

CODAGE D'IMAGES PAR UNE METHODE A DEUX COMPOSANTES
IMAGE CODING WITH A TWO-COMPONENT METHOD

J. RODRIGUEZ , D. BARBA

LABORATOIRE D'AUTOMATIQUE , I.N.S.A.
AVENUE DES BUTTES DE COESMES - 35043 RENNES CEDEX

RESUME

Nous présentons une méthode de codage d'images du type télévision ou vidéotexte, basée sur un codage séparé des composantes par "plateau" et de "texture". Le taux de compression recherché est élevé (de l'ordre de 1 bit/point) tout en assurant une bonne qualité visuelle lors de la restitution.

Le modèle d'images utilisé est un modèle à deux composantes additives qui permet de tenir compte des propriétés différentes des deux caractéristiques fondamentales du signal d'image : les contours d'une part, la texture d'autre part. La première composante appelée composante par plateau I_P supporte l'information de valeur moyenne locale, constante ou à variations simples (linéaires), et de saut brusque de valeur moyenne (contours). La composante de texture I_T caractérise les variations locales du signal d'image autour de la valeur moyenne (donnée par la composante par plateau).

L'extraction de la composante par plateau se fait par un algorithme de détection et suivi de contours afin de segmenter l'image en zones homogènes connexes fermées. Les éléments de contour sont ensuite approchés par des segments de droite délimitant des zones de forme polygonale fermée.

Après découpage des polygones en polygones convexes, on élabore les descripteurs associés qui permettent de reconstruire de façon approchée la composante par plateau \hat{I}_P . La composante de texture est obtenue par différence entre l'image originale I et la composante par plateau codée \hat{I}_P . Elle est codée par un système de codage par blocs du type B.T.C. où les formes des blocs sont les polygones décrits par la composante par plateau.

SUMMARY

We described here a picture coding method for still images transmission and videotext purposes which is based on a separate coding scheme of both "plateau" component and "texture" component. The desired compression ratio is high (about 1 bit/pel is needed) while conserving a good visual quality. The image model we use involves two additive components. It takes into account the properties of the two fundamental characteristics of the image signal : edges and texture. The first component called I_P or "plateau" component, contains informations about local-mean value. This one is supposed to be locally constant or involving simple (linear) variations, together with informations related to abrupt changes in gray levels (edges). Texture component I_T represents the local variations of the image signal apart from its mean value.

The "plateau" component is extracted by an edge-detection and contour following algorithm which splits the image into connected closed homogeneous regions.

The edge detection and contour-following method employs oriented gradient detectors. After that, we proceed with thresholding, thinning, and linking algorithms including elimination of short edge chains. Then this process is completed by a region boundary construction algorithm. Finally chains of edge elements are fitted with linear segments which split the image into closed polygonal regions.

After segmentation into convex polygons, we computed descriptors for describing the plateau component inside each of them. Texture component is obtained by subtracting the decoded plateau component \hat{I}_P from the original picture signal I . Texture is coded by a bloc truncation coding scheme where the size and shape of the blocks are the preceding convex polygons.



I - INTRODUCTION

De nombreuses méthodes de codage ont été proposées pour réduire le débit nécessaire à la transmission des images fixes ou animées [1], [2]. Dans le cas d'images fixes monochromes de type photographique ou dans le cas du codage intra-image pour les images de télévision, on peut grosso modo classer les systèmes de codage en quatre familles.

Les systèmes prédictifs à codage point par point (MICD) ne permettent pas encore actuellement de descendre à moins de 3.5 bits par point en imposant un critère de bonne qualité de restitution (défauts justes perceptibles) [3]. Encore faut-il utiliser dans ce cas une double adaptativité, pour le prédicteur et pour le quantificateur (MICDA). Ils ont pour principal intérêt leur relative simplicité si l'adaptativité reste limitée à une détection grossière des contours.

Les systèmes de codage par transformation [1] utilisent une transformation linéaire inversible (Fourier, Haar, Hadamard, Cosinus,...) appliquée à des ensembles de points d'images structurés en blocs de taille fixe. Ils permettent, pour les meilleurs d'entre eux, un taux de compression supérieur à celui obtenu par les systèmes MICDA si on accepte une qualité faible à moyenne (1 à 2 bits/point). A très bonne qualité cependant, les débits nécessaires sont comparables. Une extension de ces systèmes, le codage hybride, essaie de combiner le codage par bloc et le codage prédictif. Les coefficients obtenus par transformation d'un bloc de points d'image sont codés par un système prédictif [2]. Les débits obtenus sont encore plus faibles, à qualité identique, que précédemment (typiquement 1 bit/point à qualité moyenne).

