



UNIVERSITAT POLITÈCNICA
DE CATALUNYA

IEEC

DAPCOM
Data services

UNIVERSITAT DE BARCELONA
B

Hierarchical Pixel Averaging: A new image compression approach

Riccardo Iudica (IEEC)
Enrique García-Berro (IEEC/UPC)
Jordi Portell (IEEC/DAPCOM/UB)

Presented by Riccardo Iudica

Image Data Compression in space

- ▶ One of the most common needs
 - Many satellites generate images that must be sent to the ground requiring the minimum possible downlink rate
- ▶ Limitations:
 - On-board processing power
→ low complexity
 - Communications channel reliability
→ small independent blocks
- ▶ Image data compression:
 - Images generally contain a significant amount of redundancy
 - Lossy compression is usually preferable

FAPEC limitations

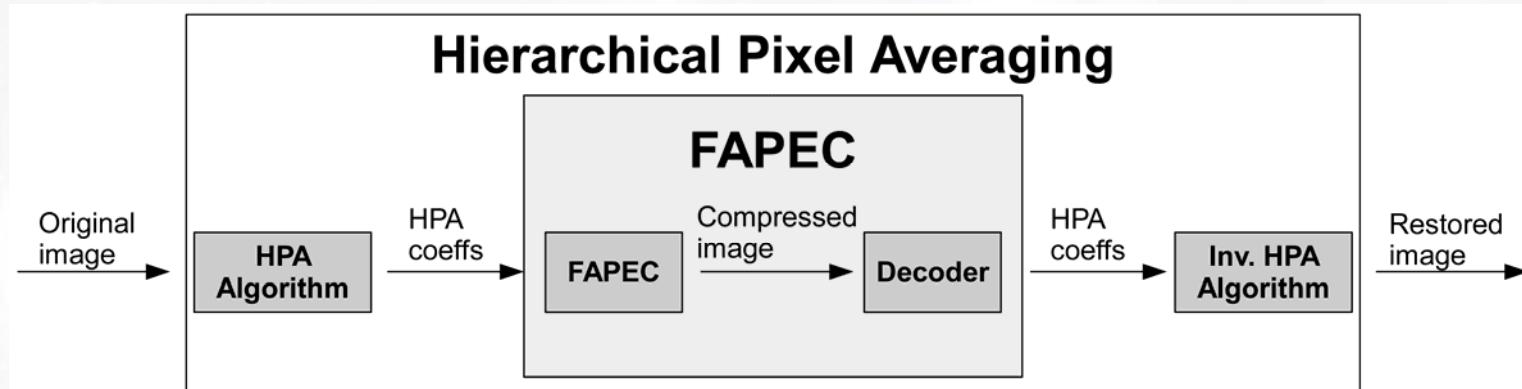
- ▶ FAPEC appears as an alternative to CCSDS 121.0
- ▶ It is based on an entropy coder that performs better than CCSDS 121.0 in presence of outliers.

FAPEC limitations:

- ▶ Needs a better pre-processing stage for images to lower their entropy level
- ▶ Could take advantage of the inter-pixel correlations present in an image
- ▶ Lossy compression is not implemented

Hierarchical Pixel Averaging

We introduce a new image processing algorithm (HPA)



Objectives:

- ▶ Use similarities between adjacent pixels
- ▶ Introduces easily controlled losses
- ▶ Improve sharpness in the lossy case
- ▶ No floating point operations

Hierarchical Pixel Averaging

The basic block is 16 x 16 pixels, divided into 4 hierarchical levels:

- ▶ Level 0: 256 pixels
- ▶ Level 1: 64 \bar{H}_1
- ▶ Level 2: 16 \bar{H}_2
- ▶ Level 3: 4 \bar{H}_3
- ▶ Level 4: 1 \bar{H}_4

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23								
32	33	34	35	36	37	38	39								
48	49	50	51	52	53	54	55								
64	65	66	67	68	69	70	71								
80	81	82	83	84	85	86	87								
96	97	98	99	100	101	102	103								
112	113	114	115	116	117	118	119								127
128															
144															
160															
176															
192															
208															
224															
240								247	248						255

for each level:

$$\overline{H_{n+1}(A)} = \frac{\overline{H_n(B)} + \overline{H_n(C)} + \overline{H_n(D)} + \overline{H_n(E)}}{4}$$

Level 0: 256 individual pixels

Level 1: 64 2x2 means (or averages)

Level 2: 16 4x4 means from 2x2 Lev1

Level 3: 4 8x8 means from 2x2 Lev2

Level 4: 1 16x16 mean from 2x2 Lev3

Hierarchical Pixel Averaging

We defined also differential coefficients for each level.

