

# Quantification couleur : comparaisons objectives et subjectives de différents algorithmes

## Color Quantization: Objective and Subjective Comparisons of Different Algorithms

**Mohamed-Chaker Larabi<sup>1</sup>, Christophe Montagne<sup>2</sup>,  
Sylvie Lelandais<sup>2</sup>, André Smolarz<sup>3</sup>, Christine Fernandez-  
Maloigne<sup>1</sup>, Philippe Cornu<sup>3</sup>**

<sup>1</sup> Laboratoire SIC - CNRS FRE 2731 - Université de Poitiers, Bat. SP2MI, Téléport 2, BP 30179 - 86962 Futuroscope Cedex - France

<sup>2</sup> Laboratoire Systèmes Complexes (LSC) - CNRS FRE 2494, Université d'Evry-Val d'Essonne (UEVE), 40 rue du Pelvoux - 91020 Evry Cedex - France

<sup>3</sup> Institut des Sciences et Technologies de l'Information de Troyes (ISTIT) - CNRS FRE 2732, Université de Technologie de Troyes (UTT) 12 rue Marie Curie, BP 2060 - 10010 Troyes Cedex - France

Manuscrit reçu le 15 juin 2004

### Résumé et mots clés

Dans cette contribution, nous proposons d'évaluer la qualité de différents algorithmes de réduction de la palette couleur d'une image. Deux techniques originales sont particulièrement détaillées (avec deux variantes pour chacune d'elles) : l'une basée sur la transformation du boulanger et l'autre employant la matrice des palettes locales. Dans la campagne d'évaluation, les résultats de ces deux techniques sont comparés à ceux d'algorithmes standards tels que « median cut », « octree » et « split & merge ». L'évaluation se veut à la fois objective (utilisation d'une métrique et de descripteurs locaux de qualité) et subjective (utilisation d'expériences psycho-visuelles). En effet, l'usage seul d'une métrique n'intègre pas la notion de HVS (Human Visual System). La couleur étant davantage considérée comme propriété perceptuelle que comme donnée quantitative, une mesure classique ne peut décrire correctement l'altération que subit une image lors d'une réduction des couleurs. Les résultats de la campagne d'évaluation psycho-visuelle montrent une fois de plus que les métriques classiques sont souvent en contradiction avec la perception humaine de la couleur.

**Imagerie couleur, réduction des couleurs, comparaison de méthodes.**

### Abstract and key words

This paper deals with quality assessment of color images dedicated to quantization algorithms. Two methods are described in details, one using the baker's transform and the other using the matrix of local palletets. In this assessment campaign, the results of the described techniques are compared to those of standard algorithms such as median cut, octree and split & merge. Both objective and subjective assessment are performed. The need of subjective evaluation comes from the fact that the usual metrics do not integrate the HVS (Human Visual System). While the color is more considered as perceptual property than a quantitative data, these standard metrics fail in describing image distortion. The preliminary results of the assessment campaign show that the described methods give good results with regards to the former ones.

Color image processing, color quantization, model comparison.

# 1. Introduction

De nos jours, l'utilisation d'images numériques colorées est très répandue que ce soit pour des applications professionnelles ou bien ludiques. Le plus souvent, ces images sont définies dans un espace couleur à trois composantes tel l'espace «RVB 24 bpp»<sup>1</sup> dans lequel la valeur de chaque canal est codée sur un octet<sup>2</sup>. Le nombre de couleurs disponibles pour coder une image est donc supérieur à 16 millions<sup>3</sup>. Ce nombre important de possibilités entraîne que, la plupart du temps, aucun des pixels de l'image n'a une couleur identique aux autres pixels. Nous parlons ici d'images réelles de scènes d'extérieur ou d'intérieur, et pour lesquelles le panel des couleurs existantes n'est limité que par la technologie employée. Ainsi, pour une image de taille  $512 \times 512$ , il n'est pas surprenant de trouver jusqu'à  $2^{18} = 262.144$  couleurs différentes. L'histogramme 3D des couleurs (figure 2) de la figure 1 illustre ce phénomène : les couleurs présentes dans l'image forment un «nuage» relativement dense notamment quand le nombre de pixels d'une même «teinte» est élevé. Cet état est dicté par le souci de reproduire le plus fidèlement possible ce que perçoit un système visuel humain standard mais entraîne que les fichiers informatiques de stockage d'images peuvent être de grande taille, au delà d'une réelle nécessité. De cette manière, une image de taille  $512 \times 512$  sera stockée dans un fichier «bitmap»<sup>4</sup> de 786.486 octets, si le codage des couleurs est en 24 bpp, alors que, si ce codage peut se faire en 8 bpp, la taille du fichier tombera à 263.222 octets, soit quasiment trois fois moins. D'autre part, de nombreux traitements d'images peuvent être rendus impossibles à cause d'un tel codage, notamment ceux utilisant un histogramme des couleurs : avec un taux proche d'une couleur par pixel, l'histogramme est quasiment plat et non significatif. Par ailleurs, des recherches psychovisuelles ont montré que, même si l'œil humain pouvait percevoir un grand nombre de couleurs différentes, le cerveau humain n'interprétait pas systématiquement la différence entre deux couleurs similaires : il apparaît donc inutile de disposer d'autant de couleurs pour coder une image. Pour ces diverses raisons, il semble utile et intéressant de définir des méthodes permettant de réduire, de manière significative, le nombre des couleurs présentes dans une image tout en conservant une qualité visuelle constante. La «quantification», ou réduction, des couleurs est la solution à ce problème. Cela consiste à remplacer une couleur de l'image par une autre issue d'une palette en contenant un nombre limité. Cette couleur doit être la plus proche possible de celle d'origine au sens d'un critère choisi au préalable [OB91, ST97]. Dans la littérature, un certain nombre d'articles traitent de la quantification de couleurs et proposent différentes techniques adaptatives pour sélectionner

la meilleure palette [CKK94, KPH<sup>+</sup>98, BT02]. L'objectif commun à toutes ces méthodes est de préserver l'apparence initiale de l'image. Dans ce document, nous proposons deux nouvelles méthodes adaptatives de quantification de couleur. La première est basée sur la «transformation du boulanger» [SC02, CS02]. La seconde est basée sur la matrice des palettes locales [LRF00]. Dans la première partie, nous présenterons ces deux méthodes ainsi que trois méthodes classiques utilisées dans notre campagne d'évaluation. Puis nous expliquerons la procédure retenue pour comparer ces techniques grâce à des outils objectifs et subjectifs. Dans la troisième partie, nous présenterons les résultats expérimentaux qui seront également discutés. Enfin, nous concluons sur ce travail et mettrons en avant diverses perspectives.



Figure 1. Image originale : perroquets (512 × 512).

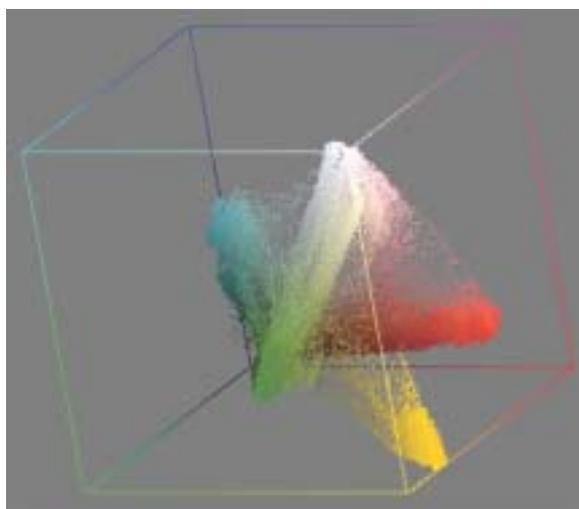


Figure 2. Histogramme 3D de perroquets.

<sup>1</sup> Rouge Vert Bleu - 24 bits par pixel

<sup>2</sup> octet = 8 bits soit  $2^8 = 256$  niveaux possibles

<sup>3</sup> plus précisément  $2^{24} = 16.777.216$

<sup>4</sup>\*.bmp

## 2. Méthodes de quantification

### 2.1. Quantification basée sur la transformation du boulanger

À toute image, il est possible d'associer une texture au moyen d'une transformation de ses pixels. Cette transformation étant une permutation ou une composition de permutations, le passage de l'image à la texture est bijectif donc réversible. Cette texture est une représentation spatiale localement très homogène de l'image qui ne l'est pas initialement. Par ailleurs, cette transformation n'agit que dans le domaine spatial et peut donc s'appliquer indifféremment aux images monochromes ou couleurs. Dans le cas d'images couleurs (dans l'espace RVB par exemple), il suffit d'appliquer la permutation sur chacun des trois plans.

#### 2.1.1. Description de la transformation

La transformation du boulanger est basée sur la définition des mélanges de systèmes dynamiques (ou *mixing*) qui sont des transformations définies sur des espaces continus. Arnold & Avez [AA67] définissent très bien ces mélanges mais nous retiendrons la définition suivante qui est assez intuitive : « Soit un shaker  $M$  rempli d'un liquide incompressible comportant 10% de gin et 90% de martini. Supposons que le gin occupe initialement une portion  $A$  de  $M$ . Après  $n$  agitations  $f$ , le gin occupe la portion  $f^n(A)$ . Physiquement, il est naturel d'espérer que, pour  $n$  assez grand, la proportion de gin contenu dans un volume quelconque  $B$  de  $M$  sera de l'ordre de 10% ». L'intérêt de ce genre de transformations apparaît car elles permettent de « mélanger » de façon très homogène tous les éléments de  $M$ . Toujours intuitivement, il est concevable qu'en prenant un sous-ensemble  $E$  de  $M$  (après mélange), et en appliquant une transformation inverse de  $f$  (restreinte à  $E$ ), les composants initiaux puissent être séparés. Évidemment cela n'a pas lieu d'être en réalité car, en fait, « pour  $n$  assez grand » veut dire « limite quand  $n$  tend vers l'infini ». Toutefois, dans le cas des images numériques (qui sont un ensemble fini de points), cette vision intuitive n'est plus erronée. Arnold & Avez donnent de nombreux exemples de mélanges qui sont définis sur le carré unité  $[0, 1] \times [0, 1]$ . Nous nous intéressons à l'une d'entre elles : la transformation du boulanger (notée  $TB$  par la suite). Notons que tous les exemples donnés par Arnold & Avez (dont la  $TB$ ) sont définis sur des ensembles continus. Or les images numériques sont des ensembles discrets et finis de points (pixels). Il apparaît malheureusement qu'une transformation d'un ensemble discret fini ne peut pas être de type *mixing* [Bil65]. Toutefois, le passage du cas continu au cas discret fait apparaître, pour certaines transformations (dont la  $TB$  justement), une propriété qu'il est possible d'appeler « quasi-mélange » (ou *quasi-mixing*) [CS02] qui est la propriété de bien mélanger (statistiquement) les éléments d'un ensemble discret fini par itérations successives de la transformation. Ainsi, il est possible de démontrer que les pixels

d'une image seront statistiquement convenablement mélangés quand la  $TB$  est itérée un nombre suffisant de fois. Une itération de la  $TB$ , pour une image telle celle de la figure 1, est basée sur trois étapes :

- une transformation affine qui conduit à l'obtention d'une image deux fois plus large et moitié moins haute que l'image d'origine (voir figure 3), le nombre total de pixels restant inchangé ;
- un découpage vertical par le milieu de l'image issue de l'étape précédente ;
- une superposition de la moitié droite sur la moitié gauche (voir figure 4), l'image finale ayant exactement la même taille que l'image d'origine.

Une image numérique étant un ensemble fini de pixels, la  $TB$  correspond à une permutation des différents pixels. Elle impose cependant l'utilisation d'images carrées de taille  $2^N \times 2^N$ . En effet, pour ces images, la  $TB$  est périodique : au bout de  $4N$  itérations, l'image obtenue est identique à l'image initiale. À  $2N$  itérations, l'image résultat a subi une rotation de  $180^\circ$  par rapport à l'image initiale. Le mélange statistiquement « convenable » [CS01] est obtenu soit à  $N$  itérations, soit à  $3N$  itérations (figure 5). Cette image transformée présente les caractéristiques d'une texture microscopique bien particulière. Comme nous l'avons dit précédemment, la  $TB$  est bijective (donc réversible) et la transformation inverse est appelée  $TB^{-1}$ . Donc, en partant du mélange de la figure 5, il est possible de revenir à l'image originale en appliquant  $N$  itérations de la  $TB$  ( $3N + N \equiv 4N$ ) ou  $3N$  itérations de la  $TB^{-1}$  ( $3N - 3N \equiv 0$ ). Une autre propriété très intéressante de la  $TB$  réside dans le fait qu'appliquée à une partie de la texture (image mélangée), elle permet de reconstruire une image fidèle à l'image originale mais de taille plus réduite. La figure 6 illustre cet aspect pour une reconstruction à partir d'une fenêtre carrée de taille  $2^p \times 2^p = 128 \times 128$  ( $p = 7$ ). Cela montre que la texture obtenue présente une grande homogénéité, que l'on peut interpréter par le fait que les éléments caractéristiques de l'image sont distribués spatialement de façon régulière et locale. Il faut toutefois préciser que la reconstruction obtenue avec la  $TB$  n'est correcte (en terme de ressemblance avec l'original) que pour des fenêtres résultant d'une division de la texture en blocs carrés de taille  $2^p \times 2^p$ ,  $p < N$ . Dans cet exemple, la reconstruction est obtenue après  $p$   $TB$  (ou  $3p$   $TB^{-1}$ ). Pour une fenêtre carrée de taille  $2^p \times 2^p$  choisie dans une zone quelconque de la texture, l'image reconstituée présente un aspect structuré mais peu ressemblant à l'original. D'autres propriétés de ces transformations existent mais, dans le cadre qui nous intéresse à savoir la quantification des couleurs de l'image, c'est bien la propriété de reconstruction qui nous importe.



