

# Transfert sécurisée d'une RI sans perte par une méthode d'insertion de données robuste à la compression JPEG

Philippe AMAT, William PUECH

Laboratoire LIRMM, UMR CNRS 5506, Université de Montpellier II  
161, rue Ada, 34392 MONTPELLIER CEDEX 05, FRANCE  
[philippe.amat@lirmm.fr](mailto:philippe.amat@lirmm.fr), [william.puech@lirmm.fr](mailto:william.puech@lirmm.fr)

**Résumé** – Dans cet article, nous présentons une méthode d'insertion de données cachées originale permettant de protéger la haute résolution d'une région d'intérêt. Les pertes de résolution dans la région d'intérêt sont récupérées puis insérées par marquage dans l'image. L'image avec la région d'intérêt sans perte sera visible seulement après l'extraction et la reconstruction de celle-ci. La méthode d'insertion s'appuie sur les coefficients DCT et sur la modification de la matrice de quantification. Cette technique a été développée de manière à être robuste à la compression JPEG.

**Abstract** – In this article, we present an original method of data hiding which allows protecting the high resolution of a region of interest. Resolution losses into region of interest are retrieved and embedded in the image. The image with the region of interest without loss will be visible only after the extraction and the reconstruction of this one. The data hiding method is based on DCT coefficients and quantization matrix modification. This method was developed so as to be robust with JPEG compression.

## 1 Introduction

Le transfert d'information sur les réseaux demande une forte compression des images. Il faut alors trouver un compromis entre rapidité de transfert et conservation d'une très bonne qualité de l'information. Généralement seules certaines régions de l'image présentent un intérêt majeur. De ce fait il n'est pas nécessaire de conserver une très haute qualité pour toute l'image [7, 5, 3, 6]. Une fonctionnalité semblable est offerte par le futur standard de compression JPEG 2000 : il est possible de conserver des régions d'intérêts (RI) sans perte en compressant le reste de l'image [2]. Dans cet article nous présentons une nouvelle méthode qui inclue par insertion de données cachées (IDC) l'information perdue d'une RI lors de la compression JPEG d'une image. L'information perdue représente les pertes engendrées par l'algorithme de compression JPEG dans une RI de l'image. Cette information est incluse dans les coefficients DCT par un algorithme d'IDC [1]. Dans le cas du JPEG2000 la RI sans perte est visible dès l'ouverture de l'image comprimée. Dans le cas de notre méthode, la RI sans perte est protégée à l'ouverture de l'image comprimée. En effet avec notre méthode, l'information complète de la RI n'est visible seulement qu'après l'extraction des données cachées et la reconstruction de la RI. La Section 2 détaille la méthode d'IDC adaptée de la méthode de [1]. Dans la Section 3 nous montrons les résultats obtenus sur l'image de Lena.

## 2 La méthode de protection proposée

### 2.1 Adaptation de la méthode de [1]

Dans la DCT les coefficients les plus importants pour la qualité de l'image sont généralement ceux correspondant aux basses fréquences. Les coefficients les plus quantifiés par la compression sont ceux correspondant aux hautes fréquences. La proposition de [1] est de cacher l'information dans les moyennes fréquences. Dans cette méthode les coefficients à marquer seront quantifiés par 1. Le Tableau 1 montre la table de quantification standard du JPEG pour un facteur de qualité de 50 avec la modification de 26 coefficients comme le suggère [1]. De ce fait, le taux de compression diminue alors que la qualité de l'image augmente. Dans cette méthode 2 bits du message sont marqués par substitution sur les deux bits de poids faible de 26 coefficients moyennes fréquences, soit une capacité de 52 bits par bloc. Cette méthode va donc nous permettre de dissimuler des messages de taille conséquente et surtout de connaître à l'avance la capacité de l'IDC. Pour notre problème, la quantité de données à insérer est variable. En effet, en fonction de la taille de la RI et du facteur de compression choisi la quantité de données varie. Il a donc fallu adapter cette méthode proposée par [1]. Dans un premier temps nous récupérerons la totalité des pertes de la RI (Section 2.2). En connaissant la quantité de bits du message à cacher, nous pouvons ensuite moduler le nombre de 1 dans la matrice de quantification (MQ) (Section 2.4). La Figure 1 montre que ce processus a une phase d'optimisation, les modifications de la MQ affectant directement les pertes de données dans la RI. Cette méthode permet d'insérer une quantité de 1 plus optimale

que dans la méthode initiale de [1]. De cette façon le taux de compression est meilleur et la qualité de l'image est plus proche de celle du JPEG original.

