Considering the Reconstruction Loop for Watermarking of I and P Frames of H.264/AVC

Zafar SHAHID, Marc CHAUMONT and William PUECH

LIRMM, UMR CNRS 5506, Université Montpellier II

zafar.shahid@lirmm.fr, marc.chaumont@lirmm.fr, william.puech@lirmm.fr

Problem Statement

THE PROBLEM:

- To embed watermark in H.264/AVC quantized transformed coefficients (QTCs) while taking into account the reconstruction loop.

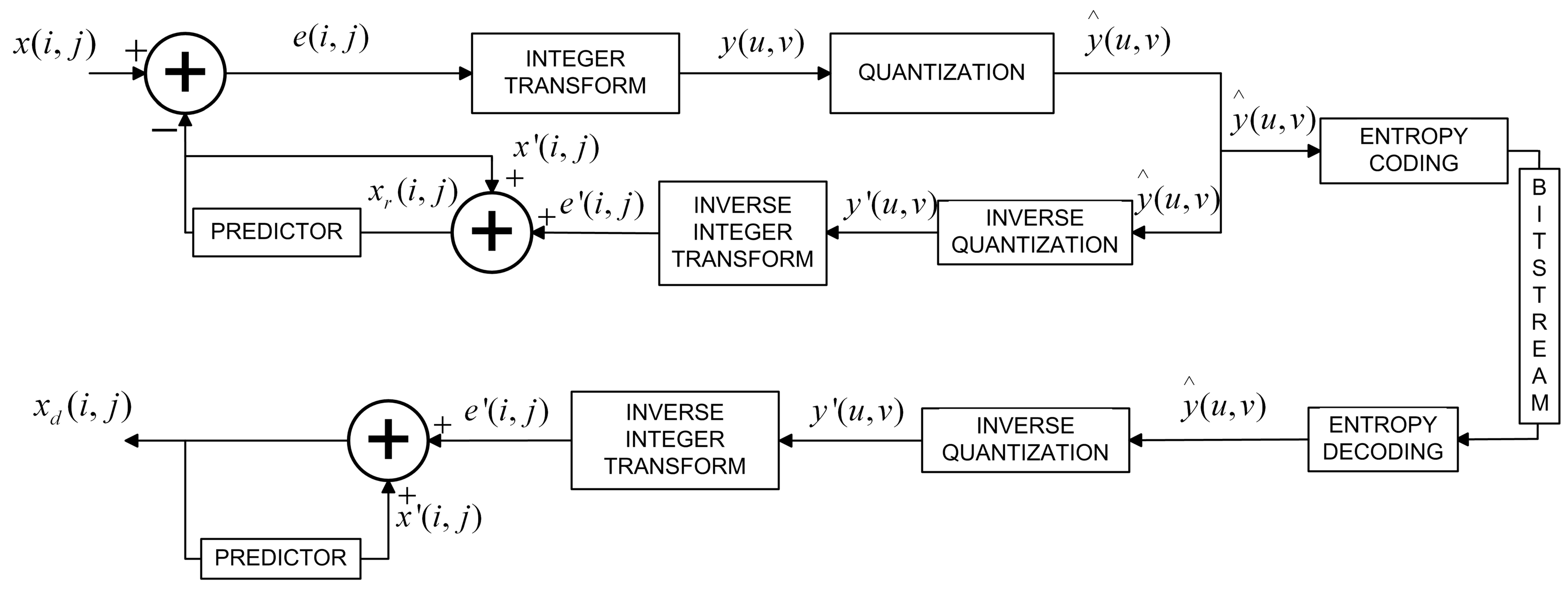
OUR APPROACH:

- Watermark is embedded into only non-zero AC QTCs: compression efficiency of Run Length Coding is preserved.
- QTCs which are to be watermarked should have magnitude greater than 1: CAVLC encodes many of them as Trailing ones (T1s) and changing them will affect the compression efficiency of CAVLC.
- Embedding watermark inside reconstruction loop avoids mismatch between encoder and decoder and also takes into account the bitrate/quality tradeoff because of watermarking of QTCs.

