

Tatouage et Compression Conjoint dans JPEG2000 avec un Algorithme de Quantification Codée par Treillis (TCQ)

D. Goudia^{1,2}

M. Chaumont¹

W. Puech¹

N. Hadj Said²

¹ LIRMM, UMR CNRS 5506, Université Montpellier II
161 rue Ada, 34392, MONTPELLIER cedex 5, FRANCE
{dalila.goudia, marc.chaumont, william.puech}@lirmm.fr

² SIMPA (Laboratoire Signal Image Parole)
USTO, Université des Sciences et de la Technologie d'Oran
BP 115, El M'aouael, ORAN, ALGERIE
{min.hadj}@yahoo.fr

Résumé

Dans cet article, nous présentons une méthode conjointe de quantification et de tatouage basée sur la Quantification Codée par Treillis (TCQ). Cette technique a été intégrée au niveau du codeur JPEG2000. Plus précisément, nous cherchons à effectuer conjointement compression et tatouage à l'aide de la TCQ en réalisant simultanément la quantification et l'insertion de la marque durant la compression. Le processus d'extraction de la marque peut être mis en œuvre à la fois durant la décompression ou après celle-ci. Les résultats obtenus ont montré que ce schéma conjoint résiste à une compression JPEG2000 avec variation du taux de compression sans dégradation de la qualité de l'image décompressée.

Mots clefs

Compression d'images, tatouage numérique, JPEG2000, TCQ.

1 Introduction

Le tatouage d'image consiste à insérer (généralement sous forme invisible) une information dans une image puis à tenter de la récupérer après que l'image ait éventuellement subi des manipulations de nature variée. Il existe de nombreuses méthodes d'insertion dans la littérature. Les méthodes de tatouage informées [1, 2, 3] reposent sur l'utilisation de dictionnaires structurés et considèrent le tatouage comme un problème de codage canal. Durant l'insertion, deux phases successives sont exécutées : le *codage informé* et l'*insertion informée*. Le codage informé consiste à construire un dictionnaire de mots de codes et ensuite à trouver le mot de code \mathbf{c}^* le plus proche du signal hôte \mathbf{x} . Ce mot de code représente le message \mathbf{m} à insérer. L'insertion informée va déplacer le signal hôte \mathbf{x}

vers le mot de code \mathbf{c}^* . Le vecteur déplacement est appelé *signal de tatouage*. Cette étape consiste à modifier \mathbf{x} pour le déplacer dans la région de détection introduisant le moins de distorsion. Les méthodes de tatouage par quantification sont une implantation pratique du principe de tatouage informé, et plus précisément du schéma de Costa [4]. En 1999, Chen et Wornell [1] ont proposé sous le nom de QIM (Quantization Index Modulation), l'implantation la plus courante du schéma de Costa. Eggers *et al.* [2] ont présenté sous le nom de Schéma de Costa Scalaires SCS (Scalar Costa Scheme) une implantation sous-optimale explicite du schéma de Costa similaire à celle de la DC-QIM [1].

La compression représente, non seulement un passage obligé du stockage ou du transfert d'images, mais aussi l'une des attaques les plus destructrices vis-à-vis du tatouage. La compression jointe au tatouage a suscité un intérêt récent. Associer tatouage et compression permet d'améliorer la capacité d'insertion du tatouage et d'assurer une meilleure détection tout en maintenant une bonne qualité d'image. En effet, cette approche offre de nombreux avantages qui sont très recherchés dans certaines applications telles que le contrôle d'intégrité, l'authentification ou la production de documents enrichis. La norme de compression JPEG2000 [5] a été développée par le groupe JPEG (Joint Photographic Experts Group). Ce standard offre de nombreuses fonctionnalités et se caractérise par une grande diversité des options d'encodage menant à de bons compromis compression/qualité. Le marquage d'images dans le domaine JPEG2000 a déjà fait l'objet de plusieurs travaux [6, 7, 8].

