HYDRA (AO7056)

Final presentation day

June 2, 2015

Politecnico di Torino, Enrico Magli

Techno System Developments, Giorgio Lopez

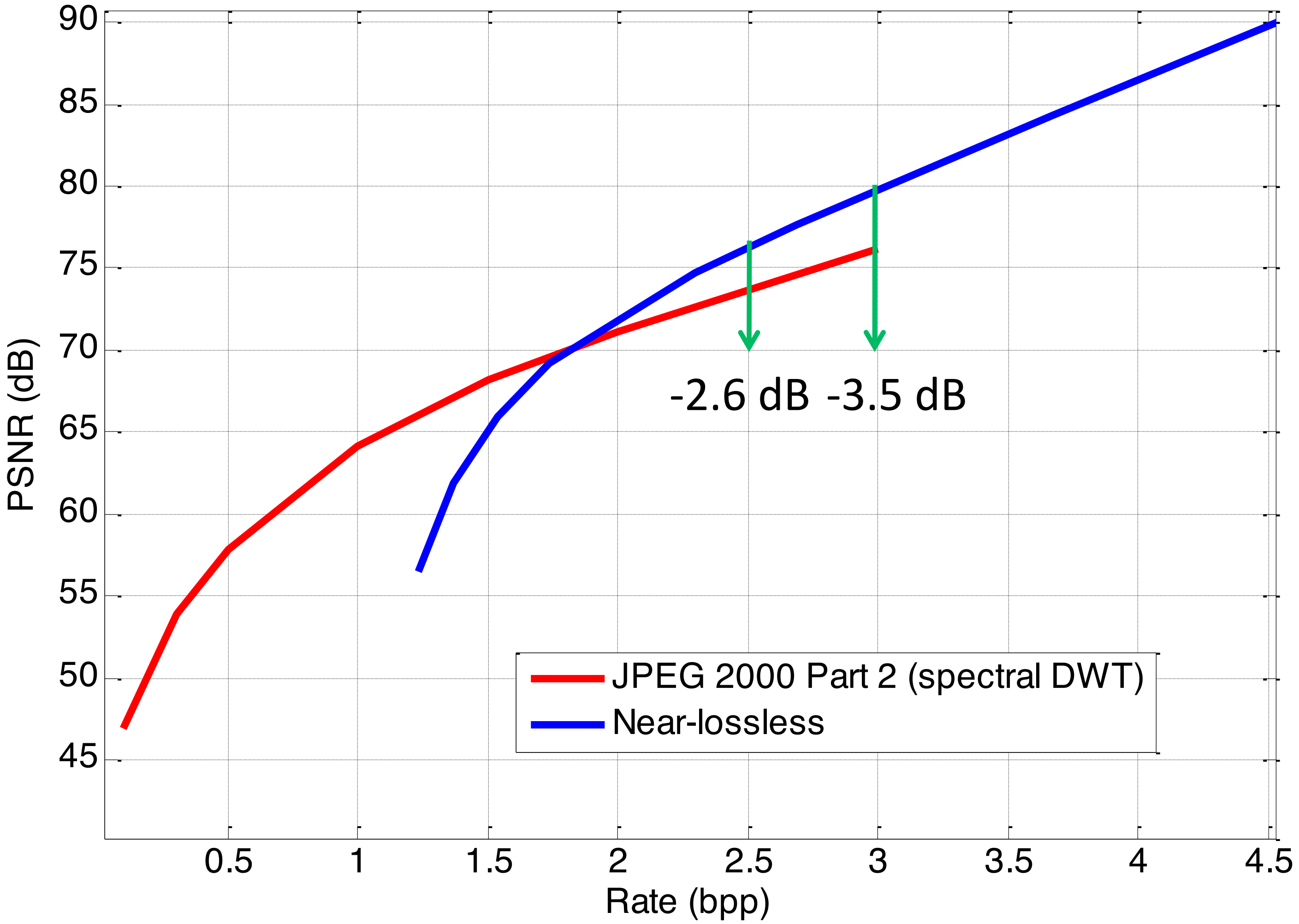
Outline

- Introduction
- Motivation – why predictive lossy compression?
- Project achievements
- Algorithm description
- Performance analysis
- Hardware architecture
- Hardware implementation

Available CCSDS standards

- CCSDS 121 “Lossless data compression”
  - Lossless, prediction-based
- CCSDS 122 “Image data compression”
  - Lossless+lossy 2D image compression, transform-based
- CCSDS 123 “Multi- and hyperspectral image compression”
  - Lossless 3D compression, prediction-based
- CCSDS 122.1 “Spectral processing transform”, extension of CCSDS 122 to 3D
  - Lossless+lossy 2D image compression, transform-based (includes POT)

Example: transform vs. prediction



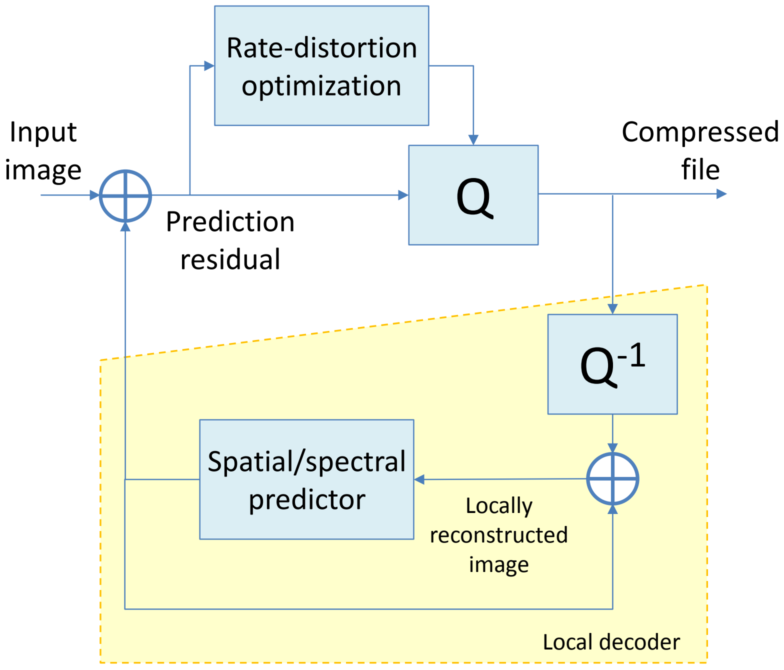
Advantages of predictive lossy compression

- Expected **better performance at high bit-rates**
- High hardware throughput (fewer calculations)
- Better error containment
  - predictor can be reset spatially/spectrally without incurring a large performance penalty
- Better quality control
  - can control error for each individual pixel
- No dynamic range expansion

… but more difficult to obtain accurate rate control

About quality control…

- Quality control in predictive lossy compression:



Local decoder inside the encoder:

- quantization error on the prediction residual is exactly the same error on the decoded pixel
- decoded pixel is available locally → define "quality policies"

PROJECT OUTCOMES AND ACHIEVEMENTS

Main project outcomes

- An algorithm extending CCSDS-123, upgraded with:
  - Quantization feedback loop
  - New entropy coding stage (range coder), required for low bit-rates
  - R/D optimization and rate control
- Main features:
  - lossless, near-lossless and lossy in one single package
  - rate ***and*** quality control
- Hardware implementation at 20 Msample/s, 16 bpp
  - Range encoder
  - Rate control

Project achievements

- The first rate control algorithm for predictive coding of multi- and hyperspectral images
- Simplified rate control implemented in hardware
- Hardware implementation also includes a significant subset of CCSDS-123
- May be a candidate for future standardization
- Range encoder: first existing hardware arithmetic coder validated for space, including
  - Optimization of statistical model for memory/performance
  - Development of ad-hoc module for division between two integer numbers

Project achievements (cont'd)

- High impact on CCSDS:
  - **CNES has changed their policy** in favor of quality control
  - A **new work item** has been requested in MHDC WG: Concept Paper for CCSDS-123.1-B "Low-Complexity Near-Lossless Multispectral & Hyperspectral Image Compression".
- Already been **adopted by several missions**:
  - implemented in hardware for METIS coronagraph
  - selected for inclusion in PRISMA (Italian Space Agency)
  - included in the baseline of the EXOMARS Rover Micromega

Project achievements (cont'd)

3 journal papers

- Diego Valsesia, Enrico Magli, "A Novel Rate Control Algorithm for Onboard Predictive Coding of Multispectral and Hyperspectral Images," IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, 2014.
- Ian Blanes, Enrico Magli, Joan Serra-Sagristà, A Tutorial on Image Compression for Optical Space Imaging Systems," IEEE GEOSCIENCE AND REMOTE SENSING MAGAZINE, 2014.
- M. Ricci, E. Magli, "Predictor analysis for onboard lossy predictive compression of multispectral and hyperspectral images," JOURNAL OF APPLIED REMOTE SENSING, 2013.