Une différence essentielle entre ces systèmes concerne la distribution des erreurs de codage. Ils sont surtout très localisés en MICD alors qu'ils sont distribués spatialement dans les systèmes de codage par transformation.

Dans une troisième classe de systèmes de codage, appelés systèmes B.T.C. [4], le codage est également effectué par blocs mais sans transformation. Avec des stratégies adaptatives de codage suivant la structure du signal d'image dans le bloc, pour une qualité moyenne à bonne, le débit nécessaire est de l'ordre de 2 bits/point.

L'optimisation de ces systèmes de codage, nécessaire à l'obtention d'un taux de compression le plus grand possible, bute sur la difficulté de s'adapter simultanément aux caractéristiques du signal d'image d'une part, et aux caractéristiques de la perception

visuelle des défauts de codage d'autre part. A supposer que la perception visuelle humaine soit bien connue, de par sa nature, seul un gain énorme dans la réduction de la dynamique de l'erreur de prédiction (gain de 90 à 95% !) peut amener les systèmes MICDA à des débits de l'ordre de 1 bit/point avec une bonne qualité. A ce débit, les systèmes de codage par blocs de taille fixe, avec ou sans transformation, sont limités dans leurs performances par le fait que le nombre de stratégies différentes de codage en fonction du contenu du bloc, doit rester faible.

Ces remarques introduisent une quatrième classe de méthodes de codage d'images fixes dont l'idée de base est la modélisation du signal d'image sous la forme de deux composantes additives. L'une comporte des variations lentes (lissées) du signal et de brusques ruptures dans son évolution spatiale, l'autre comporte des variations plus ou moins lentes mais à dynamique faible. La première composante supporte l'information de contour et de valeur moyenne locale du signal d'image, l'autre supporte l'information de texture, contours et textures étant les deux traits caractéristiques des images fixes.

Des formes plus ou moins simplifiées de cette idée avec modèle unidimensionnel ont conduit aux premiers systèmes de codage par plages et au système de codage par composante de Schreiber [5] avec séparation par filtrage linéaire des deux composantes. Des systèmes plus élaborés, avec traitements unidimensionnels (ligne par ligne) et extrapolation de ligne à ligne ont été proposés et mis en oeuvre plus récemment [6] et [7]. Cependant les performances sont limitées par le fait que les prétraitements étant uniquement unidimensionnels, contours détectés horizontalement uniquement, le rapport qualité de codage-débit peut être encore amélioré.

Le système de codage que nous proposons repose sur une segmentation de l'image en zones homogènes fermées. Elle est basée donc sur un modèle contour-texture. Un tel modèle a été utilisé en codage d'images fixes dernièrement par M. KOCHER [8]. Les taux de compression atteints sur des images de complexité faible à moyenne varient de 10 à 25 pour une qualité variant de très moyenne à faible. La perte de qualité obtenue provient en bonne partie de la méthode de codage de la composante de texture. La méthode de segmentation que nous avons utilisée est basée entièrement sur la détermination des points de contour permettant de segmenter l'image, alors que KOCHER utilise une méthode de segmentation initiale par croissance de régions et une segmentation finale par fusionnement des régions. La description des contours uniquement sous forme polygonale permet de synthétiser très simplement la première composante (composante par



CODAGE D'IMAGES PAR UNE METHODE A DEUX COMPOSANTES

IMAGE CODING WITH A TWO-COMPONENT METHOD

J. RODRIGUEZ , D. BARBA

plateau).

La seconde composante (texture + erreurs de codage de la première composante) est codée sous forme B.T.C. avec sous-échantillonnage adaptatif et dont la forme des blocs est directement donnée par la forme polygonale convexe des plateaux. Le principe du système de codage est résumé sur la figure 1. L'objectif d'un tel système de codage d'images fixes est d'atteindre moins d'un bit par point d'image avec une bonne qualité. L'algorithme de codage, du fait de l'analyse de l'image, est compliqué et demande donc des moyens de calcul puissants. Par contre, le décodage à la réception est très simple. Cette méthode permet aussi d'introduire progressivement l'image sur l'écran, d'abord la première composante qui visualise les structures importantes de l'image (contour et luminance lissés) et la seconde qui permet ensuite d'obtenir les détails plus fins (texture).

II - SEGMENTATION PAR DETECTION DE CONTOUR

La méthode que nous avons retenue pour la segmentation de l'image en zones homogènes est basée sur la détection des contours. Ceux-ci sont intéressants pour les deux raisons essentielles suivantes. Ils correspondent à des ruptures dans les valeurs du signal d'image. D'un point de vue analyse de scène, donc d'un point de vue visuel, ils se situent souvent sur la silhouette de l'image des objets où sur les arêtes des objets (modélisés sous la forme de facettes planes). A ce titre ils doivent nécessairement, s'ils sont significatifs, être tous inclus dans les frontières des régions connexes obtenues après segmentation d'une image, quelque soit la méthode utilisée.