L'objectif de cet article est de concevoir un nouveau schéma permettant de combiner la compression JPEG2000 et une méthode de tatouage fondée sur la quantification. Ce schéma conjoint repose sur un module hybride de quan-

tification codée par treillis (TCQ) [9] capable de quantifier et d'insérer la marque en une seule fois. Ce module est intégré dans la chaîne de codage de JPEG2000 à la place de l'étape de quantification scalaire uniforme avec zone morte. L'algorithme utilise la version TCQ telle que définie dans la partie 2 de la norme [10]. Nous pouvons extraire la marque de deux manières : durant ou bien après le processus de décompression. L'approche conjointe nécessite de prendre en compte le taux de compression obtenu après codage entropique : l'insertion de la marque ne doit pas aboutir à une dégradation sensible des performances du compresseur (taux de compression et qualité de l'image reconstruite). Il est nécessaire de maintenir un compromis débit-distorsion optimal. Le troisième critère à considérer concerne la robustesse du tatouage. La marque doit pouvoir survivre à certaines attaques, et ce, de manière à pouvoir assurer une extraction correcte de celle-ci à partir de l'image compressée/tatouée. Notons qu'actuellement, il n'existe pas de véritables schémas conjoints dans le domaine JPEG2000. Les travaux proposés dans ce domaine se contentent d'intégrer une étape supplémentaire d'insertion/extraction de la marque dans le schéma de fonctionnement de JPEG2000. Notre système permet à la fois de quantifier et de marquer les coefficients d'ondelettes en utilisant un seul et même composant.

Ce papier est organisé de la façon suivante : dans la section 2, nous passons en revue les techniques de tatouage basées-quantification combinées à JPEG2000. La section 3 présente la quantification TCQ dans JPEG2000 ainsi que les techniques de tatouage basées-quantification TCQ. Les détails du schéma conjoint sont donnés dans la section 4. La section 5 est consacrée à la discussion des premiers résultats obtenus. Enfin, la dernière section conclut le présent article.

2 Le tatouage dans JPEG2000

JPSEC (Secure JPEG2000) [11] qui représente la partie 8 du standard, propose des solutions permettant à des applications de générer, décoder et échanger des bit-stream JPEG2000 sécurisés. Les mécanismes de sécurisation sont intégrés dans le bit-stream sous la forme de métadonnées. Ces informations peuvent donc être perdues lors d'opérations de conversion de format. Par contre, lorsque la marque est insérée directement durant le processus de codage, celle-ci devient partie intégrante de l'image. Un certain nombre de travaux consacrés à l'aspect compression/tatouage conjoint avec JPEG2000 dans le cadre d'une application d'authentification d'images ou de protection de copie ont été proposés. Seul un petit nombre de ces travaux utilisent une technique de tatouage fondé sur la quantification. Meerwald *et al.* [6] ont proposé une méthode de tatouage aveugle de type QIM intégrée dans le schéma de fonctionnement de JPEG2000. Schlaueg *et al.* [7] ont présenté un schéma de tatouage semi-fragile sécurisé basé sur JPEG2000 utilisant la DM-QIM (Dither Modulation QIM) et des outils cryptographiques tels que les fonctions

de hachage et les méthodes de cryptographie. Makhouloufi *et al.* [8] ont développé un algorithme de tatouage aveugle basé sur la QIM qui, contrairement aux travaux précédents, effectue l'insertion de la marque avant l'étape de quantification. Le codeur utilisé est un codeur compatible avec la partie 2 de la norme et les auteurs utilisent un décalage non-linéaire dans leur méthode de tatouage pour réduire la distorsion causée par l'insertion de la marque. Aucune de ces techniques ne considère la quantification scalaire du schéma de codage JPEG2000 comme étant un bruit connu à l'insertion. Depuis la redécouverte des travaux de Costa [4], nous savons que si nous prenons en compte le bruit connu, nous augmentons la capacité du canal. Par conséquent, nous proposons un système combiné compression/tatouage informé dans JPEG2000 prenant en compte le bruit de quantification.