H.264/AVC

H.264/AVC is the state of the art video codec and performs better than previous standards owing to many new tools including:

- 4X4 integer transform.
- Better entropy coding techniques:
  - CAVLC (Adaptive technique based on Huffman coding),
  - CABAC (Adaptive technique based on Arithmetic coding).
- Quarter pixel motion estimation.
- Multiple block size.
- Multiple reference frames.



The Proposed Method

- In the proposed scheme, watermark is not embedded in the bitstream. Rather it is embedded during the encoding process.
- Watermark message is inserted in the AC QTCs which are above a certain threshold. The embedding is performed inside the reconstruction loop in such a way:
  - There is no mismatch on encoder and decoder side, thus avoiding the drift.
  - Since embedding a watermark in a video bitstream affects PSNR and bitrate of the picture, rate-distortion algorithm should work on the watermarked QTCs, thus taking into account the watermarking affect. So the Lagrangian rate distortion is given as: $J_w = D_w + \lambda R_w$.
- For the embedding process, let $\hat{Y}$ be a QTC to be watermarked. The watermark is embedded as: $\hat{Y}_W = f(\hat{Y}, W, [K])$. Where W is the watermark message and [K] is the optional key.
- Watermark can be embedded in 1, 2 or '1 or 2' LSBs of QTCs. For embedding watermark in '1 or 2' LSBs, the embedding and extraction schemes are given below.

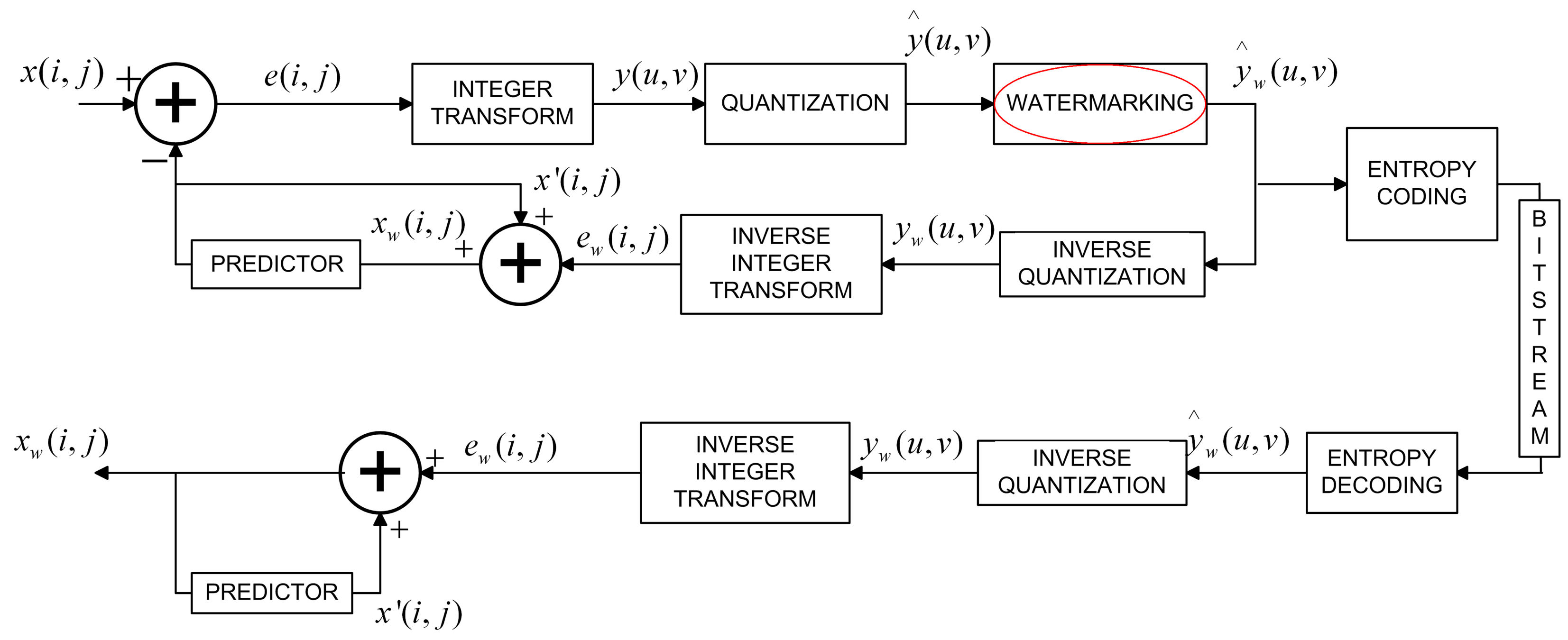
Embedding strategy in '1 or 2' LSBs:

