

# Élimination des artefacts pour un codage JPEG optimisé

Achraf MAKHLOUFI<sup>1</sup>, Azza OULED ZAID<sup>2</sup>, Christian OLIVIER<sup>2</sup>, Mohamed Salim BOUHLEL<sup>1</sup>

<sup>1</sup>Ecole Nationale d'Ingenieur de Sfax,  
Route de la Soukra, km 4, Sfax 3038, Tunisie

<sup>2</sup>Laboratoire Signal Image et Communication-UMR CNRS 6615,  
SP2MI, Teleport 2, Boulevard M. et P. Curie, 96962 Futuroscope Cedex, France  
medsalim.bouhleb@enis.rnu.tn,  
ouled@sic.sp2mi.univ-poitiers.fr, olivier@sic.sp2mi.univ-poitiers.fr

**Résumé** – Dans notre travail, nous avons traité le problème des effets de blocs qui apparaissent sur l'image reconstruite dans le cas d'une compression JPEG, en particulier à très bas débits. L'originalité consiste à appliquer une opération de lissage non linéaire sur l'image reconstruite basée sur les statistiques de l'image et d'utiliser un codeur JPEG permettant d'optimiser le compromis débit/distorsion avec un contrôle du débit. Les résultats de simulation montrent une amélioration au niveau de la qualité en terme de PSNR de l'ordre (0.2-0.8 dB) par rapport au même algorithme de compression sans lissage et une réduction importante des effets de blocs du point de vue visuel.

**Abstract** – In this paper, we address the classical problem of removing the blocking artefacts of block DCT compressed images, in the case of optimized JPEG compression with low bit rates. The originality consists on one hand with introducing an algorithm that removes blockiness by performing a simple nonlinear smoothing based on the pixel statistics. On the other hand with using an optimized JPEG coder that take into account the rate/distortion tradeoff. Experimental results show that this method achieves objective (PSNR) gains of about (0.2-0.8 dB) and significantly improves visual quality.

## 1 Introduction

L'algorithme de codage JPEG [1] est l'un des algorithmes les plus connus dans le domaine de compression d'image. Élaborée il y a maintenant plus de vingt ans, la norme JPEG avec perte, offre un bon rapport qualité/taille de l'image disponible, compte tenu d'autres facteurs contraignants que sont les vitesses de transmission et de décompression. Cependant, avec un codage JPEG, une trop forte compression crée des artefacts très visibles, ce qui a donné lieu à de nombreux travaux dont le but est d'améliorer les performances de l'algorithme de compression, surtout dans le cas de bas et moyens débits. En effet, les travaux qui ont été réalisés à ce propos interviennent au niveau de l'étape de quantification et consistent à optimiser le choix du quantificateur qui donne la meilleure qualité pour un débit alloué, ou inversement à atteindre la qualité désirée avec un débit minimum.

Dans ce papier, nous donnons une version modifiée du schéma de compression JPEG, performant au sens du compromis débit/distorsion [2]. Notre contribution consiste à incorporer un processus de seuillage global simple et rapide des coefficients DCT avant de passer à la phase de quantification. Il s'agit alors de déterminer une table  $8 \times 8$  de paramètres de seuillage à appliquer sur chaque bloc  $8 \times 8$  de l'image transformée, puis de générer la table de quantification Q adéquate en tenant compte du problème d'allocation binaire. Une fois quantifiés, les coefficients transformés seront par la suite stockés sous forme d'une chaîne de bits en utilisant un codeur entropique à longueur variable de type Huffman optimisé. Dans notre système de codage, nous avons utilisée une technique d'allocation binaire connue sous le nom RD-OPT [3]. Elle a l'avantage d'estimer les tables de débit/distorsion en fonction des statistiques

des coefficients transformés sans avoir besoin de passer par tout le cycle de codage/décodage.

A la reconstruction, nous avons intégré une opération de lissage non linéaire basée sur la loi de Weber, permettant d'atténuer les effets de blocs ou les artefacts, en particulier dans le cas de bas et moyens débits.

L'organisation de ce chapitre est la suivante : dans la section 2, nous présentons les modifications que nous avons apportées au codeur JPEG notamment l'incorporation d'un processus de seuillage global et l'utilisation d'une allocation binaire presque optimale basée sur la recherche exhaustive. Ensuite, dans la section 3, nous évoquons le principe de lissage non linéaire que nous avons intégrée au niveau du décodeur. Finalement, les résultats expérimentaux obtenus à partir de différents tests comparatifs seront commentés dans la section 4.