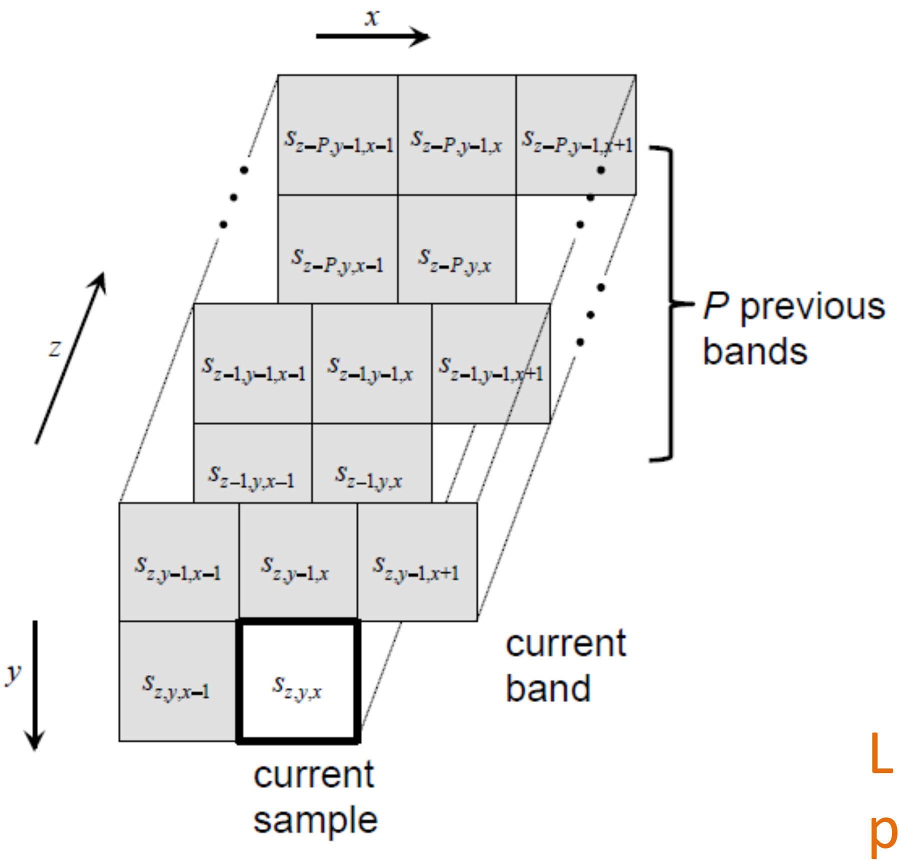
5 conference papers: IAC 2014, OBPDC 2014 (2x), ICIP 2014, OBPDC 2012

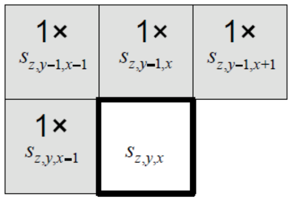
CCSDS-123

- Lossless algorithm
  - LMS adaptive predictor
  - Coding stage: two options
    - CCSDS-121 (lossless data compression), Golomb-Rice codes (block-based)
    - Golomb codes (pixel-based)

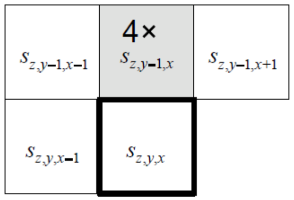
LMS predictor

- Prediction neighborhood (2 modes):





neighbor-oriented



column-oriented

LMS algorithm to update predictor coefficients

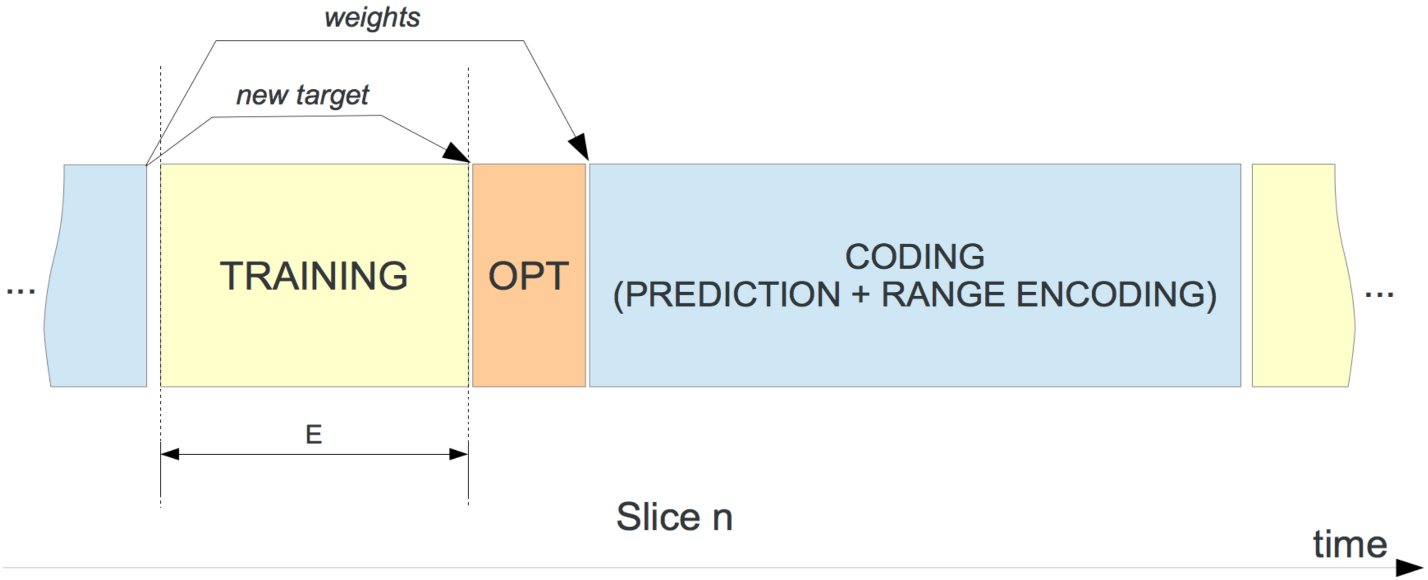
Rate control

Mode A:

- The image is partitioned into blocks of size 16 x 16
- The algorithm works one slice at a time (*slice*=row of blocks, with all the spectral channels)
- Each block is assigned a quantization step Q=$2\Delta+1$ to quantize the prediction residuals
- The Q's are written in the header of the compressed file using differential Exp-Golomb coding to keep the overhead low

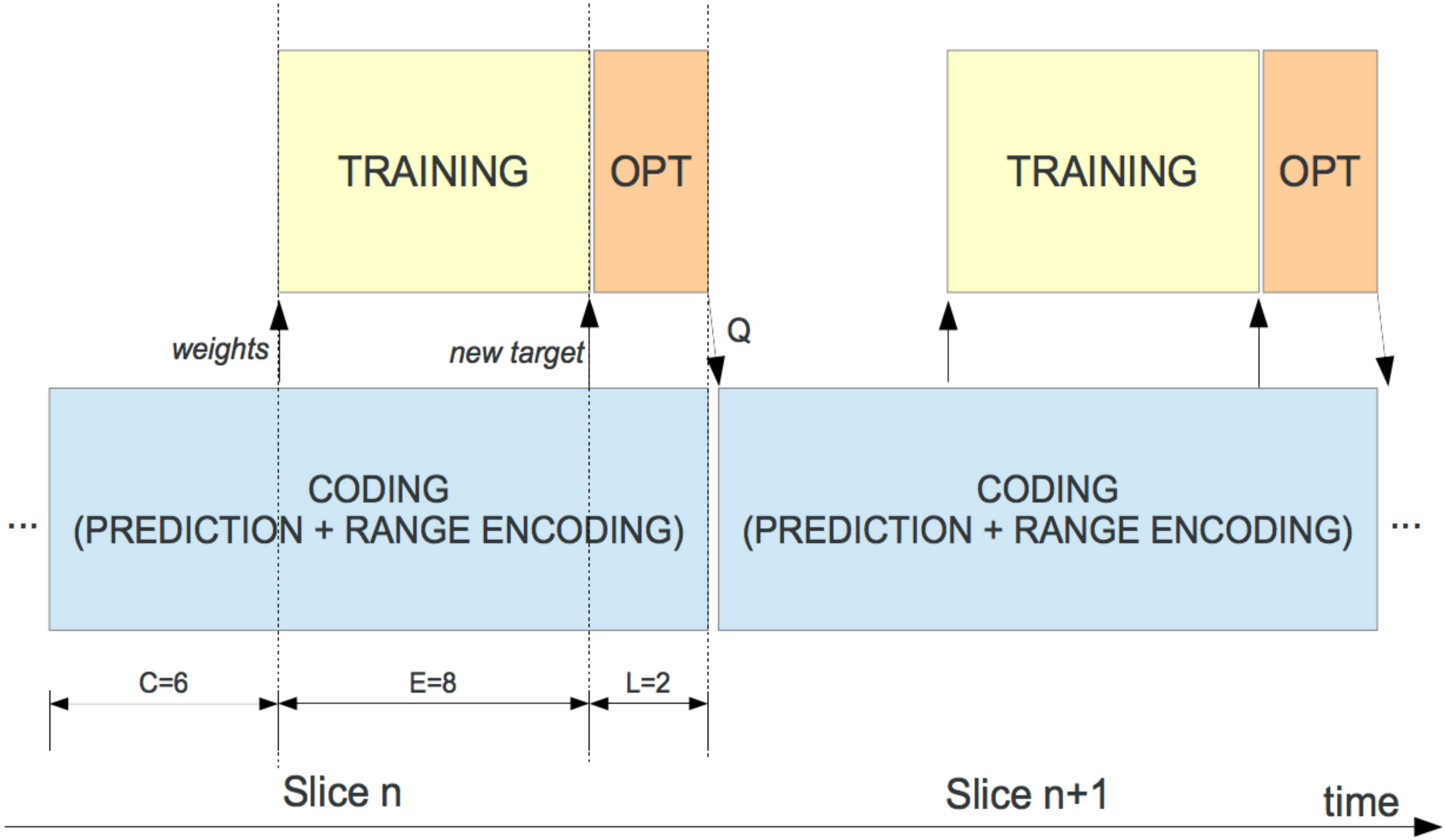
Rate control

- **Training stage**: initialize a good R/D model of the current slice
- **Optimization**: calculate set of Q's for each block, yielding the target bit-rate



Parallel version

- Pipelines rate control and coding



Rate Control: MODE B

- Sometimes MODE A does not predict the rate accurately enough
- **Mode B**: Use a slice-by-slice feedback reading how many bytes were written for the previous slice
- Update the target rate for the next slice based on this reading
    - This key step employs a tracking filter that "learns" the input-output relationship between target rate and actual rate

Simplified rate control algorithm

- The first slice (just one row) is compressed with quality parameter equal to zero
- At the end of the first slice, the actual bitrate is compared with the target
- If the actual bitrate is equal to the target the quantization parameter is unchanged; if it is above 1.25*target it is increased. If it is below 0.75*target it is decreased.

