

# TATOUAGE COULEUR PAR DCT BASÉ SUR LE CONTENU

G. Lo-varco<sup>1</sup>

W. Puech<sup>2</sup>

M. Dumas<sup>1</sup>

<sup>1</sup> Laboratoire CEM2, UMR CNRS 5507, Université Montpellier II

Pl. Gabriel Péri, 30021 Nîmes Cedex 1, France

<sup>2</sup> Laboratoire LIRMM, UMR CNRS 5506, Université Montpellier II

161, rue Ada, 34392 Montpellier Cedex 5, France

lovarco@cem2.univ-montp2.fr puech@lirmm.fr dumas@univ-montp2.fr

## Résumé

*Dans cet article, nous présentons une méthode de tatouage d'image couleur basée sur le contenu de l'image. Cette approche rend notre méthode robuste aux manipulations géométriques tels que les translations ou le fenêtrage ainsi qu'à la compression JPEG couleur. Pour acquérir cette robustesse, il faut tout d'abord extraire de l'image des composantes géométriquement invariantes. Le tatouage vient alors se synchroniser à l'image en s'appuyant sur ses invariants. La technique de tatouage s'appuie elle sur les coefficients de la DCT non arrondis.*

## Mots clefs

Tatouage basé couleur, tatouage basé contenu, coefficients de la DCT non arrondis.

## 1 Introduction

De nos jours, la masse d'informations multimédias en transit sur les réseaux ne cessent d'augmenter. Les applications sont nombreuses et variées : il y a l'imagerie médicale ou encore satellitaire ainsi que la télésurveillance ou encore la sécurité routière. Aussi, le transfert de données ainsi que sa sécurité deviennent des centres d'intérêt prioritaires. Or durant le transfert des images, l'intégrité des données n'est pas vraiment assurée [1]. Le tatouage peut être une réponse à ces problèmes de sécurité. Concernant les applications visées, l'objectif du tatouage est d'insérer un message de manière invisible dans une image couleur. Dans cet article, nous présentons une méthode de tatouage couleur [2] basée sur les coefficients DCT [3, 4, 5] non arrondis. L'originalité de cette nouvelle technique est que l'insertion de l'information est ancrée au contenu de l'image.

Dans la section 2, nous présentons l'extraction des régions d'intérêt de l'image ainsi que l'obtention de leurs caractéristiques. Nous décrivons ensuite la synchronisation du tatouage avec les régions. La section 3 montre les différentes étapes du tatouage DCT couleur en présentant tout d'abord les changements d'espaces couleurs, puis l'insertion sur les coefficients DCT non arrondis. Dans la sec-

tion 4, nous appliquons notre méthode à une image réelle et nous analysons sa robustesse face au fenêtrage.

## 2 Extraction des caractéristiques des régions

L'objectif de cette section consiste à extraire des caractéristiques des zones d'intérêt de l'image couleur pour synchroniser le message à insérer avec le contenu de l'image. L'étape préliminaire de cette extraction consiste à définir différentes régions de l'image couleur. L'idée est de diviser l'image en un nombre limité de régions convexes de taille homogène. Chaque région obtenue après étiquetage contient des pixels connexes de couleur relativement uniforme [6]. Des caractéristiques géométriquement invariantes des régions sont alors calculées.

### 2.1 Calcul des caractéristiques des régions

Pour avoir une idée de l'importance de la région par rapport à la taille de l'image, et ainsi pouvoir être robuste au changement d'échelle, nous évaluons la taille de la région  $T(r)$  en comptant le nombre de pixels appartenant à la région  $r$ , puis  $T_p(r)$  correspondant à  $T(r)$  pondérée par la taille de l'image  $T_I = M \times N$ . Nous avons donc  $T_p(r) = \frac{T(r)}{T_I} = \frac{T(r)}{M \times N}$ . Nous pouvons alors définir les caractéristiques invariantes de l'orientation avec des moments de la forme de la région [7]. Les moments de degré 1, notés  $\mu_i(r)$  et  $\mu_j(r)$ , coordonnées du centre de gravité de la région  $r$ , définissent un point d'ancrage. Ce point devient alors l'origine du référentiel utilisé pour le tatouage de cette région. Pour les calculer, nous avons :

$$\mu_i(r) = \frac{1}{T(r)} \sum_{k=0}^{T(r)} i(k) \text{ et } \mu_j(r) = \frac{1}{T(r)} \sum_{k=0}^{T(r)} j(k), \quad (1)$$

où  $i(k)$  et  $j(k)$  sont respectivement les coordonnées verticale et horizontale du pixel  $k$ . Les moments du second degré sont alors calculés, et nous obtenons les variances horizontale  $V_x(r)$  et verticale  $V_y(r)$ , ainsi que la covariance