Number of differential coefficients:

- ▶ Level 0: 256 ΔH_0
- ▶ Level 1: 64 ΔH_1
- ▶ Level 2: 16 ΔH_2
- ▶ Level 3: 4 ΔH_3
- ▶ Level 4: 1 \overline{H}_4

Total number of coefficients:
341, 33% more than 256

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23								
32	33	34	35	36	37	38	39								
48	49	50	51	52	53	54	55								
64	65	66	67	68	69	70	71								
80	81	82	83	84	85	86	87								
96	97	98	99	100	101	102	103								
112	113	114	115	116	117	118	119								127
128															
144															
160															
176															
192															
208															
224															
240								247	248						255

Level 0: 256 individual pixels

Level 1: 64 2x2 means (or averages)

Level 2: 16 4x4 means from 2x2 Lev1

Level 3: 4 8x8 means from 2x2 Lev2

Level 4: 1 16x16 mean from 2x2 Lev3

Hierarchical Pixel Averaging

$H_n(A)$: sum of the pixels contained in block A of level n

$\rho_n(A)$: remainder of its division by 4

$$\overline{H_n(B)} + \overline{H_n(C)} + \overline{H_n(D)} + \overline{H_n(E)} = 4 \times \overline{H_{n+1}(A)} + \rho_{n+1}(A)$$

The fourth element can be computed from:

$$\overline{H_n(E)} = 4 \times \overline{H_{n+1}(A)} + \rho_{n+1}(A) - \overline{H_n(B)} - \overline{H_n(C)} - \overline{H_n(D)}$$

The number of coefficients is:

- ▶ Level 0: 192 ΔH_0
- ▶ Level 1: 48 ΔH_1
- ▶ Level 2: 12 ΔH_2
- ▶ Level 3: 3 ΔH_3
- ▶ Level 4: 1 $\overline{H_4}$

Total number of coefficients: 256

Hierarchical Pixel Averaging

The overhead can be calculated for each level:

- ▶ Level 1: $64 \rho_1 = 128$ bits = 16 bytes
- ▶ Level 2: $16 \rho_2 = 32$ bits = 4 bytes
- ▶ Level 3: $4 \rho_3 = 8$ bits = 1 bytes
- ▶ Level 4: $1 \rho_4 = 2$ bits

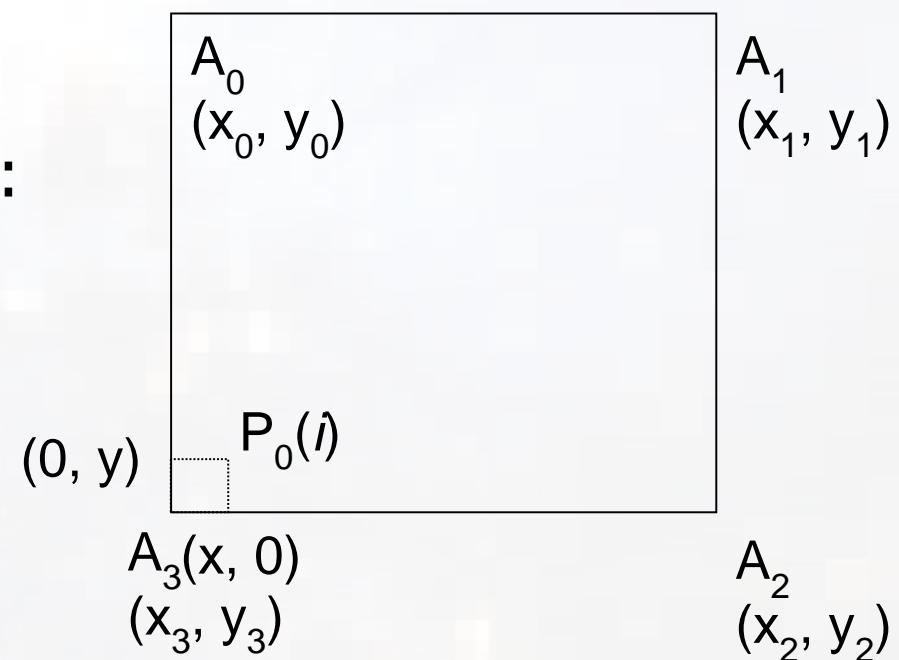
The total overhead is 21 bytes + 2 bits, i.e. an overhead of **0.66** bits per pixel