Range coding

- A simplified version of arithmetic coding
  - uniformly good performance at all rates → improved performance
  - leads to more accurate rate control
- Requires a statistical model of the prediction residuals (up to 2^{16} symbols → memory issues)
- Employs an inherently sequential coding machinery → hard to obtain high throughput

Statistical modeling

- Multiple statistical models to handle very large alphabet:
  - **Rcm_sgn**: zero/nonzero residual sample
  - **Rcm1**: *PRED_THRESHOLD* symbols, corresponding to mapped residuals lower than *PRED_THRESHOLD*
  - **Rcm2**: 256 symbols corresponding to the least significant byte of a mapped residual greater or equal than PRED_THRESHOLD
  - **Rcm3**: 256 symbols corresponding to the most significant byte of a mapped residual greater or equal than PRED_THRESHOLD
- Statistical model are reset for each new spectral slice

Coding

- 4 range encoders work in parallel
- Each of them has their own write buffer
- Once a buffer is full is is flushed to output
- Signaling is used to identify streams of different range coders

RESULTS

Results

- The algorithm has been run on the complete CCSDS image set
  - Predictor parameters taken from CCSDS-123 evaluation
  - No image- or sensor- specific optimization
- Quality metrics: SNR, MAD, ASA, MSA, POC
- Three versions of FULL, MODE B:
  - optimal (per-band statistical model for range code)
  - serial (per-slice model)
  - parallel

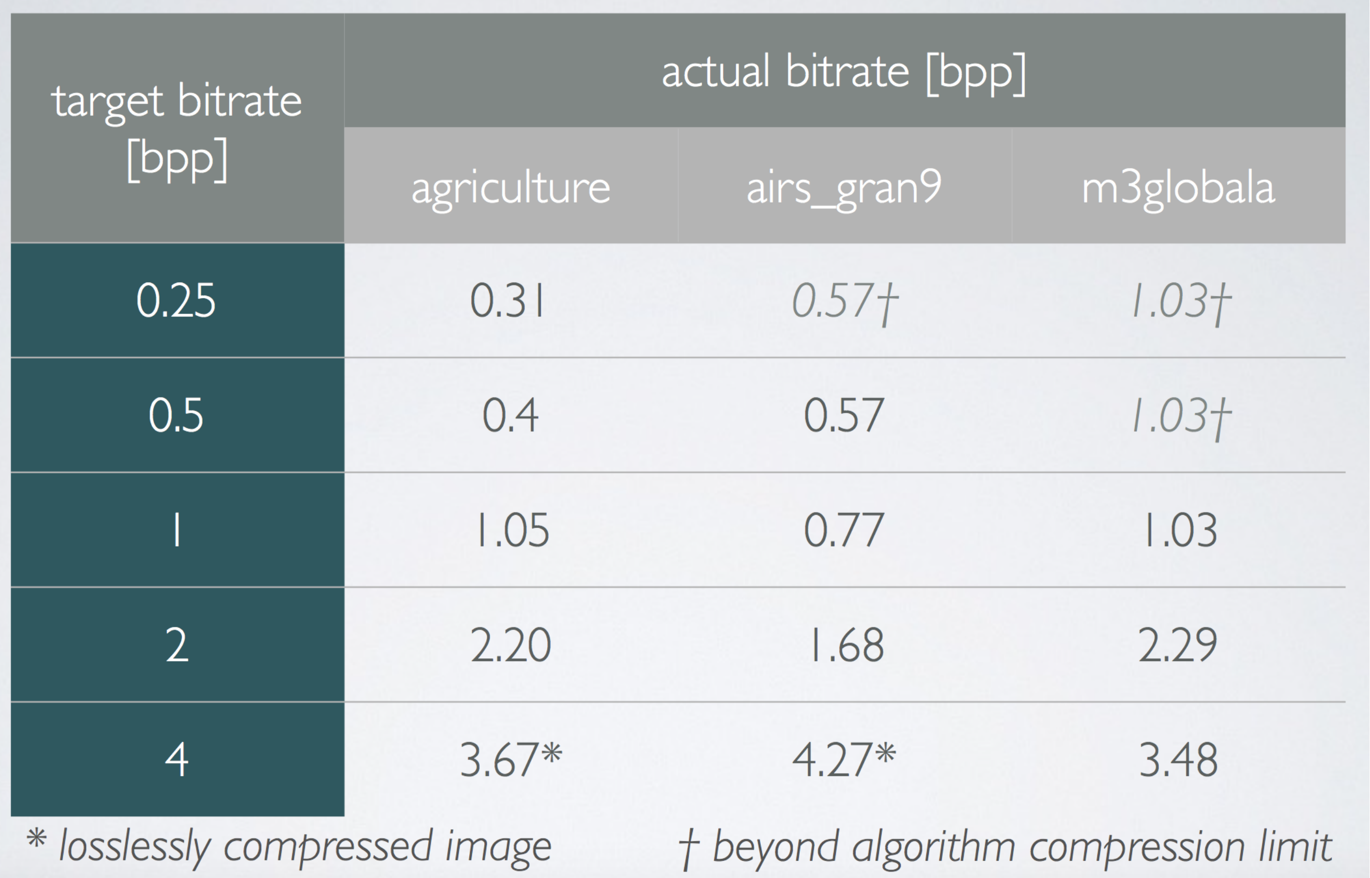
Summary of accuracy results

Target rate	ESA FULL B		ESA serial FULL B		ESA parallel FULL B	
	Mean	*St. dev.*	*Mean*	*St. dev.*	*Mean*	*St. dev.*
0.25	0.32	0.17	0.35	0.24	0.36	0.21
0.5	0.54	0.18	0.55	0.20	0.57	0.22
1.0	1.00	0.01	1.01	0.01	1.00	0.15
2.0	2.00	0.02	2.01	0.01	2.03	0.02
4.0	3.87	0.28	3.91	0.29	3.94	0.27

Note: lower mean/higher std at 4 bpp correspond to cases of lossless compression at rate below target

Simplified rate control algorithm

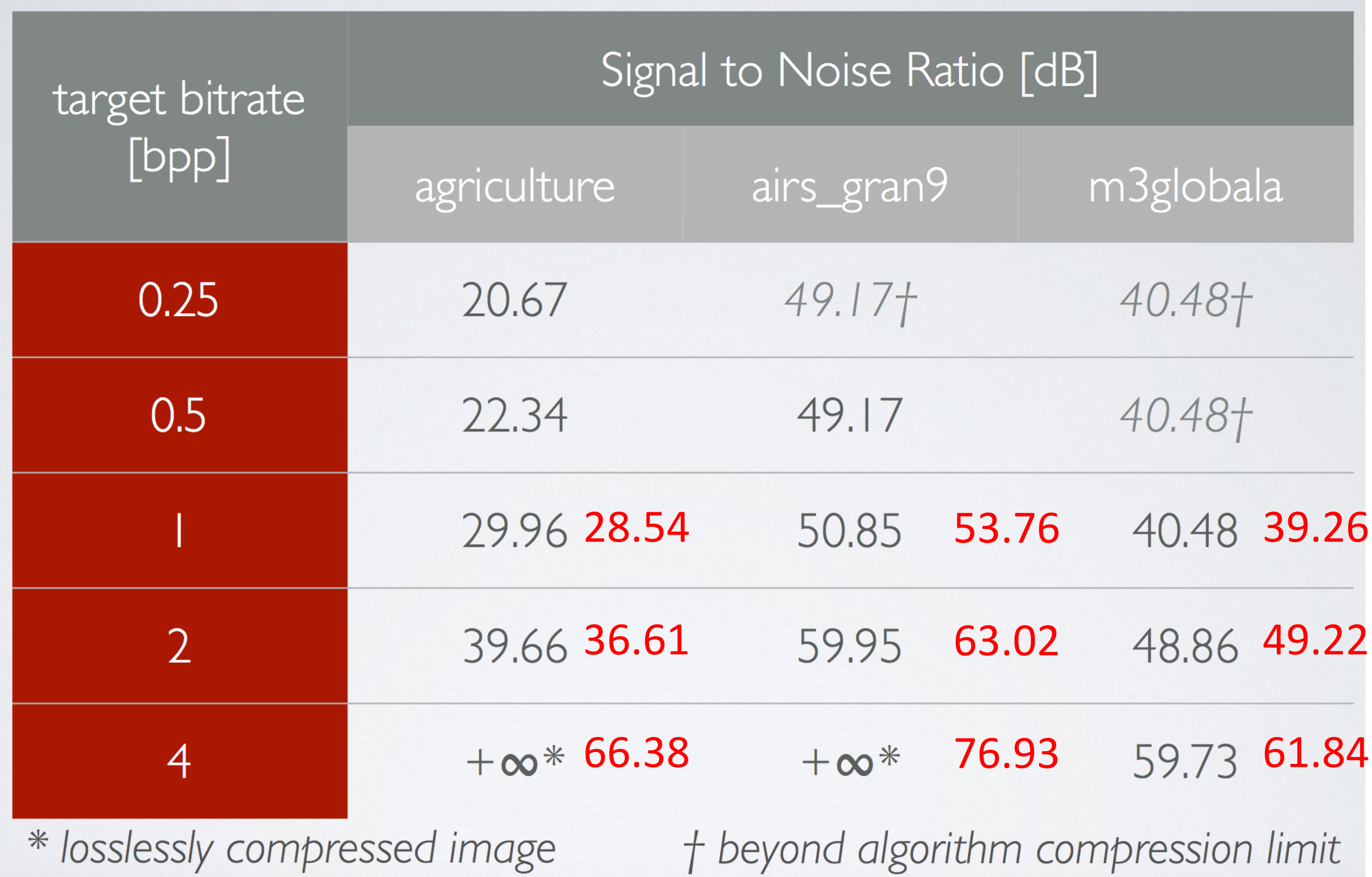
- Target accuracy ±25%



target bitrate [bpp]	actual bitrate [bpp]		
	agriculture	airs_gran9	m3globala
0.25	0.31	*0.57†*	*1.03†*
0.5	0.4	0.57	*1.03†*
1	1.05	0.77	1.03
2	2.20	1.68	2.29
4	3.67*	4.27*	3.48

* *losslessly compressed image* † *beyond algorithm compression limit*

Simplified rate control algorithm



target bitrate [bpp]	Signal to Noise Ratio [dB]		
	agriculture	airs_gran9	m3globala
0.25	20.67	*49.17†*	*40.48†*
0.5	22.34	49.17	*40.48†*
1	29.96 28.54	50.85 53.76	40.48 39.26
2	39.66 36.61	59.95 63.02	48.86 49.22
4	+∞* 66.38	+∞* 76.93	59.73 61.84

