

La Méthode LAR pour le Codage Basé Région des Images Couleur

Olivier DEFORGES, Laurent BEDAT, Joseph RONSIN

INSA de Rennes, laboratoire URER, 20 avenue des buttes de Coësmes, 35043 Rennes Cédex, FRANCE

odeforge@insa-rennes.fr, lbedat@insa-rennes.fr, ronsin@insa-rennes.fr

Résumé – Cet article présente une nouvelle solution très performante pour le codage d'images chromatiques en utilisant une technique basée région. L'originalité principale réside dans le fait que la description en régions n'est pas transmise, mais déduite de la représentation basse résolution de l'image des luminances, et donc n'introduit aucun coût supplémentaire. La composante Y est compressée à travers le codeur LAR de base, une approche basée contenu, composé d'un codeur spatial pour atteindre les bas débits, et d'un codeur spectral pour compresser l'image d'erreur. Ces deux codeurs, ainsi que la description en région, partagent la même structure de représentation des données, ce qui évite les artefacts de contours ou de blocs des méthodes traditionnelles.

Abstract – This paper presents a very efficient new solution for chromatic images coding using a region-based technique. The main originality relies in the fact that the regions description is not transmitted, but deduced from the low resolution luminance image representation, and thus do not induce any overhead. The Y component is compressed by the basic LAR codec, a content-based approach composed of a spatial coder to achieve low bit rates, and a spectral coder to compress the error image. The two coders, as well as the region description, share the same data representation structure avoiding contours or blocks artifacts of traditional techniques.

1. Introduction

La première génération des techniques de codage d'image et de vidéo reposent principalement sur la théorie de l'information. La principale limitation des performances en compression est liée aux non-stationnarités présentes dans les images et séquences naturelles qui induisent une distribution variable de l'information en espace et en fréquence. De plus, les structures de représentation de données indépendantes du contenu telles que les grilles conventionnelles de sous-échantillonnage, ou les blocs carrés de données du JPEG ou MPEG, ne permettent pas de décrire de telles non-stationnarités, et par conséquent ne peuvent constituer des structures de représentation parfaitement efficaces.

Des améliorations ont été apportées en représentant les données visuelles en termes de régions, définies par leurs contours et leur texture, et correspondant partiellement ou complètement à des objets de la scène. Cette seconde génération de codeur [1,2] a démontré toute son efficacité, particulièrement à bas débit. Toutefois, les techniques associées sont généralement complexes et souvent limitées par le surcoût introduit par la description des régions.

Nous proposons ici une nouvelle méthode de codage basée région permettant de compresser très fortement les images de chrominance sans introduire de coût pour la description des régions. La technique reste de complexité relativement faible et peut ainsi atteindre des performances de traitement temps-réel.

2. La méthode LAR

La méthode LAR (Locally Adaptive Resolution) a été initialement introduite pour une codage graduel des images en niveaux de gris, allant des bas aux moyens/hauts débits

[3]. Le schéma global (figure 1), en ne considérant que le codeur des luminances, est composé de deux niveaux : un codeur spatial et un spectral. Les forts taux de compression sont obtenus en n'utilisant que le premier codeur. Le second lui permet de coder l'image d'erreur pour de meilleures qualités d'image.

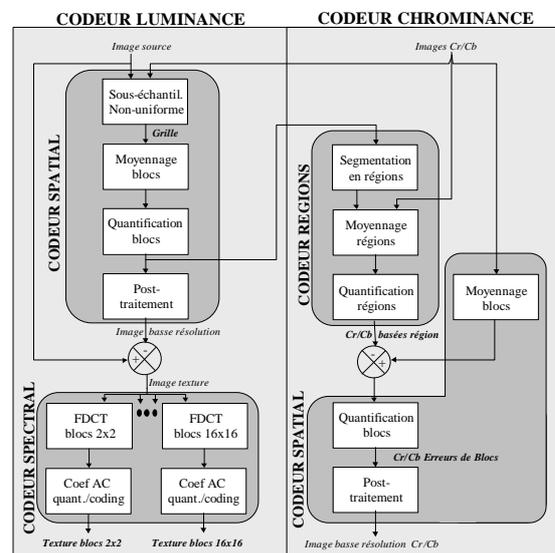


Fig. 1 : Schéma général codeur LAR

2.1 Le codeur spatial

L'image est sous-échantillonnée sur une grille carrée non-uniforme dépendant de l'activité locale. La taille maximale de carré est déterminée sur un critère de détection de frontière, à travers le calcul d'un gradient morphologique dans des blocs 16x16, 8x8, 4x4, 2x2. Un pavage de type

