Argotec

Our Boost to Your Future

argotec

On-the-Fly Hardware-Accelerated Image Processing System for Target Recognition

35th Annual Small Satellite Conference

Presenter: Eugenio **SCARPA**

Co-authors: Emilio **FAZZOLETTO**

Logan, Utah, USA

2021, August 7th -12th

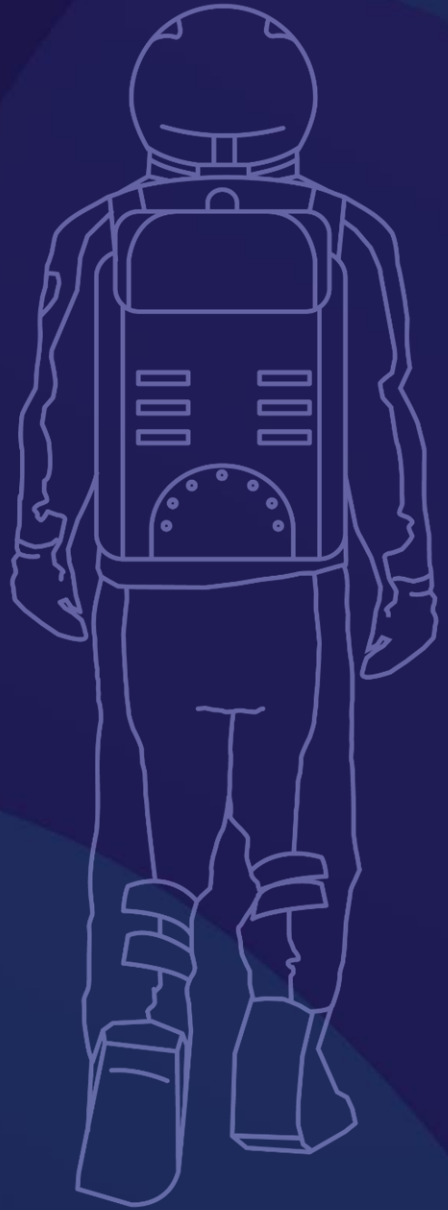
Authors

Eugenio Scarpa
FPGA Design Engineer

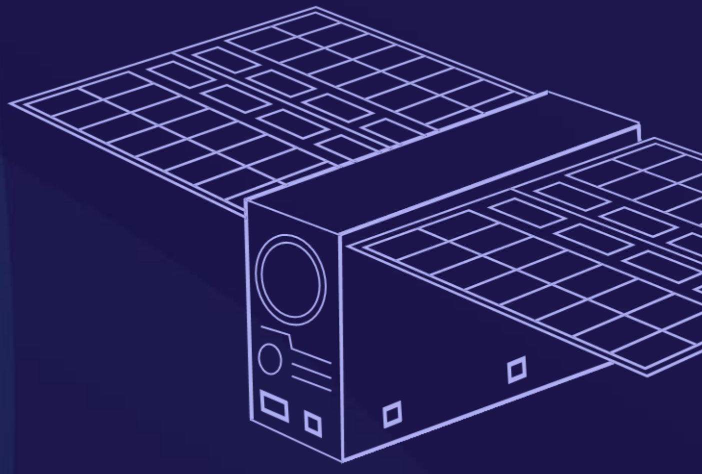
Emilio Fazzoletto
Head of Electronics Unit

Main contributors to ArgoMoon and LICIACube satellites FPGA design architectures

Summary



Introduction to Real-Time Image Processing in Space



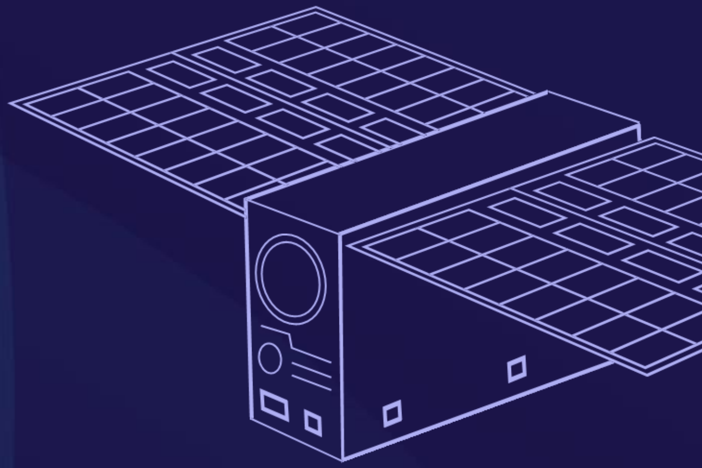
Introduction to Real-Time Image Processing in Space

- Image processing algorithms widely employed for various applications
  - Rendez-vous
  - Docking
  - Object tracking

- HW acceleration implementations improve performances
  - LICIACube satellite autonomous navigation based on image features