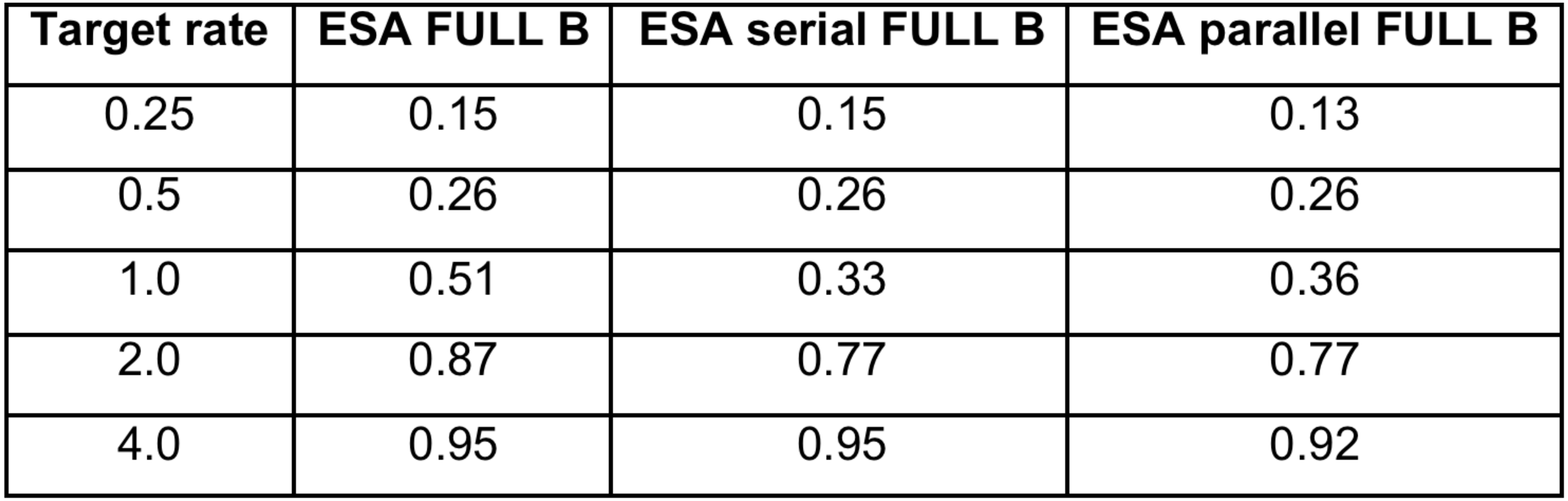
* *losslessly compressed image* † *beyond algorithm compression limit*

Comparison with transform coding

- POT + CCSDS-122 (CCSDS spring 2012 meeting).
- Rate control using buffer of 8 spectral lines
  - But proposed algorithm could use as few as 2 lines

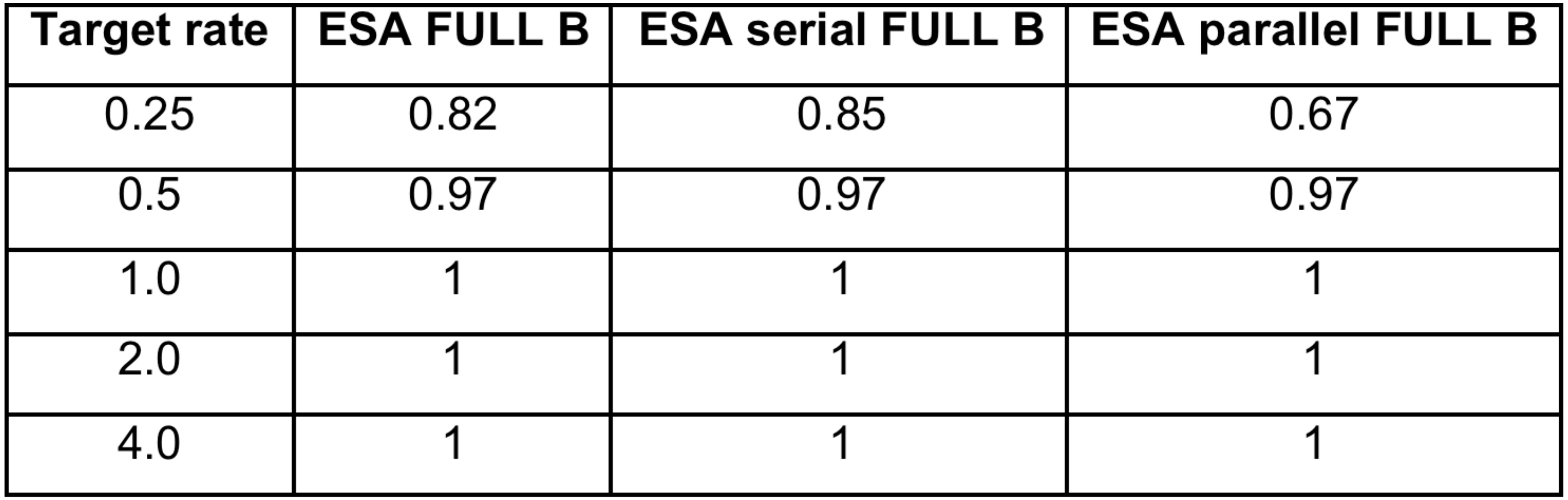
- We show % of times that proposed algorithm outperforms POT+CCSDS-122

Comparison - SNR



Target rate	ESA FULL B	ESA serial FULL B	ESA parallel FULL B
0.25	0.15	0.15	0.13
0.5	0.26	0.26	0.26
1.0	0.51	0.33	0.36
2.0	0.87	0.77	0.77
4.0	0.95	0.95	0.92

Comparison - MAD



Target rate	ESA FULL B	ESA serial FULL B	ESA parallel FULL B
0.25	0.82	0.85	0.67
0.5	0.97	0.97	0.97
1.0	1	1	1
2.0	1	1	1
4.0	1	1	1

HARDWARE IMPLEMENTATION

Hydra Soft IP Core Architecture

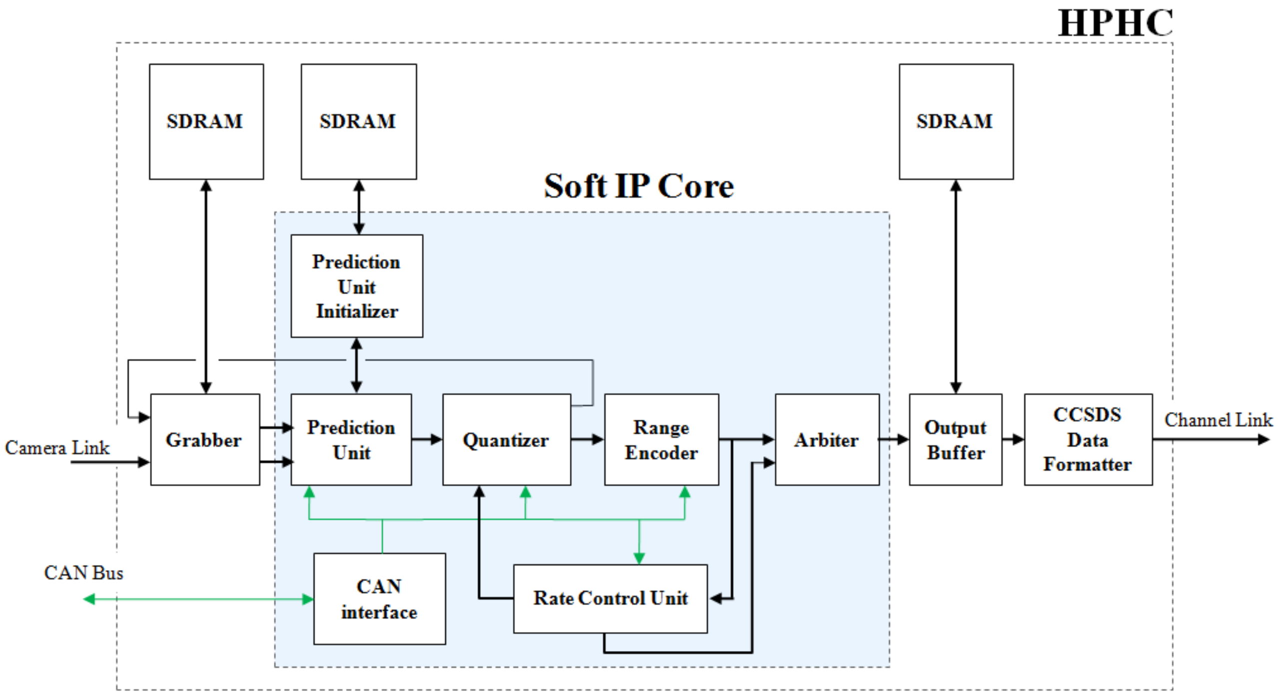


Image throughput:
20 MSamples/s

Highly reconfigurable:

- Image size *($100<Nx<4097, 1<Ny<4097, 1<Nz<4097$)*
- Prediction parameters *($t_{inc}, v_{min}, v_{max}, P$, etc.)*
- Lossless or Lossy mode
- Bitrate configuration *(lossy mode ,two operating modes)*

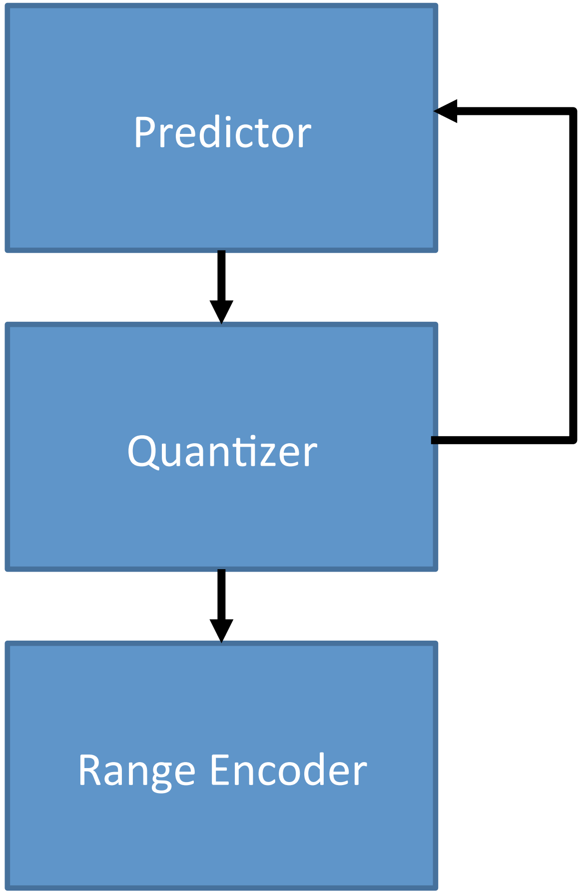
Hydra Soft IP Core: Design Flow Challenges

The design and development phases of the IP core have emphasized several criticalities:

- Algorithmic intrinsic data-dependencies (strict feedback paths in weights calculation and pixel quantization)
- Need for high performance, large FPGA devices (Xilinx 5QV FX130T)
- Need for an heavy optimization effort to reduce hardware footprint and to increase timing efficiency and clock frequency.

