



**HAL**  
open science

# Quantization Watermarking for Joint Compression and Data Hiding Schemes

Dalila Goudia, Marc Chaumont, William Puech, Naima Hadj Said

► **To cite this version:**

Dalila Goudia, Marc Chaumont, William Puech, Naima Hadj Said. Quantization Watermarking for Joint Compression and Data Hiding Schemes. Dr. Mithun Das Gupta. Watermarking - Volume 1, InTech, Chapter 1, 2012, 978-953-51-0618-0. 10.5772/2342 . lirmm-00839002

**HAL Id: lirmm-00839002**

**<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00839002v1>**

Submitted on 26 Jun 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Quantization Watermarking for Joint Compression and Data Hiding Schemes

D. Goudia<sup>1</sup>, M. Chaumont<sup>2</sup>, W. Puech<sup>2</sup> and N. Hadj Said<sup>3</sup>

<sup>1</sup>*University of Montpellier II, University of Science and Technologies of Oran (USTO)*

<sup>2</sup>*University of Nîmes, University of Montpellier II, Laboratory LIRMM, UMR CNRS  
5506, 161, rue Ada, 34095 Montpellier cedex*

<sup>3</sup>*University of Science and Technologies of Oran (USTO),  
BP 1505 El Mnaouer, Oran*

<sup>1,2</sup>*France*  
<sup>1,3</sup>*Algeria*

## 1. Introduction

Enrichment and protection of JPEG2000 images is an important issue. Data hiding techniques are a good solution to solve these problems. In this context, we can consider the joint approach to introduce data hiding technique into JPEG2000 coding pipeline. Data hiding consists of imperceptibly altering multimedia content, to convey some information. This process is done in such a way that the hidden data is not perceptible to an observer. Digital watermarking is one type of data hiding. In addition to the imperceptibility and payload constraints, the watermark should be robust against a variety of manipulations or attacks.

We focus on trellis coded quantization (TCQ) data hiding techniques and propose two JPEG2000 compression and data hiding schemes. The properties of TCQ quantization, defined in JPEG2000 part 2, are used to perform quantization and information embedding during the same time. The first scheme is designed for content description and management applications with the objective of achieving high payloads. The compression rate/imperceptibility/payload trade off is our main concern. The second joint scheme has been developed for robust watermarking and can have consequently many applications. We achieve the better imperceptibility/robustness trade off in the context of JPEG2000 compression. We provide some experimental results on the implementation of these two schemes.

This chapter will begins with a short review on the quantization based watermarking methods in Section 2. Then, the TCQ quantization is introduced along with its application in data hiding and watermarking in Section 3. Next, we present the joint compression and data hiding approach in Section 4. Afterward, we introduce the JPEG2000 standard and the state of the art of joint JPEG2000 coding and data hiding solutions in Section 5.1. We present the proposed joint JPEG2000 and data hiding schemes in Section 6. Finally, Section 7 concludes this chapter.

## 2. Quantization watermarking

Quantization watermarking techniques are widely used in data hiding applications because they provide both robustness to the AWGN<sup>1</sup> channel and high capacity capabilities while preserving the fidelity of the host document. Quantization watermarking is a part of watermarking with side information techniques. The watermarking problem is considered as a communication problem and can be modeled as a communications system with side information. In this kind of communication system, the transmitter has additional knowledge (or side information) about the channel. Quantization techniques are based on informed coding inspired from the work of Costa (1983) in information theory. Costa's result suggests that the channel capacity of a watermarking system should be independent of the cover Work. In informed coding, there is a one-to-many mapping between a message and its associated codewords. The code or pattern that is used to represent the message is dependent on the cover Work. The reader is directed to Cox et al. (2008) for a detailed discussion of these concepts.

Chen & Wornell (2001) are the first to introduce a practical implementation of Costa's scheme, called Quantization Index Modulation (QIM). The QIM schemes, also referred as lattices codes, have received most attention due to their ease of implementation and their low computational cost. Watermark embedding is obtained by quantizing the host feature sequence with a quantizer chosen among a set of quantizers each associated to a different message. In the most popular implementation of QIM, known as dither modulation or DM-QIM (Chen & Wornell (2001)), as well as in its distortion-compensated version (DC-DM), the quantization codebook consists of a certain lattice which is randomized by means of a dither signal. This signal introduces a secret shift in the embedding lattice. Although the QIM schemes are optimal from an information theoretic capacity-maximization point of view, their robustness may be too restricted for widespread practical usage. They are usually criticized for being highly sensitive to valumetric scaling. Significant progress has been made these last past years toward resolving this issue, leading to the design of improved QIM schemes, such as RDM (Pérez-González et al. (2005)) and P-QIM (Li & Cox (2007)). Scalar Costa scheme (SCS), proposed by Eggers et al. (2003), is also a suboptimal implementation of the Costa's scheme using scalar embedding and reception functions.

Another important watermarking with side information class of methods are dirty paper trellis codes (DPTC), proposed by Miller et al. (2004). These codes have the advantage of being invariant to valumetric scaling of the cover Work. However, the original DPTC scheme requires a computational expensive iterative procedure during the informed embedding stage. Some works have been proposed to reduce the computational complexity of this scheme (Chaumont (2010); Lin et al. (2005)).

## 3. TCQ and its use for data hiding

### 3.1 Generalities on TCQ

Trellis coded quantization (TCQ) is one of the quantization options provided within the JPEG2000 standard. It is a low complexity method for achieving rate-distortion performance greater to that of scalar quantization. TCQ was developed by Marcellin & Fischer (1990) and borrowed ideas from trellis coded modulation (TCM) which have been proposed by Ungerboeck (1982). It is based on the idea of an expanded signal set and it uses coded

---

<sup>1</sup> Additive White Gaussian Noise.

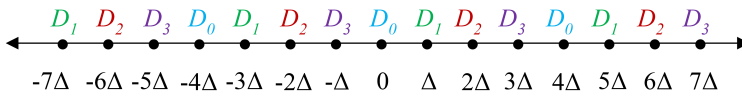


Fig. 1. Scalar codebook with subset partitioning.  $D_0, D_1, D_2$  and  $D_3$  are the subsets,  $\Delta$  is the step size and  $\dots, -2\Delta, -\Delta, 0, \Delta, 2\Delta, \dots,$  are the TCQ indices.

modulation for set partitioning. For an encoding rate of  $R$  bits/sample, TCQ takes an output alphabet  $A$  (scalar codebook) of size  $2^{R+1}$  and partitions it into 4 subsets called  $D_0, D_1, D_2$  and  $D_3$ , each of size  $2^{R-1}$ . The partitioning is done starting with the left-most codeword and proceeding to the right, labeling consecutive codewords  $D_0, D_1, D_2, D_3, D_0, D_1, D_2, D_3, \dots,$  until the right-most codeword is reached, as illustrated in Fig. 1. Subsets obtained in this fashion are then associated with branches of a trellis having only two branches leaving each state. Given an initial state, the path can be specified by a binary sequence, since there are only two possible transitions from one state to another. Fig. 2 shows a single stage of a typical 8-state trellis with branch labeling.

In order to quantize an input sequence with TCQ, the Viterbi algorithm (Forney Jr (1973)) is used to choose the trellis path that minimizes the mean-squared error (MSE) between the input sequence and output codewords. The sequence of codewords can be specified by a sequence of  $R$  bit indices. Each  $R$  bit index consists of a single bit specifying the chosen subset (trellis path) and  $R-1$  bits specifying an index to a codeword within this subset (index value). The dequantization of TCQ indices at the decoder is performed as follows. Given the initial state, 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 transition or stage.

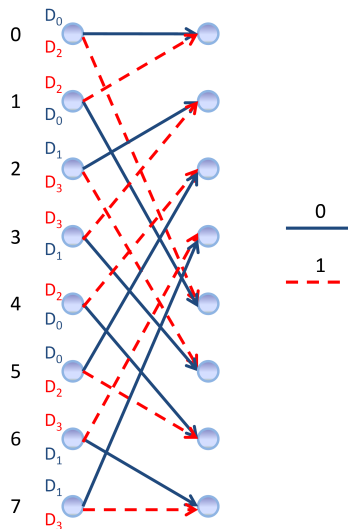


Fig. 2. A single stage of an 8-state trellis with branch labeling.

### 3.2 TCQ in data hiding and watermarking

TCQ was first used in data hiding to build practical codes based on quantization methods. Exploiting the duality between information embedding and channel coding with side information, Chou et al. (1999) proposed a combination of trellis coded modulation (TCM) and TCQ. This method is referred as TCM-TCQ and consists of partitioning a TCM codebook into TCQ subsets to approach the theory bound. Another data hiding technique based on the TCM-TCQ scheme has been proposed by Esen & Alatan (2004) and Wang & Zhang (2007). This method is called the TCQ path selection (TCQ-PS). In this algorithm, similarly to Miller et al. (2004), the paths in the trellis are forced by the values of the message and the samples of the host signal are quantized with the subset corresponding to the trellis path. Esen & Alatan (2004) also explore the redundancy in initial state selection during TCQ to hide information and compare the results with QIM (Chen & Wornell (2001)) and TCQ-PS. Wang & Zhang (2007) show that the trellis used in TCQ can be designed to achieve more robustness by changing the state transition rule and quantizer selection rule. Le-Guelvouit (2005) explores the use of TCQ techniques based on turbo codes to design a more efficient public-key steganographic scheme in the presence of a passive warden.

