ONZIEME COLLOQUE GRETSI - NICE DU 1er AU 5 JUIN 1987

PREDICTION LINEAIRE PAR BLOC, COMPENSATION DE MOUVEMENT ET FILTRAGE DANS LA BOUCLE DE CODAGE

Gérard EUDE, Jacques GUICHARD
et Marc BARRILLIET-BREAU

CENTRE NATIONAL D'ETUDES EN TELECOMMUNICATIONS
38,40 Avenue du Général LECLERC
92131 ISSY-LES-MOULINEAUX FRANCE

SOMMAIRE

Depuis plusieurs années de nombreux schémas de codage à base de transformées orthogonales (Hadamard, Haar, cosinus...) ont été étudiés pour la transmission d'images fixes ou en mouvement. Les principales applications sont le photovidéotex, la visioconférence, la visiophonie ou la distribution TV.

Nous présentons et comparons ici deux différents types de codage hybride : l'un avec intégration de la compensation de mouvement et l'autre par une technique de meilleures prédictions linéaires des blocs (plusieurs modes de codage sont alors considérés suivant les caractéristiques du bloc et du passé).

Puisque plus généralement l'emploi de plusieurs modes de prédiction est la transposition d'opérations de filtrage (pseudo-convolution) dans le domaine image nous montrons qu'il est donc possible de combiner en partie les deux techniques en réalisant un filtrage à l'intérieur de la boucle dans le cas d'un codage à compensation de mouvement.

I. INTRODUCTION: CODAGE PAR TRANSFORMATIONS

Les transformations orthogonales ont un intérêt puissant pour le codage des images en vue de réduire le débit numérique de transmission, et ce, de par leurs propriétés de décorrélation du signal et leur répartition statistique de l'énergie dans le domaine transformé. Différentes transformations ont été utilisées dans des systèmes de codage d'images- la transformée optimale de décorrelation dite de KARHUNEN-LOEVE ne pouvant être implantée dans des systèmes fonctionnant en temps réel -on utilise des transformées "sous-optimales" qui possèdent des algorithmes de calcul rapide. La transformée en cosinus discrète (TCD) dont la formule de calcul est donnée ci-après réalise un bon compromis entre la complexité de calcul et les performances.

$$F(u,v) = \sum_{x=0,y=0}^{N-1,N-1} C(u)C(v)f(x,y)T(x,u)T(y,v)$$

avec $T(x,u) = \cos(\pi u(2x+1)/2N)$

et $C(0) = 1/\sqrt{2}$ et $C(u) = 1$ pour $u \neq 0$

Pour des raisons pratiques de complexité et théoriques (l'efficacité de décorrélation est liée à la longueur de corrélation dans le signal), la transformation et le codage sont réalisés sur des blocs de N x N pixels après une partition de l'image. Le choix de la taille du bloc dépend de plusieurs paramètres tels que la définition spatiale de l'image, le débit souhaité, la complexité admissible du codage .... En règle générale cette taille se situe entre 8 x 8 et 16 x 16.

Le codage est ensuite réalisé dans le plan transformé et seuls les coefficients significatifs sont réellement transmis après quantification.

D'autres méthodes par classification des blocs permettent d'éliminer systématiquement certaines zones suivant les caractéristiques fréquentielles du bloc (ref 1).

II. SYSTEMES DE CODAGE HYBRIDES

Dans de nombreuses applications destinées aux images en mouvement il existe une forte redondance temporelle (inter-image), aussi est-il intéressant de combiner des techniques de rafraichissement conditionnel (on ne code que les blocs qui se sont modifiés d'une image à l'autre) et de codage différentiel (seules les différences inter-blocs sont codées). (ref 2,3)

II.1 La compensation de mouvement

Dans un codage de type hybride le mode inter-image est un des plus important -et consiste donc en un codage prédictif par bloc- où la prédiction est donnée par le bloc de même position spatiale de l'image précedemment codée:

PBLOC(X0,Y0,t) =ensemble des pixels
f(X0+x,Y0+y,t-1) avec $0 \leqslant x \leqslant N-1$
$0 \leqslant y \leqslant N-1$
X0 et Y0 sont les coordonnées du coin haut gauche du bloc.

Une idée fort séduisante consiste naturellement à essayer d'améliorer cette prédiction spatiale des blocs en prenant en compte une estimation du mouvement des objets dans l'image, c'est à dire en cherchant des vecteurs vx et vy tels que:

PBLOC(X0,Y0,t) = BLOC(X0+vx,Y0+vy,t-1)
avec:
$\sum |BLOC(X0,Y0,t)-PBLOC(X0,Y0,t)|$ minimum sur la fenêtre de recherche.