TAB. 1 – Matrice de quantification modifiée pour un facteur de qualité de 50

16	11	10	16	1	1	1	1
12	12	14	1	1	1	1	55
14	13	1	1	1	1	69	56
14	1	1	1	1	87	80	62
1	1	1	1	68	109	103	77
1	1	1	64	81	104	113	92
1	1	78	87	103	121	120	101
1	92	95	98	112	100	103	99

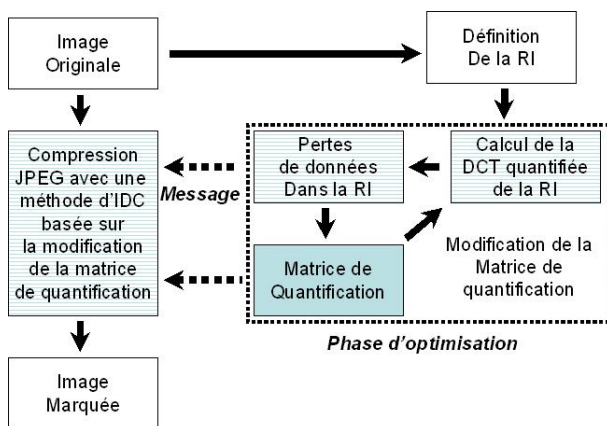


FIG. 1 – La méthode proposée de protection de données.

## 2.2 Récupération des pertes dans la RI

Afin de proposer une méthode de reconstruction d'une RI sans perte, il a fallu s'intéresser aux parties de l'algorithme qui engendrent des pertes et à la façon de les récupérer. Les pertes provenant de la quantification ne posent pas de problème quant à leur récupération. Le problème principal vient du fait que certaines pertes proviennent de l'arrondi de l'IDCT lors de la phase de décompression. Il faut aussi prendre en compte que la méthode d'IDC engendre également des pertes. La Figure 2 montre comment on arrive à calculer la totalité des pertes engendrées par la compression, la décompression et l'IDC sur chaque bloc de la RI. Pour les pertes dues à l'IDC on positionne à zéro tous les bits de poids faible dont on va se servir pour effectuer l'IDC (LSB0 et LSB1). De ce fait, lors de la phase de récupération des données cachées on remplacera les bits à zéro après lecture. Cela nous permet d'avoir une méthode de calcul des pertes qui ne dépend pas du message à marquer et donc d'éviter que l'IDC fasse varier le nombre d'erreurs. Avec cette méthode on s'assure que les pertes sont bien la différence entre la RI de l'image originale et la RI de l'image que l'on va reconstruire.

## 2.3 Codage des pertes

Les pertes sous cette forme ne nous permettent pas d'effectuer l'IDC. Etant donné qu'elles vont être insérées puis récupérées sous la forme d'un flux binaire, il est nécessaire de les mettre en forme pour faciliter leur extraction. La méthode choisie est la méthode de Huffman [4] qui utilise des tables de préfixes pour encadrer les valeurs à coder. Cette méthode comporte un avantage dans notre cas car chaque section de coefficient sera codée avec le même nombre de bits. Ainsi il nous est d'autant plus facile, pendant la phase d'optimisation, d'évaluer la quantité de bits à insérer sans avoir à effectuer le codage de Huffman. Les tables de préfixes utilisées sont celles incluses dans le JPEG.

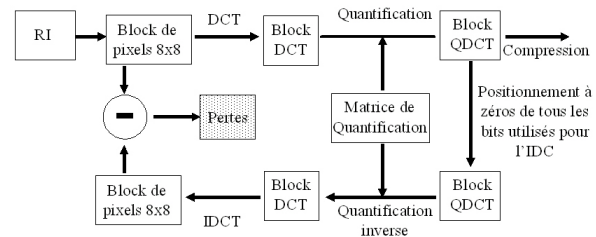


FIG. 2 – Récupération des pertes.