Watermarking techniques based on the TCQ-PS method appeared recently in the literature. Braci et al. (2009) focused on the security aspects of informed watermarking schemes based on QIM and proposed a secure version of the TCQ-PS adapted to the watermarking scenario. The main idea is to cipher the path at the encoder side by shifting randomly each obtained codeword to a new one taking from another subset. Then, according to the secret key, a codebook different from the one used for the transmitted message is chosen. Le-Guelvouit (2009) has developed a TCQ-based watermarking algorithm, called TTCQ, which relies on the use of turbo codes in the JPEG domain. Ouled-Zaid et al. (2007) have adapted the TTCQ algorithm to the wavelet domain and have studied its robustness to lossy compression attacks.

### 4. Joint compression and data hiding approach

Data hidden images are usually compressed in a specific image format before transmission or storage. However, the compression operation could remove some embedded data, and thus prevent the perfect recovery of the hidden message. In the watermarking context, the compression process also degrades the robustness of the watermark. To avoid this, it is better to combine image compression and information hiding to design joint solutions. The main advantage to consider jointly compression and data hiding is that the embedded message is robust to compression. The compression is no longer considered as an attack. Another important advantage is that it allows the design of low complex systems compared to the separate approach.

The joint approach consists of directly embedding the binary message during the compression process. The main constraints that must be considered are trade offs between data payload, compression bitrate, computational complexity and distortion induced by the insertion of the message. In other words, the embedding of the message must not lead to significant deterioration of the compressor's performances (compression rate, complexity and image quality). On the other hand, the data hiding process must take into account the compression impact on the embedded message. The latter should resist to quantization and entropy coding steps of a lossy compression scheme. In the watermarking scenario, we must also consider the watermark robustness against common image attacks after compression. The

watermark needs to be robust enough to allow a correct message extraction after some acceptable manipulations of the decompressed/watermarked image.

The data hiding technique must be adapted and integrated into the compressor's coding framework. One or several modules can be used to compress and hide data. Three strategies are commonly used as shown in fig. 3 for a lossy wavelet-based coder :

- data is hidden just after the wavelet transform step: embedding is performed on the wavelet coefficients,
- data is hidden just after the quantization stage: embedding is performed on the quantized wavelet coefficients (quantization indices),
- data is hidden during the entropy coding stage: embedding is performed directly on the compressed bitstream.

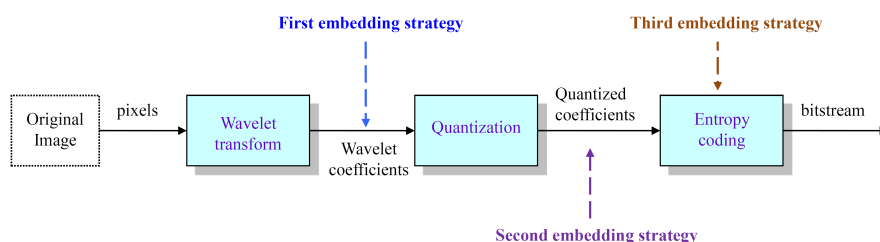


Fig. 3. Data hiding embedding strategies into a lossy wavelet-based coder.

The extraction of the hidden message can be done in two different ways. The first one consists to extract the message from the coded bitstream during the decompression stage. The second one consists to retrieve the hidden message from the data hidden or watermarked image. In this case, the extraction stage is performed after decompression and the knowledge of the compression parameters used during joint compression and data hiding is necessary. For example, if the coder used is a wavelet-based coding system, we need to know the type of wavelet transform used, the number of resolution levels and selected sub-bands.

## 5. JPEG2000 standard and data hiding in the JPEG2000 domain

### 5.1 JPEG2000 standard

The international standard JPEG2000 (Taubman & Marcellin (2001)) has been developed by the Joint Photographic Experts Group (JPEG) to address different aspects related with image compression, transmission and manipulation. JPEG2000 is a wavelet-based codec, which supports different types of still images and provides tools for a wide variety of applications, such as Internet, digital cinema and real-time transmission through wireless channels. JPEG2000 provides many features. Some of them are: progressive transmission by quality or resolution, lossy and lossless compression, region of interest (ROI) and random access to bitstream.

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). The image is partitioned into rectangular non-overlapping segments called tiles. Then, each tile is transformed by the discrete wavelet transform (DWT) into a collection of sub-bands:

LL (horizontal and vertical low frequency), HL (horizontal high frequency and vertical low frequency), LH (horizontal low frequency and vertical high frequency) and HH (horizontal high frequency and vertical high frequency) sub-bands which may be organized into increasing resolution levels. The wavelet coefficients are afterwards quantized by a dead-zone uniform scalar quantizer. The quantized coefficients in each sub-band are partitioned into small rectangular blocks which are called *code-blocks*. Next, the EBCOT<sup>2</sup> algorithm encodes each *code-block* independently during the Tier 1 encoding stage and generates the embedded bitstreams. An efficient rate-distortion algorithm called Post Compression Rate-Distortion Optimization (PCRD) provides effective truncation points of the bitstreams in an optimal way to minimize distortion according to any given target bitrate. The bitstreams of each *code-block* are truncated according to the chosen truncation points. Finally, the Tier 2 encoder output the coded data in packets and defines a flexible codestream organization supporting quality layers.

## 5.2 Data hiding in the JPEG2000 domain

Several data hiding techniques integrated into the JPEG2000 coding scheme have been proposed Chen et al. (2010); Fan & Tsao (2007); Fan et al. (2008); Meerwald (2001); Ouled-Zaid et al. (2009); Schlauweg et al. (2006); Su & Kuo (2003); Thomos et al. (2002). Some of these schemes Chen et al. (2010); Fan & Tsao (2007); Fan et al. (2008) take into account the bitstream truncation of the JPEG2000 bitstream during the rate control stage.

Chen et al. (2010) 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. Ouled-Zaid et al. (2009) proposed to integrate a QIM-based watermarking method in JPEG2000 part 2. This variant of QIM consists of reducing the distortion caused during quantization-based watermarking by using a non-linear scaling. The watermark is embedded in the LL sub-band of the wavelet decomposition before the JPEG2000 quantization stage. Fan et al. (2008) proposed region of interest (ROI)-based watermarking scheme. The embedded watermark can survive ROI processing, progressive transmission and rate-distortion optimization. The only drawback of this method is that it works only when the ROI coding functionality of JPEG2000 is activated. Fan & Tsao (2007) proposed hiding two kinds of watermarks, a fragile one and a robust one by using a dual pyramid watermarking scheme. The robust pyramid watermark is designed to conceal secret information inside the image so as to attest to the origin of the host image. The fragile pyramid watermark is designed to detect any modification of the host image. Schlauweg et al. (2006) have developed a semi-fragile authentication watermarking scheme by using an extended scalar quantization and hashing scheme in the JPEG2000 coding pipeline. This authentication scheme is secure but the embedding of the watermark induces poor quality performances. Su & Kuo (2003) proposed to hide data 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 data hiding procedure is operated in the special JPEG2000 lazy mode which requires a target bitrate higher than 2 bpp. Thomos et al. (2002) presented a sequential decoding of convolutional codes for data hiding in JPEG2000 images. Meerwald (2001) developed a watermarking process based on QIM integrated to JPEG2000 coding chain.

<sup>2</sup> Embedded Block Coding with Optimized Truncation.

Despite its robustness, this method does not fulfill the visual quality requirement. It should be noted that all these schemes integrate an additional embedding/extraction stage in the JPEG2000 compression/decompression process.

## 6. TCQ based data hiding and JPEG2000 coding schemes

We investigate the design of compression and data hiding schemes in the JPEG2000 domain. The main objective is to develop quantization-based data hiding methods to simultaneously quantize and hide data during JPEG2000 compression. Several quantization options are provided within JPEG2000 Part 2 (ISO/IEC JTC1/SC29 WG1 (2000)) such as TCQ. We propose quantization data hiding strategies based on TCQ to quantize and hide data at the same time by using a single component. This TCQ-based quantization module replaces the TCQ component used in JPEG2000 part 2. 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 proposed schemes can be viewed as "data hiding or watermarking within JPEG2000 Coding".

### 6.1 TCQ data hiding scheme in the JPEG2000 part 2 coding framework

The first joint scheme investigates the use of TCQ quantization to embed the maximum amount of data in the host image during JPEG2000 compression while minimizing perceptual degradations of the reconstructed image (Goudia et al. (2011b)). The hidden data is extracted during JPEG2000 decompression.

#### 6.1.1 The TCQ-based data hiding strategy

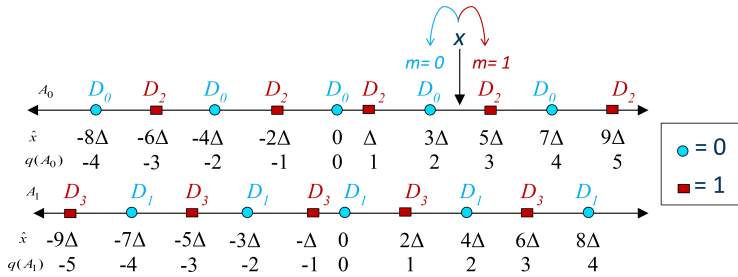


Fig. 4. The QIM principles applied to JPEG2000 part 2 union quantizers.