Une nette amélioration de la qualité des images codées peut-être ainsi obtenue au prix d'une grande quantité de calculs et d'une implémentation difficile. Des algorithmes moins performants mais plus rapides ont été développés qui peuvent être intégrés dès lors que les fréquences de traitement ne sont pas trop élevées (ref 4,5).

Cette méthode qui est basée sur la recherche d'une bonne prédiction du bloc dans le domaine spatial nécessite d'intéger la transformation dans la boucle de codage et d'effectuer une transformation inverse avant le calcul de l'estimation du vecteur mouvement (les quelques essais destinés à faire un calcul de vecteurs mouvement dans le plan transformé conduisent à des algorithmes plus complexes encore...).

Cette compensation de mouvement (par corrélation de bloc) implique la transmission au décodeur du vecteur mouvement pour chaque bloc "compensé".

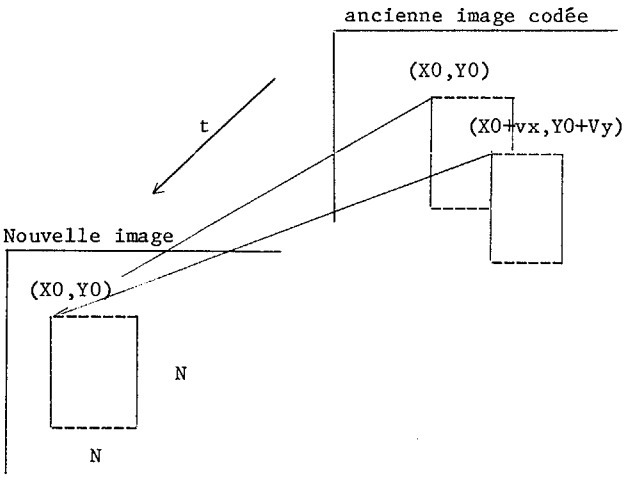

fig 1 : technique de compensation de mouvement par bloc.

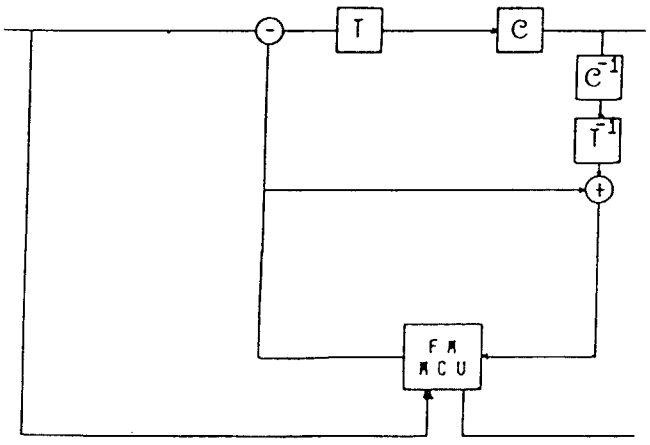

fig 2 : modèle de codage hybride avec compensation de mouvement.

T : Transformation
: Codage
FM : Mémoire d'image
MCU : Compensation de mouvement

II.2 Prédictions dans le domaine transformé.

Cette technique que nous avons voulu évaluer et comparer aux schémas avec compensation de mouvement n'est pas de fait entièrement nouvelle mais consiste en un développement de la technique de codage hybride avec une seule transformation à l'extérieur de la boucle. Dans un tel schéma il n'est la plupart du temps considéré que le codage en mode intra-bloc et en mode inter-bloc pur ce qui ne semble pas optimal du point de vue codage si l'on veut réaliser une bonne prédiction du bloc dans le domaine transformé. Nous avons donc simulé une méthode de codage dans laquelle nous disposons de plusieurs prédictions du bloc à coder.

a) Calcul de la meilleure prédiction linéaire.

Soit F(u,v,t) le bloc transformé on cherche à déterminer une prédiction de la forme:

$$P(u,v,t) = H(u,v).F(u,v,t-1) \text{ pour chaque } (u,v)$$

Il faut dès lors minimiser l'expression:

$$E((F(u,v,t)-P(u,v,t))^2)$$

C'est à dire rendre

$$E(F(u,v,t))^2 + H(u,v).E(F(u,v,t-1))^2 - 2\,H(u,v).E(F(u,v,t).F(u,v,t-1)) \text{ minimum}$$

d'où il vient en supposant F(u,v,t) stationnaire:

$$H(u,v) = E(F(u,v,t).F(u,v,t-1))/E(F(u,v,t))^2$$

Ce calcul étant effectué et cumulé sur plusieurs images (sur les blocs non fixes) on obtient ce résultat fort intéressant que les valeurs de H(u,v) sont souvent proches de 1 pour les basses fréquences et plus faibles pour les hautes fréquences.