Lossy working modes

- Lossy mode is based on the use of a quantizer to produce the mapped residual values.
- The quantization step determines the amount of the information loss.
- Quantization step can be either statically set or dynamically changed to meet a target bitrate (rate control).



Predictor

Quantizer

Range Encoder

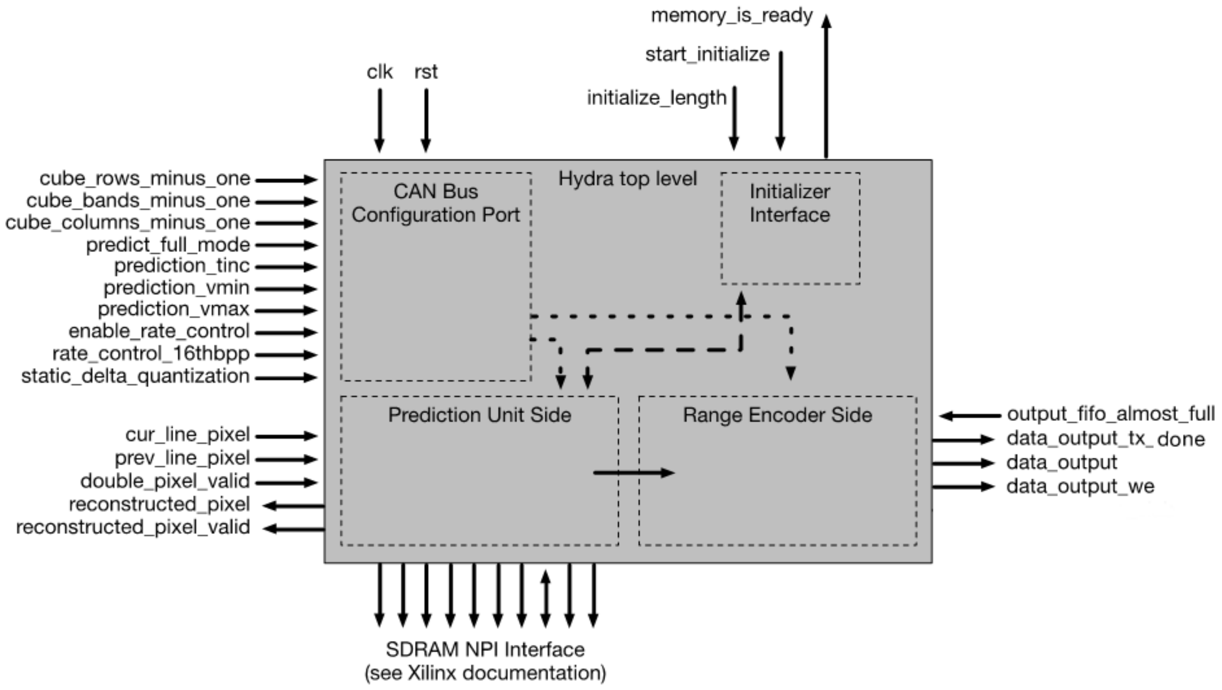
Simplified RC algorithm flow

1. A target bitrate (expressed in bits per pixel with a resolution of a 1/16th of a bit) is selected at run-time during configuration.
2. The first slice of the hyperspectral cube is compressed at a fixed delta quantization initial value (DQ-----$_0$=0).
3. At the end of the first slice, the actual bitrate (determined as the total amount of data utilized to compress the pixels up to the current point) is compared with the product of the target bitrate and the slice size.
4. If the actual bitrate is equal to the target bitrate (with a tolerance of ¼ of the target bitrate) the delta quantization is unchanged; if the actual bitrate is above 1.25*(target bitrate) the delta quantization is increased. In the last case of the actual bitrate below 0.75*(target bitrate), the delta quantization is decreased.
5. At the end of each slice steps 3 and 4 are iterated and the delta quantization value is again adapted to the actual bitrate. The delta quantization is constrained in the range [0, DQ_MAX]. In our current architecture, DQ_MAX=16.

Hydra IP core: interfaces and main blocks

3 main interfaces:

- Configuration ports (CAN Bus)
- Imager data stream input
- Compressed data stream output
- SDRAM NPI for external weights memory (68KB)



Hydra IP core resource usage

Resource	Usage	Availability
Slices	8496	20480
Slice Registers	4296	81920
LUTs	19957	81920
LUTRAM	163	25280
BRAM	158	298
SDRAM	68 kB	-
DSP48E	71	320
PLLs	1	6

Resource footprint of the Hydra IP core on a Xilinx Virtex 5QV FX130T FPGA

Space-ready flight hardware: HPHC

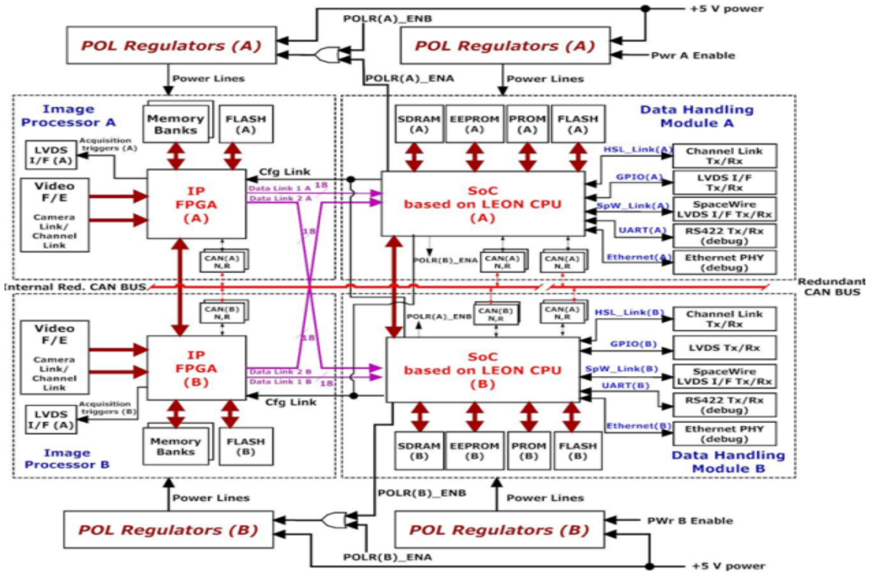
IP core has been tested on the HPHC (High performance Processing unit for Hyperspectral data Compression) EM, developed by TSD.

- High performance, compact, low mass platform
- FPGA Space-grade version available (Xilinx Virtex 5 QV FX130T)
- Space oriented interconnections and protocols (channel link, camera link, CCSDS space packet protocol etc.)

HPHC image processing architecture - 1/2

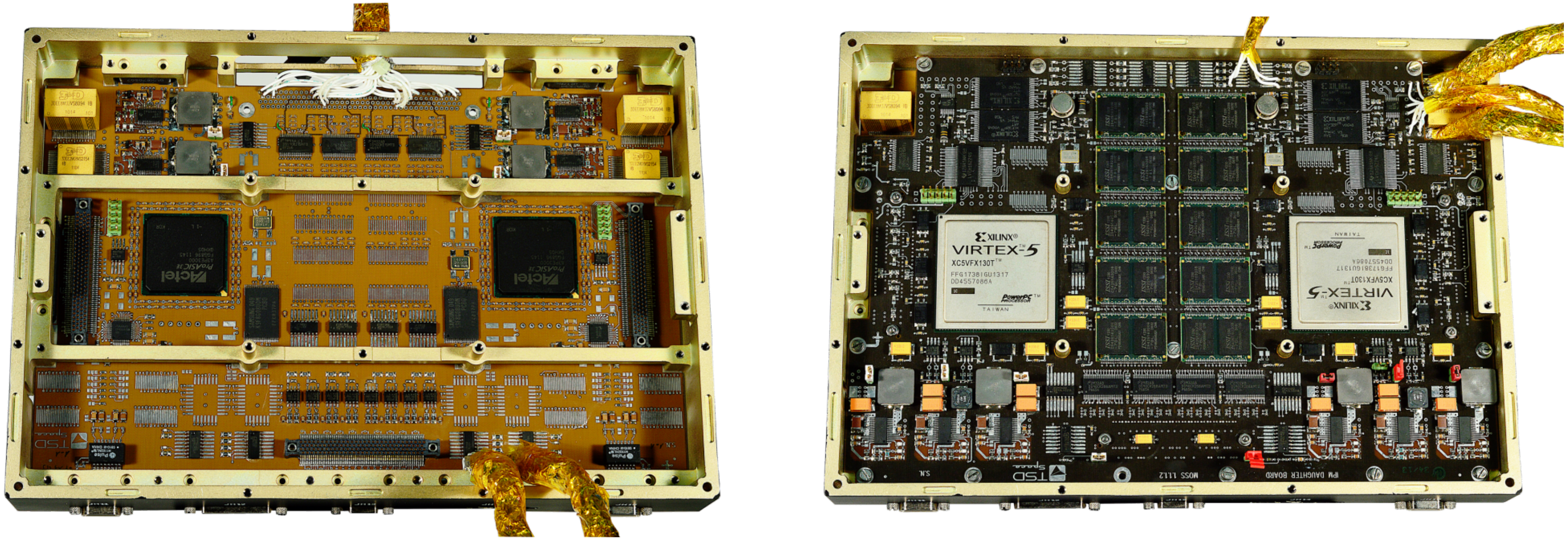
The HPHC is based on two main modules:

- Power Conditioning & Distribution Module (PCDM)
- Image Processing Module (IPM)



The IPM module is composed of two symmetric units which can be used either in cold redundancy mode (high reliability) or in Master-Slave mode (fast performance)

HPHC image processing architecture - 2/2



- Each IPM section is based on a Xilinx Virtex-5 XQ5VFX130T, the industry's first high performance rad-hard reconfigurable FPGA.