La suite d'algorithmes que nous avons utilisé et que nous décrivons maintenant a pour objet l'extraction des contours et leur fermeture. Ils délimiteront ainsi des régions fermées, formées de point 8-connexes. Les contours seront ensuite approchés sous la forme de segments de droite.

Les algorithmes mis en oeuvre sont essentiellement de type local et les principes qu'ils utilisent ne sont pas très novateurs généralement, bien que dans certains cas, pour la fermeture des contours et la recherche systématique non redondante des arcs formant les frontières des régions, les méthodes mises en oeuvre sont originales.

1) Détection des points contours

Nous avons suivi une démarche analogue à celle décrite par NEVATIA et BABU [9]. On cherche d'abord à

déterminer les points d'images pour lesquels le gradient orienté maximal est de module supérieur à un certain seuil. On utilise pour cela, 8 masques de gradient de taille 5x5 correspondant à 8 directions réparties uniformément sur le domaine angulaire $[0, 180^\circ]$. La résolution angulaire est donc égale à 22.5° (figure 2). Pour chaque point d'image $M(m,n)$, on mémorise la valeur maximale du gradient et la direction associée.

La simple comparaison du module du gradient à un seuil conduirait le plus souvent à obtenir des contours épais. Un algorithme de seuillage avec amincissement permet d'obtenir des contours tous d'épaisseur unité. Un élément de contour est présent en un point d'image $M(m,n)$ si les 3 conditions suivantes sont réunies :

a) Le gradient maximum $G_{\max}(m,n)$ est de module supérieur à un seuil fixé a priori.

b) Si $M'(m,n)$ et $M''(m,n)$ sont les deux points 8-voisins de $M(m,n)$, et situés de part et d'autre dans la direction orthogonale à θ_{\max} , $|G_{\max}(m,n)|$ est supérieur aux modules des gradients maximums des points M' et M'' .

c) Les directions θ'_{\max} et θ''_{\max} des gradients maximums aux points M' et M'' sont voisines (à ± 45 degrés) de la direction θ_{\max} du gradient maximum au point M .

Sous ces 3 conditions, non seulement M est un point contour mais M' et M'' ne sont pas points contour. Cet opérateur d'amincissement produit des chaînes de points contour de connexité 8 et dont les résultats sont indépendants de l'ordre d'examen des points.

2) Chainage des points contour

L'algorithme précédent produit des points contour distribués dans le plan de l'image. Il est nécessaire de lier ces points de façon à former des chaînes de contour. Nous avons utilisé la méthode décrite par NEVATIA [9]. Elle consiste à déterminer dans le cas général les deux points 8-voisins de chaque point contour tel que, en progressant le long du contour, on rencontre la suite P, M_c, S , où M_c est le point contour courant, P son prédécesseur, S son successeur. Le principe en est le suivant. Parmi les points 8-voisins d'un point contour M_c , seuls 3 points : S_1, S_2, S_3 , sont choisis comme successeurs potentiels de M_c (P_1, P_2, P_3 sont les prédécesseurs potentiels), le choix étant lié à la direction du gradient maximum en M_c (figure 3). On construit le sous-ensemble $S = \{S_i\}$ de successeurs à partir de $\{S_i; i = 1,3\}$ satisfaisant les critères suivants :

a) S_j est un point contour

b) $|\theta_c - \theta_{S_k}| < \Delta\theta$ (seuil fixé à 45°)



Trois cas sont à considérer suivant le nombre k de points retenus :

$k = 1$: un seul successeur est retenu (figure 4-a)

$k = 2$: - si le point courant n'est pas un point de bifurcation (figure 4-b), le successeur retenu est le point le plus proche au sens de la distance Euclidienne.

- si le point courant M_C est un point de bifurcation (figure 4-c), il a deux successeurs et M_C est un point fourche (noeud).

$k = 3$: l'examen des différentes possibilités dont certaines sont représentées (figure 4-d), conduit à ne retenir que deux successeurs et M_C est un point fourche.

Les 2 successeurs d'un point fourche sont classés suivant l'importance de leur gradient. Ils sont les points de départ respectivement d'une branche principale et d'une branche secondaire. Les extrémités des chaînes de contour n'ont pas, soit de prédécesseur, soit de successeur.

3) Fermeture des contours

La présence de texture, et en particulier de texture structurée, l'échantillonnage de l'image, la discrétisation angulaire des orientations pour le calcul des gradients et l'algorithme d'amincissement peuvent entraîner des ruptures de connexité entre les éléments de contour, ruptures qu'il est nécessaire de combler.