- This paper focused on an Imaging Subsystem creation for the LCC FERMI OBC

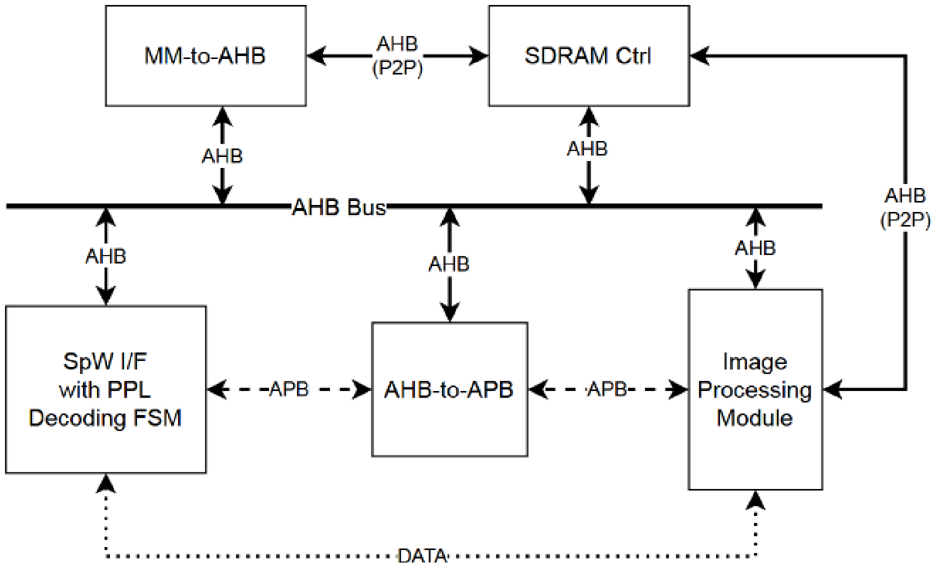
ARG Image Subsystem



ARG Image Subsystem

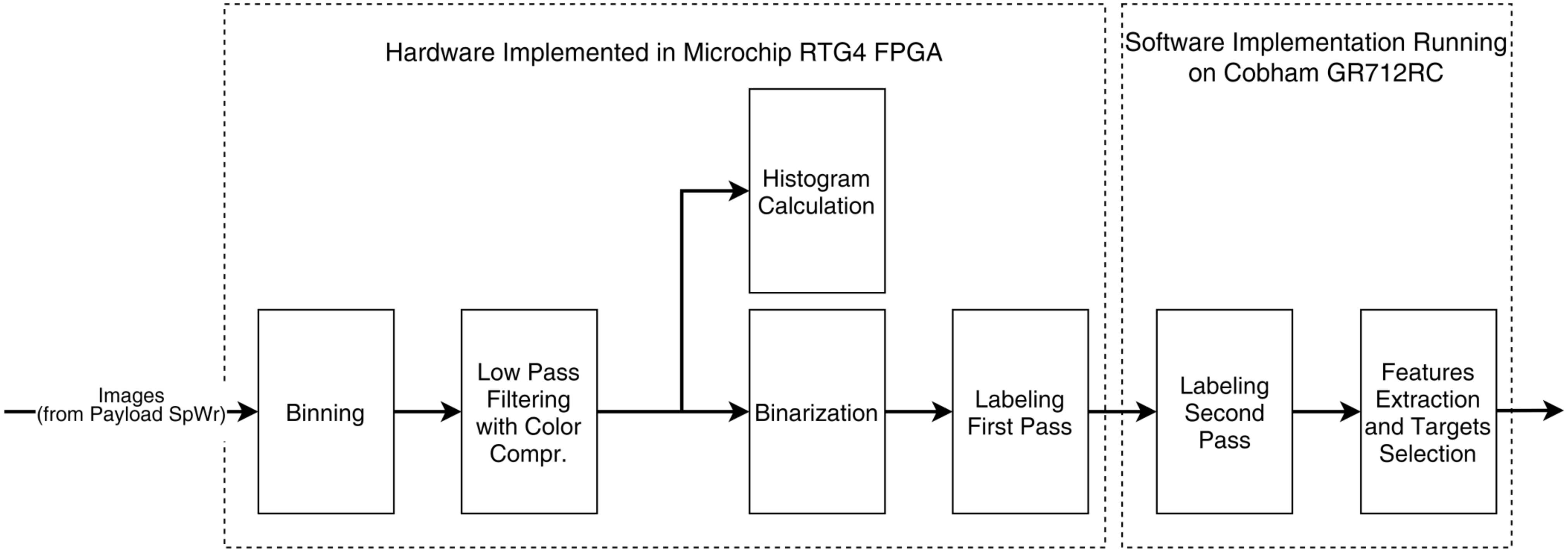
Background and Purpose

- Design developed to operate with a 4.2MP CMOS sensor
- Image size is 2048x2048 with 16-bit encoded pixels, 8MB transferred
- SW-implemented algorithms run in 10s on CPU
- FPGA solution increases performances and facilitates datapath



ARG Image Subsystem

HW/SW Block Diagram



ARG Image Subsystem

Pixel binning and Color Depth Compression

- Binning performs arithmetical average of four consecutive pixels

$$P'_{\left(\frac{r}{2},\frac{c}{2}\right)} = \frac{P_{(r,c)}+P_{(r,c+1)}+P_{(r+1,c)}+P_{(r+1,c+1)}}{4}$$

- Resulting image size on $R \times C$ input is $R/2 \times C/2$
- Core latency is 2 clock cycles for each pixel (1 ADD, 1 ADD+RSH)
- Design include 4 FIFOs to guarantee data coherency based the (R, C) indexes
- CDC consists of each pixel Least Significant Bits (LSBs) truncation
- Scale color dynamic from 12 bits to 8 bits
- Operation is performed combinatorially, no additional latency