The only current limitation is imposed by the dimensions of the image, which must be multiple of 16.

Hierarchical Pixel Interpolation

- ▶ We introduce an evolution of the HPA algorithm
- ▶ The average coefficients of level 1 can be interpolated to predict the pixels value.
- ▶ A bilinear interpolation is used to predict the pixels, which stands in a square that has the level 1 average values as corner.
- ▶ The new differential coefficient is calculated as:

$$\Delta H_0(i) = P_0(i) - P_0(i)'$$

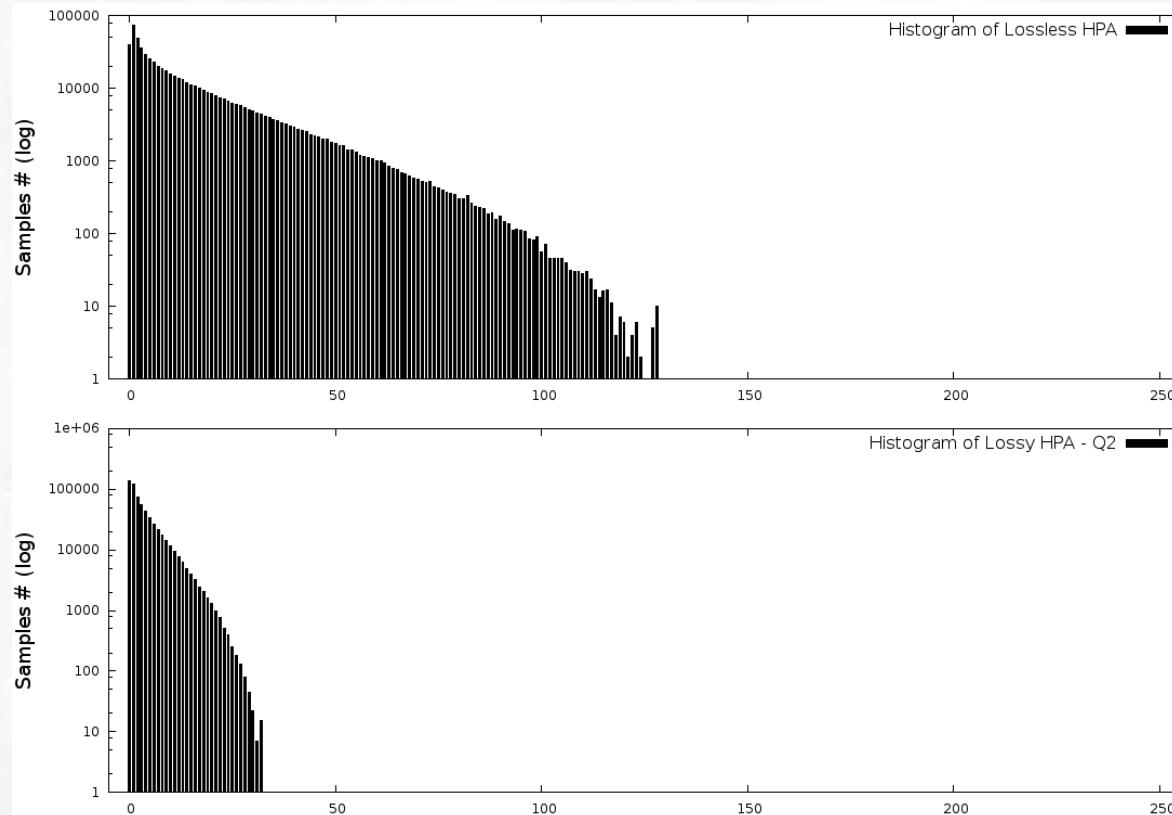


Lossy HPA+FAPEC