Dans une première étape, nous ne considérerons que le raccordement de chaînes de points contour voisines, séparées par deux points au maximum. Pour chaque point terminal des différentes chaînes de contour on examine dans un voisinage 5×5 ou 7×7 selon le cas, les configurations de points contour existants. Les points contour réintroduits doivent satisfaire les conditions énoncées lors de la procédure d'amincissement des contours et de l'établissement des connexités. Parmi l'ensemble de combinaisons possibles, on choisit celle de moindre coût (fonction dépendante de la distance et des différences d'orientations). Les chaînes de contours sont modifiées en conséquence.

La segmentation de l'image en régions connexes nécessite que tous les contours soient fermés. On peut chercher à utiliser la méthode décrite précédemment pour prolonger les chaînes de contour ouvertes, associée à une méthode de programmation dynamique. Cependant la qualité du prolongement dépend essentiellement de la fonction de coût utilisée et la définition d'une "bonne" fonction de coût valable sur des chaînes de grande longueur n'est pas aisée.

Nous avons préféré chercher à utiliser l'information disponible dans l'image à segmenter pour prolonger et fermer les contours à "grande distance". Pour cela on utilise l'information donnée par le Laplacien calculé suivant la direction θ_{\max} . Soit M_e une extrémité de chaîne de contour. On appelle M_D et M_G les deux points 8-voisins de M_e suivant les directions $\theta_{\max} - \pi/2$ et $\theta_{\max} + \pi/2$ respectivement. $L(M_e)$, $L(M_D)$ et $L(M_G)$ sont les valeurs des Laplaciens orientés aux points M_e , M_D et M_G . Suivant que L_D ou L_G ont même signe que L_e , et qu'alors L_G ou L_D est de signe opposé, on définit un suivi de contour à droite ou à gauche. Une fois le mode de suivi de contour choisi, le successeur d'un point courant M_C est le point M_S le plus à droite (resp. le plus à gauche) par rapport au segment $M_P M_C$ (M_P : prédécesseur de M_C) et tel que l'on ait simultanément :

a) M_S est 8-voisin de M_C et non 4-voisin de M_P

b) $L(M_S)$ est de même signe que $L(M_C)$

c) Si $(M_S)_G$ est 8-voisin de M_C (resp. $(M_S)_D$), 4-voisin de M_S et situé à gauche (resp. à droite) du segment de contour prolongé $M_P M_C$, alors $L((M_S)_G)$ est de signe opposé à $L(M_C)$, (resp. $L((M_S)_D)$).

Trois situations peuvent être rencontrées.

La chaîne de contour prolongée provisoirement se ferme sur une extrémité de contour de direction compatible (on interdit la connexité d'éléments de contour anti-parallèles) : la chaîne prolongée provisoirement devient une chaîne de contour définitivement.

La chaîne de contour provisoire se ferme sur un contour. La connexion est établie de façon à respecter un certain nombre de conditions. Un noeud est créé sur le contour rencontré et la chaîne de contour provisoire devient définitive.

La chaîne de contour provisoire se ferme sur elle-même. Des retours en arrière successifs suivis de nouvelles prolongations sont entrepris jusqu'à rencontre d'une fermeture valide.

4) Construction des régions

A ce stade, l'information disponible sur les contours se présente sous la forme de deux fichiers : le fichier prédécesseur et le fichier successeur. Ceux-ci permettent, entre autre, de suivre un contour entre deux noeuds consécutifs N_i et N_{i+1} . Nous cherchons à construire automatiquement les régions segmentant le plan de l'image à partir de la connaissance des chaînes de contour. On considérera que la frontière d'une région est décrite dans le sens trigonométrique inverse. - On appelle arc toute chaîne orientée de points contour connexes telle que ses extrémités soient des noeuds



CODAGE D'IMAGES PAR UNE METHODE A DEUX COMPOSANTES

IMAGE CODING WITH A TWO-COMPONENT METHOD

J. RODRIGUEZ , D. BARBA

et qu'aucun de ses points intérieurs ne soit un noeud.

- Les régions étant toutes fermées, la bordure du plan de l'image est considérée comme chaîne de points contour également.

- Les quatre sommets du plan rectangulaire de l'image sont considérés comme des noeuds.

- On appelle arc intérieur tout arc dont tous les éléments, sauf les extrémités au plus, sont intérieurs à l'image.

- On appelle arc extérieur, tout arc dont tous les éléments sont sur la bordure de l'image.