$V_{xy}(r) :$

$$\begin{cases} V_x(r) = \frac{1}{T(r)} \sum_{k=0}^{T(r)} (j(k) - \mu_j(r))^2, \\ V_y(r) = \frac{1}{T(r)} \sum_{k=0}^{T(r)} (i(k) - \mu_i(r))^2, \\ V_{xy}(r) = \frac{1}{T(r)} \sum_{k=0}^{T(r)} [i(k) - \mu_i(r)][j(k) - \mu_j(r)]. \end{cases} \quad (2)$$

Ces coefficients forment une matrice de covariance  $C_0(r)$  qui est propre à chaque région  $r$  détectée dans l'image. Une analyse des composantes principales de cette matrice aboutit à l'obtention de 2 valeurs propres  $\lambda_1(r)$  et  $\lambda_2(r)$  ainsi qu'à leurs vecteurs propres associés  $u_1(r)$  et  $u_2(r)$  [8]. Ces couples  $\{\lambda, u\}$  définissent les axes majeur et mineur de la région. Ces axes, présentés Figure 1.a, associés au centre de gravité  $G_r(\mu_i(r), \mu_j(r))$ , forment un repère propre à la région et apte à accueillir le tatouage.

## 2.2 Synchronisation du tatouage avec les régions

L'étape suivante consiste à synchroniser l'insertion d'un message dans chacune des régions de l'image. La technique de tatouage, présentée Section 3.2, est basée sur des blocs de  $N$  pixels. Pour rendre cette insertion invariante aux modifications géométriques, nous cherchons à engendrer des blocs qui se modifieront de la même manière que la région concernée, tout en conservant l'information tatouée. Pour chacune des régions, les blocs sont créés strictement à l'intérieur des régions, à partir du centre de gravité et suivant leurs 2 axes, majeur et mineur. Par conséquent nous n'utilisons pas les pixels appartenant aux frontières des régions. De cette manière, lors des rotations par exemple, les pixels appartenant à nos blocs de tatouage ne peuvent changer de région et donc d'étiquette. Dans ce cas, l'information tatouée dans les blocs ne peut être perdue. De plus, la création de ces blocs doit suivre un ordre établi sinon aucune reconstruction du message ne sera possible. Nous avons donc choisi un développement de type excentrique, qui s'appuie sur la forme des régions, par l'intermédiaire des valeurs propres, illustrée Figures 1.a et b.

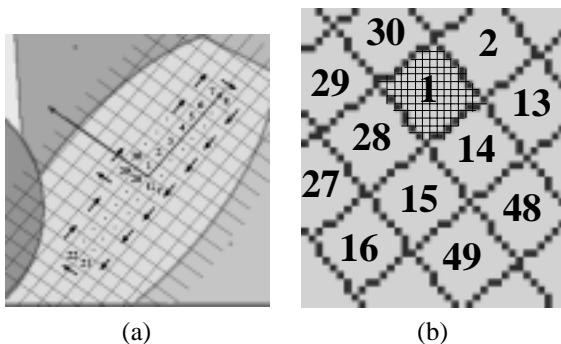


Figure 1 – a) Ordre de création des blocs utilisés pour le tatouage, b) Agrandissement des blocs créés pour le tatouage.

## 3 Tatouage DCT couleur par régions

### 3.1 Décomposition dans l'espace Y, Cr, Cb

Les images couleurs sont généralement construites à partir de trois plans correspondant aux composantes rouge, verte et bleue (R, G, B). Par conséquent, lorsqu'on veut insérer un message dans une image couleur, nous disposons de ces trois plans qui n'ont pas la même fonction selon l'espace de couleur considéré. Dans cette section, nous allons présenter comment nous mettons à profit les informations couleurs pour appuyer notre tatouage afin de résister à la compression JPEG couleur.

A partir de l'image couleur décomposée en trois plans RGB et avec une matrice de changement de base standard nous obtenons trois autres plans notés Y, Cr et Cb. La composante Y correspond à l'information luminance,  $Y = 0.299R + 0.587G + 0.114B$ , et les composantes Cr et Cb aux informations chrominances.