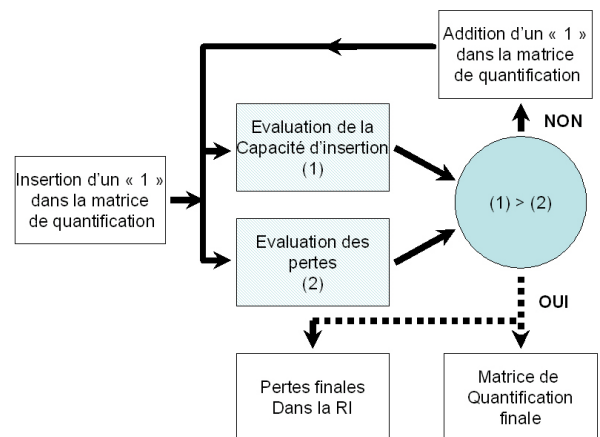


FIG. 3 – Modification de la matrice de quantification.

## 2.4 Modification de la matrice de quantification

La récupération des pertes, Section 2.2, nous permet d'évaluer précisément la quantité de données perdues. Cette quantité va nous permettre de calculer exactement la quantité de 1 à insérer dans la matrice de quantification afin de pouvoir réaliser l'IDC. La Figure 3 montre comment la MQ est modifiée afin d'avoir le nombre optimum de 1 nécessaire à l'IDC. Le fait d'insérer des 1 dans la MQ modifie la quantité de pertes de la RI. En effet, certains coefficients ne seront plus quantifiés. Cette opération est donc réalisée plusieurs fois afin que les pertes de données et le nombre de 1 dans la MQ soient optimaux. Le résultat de cette optimisation nous concède ainsi la MQ qui sera

utilisée pour exécuter la compression JPEG et l'IDC.

## 2.5 Méthode d'IDC

L'IDC va être effectuée sur les coefficients DCT qui seront quantifiés par 1 (Tableau 1) [1]. L'IDC se fera par substitution des deux bits de poids faible LSB0 et LSB1. Il sera effectué dans tous les blocs de l'image même ceux contenus dans la RI, ceci est rendu possible grâce à la mise à zéro des coefficients non quantifiés Section 2.2.

## 2.6 Extraction et reconstruction

Cette section traite de l'extraction des données cachées et de la reconstruction de la RI. La reconstruction de la RI se passe en deux étapes :

- La première étape consiste à récupérer les données cachées, pour cela il suffit de lire les bits de poids faible pour les coefficients dont la position dans le bloc correspond à un 1 dans la matrice de quantification (Celle-ci est contenue dans les entêtes du JPEG). Les premières données reçues sont la position et la taille de la RI. Tous les autres bits récupérés sont placés dans un vecteur binaire qui est ensuite décodé à l'aide de la table de Huffman afin de créer un nouveau vecteur de pertes. A la fin de cette opération nous obtenons une image décompressée (mais ne contenant pas la RI reconstruite), les coordonnées ainsi que la taille de la RI et un vecteur de pertes.
- La deuxième étape est la reconstruction de la RI. A l'aide du vecteur de pertes on reforme des matrices  $8 \times 8$  représentant les pertes dans chaque bloc de la RI. Les pertes récupérées étant des pertes dans le domaine spatial, il suffit de les ajouter aux blocs décompressés correspondant de la RI.

## 3 Résultats expérimentaux

Nous avons appliqué notre méthode de protection sur l'image de Lena en choisissant une RI autour des yeux illustrée Figure 4. La taille de l'image originale est de  $512 \times 512$  pixels et la taille de la RI est  $144 \times 80$  pixels (4.39%). La Figure 5 montre l'image comprimée et marquée avec notre méthode pour un facteur de qualité de 10%. Le Tableau 2 compare les résultats de compression et de qualité entre l'algorithme JPEG et notre méthode pour différents facteurs de qualité. Nous observons que le taux de compression est moins significatif dans notre méthode, ceci est dû aux 1 insérés dans la matrice de quantification. Ces 1 changent également le facteur de qualité que nous avons choisi au début. La différence de qualité entre l'image comprimée avec l'algorithme JPEG et celle comprimée et marquée avec notre méthode, Figure 5, est inférieure à 2.2 dB. Ceci prouve que notre méthode d'insertion de données est invisible à l'oeil humain. Le Tableau 3 nous donne le nombre de 1 insérés dans la matrice de quantification ainsi que le nombre de bits utilisés pour effectuer l'IDC. La Figure 6 montre le résultat obtenu après l'extraction et la

reconstruction de la RI. La Figure 7 montre les différences entre l'image originale et l'image reconstruite Figure 6, la zone noire prouve que la RI n'a subi aucune perte de données. La qualité de la RI après la reconstruction de l'image est identique à la RI de l'image originale (PSNR infini).