Our data hiding strategy is derived from the QIM (Chen et al. (2010)) principles and is integrated into a TCQ approach. It is a variant of the TCQ-PS method (Section 3.2). In the TCQ quantization specified in JPEG2000 part 2 recommendations, the two scalar quantizers associated with each state in the trellis are combined into *union quantizers*  $A_0 = D_0 \cup D_2$  and  $A_1 = D_1 \cup D_3$ . The trellis is traversed by following one of the two branches that emanate from each state. The straight branch is labeled by  $D_0$  or  $D_1$  and the dotted branch with  $D_2$  or  $D_3$  as shown in Fig. 2. We propose the following principle as illustrated fig. 4:

- 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 dotted branches when we embed a 0-bit, and suppressing straight 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 5.1. 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  $\mathbf{m}$  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 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. The process of quantization produces the sequence of TCQ quantization indices  $\mathbf{q}$  given by:

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

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  $\Delta$ ,  $\mathbf{q}[i]$  can be computed as:  $\mathbf{q}[i] = \text{sign}(\mathbf{x}[i]) \left\lfloor \frac{|\mathbf{x}[i]|}{\Delta} \right\rfloor$ . We are able to extract the embedded message during the inverse TCQ quantization stage of JPEG2000 decompression by retrieving the path bits at the transitions which correspond to the *selected coefficients*. For each *code-block*, the decoder produces an estimate of  $\mathbf{x}$  as follows:

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

where  $\bar{Q}^{-1}$  is the dequantization function. For a given step size  $\Delta$ , the reconstructed value  $\hat{\mathbf{x}}$  can be computed as:  $\hat{\mathbf{x}}[i] = \text{sign}(\mathbf{q}[i])(|\mathbf{q}[i]| + \delta)\Delta$ , where  $\delta$  is a user selectable parameter within the range  $0 < \delta < 1$  (typically  $\delta = 0.5$ ).

### 6.1.2 The proposed joint JPEG2000 and data hiding scheme

The block diagram of the joint JPEG2000 encoder and data hiding scheme is shown in Fig. 5. 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 the selected resolution levels. The selection criteria that allows us to perform the selection will be discussed Section 6.1.3. Next, the data is hidden during the TCQ quantization stage which is performed independently on each *code-block*. Afterward, EBCOT executes the entropy coding. Subsequently, rate-distortion optimization arranges the

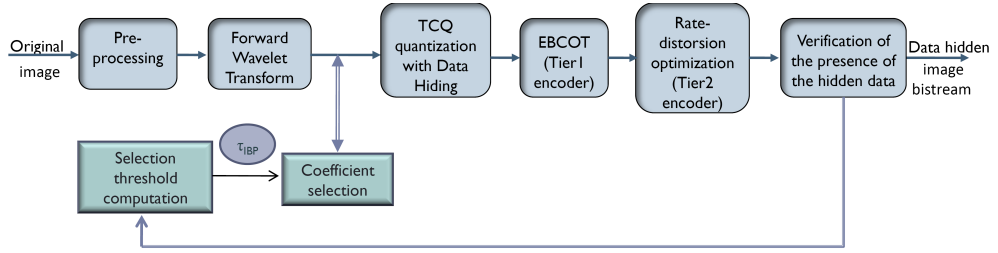


Fig. 5. The block diagram of the joint JPEG2000 codec and data hiding scheme.

*code-blocks* bitstreams into quality layers according to the target bitrate and proceeds to the formation of the JPEG2000 codestream.

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. 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 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 during JPEG2000 decompression: 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*. Next, the inverse DWT and the post-processing operations are performed to reconstruct the image.

### 6.1.3 Selection of the wavelet coefficients included in the data hiding process

Data is 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:

$$q = s, q_0 q_1 \dots q_{L-1}, \quad (3)$$

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  $\tau_{IBP}$  (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*  $\mathbf{C}$ ,  $\tau_{IBP}$  is computed as follows:  $\tau_{IBP} = \lfloor \alpha * L \rfloor$ , where  $\alpha$  is a real

factor between 0 and 1 initialized with a predefined value for each sub-band. The selection of coefficients included in the data hiding process is done as follows:

$$\text{if } \lceil \log_2(|\mathbf{q}[i]| + 1) \rceil > \tau_{IBP}, \text{ then add } \mathbf{x}[i] \text{ to } S_C, \quad (4)$$

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

To be sure that the path will not be partially lost during the rate-distortion optimization stage, especially at low bitrates, we propose to move 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 causing the loss of the MSB: indeed, if the LSB value is 0 and if it is moved at  $q_0$ , this will cause the loss of a bit plane because the MSB value will be 0.

The thresholds  $\tau_{IBP}$  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.

### 6.1.4 Experimental results

To implement our joint JPEG2000 and data hiding scheme, we choose to use the OpenJPEG library<sup>3</sup> which is a JPEG2000 part 1 open-source codec written in C language. We replaced the scalar uniform quantization component by a JPEG2000 part 2 compliant TCQ quantization module. Simulations were run on 200 grayscale images of size 512 x 512 randomly chosen from the BOWS2 database<sup>4</sup>. The JPEG2000 compression parameters are the following: the source image is partitioned into a single tile and a five levels of irreversible DWT 9-7 is performed. The size of the *code-blocks* is: 64 x 64 for the first to the third level of resolution, 32 x 32 for the fourth level and 16 x 16 for the fifth level. We set the compression ratio from 2.5 bpp to 0.2 bpp. 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-blocks* included in the data hiding process. The size of the side information file containing the 4-bit thresholds  $\tau_{IBP}$  is equal to 84 bits (21 x 4 = 84). Performance evaluation of the proposed joint scheme covers two aspects: the compression performances and the data hiding performances.

#### 6.1.4.1 Compression performances

We study the compression performances of the proposed joint scheme under various compression bitrates in terms of image quality and execution time. We seek to know if the embedding of the message leads to significant degradation of the JPEG2000 performances. We point out 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. The proposed joint scheme produces a JPEG2000 syntax compliant bitstream.

<sup>3</sup> The openjpeg library is available for download at <http://www.openjpeg.org>

<sup>4</sup> The BOWS2 database is located at <http://bows2.gipsa-lab.inpg.fr>

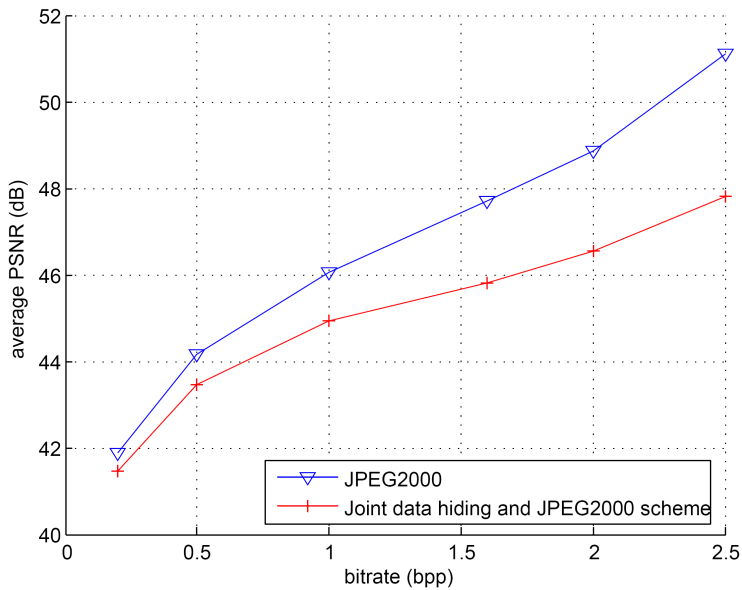


Fig. 6. Comparison between average PSNR results obtained by the proposed data hiding and JPEG2000 scheme and those obtained with JPEG2000 on 200 images of size  $512 \times 512$ .

Quality assessment was carried out using two objective evaluation criteria, PSNR<sup>5</sup> and SSIM<sup>6</sup>. For each bitrate, we compute the PSNR (respectively SSIM) of every image of the database. Next, the average PSNR (average SSIM) of all the tested images is computed. We compare respectively between the average PSNR (and the SSIM) computed for the JPEG2000 compressed images and those computed for the compressed and data hidden images. The fig. 6 and 7 show respectively the average PSNR curves and average SSIM curves obtained for the two coders. The average PSNR of the joint scheme is greater than 40 dB for all compression bitrates as shown in fig. 6. The quality degradation resulting from data hiding is relatively small when we compare between the joint scheme and JPEG2000 curves. At 2.5 dB, the difference between the two PSNR values is approximatively of 3 dB. When the bitrate decreases, this difference decreases to reach 0.4 dB at 0.2 bpp. When considering the SSIM results shown in fig. 7, we notice that the average SSIM provided by the joint scheme remains above 90% until 0.5 bpp. The difference between these values and those provided by JPEG2000 is relatively the same for all the tested bitrates (approximatively 1.6 %). Given the results, we can say that the proposed joint data hiding and JPEG2000 scheme exhibits relatively good quality performances in terms of PSNR and SSIM. An example of data hidden and compressed image at 1.6 bpp is presented in fig. 8 for the well known image test Lena.