## 2 Raffinement du quantificateur JPEG

D'une manière générale, dans le cas du codage JPEG, les éléments de la table de quantification sont les seuls paramètres réglables pour faire varier le débit ou la qualité de l'image reconstruite. Ainsi, la table de quantification contrôle l'évolution du compromis débit/distorsion. Les travaux qui ont été rapportés dans la littérature, concernant l'optimisation du codage JPEG, sont tous basés sur le choix du meilleur quantificateur. Dans le cadre de notre travail, nous avons proposé une technique de seuillage des coefficients DCT de l'image transformée couplée à une quantification presque optimale au sens du compromis débit/distorsion [4]. Notre idée consiste à utiliser un processus de seuillage global basé sur la modélisation Laplacienne des coefficients. Les coefficients seuillés sont par la suite

quantifiés par bloc en utilisant une table de quantification générée selon l'algorithme RD-OPT.

## 2.1 Seuillage global

Le seuillage global utilise une table  $T$  de 64 valeurs  $n$  qui s'applique sur chaque bloc DCT. Les éléments de  $T$  sont les valeurs des seuils à déterminer pour chaque localisation fréquentielle. Contrairement au seuillage local, le seuillage d'un coefficient DCT donné est indépendant du bloc auquel il appartient. Pour un coefficient  $F[n]$  d'indice  $n$ , le résultat de l'opération de seuillage couplée à celle de la quantification à travers tous les blocs DCT de l'image transformée est donné par :

$$\begin{aligned} \text{si } |F[n]| \geq T[n] &\Rightarrow F_{Q,T}[n] = \left\lfloor \frac{F[n]}{Q[n]} + 0.5 \right\rfloor \\ \text{sinon} &F_{Q,T}[n] = 0 \end{aligned} \quad (1)$$

Notons que la valeur de  $T[0]$  est toujours nulle, car le coefficient (coefficient DC) contient l'information la plus pertinente, et doit donc toujours être conservé. De même, soulignons que la table  $T$  ne sera pas incluse dans l'image compressée (flux binaire à transmettre), car seule la table de quantification est nécessaire pour reconstruire l'image.

## 2.2 Technique proposée pour la sélection des valeurs des seuils

Notre approche de seuillage qui consiste à générer la table de seuillage en tenant compte de la densité de probabilité a priori  $\mu_n$ , pour  $n \in \{1, \dots, 63\}$ .

Il s'agit d'une méthode adaptative, permettant le réglage automatique du paramètre de seuillage suivant les localisations fréquentielles, à partir d'une simple expression analytique.

Étant donné que les coefficients ne représentent pas tous la même quantité d'information, la réduction de la dynamique, voir la mise à zéro d'un coefficient correspondant aux hautes fréquences, ne provoquera pas autant de dégradation que celle d'un coefficient correspondant aux basses fréquences. Ainsi, connaissant les fonctions de répartition des coefficients, nous pouvons définir les seuils au-dessus desquels ils ont une probabilité faible d'apparaître. Ces coefficients ne doivent pas être rendus nuls sous peine de dégradation visible puisqu'ils représentent souvent un détail particulier sur l'image originale.

Nous pouvons considérer qu'un coefficient d'indice  $n$  a peu de chance d'apparaître si sa valeur est supérieure à une valeur limite  $L_n$ , positive non nulle, définie par :

$$\int_{-\infty}^{L_n} \mu_n(x) dx = 0.95. \quad (2)$$

La meilleure approximation de  $\mu_n$  est une loi Laplacienne définie par [5] :

$$\mu_n(x) = \frac{\alpha}{2} e^{-\alpha|x|} \quad (3)$$

où  $\alpha_n$  est le paramètre caractéristique du modèle Laplacien, et qui peut être défini en fonction de l'écart-type de la distribution réelle. L'expression analytique donnant la valeur du seuil pour chacune des localisations fréquentielles  $n$  est la suivante :

$$T[n] = \frac{F_e \sqrt{2}}{\sigma_n \ln 10} \quad (4)$$

où  $\sigma_n$  est l'écart type de la distribution réelle des coefficients AC indexés par  $n$  et  $F_e$  un facteur d'échelle fixé a priori.

Ainsi à partir des 63 modalités Laplaciennes à travers toute l'image transformée, nous définissons la valeur de  $T[n]$  pour chaque coefficient et chaque modalité.

