Hefei, China, in 1989, and M.S. and Ph.D. degrees from Beijing Institute of Technology (BIT), Beijing, China, in 1991 and 1995, respectively. He became a full professor at the Department of Electronic Engineering, BIT, in 2000. He was a visiting scholar with Stanford

University, Stanford, CA, USA, and University College London, London, U.K., in 1999 and 2002, respectively. From 2009 to 2016, he was the Dean of the School of Information and Electronics, BIT. Since 2019, he has been the President of BIT. He is an Academician of Chinese Academy of Engineering. He has authored or coauthored more than 300 articles. His research is mainly about synthetic aperture radar (SAR) systems and real-time digital signal processing, with applications to radar and communication systems.

E-mail: longteng@bit.edu.cn



ZHAO Baojun was born in 1960. He received his Ph.D. degree in electromagnetic measurement technology and equipment from Harbin Institute of Technology, Harbin, China, in 1996. From 1996 to 1998, he was a postdoctoral fellow at Beijing Institute of Technology (BIT), Beijing, China. Currently, he is a full professor, Vice Director of Laboratory and Equipment Manage-

ment, and Director of the National Signal Acquisition and Processing Professional Laboratory. His research is mainly about the theory and technology of multi-spectral image fusion, recognition and tracking, real-time compression, restoration and transmission of high-resolution images.

E-mail: zbj@bit.edu.cn