3 La Quantification Codée par Treillis (TCQ)

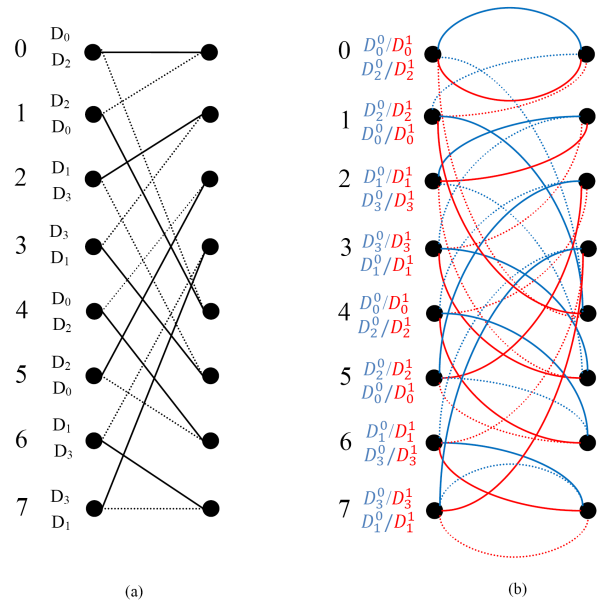


Figure 1 – La structure du treillis utilisé : a) dans JPEG2000 b) dans notre schéma de compression/tatouage conjoint.

La Quantification Codée par Treillis (TCQ) est une technique de quantification rapide proposée par Marcelin et Fisher [9]. La TCQ est basée sur l'idée de partitionnement d'ensembles proposée par Ungerboeck [12] pour combiner la modulation et le codage de canal. Elle consiste à partitionner un dictionnaire de quantification initial en sous-dictionnaires complémentaires associés aux transitions entre les états d'un code convolutif.

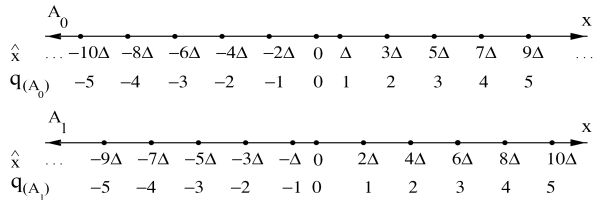


Figure 2 – Les quantificateurs d'union de la TCQ dans JPEG2000.

3.1 La TCQ dans JPEG2000

Plusieurs techniques de quantification sont proposées dans la partie 2 du standard. Parmi celles-ci, nous pouvons citer la TCQ [5, 10]. Cette technique permet d'atteindre de meilleures performances en termes de débit-distorsion par rapport à la quantification scalaire, et ce, à différents taux de compression. La complexité du codeur s'en trouve cependant augmentée. Le code convolutif est représenté par un treillis. Le treillis est constitué d'un ensemble d'états reliés entre eux par des arcs ou branches. Il s'agit d'une variante du diagramme de transition avec prise en compte du temps. Chaque branche dans le treillis représente une transition entre deux états à un instant donné. A partir d'un état initial, le chemin dans le treillis peut être spécifié par une séquence binaire puisqu'il n'y a que deux transitions possibles d'un état à un autre. Concernant la variante de la TCQ incluse dans la norme JPEG2000, le dictionnaire de quantification du quantificateur scalaire uniforme de pas de quantification Δ est partitionné en 4 sous-dictionnaires notés D_0, D_1, D_2 et D_3 . Chaque branche du treillis est étiquetée par un sous-dictionnaire D_i . La Figure 1.a décrit la structure d'un treillis à 8 états ainsi que les sous-dictionnaires associés aux branches de ce treillis. L'union des quantificateurs associés à chaque état est appelé quantificateur d'union. Les quantificateurs d'union utilisés dans le treillis de la figure 1.a sont $A_0 = D_0 \cup D_2, A_1 = D_1 \cup D_3$. Les dictionnaires de ces quantificateurs sont illustrés au niveau de la Figure 2.