Le message est alors inséré sur chacun de ces plans avec la méthode de tatouage DCT par régions décrite dans la section 3.2. Nous obtenons trois nouveaux plans  $Y'$ ,  $Cr'$  et  $Cb'$  où chacune des régions retenues est tatouée. A partir de ces plans tatoués, nous reconstruisons trois nouvelles composantes tatouées  $R'$ ,  $G'$  et  $B'$  afin de recomposer l'image couleur tatouée. Toutes les étapes des changements d'espaces couleurs sont décrites dans le schéma illustré Figure 2.

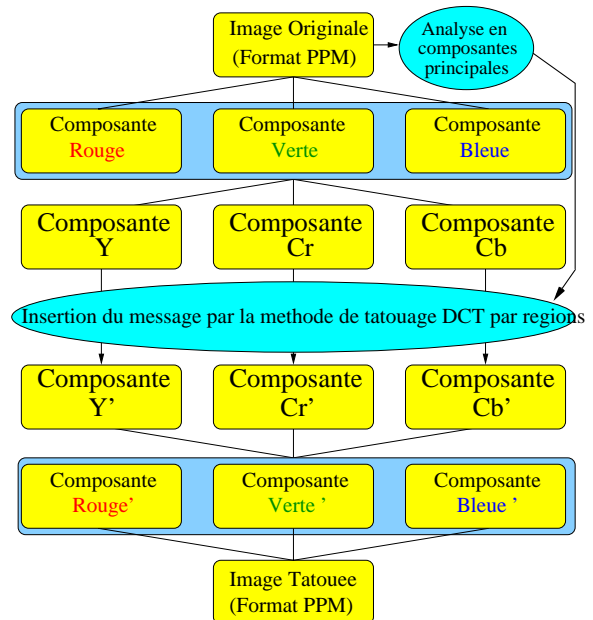


Figure 2 – Décomposition et tatouage dans l'espace Y, Cr, Cb.

### 3.2 Méthode de tatouage s'appuyant sur les coefficients DCT non arrondis

Nous proposons une méthode de tatouage fréquentiel basé sur la Transformée en Cosinus Discrète (DCT). L'objectif

est d'insérer dans chaque région de l'image un message  $M$  constitué de  $m$  bits  $b_k$ ,  $k \in [1 \dots, m]$ . Pour cela, on calcule  $F_k(0, 0)$ , la composante DC de la DCT pour des blocs de pixels obtenus dans chacune des régions de l'image.

$$F_k(0, 0) = \frac{1}{n} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} p(i, j) = \frac{1}{n} \sum_{j=0}^{N-1} p(j). \quad (3)$$

Ensuite,  $F_k(0, 0)$  est quantifiée et nous obtenons une valeur réelle non arrondie, notée  $F'_k(0, 0)$ . Celle-ci est alors utilisée afin d'augmenter la robustesse face à la compression.

Le but de la méthode est de ramener cette valeur  $F'_k(0, 0)$  à une valeur  $F''_k(0, 0)$  plus stable face aux modifications dues à la quantification des coefficients et adaptée à la valeur du bit  $b_k$  à insérer. Pour cela on calcule  $R_{F'_k(0,0)}$ , le reste non arrondi de la division par 2 de  $F'_k(0, 0)$ , et  $R_{F''_k(0,0)}$  qui correspond au reste idéal au niveau de la stabilité, soit  $b_k + 0.5$ . On note  $d_k$  la différence entre ces valeurs d'où :

$$d_k = b_k + 0.5 - R_{F'_k(0,0)} \quad \text{et} \quad N_{d_k} = [|d_k| \times N], \quad (4)$$

où  $N$  est le nombre de pixels d'un bloc et  $N_{d_k}$  le nombre de pixels à modifier pour passer de  $F'_k(0, 0)$  à  $F''_k(0, 0)$ . Pour calculer l'intensité d'un pixel modifié, on applique la formule suivante :

$$p'(j) = p(j) + \text{signe}(d_k), \quad (5)$$

où  $p'(j)$  désigne le niveau de gris du pixel modifié  $j$  du bloc  $k$ . Finalement, nous obtenons :

$$F''_k(0, 0) = \frac{1}{n \times q(0, 0)} \left[ \sum_{j=0}^{N_{d_k}-1} p'(j) + \sum_{j=N_{d_k}}^{N-1} p(j) \right]. \quad (6)$$

Le bit  $b_k$  du message inséré correspond alors au bit de poids le plus faible (LSB) de la partie entière de  $F''_k(0, 0)$ .