Figure 3. *TB* : itération 1 - étape 1 (256 × 1024).



Figure 4. *TB* : itération 1 - étape 3 (512 × 512).



Figure 5. Image convenablement mélangée (après  $3N = 27$  itérations) (512 × 512).



Figure 6. Fenêtre de taille  $128 \times 128$  ( $p = 7$ ) extraite de la figure 5 et résultat obtenu après  $p = 7TB$ .

### 2.1.2. Définition des palettes de couleurs

Comme nous venons de le montrer dans le paragraphe précédent, une fenêtre de taille  $2^p \times 2^p$ , convenablement positionnée, conserve les caractéristiques de l'image d'origine tant au niveau des formes que des textures et des couleurs. Nous avons vérifié cette assertion, pour ce qui est de la couleur, en comparant les valeurs moyennes des axes Rouge, Vert et Bleu d'environ 300 images avec les valeurs des mêmes axes d'une fenêtre de 256 pixels extraite par la *TB* de chacune de ces images. Nous avons constaté le peu d'écart entre ces moyennes [MLT+04] ce qui illustre la bonne représentation ainsi obtenue pour la couleur. Partant de cette constatation, nous proposons de considérer une telle fenêtre, non pas comme un ensemble de

pixels à restructurer pour obtenir une image reconstruite de plus petite taille, mais comme un ensemble de pixels dont les couleurs sont représentatives de la gamme complète des couleurs présentes dans l'image. Ainsi, en fonction des dimensions de la fenêtre, nous avons à disposition une palette de couleurs adaptée à l'image et obtenue de manière simple et rapide. Une difficulté évidente de notre proposition concerne la manière de choisir la fenêtre à extraire. Avec une image de taille  $2^N \times 2^N$  pixels, il est possible d'obtenir  $K = 2^{2(N-p)}$  palettes contenant au maximum  $2^{2p}$  couleurs. Dans le cadre de notre protocole expérimental, nous nous limitons à l'obtention de palettes d'au mieux 256, 64, 16 ou 4 couleurs. Ces palettes correspondent à des fenêtres extraites de taille  $2^4 \times 2^4$ ,  $2^3 \times 2^3$ ,  $2^2 \times 2^2$ ,  $2^1 \times 2^1$  (ie  $p = 4, 3, 2, 1$ ). Le tableau n°1 illustre, pour une image  $512 \times 512$ , le nombre de palettes disponibles, qui peut être très élevé selon le nombre de couleurs choisies.

Tableau 1. Valeurs pour une image  $512 \times 512$   
( $2^9 \times 2^9$ ,  $N = 9$ ).

Nombre de couleurs souhaitées	Dimensions nécessaires de la fenêtre	p	Nombre possible de fenêtres-palettes $K = 2^{2(N-p)}$
65 536	$256 \times 256$	8	4
16 384	$128 \times 128$	7	16
4 096	$64 \times 64$	6	64
1 024	$32 \times 32$	5	256
256	$16 \times 16$	4	1 024
64	$8 \times 8$	3	4 096
16	$4 \times 4$	2	16 384
4	$2 \times 2$	1	65 536

### 2.1.3. Choix de la palette finale et quantification des couleurs

Le problème posé consiste, à partir des  $K$  palettes existantes, à aboutir à une palette unique qui sera utilisée pour la réduction des couleurs de l'image. Deux voies sont possibles : soit choisir une palette parmi les  $K$  disponibles, soit construire une nouvelle palette en utilisant l'information disponible dans les  $K$  palettes. Différentes possibilités ont été expérimentées [MLSC04] parmi lesquelles nous présentons dans cet article une relevant de chacune de ces deux approches.

#### - Fenêtre-palette médiane :

La première approche consiste à choisir une fenêtre parmi les  $K$  disponibles. Nous avons choisi de calculer l'intensité moyenne de chaque fenêtre, puis de retenir la fenêtre ayant la valeur **médiane** de cette intensité moyenne (comme le montre l'algorithme suivant). Il est possible de simplifier cette méthode en utilisant comme critère non pas l'intensité moyenne des fenêtres mais la somme des niveaux des trois composantes de tous les pixels.

Pour  $k$  allant de 1 à  $K$

$$I(k) = \frac{\sum_{i=1}^{2^p} \sum_{j=1}^{2^p} \frac{R(i,j,k)+V(i,j,k)+B(i,j,k)}{3}}{2^p \times 2^p}$$

Fin

Classer par ordre croissant les  $I(k)$ .

$$k_{\text{sortie}} = \left\{ k \mid I(k) = I_{\text{classées}} \left( \frac{K}{2} \right) \right\}$$

Fenêtre<sub>sortie</sub> = Fenêtre( $k_{\text{sortie}}$ )

#### - Fenêtre-palette moyenne :

La seconde approche consiste en la création d'une  $K + 1^{\text{ème}}$  fenêtre, à partir des  $K$  fenêtres initiales. La valeur de la couleur

finale, à la position  $(i, j)$ , est calculée à partir de la valeur moyenne des  $K$  couleurs à la même position (comme le montre l'algorithme suivant).

Pour  $i$  allant de 1 à  $2^p$

Pour  $j$  allant de 1 à  $2^p$

$$R_{\text{sortie}}(i, j) = \frac{\sum_{k=1}^{k=K} R(i, j, k)}{K}$$

$$V_{\text{sortie}}(i, j) = \frac{\sum_{k=1}^{k=K} V(i, j, k)}{K}$$

$$B_{\text{sortie}}(i, j) = \frac{\sum_{k=1}^{k=K} B(i, j, k)}{K}$$

Fin

Fin

Une fois la palette choisie, le passage de l'image d'origine à l'image quantifiée se fait de manière classique. Pour chaque pixel de l'image d'origine, nous calculons une distance (euclidienne par défaut) entre sa couleur initiale et les  $2^{2^p}$  couleurs de la palette. La couleur de la palette affectée au pixel est celle conduisant à la distance minimale.

Remarquons toutefois qu'extraire une fenêtre avec pour objectif un nombre donné de couleurs, quelle que soit la méthode retenue, ne garantit pas d'obtenir ce nombre. Par exemple, une image unie donnera systématiquement une couleur unique quelles que soient les dimensions de la fenêtre retenue.

## 2.2. Quantification basée sur la matrice des palettes locales (MPL)

La méthode de quantification, baptisée Méthode des Palettes Locales (*MPL*), utilise deux aspects différents lors de la réduction du nombre de couleurs. D'une part les occurrences globales des couleurs dans l'histogramme, et d'autre part la répartition spatiale (locale) de ces couleurs dans l'image.

L'aspect spatial a été étudié et intégré de plusieurs façons dans les algorithmes de quantification couleur. Par exemple, diverses méthodes pour pondérer l'importance d'un pixel, en se basant sur l'activité spatiale de la région de l'image où le pixel apparaît, ont été incorporées aux algorithmes de quantification de couleur [BA91, BO91, BAB94, CTM94]. L'aspect spatial de la méthode MPL ne consiste pas en un calcul d'un critère d'activité mais dans l'approche même de l'extraction des palettes locales

Son développement s'est effectué principalement en deux phases. Tout d'abord, la *MPL* standard [LRF00] qui est basée sur une division adaptative de l'histogramme unidimensionnel de la matrice des palettes locales, et ensuite la *MPL* colorimétrique [LRF01, LRF02] qui fait intervenir des critères de similarité de couleurs pour la sélection de la palette finale. Avant de détailler ces deux phases, nous commencerons par décrire les étapes pré-

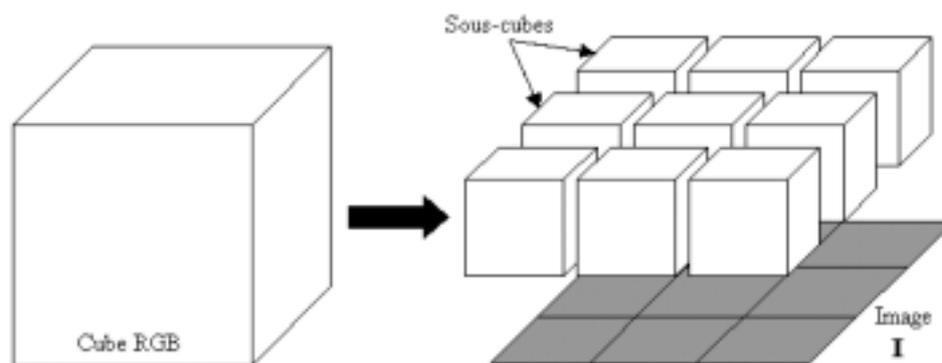


Figure 7. Découpage spatial du cube RGB de l'image I.

liminaires que sont le partitionnement de l'image et la formation de la matrice des palettes locales.

### 2.2.1. Partitionnement de l'image

La première phase de la méthode des palettes locales est celle du partitionnement de l'image. Il s'agit de découper l'image  $I$  en imagettes  $\Omega_j^I$ . Ce découpage peut être vu, comme le montre la figure 7, comme l'ajout d'une information spatiale au niveau de l'histogramme de l'image. Ainsi, au lieu de manipuler un histogramme global  $H^I$  de l'image, comme le font la plupart des méthodes existantes, nous proposons de manipuler un ensemble d'histogrammes partiels  $S^{H^I}$  associé à l'ensemble des imagettes  $\Omega_j^I$ . Autrement dit, il s'agit d'histogrammes partiels spatialement liés entre eux.

Après partitionnement, les imagettes sont toutes de tailles identiques et représentent des fenêtres carrées de taille  $f$  dépendant à la fois de la variance de l'image et de la taille de la palette finale  $K$ .

En effet, la variance nous donne une information sur la dispersion des couleurs dans l'image. Ainsi, quand la variance  $V$  de l'image est supérieure à un certain seuil  $S_V$  déterminé expérimentalement, l'image (*resp.* la fenêtre) est susceptible de renfermer un nombre élevé de couleurs et par conséquent, la taille de la fenêtre doit être la plus petite possible afin de respecter la dynamique couleur de l'image. *A contrario*, une faible variance signifie un faible nombre de couleurs et par conséquent, la taille de la fenêtre doit être aussi grande que possible afin de réduire la complexité du processus.

D'un autre côté, la taille des fenêtres dépend du nombre  $K$  de couleurs à préserver et il est très important que le nombre de couleurs pouvant être retrouvées dans une fenêtre soit supérieur à la taille de la palette finale. Un facteur  $a$  est introduit afin d'imposer une population plus importante que le nombre de couleurs requis et d'avoir une statistique correcte. Par conséquent, la taille d'une fenêtre doit respecter la condition suivante :

$$f^2 \geq a * K. \quad (1)$$

L'équation 1 est purement empirique puisqu'elle est issue des nombreuses expérimentations que nous avons menées afin de

trouver un compromis en complexité et qualité. Le facteur 4 est celui qui permet d'optimiser ce compromis. Les tailles de fenêtres employées dans cet algorithme représentent des puissances de 2. Ainsi, en prenant  $K$  égal à 16, la taille des fenêtres peut être :

$$\begin{aligned} & 8 \quad \text{si } V \geq S_V, \\ & 16 \quad \text{sinon.} \end{aligned} \quad (2)$$

Finalement, la variance, utilisée comme critère pour le choix de la taille des fenêtres, est calculée globalement sur l'image. Bien que le calcul local pourrait paraître comme plus adapté, ce dernier a été expérimenté mais n'a rien apporté en termes de performance.