Chaque état dans le treillis est associé à l'un des deux quantificateurs d'union. Afin de quantifier la séquence source \mathbf{x} , l'algorithme de Viterbi [13] est utilisé pour trouver le chemin optimal à travers le treillis. Le chemin optimal indique la séquence de bits de transition pour laquelle la distorsion totale est minimale, c'est-à-dire que l'on recherche le vecteur quantifié $\hat{\mathbf{x}}$ le plus proche du vecteur source \mathbf{x} au sens de l'erreur quadratique. En plus de la séquence de bits indiquant le chemin optimal, l'algorithme de Viterbi produit également une séquence d'indices de quantification TCQ, nécessaire pour indiquer les niveaux choisis à l'intérieur des sous-dictionnaires du chemin optimal. A la réception, le décodeur reconstruit la source quantifiée comme suit : la séquence de bits indiquant le chemin optimal à travers le treillis ainsi que la séquence d'indices TCQ sont utilisés comme entrée du codeur convolutif. A chaque transition i , le bit de chemin permet de retrouver l'état suivant et

donc le sous dictionnaire de quantification utilisé. L'indice TCQ permet de reconstruire l'échantillon source qui sera présenté en sortie du codeur.

3.2 La TCQ en tatouage

Les techniques de tatouage basé sur la TCQ sont peu nombreuses. Elles reposent toutes sur le principe suivant : durant l'encodage, on quantifie la séquence source \mathbf{x} en forçant les transitions du treillis afin qu'elles correspondent au message à encoder. Le rendement de la TCQ est de 1/1 (un bit inséré pour un échantillon source). Cette approche est appelée sélection de chemin TCQ (TCQ-PS : TCQ path selection). Braci *et al.* [14] se sont focalisés sur l'aspect sécurité des schémas de tatouage informé basé sur la QIM et ont proposé une version sécurisée de la TCQ-PS. Ouled Zaid *et al.* [15] ont développé un algorithme de tatouage basé sur la Turbo TCQ dans le domaine ondelettes. Aucune de ces techniques de tatouage n'a été combinée à un codeur JPEG2000. Le principe sur lequel repose notre approche est le suivant : quantifier et marquer les coefficients d'ondelettes en même temps grâce à l'utilisation d'un module hybride de quantification TCQ. Notre technique de tatouage possède des similitudes avec celle des codes à papier sales ou DPTC [3]. Les deux méthodes reposent sur l'utilisation d'un treillis modifié associé à un dictionnaire. Cependant, nous utilisons un dictionnaire de quantification alors que Miller *et al.* [3] utilise un dictionnaire de codage structuré. De plus, l'insertion de la marque se fait différemment. Notre schéma conjoint intègre un système de tatouage informé basé sur une quantification alors que celui de Miller *et al.* est un algorithme d'insertion itératif permettant de construire pas à pas le tatouage jusqu'à atteindre les conditions de transmission robuste.

4 Le schéma conjoint de compression et de tatouage dans JPEG2000

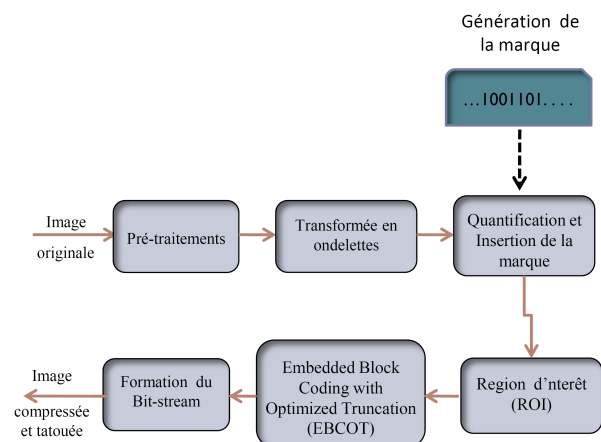


Figure 3 – Schéma de fonctionnement du système conjoint codage JPEG2000/tatouage.