Nous avons utilisé la méthode suivante qui permet de construire progressivement chacune des régions en déterminant la suite ordonnée des arcs orientés formant sa frontière. Les informations élaborées et utilisées par l'algorithme sont les suivantes.

a) La table des noeuds N_i : coordonnées, arcs entrant, arcs sortant, marqueur d'utilisation complète du noeud.

b) La table des arcs A_j : noeud de départ N_D , noeud d'arrivée N_F , code de Freeman de la chaîne au point N_D pour le successeur. Indicateur d'arc intérieur.

c) La table des régions R_k : suite ordonnée des arcs connexes.

Pour chaque région R en cours de construction, si $(N_F)_e$ est le noeud d'arrivée du dernier arc A_e de la suite décrivant R, on suit la chaîne de points contour la plus à droite du segment d'arrivée en $(N_F)_e$ et on marque $(N_F)_e$ que l'on vient de quitter. Si l'un au moins des deux noeuds d'un arc n'est pas marqué, alors c'est un arc parcouru pour la première fois et il est rajouté à la table des arcs. La table des noeuds est complétée à chaque fois que l'on parcourt un arc. Une région est fermée dès que : $(N_F)_e = (N_D)_i$.

La poursuite ou la fin de la construction des régions se fait en examinant la table des noeuds et en utilisant le principe suivant. Tout arc extérieur n'est parcouru qu'une seule fois car il n'est frontière que d'une seule région. Tout arc intérieur est parcouru une fois dans un sens $N_D \rightarrow N_F$ (sens direct), une fois dans le sens $N_F \rightarrow N_D$ (sens inverse). Tant qu'il reste un arc intérieur parcouru dans un seul sens, la construction des régions n'est pas terminée. En examinant la table des noeuds, le premier arc intérieur parcouru une seule fois permet, en inversant le sens de son parcours, de déterminer le premier arc de la frontière de la nouvelle région à construire. Le processus de construction démarre à partir de l'un des 4 sommets S_i en parcourant l'arc extérieur, de noeud de départ S_i , dans le sens trigonométrique inverse jusqu'au premier

noeud suivant.

5) Approximation polygonale des contours

On cherche à décrire les frontières des régions sous une forme polygonale. Il suffit donc de décrire chacun des arcs A_j de la table des arcs sous forme d'une suite de segments de droites connexes. Pour cela, nous avons utilisé la méthode proposée par HUNG et al [10] qui détermine dans une première phase les points critiques S_i de l'arc. Ceux-ci sont les points de changement de courbure et sont considérés comme significatifs de la courbe.

Ainsi chaque arc A_j se décompose en une suite de sous-arcs SA_{jk} adjacents dont les extrémités sont soit un point critique P_{jk} soit un noeud $(N_{jD}$ ou $N_{jF})$. Chaque sous-arc SA_{jk} est formé d'une chaîne de points connexes représentant une ligne discrète dans la courbure locale ne peut varier que doucement. L'approximation par segments de droites de chaque sous-arc peut nécessiter des sous-divisions supplémentaires. Nous avons utilisé pour cela l'algorithme itératif de découpage par point de distance d'Euclidienne la plus grande à la droite considérée (si d dépasse un certain seuil fixé a priori).

Ce dernier algorithme termine la phase d'analyse et de segmentation de l'image à coder sous la forme de régions homogènes de formes polygonales. Les informations se présentent sous la forme de trois listes : la liste des sommets noeuds, la liste des arcs et leur approximation polygonale, la liste des régions et la suite des arcs convexes orientés délimitant leur frontière.

III - CODAGE DE LA COMPOSANTE PAR PLATEAU

La composante par plateau supporte l'information de valeur moyenne locale et de rupture de cette valeur moyenne. Les ruptures sont toutes situées sur les contours. Les variations de la valeur moyenne du signal d'image à l'intérieur d'une région sont faibles. Nous avons choisi de représenter ces variations de la composante par plateau par des variations linéaires avec les coordonnées spatiales des points d'image. De plus, de façon à ce que le système de décodage soit très simple, les régions segmentées, obtenues sous formes polygonales quelconques, sont elles-mêmes segmentées en un certain nombre de régions de formes polygonales convexes appelées facettes.

Du fait de la forme polygonale quelconque des régions obtenues, le découpage d'un tel polygone en un nombre minimal de polygones convexes n'est pas simple. Un exemple réel est représenté sur la figure 5 et son



CODAGE D'IMAGES PAR UNE METHODE A DEUX COMPOSANTES
IMAGE CODING WITH A TWO-COMPONENT METHOD

J. RODRIGUEZ , D. BARBA

découpage associé. Le principe d'un tel découpage est le suivant. On établit la liste ordonnée $\{L S\}$ des sommets en marquant les sommets concaves. On cherche dans cette liste toutes les suites de sommets de type :

$$\{L S\}^k = \{S C_i^k, S O_{i+1}, \dots, S C_{i+j}^{k+1}\} \quad \text{où :}$$

- $S C_i^k$ est le $i^{\text{ème}}$ sommet qui est simultanément le $k^{\text{ème}}$ sommet concave,

- $S O_{i+1}$ est le $(i+1)^{\text{ème}}$ sommet qui est un sommet convexe,

- $i+j$ est calculé module N , nombre de sommets du polygone à découper. L'indice j est obligatoirement ≥ 2 .

