A JOINT TRELLIS CODED QUANTIZATION (TCQ) DATA HIDING SCHEME IN THE JPEG2000 PART 2 CODING FRAMEWORK

D. Goudia[1,2], *M. Chaumont*[1,3], *W. Puech*[1] *and N. Hadj Said*[2]

[1] University of Montpellier II, Laboratory LIRMM,
UMR CNRS 5506, 161, rue Ada, 34095 Montpellier cedex 05, France
phone: + (33) 467 41 85 85 , fax: + (33) 467 41 85 00 , email: dalila.goudia@lirmm.fr
[2] University of Science and Technologies of Oran (USTO), Oran, Algeria
[3] University of Nîmes, Rue du Docteur Georges Salan, 30021, Nîmes Cedex 1, France

ABSTRACT

A new Treillis Coded Quantization (TCQ)-based data hiding method for JPEG2000 part 2 is presented. The main feature of the proposed scheme is that it jointly achieves quantization and information embedding. The characteristics of the TCQ quantization are exploited to quantize selected wavelet coefficients with specific codebooks. These codebooks are associated with the values of the data to be inserted. The wavelet coefficients included in the data hiding process are selected carefully in order to survive the entropy coding stage and the rate-distortion optimization stage of JPEG2000 compression. Experimental results are given to demonstrate the performance of the proposed algorithm.

1. INTRODUCTION

With the increasing use of multimedia technologies, image compression becomes an essential process in digital imaging. JPEG2000 [1] is a compression standard developed by the Joint Photographic Experts Group (JPEG) that supports lossy or lossless compression of grayscale or color images. Enrichment and protection in the JPEG2000 images is an important issue. Data hiding techniques are a good solution to content description and protection. The important requirement of data hiding is to embed the maximum amount of data in the host image without perceptually distorting it. The information embedded must be recovered without error during the extraction stage. In the data hiding scenario, no attacks are taken into account and it really differs from the watermarking scenario [2].

Instead of treating data hiding and compression separately, it is beneficial to consider the joint design of data hiding and compression system. Given that images are usually compressed before transmission or storage, data hiding in a compressed data format, such as the JPEG2000 format, should be a better choice. In this case, compression is no longer considered as an attack because the data is hidden during the compression process. Several attempts to introduce data hiding technique into JPEG2000 coding pipeline have been reported in the literature [3, 4, 5]. Chen *et al.* [3] proposed to perform hiding in the compressed bitstream from rate allocation by simulating a new rate-distortion optimization stage. The new bitrate must be smaller than the original one. A simulated layer optimization induces readjustments of bits in the output layers of the compressed bitstream. These readjustments cleared space in the last output layer for hiding data. This scheme is applicable only for bitrates lower than 0.65 bpp. Su *et al.* [4] proposed to embed the hidden information in the JPEG2000 compressed bitstream by exploiting the *lazy mode* coding option. Information hiding is achieved after the rate-distortion optimization stage (Tier2 coding) by modifying the data in the magnitude refinement passes. The main drawback of this scheme is that the information hiding procedure is operated in the special JPEG2000 lazy mode. Moreover, it requires the target bitrate to be higher than 2 bpp to allow the activation of the lazy mode. Thomos *et al.* [5] presented a sequential decoding of convolutional codes for data hiding in JPEG2000 images. The hiding payload is small (between 100 and 250 bits in 512 x 512 images). Furthermore, the hidden data is not extracted correctly at bitrates lower than 0.3 bpp despite the use of an error correcting code. It should be noted that all these schemes integrate an additional embedding/extraction stage in the JPEG2000 compression/decompression process.

Hiding information during the quantization stage ensures that the distortion induced by the information embedding will be minimized and thus obtaining a good image quality. It represents a real joint solution because the quantization and the data hiding aspects are considered together. The quantization and information hiding processes are performed simultaneously using a single component. In this paper, we propose a joint data hiding and quantization scheme based on Treillis Coded Quantization (TCQ) incorporated in the JPEG2000 part 2 codec [6]. The data is hidden during the quantization process without any additional stage for hiding data. Our technic uses the properties of the entropy coded TCQ (ECTCQ) as specified in the recommendations of the part 2 of the standard [6].