### 2.2.2. Matrice des palettes locales

Dans cette étape, nous supposons que l'image à quantifier a été découpée en plusieurs fenêtres et que la taille de fenêtre choisie est optimale. À partir de là, nous nous intéressons localement, à chaque fenêtre (imagette) de l'ensemble  $\Omega^I$ . L'idée finale est de récupérer un ensemble de  $K$  couleurs traduisant la répartition statistique et spatiale des couleurs de l'image. Ainsi, pour chaque fenêtre  $\Omega_j^I$ , l'équivalent de ce que fait l'algorithme *Popularity* [Hec82] est appliqué. Ce choix a été guidé par un souci de rapidité de l'algorithme proposé. Il s'agit donc de récupérer l'ensemble des  $K$  couleurs les plus fréquentes de la fenêtre. Cet ensemble est appelé par la suite *palette locale*  $P_j$  et est considéré comme la représentation colorimétrique de la fenêtre  $\Omega_j^I$ .

Cependant deux cas extrêmes peuvent se présenter :

- La fenêtre contient un nombre de couleurs inférieur à  $K$ . Dans ce cas, la palette locale  $P_j$  est maintenue à la taille initiale et les couleurs manquantes sont représentées par des cases vides.

- La fenêtre contient plus de  $K$  couleurs à la même fréquence. Dans ce genre de situation, l'algorithme fait appel à l'histogramme global de l'image afin de départager les différentes couleurs. Ensuite, quand toutes les palettes locales sont extraites, la matrice des palettes locales *MPL* est enfin construite. Elle est considérée comme l'ensemble des couleurs localement fréquentes où chaque ligne représente une palette locale  $P_j$  et par conséquent, une fenêtre de l'image. La *MPL* peut être vue comme une image

mais aussi une version réduite de l'image originale fournissant des informations concentrées sur la dynamique couleur de l'image et sur la répartition spatiale des couleurs dominantes. Le fait que cette information soit concentrée va beaucoup nous aider dans les étapes suivantes et surtout la sélection de la palette finale  $P_F$ .

Après avoir construit la  $MPL$ , celle-ci est considérée comme une image et son histogramme unidimensionnel, que nous appellerons  $\Pi_{MPL}$  est alors calculé. Il s'agit de comptabiliser le nombre de fois où une couleur  $c_n$ , d'une palette locale  $P_m$ , est retrouvée dans les autres palettes locales  $P_j$  de la  $MPL$ . Ce nombre d'occurrences est ensuite affecté à la case associée à la couleur dans  $\Pi_{MPL}$  qui n'est qu'un vecteur de taille  $N$  (généralement différent de  $K$ ) représentant les couleurs et leurs occurrences. L'intérêt d'une telle approche est de fournir un information pertinente sur la distribution spatiale des couleurs dans l'image. Ceci permet, à la différence des approches globale, de prendre en compte des couleurs qui sont faiblement présentes d'un point de vue occurrence mais dont la répartition spatiale a un effet déterminant sur l'aspect final de l'image.

2.2.3. Sélection de la palette finale : Première approche

La sélection de la palette finale est l'une des parties les plus importantes d'un algorithme de quantification puisqu'elle permet l'extraction ou la sélection des  $K$  couleurs représentatives de l'image. Nous avons intitulé la section « première approche » puisqu'il s'agit de la méthode des palettes locales telle qu'elle a été présentée pour la première fois dans [LRF00]. Ceci nous permettra par la suite de décrire l'évolution de la méthode et les améliorations qui lui ont été apportées. Cette première approche s'appuie sur le découpage 1D d'un vecteur classé par ordre colorimétrique par rapport à la couleur la plus fréquente de la  $MPL$ . L'idée est de regrouper les couleurs de la  $MPL$  par similarité colorimétrique et ensuite de ne garder qu'une couleur par intervalle équi-fréquent.

Dans cette première approche, il est supposé que le vecteur  $\Pi_{MPL}$ , associé à la matrice des palettes locales, est calculé. La première tâche consiste à localiser la couleur  $c_{max}$  ayant le nombre d'occurrences le plus important dans l'histogramme unidimensionnel. Ensuite, le vecteur  $\Pi_{MPL}$  est classé, dans un ordre croissant de similarité entre une couleur  $c_n$  et  $c_{max}$ . Cette similarité de couleurs est calculée grâce à la formule donnée par l'équation 3 représentant une distance Euclidienne dans l'espace  $L^*a^*b^*$ . Cet espace est indépendant des périphériques puisqu'il

a été développé à l'aide d'observations et d'expérimentations basées sur la perception humaine. Il est dit perceptuellement uniforme car un écart géométrique dans cet espace correspond à une différence de perception.

$$\Delta^2(c_n, c_{max}) = \Delta^2 L^*(c_n, c_{max}) + \Delta^2 a^*(c_n, c_{max}) + \Delta^2 b^*(c_n, c_{max}), n = 1, \dots, N. \tag{3}$$

Le classement se justifie par le fait de vouloir mettre chaque couleur à côté de celles qui lui sont similaires. Le choix de la couleur la plus fréquente comme « pivot » de similarité est arbitraire ; il peut s'agir de n'importe quelle couleur de la  $MPL$ .

Le cumul  $C_\Pi$  des occurrences contenues dans  $\Pi_{MPL}$  est égal à  $K \times F$ , où  $F$  représente le nombre de palettes locales (fenêtres). Ce cumul est utilisé pour diviser adaptativement et en  $K$  itérations, le vecteur  $\Pi_{MPL}$  considéré comme un intervalle de couleur pondéré par les occurrences. Ainsi, à la première itération, les occurrences des couleurs sont cumulées successivement jusqu'à atteindre la valeur  $C_\Pi^1 = C_\Pi / K$ . Une fois la valeur atteinte, l'intervalle des couleurs  $Int$ , dont les occurrences ont été cumulées, est étiqueté comme zone dédiée à l'extraction de la première couleur représentative. Néanmoins, il est le plus souvent difficile d'avoir un cumul strictement égal à  $C_\Pi / K$ , la solution étant donc, de s'arrêter à la couleur offrant l'erreur de cumul  $|\epsilon_C|$  minimale. Aux prochaines itérations, le cumul restant n'étant plus égal à  $(C_\Pi - (C_\Pi / K))$ , il est alors recalculé en fonction du nouveau contexte. Cette division itérative et adaptative, comme le montre la figure 8, permet d'obtenir  $K$  intervalles équi-fréquents qui vont servir à l'extraction des couleurs représentatives de la palette finale  $P_F$ .

Finalement, de chaque intervalle, nous extrayons une couleur destinée à la palette finale et représentant la plus grande occurrence. Si plusieurs couleurs sont à la même occurrence, nous faisons appel à leurs occurrences globales dans l'histogramme de l'image  $I$ .

2.2.4. Sélection de la palette finale : Seconde approche

La première approche présentée précédemment a donné lieu à des résultats satisfaisants. Néanmoins, sur certains types d'images, et pour des quantifications avec un faible nombre de couleurs, l'approche a montré des défauts de couleur, comme par exemple, la disparition du violet du chapeau de *Lena*, etc. Afin de pallier ces problèmes et d'améliorer la méthode des palettes locales du point de vue colorimétrique, nous avons pro-

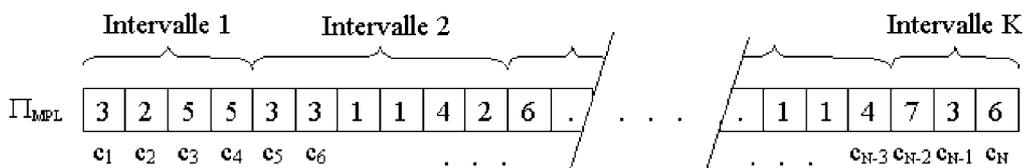


Figure 8. Découpage adaptatif du vecteur  $\Pi_{MPL}$ .

posé une deuxième approche [LRF01,LRF02]. Celle-ci part du principe que deux couleurs très proches ne doivent pas exister en même temps dans la palette finale et ce, quelles que soient leurs occurrences <sup>5</sup>.

Dans cette approche, nous faisons appel à la notion d'arbre colorimétrique. Il s'agit d'un arbre binaire où chaque nœud père est lié aux nœuds enfants grâce à une relation de similarité de couleurs. Dans cet arbre, la base est de dimension  $N$  et représente le nombre de couleurs de la matrice des palettes locales  $MPL$ . La plus haute hiérarchie est de dimension  $K$  et représente les couleurs de la palette finale. Cette structure peut être considérée par les spécialistes des représentations pyramidales (graphes, arbres, etc.), comme une forêt où chacune des  $K$  couleurs représente le sommet d'un arbre indépendant.

Revenons à la phase de construction de l'arbre colorimétrique afin d'en détailler les différentes étapes. Cette construction débute à partir du vecteur d'occurrences  $\Pi_{MPL}$  associé à la matrice des palettes locales. Ce vecteur est de taille  $N$  comprise entre  $2^\alpha$  et  $2^{\alpha+1}$  où  $\alpha$  représente le nombre de niveau de l'arbre binaire standard. Ainsi, la première étape de la construction consiste à réduire le nombre de couleurs de  $N$  à une puissance de deux, représentant la borne inférieure ( $2^\alpha$ ). La réduction s'effectue par agglomération des couleurs, deux par deux, grâce au critère de similarité colorimétrique représentée par une distance Euclidienne (cf. équation 3) dans l'espace  $L^*a^*b^*$ .

Autrement dit, l'algorithme commence par sélectionner la couleur la plus fréquente de  $\Pi_{MPL}$  (ou une des couleurs ayant le nombre d'occurrences maximal). Ensuite, une mesure de distance, entre la couleur sélectionnée et les autres couleurs du vecteur, permet d'y retrouver la plus similaire. Par la suite, cette dernière n'existera plus puisqu'elle aura comme représentant, la couleur qui a été sélectionnée auparavant. Ces étapes sont répétées plusieurs fois jusqu'à ce que toutes les couleurs soient traitées. Notons tout de même qu'une seconde phase d'agglomération est souvent nécessaire afin d'obtenir un nombre de nœuds égal à une puissance de deux.

Une fois la hiérarchie en puissance de deux obtenue, la construction de l'arbre peut continuer en fusionnant, à chaque niveau, les nœuds deux par deux, en se basant sur les critères de fréquence et de similarité colorimétrique. Le processus de fusion est arrêté quand la valeur  $K$  est comprise entre  $2^\alpha$  et  $2^{\alpha-1}$ . Dans le cas où  $2^\alpha = K$ , nous serions alors à un niveau de l'arbre représentant directement les  $K$  couleurs de la palette finale. Cependant, si le nombre de couleurs désiré ne représente pas une puissance de deux, alors  $2^\alpha - K$  couleurs doivent être à nouveau fusionnées.

Dans cette dernière fusion, des conflits peuvent exister pour deux raisons principales. La première provient du fait que la couleur à fusionner donne des distances de similarité égales avec deux ou plusieurs couleurs représentatives de mêmes fréquences. La seconde s'explique par le fait que toutes les cou-

leurs du dernier niveau soient à la même fréquence d'apparition dans la  $MPL$ . Afin de résoudre ce conflit, nous avons fait appel à la distance  $EMD$  <sup>6</sup>. Cette métrique a été introduite par Rubner [RT98] et permet de mesurer fidèlement la distance entre deux ensembles, même s'ils sont de cardinalités différentes. Cette propriété est exploitée pour mesurer la distorsion introduite par chacune des couleurs faisant partie du conflit. En effet, la distance  $EMD$  est calculée entre la couleur conflictuelle et l'ensemble des couleurs de la  $MPL$ , en prenant des triplets constitués des valeurs des trois composantes couleur de l'espace  $L^*a^*b^*$ , des poids égaux à 1 et une distance Euclidienne. Enfin, la couleur ayant donné la distance la plus importante est considérée comme non-représentative et est fusionnée avec la couleur représentative la plus similaire.

Dixit a proposé une méthode qui peut être considérée par certains comme étant proche de celle proposée précédemment. Dans [Dix91], l'image est aléatoirement échantillonnée et un ensemble de clusters couleur est généré grâce à une fusion par la méthode des plus proches voisins (*Pairwise Nearest Neighbor*). Dixit propose ainsi une approche permettant d'obtenir une hiérarchie de clusters. Bien que cette approche semble avoir un lien de parenté avec celle proposée, elle reste néanmoins relativement différente. En effet, l'algorithme de Dixit agit directement sur l'espace de représentation au travers des pixels réunis en clusters à l'étape initiale. Ensuite, ces clusters sont fusionnés en fonction d'un critère de similarité combinant leur cardinal et leur centroïde.