## 4 Résultats

### 4.1 Détection du message dans une image réelle

Dans l'image originale illustrée Figure 3.a, nous avons inséré avec notre méthode de tatouage un message différent dans chaque région. La liste des messages insérés, de longueur 8 caractères, dans chaque région est présentée dans le tableau 1. Pour insérer ou détecter nos messages, il faut tout d'abord détecter les régions d'intérêt de l'image. Nous obtenons alors une image étiquetage, illustrée Figure 3.b, dans laquelle nous construisons les blocs qui vont être utilisés pour l'insertion des messages. Comme chacun des plans a été tatoué séparément, la détection va se faire sur les trois plans. En tatouant trois fois, nous augmentons la redondance du message et donc les chances de bien le détecter. La lecture des blocs se fait alors comme leur écriture de manière excentrique. Les messages lus par extraction, présentés tableau 1, correspondent aux messages

insérés. Afin de visualiser les blocs tatoués correspondant aux différents messages insérés, nous avons construit l'image différence entre l'image originale et l'image tatouée, présentée Figure 3.c.

Pour évaluer l'impact de l'insertion de notre message sur l'image, nous avons calculé le PSNR entre les plans couleurs originaux et les plans couleurs tatoués. Les résultats sont regroupés dans le tableau 2. En analysant ces résultats, nous constatons que le PSNR total entre l'image couleur originale et l'image tatouée est de 44, 42 dB, ce qui nous autorise à dire que notre tatouage a un faible impact sur l'image.

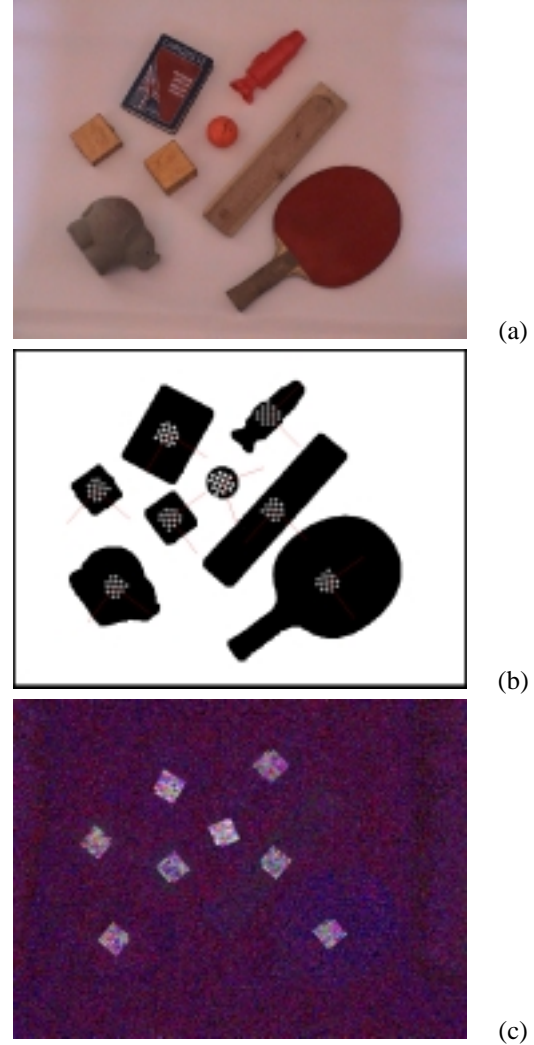


Figure 3 – a) Image originale, b) Extraction des régions et construction des blocs, c) Image différence entre l'image originale et l'image tatouée.

### 4.2 Analyse de la robustesse face au fenêtrage

Intéressons nous maintenant à la robustesse de notre méthode face au fenêtrage. Nous présentons une image Figure 4.a qui est le résultat d'un fenêtrage opéré sur l'image tatouée. L'image pré-étiquetage, illustrée Figure

Région	Message inséré	Message détecté sur le plan Y'	Message détecté sur le plan Cr'
1	Jouet1	Jouet1	Jouet1
2	Livre0	Livre0	Livre0
3	Etui00	Etui00	Etui00
4	Cube01	Cube01	Cube01
5	Golf00	Golf00	Golf00
6	Cube02	Cube02	Cube02
7	Raquette	Raquette	Raquette
8	Jouet2	Jouet2	Jouet2

Tableau 1 – Messages détectés sur les plans Y tatoué (Y') et Cr tatoué (cr').