La contribution majeure apportée par notre travail concerne l'utilisation d'un module hybride de quantification permettant de quantifier et de tatouer simultanément les coefficients d'ondelettes. Le schéma de fonctionnement de notre système de compression/tatouage conjoint est illustré sur la Figure 3. La TCQ-PS [14] ne peut pas être adaptée à notre approche conjointe car le chemin dans le treillis correspond au message inséré. La sécurité du message n'est donc pas assurée puisque le chemin fait partie intégrante du bit-stream JPEG2000. Nous proposons une nouvelle technique de tatouage basée sur la TCQ qui est indépendante de la sélection du chemin. L'insertion de la marque se fait au niveau des sous-bandes de détails LH, HL et HH issues de la décomposition en ondelettes. Le nombre de sous-bandes qui seront quantifiées/tatouées dépend du nombre de niveaux de décomposition inclus dans le processus de tatouage. La capacité est calculée à partir du nombre de sous-bandes candidates au tatouage. Les coefficients d'ondelettes appartenant aux autres sous-bandes sont quantifiés avec le quantificateur scalaire uniforme classique. Les quantificateurs tels que définis dans la norme JPEG2000 partie 2 sont remplacés par des quantificateurs décalés. Ces quantificateurs décalés possèdent le même pas de quantification Δ que les originaux mais ils diffèrent de ces derniers par l'utilisation d'un décalage $\mathbf{d} \in [-\Delta/2, \Delta/2]$ [1]. Ce décalage est obtenu aléatoirement à partir d'une clé secrète. Le principe de tatouage est le suivant : on associe à chaque bit du message à insérer un décalage \mathbf{d} . Si le bit à insérer est égal à 0 alors on utilise le quantificateur $D_j^0, j = 0, 1, 2, 3$ avec le décalage $d[0]$. Si le bit est égal à 1 alors cela induit l'utilisation du quantificateur D_j^1 avec le décalage $d[1]$ remplissant la condition : $|d[1] - d[0]| = \Delta/2$. Nous disposerons de deux groupes de quantificateurs d'union dans notre treillis : le groupe 0, $A_0^0 = D_0^0 \cup D_2^0, A_1^0 = D_1^0 \cup D_3^0$ qui représente l'insertion du bit 0 et le groupe 1, $A_0^1 = D_0^1 \cup D_2^1, A_1^1 = D_1^1 \cup D_3^1$ correspondant à l'insertion d'un bit 1. Soit le message binaire \mathbf{m} à insérer et le signal hôte \mathbf{x} . L'insertion de la marque guide le processus de quantification c'est-à-dire qu'à chaque transition i dans le treillis, la valeur du bit $\mathbf{m}[i]$ va déterminer quel sera le quantificateur d'union à utiliser pour quantifier le coefficient d'ondelette $\mathbf{x}[i]$. Les sous-dictionnaires $D_j^{m[i]}, j = 0, 1, 2, 3$ vont étiqueter les branches du treillis. Le treillis classique tel qu'utilisé dans JPEG2000 comporte deux arcs ou branches par état. A chaque branche est attribuée un bit appelé bit de chemin (0 ou 1). Dans notre treillis, le nombre d'arcs est multiplié par 2. Au niveau de la transition i , pour un état e et un bit de chemin donné (0 ou 1), il y a 2 arcs possibles vers un état de la transition $i+1$ comme l'illustre la Figure 1.b. A l'un des arcs est associé un sous-dictionnaire du groupe 0 (par exemple le sous-dictionnaire D_0^0 du groupe A_0^0) et à l'autre arc le sous-dictionnaire appartenant au groupe 1 (le sous-dictionnaire D_0^1 du groupe A_0^1). Avant de procéder à la quantification, le treillis est élagué de façon à supprimer toutes les branches ne correspondant pas aux sous-dictionnaires encodant le

message. On se retrouvera avec un treillis à deux branches par état mais celles-ci sont associées aux quantificateurs encodant le message \mathbf{m} . L'étape de quantification produit la séquence de bit de chemin \mathbf{p} et la séquence d'indice de quantification TCQ notée \mathbf{q} donnée par :

$$\begin{aligned} q[i] &= Q_{D_j^{m[i]}}(x[i]) \\ &= \text{sign}(x[i] - d_i[m[i]]) \lfloor \frac{|x[i] - d_i[m[i]]|}{\Delta_j} \rfloor \end{aligned} \quad (1)$$