Le choix du polygone convexe extrait est fait à partir du classement décroissant suivant. Choisir le premier polygone qui, après découpage, transforme dans la liste $\{L S\}$ du polygone restant ses deux sommets concaves en sommets convexes. Sinon choisir le premier polygone qui, après découpage, transforme dans la liste $\{L S\}$ du polygone restant un de ses deux sommets concaves en un sommet convexe.

L'algorithme épuise ainsi les sommets concaves du polygone initial en un nombre minimum d'itérations et donc de polygones convexes construits. Pour chacun d'eux nous associons un descripteur qui permet de coder et donc de décoder la composante par plateau suivant une méthode utilisée en synthèse d'images [12]. Chaque descripteur contient les informations permettant de retrouver les coordonnées des sommets du polygone convexe et les luminances associées. Le long d'un côté du polygone, appelé arête, la luminance I_p varie linéairement. De même, le long d'une ligne de balayage de l'image (ligne horizontale), I_p varie également linéairement. La détermination de la luminance I_p en un sommet S est obtenue par filtrage médian de quelques points d'images voisins de S , et intérieurs au polygone.

IV - CODAGE DE LA COMPOSANTE DE TEXTURE

La segmentation initiale de l'image obtenue par extraction des contours et leur approximation polygonale, et un modèle à variations linéaires de la composante par plateau, entraînent que celle-ci est codée de façon approximative. Aussi est-il intéressant, sinon nécessaire, que le codage de la composante de texture à coder incorpore les erreurs introduites dans la détermination et le codage de la composante par plateau. C'est pourquoi le codeur comporte un système de décodage de la composante par plateau, identique à celui

rencontré dans le système de réception et de décodage de l'image.

La composante de texture en elle-même est un signal dont la modélisation est assez complexe dans le cas général du fait de la grande variété des textures rencontrées : textures fortement structurées jusqu'aux textures typiquement aléatoires. Dans le cas d'un modèle de texture aléatoire sous forme autorégressive et stationnaire, le système de codage optimal, au sens de l'erreur quadratique moyenne maximale, est un système à transformation Karhunen Loève [11]. Cependant les textures ne peuvent être considérées de même modèle sur des régions différentes. De plus l'utilisation des méthodes de codage par transformation nécessite en pratique des blocs de taille carré.

Pour ces raisons, nous avons mis en oeuvre une méthode de codage par bloc sans transformation [4]. Du fait que les blocs sont à signal homogène, une seule itération est nécessaire pour le codage de la composante de texture. La forme des blocs est celle des polygones convexes. Par point de signal codé, le débit nécessaire est de 1 bit/point. Aussi est-il envisagé de réduire le nombre de points par la mise en oeuvre d'un sous-échantillonnage adaptatif, l'adaptativité dépendant des corrélations horizontales et verticales du signal de texture. A l'intérieur d'un même polygone initial (non convexe a priori) la structure d'échantillonnage reste constante.

V - PRESENTATION DES RESULTATS ET CONCLUSION

L'algorithme complet de codage est en cours d'évaluation sur un ensemble d'images à caractéristiques de contours et de textures assez variées. Les images utilisées sont des images normalisées dans le codage des images de télévision. L'exemple présentée dans ce papier correspond à un médaillon 256-256 extrait de l'image "Fille aux jouets", codée à 8 bits par point. Les photographies 1 à 6 présentent successivement l'image d'origine, l'image des modules de gradients maximums, l'image des points contour ($|G_{\max}| > 2000$), l'image des contours amincis, l'image des contours suivis et enfin l'image des contours fermés. On se rend ainsi compte de la progression des traitements effectués. Le codage de la composante par plateau et de la composante de texture sont en cours d'optimisation. Ce n'est qu'après cette dernière étape que l'évaluation des performances, au niveau qualité visuelle-compression d'information, pourra être entreprise et une comparaison objective avec les autres systèmes de codage pourra être faite.