The specific organization of the paper is as follows. Section 2 presents the JPEG2000 coder. Section 3 recalls the TCQ quantization of JPEG2000 part 2. Section 4 describes the proposed joint JPEG2000 coding and data hiding TCQ-based scheme. Section 5 presents experimental results and finally Section 6 concludes this work.

2. JPEG2000 CODER

JPEG2000 [1] is based on discrete wavelet transformation and it provides several important features such as resolution/quality progressive image transmission, better resilience to bit-errors and Region of Interest (ROI) coding. The main encoding procedures of JPEG2000 (Part 1) are the following:first, the original image undergoes some pre-processing operations (level shifting and color transformation). Then, the input image is transformed by the discrete wavelet transform (DWT) into a collection of sub-bands (LL, HL, LH and HH) which may be organized into increasing resolution levels. It is then quantized by a dead-zone uniform scalar quantizer. The quantized wavelet coefficients in each sub-band are partitioned into small rectangular blocks which are called *code-blocks*. Each *code-block* is encoded independently during the Tier 1 encoding stage by using a bit-plane coder named Embedded Block Coding with Optimal Truncation (EBCOT). The *code-block* is coded one bit-plane at a time, starting from the most significant bit-plane to the least significant bit-plane. Each individual bit-plane is coded with three coding passes. The next step is the Tier 2 encoding stage:based on the result of rate control stage (rate-distortion optimization stage), the bitstreams of all the code blocks are combined into a single bitstream. During rate allocation, truncation points will be attached to the bitstream from each encoding pass. The rate control algorithm finds the optimal bit allocation for all *code-blocks*, such that the total distortion is minimized subject to the target bitrate. The effective truncation point is chosen from the feasible ones in each *code-block* so as to truncate the bitstream in an optimal way. The coded data are outputted to the code-stream in packets and the JPEG2000 file stream is finally formed.

3. TRELLIS CODED QUANTIZATION (TCQ)

JPEG2000 Part 2 [6] allows for the use of Trellis Coded Quantization (TCQ) as a replacement for scalar quantization. TCQ is a particular kind of vector quantization proposed by Marcellin and Fischer [7]. TCQ employs the trellises and set partitionning ideas from trellis coded modulation (TCM) [8] to achieve performance near rate distortion bound. A uniform scalar quantizer with step size Δ is partitioned into four subsets called D_0, D_1, D_2 and D_3. Subsets D_i are used to label the branches of a trellis. A trellis is a state transition diagram (that takes time into account) for a finite state machine. Each branch in the trellis represents a transition from one state to another, at the next point in time. Given an initial state at t = 0, the path can be specified by a binary sequence, since there are only two possible transitions from one state to another. Fig. 1 shows a single stage of a typical 8-state trellis with branch labeling. In TCQ, we pick up connected branches in trellis by using Viterbi algorithm [9] to quantize a sequence in order to have less accumulated distortion. The Viterbi Algorithm produces two sequences: the first one is a binary sequence defining the minimum distortion path. The second one is the sequence of corresponding TCQ indices. The dequantization of TCQ indices at the decoder is straightforward. Given the initial state and the path sequence, the decoder is able to reproduce the reconstructed values by using the sequence of indices specifying which codeword was chosen from the appropriate subset D_0, D_1, D_2 or D_3 at each stage.

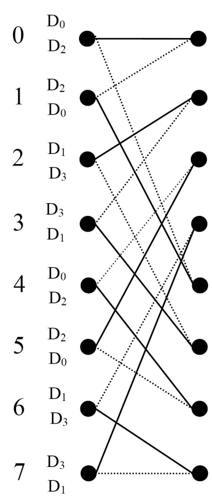

Figure 1: A single stage of an 8-state trellis with branch labeling used in JPEG2000.