où d_i représente le décalage du quantificateur décalé $D_j^{m[i]}$ à la transition i et Q est la fonction de quantification. En plus des séquences \mathbf{p} et \mathbf{q} , nous avons besoin d'une information supplémentaire qui sera codée et stockée au niveau de la séquence \mathbf{l} . Celle ci va nous permettre de retrouver, durant l'étape de quantification inverse, la structure du treillis modifié utilisé lors de la quantification. Connaissant $\mathbf{p}, \mathbf{q}, \mathbf{l}$ et \mathbf{d} , les valeurs de reconstruction $\hat{\mathbf{x}}$ sont obtenues de la manière suivante :

$$\begin{aligned} \hat{x}[i] &= \bar{Q}_{D_j^{m[i]}}^{-1}(q[i]) \\ &= \text{sign}(q[i])(|q[i]| + \delta)\Delta_j + d_i[m[i]] \end{aligned} \quad (2)$$

où \bar{Q}^{-1} est la fonction de quantification inverse et δ le paramètre de reconstruction avec $0 < \delta < 1$. Le treillis utilisé est le même que celui employé lors de la quantification.

4.1 Quantification et insertion de la marque

L'insertion de la marque est effectuée durant la quantification de manière indépendante au niveau de chaque code-block [5]. Dans un premier temps, le message à insérer est généré puis codé à l'aide d'un code correcteur d'erreur afin d'augmenter la robustesse du message transmis. Pour chaque code-block, les étapes suivantes sont exécutées :

- Tirage de nombres aléatoires à l'aide d'un générateur pseudo-aléatoire initialisé par une clef secrète : ces nombres seront utilisés comme décalage \mathbf{d} dans les quantificateurs décalés.
- Génération des deux groupes de quantificateurs d'union (le groupe 0 et le groupe 1) pour chaque transition i .
- Elagage du treillis : l'état initial est mis à 0. Nous parcourons le treillis et pour chaque transition nous supprimons les branches dont les sous-dictionnaires associés ne correspondent pas au message à encoder (figure 4).
- Quantification et tatouage : exécution de l'algorithme de Viterbi afin de trouver le chemin optimal. La figure 4 illustre un exemple d'insertion du message $\mathbf{m} = \{1, 0, 1\}$. Le chemin en gras représente le chemin à distorsion minimal déterminé par l'algorithme de Viterbi.

4.2 Extraction de la marque à partir de l'image décompressée

L'image décompressée/marquée est transformée à l'aide d'une transformée en ondelettes discrète. Les coefficients d'ondelettes appartenant aux sous-bandes incluses dans le processus de tatouage sont sélectionnés et placés dans le

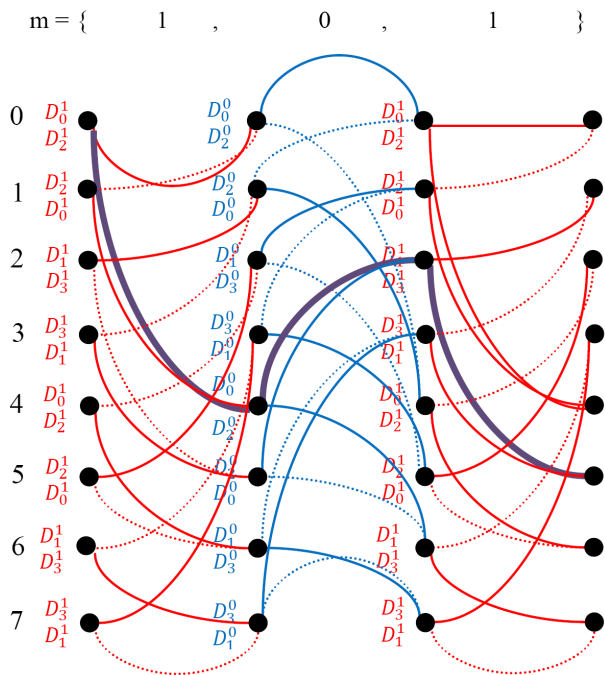


Figure 4 – Insertion du message $m = 1, 0, 1$.

vecteur y . L'extraction du message est réalisée en décodant y par un algorithme de Viterbi appliqué au treillis complet, c'est à dire sans élagage.