ARG Image Subsystem

Low-Pass Filtering

- 2-D convolutional filter between image and user kernel
- LPF smooths objects edges, leveling pixels values in the area
- Background noise is filtered (Deep Space one-pixel-stars removed)
- Core receives 1 pixel per cycle
- Pipelined datapath, latency of 6 cycles

$$P'_{(r,c)} = \frac{A_{r-1}+A_r+A_{r+1}}{9}$$

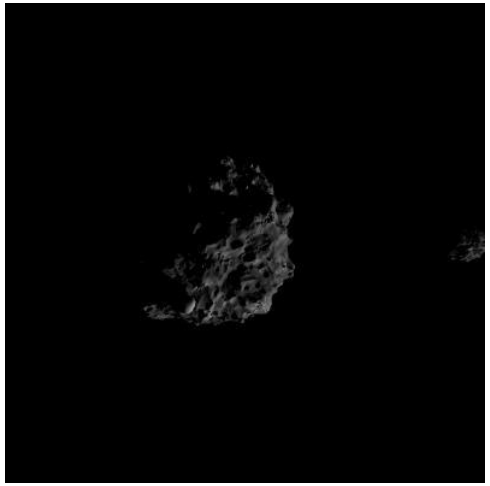
$$\begin{cases} A_{r-1} = P_{(r-1,c-1)} \cdot K_{(0,0)} + P_{(r-1,c)} \cdot K_{(0,1)} + P_{(r-1,c+1)} \cdot K_{(0,2)} \\ A_r = P_{(r,c-1)} \cdot K_{(1,0)} + P_{(r,c)} \cdot K_{(1,1)} + P_{(r,c+1)} \cdot K_{(1,2)} \\ A_{r+1} = P_{(r+1,c-1)} \cdot K_{(2,0)} + P_{(r+1,c)} \cdot K_{(2,1)} + P_{(r+1,c+1)} \cdot K_{(2,2)} \end{cases}$$

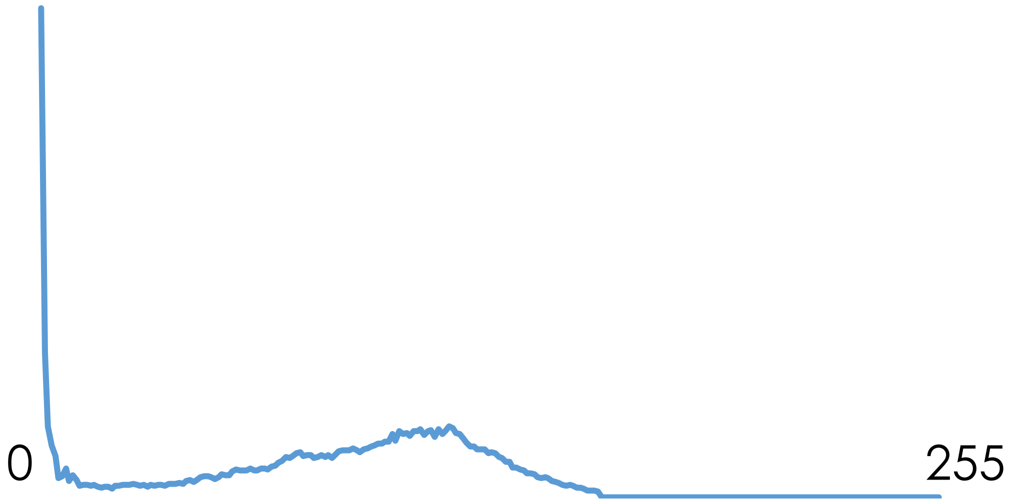
<table>
<tr><th>Cycle 0</th><th>Cycle 1</th><th>Cycle 2</th><th>Cycle 3</th><th>Cycle 4</th><th>Cycle 5</th></tr>
<tr><td>M0</td><td rowspan="2">A0</td><td rowspan="4">A4</td><td rowspan="8">A6</td><td rowspan="10">A7</td><td rowspan="11">dout</td></tr>
<tr><td>M1</td></tr>
<tr><td>M2</td><td rowspan="2">A1</td></tr>
<tr><td>M3</td></tr>
<tr><td>M4</td><td rowspan="2">A2</td><td rowspan="4">A5</td></tr>
<tr><td>M5</td></tr>
<tr><td>M6</td><td rowspan="2">A3</td></tr>
<tr><td>M7</td></tr>
<tr><td>M8</td><td rowspan="2">A4</td><td rowspan="2">A4</td><td rowspan="2">A4</td></tr>
<tr><td>0</td></tr>
<tr><td></td><td></td><td></td><td></td><td>DIV, 9</td></tr>
</table>

ARG Image Subsystem

Luminance Histogram and Threshold Computation

- Histogram generates from pixel values occurrences in the picture
- Each input data increases the related pixel accumulator
- Result is provided at end of image
- Accumlation compared with user threshold, returning Background Area