The variant of TCQ used in the part 2 of the JPEG2000 standard is the Entropy Coded TCQ (ECTCQ). The two scalar quantizers associated with each state in the trellis are combined into *union quantizers* A_0 and A_1. The two union quantizers $A_0 = D_0 \cup D_2, A_1 = D_1 \cup D_3$ are illustrated in Fig. 2. This figure shows the reconstruction values $\hat{x}$ corresponding to each union quantizer. The corresponding indices $q(A_i)$ are also shown. At each state, we could choose between one of the two quantizers belonging to the union quantizer to quantize the input sequence **x**. Codewords from the two subsets (within a union quantizer) differ in the least significant bit (LSB) of their index $q(A_i)$. Note that by construction of the trellis, the least significant bit of each quantized indice $q(A_i)$ corresponds to the path since there are two possible codewords for each index (D_0 or D_2 and D_1 or D_3). The least significant bit determines the path through the trellis.

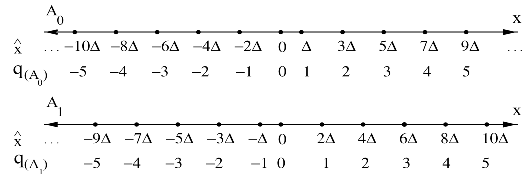

Figure 2: Union quantizers for TCQ in JPEG2000.

4. THE PROPOSED JOINT JPEG2000 AND DATA HIDING SCHEME

4.1 The data hiding insertion and extraction method

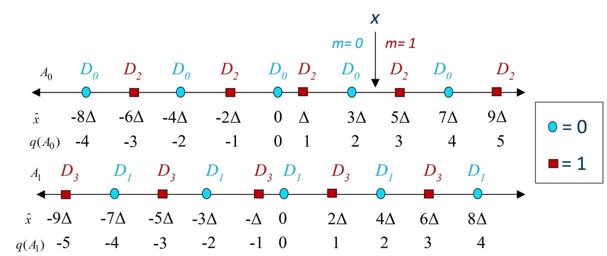

Figure 3: The QIM principles applied to TCQ union quantizers.

Our data hiding strategy is derived from the QIM [10] (Quantization Index Modulation) principles and is integrated into a TCQ approach. To embed the data hiding information, a sequence of quantizers can be used. The elements of the message to hide act as an index that select a particular quantizer from this sequence of quantizers. In other words, quantizers are modulated according to the data to hide. The host sample signal is then quantized with the associated quantizer.

TCQ quantization as specified in JPEG2000 part 2 recommendations works with two union quantizers, A_0 and A_1, which are used to label the branches of a trellis. The trellis is traversed by following one of the two branches that emanate from each state. The bold branch is labeled by D_0 or D_1 and the non-bold branch with D_2 or D_3 as shown in Fig. 1. We propose the following principle as illustrated Fig. 3:

- For union quantizer A_0: if the bit to embed is the bit 0, then the quantizer D_0 is used to quantize the wavelet coefficient. Otherwise the quantizer D_2 is used.
- For union quantizer A_1: if the bit to embed is the bit 0, then the quantizer D_1 is used to quantize the wavelet coefficient. Otherwise the quantizer D_3 is used.

The choice of the branch to traverse is determined by the value of the bit to be embedded. This is achieved by removing non-bold branches when we embed a 0-bit, and supressing bold branches when we embed a 1-bit. In other words, the path corresponds to the hidden data. However, there is a problem when we integrate this method in the JPEG2000 coding pipeline. EBCOT and the rate-distortion optimization stage must be taken into account in the design of a joint data hiding and JPEG2000 scheme. In JPEG2000, the bitstream truncation produces some bit discards after rate allocation, as described in Section 2. There will be $p >= 1$ missing LSBs after rate allocation for a given bitrate. Consequently, the embedded information will not be perfectly recovered knowing that the LSBs of the TCQ indices determine the path through the trellis. A reasonable policy used in JPEG2000 decompression is to set the missing LSBs to 0 to obtain an approximate reconstruction value. This is not appropriate for a TCQ-based joint scheme because losing LSBs means losing some of the hidden data. Significant coefficients with higher bit-planes have a greater chance of having their TCQ indices being kept complete after JPEG2000 compression. We propose to embed data only in the significant coefficients which have a better chance of survival. These coefficients are called *selected coefficients*. Therefore, the trellis is pruned only at the transitions which correspond to the *selected coefficients*. Moreover, in order to be sure that the LSB value (the path information) will be unchanged after rate allocation, we move the LSB bit-plane of the TCQ indices of the *selected coefficients* to a higher bit-plane.