CODAGE D'IMAGES PAR UNE METHODE A DEUX COMPOSANTES

IMAGE CODING WITH A TWO-COMPONENT METHOD

J. RODRIGUEZ , D. BARBA

BIBLIOGRAPHIE

- [1] A.N. NETRAVALI, J.O. LIMB : "Picture Coding : a review". Proc. of the IEEE, vol. 68, n°3, March 1980, pp. 366-406.
- [2] W.K. PRATT et al : "Image transmission techniques" Academic Press, 1979.
- [3] J. DEWITTE : "Un système de codage MICD adaptatif avec critère psychovisuel de qualité". Thèse de D.I., INSA de Rennes, 1981.
- [4] J. DEWITTE, J. RONSIN : "Original block coding scheme for low bit rate image transmission". Congrès EUSIPCO, Erlangen, Septembre 1983.
- [5] W.F. SCHREIBER, C.F. KNAPP, N.D. KAY : "Synthetic highs : an experimental TV bandwidth reduction system ". J.S. Motion Pic. Tel. Eng., vol. 68, pp. 525-537, Aug. 1959.
- [6] YAN J.K., SAKRISON D.J. : "Encoding of images based on a two-component source-model". I.E.E.E. Trans. On Com., Vol. 25, n° 11, pp. 1315-1322, Nov. 1977 .
- [7] L. FEIJS, L. CHIARIGLIONE : "Image coding by means of two-component source-coding scheme". CSELT Rap. Tech. Vol. VIII, n°2, pp. 83-92, 1980.
- [8] M. KOCHER : "Codage d'images à haute compression basé sur un modèle contour-texture". Thèse de Doctorat E.P.F.L., Lausanne, 1983.
- [9] A.R. NEVATIA, K.R. BABU : "An edge detection, linking and line finding program". Technical Report, I.P.I.-U.S.C., pp. 103-121, 30 Sept. 1978.
- [10] S.H.Y. HUNG, T. KASUAND : "Linear approximation of quantized thin line" in "Pictorial Data Analysis". Ed. R.M. Haralick, NATO ASI Series F, n°4, pp. 15-28.
- [11] A.K. JAIN : "Image data compression : a review" Proc. of I.E.E.E., Vol. 69, n°3, pp. 349-389, March 1981.
- [12] D. BARBA, A. BRUNO, H. TORRES : "Génération d'images synthétiques avec texture en temps réel". I.A.S.T.E.D. Symposium on Applied Informatics, Lille, Mars 1983.

0 DEGRES	22.5 DEGRES
10, 10, 10, 10, 10,	10, 10, 10, 10, 10,
10, 10, 10, 10, 10,	10, 10, 10, 6, 2,
0, 0, 0, 0, 0,	8, 4, 0, -4, -8,
-10, -10, -10, -10, -10,	-2, -6, -10, -10, -10,
-10, -10, -10, -10, -10,	-10, -10, -10, -10, -10,
	45 DEGRES
	10, 10, 10, 10, 0,
	10, 10, 10, 0, -10,
	10, 10, 0, -10, -10,
	10, 0, -10, -10, -10,
	0, -10, -10, -10, -10,

FIGURE 2 : FILTRES DE GRADIENTS 5x5

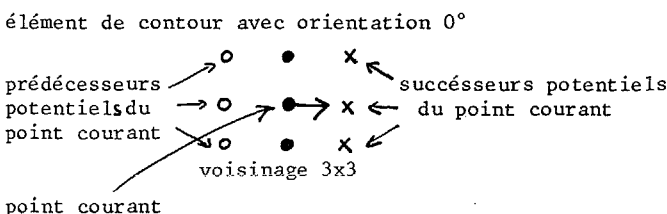


FIGURE 3 : CHOIX DES POINTS PREDECESSEURS ET SUCESSEURS POTENTIELS D'UN ELEMENT DE CONTOUR (EXEMPLE A 0°)