- Each FPGA is provided with 5Gbit SDRAM and two image data inputs (1.575 Gbits/s each)

Validation procedure: conducted tests

The validation of the Hydra IP core has been conducted through the following phases:

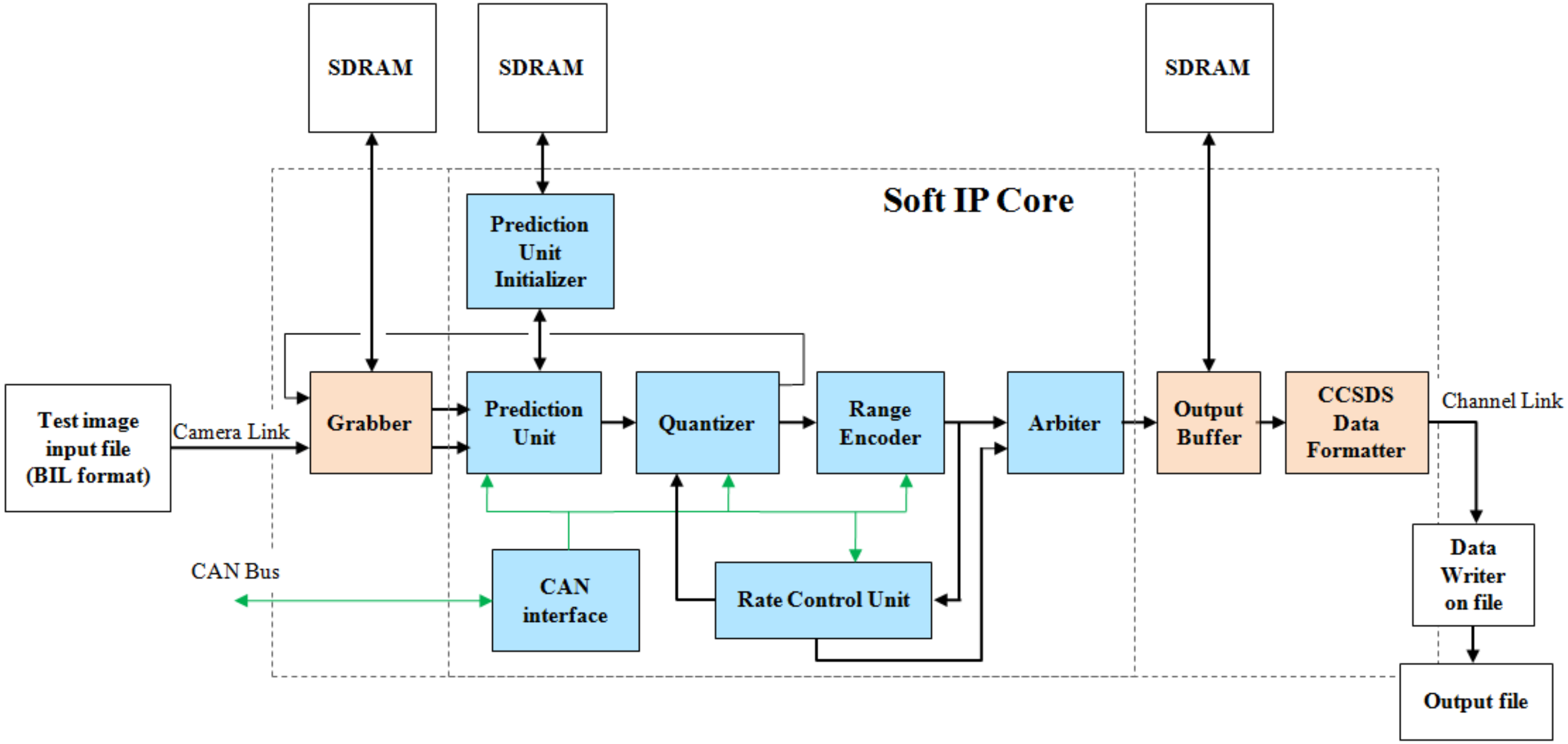
- 1: Algorithmic validation (for the simplified RC only)
- 2: VHDL Simulations:
  - 2.1: Prediction Unit
  - 2.2: Prediction Unit + Quantizer
  - 2.3: Range Encoder
  - 2.4: Prediction Unit + Quantizer + Range Encoder
  - 2.5: Full IP core elaboration pipeline
- 3: Hardware tests on the HPHC

Test vectors: configurable parameters selection

Parameter	Register Address	Register bits	Remark
Image columns	0x10	27:16	0≤ value-1 ≤4095
Image rows	0x10	11:0	0≤ value-1 ≤4095
Image bands	0x11	11:0	0≤ value-1 ≤4095
Prediction vmax	0x12	5:0	Always fixed to 0x03
Prediction vmin	0x12	12:8	Always fixed to 0x1F
Prediction tinc	0x12	18:16	Always fixed to 0x02
Prediction full mode	0x12	24	Always fixed to 1
Delta Quantization	0x13	3:0	0≤ Value ≤ 15 0 = lossless mode
Enable RC	0x13	31	1 = RC enabled
bpp rate x16	0x13	23:16	0≤ value ≤255

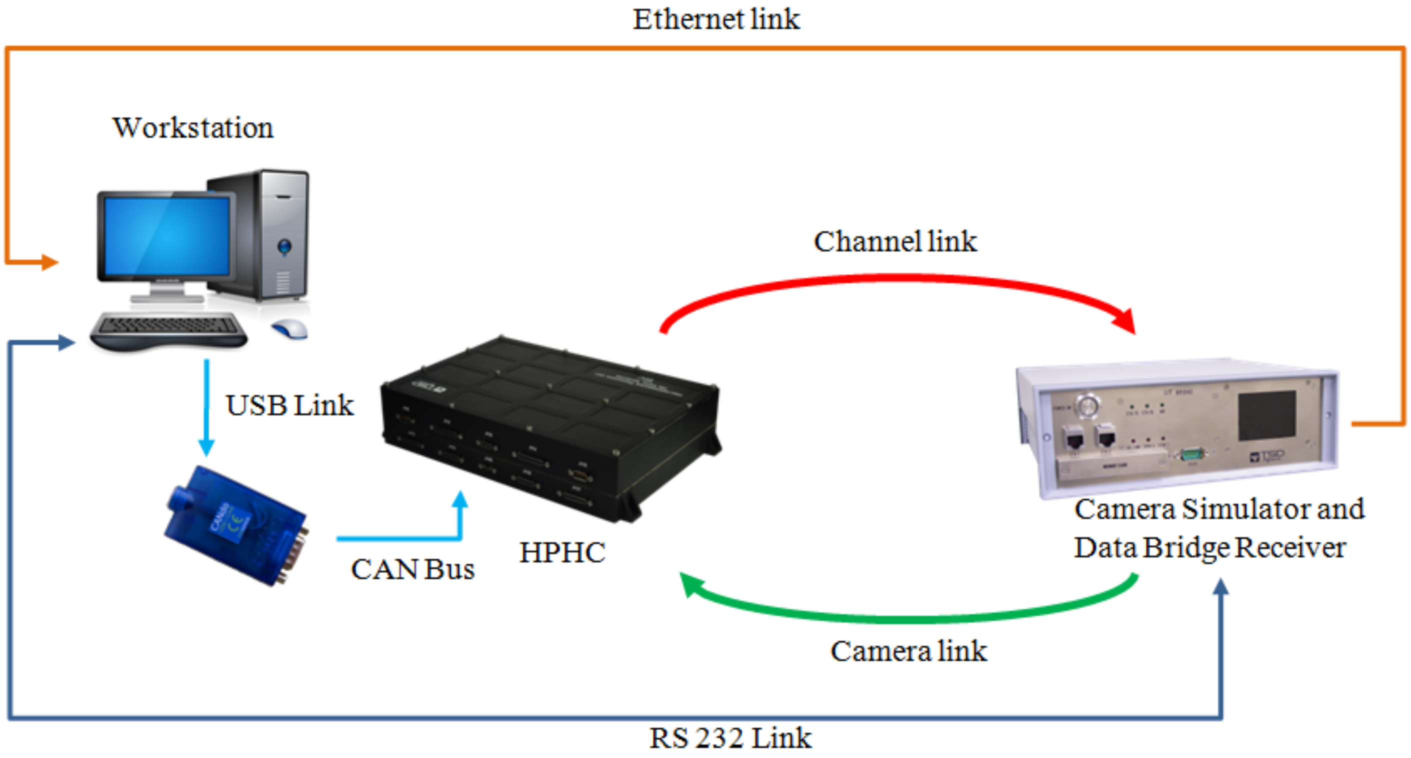
The table shows the selected parameter values for the configurable options of the Hydra IP core