The computational complexity of the proposed joint scheme is investigated. We first consider the encoding execution time. The joint scheme uses an iterative embedding algorithm during

<sup>5</sup> Pick Signal to Noise Ratio

<sup>6</sup> Structural SIMilarity. SSIM is a perceptual measure exploiting Human Visual System (HVS) properties. The SSIM values are real positive numbers in the range 0 to 1. Stronger is the degradation and lower is the SSIM measure.

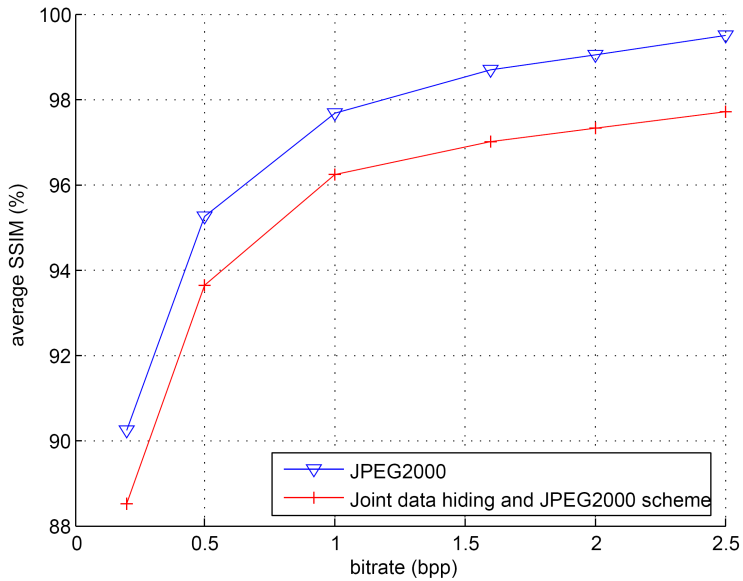


Fig. 7. Comparison between average SSIM results obtained by the proposed data hiding and JPEG2000 scheme and those obtained with JPEG2000 on 200 images of size 512 x 512.



JPEG2000



Joint data hiding and JPEG2000 scheme

Fig. 8. Comparison between Lena image data hidden and compressed with our joint scheme and the same image compressed with JPEG2000 at 1.6 bpp.

the compression stage. TCQ quantization, Tier 1 encoding and Tier 2 encoding steps are repeated until the message can be correctly extracted. The number of iterations depends on the target bitrate, the selection criteria and the content of the processed image. Execution time increases as the number of iterations increases. Table 1 gives the number of iterations and encoding execution times needed to achieve data hiding for the three test images Lena, Clown and Peppers. These execution times have been obtained on Intel dual core 2 GHZ processor with 3 GB of RAM. When the bitrate decreases, the number of iterations, and therefore, the execution time increases. The execution times obtained by the joint scheme are higher than those obtained with JPEG2000. The JPEG2000 average encoding execution time is 1.90 sec. for an image of size 512 x 512. JPEG2000 is faster than the proposed joint scheme during the compression stage. When considering the decoding execution time, we note that the two coders provide similar decoding times. The average decoding time is approximatively 0.55 sec for an image of size 512 x 512.

bitrate (bpp)	Test image	Number of iterations	Encoding execution time (sec.)
2.5	Lena	1	3.68
2		1	3.42
1.6		1	3.31
1		1	3.13
0.5		2	5.83
0.2		2	5.56
2.5	Clown	1	3.66
2		1	3.46
1.6		1	3.40
1		1	3.19
0.5		2	5.94
0.2		5	13.77
2.5	Peppers	1	3.57
2		1	3.43
1.6		1	3.34
1		1	3.13
0.5		2	6.06
0.2		2	5.78

Table 1. Encoding execution time and number of iterations of the iterative embedding algorithm.

#### 6.1.4.2 Data hiding performances

Bitrate (bpp)	2.5	2	1.6	1	0.5	0.2
Average payload	11254	11203	11172	7384	4213	1573
Minimum payload	1266	1266	1266	1266	926	422
Maximum Payload	36718	26180	21903	13470	7530	2621

Table 2. Payloads obtained with the proposed joint scheme on 200 grayscale images of size 512 x 512.

We have noticed that the hidden message is imperceptible as seen in Section 6.1.4.1. We study now the data hiding performances of the proposed joint scheme in terms of data payload.

For each tested bitrate, the average, minimum and maximum payloads are computed. The results are summarized in Table 2. We note that high average payloads are achieved at high bitrates. We can embed a message having a payload higher than 11000 bits until 1.6 bpp. The maximum payload is 36718 bits at 2.5 bpp and falls below 27000 bits for the remaining bitrates. The minimum payload is 1266 bits until 1 bpp and decreases up to 422 bits at 0.2 bpp. The large difference between the minimum and maximum payloads is due to the fact that the number of selected coefficients depends mainly on the content of the original image. Textured and complex shaped images give a great number of wavelet coefficients which are large and sparse. On the contrary, simple images with low contrast and poor textures give a limited number of significant wavelet coefficients. The hiding payload is also dependent on the compression bitrate. We note that from 1 bpp, we obtain lower payloads than those obtained at high bitrates. This is due to the bitstream truncation during the JPEG2000 rate allocation stage. The payload decreases as the bitrate decreases.

## 6.2 A joint TCQ watermarking and JPEG2000 scheme

The second joint scheme was designed to perform simultaneously watermarking and JPEG2000 coding (Goudia et al. (2011a)). We use a different TCQ-based watermark embedding method from the one used in the first joint scheme to embed the watermark. The watermark extraction is performed after JPEG2000 decompression.

### 6.2.1 The TCQ-based watermarking strategy

The watermarking strategy is based on the principles of the DM-QIM (Chen et al. (2010)) approach associated with a trellis. We replace the uniform scalar quantizers used in JPEG2000 part 2 by shifted scalar quantizers with the same step size  $\Delta$  as for the original ones. We can also use a higher step size by multiplying the original step size by a constant. These quantizers differ from the previous quantizers by the introduction of a shift  $d$  which is randomly obtained with a uniform distribution over  $[-\Delta/2, \Delta/2]$ <sup>7</sup>. We propose the following principle: if the bit to embed is the bit 0 then the quantizer  $D_j^0, j = 0, 1, 2, 3$  with the shift  $d_0$  is used. If it is the bit 1 then we employ the quantizer  $D_j^1$  with the shift  $d_1$  satisfying the condition:  $|d_0 - d_1| = \Delta/2$ . For each transition  $i$  in the trellis, two shifts  $\mathbf{d}_0[i]$  and  $\mathbf{d}_1[i]$  and four union quantizers  $A_{0,i}^0 = D_{0,i}^0 \cup D_{2,i}^0, A_{1,i}^0 = D_{1,i}^0 \cup D_{3,i}^0, A_{0,i}^1 = D_{0,i}^1 \cup D_{2,i}^1, A_{1,i}^1 = D_{1,i}^1 \cup D_{3,i}^1$  are constructed. Thus, we will have two groups of union quantizers for the trellis structure used in our approach: the group 0, which consists of all shifted union quantizers corresponding to the watermark embedded bit 0 and the group 1, which incorporates shifted union quantizers corresponding to the embedded bit 1. The trellis structure used in the proposed method has four branches leaving each state (Fig. 9.a). For each state of the trellis, two union quantizers instead of one are associated with branches exiting this state.

The watermark embedding process is split into two steps to perform watermarking within JPEG2000. The first step is achieved during the quantization stage of the JPEG2000 compression process. Let us consider a binary message  $\mathbf{m}$  to embed and a host signal  $\mathbf{x}$ . The quantization stage produces the sequence of TCQ quantization indices  $\mathbf{q}$ . For each transition  $i$  in the trellis, the union quantizers are selected according to the value  $\mathbf{m}[i]$ . The trellis is thus

<sup>7</sup> Schuchman (1994) showed that the subtractive dithered quantization error does not depend on the quantizer input when the dither signal  $\mathbf{d}$  has a uniform distribution within the range of one quantization bin ( $d \in [-\Delta/2, \Delta/2]$ ) leading to an expected squared error of  $E^2 = \Delta^2/12$ .

modified in order to remove all the branches that are not labeled with the union quantizers that encode the message as illustrated in Fig. 9.b. The subsets  $D_{j,i}^{m[i]}$ ,  $j = 0, 1, 2, 3$  are associated to the branches of the modified trellis. The quantization index  $\mathbf{q}[i]$  is given by:

$$\mathbf{q}[i] = Q_{D_{j,i}^{m[i]}}(\mathbf{x}[i]), \quad (5)$$

where  $Q$  is the quantization function of JPEG2000,  $\mathbf{m}[i]$  is the bit to embed at transition  $i$  and  $D_{j,i}^{m[i]}$  is the shifted quantizer. For a given step size  $\Delta$ ,  $\mathbf{q}[i]$  can be computed as:

$$\mathbf{q}[i] = \text{sign}(\mathbf{x}[i] - \mathbf{d}_{\mathbf{m}[i]}[i]) \left\lfloor \frac{|\mathbf{x}[i] - \mathbf{d}_{\mathbf{m}[i]}[i]|}{\Delta} \right\rfloor, \quad (6)$$

where  $\mathbf{d}_{\mathbf{m}[i]}[i]$  is the shifting of the shifted quantizer  $D_{j,i}^{m[i]}$ . In addition to  $\mathbf{q}$ , the sequence  $\mathbf{l}$  is generated. It contains an extra information which ensures that the modified trellis structure is properly retrieved during the inverse quantization step.