Cela nous amène tout naturellement à essayer de faire une différence de modes entre les basses et les hautes fréquences -c'est à dire à considérer des prédicteurs mixtes dans lesquels certaines zones seraient pratiquement codées en intra-bloc (H(u,v)-0) et d'autres en inter-bloc (H(u,v)-1).

Pour compléter tout cela il est nécessaire d'introduire une classification des blocs de l'images -ou plutôt d'introduire un ensemble de prédicteurs possibles (fonctions H)-afin de choisir à chaque fois celui qui minimise réellement pour le bloc considéré l'erreur de prédiction.

b) Détermination d'un ensemble de prédicteurs

Pour déterminer cet ensemble de prédicteurs deux algorithmes ont été essayés et mis en oeuvre avec des notions de distances du type MINKOWSKI:

$$D(X,Y) = \left(\sum_{1}^{n} ABS(Xi-Yi)\right)$$

Algorithme K-means

Etape 1: Initialisation avec un ensemble pré-défini de prédicteurs simples.

Etape 2: Attribution à chaque bloc de la séquence d'apprentissage à l'un des prédicteurs suivant une règle des plus proches voisins (distance euclidienne).

Etape 3: Mise à jour des centroïdes à partir de la nouvelle classification.

Etape 4: Test de fin d'itération, calcul d'une nouvelle distorsion et RETOUR en 2 si la distorsion est supérieure au seuil.

Algorithme à seuil

Ce deuxième algorithme consiste à fabriquer un dictionnaire de prédicteurs avec chaque bloc de la séquence d'apprentissage qui ne peut être assimilé aux précédents du fait d'une trop grande différence.

D'une façon générale, si E=(X1,X2....Xm) est la séquence dont on veut extraire un ensemble de représentants l'algorithme commence par prendre X1 comme représentant de la 1ere classe ensuite de construire une nouvelle classe chaque fois que la distance entre un nouveau vecteur Xi et la plus proche de toutes les classes déja crées est supérieure à un seuil qui dépend du nombre de classes que l'on veut.

Cet algorithme qui n'est pas lié au choix d'un dictionnaire donné a priori donne de meilleurs résultats.

Dans le contexte d'un codage à 300 Kbit/s pour des images de visioconférence, et pour des blocs de taille 8 x 8, le nombre suffisant de classes à considérer est entre 8 et 16.

- EXEMPLES DE CLASSES -

1.00	0.98	0.97	0.95	0.93	0.90	0.75	0.67
0.95	0.93	0.92	0.90	0.88	0.81	0.64	0.54
0.92	0.88	0.86	0.74	0.70	0.61	0.56	0.47
0.62	0.56	0.54	0.49	0.38	0.36	0.34	0.25
0.45	0.37	0.35	0.30	0.26	0.22	0.20	0.17
0.30	0.20	0.20	0.20	0.20	0.16	0.15	0.11
0.25	0.18	0.16	0.14	0.13	0.12	0.10	0.06
0.17	0.12	0.10	0.08	0.07	0.05	0.05	0.05

- Coefficients H6(u,v) -

1.00	0.20	0.16	0.15	0.12	0.11	0.08	0.07
0.76	0.17	0.12	0.10	0.08	0.05	0.00	0.00
0.51	0.14	0.07	0.06	0.05	0.00	0.00	0.00
0.35	0.06	0.05	0.00	0.00	0.00	0.00	0.00
0.30	0.05	0.00	0.00	0.00	0.00	0.00	0.00
0.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.20	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.15	0.00	0.00	0.00	0.00	0.00	0.00	0.00

- Coefficients H2(u,v) -

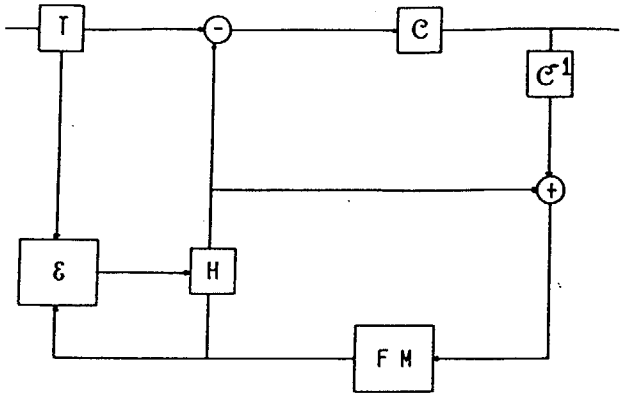

fig 3 : modèle de codage par "meilleures prédictions"