quad-tree peut être utilisé, mais nous avons également développé d'autres types de pavage plus corrélés au contenu. La représentation topologique de l'image ainsi obtenue fournit la résolution locale suffisante pour une reconstruction à basse résolution, mais aussi une carte de segmentation grossière de l'image qui sera utilisée par la suite (voir figure 2.b) : les petits blocs sont localisés sur les contours, alors que les grands se situent à l'intérieur des objets, dans les zones uniformes. L'image basse résolution peut ainsi être obtenue en associant à chaque bloc la moyenne des luminances correspondante dans l'image source. Celle-ci est codée par une approche de type MICD.

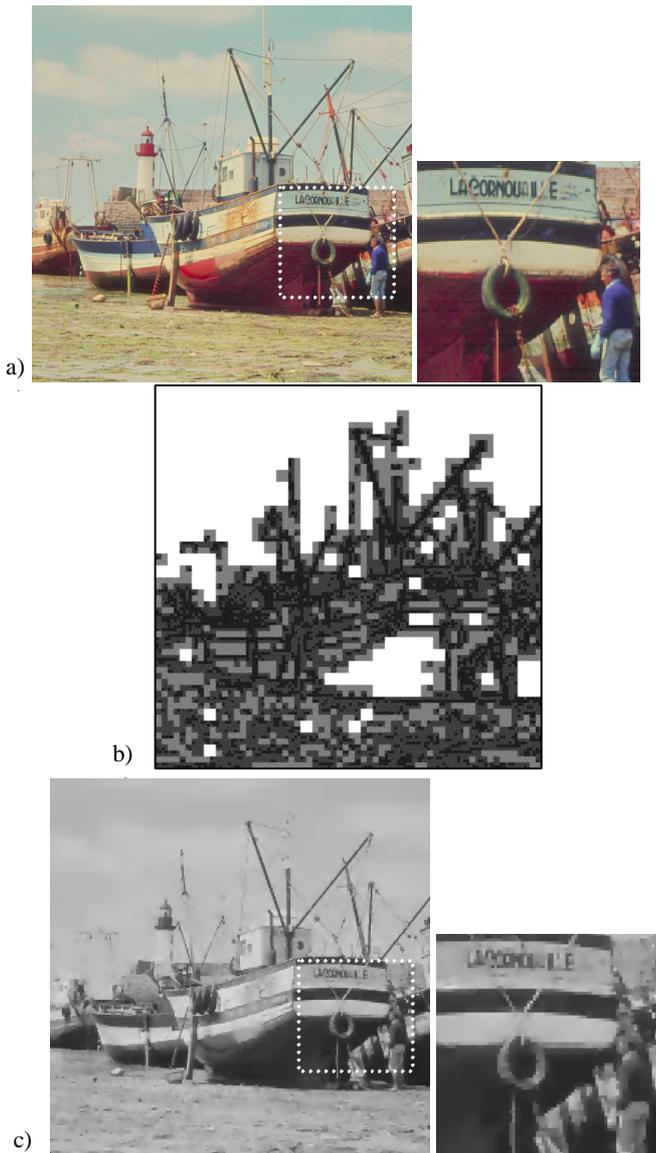


Fig. 2 : a) Image source b) grille de représentation
c) Y issue du codeur spatial (0.164 bpp, 28.5 db)

L'information sur le contenu de l'image donnée à travers la taille des blocs contrôle la quantification de l'erreur de prédiction des luminances : les grands blocs nécessitent une quantification fine car situés dans les zones homogènes où l'œil humain est sensible à des variations de luminosité, alors que les petits blocs supportent une forte quantification car les

zones de contour présentent un effet de masquage des erreurs.