The second step is performed during the inverse quantization stage of the JPEG2000 decompression process, yielding the watermarked signal  $\hat{\mathbf{x}}$ . The inverse quantization stage utilizes the same trellis employed in the quantization step. The reconstructed values  $\hat{\mathbf{x}}$  are produced as:

$$\hat{\mathbf{x}}[i] = \bar{Q}_{D_{j,i}^{m[i]}}^{-1}(\mathbf{q}[i]), \quad (7)$$

where  $\bar{Q}^{-1}$  is the inverse quantization function of JPEG2000. For a given step size  $\Delta$ , the reconstructed value  $\hat{\mathbf{x}}$  can be computed as:

$$\hat{\mathbf{x}}[i] = \text{sign}(\mathbf{q}[i])(|\mathbf{q}[i]| + \delta)\Delta + \mathbf{d}_{\mathbf{m}[i]}[i], \quad (8)$$

where  $\delta$  is a user selectable parameter within the range  $0 < \delta < 1$ .

### 6.2.1.1 Watermark embedding

The watermark embedding process is performed independently into each *code-block*.

#### Quantization

For each *code-block*  $\mathbf{C}$ , the quantization/watermark embedding procedures are:

- **Computation of the shiftings  $\mathbf{d}_0$  and  $\mathbf{d}_1$ :** we use a pseudo random generator initialized by the secret key  $k$  to compute the shiftings.
- **Generation of the group 0 and group 1 union quantizers:** for each transition  $i$ , we design shifted scalar quantizers. We label the branches of the trellis with these quantizers. Fig. 9.a. shows a three-stage of the trellis structure used in our joint scheme. The trellis is simplified so that all the branches through the trellis, and thus all the associated union quantizers, encode the message  $\mathbf{m}$  as illustrated in Fig. 9.b.
- **Finding the optimal path:** the initial state of the given trellis structure is set to 0. The Viterbi Algorithm (Forney Jr (1973)) is applied in order to find the minimum distortion path (Fig. 9.b). The TCQ indices are produced (equation 6).



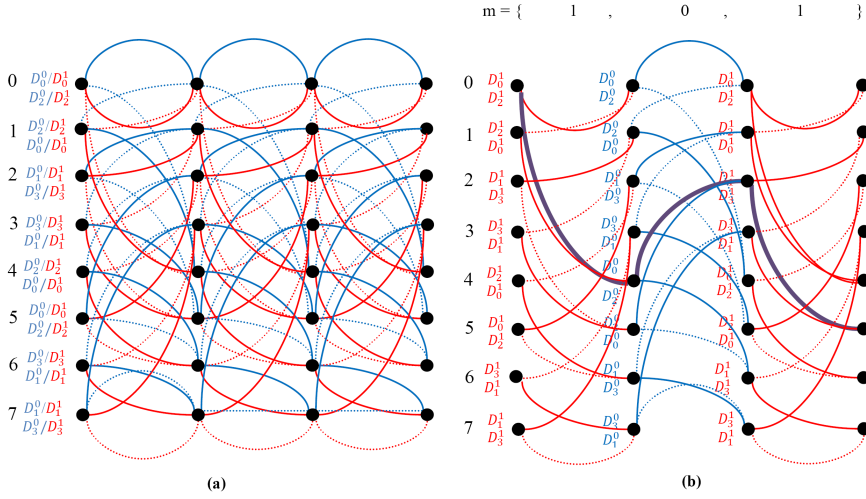


Fig. 9. a) A three-stage of the modified trellis structure, b) Insertion of the message  $m=\{1,0,1\}$ : all the branches that are not labeled with the union quantizers that encode the message are removed. The bold branches represent the optimal path calculated by the Viterbi algorithm

#### Inverse quantization

The watermark embedding is completed during the inverse quantization of the JPEG2000 decompression stage. The image bitstream is decoded by the EBCOT decoder (Tier 2 and Tier 1 decoding) to obtain the sequence of decoded TCQ indices. For each *code-block C*, the inverse quantization steps are the following:

- **Computation of the shiftings  $d_0$  and  $d_1$ .**
- **Generation of the group 0 and group 1 union quantizers.**
- **Inverse quantization:** the trellis structure with four branches leaving each state is generated. Each branch of the trellis is afterwards labeled with the shifted quantizers. The sequence  $I$  enables us to retrieve the pruned trellis used during the quantization stage. This trellis is used to reconstruct the wavelet coefficients. Given the TCQ indices, the embedding of the watermark is achieved during the computation of the reconstructed wavelet coefficients (equation 8).

#### 6.2.1.2 Watermark extraction

The watermark recovery from the decompressed/watermarked image is a blind watermarking extraction process. In order to extract the embedded message within the decompressed image, we perform the following operations:

- **Apply the DWT:** we apply the DWT on the decompressed watermarked image. Each sub-band included in the watermarking process is partitioned into blocks of same size as the JPEG2000 code-blocks. The coefficients belonging to the current block are stored in the vector  $\mathbf{y}$ . The following steps are repeated for each processed block.
- **Retrieve the shiftings  $d_0$  and  $d_1$ :** we retrieve the shiftings by using the secret key  $k$  and we set the union quantizers group 0 and group 1.

- **Perform the TCQ quantization:** the decoder applies the Viterbi algorithm to the entire trellis (Fig. 9.a). The Viterbi algorithm identifies the path that yields the minimum quantization distortion between  $y$  and the output codewords. The hidden message is then decoded by looking at the TCQ codebook labeling associated to the branches in that path.

## 6.2.2 The proposed joint watermarking and JPEG2000 scheme

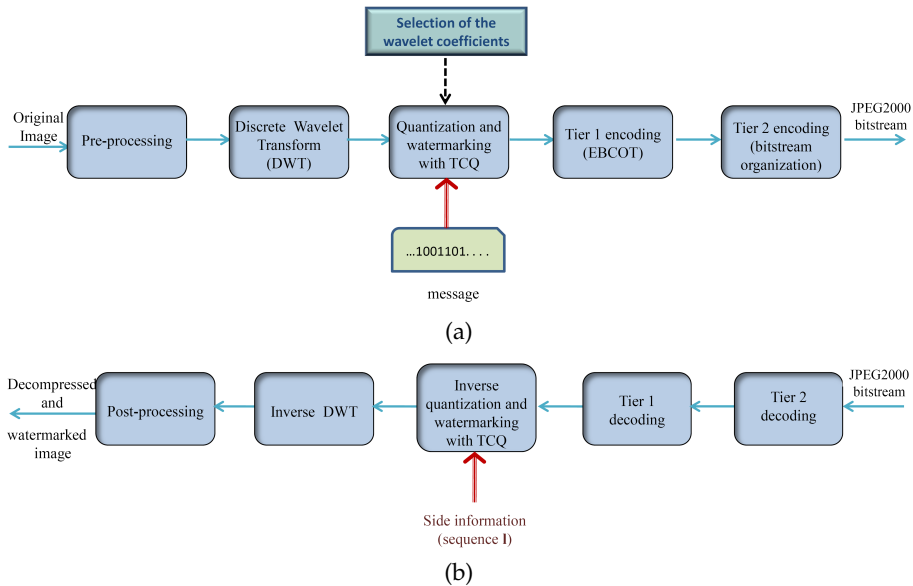


Fig. 10. The joint JPEG2000/watermarking scheme, a): compression process, b): decompression process

The block diagram of the joint JPEG2000 part 2 and watermark embedding scheme is illustrated in fig. 10. The classical TCQ quantization component of the JPEG2000 encoder and decoder is replaced by a hybrid TCQ module which can perform at the same time quantization and watermark embedding. One of the most important parameter to consider is the selection of the wavelet coefficients that must be included in the watermarking process. We chose to embed the watermark in the HL, LH and HH detail sub-bands of the selected resolution levels. All coefficients of these sub-bands are watermarked. Wavelet coefficients of the other sub-bands are quantized with the classical TCQ algorithm of JPEG2000 part 2. The watermarking payload is determined by the number of detail sub-bands included in the watermarking process. The payload increases when we add more detail sub-bands from a new selection of resolution levels. EBCOT and the rate-distortion optimisation stage are taken into account by the use of an error correcting code to add redundancy. The rate of this error correcting code must be low in order to allow reliable recovery of the watermark during the extraction stage. High watermarking payloads can be achieved by including as many detail sub-bands as necessary and by adjusting the rate of the error correcting code. Another important parameter to consider is the quantizer step size value of the selected sub-bands. The step size value of each selected sub-band should be large enough to obtain an acceptable watermarking power without affecting the quantization performances of JPEG2000.