## 2.3. Méthodes classiques de quantification

Cette section présente, sans avoir la prétention d'être exhaustive, quelques méthodes qui ont marqué le domaine telles que le *median cut* de Heckbert [Hec82], la méthode *Octree* de Gervautz et al. [GP90], ... Certaines des méthodes présentées vont nous servir de base de comparaison pour les algorithmes proposés précédemment. Pour une présentation plus large et une étude comparative, le lecteur pourra se référer aux travaux de Chan et al. [CNW94], de Trémeau et Laget [TL95], de Brun et Braquelaire [BB97], de Lozano [Loz98] ou au chapitre quantification de Brun et Trémeau [BT02].

### 2.3.1. Méthode « median cut »

L'algorithme *Median Cut* a été proposé par Heckbert en 1980 [Hec80, Hec82] lors de son stage d'ingénieur au MIT (*Massachusetts Institute of Technology*). Son concept est basé sur l'utilisation de chacune des couleurs de la table (*colormap*) synthétisée afin de représenter un nombre égal de pixels dans l'image originale. Ceci consiste donc, à subdiviser récursivement le cube des couleurs  $RGB$ , contenant les  $n \times m$  couleurs

<sup>5</sup> Cette règle n'est pas respectée lorsqu'il s'agit d'images avec une faible variance colorimétrique.

<sup>6</sup> Earth Mover's Distance.

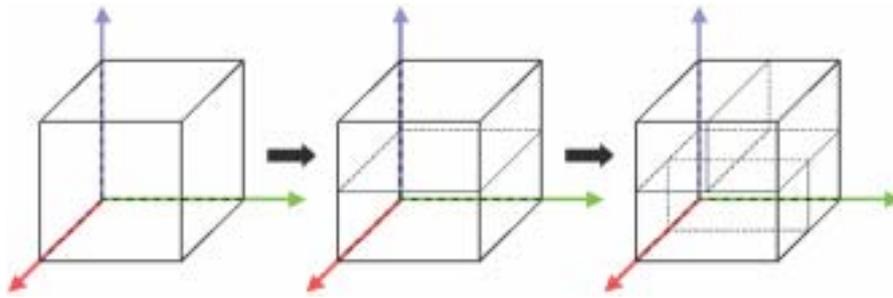


Figure 9. Le schéma fonctionnel de l'algorithme Median Cut.

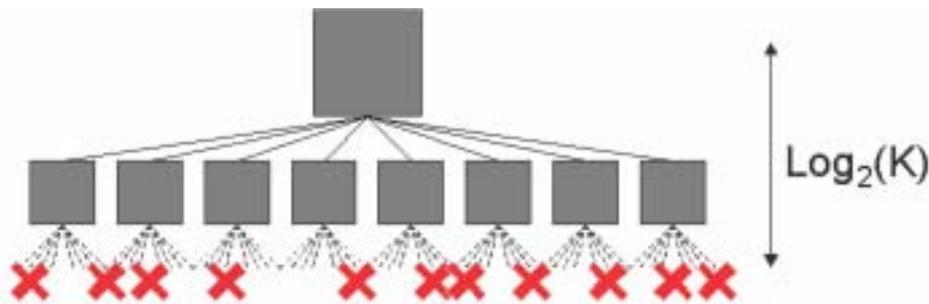


Figure 10. La construction de l'arbre de profondeur  $\text{Log}_2(K)$  pour  $K$  couleurs finales dans l'algorithme Octree. Les croix en rouge représentent les feuilles supprimées.

(avec  $n$  et  $m$  respectivement le nombre de lignes et le nombre de colonnes de l'image), en des boîtes parallélépipédiques de plus en plus petites comme le montre la figure 9.

Cette subdivision consiste comme le souligne *Lozano* [Loz98] à choisir le plan perpendiculaire à l'axe ayant la distribution la plus large et passant par la médiane de la distribution. Ainsi, il est possible d'avoir un nombre de couleurs relativement égal de part et d'autre du plan de subdivision. Le processus est arrêté dès que le nombre de boîtes atteint la valeur  $K$ .

L'idée de base de cet algorithme réside dans l'équi-répartition des pixels dans les différentes boîtes. Elle est certes simple et facile à implémenter mais n'implique en aucun cas une subdivision basée sur des critères perceptuels. Les résultats sont néanmoins satisfaisants et les tests subjectifs menés par *Heckbert* ont montré que cet algorithme produit de meilleurs résultats que l'algorithme *Popularity* [Hec82] basé sur un seuillage de l'histogramme.

Une variante de la méthode *Median Cut* a été proposée par *Xiang* dans [Xia97]. Cette variante sera également expérimentée dans le cadre de cette étude. Malgré, la similarité des deux approches, les résultats sont loin d'être du même ordre de qualité.

### 2.3.2. Méthode « octree »

La méthode *Octree* a été introduite par *Gervautz* et *Purgathofer* en 1988 [GP88,GP90]. Elle est souvent considérée comme une variante du *Median Cut* puisqu'elle se présente comme une méthode basée sur le partitionnement de l'espace des couleurs

*RGB*. Trois étapes peuvent être distinguées : la classification, la réduction et l'affectation. Le principe de base de la classification est de subdiviser récursivement chaque cube en 8 sous-cubes de taille identique et de construire un arbre de représentation. Ainsi, la racine de l'arbre est le cube *RGB*, chaque nœud de cet arbre a 8 enfants et les feuilles sont les couleurs de l'image. Les nœuds n'ayant pas de feuille sont alors supprimés. La profondeur de l'arbre dépend de  $K$  puisqu'elle est égale à  $\text{Log}_2(K)$  (figure 10).

À chaque nœud on affecte cinq variables :

- $n_1$  : nombre de pixels de l'image contenus dans le cube associé à ce nœud.
- $n_2$  : nombre de pixels dont la couleur n'est pas associée à un nœud situé plus bas dans l'arbre ( $n_2 = 0$  dans la phase de construction mais sa valeur change dans la phase de réduction).
- $\Sigma_R, \Sigma_G$  et  $\Sigma_B$  : sommes respectives des composantes rouge, verte et bleue, des couleurs non-représentées plus bas dans l'arbre.

Dans la phase de réduction, les feuilles représentant les valeurs les plus faibles de  $n_1$  sont supprimées. Les valeurs  $n_2, \Sigma_R, \Sigma_G$  et  $\Sigma_B$  du nœud père sont alors mises à jour. Cette opération est répétée plusieurs fois, jusqu'à atteindre un nombre de feuilles égal à  $K$ .

Finalement, la phase d'affectation permet de remplacer la couleur d'un pixel par la couleur représentative. Celle-ci est donnée par la feuille située en dessous d'elle et a pour composantes

$$\frac{\Sigma_R}{n_2}, \frac{\Sigma_G}{n_2} \text{ et } \frac{\Sigma_B}{n_2}.$$

### 2.3.3. Méthode « split & merge »

Cette méthode introduite par Brun [BM00] effectue la quantification des couleurs en deux phases distinctes. La première consiste à diviser l'espace de représentation de la couleur en  $N$  classes en utilisant la technique de quantification uniforme (*split*). Cette phase permet d'éviter un nombre important de fusions à effectuer par la suite. Dans la seconde phase, le nombre de couleurs final  $K$  étant donné,  $N - K$  fusions sont effectuées (*merge*). Ainsi, à chaque étape, les deux classes fusionnées sont celles qui produisent l'erreur minimale. En reprenant les notations de Brun [BM00],  $P = \{C_0, \dots, C_N\}$  décrit la partition initiale de l'espace couleur. La fusion de deux classes  $C_i$  et  $C_j$  crée une nouvelle partition  $P'$  avec une erreur égale à :

$$E(P') = E(P) + \frac{|C_i||C_j|}{|C_i| + |C_j|} \|\mu(C_i) - \mu(C_j)\|^2, \quad (4)$$

où  $E()$  représente l'erreur et  $|C| = \sum_{c \in C} f(c)$  est le cardinal d'une classe ou partition avec  $f(c)$  le nombre de pixels de couleur  $c$  dans la classe. La moyenne couleur étant définie comme

$$\mu(C) = \frac{\sum_{c \in C} f(c)c}{|C|}.$$

Ainsi, l'algorithme de fusion crée un graphe appelé *Graphe d'Adjacence de Classes (GAC)* en partant de la partition initiale en  $N$  classes. Par la suite, les nœuds de ce graphe représentant les différentes classes sont fusionnés en minimisant l'erreur produite.

Les résultats obtenus par cet algorithme sont de très bonne qualité pour un temps de calcul très faible. L'auteur a noté que l'erreur quadratique reste presque constante pour des valeurs de  $N > 200$ .

## 3. Outils d'évaluation de la qualité

Évaluer la qualité des nouvelles méthodes est indispensable et inévitable. Dans le cas des méthodes de quantification des couleurs, cette évaluation est un peu particulière car elle consiste principalement à estimer l'écart visuel entre image quantifiée et image originale. Qui dit écart visuel dit écart perceptible par le système visuel humain. L'objectif consiste donc à tromper notre vision. Une « bonne » quantification doit alors conserver le plus possible les formes, les textures, les couleurs mais aussi d'autres éléments moins élémentaires tels les dégradés de couleur. De fait, nous procédons à deux types d'évaluation : évaluation objective et évaluation subjective. Objective car les outils employés pour ce faire sont répandus et faciles à mettre en oeuvre. De plus les valeurs numériques sont l'essence même de

l'esprit d'analyse scientifique. Subjective car l'image est une notion directement liée à la perception visuelle humaine et qu'*a priori* les avancées scientifiques actuelles ne permettent pas de quantifier la qualité tel qu'effectué par un observateur humain.

### 3.1. Évaluation objective

Dans la plupart des travaux sur la quantification, les auteurs s'intéressent seulement au critère d'évaluation d'Erreur Quadratique Moyenne (*EQM*). Cette dernière permet d'évaluer de façon quantitative la distorsion apportée à l'image originale. Néanmoins, elle ne fournit aucune information sur la qualité visuelle de l'image. Nous pouvons nous retrouver devant le cas de deux images où, celle qui donne l'erreur la plus importante est, du point de vue qualité visuelle, la plus acceptable. C'est pour cette raison que nous avons fait appel à des critères, dits de qualité ou de fidélité, introduits par Trémeau *et al.* [TCF97, Tré98]. Nous reviendrons sur ces derniers dans les prochaines sections.

#### 3.1.1. Erreur quadratique moyenne

Dans la littérature concernant la quantification couleur, l'utilisation de l'Erreur Quadratique Moyenne (*EQM*) pour la caractérisation de la qualité est très fréquente. Elle est donnée par l'équation suivante :

$$EQM(I, I_Q) = \frac{1}{N} \sum_{(x,y) \in I} D^2(I(x,y), I_Q(x,y)), \quad (5)$$

où  $I$  et  $I_Q$  sont respectivement l'image et l'image quantifiée et  $N$  correspond au nombre total de pixels contenus dans ces images. La figure 11 donne un aperçu de l'erreur pouvant être générée par une opération de quantification en 64 couleurs.

#### 3.1.2. Les descripteurs locaux de qualité

Comme deuxième critère d'évaluation de la qualité de la quantification, nous nous sommes intéressés aux descripteurs locaux de qualité introduits par Trémeau *et al.* dans [TCF97]. Ces derniers offrent la possibilité de caractériser la qualité d'une image par le biais de quatre descripteurs spécifiques et complémentaires.