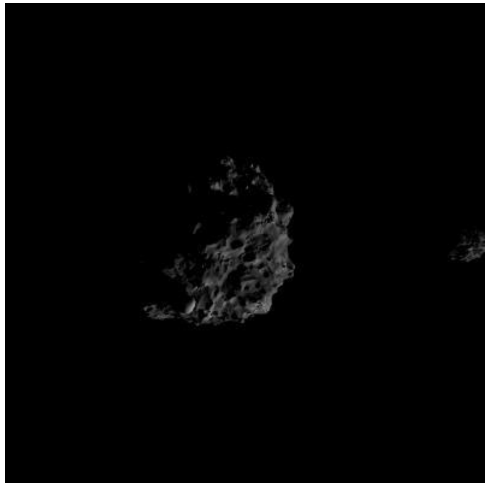
ARG Image Subsystem

Binarization

- Performed combinatorially in parallel to the luminance histogram
- Operates a comparison of input data wrt to BA threshold

$$P'_{(r,c)} = \begin{cases} 1, & P_{(r,c)} > T \\ 0, & P_{(r,c)} \leq T \end{cases}$$

- Processed pixels are either black (0) or white (1)
- If black, pixel becomes part of Deep Space background
- Target identification executed only on white pixels, care on threshold



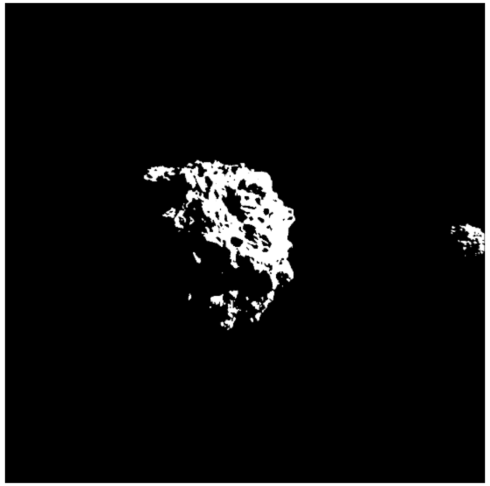
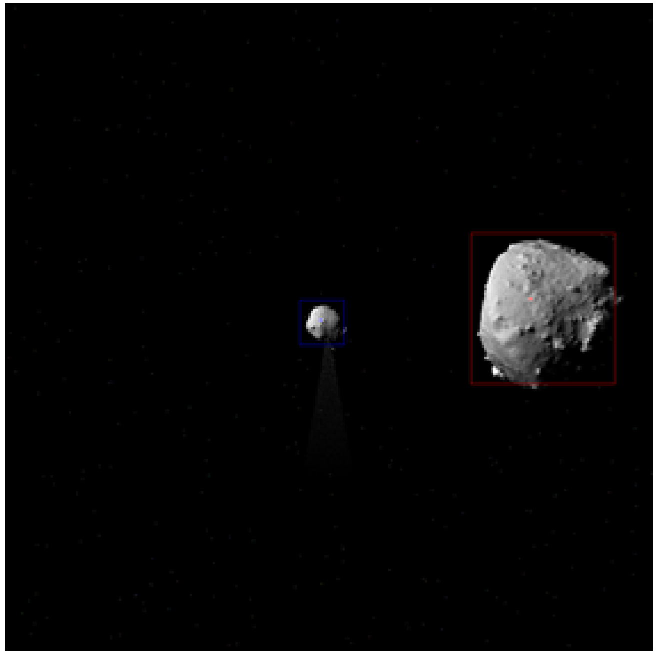
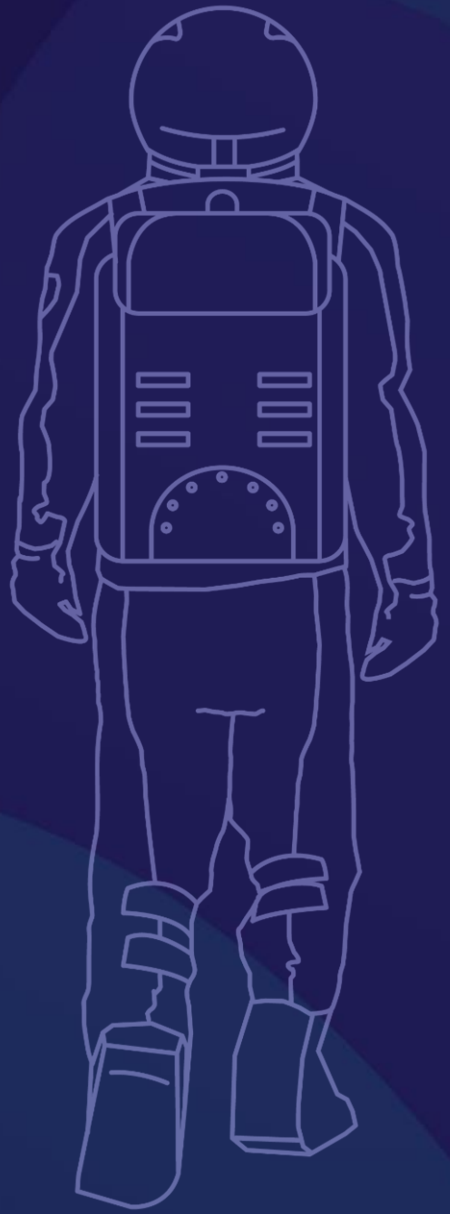