### 6.2.3 Experimental results

The image database and the compression parameters used during the experimentations are the same as those used in the joint data hiding and JPEG2000 scheme (Section 6.1.4). The watermarking parameters are the following: binary logo of size  $32 \times 32$  is used in the experiments (Fig. 14.(a)). The message of 1024 bits length is inserted in the detail sub-bands of the second to the fourth resolution level. The joint scheme embed one bit of the (non-coded) message for every 256 pixels in an image of size  $512 \times 512$ . The message is encoded with a very simple repetition code of  $1/63$ -rate. We shuffle (scatter) pseudo randomly the bits of the coded message with a secret key.  $\Delta_{sb}/4$  is the TCQ quantizer step size value of the sub-band  $sb$  used in JPEG2000 part 2. The selection of the step size value  $\Delta_{sb,TCQ}$  for the sub-bands included in the watermarking process is done so that the best trade off between robustness and quality degradation is achieved. We select the value  $\Delta_{sb,TCQ} = \Delta_{sb}$  after experimenting different step size values. We study the compression performances of the proposed joint scheme under various compression bitrates. We also evaluate the robustness of watermarked images against four attacks.

#### 6.2.3.1 Compression performances

The compression performances and the impact of watermark embedding on the reconstructed image quality are investigated. An example of watermarked and compressed image at 1.6 bpp is presented in Fig. 11 for the well known test image Bike. We evaluate the image quality performances of the proposed joint scheme under various compression bitrates in terms of PSNR and SSIM. The obtained results are shown in Fig. 12 and 13.

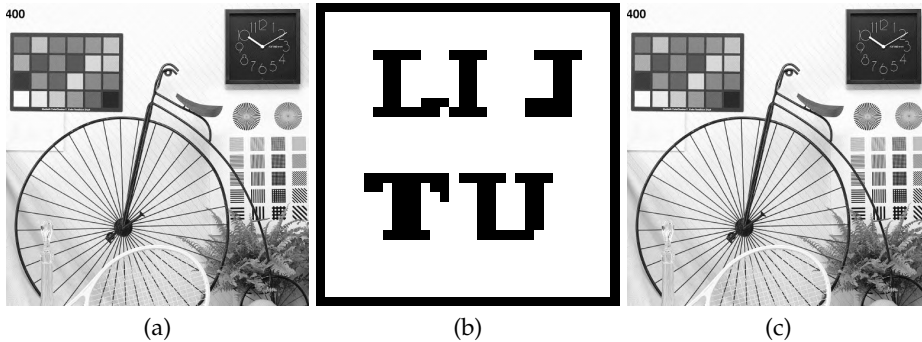


Fig. 11. Comparison between : (a) Bike image compressed with JPEG2000 at 1.6 bp, and (c) the same image watermarked and compressed with our joint scheme at 1.6 bp. The extracted watermark is : (b) binary logo of size  $32 \times 32$ .

We first consider the PSNR results (Fig. 12). The curves representing the results obtained for JPEG2000 and the proposed joint scheme are quite far from each other. This is due to the use of a large step size value for the sub-bands included in the watermarking process. The step size value used for watermark embedding is four times higher than the JPEG2000 part 2 step size. The use of a large step size value allows a higher watermarking power. We obtain a lower fidelity in comparison with that obtained using the original TCQ quantizer step size while achieving better watermark robustness. We note that the average PSNR is greater than 40 dB until 0.5 bpp. At 2.5 bpp, the difference between the two PSNR values is approximately of 4.7 dB. At 0.2 bpp, the difference is less (2.4 dB). When considering the SSIM results shown in

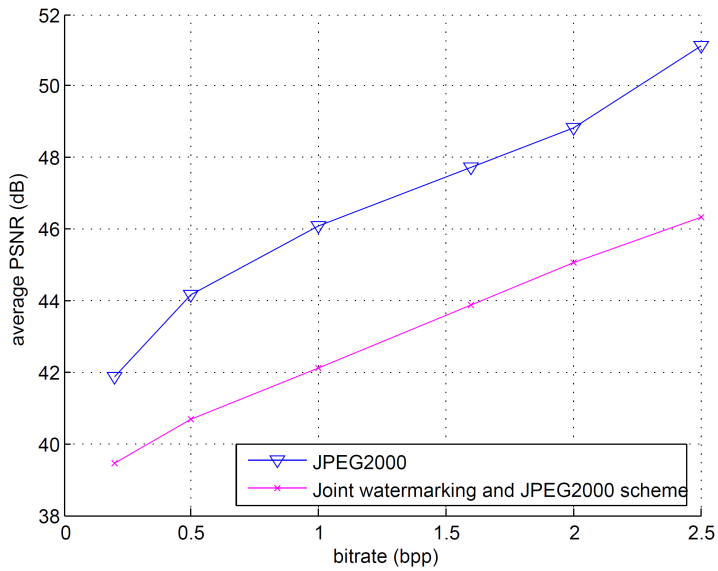


Fig. 12. Comparison between average PSNR results obtained by the proposed watermarking and JPEG2000 scheme and those obtained with JPEG2000 on 200 images of size 512 x 512.

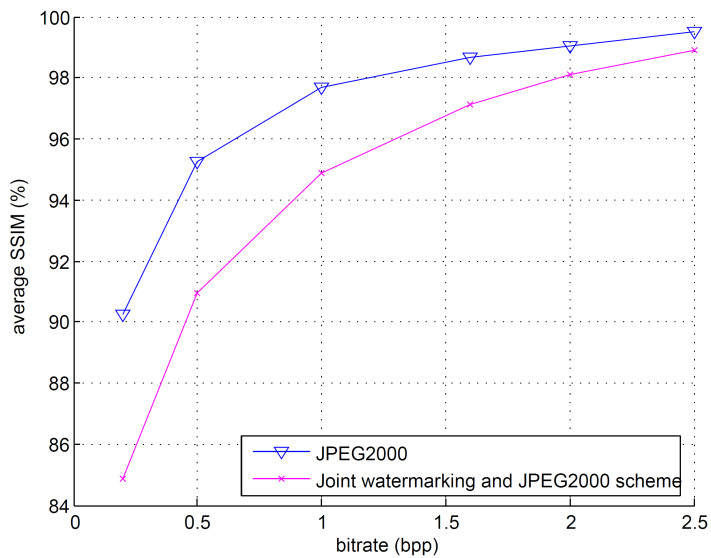


Fig. 13. Comparison between average SSIM results obtained by the proposed watermarking and JPEG2000 scheme and those obtained with JPEG2000 on 200 images of size 512 x 512.

fig. 13, we notice that the two curves are close to each other at high bitrates. The curves move away from each other from 1 bpp. The average SSIM provided by the joint scheme remains above 90% until 0.5 bpp. It drops below 86% at 0.2 bpp.

Image test	bitrate (bpp)	PSNR (dB)		SSIM	
		Joint decoder	JPEG2000 decoder	Joint decoder	JPEG2000 decoder
Lena	2.5	40.75	40.83	0.9612	0.9616
	2	39.45	39.57	0.9480	0.9484
	1.6	39.99	40.11	0.9407	0.9422
	1	38.25	38.22	0.9206	0.9221
	0.5	36.46	36.50	0.8886	0.8904
	0.2	33.58	33.59	0.8263	0.8283
Goldhill	2.5	40.93	40.94	0.9564	0.9558
	2	40.93	40.99	0.9394	0.9398
	1.6	39.32	39.27	0.9149	0.9154
	1	39.23	39.21	0.8761	0.8770
	0.5	38.19	38.14	0.7946	0.7956
	0.2	37.17	37.18	0.6987	0.6998
Bike	2.5	39.84	39.77	0.9610	0.9615
	2	38.70	38.59	0.9336	0.9343
	1.6	37.78	37.83	0.9064	0.9073
	1	34.59	34.62	0.8430	0.8439
	0.5	33.62	33.63	0.7316	0.7324
	0.2	34.67	34.77	0.5681	0.5690

Table 3. Comparison between the PSNR(dB) and SSIM of the images obtained from the watermarked bitstream with the proposed joint JPEG2000/watermarking decoder and the JPEG2000 part 2 decoder.

The computational complexity of the proposed joint scheme and JPEG2000 are similar. The encoding and decoding execution times of the two coders are nearly the same because, except the quantization stage, there are no additional processing to watermark the image.

The proposed joint scheme produces a JPEG2000 syntax compliant bitstream. This bitstream can be decoded by a classical JPEG2000 decoder. In this case, the two union quantizers  $A_0$  and  $A_1$  are used to dequantize the decoded wavelet coefficients instead of group 0 and 1 dithered union quantizers (the step size values are stored in the header of the JPEG2000 codestream). However, the JPEG2000 decoder produces an image which is close in quality to the one decoded with our joint scheme as shown in Table 3. For the three test images Lena, Goldhill and Bike, the PSNR and SSIM results are similar and sometimes better than those obtained with the joint decoder.