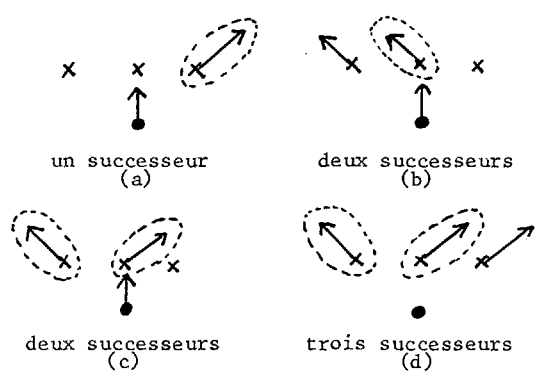


FIGURE 4 : CHAINAGE DES ELEMENTS DE CONTOUR

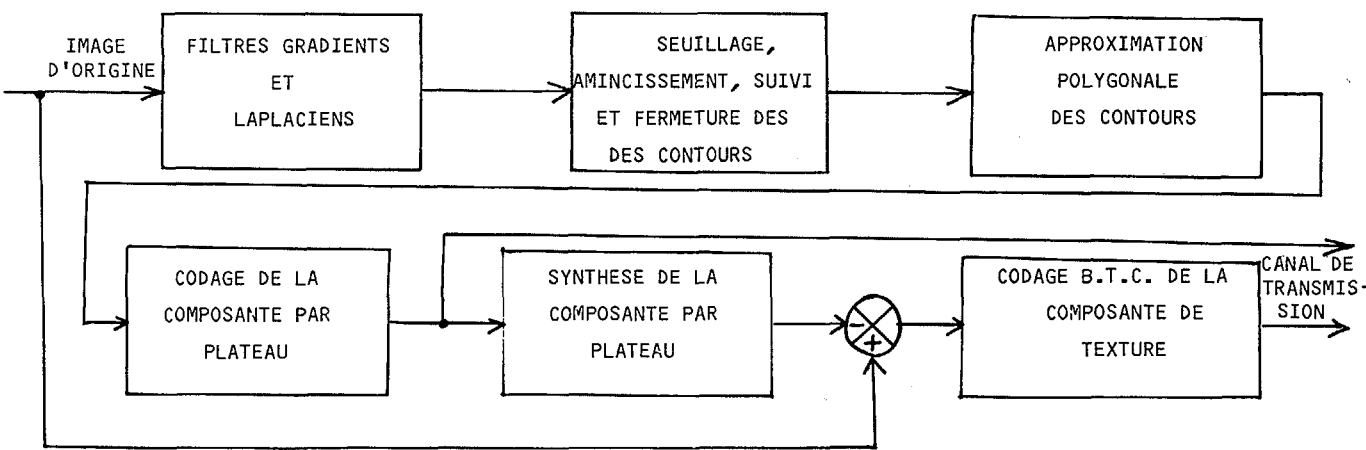


FIGURE 1 : PRINCIPE DU SYSTEME DE CODAGE A DEUX COMPOSANTES

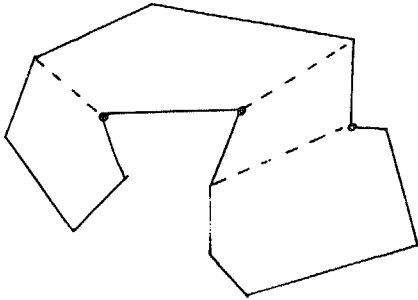
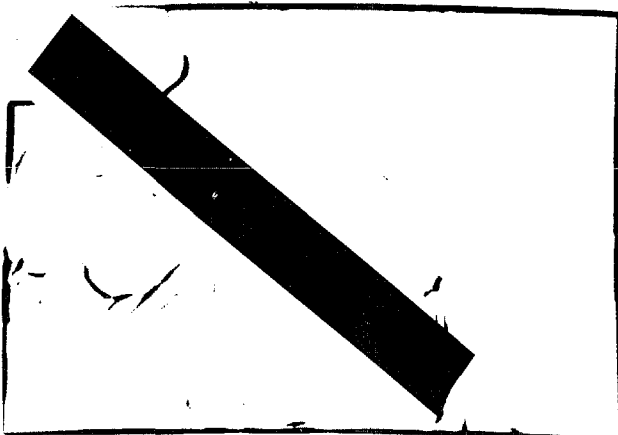


FIGURE 5 : DECOUPAGE D'UN POLYGONE NON-CONVEXE EN POLYGONES CONVEXES



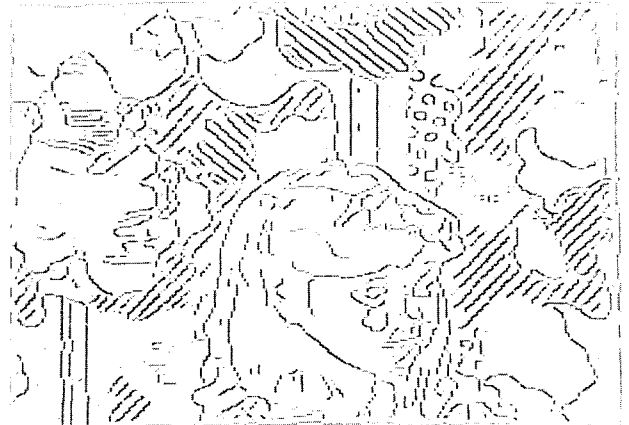
P 1 : Image d'origine (256 x 256 x 8 bits/point)



P 2 : Image des modules des gradients maximums



P 3 : Image des points de forts gradients ($|G_{max}| > 2000$)



P 4 : Image des points contours amincis



P 5 : Image des contours suivis avec chaînage



P 6 : Image des contours fermés