1. **La différence locale de luminance** : ce descripteur permet, en se basant sur la loi de *Weber*, de tenir compte de la sensibilité de l'œil humain à la luminance :

$$D_{lum}(x,y) = 1 - \frac{|\ln(\mu_L^1(x,y)) - \ln(\mu_L^2(x,y))|}{\ln(I_L^M(x,y)) - \ln(I_L^m(x,y))}. \quad (6)$$

2. **La différence locale de chrominance** : ce descripteur est basé sur le fait qu'un observateur est plus sensible à des dégradations de couleur dans des zones homogènes que dans les zones texturées. La formule donnée est exprimée en coordon-

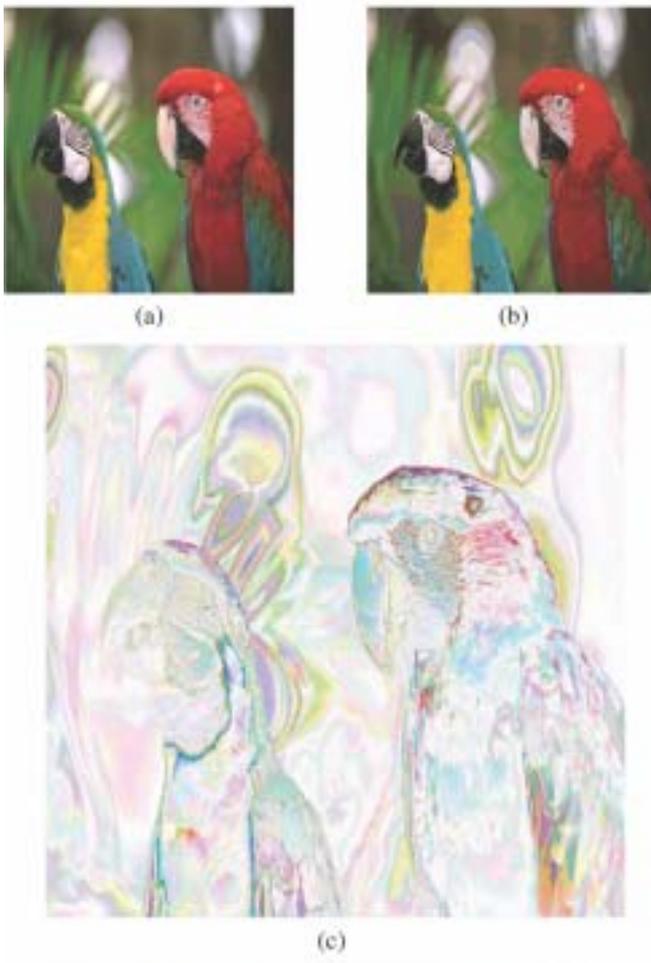


Figure 11. (a) Image originale, (b) image quantifiée en 64 couleurs, et (c) leur différence (images 512 × 512).

nées  $a^*b^*$  mais comme le précise l'auteur [Tré98], elle peut être étendue à toute représentation 2D ou 3D de la chrominance :

$$D_{Chr}(x, y) = 1 - \frac{\sqrt{(\mu_{a^*}^1(x, y) - \mu_{a^*}^2(x, y))^2 + (\mu_{b^*}^1(x, y) - \mu_{b^*}^2(x, y))^2}}{\sqrt{(I_{a^*}^M(x, y) - I_{a^*}^m(x, y))^2 + (I_{b^*}^M(x, y) - I_{b^*}^m(x, y))^2}} \quad (7)$$

**3. La différence locale d'émergence :** ce descripteur repose sur le calcul de l'émergence d'un point (ou d'une couleur)  $(x, y)$  par rapport à son voisinage.

$$D_{Emg}(x, y) = 1 - M \left( \frac{|e^{11}(x, y) - e^{12}(x, y)|}{M_{(e^{11}, e^{12})} - m_{(e^{11}, e^{12})}}, \frac{|e^{21}(x, y) - e^{22}(x, y)|}{M_{(e^{21}, e^{22})} - m_{(e^{21}, e^{22})}} \right), \quad (8)$$

où

- $M$  et  $m$  représentent respectivement les fonctions max et min,
- $e^{ij}$  compare un pixel de l'image  $i$  par rapport au voisinage  $V$  du pixel correspondant dans l'image  $j$  :

$$e^{ij}(x, y) = \frac{1}{Card(V)} \sum_{(x', y') \in V} d(I^i(x, y), I^j(x', y')), \quad \forall (i, j) \in \{1, 2\} \quad (9)$$

**4. La différence locale de corrélation :** ce descripteur permet d'effectuer, entre des régions, des comparaisons selon la dispersion spatio-colorimétrique de chacune des deux images couleur :

$$D_{Cor}(x, y) = \frac{|\sum_{i=1}^3 cov_i^{12}(x, y)|}{\sqrt{\sum_{i=1}^3 (\sigma_i^1(x, y))^2} \cdot \sqrt{\sum_{i=1}^3 (\sigma_i^2(x, y))^2}} \quad (10)$$

où

-  $i$  représente la composante couleur à considérer,

$$- (\sigma_i^j(x, y))^2 = \frac{1}{Card(V)} \sum_{(x', y') \in V} (I_i^j(x', y') - \mu_i^j(x, y))^2$$

représente la variance de voisinage  $V$  du pixel considéré dans l'image  $j$ ,

$$- cov_i^{12}(x, y) = \frac{1}{Card(V)} \sum_{(x', y') \in V} (I_i^1(x', y') - \mu_i^1(x, y))(I_i^2(x', y') - \mu_i^2(x, y))$$

est la covariance du voisinage  $V$  du

pixel considéré de l'image 1 par rapport à l'image 2.

Cependant, les auteurs remarquent que cette distance n'est cohérente que si les valeurs des variances sont supérieures à un seuil  $\varepsilon$ . Ainsi, si les deux variances sont inférieures à  $\varepsilon$ , la différence locale de corrélation est considérée comme inexistante et par conséquent  $D_{Cor}(x, y)$  est égale à 1. Cependant, si les deux variances sont supérieures à  $\varepsilon$ , la différence locale de corrélation est considérée comme maximale et est égale à 0.

Finalement, afin d'avoir une valeur finale  $D_Q$  représentant la qualité de la quantification obtenue, Charrier et al. [Cha98] propose de calculer pour chaque descripteur, une valeur normalisée comprise entre 1 (pas de différence) et 0 (différence maximale), puis de combiner les descripteurs grâce à la formule suivante :

$$D_Q = \frac{1}{2} \sqrt{D_{Lum}^2 + D_{Chr}^2 + D_{Emg}^2 + D_{Cor}^2} \quad (11)$$

La valeur ainsi obtenue représente une note de qualité comprise entre 0 et 1. Plus la grandeur  $D_Q$  tend vers 1, meilleure est la qualité de la quantification et *vice versa*.

### 3.2. Évaluation subjective

Malgré les efforts énormes de recherche, la compréhension de l'interaction complexe existant entre les modèles d'images et les modèles du Système Visuel Humain (SVH) est insuffisante pour pouvoir quantifier la qualité visuelle. De ce fait, les expériences psychovisuelles restent le seul moyen viable pour l'évaluation des performances qualitatives des traitements effectués sur les images.

Les évaluations psychovisuelles sont des expériences du type «*Human in the loop*» (l'humain dans la boucle). Elles se caractérisent par la participation d'observateurs humains afin d'exploiter les capacités du SVH à mesurer la qualité d'un objet. Il peut s'agir d'expériences où l'observateur est appelé à faire un choix entre deux images ou d'expériences de classement d'une série d'images par ordre de qualité décroissante. Cette nécessité de faire participer des observateurs humains à l'évaluation de la qualité s'explique par le fait que l'être humain est doté d'une capacité à évaluer la qualité. Cette capacité de jugement est en plus adaptative puisqu'elle s'adapte à différents cas de figure où la notion de qualité n'est pas du tout la même.

Lors des expériences psychovisuelles, il est indispensable de s'assurer que l'observateur humain possède d'une part une vision normale des couleurs et d'autre part, une acuité visuelle correcte. Pour ce faire, des tests ophtalmologiques existent comme le fameux test d'*Ishihara* [McI02].

D'autres précautions sont à prendre par rapport au matériel utilisé et l'environnement dans lequel les tests psychovisuels se déroulent. En effet, nous ne pouvons parler d'évaluation de la qualité sans être dans un environnement normalisé. Cette normalisation [Bed98] regroupe plusieurs critères comme la distance entre l'écran de visualisation et le fond de la salle, la réflexion des revêtements muraux, sans oublier la correction gamma.

### 3.2.1. Protocole

Le schéma de la figure 12 montre le protocole adopté pour l'évaluation subjective des méthodes de quantification présentées précédemment. Les expériences s'effectuent dans une salle spécialisée, établie selon les normes internationales ITU-R 500-10 [ITU00]. Les murs de la pièce sont de gris neutre, afin d'éviter l'effet de réflexion de la lumière. L'illumination de la pièce est artificielle, variable et totalement contrôlée. Les observateurs sont assis à une distance de 90 cm de l'écran où sont projetés les images, afin de permettre une très bonne perception des détails dans des conditions d'environnement mésopique. L'écran délivre suffisamment de lumière pour stimuler correctement le SVH.

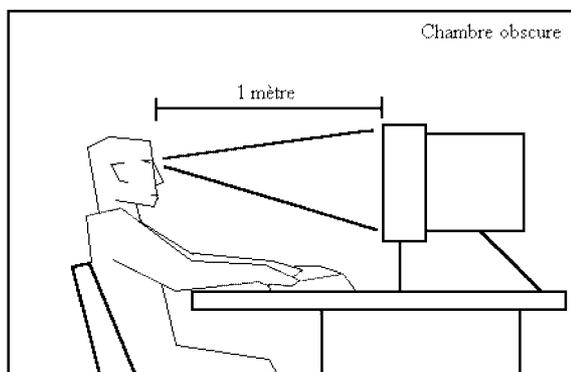


Figure 12. Protocole psychovisuel d'évaluation de qualité.

### 3.2.2. Test d'ordonnement

Le but de ce test est de classer (de la meilleure à la plus mauvaise) une série d'images par rapport à une image originale. Pour les besoins de cette expérimentation psychophysique, nous avons retenu, comme dispositif d'étude, un processus affichant simultanément neuf images à l'écran.

L'image originale est placée au centre et huit images à étudier sont situées autour d'elle comme le montre la figure 13. L'observateur humain est invité à indiquer l'image qu'il voit ou considère la plus éloignée qualitativement de l'image originale. Cette image ainsi désignée est alors masquée et la même question est reposée concernant les images restantes. Le processus se répète jusqu'à ce que les huit images soient masquées. Cette technique permet de classer les images par rapport à une référence (souvent l'image originale). Et contrairement aux techniques classiques où l'observateur doit classer toutes les images par ordre croissant ou décroissant de qualité, cette technique présente l'avantage de ne jamais mettre l'observateur devant un choix complexe. En effet, il doit juste répondre à chaque fois au problème par «*voici l'image que je n'apprécie pas dans le lot*».



Figure 13. Capture d'écran d'un test d'ordonnement.

À l'issue de ce test, chaque image du test se voit assigner un score correspondant à son classement dans la série d'observation. Ce score est compris entre 1 pour la plus mauvaise et 8 pour la meilleure.

## 4. Résultats et discussion

Pour nos besoins expérimentaux, nous avons utilisé 7 images de la base Kodak (cf. figure 14). Ces dernières ont été choisies parce qu'elles fournissent une large gamme de couleurs et de textures pouvant être intéressante dans le cadre de la mesure des limites des méthodes de quantification. Toutes les images sont

codées en 8 bits par composante et 3 composantes par pixel (soit 24 bits par pixel). Les méthodes de quantification présentées précédemment ont été appliquées aux 7 images de la figure 14. Quatre niveaux de quantification ont été choisis à savoir : 4, 16, 64 et 256 couleurs. Chaque image a donc été quantifiée à ces divers nombres de couleurs. Huit algorithmes sont employés pour l'évaluation (objective et subjective). Afin de simplifier l'écriture, nous avons attribué à chacune des méthodes un label : "mcut" pour «median cut», "mcutx" pour «median cut» de Xiang, "S&M" pour «split and merge», "Oct" pour «octree»,



Figure 14. a) façade, b) casquettes, c) fleurs, d) maisons, e) enfant, f) maison, et g) perroquets (images  $512 \times 512$ ).

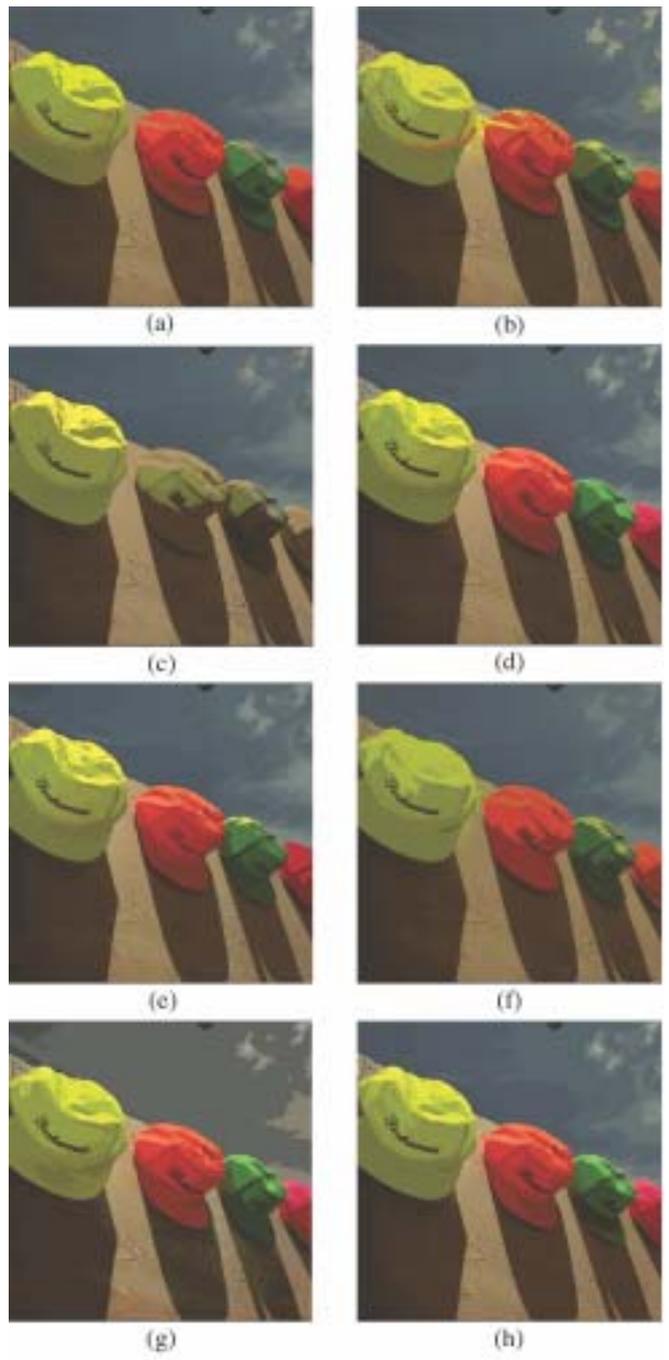


Figure 15. a) TBmed, b) Oct, c) mcut, d) MPL2, e) MPL1, f) TBmoy, g) mcutx et h) S&M.