### 6.2.3.2 Watermarking performances

In order to analyze the performance of the proposed joint system in terms of robustness, we compare its robustness with those of two conventional watermarking schemes: the dirty paper trellis codes (Miller et al. (2004)) and the TTCQ scheme (Le-Guelvouit (2009)). The two schemes use a trellis during watermark embedding and extraction stages. The TTCQ algorithm is a TCQ-based scheme which relies on the use of turbo codes to embed the watermark in the JPEG domain (Section 3.2). We use a specific protocol for the two

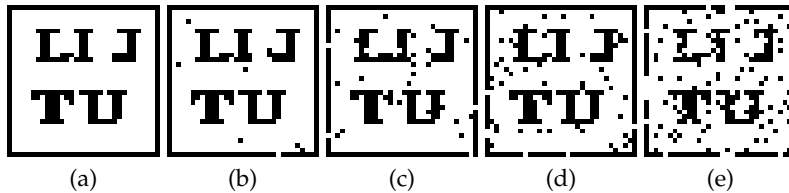


Fig. 14. Various examples of extracted watermarks after attacks: (a) original watermark, the extracted watermark with a BER of (b) 1.07%, (c) 3.34%, (d) 5.37%, and (e) 10.25%.

watermarking schemes to be able to make a valid comparison: we perform JPEG2000 compression attack after watermark embedding and before performing robustness attacks. The compression bitrate is fixed to 2 bpp. We fixed the degradation to an average PSNR value of 45 dB for the three schemes. The payload is the same as the one used in the experimentations (1 bit embedded in 256 pixels). We made a slight modification to the TTCQ algorithm to be able to use a payload of 1024 bits instead of 4096 bits (which is the fixed payload for an image of size 512 x 512 for this scheme). We use a very simple convolutional code 1/4-rate to code the message of 1024 bits.

The same database of 200 images has been considered to compare the watermark robustness of the joint scheme with those of DPTC and TTCQ. Four kinds of attacks have been performed: gaussian filtering attack, Gaussian noise attack, valumetric attack and JPEG attack. The Bit Error Rate (BER) is computed for each attack. The BER is the number of erroneous extracted bits divided by the total number of embedded bits. In Fig. 14, various examples of the extracted watermarks according to their BER are shown. As can be seen in the figure, an extracted watermark with a BER bigger than 10% is hard to recognize. The BER results for the four attacks are presented in Fig. 15 Fig. 16, Fig. 17 and Fig. 18. The logarithmic (base 10) scale is used for the Y-axis (BER results).

The watermarked images are filtered by gaussian filter of width  $\sigma$ . The experiment was repeated for different values of  $\sigma$ , and the BER has been computed. The obtained results are reported in Fig. 15. We notice that the watermark robustness against this kind of attack is relatively the same for the three schemes but the BER values obtained by DPTC are much more lower than ours and TTCQ for low and middle power attack. The joint scheme gives better BER value than TTCQ until  $\sigma = 0.9$ . Fig. 16 shows the results obtained when the watermarked images are corrupted by additive gaussian noise with mean 0 and standard deviation  $\sigma$ . The experimental results shows that DPTC outperforms the two schemes. The proposed joint scheme is not robust to this kind of attack. The TTCQ scheme provides a better robustness than our joint scheme. The results against the valumetric scaling attack (each pixel is multiplied by a constant) are summarized in Fig. 17. The joint scheme gives better performances than TTCQ for this kind of attack. DPTC allows to obtain a null BER until a scaling factor value of 1.1. From this value, the joint scheme gives better BER than the two other approaches. Fig. 18 shows the BER results against JPEG attack. The two watermarking shemes provide better performances than the proposed joint scheme. The weak robustness to JPEG attack is inherent to the joint approach since the transformed domain is the wavelet domain and the coefficients included in the watermarking process are partly high frequency wavelet coefficients.

To sum up, the two watermarking schemes provide better watermark robustness than the proposed joint scheme facing gaussian noise attack and JPEG attack. The joint scheme is more

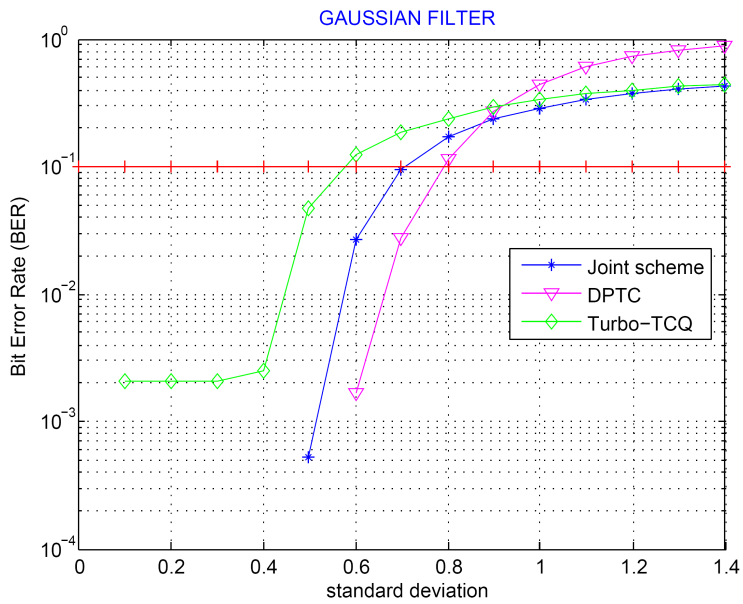


Fig. 15. BER results for Filtering attack

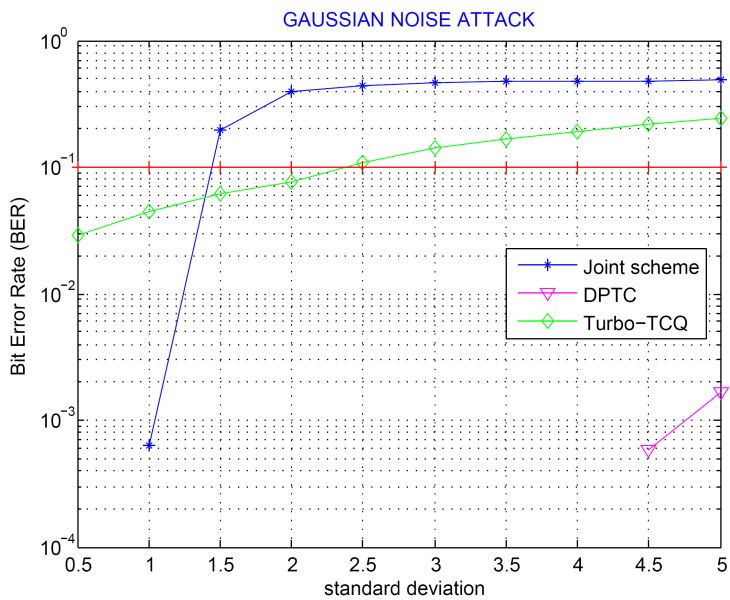


Fig. 16. BER results for Gaussian attack.

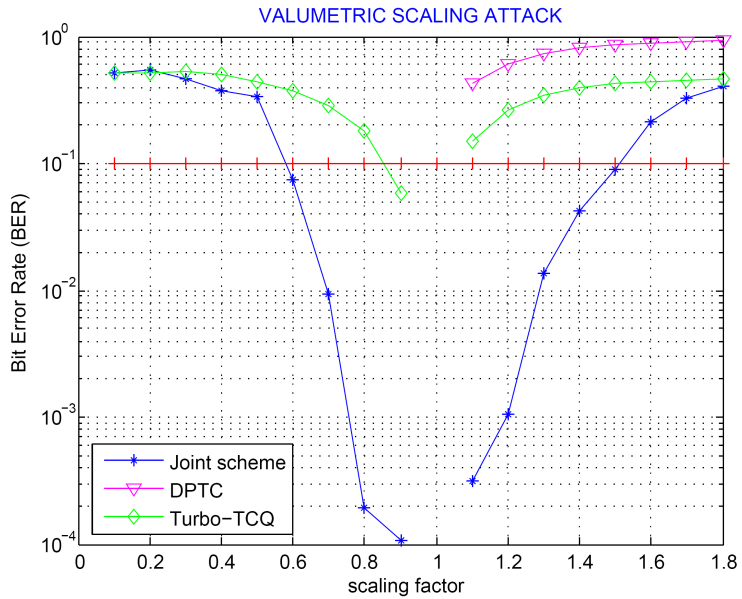


Fig. 17. BER results for Scaling attack

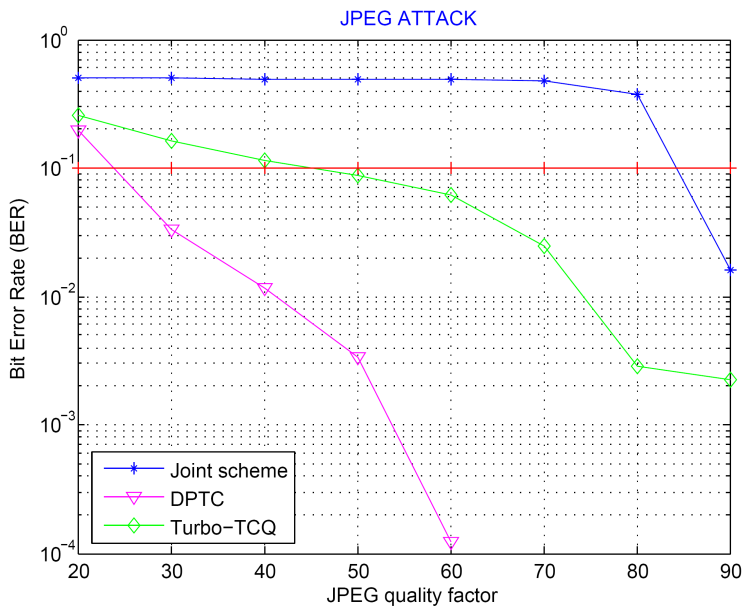


Fig. 18. BER results for JPEG attack.



resistant than TTCQ against the valumetric attack without exceeding DPTC performances. Finally, the robustness against gaussian filter is comparable for the three approaches.