TAB. 2 – Comparaison des résultats obtenus entre JPEG et notre méthode.

FQ	10%	30%	50%
PSNR de l'image comprimée JPEG (dB)	29.51	33.03	34.43
Taux de compression du JPEG	30.7	15.7	11
PSNR de l'image comprimée avec notre méthode (dB)	27.39	32	32.56
Taux de compression avec notre méthode	6.1	5.8	5.6



FIG. 4 – Image originale avec sélection d'un RI.

TAB. 3 – Résultats obtenus avec notre méthode d'IDC.

Facteur de qualite (FQ)	10%	30%	50%
Nombre de 1 dans MQ	10	9	8
Bits insérés	72844	63295	59988

## 4 Conclusion

Dans cet article nous avons présenté une méthode permettant de sauvegarder une RI sécurisée sans perte dans une image comprimée avec l'algorithme JPEG. Elle améliore un service qui existe dans JPEG 2000 et mis en application dans l'algorithme JPEG. Elle est basée sur le travail



FIG. 5 – Image comprimée et marquée (FQ = 10).



FIG. 6 – Image avec RI reconstruite.

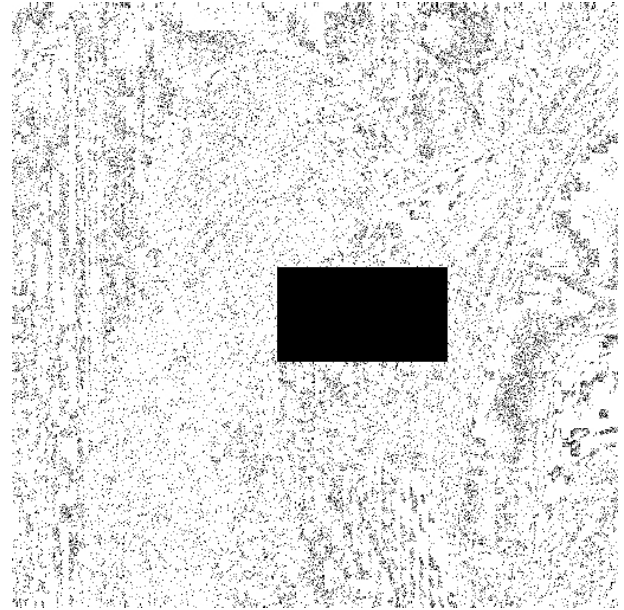


FIG. 7 – Différence entre image originale et image avec RI reconstruite.

de [1]. Nous avons amélioré leur méthode pour y inclure l'information d'une zone déterminée d'une image. Nous étendons actuellement cette méthode à des images couleurs.

## Références

- [1] C.-C. CHANG and T.-S. L.-Z. CHEN CHUNG, *A Steganographic Method Based Upon JPEG and Quantization Table Modification*, *Information Sciences*, vol. 141, pp. 123-138, 2002.
- [2] C. Christopoulos, J. Askelöf M. Larsson, *Efficient Methods for Encoding Regions of Interest in the Upcoming JPEG2000 Still Image Coding Standard*, *IEEE Signal Processing Letters*, Vol. 7, No. 9, pp. 247-249, September 2000.
- [3] S.B. GOKTURK, C. TOMASI, B. GIROD, and C. BEAULIEU, *Medical image compression based on region of interest, with application to colon CT images*, *Engineering in Medicine and Biology Society (EMBS), Proceedings of the 23rd Annual International Conference of the IEEE*, vol. 3, pp. 2453 -2456, 2001.
- [4] Huffman, *A Method for the Construction of Minimum Redundancy Codes*. In *Proceedings IRE*, vol. 40, pp. 1098-1101, 1962.
- [5] W. JIA, X. HE, and Q. LIN, *Echocardiography Sequential Images Compression Based on Region of Interest*, *2nd International Conference on Information Technology for application, ICITA, Harbin, China, Jan. 2004*.
- [6] J. STROM, and P. COSMAN, *Medical Image Compression with Lossless Regions Of Interest*, *Signal Processing*, vol. 59, No.2, pp.155-171, Jun. 1997.
- [7] A. WAKATAMI, *Digital Watermarking for RI Medical Images by Using Compresses Signature Image*, In *WIAMIS'04, Lisboa, Portugal, 2004*.