- ▶ Losses are introduced by removing some least significant bits (LSB) from the differential coefficients.
- ▶ Different quality levels (QL) are defined, e.g.
 - ▶ QL 1: removal of 1 bit from ΔH_0 and 1 bit from ρ_1 . This is equivalent to remove 1 bit per pixel.
 - ▶ ...
 - ▶ QL 8: we drop 8 bits from ΔH_0 (that is, this coefficient is completely removed), 6 bits from ΔH_1 , 4 bits from ΔH_2 , 2 bits from ΔH_3 , and all ρ remainders

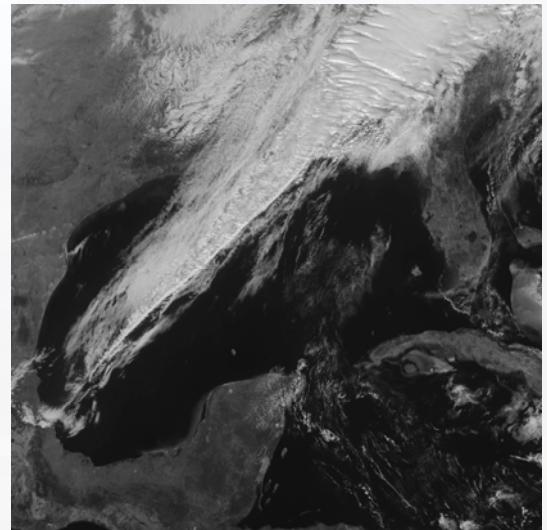
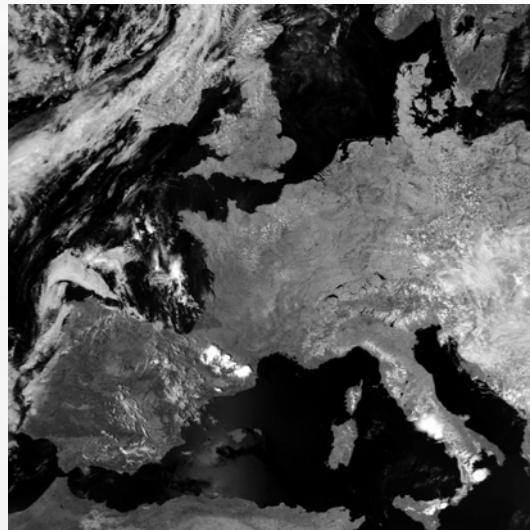
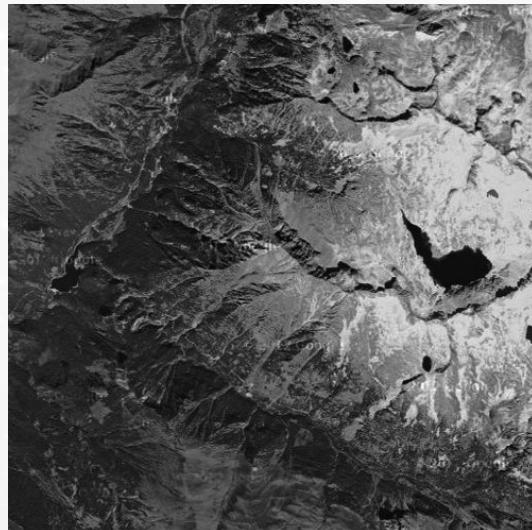
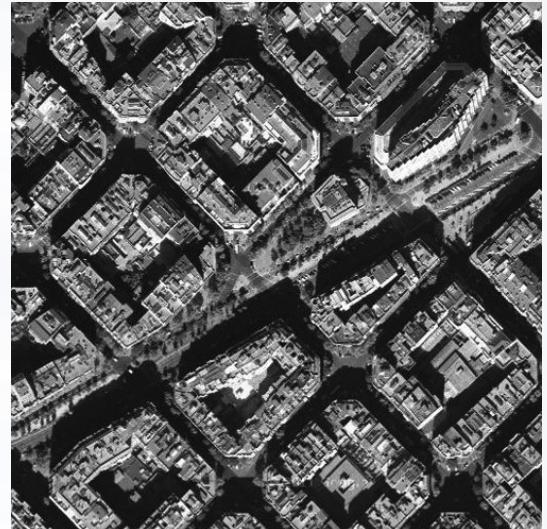
Lossy HPA+FAPEC