Différence entre image originale et image tatouée	Plan Y	Plan Cr	Plan Cb
PSNR (en dB)	44,149	42,300	49,546

Tableau 2 – PSNR calculé pour les images différences de chaque plan.

4.b, nécessaire pour la détection des blocs, laisse apparaître six régions. Il s'agit de trois régions entières déjà présentes dans l'image originale ainsi que trois morceaux de régions. Les résultats obtenus après détection du tatouage sont présentés dans le tableau 3. Les messages insérés dans les régions entières sont donc récupérés. Le message faux correspond en fait à une fausse détection des blocs car la région correspondante a été tronquée, comme illustré Figure 4.b. Ainsi, si nous insérons dans chaque région de l'image le même message, il suffit d'une seule région entière à la détection pour récupérer notre message.

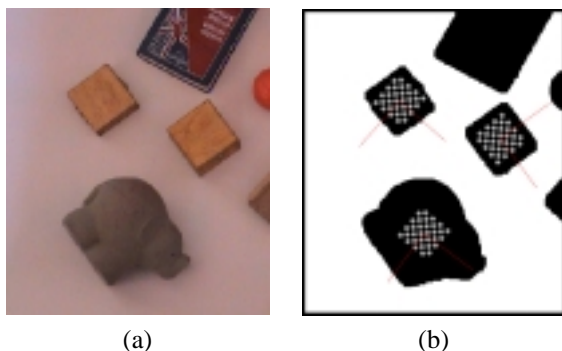


Figure 4 – a) Fenêtrage de l'image originale, b) Reconnaissance des régions dans la fenêtrage de l'image originale.

## 5 Conclusion

Nous présentons ici une technique de tatouage fréquentiel d'image couleur basée sur le contenu. Pour cela une analyse de l'image est effectuée pour obtenir des repères propres aux zones d'intérêt. Ces repères vont servir ensuite à synchroniser notre tatouage avec l'image. Le ta-

Région	Message inséré	Message détecté sur le plan Y'	Message détecté sur le plan Cr'
1	Jouet1	—	—
2	Livre0	GB9	WøÇí
3	Etui00	—	—
4	Cube01	Cube01	Cube01
5	Golf00	—	—
6	Cube02	Cube02	Cube02
7	Raquette	—	—
8	Jouet2	Jouet2	Jouet2

Tableau 3 – Messages détectés sur les plans Y tatoué (Y') et Cr tatoué (cr').

touage quant à lui, se fait de manière inductive, par modification des coefficients non arrondis de la DCT. Comme les trois plans couleur sont tatoués, les chances de détection du message sont augmentées. Le fait d'ancrer le message à l'image rend notre méthode robuste aux déformations géométriques telles les translations ou encore le fenêtrage. Les prochains axes de recherche concernent la robustesse de notre méthode face au changement d'échelle ou encore à la compression.

## Références

- [1] F. Deguillaume, S. Voloshynovskiy, et T. Pun. Hybrid robust watermarking resistant against copy attack. Dans *EUSIPCO'02, Toulouse, France, 2002*.
- [2] G. Chareyron et A. Tremeau. Watermarking of color images based on a multi-layer process. Dans *CGIV'02, Poitiers, France, pages 77–80, 2002*.
- [3] I.J. Cox, J. Killian, T. Leighton, et T. Shamoon. Secure spread spectrum watermarking for multimedia. *IEEE Trans. on Image Processing*, 6(12):1673–1687, 1997.
- [4] A.G. Bors et I. Pitas. Image watermarking using block site selection and DCT domain constraints. *Optics Express*, 3(12):512–522, 1997.
- [5] E. Koch et J. Zhao. Towards robust and hidden image copyright labeling. Dans *IEEE Workshop on Nonlinear Signal and Image Processing, Neos Marmaras, Greece, pages 452–455, 1995*.
- [6] P. Lamaty et D. Demigny. Opérateur matériel d'étiquetage de régions temps réel et flot de données. Dans *Proc. of GRETSI'99, Vannes, France, pages 311–314, 1999*.
- [7] A. Nikolaidis et I. Pitas. A novel technique for region based image watermarking. *IEEE Transactions on Image Processing*, vol. 10, (11):1726–1740, nov 2001.
- [8] P. Bas et B. Macq. A new video-object watermarking scheme robust to object manipulation. Dans *Proc. of ICIP'01, Tesseloniki, Greece, pages "526–529", 2001*.