Image processing functionalities

Multi-Target Identification

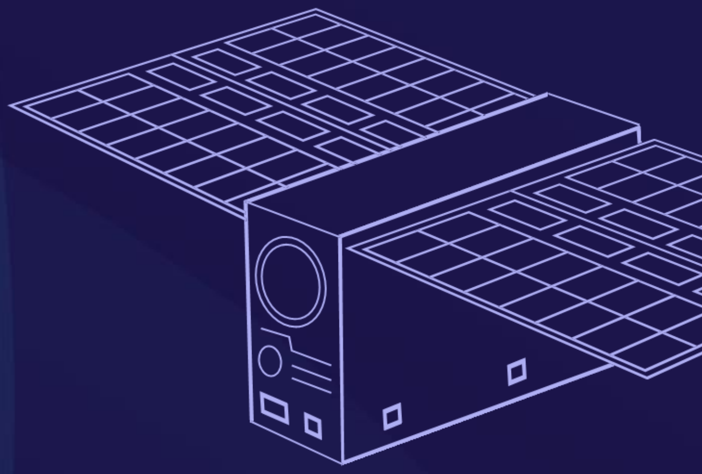
- Executed on binarized white pixels
- Recognizes single object by analyzing each pixel surrounding area
- Assigns a color (16-bit label) to the identified object
- Increments the assigned colors to allow unique object identification
- Receives 1 pixel for cycle, evolving on a FSM
- Evolves through Retrieve-Analyze-Update states
- Core integrates FIFOs for synchronization and data coherency