The message to hide is noted $\mathbf{m} \in \{0,1\}^N$. In order to secure the data to hide, we shuffle (scatter) pseudo randomly the bits of the message to be hidden with a secret key. We obtain another message noted $\mathbf{b} \in \{0,1\}^N$. It prevents all unauthorized users to retrieve the proper values of the hidden data during the JPEG2000 decompression. For each *code-block*, the trellis is pruned at the transitions associated to the *selected wavelet coefficients*. The pruning consists of selecting the right branch depending on the value of the bit to embed $\mathbf{b}_k, k \in [0,N]$ at the considered transition step. In other words, for transition i in the trellis which corresponds to the position of one of the *selected coefficients*, the quantizer is selected according to the value of $\mathbf{b}_k$. The process of quantization produces the sequence of TCQ quantization indices $\mathbf{q}$ given by:

$$\mathbf{q}[i] = Q_{D_j}(\mathbf{x}[i]), \tag{1}$$

where Q is the quantization function and D_j is the quantizer used to quantize $\mathbf{x}$[i]. D_j is selected according to the bit to hide $\mathbf{b}_k$. For a given step size Δ, $\mathbf{q}$[i] can be computed as: $\mathbf{q}[i] = sign(\mathbf{x}[i]) \left\lfloor \frac{|\mathbf{x}[i]|}{\Delta} \right\rfloor$. We are able to extract the embedded message during the inverse TCQ quantization stage of the JPEG2000 decompression. For each *code-block*, the decoder produces an estimate of $\mathbf{x}$ as follows :

$$\hat{\mathbf{x}}[i] = \bar{Q}^{-1}_{D_j}(\mathbf{q}[i]), \tag{2}$$

where $\bar{Q}^{-1}$ is the dequantization function. For a given step size Δ, the reconstructed value $\hat{\mathbf{x}}$ can be computed as: $\hat{\mathbf{x}}[i] = sign(\mathbf{q}[i])(|\mathbf{q}[i]| + \delta)\Delta$ where δ is a user selectable parameter within the range $0 < \delta < 1$ (typically δ = 0.5). The hidden message $\mathbf{b}$ can then be determined by examining each branch in the path through the trellis at transitions which correspond to the positions of the *selected coefficients*: if it is a bold branch, a 0-bit is recovered. Otherwise, a 1-bit is retrieved.

The proposed data hiding method have similarities with the dirty paper trellis codes (DPTC) [11]. Both methods rely on the use of a trellis associated to a codebook and on the pruning of this trellis. However, we use a quantization codebook partitioned into subsets while Miller *et al.* use a pseudo-random code. Moreover, the embedding of the watermark is done in a different way. Our joint scheme integrate a quantization-based method whereas DPTC codes optimally embed a watermark by applying an iterative embedding procedure with the constraint of minimizing the perceptual distance while keeping a fixed robustness. The codeword is determined by using a correlation instead of a quantization.

4.2 The proposed joint JPEG2000 and data hiding scheme

The block diagram of the joint JPEG2000 encoder and data hiding scheme is shown in Fig. 4. First, the original image is processed by some pre-processing operations. Then, it is decomposed by the DWT into a collection of sub-bands. Afterwards, we select the coefficients included in the data hiding process within the wavelet coefficients of the HL, LH and HH detail sub-bands of all resolution levels except the first one. The selection criteria that allow us to perform the selection will be discussed Section 4.3. The number of selected coefficients allow us to determine the amount of data to hide. Next, the data is hidden during the TCQ quantization stage which is performed independently on each *code-block*. After TCQ quantization with data hiding, EBCOT executes the entropy coding. Subsequently, rate-distortion optimization will arrange the *code-blocks* bitstreams into quality layers to form the JPEG2000 bitstream.