“TBmed” et “TBmoy” pour les deux variantes de la transformation du boulanger, et enfin “MPL1” et “MPL2” pour les deux variantes de la méthode MPL.

La figure 15 montre quelques résultats de quantification de l'image «casquettes» à 64 couleurs en utilisant les huit méthodes décrites précédemment.

Dans les sections suivantes, nous allons présenter et commenter les résultats obtenus en utilisant l'erreur quadratique moyenne, les descripteurs locaux de qualité et les expériences psychovisuelles.

#### 4.1. Erreur quadratique

Nous avons calculé le critère EQM pour les différentes méthodes de quantification utilisées, en employant les sept images de test aux différents taux de quantification retenus. Ces calculs ont permis de construire les graphes donnés par les figures 16, 17, 18 et 19, qui donnent respectivement les valeurs EQM par image et par méthode, pour l'image « façade », pour l'image « perroquets » et enfin pour toutes les images confondues. De ces graphes, nous constatons les points suivants :

- Comme l'illustre la figure 16, qui représente les valeurs moyennes pour tous les niveaux de quantification, le critère EQM est très dépendant de la « nature » de l'image et par conséquent une méthode de quantification, qui obtient une valeur élevée sur une image, ne peut être directement classée comme la plus mauvaise.

- Les figures 17 et 18 permettent de montrer la variabilité des résultats et par conséquent les taux de distortion qui diffèrent d'une image à l'autre.

- Le comportement moyen des méthodes de quantification étudiées est donné par la figure 19. Les valeurs de ce graphique nous permettent de constater que les méthodes proposées produisent, dans l'ensemble, des distorsions très faibles quelque soit le niveau de quantification. Notons tout de même que, pour les palettes finales de grande taille (256 et 64 couleurs), quatre méthodes conduisent à d'excellents résultats : Oct, TBmed, MPL1 et MPL2. De même, la méthode split & merge est excellente pour les faibles nombres de couleurs résultats.

Si nous observons que les méthodes proposées, à savoir TBmed, MPL1 et MPL2, offrent des résultats très satisfaisants, nous constatons néanmoins qu'il est très difficile de tirer des conclusions quant à la meilleure méthode entre les huit. Alors, pouvons-nous mesurer la qualité d'une quantification avec le critère EQM ? Rappelons que ce critère peut représenter significativement la distorsion entre l'image originale et l'image quantifiée mais peut être facilement mis en défaut par son aspect ponctuel. En effet, il s'agit d'une erreur calculée entre pixels de même position sur deux images, sans prendre en compte leurs voisinages qui a une relation directe avec la qualité visuelle. Ce problème de robustesse du critère EQM nous a amené à prendre en compte les descripteurs locaux de qualité.

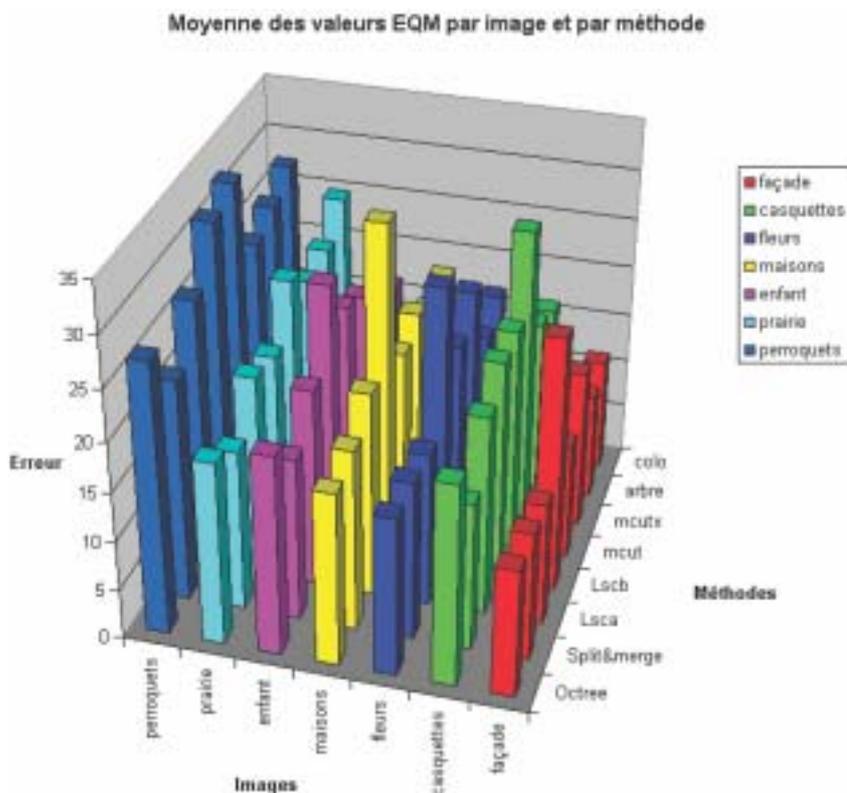


Figure 16. Erreur quadratique moyenne par image et par méthode.

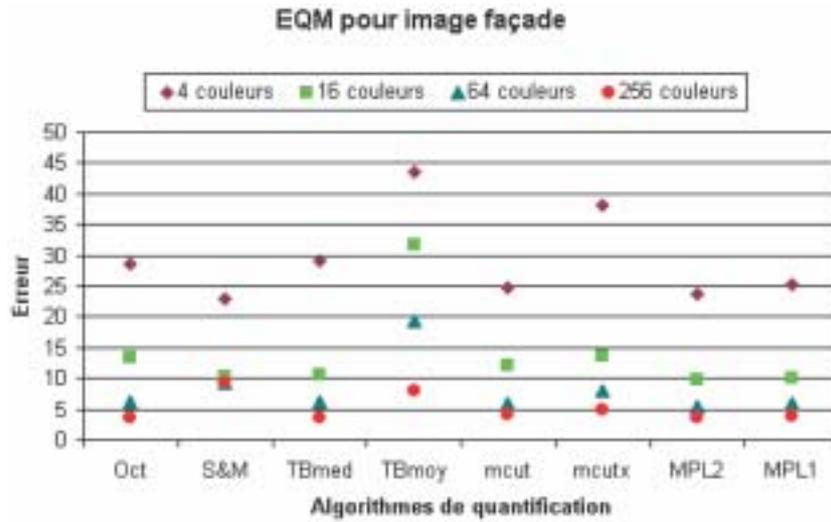


Figure 17. Erreur quadratique moyenne pour l'image façade.

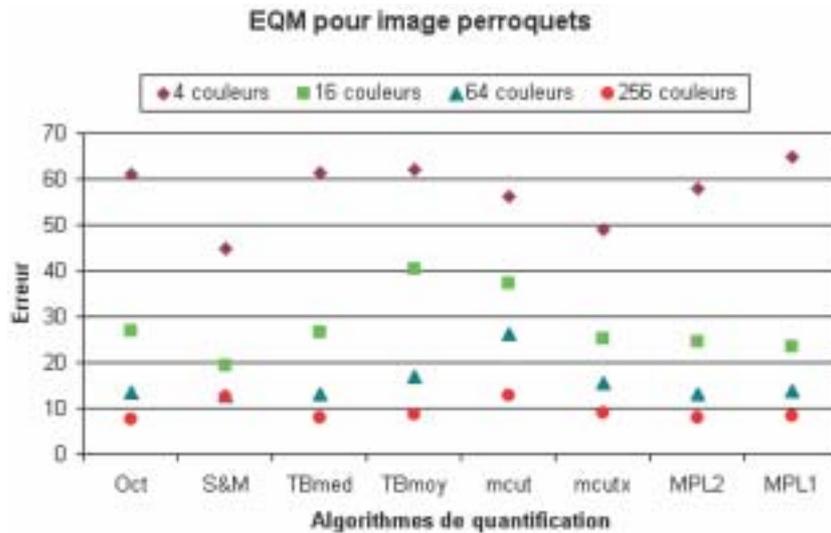


Figure 18. Erreur quadratique moyenne pour l'image perroquets.

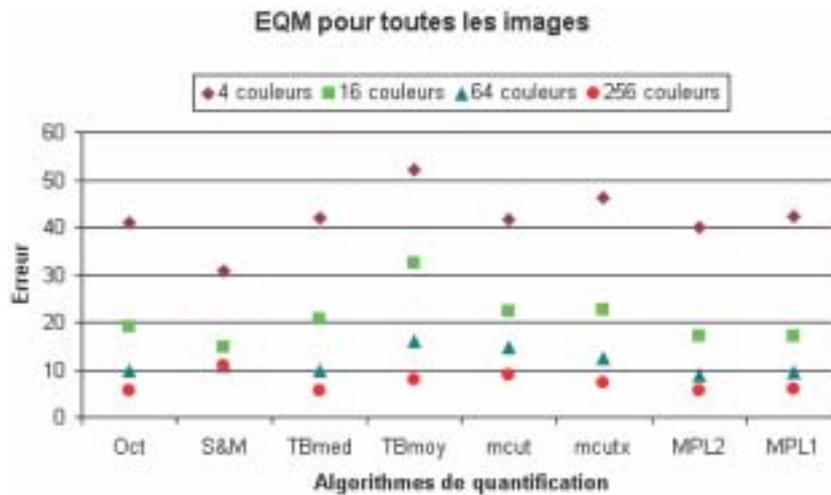


Figure 19. Erreur quadratique moyenne pour toutes les images.

### 4.2. Descripteurs locaux de qualité

La figure 20 montre les images de distances obtenues à partir des quatre descripteurs cités précédemment.

De même que pour le critère  $EQM$ , afin de comparer les différentes méthodes de quantification en employant le descripteur de qualité, nous avons calculé  $D_Q$  pour toutes les méthodes en utilisant les sept images de test aux différents taux de quantification retenus. Ces calculs ont permis de construire les graphes donnés par les figures 21, 22 et 23, qui donnent respectivement les valeurs du critère  $D_Q$ , pour l'image « façade », pour l'image « perroquets » et enfin pour toutes les images. Ces graphes nous permettent de faire les remarques suivantes :

- Le calcul du critère  $D_Q$  est très dépendant de l'image traitée. Rappelons que plus la valeur est élevée, meilleure est la qualité.
- Les valeurs moyennes des descripteurs locaux de qualité, représentés par la figure 23, permettent de remarquer un com-

portement stable de la méthode TBmed basée sur la transformation du boulanger. En effet, cette dernière présente les meilleures performances pour les quantifications en 256, 64 et 4 couleurs. Elle est suivie de très près par les méthodes basées sur la MPL.

En conclusion de cette évaluation par les descripteurs locaux de qualité, il est possible de dire que les méthodes proposées offrent des résultats dans la gamme de qualité des méthodes existant dans la littérature, voire meilleurs pour certaines images. De plus, il est facile de vérifier que sur certaines images, les valeurs obtenues coïncident avec l'appréciation humaine de la qualité de la quantification. Cependant, il est impossible de généraliser cette constatation car, sur d'autres images, cette coïncidence ne peut être confirmée. Rappelons que les descripteurs locaux de qualité ont été définis en se basant sur les critères du système visuel humain [TDF96]. Cependant, un travail de mise en correspondance entre ces critères et un observateur humain grâce à des expériences psychovisuelles doit être effectué. Celui-ci peut permettre d'attribuer des poids aux différents descripteurs en fonction de leur influence sur l'appréciation de la qualité.