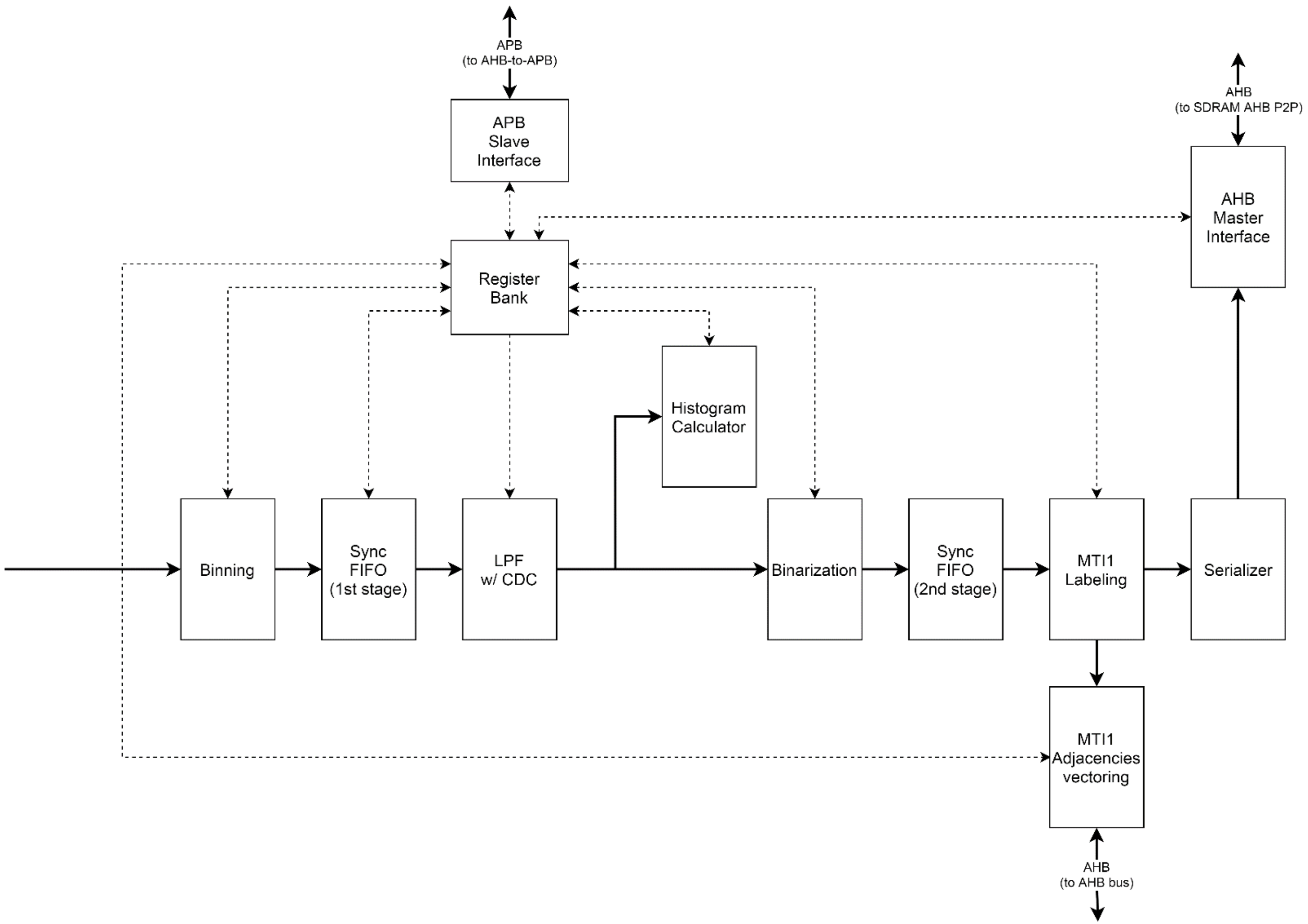
argotec

FPGA Implementation



FPGA Implementation

Datapath Architecture



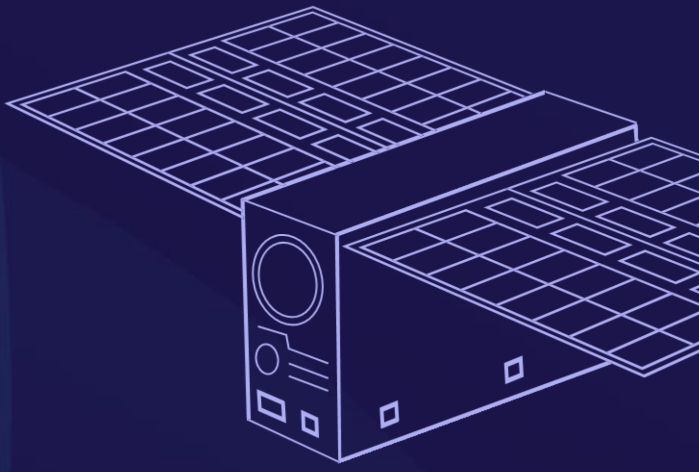
FPGA design results

Device resources occupation

- Microsemi RT4G150-CG1657B FPGA
- 4LUT occupation: 5.2%
- DFF occupation: 2.7%
- 1Kbit SRAM occupation: 17.1%
- 18Kbit SRAM occupation: 14.4%

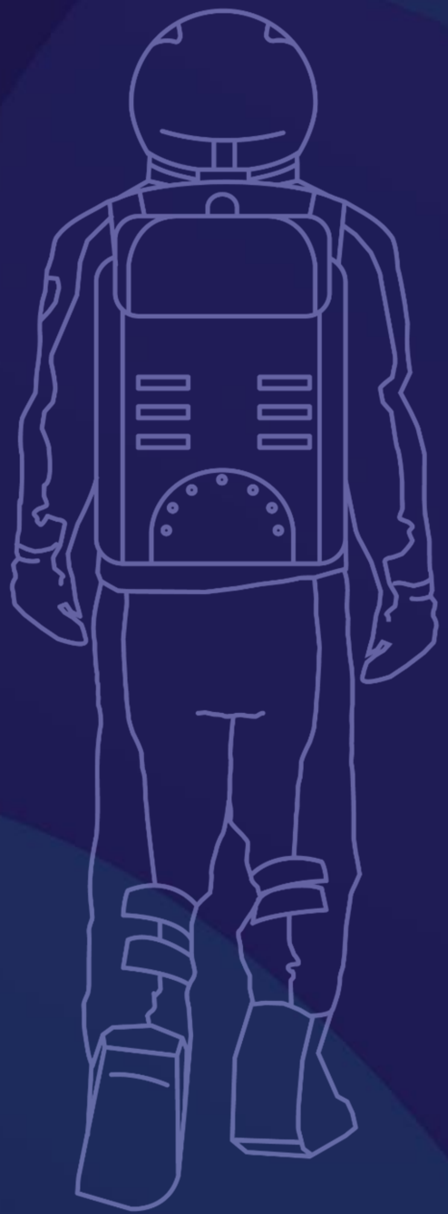
Module	Fabric 4LUTs	Fabric DFFs	uSRAM 1Kbit	LSRAM 18Kbit
Overall	7928	4085	36	30
Binning	1200	564	8	0
LPF	1467	445	36	0
HL	627	362	0	3
Binarization	306	211	0	0
MTI	1673	718	0	12

Theoretical Performances and Limitations

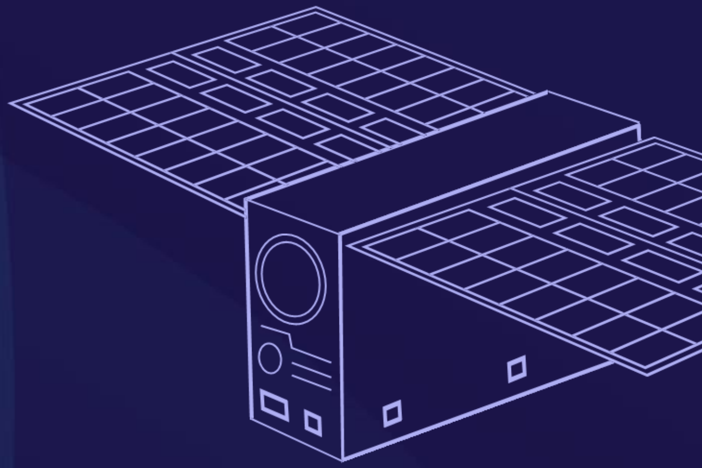


Theoretical Performances and Limitations

- All modules but MTI guarantee a throughput of 50Mpixel/s
- Chain bottleneck is the receiving SpW interface
- Maximum transfer throughput is 8Mpixel/s on the test application
- Streaming data coherency is guaranteed if $T_{SpW} < T_{min,IS}$
- MTI is the slowest core (5 clock cycles/processed pixel)
- Synchronization FIFOs inserted for guaranteeing data safety