Depending on the target compression ratio and on the information content of the processed image, some bits of the hidden data will be lost after rate-distortion optimization (bitstream truncation). To ensure the proper recovery of the hidden data, a verification process is performed after rate allocation to check if there is no data loss. This process consists of performing an EBCOT decoding and data extraction. If the embedded information is not perfectly recovered, a feedback process is employed to modify the value of the selection criteria for the considered *code-blocks* where erroneous bits were found. This allows us to select the coefficients that have survived the previous rate allocation stage and to exclude those who did not survive. Of course, we will have less *selected coefficients* than before. It means that less bits will be hidden during the next TCQ quantization. In this way, we may tune the selection criteria recursively during the ongoing process of TCQ quantization, EBCOT, rate-distortion optimization and verification until there is no truncation to the hidden data during the JPEG2000 compression procedure. At each iteration of this feedback process, we make a new selection and embedding. The algorithm stops when the hidden bits are extracted correctly during the verification process. The number of iterations depends on the target bitrate, the selection criteria and the content of the processed image. The payload is determined by the number of *selected coefficients*. So, we will have a different hiding payload for each bitrate. Basically, hiding payloads are smaller for images compressed at lower bitrates.

The following steps ensure the extraction of the hidden data: the image bitstream is decoded by the EBCOT decoder. Then the hidden data is extracted during the inverse TCQ quantization according to the previous positions of the *selected coefficients* respectively from each *code-block*. After data-hiding extraction, the inverse DWT and the post-processing operations are performed to reconstruct the image.

4.3 Selection of the wavelet coefficients included in the data hiding process

Data will be hidden in the least significant bits of the TCQ indices which represent the path through the trellis. We can represent the TCQ index q of the wavelet coefficient x in *sign magnitude form* as

$$q = s, q_0 q_1 q_2 .... q_{L-1}, \tag{3}$$

where s is the sign, q_0 is the most significant bit (MSB), and q_{L-1} is the least significant bit (LSB) of q. L is the number of bits required to represent all quantization indices in the *code-block*. The calculation of the selection threshold τ_{IBP} (IBP: Intermediate Bit-Plane) for each *code-block* will allow us to select a sequence of significant coefficients $\mathbf{S}$. Assuming that we have L bit-planes in the current *code-block* $\mathbf{C}$, τ_{IBP} is computed as follows:

$$\tau_{IBP} = \lfloor \alpha * L \rfloor, \tag{4}$$

where α is a real factor between 0 and 1 initialized with a predefined value for each sub-band. The choice of the predefined values is done empirically through experimentation with several images. Table 1 shows predefined values of α for some target bitrates. The α values have an impact on the payload and the number of iteration of the feedback process. The selection of coefficients included in the data hiding process is done as follows:

$$\text{if } \lceil \log_2(|\mathbf{q}[i]| + 1) \rceil > \tau_{IBP}, \text{then add } \mathbf{x}[i] \text{ to } S_C, \tag{5}$$

where $\lceil \log_2(|\mathbf{q}[i]| + 1) \rceil$ is the number of bits used to represent the TCQ index q of the i^{th} wavelet coefficient x of the *code-block* $\mathbf{C}$. We select coefficients whose TCQ indices have their absolute magnitude bits greater than τ_{IBP}. In the case of a data loss after rate allocation, the value of τ_{IBP} is incremented during the backward process and we re-run selection and embedding until the hidden message is correclty recovered.