### 4.3. Évaluations psychovisuelles

Les expériences subjectives sont réalisées sur un panel de 20 observateurs, ayant différents niveaux de connaissance en matière de traitement d'images, dont la vision des couleurs a été vérifiée en utilisant le test d'Ishihara. Ce faible nombre d'observateurs s'explique par le fait que les campagnes d'évaluation subjective sont très fastidieuses et nécessitent de nombreuses heures de présence.

Les résultats d'une évaluation psychovisuelle sont souvent décrites en termes de MOS (*Mean Opinion Score*). Ces valeurs reflètent le comportement des observateurs face à la batterie de tests à laquelle ils sont soumis.

Le tableau 2 donne les valeurs moyennes du MOS représentant les opinions des observateurs. Il donne aussi les valeurs de l'écart-type pour chaque algorithme de quantification.

Notons que nous disposons d'un intervalle de confiance, calculé à partir de l'ensemble des évaluations de la campagne psy-

Tableau 2. MOS et écart-type moyen de toutes les évaluations menées.

algorithme	MOS	écart-type
MPL2	5.614286	1.855824
MPL1	5.542857	1.986999
S&M	5.278571	2.328604
Oct	5.100000	1.939440
TBmed	4.596429	2.064493
mcutx	3.814286	2.163284
mcut	3.042857	2.071478
TBmoy	3.010714	2.018634

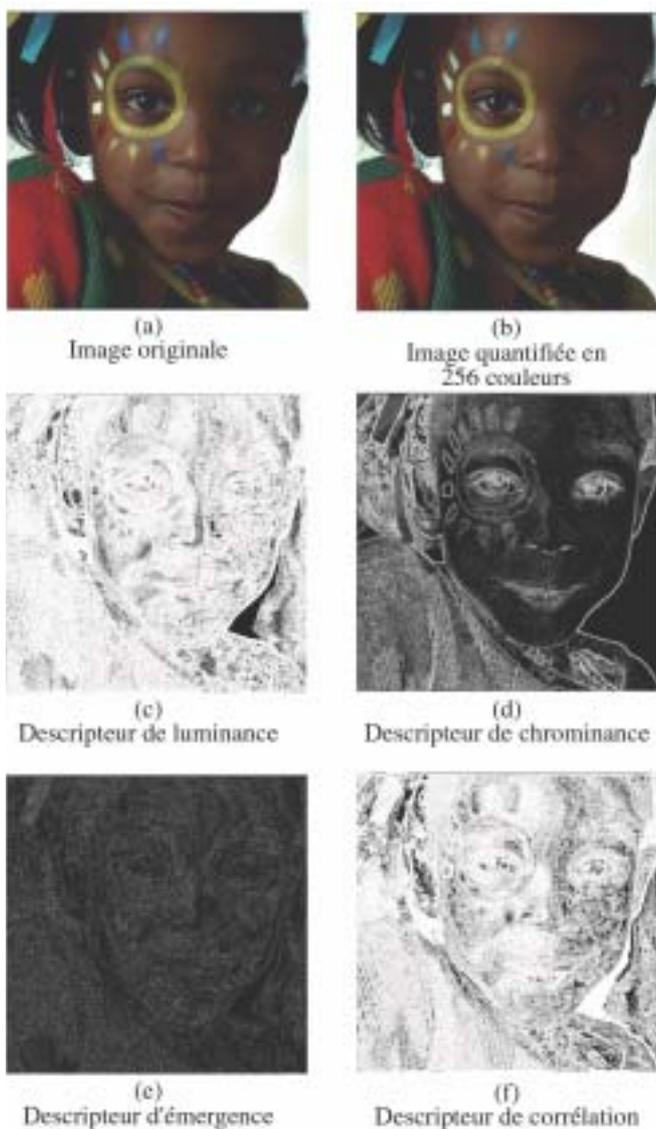


Figure 20. Images obtenues avec les différents descripteurs de qualité.

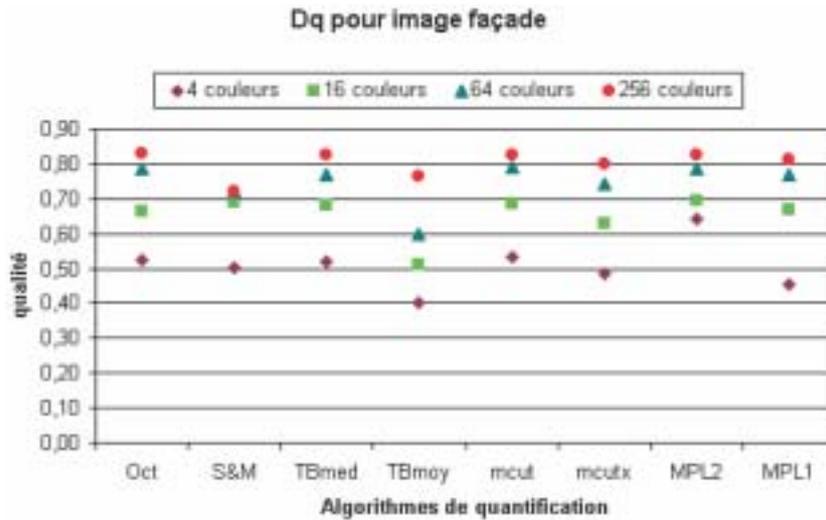


Figure 21. Descripteurs de qualité pour l'image façade.

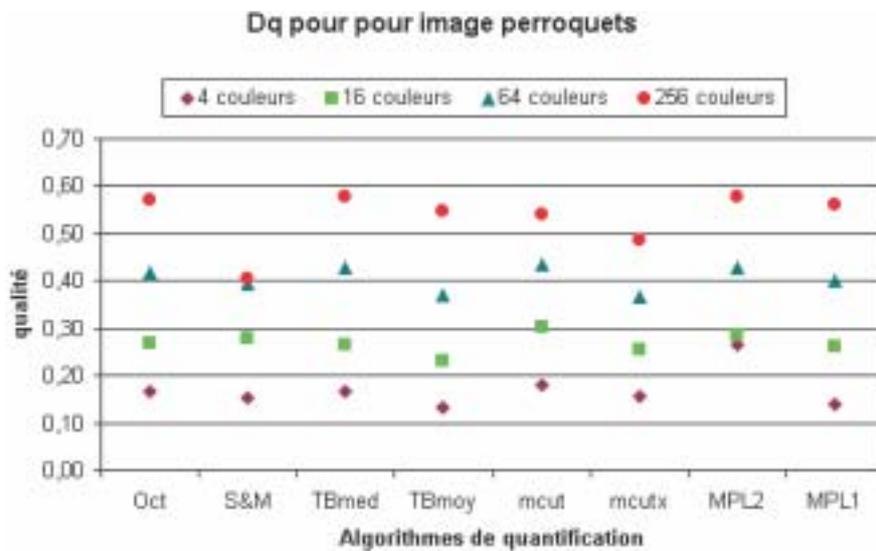


Figure 22. Descripteurs de qualité pour l'image perroquets.

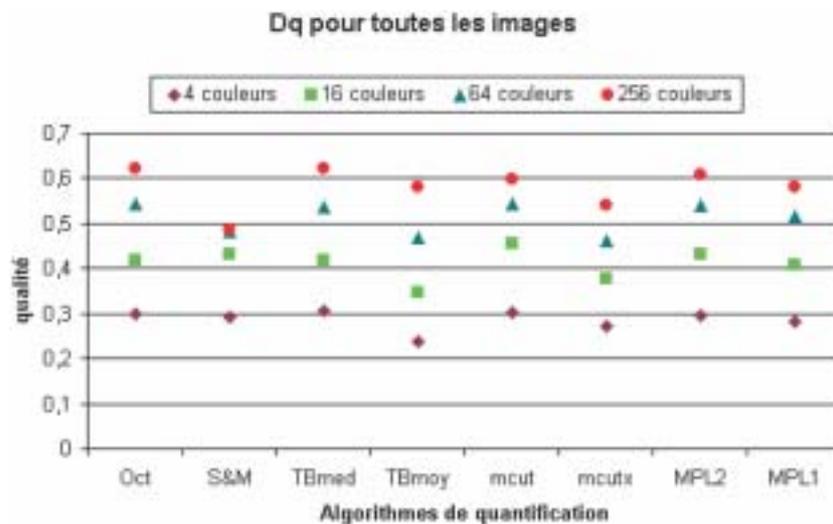


Figure 23. Descripteurs de qualité pour toutes les images.

chovisuelle, permettant de sortir ou de rejeter les observations qui présentent une allure aléatoire. Ceci permet de préserver une statistique correcte relatant la qualité des résultats. La méthode utilisée est basée sur les recommandations ITU-R 500-10 [ITU00]. Cette méthode exploite le MOS et l'écart-type associé et contrôle deux variables que sont les observateurs et les présentations. Un descriptif détaillé peut être trouvé dans le texte de la recommandation ITU.

La figure 24 donne d'une façon graphique les résultats des observations de 5 observateurs parmi les 20. L'allure générale montre une certaine cohérence des réponses des observateurs. Ce qui n'est pas contredit par la moyenne de toutes les observations. Il en sort tout de même qu'une majorité des observateurs ont préférés les résultats de la méthode MPL. Néanmoins, toutes les méthodes ayant un score supérieur à 4 sont considérées comme offrant une très bonne qualité de résultats.

La figure 25 reprend les MOS moyens en y ajoutant les écarts-types afin d'avoir une meilleure idée sur la variabilité des résultats.

## 5. Conclusions et perspectives

Cet article aborde le problème, souvent mal posé, de la quantification et plus précisément la quantification couleur qui consiste à réduire le nombre de couleurs d'une image en altérant le moins possible sa qualité. L'intérêt que nous avons porté à ce traitement réside au-delà du problème d'affichage d'images puisqu'il est question de réduction de la complexité de l'image. Cette réduction est nécessaire dans l'optique d'applications comme l'indexation d'images par le contenu où chaque image est représentée par un ensemble d'attributs de bas niveaux. La multitude d'information qu'achemine l'image, ne permettant pas de fonctionner en temps réel, plusieurs approches ont fait appel à la quantification.

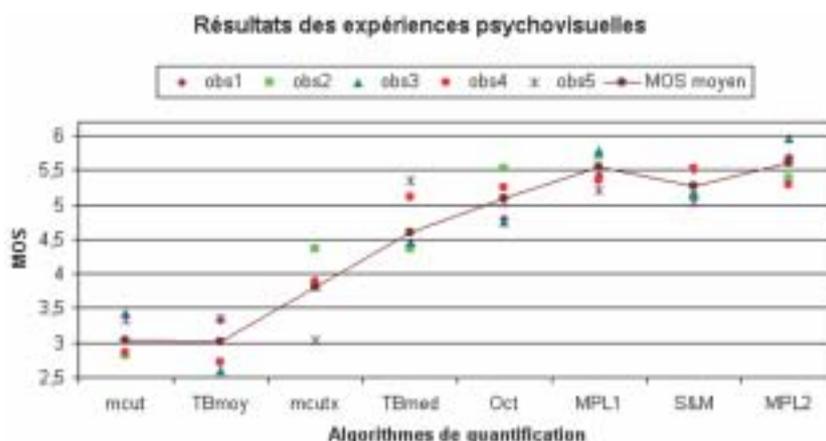


Figure 24. MOS de 5 observateurs parmi les 20.

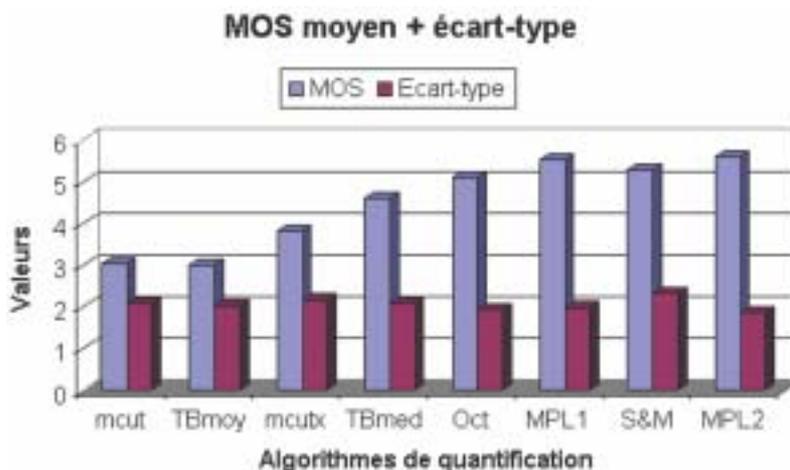


Figure 25. MOS moyen et son écart-type.