1: **if** $|QTC| > 3$ **then**
2: $|QTCw| \leftarrow |QTC| - |QTC| \ mod\ 4 + WMBits$
3: **else**
4: **if** $|QTC| > 1$ **then**
5: $|QTCw| \leftarrow |QTC| - |QTC| \ mod\ 2 + WMBit$
6: **end if**
7: **end if**
8: **end**

Extraction strategy using '1 or 2' LSBs:

1: **if** $|QTC| > 3$ **then**
2: $WMBits \leftarrow |QTC_w| \ mod\ 4$
3: **else**
4: **if** $|QTC| > 1$ **then**
5: $WMBit \leftarrow |QTC_w| \ mod\ 2$
6: **end if**
7: **end if**
8: **end**

Watermark capable H.264/AVC



Experimental Results

- For experimental simulation, H.264/AVC JSVM 10.2 in AVC mode is used.
- 150 frames in CIF resolution.
- Analysis at QP values of 18 & 36.
- Intra period is 15 in case of I & P frames.
- On average increase in frame size is 3.2%, 2.7% and 2.8% for I, P and I&P respectively for QP value of 18.

- Overall analysis of watermark embedding in *intra* frames for *foreman* sequence.

		Payload Kbits/frame	Frame Size Kbytes	PSNR dB
QP 18 I frames	0	0	2.815	44.883
	LSB1	9.375	2.889	43.801
	LSB2	5.605	2.875	43.605
	LSB1&2	12.484	2.909	42.928
QP 36 I frames	0	0	0.377	32.628
	LSB1	0.206	0.381	32.536
	LSB2	0.012	0.377	32.612
	LSB1&2	0.214	0.381	32.526

- Overall analysis of watermark embedding in *intra* and *inter* frames for *foreman* sequence at QP=18.