Testbench for VHDL simulations



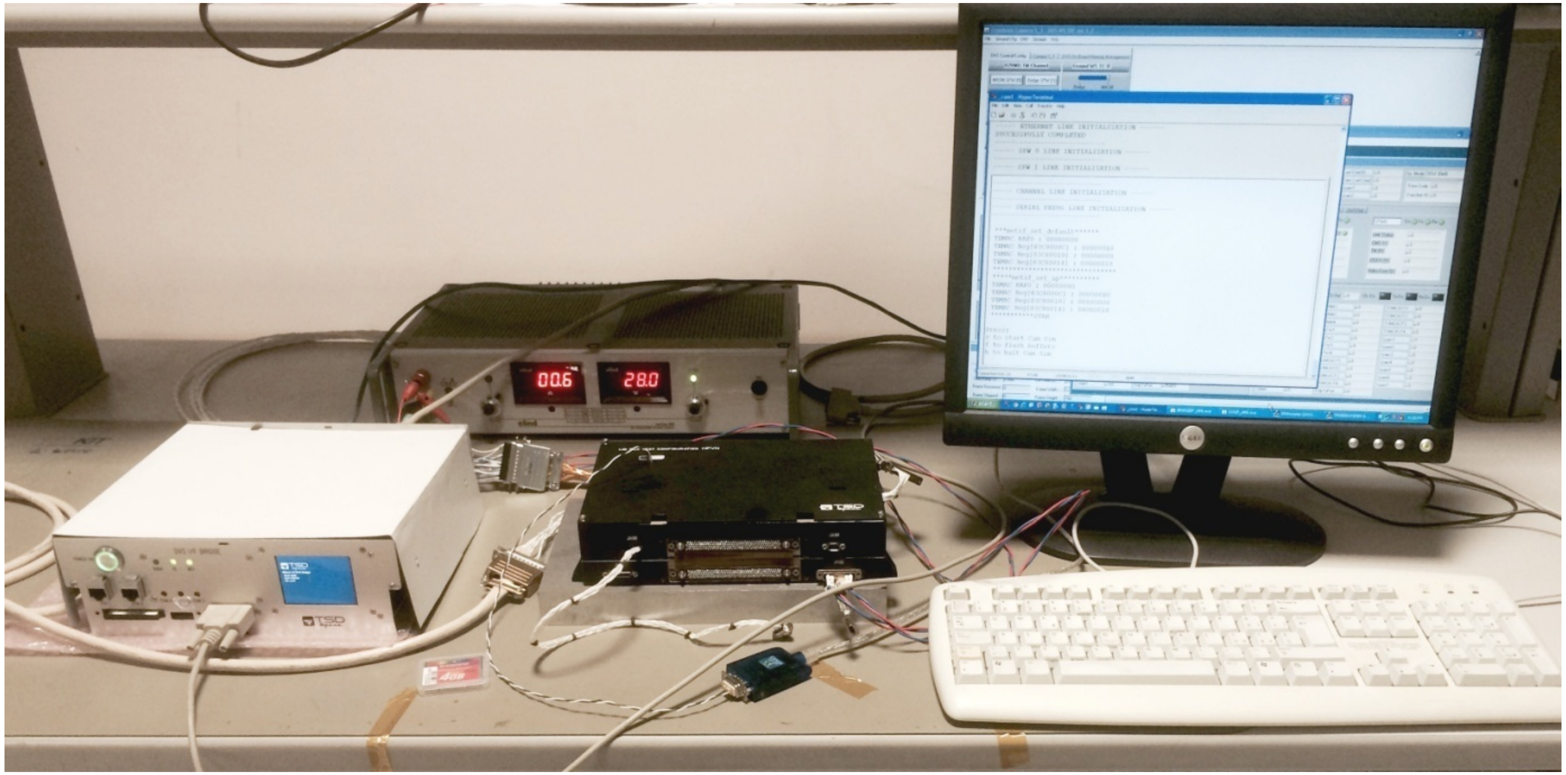
The figure shows the testbench for phase 2.5: orange colored blocks are not part of the final Hydra IP core.

Testbench for hardware validation – 1/2



Block diagram of the hardware validation testbench

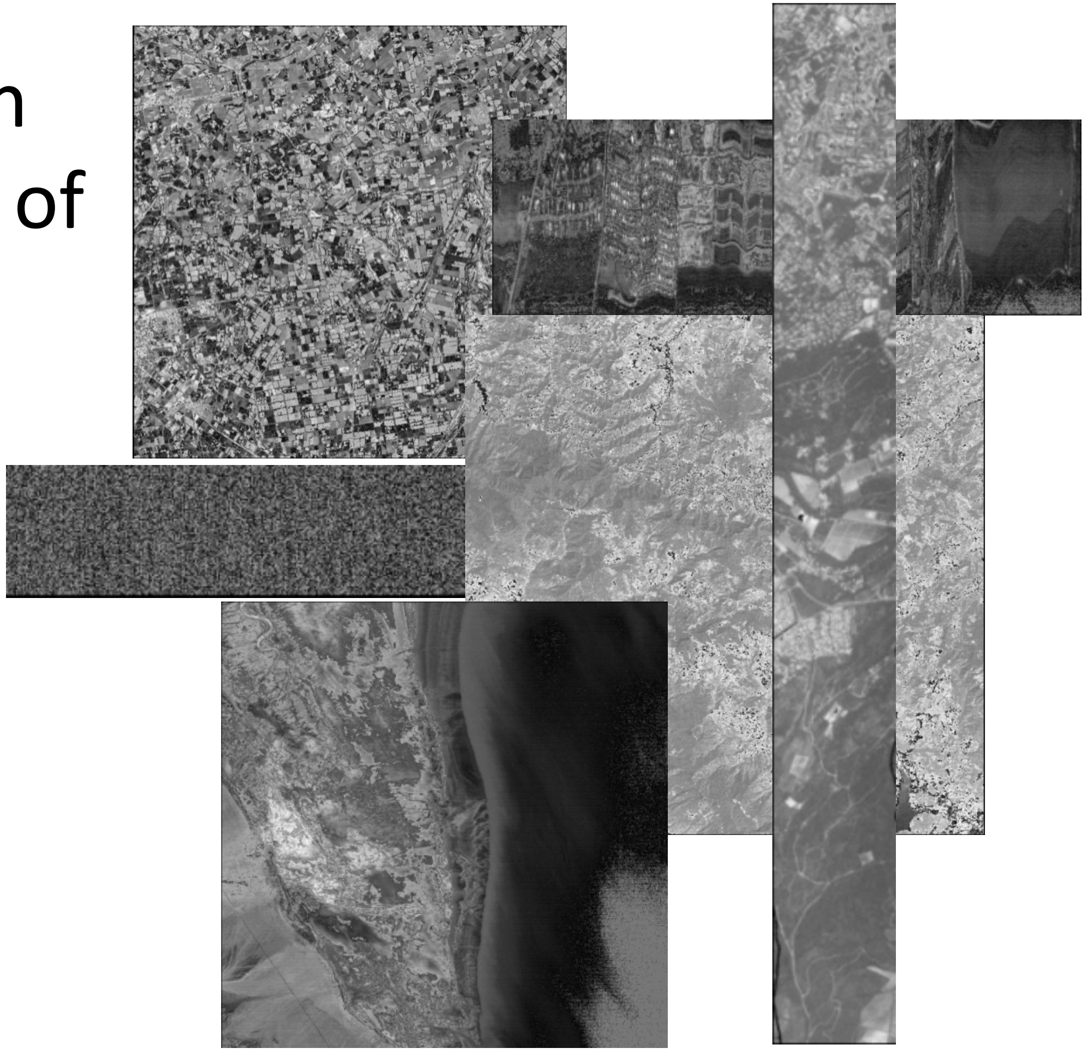
Testbench for hardware validation – 2/2



The deployed hardware validation testbench

Summary of the conducted tests: test vector images

The tests have been conducted on a set of 10 images of heterogeneous sizes, captured from different spectrometers.



Summary of the conducted tests: obtained results

- Extensive tests have been conducted showing a 100% compliance on all images and in several working conditions, both in a simulated environment and on the deployed hardware platform (HPHC).

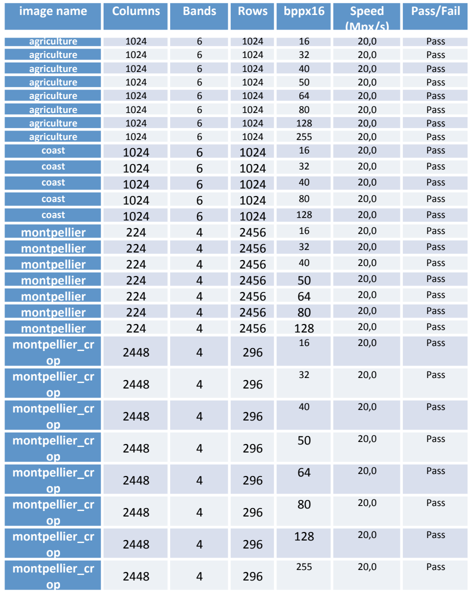


image name	Columns	Bands	Rows	bppx16	Speed (Mpx/s)	Pass/Fail
agriculture	1024	6	1024	16	20,0	Pass
agriculture	1024	6	1024	32	20,0	Pass
agriculture	1024	6	1024	40	20,0	Pass
agriculture	1024	6	1024	50	20,0	Pass
agriculture	1024	6	1024	64	20,0	Pass
agriculture	1024	6	1024	80	20,0	Pass
agriculture	1024	6	1024	128	20,0	Pass
agriculture	1024	6	1024	255	20,0	Pass
coast	1024	6	1024	16	20,0	Pass
coast	1024	6	1024	32	20,0	Pass
coast	1024	6	1024	40	20,0	Pass
coast	1024	6	1024	80	20,0	Pass
coast	1024	6	1024	128	20,0	Pass
montpellier	224	4	2456	16	20,0	Pass
montpellier	224	4	2456	32	20,0	Pass
montpellier	224	4	2456	40	20,0	Pass
montpellier	224	4	2456	50	20,0	Pass
montpellier	224	4	2456	64	20,0	Pass
montpellier	224	4	2456	80	20,0	Pass
montpellier	224	4	2456	128	20,0	Pass
montpellier_crop	2448	4	296	16	20,0	Pass
montpellier_crop	2448	4	296	32	20,0	Pass
montpellier_crop	2448	4	296	40	20,0	Pass
montpellier_crop	2448	4	296	50	20,0	Pass
montpellier_crop	2448	4	296	64	20,0	Pass
montpellier_crop	2448	4	296	80	20,0	Pass
montpellier_crop	2448	4	296	128	20,0	Pass
montpellier_crop	2448	4	296	255	20,0	Pass