4.3 Extraction de la marque pendant la décompression

On peut également extraire le message inséré durant la décompression JPEG2000 lors de l'étape de quantification inverse. Pour chaque code-block, les étapes sont les suivantes :

- Récupération des décalages d à l'aide de la clé secrète et génération des groupes de quantificateurs d'union 0 et 1.
- Extraction de la marque et quantification inverse : La séquence I est utilisée pour reconstruire la structure du treillis employée lors de la quantification. Le message inséré peut alors être extrait en examinant le type de quantificateurs qui étiquettent les branches du treillis. Si le groupe 0 est utilisé au niveau de la transition i , cela signifie que le bit inséré $\hat{m}[i]$ est égal à 0. Sinon, c'est le groupe 1 qui est utilisé et $\hat{m}[i] = 1$. Le processus de quantification inverse est ensuite exécuté afin de reconstruire les coefficients d'ondelettes.

Remarquons qu'il est également possible de décompresser le bit-stream JPEG2000 marqué à l'aide d'un décodeur JPEG2000 classique. Cependant, l'image obtenue sera différente en terme de qualité avec celle obtenue à l'aide du schéma conjoint. De plus, l'information de tatouage sera perdue car le décodeur classique n'utilise pas les quantificateurs décalés. Il sera donc impossible de procéder à une

extraction de la marque pendant ou après la décompression.

5 Résultats expérimentaux

Afin de procéder à l'implémentation de notre schéma conjoint compression/tatouage, nous avons choisi de travailler avec la librairie OpenJPEG [16]. Cette librairie est un code source libre écrit en langage C implémentant la norme JPEG2000 partie 1. Les paramètres de compression et de tatouage qui ont été utilisés dans nos tests sont les suivants : une transformée en ondelettes 9/7 sur 5 niveaux de décomposition, un découpage de l'image en une seule tuile, cette dernière étant découpée en un certain nombre de code-blocks. La taille des code-blocks est égale à 64 x 64 pour les premier et second niveaux de résolution et 16 x 16 pour les niveaux restants. Un message binaire de longueur égale à 4080 bits est inséré au niveau des sous-bandes de détail de tous les niveaux de résolution sauf le premier niveau. Cela signifie que l'on a un bit inséré pour 64 pixels. Le message est codé à l'aide d'un code convolutif de rendement 1/16. Les vecteurs de décalage sont gardés secret afin de protéger la marque. Plusieurs tests ont été ef-

Image test	Débit (bpp)	PSNR (dB) avec JP2k	PSNR (dB) schéma conjoint
Bike	2.5	43.46	43.09
	2	41.47	41.16
	1.6	39.68	39.56
	1	38.07	38.19
	0.5	36.72	36.57
	0.2	33.31	32.92
Peppers	2.5	44.41	44.23
	2	43.25	42.24
	1.6	40.17	40.10
	1	39.05	38.96
	0.5	36.41	36.41
	0.2	29.04	29.42

Tableau 1 – PSNR des images compressées en utilisant JPEG2000 avec et sans tatouage

fectués sur un certain nombre d'images en niveaux de gris. Nous présentons ici les résultats de nos tests sur les images bike et peppers de taille 512 x 512. Nous avons observé les effets induits par une variation du taux de compression comprise entre 2.5 et 0.2 bpp sur la qualité de l'image décompressée/tatouée comme le montre le Tableau 1. Au-delà de 0.2 bpp, il a été constaté que l'image décompressée est trop dégradée pour que la persistance du tatouage soit nécessaire. La qualité de l'image reconstruite est évaluée à l'aide du PSNR. La marque est correctement détectée durant l'étape de décompression, et ce, pour tous les débits binaire testés. La comparaison des mesures de PSNR entre les images compressées et les images compressées/tatouées montrent que l'insertion de la marque entraîne une légère baisse du PSNR. De même, nous avons remarqué que la qualité visuelle de l'image compressée/tatouée est proche

de celle obtenue avec une compression JPEG2000 classique comme l'illustre la figure 5. L'ajout de la marque ne dégrade pas de manière significative le taux de compression. Les résultats obtenus montrent que le tatouage résiste à la phase de contrôle de taux de JPEG2000 et que l'insertion de la marque conduit à une dégradation minimale des performances du codeur.