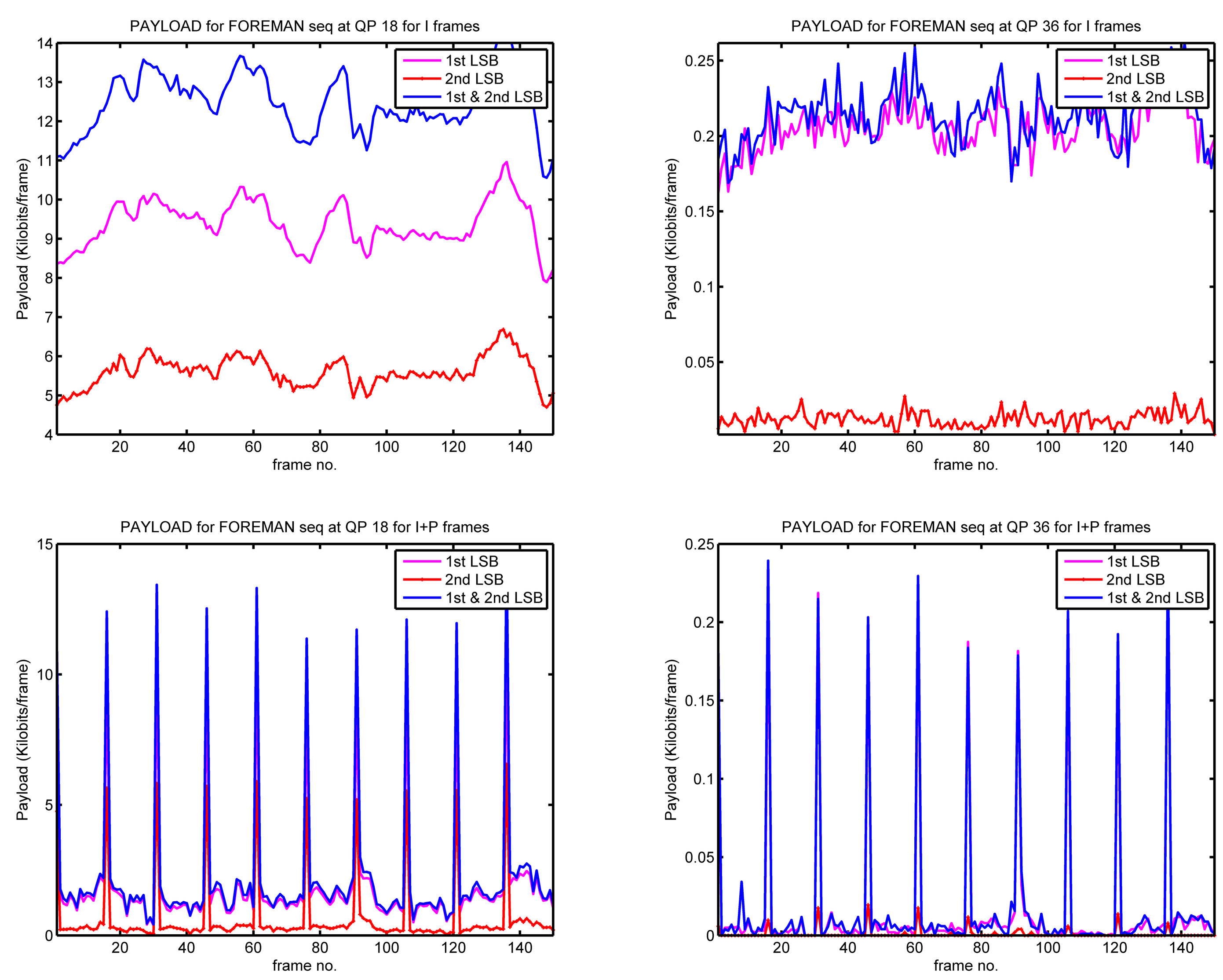
		Payload Kbits/frame	Frame Size Kbytes	PSNR dB
QP 18 I frames	0	0	2.818	44.876
	LSB1	9.352	2.892	43.800
	LSB2	5.586	2.878	43.605
	LSB1&2	12.452	2.913	42.917
QP 18 P frames	0	0	1.317	44.541
	LSB1	1.378	1.343	44.302
	LSB2	0.280	1.333	44.449
	LSB1&2	1.530	1.354	44.230
QP 18 I + P	0	0	1.417	44.563
	LSB1	1.909	1.446	44.269
	LSB2	0.633	1.436	44.392
	LSB1&2	2.258	1.458	44.144

- Overall analysis of watermark embedding in *intra* and *inter* frames for *foreman* sequence at QP=36.

		Payload Kbits/frame	Frame Size Kbytes	PSNR dB
QP 36 I frames	0	0	0.376	32.613
	LSB1	0.198	0.380	32.523
	LSB2	0.011	0.376	32.589
	LSB1&2	0.207	0.381	32.520
QP 36 P frames	0	0	0.074	32.353
	LSB1	0.005	0.074	32.315
	LSB2	1×10^{-4}	0.073	32.336
	LSB1&2	0.005	0.074	32.318
QP 36 I + P	0	0	0.094	32.370
	LSB1	0.017	0.094	32.329
	LSB2	8×10^{-4}	0.093	32.353
	LSB1&2	0.019	0.095	32.331

Framewise Analysis

- Framewise analysis of payload capability in I & P at QP 18 & 36.



Conclusion

Encouraging results in the following contexts:

- RD optimized watermark for I & P frames.
- Higher payload with :
  - negligible increase in bitrate,
  - minimum compromise on PSNR.
- P frames are also good for watermarking owing to motion and texture masking.