To be sure that the path will not be partially lost during the rate-distortion optimization stage, especially at low bitrates, we propose

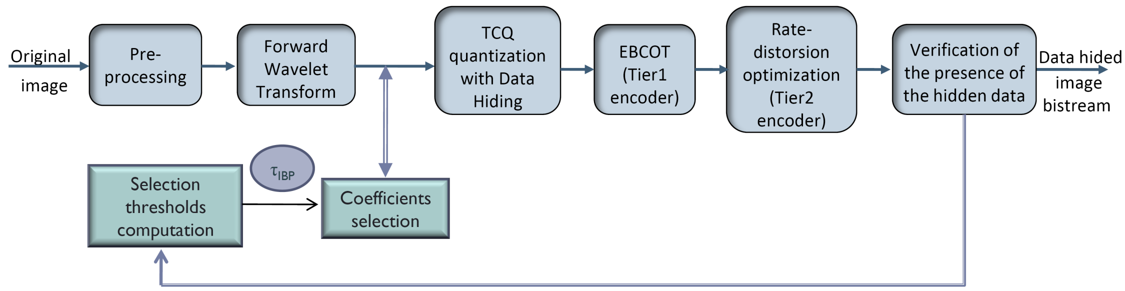

Figure 4: The block diagram of the joint JPEG2000 codec-data hiding scheme.

Resolution level	Sub-band	bitrate (bpp)	α
3	1	2.5	0.20
		1.6	0.33
		0.2	0.50
4	2	2.5	0.33
		1.6	0.33
		0.2	0.75

Table 1: Predefined α values for different target bitrates

to displace the LSBs of the TCQ indices of the *selected coefficients* to another position. The new position is located at q_1 (Eq. 3). It is the most higher position at which we can move the LSB without inducing a left shift:in case the LSB value is 0, we can indeed obtain a left shift when we move the LSB at q_0. The thresholds τ_{IBP} for each *code-block* are stored as side information and transmitted to the decoder. In this way, we are able to retrieve the right positions of the selected TCQ indices during the decompression. Thus, we do not need to save the localization of the selected quantization indices. The size of the transmitted file is very small compared to the hiding payload and to the JPEG2000 file size. This file can be encrypted to increase security.

5. EXPERIMENTAL RESULTS

To implement our joint JPEG2000-data hiding scheme, we choose to use the OpenJPEG library [12] which is an open-source JPEG2000 codec written in C language. We replace the scalar uniform quantization component by a JPEG2000 part 2 compliant TCQ quantization module. Simulations were run on several grayscale images of size 512 x 512. We present here the results obtained for four well known test images : Lena, Gold, Girl and Barbara. A five levels of irreversible DWT 9-7 is performed. The data to hide is embedded in the HL, LH and HH detail sub-bands of the second, third, fourth and fifth resolution levels. We have a total of 21 *code-block* included in the data hiding process. The size of the side information file containing the 8-bit thresholds τ_{IBP} is equal to 168 bits (21 x 8 = 168). We set the compression ratio from 2.5 bpp to 0.2 bpp. In Table 2, we summarized the performance of the proposed joint scheme in terms of PSNR. The results confirm that the TCQ-based scheme exhibits a good quality performance. Indeed, the PSNR obtained for those images are greater than 30 dB for all compression bitrate except for Barbara image at 0.5 bpp and 0.2 bpp. We should also notice that the quality degradation resulting from information embedding is relatively small when we compare between the PSNR computed for the compressed image obtained with the classical JPEG2000 part 2 coder and the PSNR computed for the compressed and data hidden image as shown in Table 2. For example, there is a loss of 1.29 dB between the two PSNR for Gold image at 2 bpp and 0.95 dB for Barbara image at 0.2 bpp. Moreover, the PSNR obtained for Gold image at 1 bpp and 0.5 bpp are slightly greater than those obtained by the classical JPEG2000 coder. This is due to the preservation of the LSB values of TCQ indices of the selected coefficients. This provides reconstructed values closer to the original ones while the policy used in JPEG2000 decompression is to set the missing LSBs to 0 to obtain an approximate reconstruction value. It is clear that there is no overhead in the JPEG2000 file format introduced by the data hiding process. In fact, the data is hidden during the quantization stage and is part of the TCQ indices within the JPEG2000 bitstream.