## 3 Élimination des effets de blocs

Comme nous l'avons dit et malgré les améliorations apportées de la quantification par la méthode du § 2, on ne peut empêcher un effet de bloc très visible dès que les débits sont faibles.

On propose dans cette partie un algorithme rapide et simple pour éliminer l'effet de bloc en effectuant un lissage non linéaire sur les pixels de l'image. L'algorithme qu'on propose se base sur deux principes :

1. Lissage des discontinuités artificielles (dues au bruit de quantification) entre les blocs.
2. Les paramètres de lissage sont basés sur des critères psychovisuels.

Pour distinguer les discontinuités artificielles du contour de l'image originale nous estimons le bruit de quantification par le maximum de vraisemblance [6]. On effectue ensuite une opération de lissage non linéaire sur toute l'image selon la loi de Weber.

### 3.1 Calcul de l'erreur de quantification

Pour calculer l'erreur de quantification il faut disposer des coefficients DCT avant quantification et après déquantification. Au niveau du décodeur on n'a pas accès aux valeurs des coefficients DCT avant quantification; on propose alors de faire des estimations à partir des coefficients DCT déquantifiés. Étant donné que les 63 distributions des coefficients AC à travers l'image transformée sont approchées par des loi Laplaciennes (voir equation 3). Les valeurs des coefficients DCT de l'image transformée sont alors estimés par une expression de vraisemblances dans un bloc  $N \times N$  comme suit :

$$\hat{C}_{uv} = \frac{\int_{L_{uv}}^{R_{uv}} y p_{uv}(y) dy}{\int_{L_{uv}}^{R_{uv}} p_{uv}(y) dy}, u, v = 1, \dots, N \quad (5)$$

avec  $p_{uv}(y)$  la loi de probabilité de distribution du coefficient DCT avant quantification (on utilise, ici, les coefficients DCT déquantifiés).  $L_{uv}$  et  $R_{uv}$  représentent respectivement, la limite inférieure et supérieure du pas de quantification.

L'erreur quadratique de quantification de chaque coefficient est donnée par :

$$E_{uv} = \frac{\int_{L_{uv}}^{R_{uv}} (y - \hat{C}_{uv})^2 p_{uv}(y) dy}{\int_{L_{uv}}^{R_{uv}} p_{uv}(y) dy}, u, v = 1, \dots, N \quad (6)$$

### 3.2 Calcul du seuil

En tenant compte de la linéarité de la transformée DCT, l'erreur de quantification notée  $T_e$  de chaque bloc  $N \times N$  dans le domaine spatial est estimée à partir de la somme des erreurs de quantification  $E_{uv}$  dans le domaine fréquentiel par :

$$T_e = \frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} E_{uv} \quad (7)$$

calcul ensuite le seuil  $t = 2\sqrt{T_e}$ .

### 3.3 Élimination des discontinuités artificielles

Afin d'éliminer les effets de blocs qui peuvent affecter l'image reconstruite, nous nous proposons de déterminer les discontinuités entre les pixels situés sur les bords des blocs adjacents en fonction d'un paramètre de perception visuel  $S$ , dont la valeur est approchée selon la loi de Weber [7]. Puis on effectue un lissage non linéaire sur toute l'image.

Pour déterminer la discontinuité entre les bords des blocs IDCT (la DCT inverse) adjacents, on prend la différence entre les coefficients qui constituent la limite d'un bloc et ceux du bloc adjacent.

Pour chaque colonne d'indice  $i = 8 \times k, k \in \mathbb{N}$  de l'image que nous désignons par  $X_{:,i}$ , on lui associe le vecteur  $dcol$  qui représente la différence entre ces valeurs et celles de la colonne qui lui est adjacente, soit  $X_{:,i+1}$ . De la même manière, on associe à chacune des lignes  $Y_{j,:}$ , d'indice  $j, j = 8 \times k, k \in \mathbb{N}$ , le vecteur  $dlig$  (vecteur ligne) qui lui correspond.

A partir des vecteurs  $dcol$  et  $dlig$ , l'opération de lissage est alors appliquée comme suit :

Un élément appartenant à l'un de ces deux vecteurs est considéré comme une discontinuité artificielle si sa valeur est inférieure à la valeur du seuil  $t$  (voir l'équation 7). Afin d'atténuer cette discontinuité et pour quelle soit imperceptible à l'œil nu, on remplace la valeur de  $x, x \in X_{:,i} \cup Y_{j,:}$  par  $a \times d$ , tel que :

$$a = (t - y)/2t \quad \text{et} \quad d = x - y$$

où  $y$  est le coefficient adjacent à  $x$ .