The removal of n LSB is performed with a right shift
This is equivalent to dividing the coefficient by 2^n



Histogram of the ΔH_0 of QL2 with Lossy FAPEC

Image compression corpus: examples



Lossless results

- ▶ The HPA algorithm improves the compression of some image w.r.t. FAPEC alone
- ▶ C_R of FAPEC prototype are close to the C_R achieved by CCSDS 122.0
- ▶ FAPEC + HPA is faster than CCSDS 122.0

Compressor	FAPEC		FAPEC + HPA		CCSDS 122.0	
	C_R	Exec. time	C_R	Exec. time	C_R	Exec. time
banyoles-1024x592.raw	1.28	78 ms	1.29	164 ms	1.45	209 ms
catedral-1024x592.raw	1.09	81 ms	1.08	171 ms	1.17	231 ms
eixample-1024x592.raw	1.13	80 ms	1.14	172 ms	1.25	228 ms
pirineus-1024x592.raw	1.32	75 ms	1.29	166 ms	1.47	211 ms
florida-3600x2992.raw	1.85	1201 ms	1.76	2660 ms	2.27	2999 ms
eixample-1024x592.raw	1.59	210 ms	1.54	449 ms	1.93	539 ms

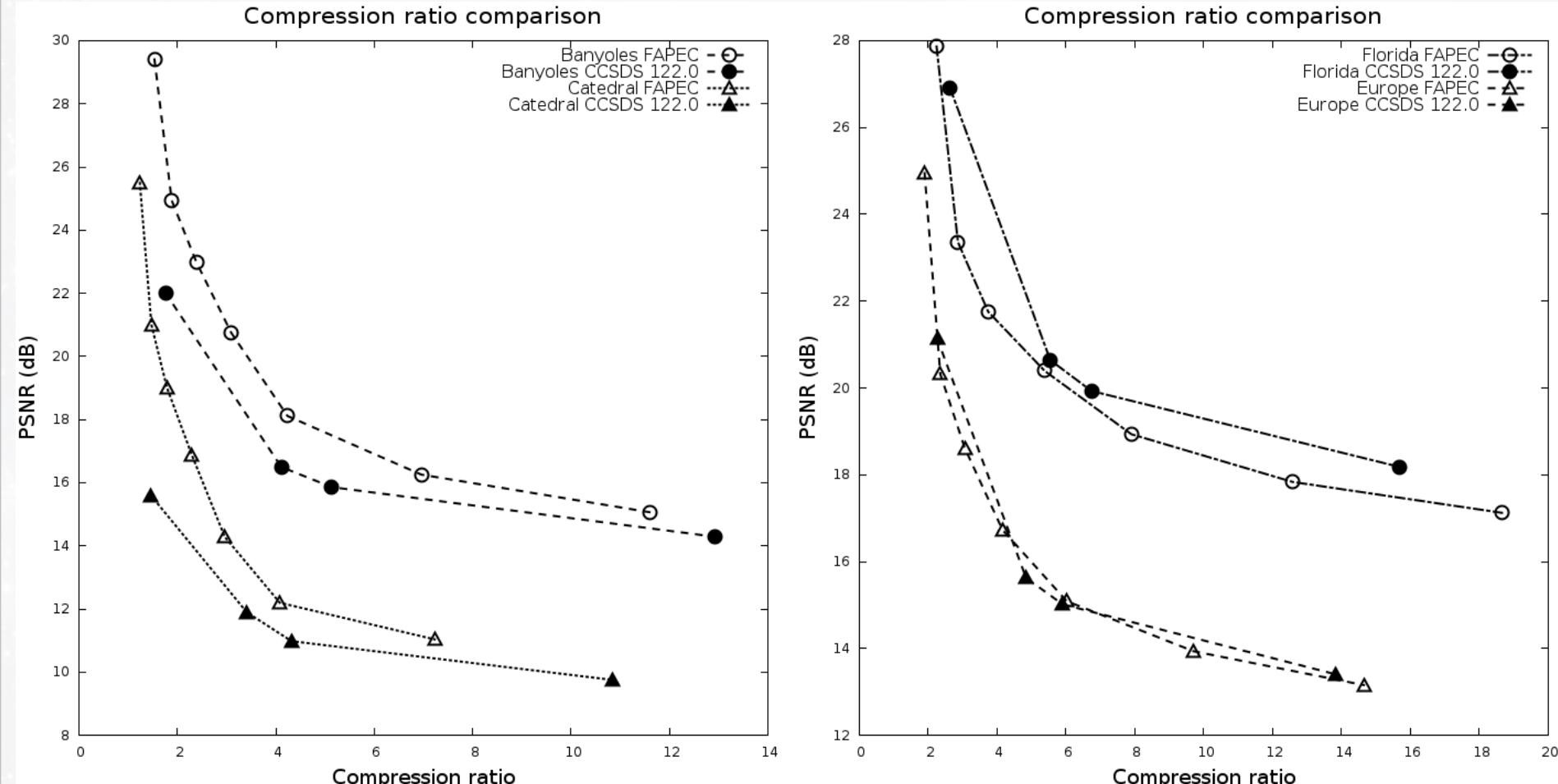
Noisy Lossless results

- ▶ We simulated a noisy image by adding white noise to every 100th pixel
- ▶ The loss of CR is between half and one third with FAPEC, compared to the loss with CCSDS 122.0

Compressor	HPA+FAPEC		CCSDS 122.0	
Performance parameter	C _R	C _R loss	C _R	C _R loss
banyoles-1024x592.raw	1.27	1.25 %	1.41	3.22 %
catedral-1024x592.raw	1.07	0.34 %	1.16	0.77 %
eixample-1024x592.raw	1.13	0.59 %	1.23	1.44 %
pirineus-1024x592.raw	1.27	1.05 %	1.43	2.90 %
florida-3600x2992.raw	1.72	2.53 %	2.17	4.50 %
europe-1408x1248.raw	1.50	2.49 %	1.78	8.11 %