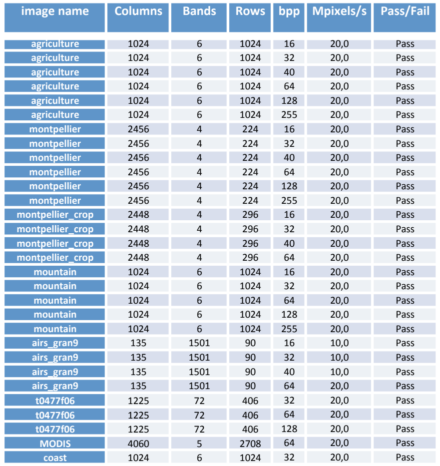


image name	Columns	Bands	Rows	bpp	Mpixels/s	Pass/Fail
agriculture	1024	6	1024	16	20,0	Pass
agriculture	1024	6	1024	32	20,0	Pass
agriculture	1024	6	1024	40	20,0	Pass
agriculture	1024	6	1024	64	20,0	Pass
agriculture	1024	6	1024	128	20,0	Pass
agriculture	1024	6	1024	255	20,0	Pass
montpellier	2456	4	224	16	20,0	Pass
montpellier	2456	4	224	32	20,0	Pass
montpellier	2456	4	224	40	20,0	Pass
montpellier	2456	4	224	64	20,0	Pass
montpellier	2456	4	224	128	20,0	Pass
montpellier	2456	4	224	255	20,0	Pass
montpellier_crop	2448	4	296	16	20,0	Pass
montpellier_crop	2448	4	296	32	20,0	Pass
montpellier_crop	2448	4	296	40	20,0	Pass
montpellier_crop	2448	4	296	64	20,0	Pass
mountain	1024	6	1024	16	20,0	Pass
mountain	1024	6	1024	32	20,0	Pass
mountain	1024	6	1024	64	20,0	Pass
mountain	1024	6	1024	128	20,0	Pass
mountain	1024	6	1024	255	20,0	Pass
airs_gran9	135	1501	90	16	10,0	Pass
airs_gran9	135	1501	90	32	10,0	Pass
airs_gran9	135	1501	90	40	10,0	Pass
airs_gran9	135	1501	90	64	20,0	Pass
t0477f06	1225	72	406	32	20,0	Pass
t0477f06	1225	72	406	64	20,0	Pass
t0477f06	1225	72	406	128	20,0	Pass
MODIS	4060	5	2708	64	20,0	Pass
coast	1024	6	1024	32	20,0	Pass

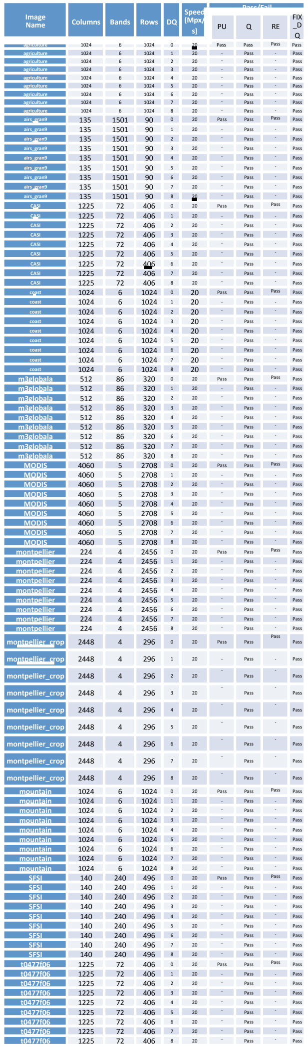


Image Name	Columns	Bands	Rows	DQ	Speed (Mpx/s)	Pass/Fail			
						PU	Q	RE	FIX_DQ
agriculture	1024	6	1024	0	20	Pass	Pass	Pass	Pass
agriculture	1024	6	1024	1	20	-	Pass	-	Pass
agriculture	1024	6	1024	2	20	-	Pass	-	Pass
agriculture	1024	6	1024	3	20	-	Pass	-	Pass
agriculture	1024	6	1024	4	20	-	Pass	-	Pass
agriculture	1024	6	1024	5	20	-	Pass	-	Pass
agriculture	1024	6	1024	6	20	-	Pass	-	Pass
agriculture	1024	6	1024	7	20	-	Pass	-	Pass
agriculture	1024	6	1024	8	20	-	Pass	-	Pass
airs_gran9	135	1501	90	0	20	Pass	Pass	Pass	Pass
airs_gran9	135	1501	90	1	20	-	Pass	-	Pass
airs_gran9	135	1501	90	2	20	-	Pass	-	Pass
airs_gran9	135	1501	90	3	20	-	Pass	-	Pass
airs_gran9	135	1501	90	4	20	-	Pass	-	Pass
airs_gran9	135	1501	90	5	20	-	Pass	-	Pass
airs_gran9	135	1501	90	6	20	-	Pass	-	Pass
airs_gran9	135	1501	90	7	20	-	Pass	-	Pass
airs_gran9	135	1501	90	8	20	-	Pass	-	Pass
CASI	1225	72	406	0	20	Pass	Pass	Pass	Pass
CASI	1225	72	406	1	20	-	Pass	-	Pass
CASI	1225	72	406	2	20	-	Pass	-	Pass
CASI	1225	72	406	3	20	-	Pass	-	Pass
CASI	1225	72	406	4	20	-	Pass	-	Pass
CASI	1225	72	406	5	20	-	Pass	-	Pass
CASI	1225	72	406	6	20	-	Pass	-	Pass
CASI	1225	72	406	7	20	-	Pass	-	Pass
CASI	1225	72	406	8	20	-	Pass	-	Pass
coast	1024	6	1024	0	20	Pass	Pass	Pass	Pass
coast	1024	6	1024	1	20	-	Pass	-	Pass
coast	1024	6	1024	2	20	-	Pass	-	Pass
coast	1024	6	1024	3	20	-	Pass	-	Pass
coast	1024	6	1024	4	20	-	Pass	-	Pass
coast	1024	6	1024	5	20	-	Pass	-	Pass
coast	1024	6	1024	6	20	-	Pass	-	Pass
coast	1024	6	1024	7	20	-	Pass	-	Pass
coast	1024	6	1024	8	20	-	Pass	-	Pass
m3globala	512	86	320	0	20	Pass	Pass	Pass	Pass
m3globala	512	86	320	1	20	-	Pass	-	Pass
m3globala	512	86	320	2	20	-	Pass	-	Pass
m3globala	512	86	320	3	20	-	Pass	-	Pass
m3globala	512	86	320	4	20	-	Pass	-	Pass
m3globala	512	86	320	5	20	-	Pass	-	Pass
m3globala	512	86	320	6	20	-	Pass	-	Pass
m3globala	512	86	320	7	20	-	Pass	-	Pass
m3globala	512	86	320	8	20	-	Pass	-	Pass
MODIS	4060	5	2708	0	20	Pass	Pass	Pass	Pass
MODIS	4060	5	2708	1	20	-	Pass	-	Pass
MODIS	4060	5	2708	2	20	-	Pass	-	Pass
MODIS	4060	5	2708	3	20	-	Pass	-	Pass
MODIS	4060	5	2708	4	20	-	Pass	-	Pass
MODIS	4060	5	2708	5	20	-	Pass	-	Pass
MODIS	4060	5	2708	6	20	-	Pass	-	Pass
MODIS	4060	5	2708	7	20	-	Pass	-	Pass
MODIS	4060	5	2708	8	20	-	Pass	-	Pass
montpellier	224	4	2456	0	20	Pass	Pass	Pass	Pass
montpellier	224	4	2456	1	20	-	Pass	-	Pass
montpellier	224	4	2456	2	20	-	Pass	-	Pass
montpellier	224	4	2456	3	20	-	Pass	-	Pass
montpellier	224	4	2456	4	20	-	Pass	-	Pass
montpellier	224	4	2456	5	20	-	Pass	-	Pass
montpellier	224	4	2456	6	20	-	Pass	-	Pass
montpellier	224	4	2456	7	20	-	Pass	-	Pass
montpellier	224	4	2456	8	20	-	Pass	-	Pass
montpellier_crop	2448	4	296	0	20	Pass	Pass	Pass	Pass
montpellier_crop	2448	4	296	1	20	-	Pass	-	Pass
montpellier_crop	2448	4	296	2	20	-	Pass	-	Pass
montpellier_crop	2448	4	296	3	20	-	Pass	-	Pass
montpellier_crop	2448	4	296	4	20	-	Pass	-	Pass
montpellier_crop	2448	4	296	5	20	-	Pass	-	Pass
montpellier_crop	2448	4	296	6	20	-	Pass	-	Pass
montpellier_crop	2448	4	296	7	20	-	Pass	-	Pass
montpellier_crop	2448	4	296	8	20	-	Pass	-	Pass
mountain	1024	6	1024	0	20	Pass	Pass	Pass	Pass
mountain	1024	6	1024	1	20	-	Pass	-	Pass
mountain	1024	6	1024	2	20	-	Pass	-	Pass
mountain	1024	6	1024	3	20	-	Pass	-	Pass
mountain	1024	6	1024	4	20	-	Pass	-	Pass
mountain	1024	6	1024	5	20	-	Pass	-	Pass
mountain	1024	6	1024	6	20	-	Pass	-	Pass
mountain	1024	6	1024	7	20	-	Pass	-	Pass
mountain	1024	6	1024	8	20	-	Pass	-	Pass
SFSI	140	240	496	0	20	Pass	Pass	Pass	Pass
SFSI	140	240	496	1	20	-	Pass	-	Pass
SFSI	140	240	496	2	20	-	Pass	-	Pass
SFSI	140	240	496	3	20	-	Pass	-	Pass
SFSI	140	240	496	4	20	-	Pass	-	Pass
SFSI	140	240	496	5	20	-	Pass	-	Pass
SFSI	140	240	496	6	20	-	Pass	-	Pass
SFSI	140	240	496	7	20	-	Pass	-	Pass
SFSI	140	240	496	8	20	-	Pass	-	Pass
t0477f06	1225	72	406	0	20	Pass	Pass	Pass	Pass
t0477f06	1225	72	406	1	20	-	Pass	-	Pass
t0477f06	1225	72	406	2	20	-	Pass	-	Pass
t0477f06	1225	72	406	3	20	-	Pass	-	Pass
t0477f06	1225	72	406	4	20	-	Pass	-	Pass
t0477f06	1225	72	406	5	20	-	Pass	-	Pass
t0477f06	1225	72	406	6	20	-	Pass	-	Pass
t0477f06	1225	72	406	7	20	-	Pass	-	Pass
t0477f06	1225	72	406	8	20	-	Pass	-	Pass

Conclusions and outlook

- Developed a state-of-the-art algorithm for lossless, near-lossless and lossy compression
  - To be proposed for **standardization in CCSDS**
- Several innovative elements (algorithms & hardware)
  - Rate control, range encoder
- Already selected for several missions
- Validated for space @20 Msamples/s
- Several improvements still possible
  - Throughput → 50 Msamples/s
  - New rate and quality control algorithms
  - Better and faster entropy coder