Nous avons proposé deux nouvelles approches pour la quantification des images couleurs, l'une basée sur l'utilisation de la transformation du boulanger et l'autre sur la construction d'une matrice des palettes locales. L'étude des performances de ces algorithmes par rapport à des méthodes standards étant une nécessité afin de les valider, nous avons fait appel à deux types de critères quantitatifs. En premier lieu, l'erreur quadratique moyenne (*EQM*) qui représente une erreur ponctuelle permettant de mesurer la distorsion entre l'image originale et l'image quantifiée. Ce critère a montré que les méthodes proposées donnent des résultats satisfaisants, à l'exception d'une légère faiblesse de la variante *TBmoy*. Le second critère représente les descripteurs locaux de qualité qui ont la particularité d'intégrer des informations sur le système visuel humain et par conséquent, sur son mode d'évaluation de la qualité. Ce critère, en plus de sa confirmation des résultats de l'*EQM*, a permis de montrer que, malgré la distorsion apportée à certaines images, ces dernières offrent une meilleure qualité visuelle que d'autres images avec une distorsion moindre.

L'étude s'est terminée par des expériences psychovisuelles permettant d'introduire le jugement humain dans la boucle d'évaluation de la qualité. Cette étude a permis de mettre en place un protocole d'ordonnement dédié à la quantification. Les résultats de cette approche, qui étaient prévisibles, ont montré une contradiction avec les résultats de l'étude objective et plus particulièrement de l'erreur quadratique moyenne.

Comme conclusion finale, les méthodes proposées, plus spécifiquement *TBmed* et *MPL2*, sont meilleures que les approches standards d'un point de vue structurel et colorimétrique. Leur robustesse a été démontrée grâce à la campagne d'évaluation que nous avons menée.

La complexité des algorithmes représente l'un des critères majeurs lors du choix d'une méthode de quantification. Dans cette étude, les algorithmes étaient codés de différentes façons et sous différentes plate-formes. Cet aspect ne nous a pas permis d'étudier leur complexité et d'établir une table de compromis qualité/complexité. Néanmoins, il fera l'objet d'une nouvelle étude en y intégrant de nouvelles approches de quantification.

À partir du travail décrit dans cet article, deux perspectives s'ouvrent à nous. Tout d'abord, l'extension des méthodes de quantification proposées à la vidéo. En effet, la vidéo, tout comme l'image, nécessite une phase de réduction de complexité afin de pouvoir être traitée. Cette réduction concerne la réduction du nombre de couleurs mais aussi la réduction des redondances des trames dans un même plan. Ce travail permettra entre autre la mise en place de la notion de contexte colorimétrique similaire. La seconde perspective concerne l'intégration de critères psychovisuels dans la structure des algorithmes de quantification proposés. Cette intégration permettra l'appel d'une procédure de jugement préalable de la qualité à la fois lors de la sélection des couleurs de la palette finale et lors du remplacement des couleurs dans l'image finale.

## Références

- [AA67] V.I. ARNOLD, A. AVEZ, *Problèmes ergodiques de la mécanique classique*, Gauthier-Villars, Paris, 1967.36
- [BA91] R. BALASUBRAMIAN, J. ALLEBACH, A new approach to palette selection for color images, *Journal of Imaging Technology*, 17(6) : 284-290, Dec. 1991.
- [BAB94] R. BALASUBRAMIAN, J. ALLEBACH, C. A. BOUMAN, Color image quantization with use of a fast binary splitting technique, *Journal of the Optical Society of America*, 11(11) : 2777-2786, Nov. 1994.
- [BB97] J. P. BRAQUELAIRE, L. BRUN, Comparison and optimization of methods of color image quantization, *IEEE Trans. on Image Processing*, 6(7) :1048-1052, 1997.
- [Bed98] L. BEDAT, *Aspects psychovisuels de la perception des couleurs. Application au codage d'images couleur fixes avec compression de l'information*, Thèse de doctorat, Université de Nantes, octobre 1998.
- [Bil65] P. BILLINGSLEY, *Ergodic Theory and Information*, John Wiley & Sons Inc., New-York, 1965.
- [BM00] L. BRUN, M. MOKHTARI. Two high speed color quantization algorithms, In *Proc. of CGIP'2000*, Saint-Étienne, 2000.
- [BO91] C. BOUMAN, M. ORCHARD, Color quantization of images, *IEEE Trans. on Signal Processing*, 39(12) :2677-2690, Dec. 1991.
- [BT02] L. BRUN, A. TRÉMEAU, Quantization, In *chapter of Digital Color Imaging Handbook*, CRC Press, gaurav sharma edition, 2002.
- [Cha98] C. CHARRIER, *Vers l'optimisation statistique et perceptuelle de la qualité pour la compression des images couleur par quantification vectorielle*, Thèse de doctorat, Université Jean Monnet Saint-Etienne, 1998.
- [CKK94] X. CHEN, R. KOTHARI, P. KLINKHACHORN. Reduced color image based on adaptive palette color selection using neural networks, In *IEEE Proceedings*, pages 555-558, 1994.
- [CNW94] S. S. CHAN, R. NERHEIM-WOLFE, An empirical assessment of selected color-quantizing algorithms, In *Human Vision, Visual Processing, and Digital Display IV*, volume 2179, pages 298-309, 1994.
- [CS01] P. CORNU, A. SMOLARZ, Caractérisation de la signature textuelle d'une image. In *Proc. du 18<sup>ème</sup> colloque GRETSI sur le traitement du signal et des images*, Toulouse - France, septembre 2001.
- [CS02] P. CORNU, A. SMOLARZ, Caractérisation d'images par textures associées, *Traitement du Signal*, 19(1) :29-35, 2002.
- [CTM94] NAVID CHADDHA, WEE-CHIEW TAN, TERESA H. Y. MENG, Color Quantization of Images Based on Human Visual Perception, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-94)*, 5 :89-92, April 1994.
- [Dix91] S. S. DIXIT, Quantization of color images for display/printed on limited color output devices, *Computers and Graphics*, 15(4) : 561-568, 1991.
- [GP88] M. GERVAUTZ, W. PURGATHOFER, A simple method for color quantization: Octree quantization, *New Trends in Computer Graphics*, pages 219-231, 1988.
- [GP90] M. GERVAUTZ, W. PURGATHOFER, A simple method for color quantization: Octree quantization, *Graphics Gems*, pages 287-293, 1990.
- [Hec80] P. S. HECKBERT, Color Image Quantization for Frame Buer Display, B.S. thesis, Architecture Machine Group, MIT, mai 1980.
- [Hec82] P. S. HECKBERT, Color Image Quantization for Frame Buer Display, *ACM Computer Graphics*, 16(3) :297-307, 1982.
- [ITU00] ITU., Itu-r recommendation bt.500-10: Methodology for the subjective assessment of the quality of television pictures, Technical report, ITU, Geneva, Switzerland, 2000.

- [KPH+98] J. KETTERER, J. PUZICHA, M. HELD, M. FISCHER, J. M. BUHMANN, D. FELLNER, On spatial quantization of color images, In *Proc. of European Conference on Computer Vision*, pages 563-577, Freiburg, Germany, 1998.
- [Loz98] V. LOZANO, *Contribution de l'analyse d'image couleur au traitement des images textile*, Thèse de doctorat, Université Jean-Monnet, Saint-Étienne, janvier 1998.
- [LRF00] M. C. LARABI, N. RICHARD, C. FERNANDEZ, A Fast Color Quantization Using a Matrix of Local Pallets, In *IEEE Applied Imagery Pattern Recognition*, pages 136-140, Washington, USA, octobre 2000.
- [LRF01] M. C. LARABI, N. RICHARD, C. FERNANDEZ, Une nouvelle méthode de quantification sous contraintes colorimétriques, In *GRETSI'2001*, Toulouse, France, septembre 2001.
- [LRF02] M. C. LARABI, N. RICHARD, C. FERNANDEZ, A new quantification method under colorimetric constraints. In *IS&T Conf. on Color in Graphics, Image and Vision*, pages 412-415, Poitiers, France, 2002.
- [Mc102] D. MCINTYRE, *Colour Blindness: Causes and Effects*, Dalton Publishing, chester, 2002.
- [MLSC04] C. MONTAGNE, S. LELANDAIS, A. SMOLARZ, P. CORNU, Adaptive color quantization using the baker's transform, In *Proc. of CGIV 2004*, Aachen, Germany, April 2004.
- [MLT+04] C. MONTAGNE, S. LELANDAIS, J. TRIBOULET, A. SMOLARZ, E. COLLE, New features color for indoor processing, In *Proc. of SETIT 2004*, Sousse, Tunisia, March 2004.
- [OB91] M. T. ORCHARD, C. A. BOUMAN, Color quantization of images, *IEEE Transactions on Image Processing*, 39(12): 2677-2690, December 1991.
- [RT98] Y. RUBNER, C. TOMASI, Comparing the EMD to other dissimilarity measures for color images, In *DARPA Image Understanding Workshop*, pages 331-339, Monterey, CA, 1998.
- [SC02] A. SMOLARZ, P. CORNU, A color image classification by means of image transformations, In *Proc. of CGIV'2002*, Poitiers, France, June 2002.
- [ST97] G. SHARMA, H.J. TRUSSEL. Digital color imaging, *IEEE Transaction on Image Processing*, 6(7) :901-932, July 1997.
- [TCF97] A. TRÉMEAU, C. CHARRIER, É. FAVIER, Quantitative description of image distortions linked to compression schemes. In *Int. Conf. on the Quantitative Description of Materials Microstructures*, Varsovie, Pologne, 1997.
- [TDF96] A. TRÉMEAU, É. DINET, É. FAVIER, Measurement and display of color image differences based on visual attention, *Journal of Imaging Science and Technology*, 40(6) : 522-534, 1996.
- [TL95] A. TRÉMEAU, B. LAGET, Quantification couleur et analyse d'images, *Traitement du signal*, 12(1) : 1-29, 1995.
- [Tré98] A. TRÉMEAU, *Analyse d'image couleurs: du pixel à la scène*, Habilitation à diriger de recherche, Université Jean Monnet de Saint-Étienne, mai 1998.
- [Xia97] Z. XIANG, Color image quantization by minimizing the maximum intercluster distance, *ACM Transactions on Graphics*, 16(3): 260-276, juillet 1997.





Mohamed-Chaker **Larabi**

Mohamed-Chaker Larabi a reçu son DEA en automatique industrielle de l'Université de Lyon en 1999 et son Doctorat en électronique de l'Université de Poitiers en 2002. Il est actuellement Maître de Conférences au sein de cette même Université et est membre du laboratoire SIC. Ses intérêts de recherches sont concentrés sur l'évaluation de la qualité, les modèles d'apparence couleur, la compression d'images (JPEG2000) et l'indexation de vidéo par le contenu.



Christophe **Montagne**

Christophe Montagne a obtenu le titre d'Ingénieur-Maître en Génie des Systèmes Industriels de l'Université d'Evry en 1999. Après un DEA de Réalité Virtuelle et Maîtrise des Systèmes Complexes, il mène maintenant une thèse au LSC. Ses thématiques de recherche concernent l'intégration de la couleur dans des traitements d'images classiques.



Christine **Fernandez-Maloigne**

Christine Fernandez-Maloigne a reçu son DEA et son Doctorat de l'Université Technologique de Compiègne respectivement en 1986 et 1989. Elle est actuellement Professeur à l'Université de Poitiers et responsable de l'équipe ICONES du laboratoire de SIC. Son travail est concentré sur le traitement d'images couleur, les modèles d'apparence couleur et l'évaluation de qualité.



Sylvie **Lelandais**

Sylvie Lelandais a soutenu son doctorat de troisième cycle en automatique et traitement du signal à l'Université de Technologie de Compiègne en 1984. De 1985 à 1990, elle a travaillé à l'IRESTE à NANTES. Elle a maintenant rejoint le Laboratoire Systèmes Complexes de l'Université d'Evry où après avoir mené des travaux dans le domaine du « Shape from texture », elle s'est orientée vers la vision en robotique, l'imagerie couleurs et, plus récemment sur la biométrie et l'analyse de visages. Son habilitation à diriger des recherches soutenue en 2002 portait sur l'analyse de textures et la vision.



Philippe **Cornu**

Philippe Cornu a soutenu sa thèse de troisième cycle à l'Université de Montpellier et sa thèse d'Etat à l'UTC. Il est enseignant-chercheur à l'UTT depuis 1997 où il enseigne la programmation et la sécurité des systèmes d'information. Ses travaux de recherche au sein de L'ISTIT / M2S concernent l'analyse d'images.



André **Smolarz**

André Smolarz est enseignant-chercheur à l'Université de Technologie de Troyes au sein de l'équipe Modélisation et Sécurité des Systèmes (M2S) de l'Institut des Sciences et Technologies de l'Information de Troyes (ISTIT - CNRS FRE 2732). Ses activités de recherche portent sur la modélisation des textures couleur et la reconnaissance statistique des formes.