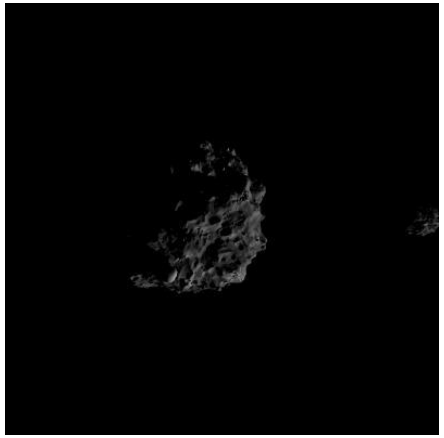
Testing and application in Space



Testing and Application

Testing Results

- Comparison include SW vs. HW execution
- SW run performed on a single SPARC V8 CPU at 50MHz with RTEMS
- HW run performed on target FPGA with system clock at 50MHz
- Test image from LCC mission simulator, improvement of 17.2x
- Maximum bandwidth limited on transmission interface

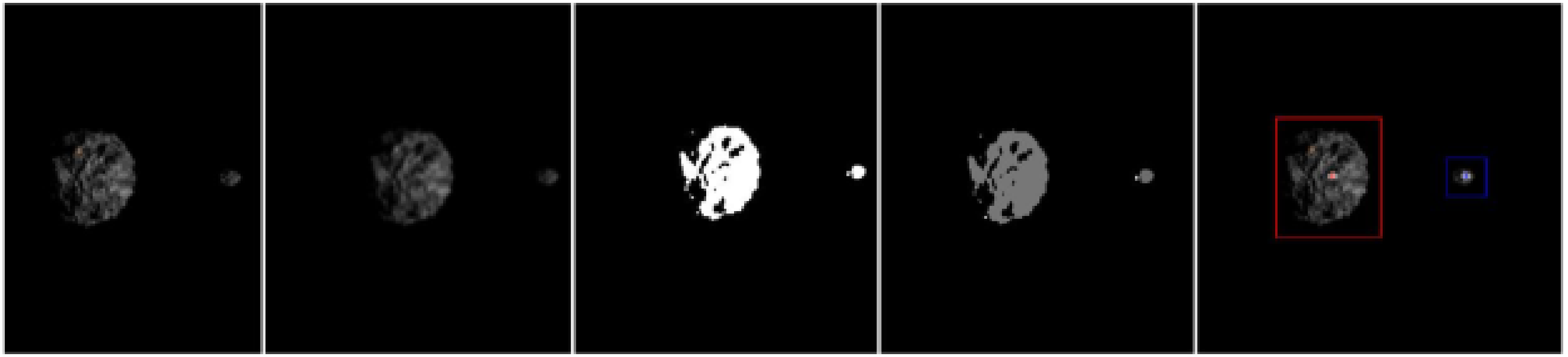


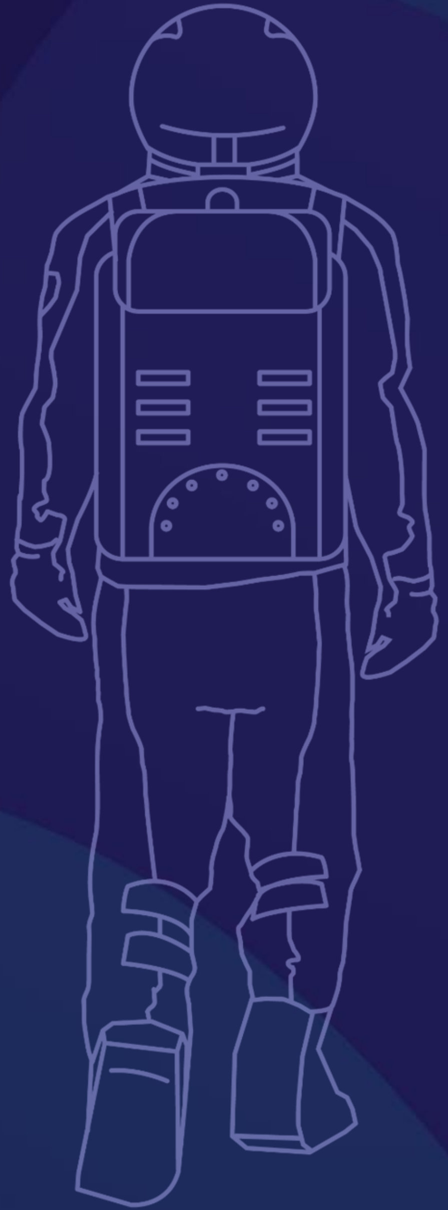
Module	SW [ms]	HW [ms]
Binning	3446	653.93
LPF	4473	653.33
HL	505	653.36
Binarization	663	653.31
MTI	2186	653.56
Total	11273	653.93