Les nouvelles valeurs de  $x$  et  $y$  sont ainsi, données par :

$$x' = x - a \times d \quad \text{et} \quad y' = y + a \times d$$

Ces opérations de lissage, appliquées sur les coefficients localisés dans les bords des blocs IDCT, peuvent à nouveau engendrer une discontinuité entre ces coefficients et ceux qui leurs sont immédiatement adjacents. Ce problème peut être résolu en remplaçant le coefficient immédiatement adjacent au coefficient modifié (ayant subi une opération de lissage) par la moyenne entre sa valeur et celle du pixel modifié. Ce processus est répété sur tous les coefficients jusqu'au centre du bloc.

## 4 Résultats de simulation

De façon à évaluer les performances de notre méthode de lissage, d'un point de vue visuel, nous avons effectué des tests comparatifs sur différentes images représentées en niveaux de gris, après une compression du type JPEG optimisé et une décompression avec ou sans lissage. Dans les figures 1b et 2b, nous donnons respectivement les images Peppers et Albert, reconstruites après une compression à 0.15 bpp et une décompression sans lissage. Les figures 1c et 2c, donnent les mêmes images reconstruites après une compression au même débit et une décompression avec lissage. A partir de ces images, la première constatation qui s'impose est l'atténuation importante des effets de blocs entre chacune des deux images reconstruites avec ou sans lissage.

Afin d'étudier l'efficacité du processus de lissage dans l'amélioration de la qualité de reconstruction objective en terme de

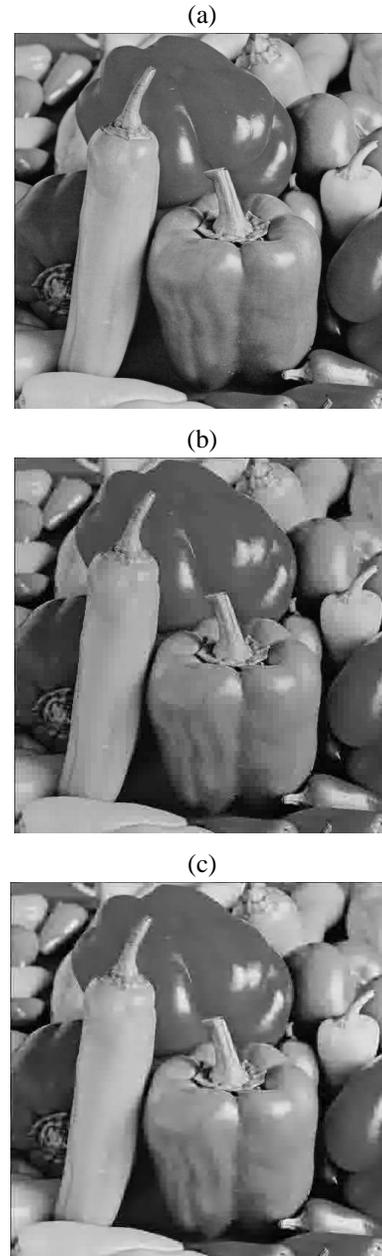


FIG. 1: (a) Image originale Peppers; (b) reconstruite sans lissage après compression à 0.15 bpp; (c) reconstruite avec lissage après compression à 0.15 bpp.

PSNR, le tableau 1 donne les valeurs du PSNR en fonction du débit pour les deux images citées ci-dessus. D'après ce tableau, nous constatons que pour des moyens et hauts débits, Il n'y a pas d'amélioration du point de vue qualité de reconstruction. Bien en contraire nous obtenons des valeurs de PSNR inférieures à celles obtenues à partir d'une reconstruction sans lissage.