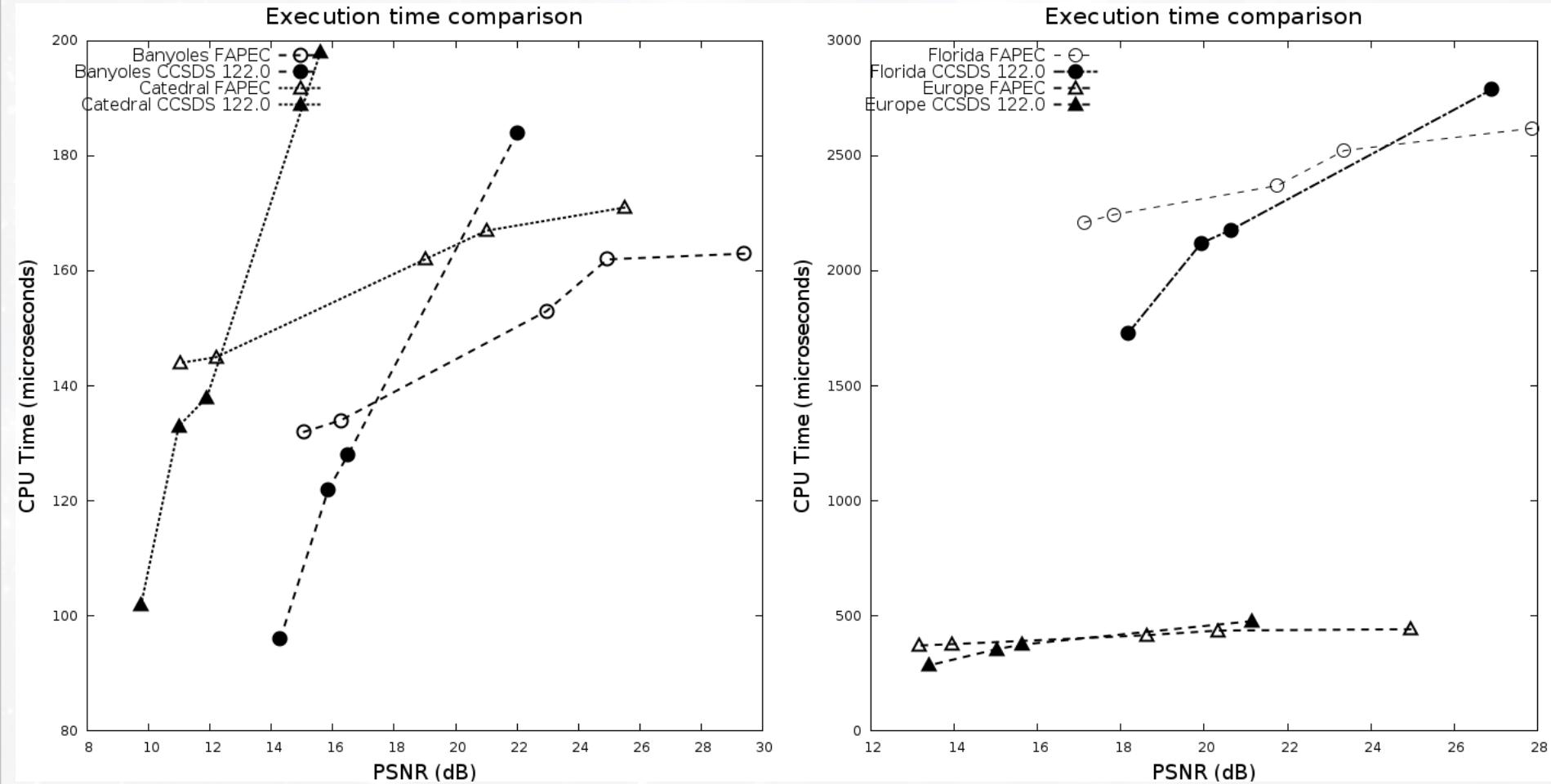
Lossy results - PSNR

In general, FAPEC + HPA achieves a better PSNR



Lossy results – Execution time

CCSDS 122.0 is slower than FAPEC + HPA in lower QLs,
while it runs faster at higher QLs