Testing and Application

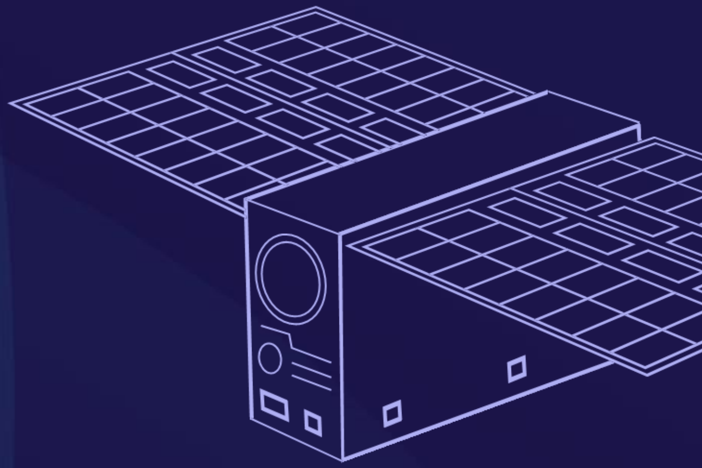
Application in Space

- LICIACube will perform high-speed Didymoon fly-by
- Autonomous navigation is too fast to operate control after SW
- HW solution process data on-the-fly, returning partial results to SW
- LICIACube autonomous navigation runs with FPGA-accelerated IS



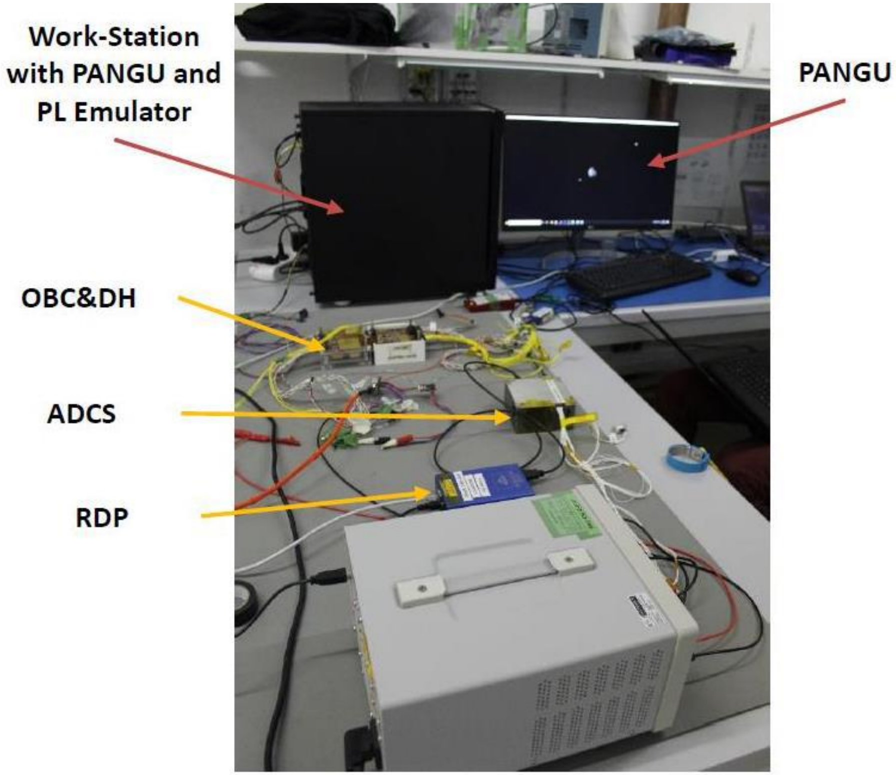


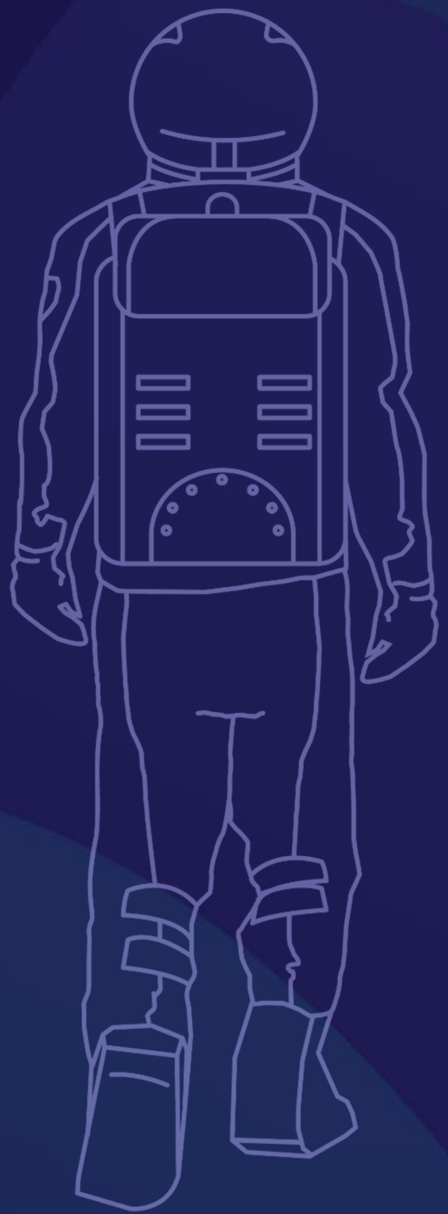
Conclusions



Conclusions

- Datapath integrates binning, filter, histogram, binarization, MTI
- Flexible, configurable and integrable with standard data interfaces
- High-performance (50Mpixels/s) and low-latency
- Reduced resource occupation (5% LUT, 3% FF, 15% BRAM) on RTG4 FPGA
- About 20x faster than SW solution on 2048x2048 image
- Enhances autonomous tracking operations in Space applications





argotec

Thank you!

www.argotecgroup.com

eugenio.scarpa@argotecgroup.com