Ceci est du au fait que l'algorithme de compression utilisé est déjà optimal dans le cas de moyens et hauts débits, et que l'utilisation d'une opération de lissage qui consiste à modifier certaines valeurs reconstruites peut générer un bruit supplémentaire. Par contre, la différence de qualité de reconstruction est importante dans les bas débits, ce qui est le but de l'étude. Notons que le codeur JPEG ne permet pas de descendre à de très faibles valeurs de débit (en dessous de 0.15 bpp), contrairement

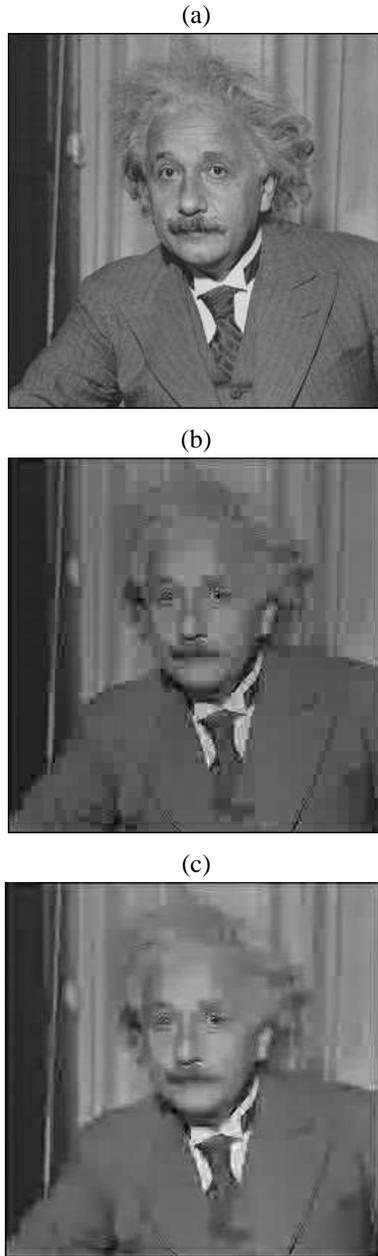


FIG. 2: (a) Image originale Albert; (b) reconstruite sans lissage après compression à 0.15 bpp; (c) reconstruite avec lissage après compression à 0.15 bpp.

TAB. 1: Tableau comparatifs des résultats obtenus en PSNR (dB) Pour deux images tests : Peppers et Alberts.

Images	Débit (bpp)	Décompression sans lissage	Décompression avec lissage
Peppers	0.15	29.7	30.03
	0.2	30.56	30.97
	0.3	32.43	32.17
	0.4	33.65	32.89
Albert	0.15	27.54	27.89
	0.2	28.08	28.95
	0.3	30.54	30.36
	0.4	31.83	31.26

au codeur JPEG2000.

## 5 Conclusion

Le présent article a présenté la technique de décompression que nous avons proposé dans le cas d'une compression JPEG optimisé où nous faisons appel à un seuillage global des coefficients transformés DCT. Dans notre schéma de décompression, nous faisons appel à une opération de lissage non linéaire afin d'atténuer les effets de bloc qui peuvent affecter l'image reconstruite dans le cas d'une compression à très bas débits. Par exemple, à 0.15-0.2 bpp, nous constatons un gain de 0.3-0.4 dB. Néanmoins dans le cas d'une compression à moyens ou hauts débits, l'utilisation d'un lissage s'avère inutile vu que l'algorithme de compression utilisé est déjà optimisé pour des taux de compression élevés.

## Références

- [1] W. B. Pennebaker et J. L. Mitchell. *JPEG Still Image Data Compression Standard*. Van Nostrand Reinhold, New York, December 1992.
- [2] M. Crouse et K. Ramchandran. *JPEG optimization using an entropy-constrained quantization framework*. Proceedings of Data Compression Conference, pp. 342-351, 1995.
- [3] V. Ratnakar et M. Livny. *An Efficient Algorithm For Optimizing DCT Quantization*. IEEE Transaction on Image Processing, vol. 9, n°2, pp. 267-270, février, 2000.
- [4] A. Ouled Zaid. *Amélioration des performances des systèmes de compression JPEG et JPEG2000*. Thèse de doctorat, Université de Poitiers, décembre 2002.
- [5] R. C. Reininger et J. D. Gibson. *Distributions of the Two-Dimensional DCT Coefficients for Images*. IEEE Transaction on Communications, vol. 31, n°6, pp. 835-839, juin 1983.
- [6] Y. Yang, N. Galatsanos et A. Katsaggelos. *Projection-based spatially adaptive reconstruction of block-transform compressed images*. IEEE Transaction on Image Processing, vol. 4, pp. 896-908, july 1995.
- [7] A. Jain. *Fundamentals of Digital Image Processing*. New York: Van Nostrand Reinhold, first ed., 1993.