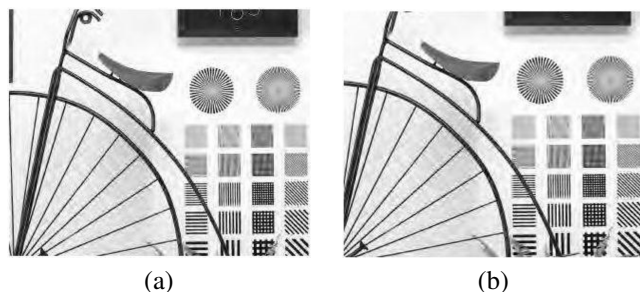


Figure 5 – Image compressée Bike avec et sans tatouage : a) image compressée à 0.5 bpp , b) image compressée et tatouée à 0.5 bpp.

6 Conclusion

Nous avons proposé un algorithme de compression/tatouage conjoint avec JPEG2000 basé sur la quantification TCQ. Les résultats expérimentaux ont prouvé que ce système conjoint fournit de bonnes performances de compression. Il permet d'obtenir une bonne qualité d'image en termes de PSNR. De plus, l'insertion de la marque n'a qu'un faible impact sur le taux de compression. Un autre avantage est la faible complexité de l'ensemble du système du fait d'opérations communes au tatouage et à la compression (transformée, quantification). La capacité atteinte est élevée et permet à l'utilisateur d'insérer suffisamment d'informations sur l'image et sur son auteur. Par conséquent, le schéma proposé constitue une bonne solution pour des applications d'enrichissement de contenu ou d'auto-indexation. Les travaux futurs porteront sur l'étude de la robustesse de ce système conjoint face à certaines attaques d'effacement ainsi qu'à la recherche du meilleur compromis possible entre la robustesse du tatouage et la qualité visuelle de l'image obtenue.

Références

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

[2] J. J. Eggers, R. Bäuml, R. Tzschoppe, et B. Girod. Scalar costa scheme for information embedding. *IEEE Transaction on Signal Processing*, 51(4) :1003–1019, Janvier 2003.

[3] M. L. Miller, G. J. Doerr, et I. J. Cox. Applying informed coding and informed embedding to design a robust, high capacity watermark. Dans *IEEE Transactions on Image Processing*, 2004.

[4] M. Costa. Writing on dirty paper. *IEEE Transaction on Information Theory*, 29(3) :439–441, Janvier 1983.

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

[6] P. Meerwald. Quantization watermarking in the jpeg2000 coding pipeline. Dans *Communications and Multimedia Security*, pages 69–79, Darmstadt, Germany, Mai 2001.

[7] M. Schlauweg, D. Pröfrock, et E. Müller. Jpeg2000-based secure image authentication. Dans *workshop on Multimedia and security*, pages 62–67, Geneva, Switzerland, Septembre 2006.

[8] A. Makhloufi, A. Ouled Zaid, R. Boualleg, et A. Boualleg. Improved quantization index modulation based watermarking integrated to jpeg2000 coding scheme. Dans *SPIE*, pages 17–21, San Diego, USA, Janvier 2010.

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

[10] JPEG2000 Part II Final Committee Draft Version 1.0. *ISO/IEC JTC1/SC29 WG1*. 2000.

[11] F. Dufaux, S. Wee, J. Apostolopoulos, et T. Ebrahimi. Jpsec for secure imaging in jpeg2000. Dans *SPIE, Application of Digital Image Processing*, pages 319–330, Denver, Août 2004.

[12] G. Ungerboeck. Channel coding with multilevel/phase signals. *IEEE Transaction on Information Theory*, 28(1) :55–67, Janvier 1982.

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

[14] S. Braci, R. Boyer, et C. Delpha. Security evaluation of informed watermarking schemes. Dans *ICIP*, Cairo, Egypt, Novembre 2009.

[15] A. Ouled Zaid, A. Makhloufi, et A. Bouallegue. Wavelet domain watermark embedding strategy using ttcq quantization. *IJCSNS*, 7(6) :268–278, Janvier 2007.

[16] The OpenJPEG library is available for download at <http://www.openjpeg.org>.