Image test	Bitrate (bpp)	PSNR (dB) with JPEG2000	PSNR (dB) with the joint scheme	Difference between the PSNRs (dB)
Lena	2.5	47.47	45.17	2.3
	2	45.33	43.83	1.5
	1.6	43.38	42.63	0.75
	1	41.55	40.93	0.62
	0.5	40.03	38.10	1.93
	0.2	36.56	36.00	0.56
Gold	2.5	43.63	42.32	1.31
	2	43.33	42.04	1.29
	1.6	43.30	42.09	1.21
	1	41.63	41.79	-0.16
	0.5	39.87	40.18	-0.31
	0.2	38.74	38.56	0.18
Girl	2.5	42.58	40.57	2.01
	2	40.10	37.79	2.31
	1.6	38.68	36.61	2.07
	1	34.72	34.68	0.04
	0.5	33.13	32.23	0.9
	0.2	31.63	30.82	0.81
Barbara	2.5	42.78	40.30	2.48
	2	41.15	38.72	2.43
	1.6	39.41	37.59	1.82
	1	35.93	34.21	1.72
	0.5	30.73	28.95	1.78
	0.2	26.23	25.28	0.95

Table 2: PSNR(dB) for compressed image tests obtained with the joint scheme.

The payload is quite large in our experiments. We can embed a message having a payload higher than 11,000 bits with a PSNR greater than 40 dB at 2 bpp and 2.5 bpp. As seen in Fig. 5, at higher bitrates, more bits were hidden. Over 13,040 bits were hidden in Lena with a PSNR of 45.17 dB at 2.5 bpp. At 0.2 bpp, only 2735 bits were hidden. The hiding payload is also dependent on the content of the original image. In Gold, hiding payload was 18,980 bits at bitrate 1.6 bpp compared to 11,089 bits in Girl for the same bitrate, and 1848 bits at bitrate 0.2 bpp compared to 1659 bits in Barbara. Table 3 gives the number of iterations needed to achieve data hiding for the test images at the specified bitrates. As expected, the value of this number is increasing gradually as the bitrate decreases. This is due to the bitstream truncation during the rate allocation stage, forcing the algorithm to do another loop after changing the selection criteria. Adaptive adjustment of the α values for each image may reduce the number of iterations of the feedback process.

Bitrate (bpp)	Image test	Number of iterations	Image test	Number of iterations
2	Lena	1	Gold	1
1.6		2		2
1		3		4
0.5		5		8
0.2		7		15
2	Girl	1	Barbara	1
1.6		2		2
1		4		4
0.5		6		7
0.2		8		10

Table 3: Number of iterations of the feedback process.

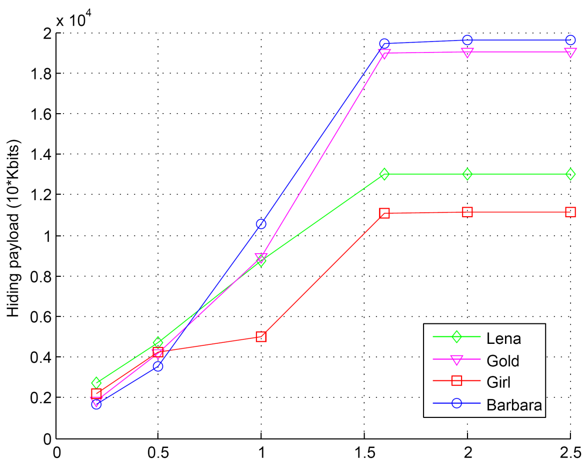

Figure 5: Hiding payload vs bitrate.

We compare the results of the proposed joint scheme with those given in [3]. Table 4 presents experimental results for the same images at low bitrates. This is not an easy task to compare the two approaches since there are quite different. The obtained hiding payloads are not the same. However, the results in terms of PNSR prove that our joint scheme provide a similar and sometimes a better quality image than [3], especially at low bitrates. For Lena image at 0.10 bpp, the PSNR obtained by our approach (33.26 dB) is better than [3] (30.74 dB) with a higher payload (1.270 bits compared to 600 bits for [3]). In Gold, the PSNR obtained by our proposition is better than 8 dB compared to [3] at 0.17 bpp with payloads close to each other.