We compare between the computational complexity of the joint scheme and those of watermarking schemes and separate JPEG2000 compression. DPTC achieve high performances with respect to robustness. However, this watermarking scheme suffers from its CPU computational complexity. Three hours are necessary to watermark an  $512 \times 512$  image with the DPTC algorithm on an Intel dual core 2 GHZ processor while it requires only 2 seconds to watermark and compress the same image with our joint scheme. The TTCQ watermark embedding / JPEG2000 compression / TTCQ watermark extraction operations need 15 seconds to be performed. So, the joint scheme is much more faster than the two other schemes.

## 7. Conclusion

In this chapter, the use of quantization watermarking in the design of joint compression and data hiding schemes is investigated in the JPEG2000 still image compression standard framework. Instead of treating data hiding and compression separately, it is interesting and beneficial to look at the joint design of data hiding and compression systems. We have proposed two TCQ quantization strategies in the coding pipeline of JPEG2000 part 2, leading to the design of two joint schemes. We exploit the characteristics of the TCQ quantization as defined in the part 2 of the standard to perform information hiding. The main contribution of this work is that the proposed schemes allows both quantization of wavelet coefficients and data hiding by using the same quantization module.

In the first joint data hiding and JPEG2000 scheme, we propose to hide the message by quantizing selected wavelet coefficients with specific codebooks from the JPEG2000 union quantizers. 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 control stage of JPEG2000 compression. In the second joint watermarking and JPEG2000 scheme, we propose to watermark the image during JPEG2000 compression and decompression stages by using two groups of dithered union quantizers. We replace the uniform scalar quantizers used in JPEG2000 part 2 by shifted scalar quantizers. The trellis structure used during quantization is modified in order to add branches labeled by these dithered quantizers. The trellis is afterwards pruned so that all the branches through the trellis, and thus all the associated union quantizers, encode the message to embed.

Experimental investigations covered both the compression efficiency and the data hiding performances of the proposed schemes. The properties of our joint schemes are the following:

- robustness to JPEG2000 compression for all tested bitrates,
- good image quality,
- high payloads,
- lower complexity in comparison to the separate approach.

The work presented in this chapter suggests that it is possible to design joint data hiding and compression coders in the JPEG2000 domain with smaller complexity and relatively good performances. The proposed joint schemes can be used in enrichment and management applications.

## 8. References

- Braci, S., Boyer, R. & Delpha, C. (2009). Security evaluation of informed watermarking schemes, *Proc. International Conference on Image Processing, ICIP 2009* pp. 117–120.
- Chaumont, M. (2010). A novel embedding technique for dirty paper trellis codes watermarking, *Visual Information Processing and Communication, VIPC 2010, Part of IS&T/SPIE 22th Annual Symposium on Electronic Imaging, SPIE 2010*, Vol. 7543, paper 7543-38, San Jose, California, USA.
- Chen, B. & Wornell, G. W. (2001). Quantization index modulation: A class of provably good methods for digital watermarking and information embedding, *IEEE Transactions on Information Theory* 47(4): 1423–1443.
- Chen, J., Chen, T. S., Lin, C. N. & Cheng, C. Y. (2010). A bitrate controlled data hiding scheme for JPEG2000, *International Journal of Computers and Applications* 32(2): 238–241.
- Chou, J., Pradhan, S. S. & Ramchandran, K. (1999). On the duality between distributed source coding and data hiding, *Proc. of the Thirty-third Asilomar Conference on Signals, Systems, and Computers*, Vol. 2, Pacific Grove, CA, USA, pp. 1503–1507.
- Costa, M. (1983). Writing on dirty paper, *IEEE Transactions on Information Theory* 29(3): 439–441.
- Cox, I. J., Miller, M. L., Bloom, J. A., Fridrich, J. & Kalker, T. (2008). *Digital watermarking and steganography*, second edn, Morgan Kaufmann.
- Eggers, J., Bauml, R., Tzschoppe, R. & Girod, B. (2003). Scalar costa scheme for information embedding, *IEEE Transactions on Signal Processing* 51(4): 1003–1019.
- Esen, E. & Alatan, A. A. (2004). Data hiding using trellis coded quantization, *Proc. of the IEEE International Conference on Image Processing, ICIP2004*, Vol. 1 of *Proc. ICIP 2004*, Singapore, pp. 59–62.
- Fan, Y.-C. & Tsao, H. (2007). A dual pyramid watermarking for JPEG2000, *International Journal of High Performance Computing and Networking* 5: 84–96.
- Fan, Y., Chiang, A. & Shen, J. (2008). ROI-based watermarking scheme for JPEG 2000, *Springer journal of Circuits, Systems, and Signal Processing* 27(5): 763–774.
- Forney Jr, G. D. (1973). The viterbi algorithm, *IEEE Transaction on Information Theory* 61: 268–278.
- Goudia, D., Chaumont, M., Puech, W. & Said, N. H. (2011a). A joint JPEG2000 compression and watermarking system using a TCQ-based quantization scheme, *VIPC 2011, SPIE 2011, Visual Information Processing and Communication II, Part of SPIE 23th Annual Symposium on Electronic Imaging*, Vol. 7882-11, San Francisco, California, USA.
- Goudia, D., Chaumont, M., Puech, W. & Said, N. H. (2011b). A Joint Trellis Coded Quantization (TCQ) Data Hiding Scheme in the JPEG2000 Part 2 Coding Framework, *The 19th European Signal Processing Conference, EUSIPCO 2011*, Barcelona, Spain.
- ISO/IEC JTC1/SC29 WG1 (2000). JPEG2000 part II final committee draft version 1.0.
- Le-Guelvouit, G. (2005). Trellis-coded quantization for public-key watermarking, accepted for IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2005), 2005, see the website <http://www.gleguelv.org/pub/index.html>.
- Le-Guelvouit, G. (2009). Tatouage robuste d'images par turbo tcq, *Traitement du Signal* 25(6).
- Li, Q. & Cox, I. (2007). Using perceptual models to improve fidelity and provide resistance to valumetric scaling for quantization index modulation watermarking, *IEEE Transactions on Information Forensics and Security* 2(2): 127–139.
- Lin, L., Cox, I. J., Doërr, G. & Miller, M. L. (2005). An efficient algorithm for informed embedding of dirty paper trellis codes for watermarking, *Proc. of the IEEE International Conference on Image Processing, ICIP 2005*, Vol. 1, Genova, Italy, pp. 697–700.

- Marcellin, M. & Fischer, T. (1990). Trellis coded quantization of memoryless and gauss-markov sources, *IEEE Transaction on communication* 38: 82–93.
- Meerwald, P. (2001). Quantization watermarking in the JPEG2000 coding pipeline, *Proc. of the 5th joint working Conference on Communications and Multimedia Security, Communications and Multimedia Security Issues of the new century, IFIP TC6/TC11, Darmstadt, Germany*, pp. 69–79.
- Miller, M., Doerr, G. & Cox, I. (2004). Applying Informed Coding and Embedding to Design a Robust High Capacity Watermark, *IEEE Transactions on Image Processing* 13(2): 792–807.
- Ouled-Zaid, A., Makhoulfi, A. & Bouallegue, A. (2007). Wavelet Domain Watermark Embedding Strategy using TTCQ Quantization, *International Journal of Computer Science and Network Security (IJCSNS)* 7(6): 165–170.
- Ouled-Zaid, A., Makhoulfi, A. & Olivier, C. (2009). Improved QIM-Based Watermarking Integrated to JPEG2000 Coding Scheme, *Springer journal of Signal, Image and Video Processing* 3: 197–207.
- Pérez-González, F., Mosquera, F., Barni, M. & Abrardo, A. (2005). Rational dither modulation: A high-rate data-hiding method invariant to gain attacks, *IEEE Transactions on Signal Processing, Supplement on Secure Media* 53(10): 3960–3975.
- Schlauweg, M., Profrock, D. & Muller, E. (2006). JPEG2000-based secure image authentication, *Proc. of the 8th ACM Multimedia and Security Workshop, Proc. MM&Sec 2006, Geneva, Switzerland*, pp. 62–67.
- Schuchman, L. (1994). Dither signals and their effect on quantization noise, *IEEE Transaction on Communication Technology (COM)* 12: 162–165.
- Su, P. C. & Kuo, C. J. (2003). Steganography in JPEG2000 compressed images, *IEEE Transaction on Consumer Electronics* 49(4): 824–832.
- Taubman, D. & Marcellin, M. (2001). *JPEG2000: Image Compression Fundamentals, Standards, and Practice*, Kluwer Academic Publishers, Dordrecht.
- Thomos, N., Boulgouris, N. V., Kokkinou, E. & Strintzis, M. G. (2002). Efficient data hiding in JPEG2000 images using sequential decoding of convolutional codes, *Proc. of the International Conference in Digital Signal Processing 2002, Vol. 2*, pp. 710–720.
- Ungerboeck, G. (1982). Channel coding with multilevel/phase signals, *IEEE Transaction on Information Theory* 28: 55–67.
- Wang, X. & Zhang, X. P. (2007). Generalized trellis coded quantization for data hiding, *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2007* pp. 269–272.