Lossy compression

► Example with 'Banyoles' image

FAPEC

PSNR: 16.26 dB

$C_R = 6.96$



CCSDS 122.0

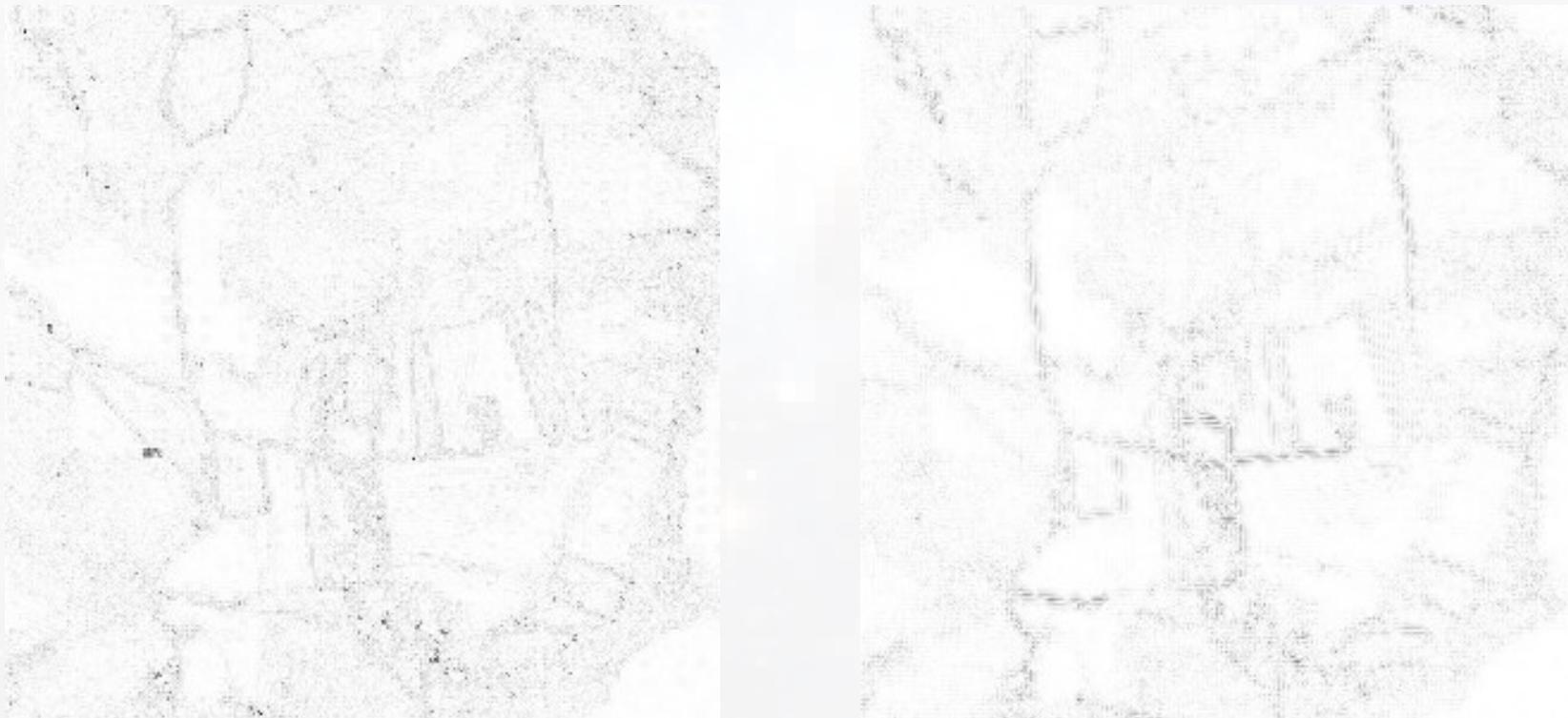
PSNR: 16.5 dB

$C_R = 4.11$



Lossy compression

- ▶ Error map between FAPEC and CCSDS 122.0
- ▶ FAPEC shows more distributed errors, sharper edges
- ▶ CCSDS 122.0 shows more errors along the edges



Conclusions and forthcoming work

- ▶ We developed a new pre-processing stage for FAPEC, specific for images
- ▶ It improves the C_R of FAPEC alone with some images and achieves a C_R close to that of CCSDS 122.0
- ▶ It runs faster than CCSDS 122.0 and is more robust in presence of noise or outliers
- ▶ The reconstructed images show sharper edges at low quality levels
- ▶ Future improvements:
 - ▶ Make the algorithm able to handle any image size
 - ▶ Implement colour and hyperspectral compression
 - ▶ Improve the FAPEC core for very low entropy levels
 - ▶ Further optimize the algorithm to reduce the execution time
 - ▶ Implement a logic for fixed-rate lossy compression



UNIVERSITAT POLITÈCNICA
DE CATALUNYA

IEEC

DAPCOM
Data services

UNIVERSITAT DE BARCELONA
B

Hierarchical Pixel Averaging: A new image compression approach

Riccardo Iudica (IEEC) – *riccardo.iudica@outlook.com*

Enrique García-Berro (IEEC/UPC)

Jordi Portell (IEEC/DAPCOM/UB)

Presented by Riccardo Iudica