A la fois la taille des blocs et l'erreur de prédiction des luminances sont en final compressées par un codeur arithmétique adaptatif.

Les effets de blocs, perceptibles après reconstruction dans les zones uniformes du fait d'un pavage par des carrés de grande taille, sont simplement éliminés par un post-traitement. Une image résultante est donnée figure 2.c.

Nous pouvons résumer les principales caractéristiques de cette méthode :

- efficace et très rapide pour de forts taux de compression,
- simplifie l'image source en supprimant la texture locale, mais préserve globalement les contours principaux.

2.2 Codeur spectral

Afin d'obtenir des images de meilleures qualités, le second codeur peut être ajouté pour compresser l'image d'erreur (texture locale). Nous avons adopté une approche de type DCT à taille de bloc variable [4], mais où à la fois la taille des blocs et la composante DC sont fournies par le premier codeur. L'utilisation de tailles de bloc adaptées au contenu offre trois avantages majeurs :

- Par construction, le contenu des blocs est stationnaire. Moins de coefficients AC sont donc générés par rapport à une DCT standard à taille fixe, et de plus avec une énergie plus faible.
- Pour chaque taille de bloc, une table de quantification spécifique est définie, introduisant plus de flexibilité dans le contrôle de la qualité finale (ex : une quantification fine des blocs 2x2 aboutit à bonne qualité sur les contours).
- Les blocs sont considérés indépendamment en fonction de leur taille, ce qui permet directement un codage graduel. On peut transmettre par exemple en premier lieu les blocs 2x2 pour raffiner uniquement les contours, puis les 4x4, 8x8 et enfin 16x16 pour l'intérieur des objets.

3. Compression des images de chrominance

Le système visuel humain est moins sensible à la résolution spatiale dans l'espace des chrominances que dans celui des luminances [5,6]. Les résultats ont montré que les images de chrominance issues du codeur spatial étaient suffisantes pour de bonnes qualité de reconstruction, pour peu que la grille de sous-échantillonnage tienne compte des trois composantes couleur. Ainsi, une unique grille de représentation est utilisée. Seule la valeur moyenne des chrominances par bloc nécessite d'être transmise.

Toutefois, nous ne proposons pas simplement ici de dupliquer le codeur spatial pour les deux images de chrominance. Nous introduisons au préalable une approche basée région (voir fig. 1) permettant de profiter de

l'importante corrélation existante entre l'image de luminance et celle des chrominances. En transmettant d'abord l'image Y basse résolution (grille + luminance des blocs), une segmentation en régions est effectuée à la fois au codeur et au décodeur. Les régions sont construites comme des unions de blocs connexes à luminances proches. Un résultat important est que la grille de sous-échantillonnage en blocs constitue une unique structure de représentation à la fois pour les codeurs spatiaux/spectraux et les régions.

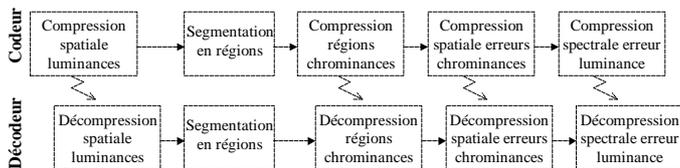


Fig. 3 : Synoptique général de la compression couleur

Concernant le processus de segmentation, l'approche croissance de régions basée blocs plutôt que pixels apporte beaucoup plus de robustesse dès lors que la taille renseigne sur le contenu. D'un point de vue temps de calcul, c'est bien sûr également un élément déterminant.

La segmentation en régions est réalisée en deux étapes appelées *fusion des blocs*, puis *fusion des régions*.