6. CONCLUSION

In this paper we have developped an information hiding scheme under the framework of JPEG2000 part 2 that uses characteristics of the Trellis Coded quantization (TCQ). Data is hidden in the TCQ indices. The path bit discards caused by rate allocation is avoided by carefully selecting coefficients included in the Data Hiding process and by moving the LSBs of the TCQ indices to a higher bit-plane position. The proposed joint TCQ-based scheme can realize a high embedding rate and gives good visual quality performances. Therefore, it can be used in content description and management applications. Future work will involve the possible extension of the joint scheme to robust watermarking. The watermark should be resistant against the JPEG2000 scalability functionnality.

Image test	bitrate (bpp)	Our proposition		Chen *et al.*	
		Payload (bits)	PSNR (dB)	Payload (bits)	PSNR (dB)
Lena	0.10	1,270	33.26	600	30.74
	0.12	1,270	33.84	2,488	-
	0.14	1,270	34.82	3,808	-
	0.16	2,218	36.21	5,568	-
	0.20	2,735	36.00	6,768	34.29
Gold	0.17	1,079	39.45	1,752	30.77
	0.20	2,004	38.51	3,672	-
	0.25	2,840	39.66	6,528	-
	0.35	2,840	39.72	8,008	-
	0.40	2,840	39.86	9,200	34.09
Girl	0.12	1,704	30.35	744	30.84
	0.15	2,346	29.12	4,200	-
	0.18	2,346	30.38	6,584	-
	0.23	2,830	31.98	7,824	-
	0.25	3,073	32.22	8,152	34.17
Barbara	0.35	3,079	28.91	2,264	30.76
	0.40	3,540	28.40	4,888	-
	0.45	4,934	28.04	6,592	-
	0.55	4,934	30.10	8,200	-
	0.60	4,934	30.32	10,480	34.01

Table 4: Comparison of the results obtained by our joint scheme with those of the article referenced in [3].

REFERENCES

[1] D. S. Taubman and M. W. Marcellin. JPEG2000 image compression fundamentals, standards and practice. Kluwer Academic Publishers, 2002.

[2] I. J. Cox, M. L. Miller, J. A. Bloom, J. Fridrich, and T. Kalker. Digital watermarking and steganography. Morgan Kaufmann Publishers, 2008.

[3] J. Chen, T. S. Chen, C. N. Lin, and C. Y. Cheng. A bitrate controlled data hiding scheme for JPEG2000. *International Journal of Computers and Applications*, 32(2):238–241, 2010.

[4] P. C. Su and C. J. Kuo. Steganography in JPEG2000 compressed images. *IEEE Transaction on Consumer Electronics*, 49(4):824–832, 2003.

[5] N. Thomos, N. V. Boulgouris, E. Kokkinou, and M. G. Strintzis. Efficient data hiding in JPEG2000 images using sequential decoding of convolutional codes. In *International Conference in Digital Signal Processing 2002*, volume 2, pages 710–720, 2002.

[6] ISO/IEC JTCI/SC29 WG1. JPEG2000 part 2 final committee draft version 1.0, 2000.

[7] M. W. Marcellin and T. R. Fischer. Trellis coded quantization of memoryless and gauss-markov sources. *IEEE Transaction on communication*, 38:82–93, 1990.

[8] G. Ungerboeck. Channel coding with multilevel/phase signals. *IEEE Transaction on Information Theory*, 28:55–67, 1982.

[9] G. D. Forney-Jr. The viterbi algorithm. *IEEE Transaction on Information Theory*, 61:268–278, 1973.

[10] B. Chen and G. Wornell. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. *IEEE Transaction on Information Theory*, 47:1423–1443, 2001.

[11] M. L. Miller, G. J. Doerr, and I. J. Cox. Applying informed coding and informed embedding to design a robust, high capacity watermark. *IEEE Transactions on Image Processing*, 13(6):792–807, 2004.

[12] The openjpeg library is available for download at. `http://www.openjpeg.org`.