3.1.1 Fusion des blocs

Les régions élémentaires sont construites par des étapes de croissance de régions successives, en ne considérant initialement que les blocs 16x16, puis les 8x8, etc. Ceci assure que le germe des régions soit situé dans les zones homogènes. Le processus de fusion est implanté comme une fonction récursive, partant de l'intérieur des objets pour se propager vers les contours. Un nouveau bloc ne peut être fusionné que si sa taille est au maximum égal à celle du bloc courant. Ceci a pour conséquence d'utiliser les blocs 2x2 (situés sur les frontières) comme séparateur entre deux régions. L'approche générale peut donc être considérée comme conjointe contour/région, tenant compte des images de chrominance qui interviennent dans le sous-échantillonnage.

3.1.2 Fusion des régions

L'étape précédente fournit des régions élémentaires fiables, mais sur-segmente l'image. Cette deuxième phase est un processus "bottom-up" visant à fusionner les régions proportionnellement à leur surface, et ce de manière hiérarchique. Plus une région est petite, plus elle tendra à fusionner avec la région la plus proche en terme de luminance. Le degré de simplification de l'image est contrôlé par un paramètre exprimé en terme de surface.

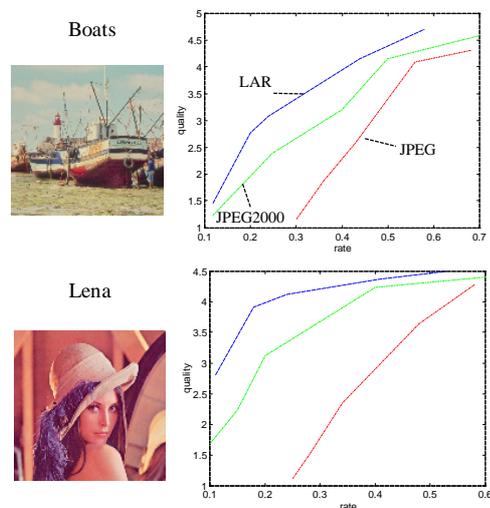
La segmentation en régions ainsi disponible au codeur et au décodeur fournit une représentation de haut niveau de l'image. Les images de chrominance à très bas débit sont ensuite obtenue en transmettant pour chaque région la chrominance moyenne. Des résultats sont visibles figure 5. Pour obtenir de meilleures qualité des couleur, il est possible

alors d'utiliser le codeur spatial pour coder l'erreur de chrominance de chaque bloc.

4. Résultats comparatifs

Le critère débit/distorsion n'est pas bien adapté pour juger des performances du codeur LAR, essentiellement à cause de la "quantification psychovisuelle" réalisée. En fait, les PSNR sont quasiment identiques entre le JPEG standard et le LAR, alors que la qualité d'image est nettement supérieure pour ce dernier. Des tests comparatifs ont été menés entre JPEG, JPEG2000 et le LAR, au laboratoire IRCYN de Nantes. Ces tests visaient à mesurer la qualité subjective des trois techniques, à différentes taux de compression. L'environnement était parfaitement standardisé en terme de distance à l'écran, de luminosité. Le protocole bien établi : image source durant six secondes, pause de deux secondes, image compressée pendant six secondes. Chaque observateur (quatorze au total) devaient ensuite donner une note entre un (très mauvaise qualité) et cinq (très bonne qualité). Quelques résultats comparatifs sont donnés figure 4. Sur huit séries d'images différentes, le LAR s'est montré supérieur pour sept d'entre elles.

D'un point de vue complexité, le LAR se situe entre le JPEG et JPEG2000. Une implantation temps-réel a même été implantée pour un codeur d'images en niveaux de gris sur



une PC à 350 MHz, dans le cadre d'une application de vidéo-surveillance.

Fig. 4 : Résultats comparatifs

5. Conclusions et perspectives

La méthode LAR est une technique de compression basée contenu originellement dédiée aux images en niveaux de gris. Ce travail a montré son extension aux images couleur. En tirant profit de la représentation d'image fournie par le codeur des luminances, et en considérant quelques caractéristiques du système visuel humain, nous avons montré que l'information couleur pouvait être codée très efficacement. Les travaux actuels concernent l'étude d'un