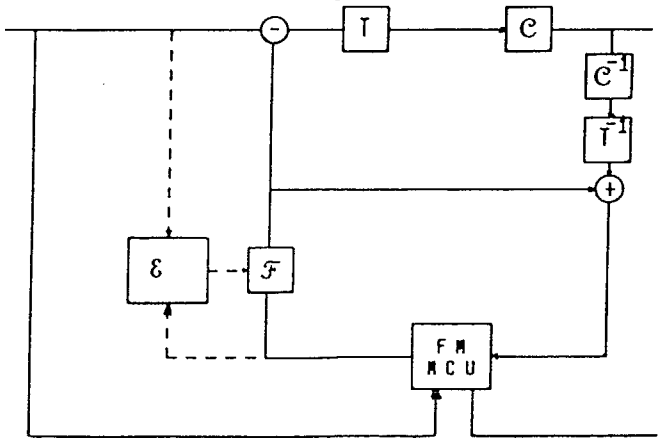

fig 4 : Compensation de mouvement et filtrage dans la boucle

II.3 Description du codeur meilleures prédictions linéaires.

Pour chaque bloc à coder il est donc nécessaire de calculer quel est le meilleur prédicteur à utiliser (de l'intra-bloc pur à l'inter pur en passant par tous les modes "mixtes" définis).

Pour cela nous commençons par effectuer la TCD du bloc entrant, et calculons toutes les prédictions possibles afin de retenir celle qui minimise l'erreur de prédiction.

Etape 1: Calcul de la TCD du bloc: $F(u,v,t)$ $0 \leq u,v \leq N-1$

Etape 2: Calcul des 8 prédictions:

$$\begin{pmatrix} 0.F(u,v,t-1) \\ H1(u,v).F(u,v,t-1) \\ .. \\ .. \\ H6(u,v).F(u,v,t-1) \\ 1.F(u,v,t-1) \end{pmatrix}$$

Etape 3: Calcul des 8 erreurs de prédiction avec un critère du type somme des valeurs absolues.

Etape 4: Choix du meilleur prédicteur et codage de l'erreur de prédiction.

Avec cette technique le choix du prédicteur doit être transmis au décodeur (3 bits par bloc avec un code à longueur fixe pour 8 classes). Le reste du codage: quantification des coefficients, adressage des points, méthodes de régulation... est alors réalisé de la même manière pour les 2 modèles de codage.

II.4 Compensation de mouvement et filtrage dans la boucle de prédiction.

Dans les codeurs hybrides à compensation de mouvement il a été mis en évidence tout l'intérêt que l'on peut tirer à opérer un filtrage linéaire passe-bas avant d'effectuer la différence pour les blocs non totalement fixes, cela revient à réaliser une nouvelle prédiction:

$$PBLOC(X0,Y0,t) = \mathcal{F}(BLOC(X0+vx,Y0+vy,t-1))$$

Il est facile de voir que ce filtrage linéaire s'apparente à une opération de multiplication qui serait effectuée dans le domaine transformée avec le modèle "meilleures prédictions linéaires".

Néanmoins - dès lors qu'il ne s'agit pas d'une transformée de Fourier - l'opération comparable à la multiplication par le facteur de prédiction Hi(u,v) conduit dans le plan spatial à une pseudo-convolution compliquée avec des filtres de longueur importante et dont les coefficients varient en fonction de la position spatiale des points.

En fait dans le modèle avec compensation de mouvement et deux transformées on est obligé de limiter le filtrage à des cas simples, c'est à dire à ne considerer que des modes mixtes simplifiés.

II.4 Comparaison des 2 modèles de codage.

La comparaison des 2 modèles -réalisée dans le cadre d'un codage à 300 Kbit/s- donne l'avantage au codage avec compensation de mouvement intégrant un filtre dans la boucle (filtre 1 2 1 dans les deux directions effectué suivant un critère de minimisation d'erreur).

Néanmoins nous devons noter que la différence de qualité subjective n'est pas très importante au regard de la différence de complexité.

modèle compensation + filtre	modèle "meilleures prédictions"
2 TCD	1 TCD
compensation 225 fois 64 (OPE) ou 25 fois 64 (OPE) (≠)	recherche de classe 6 fois 64 (x) et 8 fois 64 (OPE)
filtrage 64 fois 9 (x) + 64 (OPE)	

TAB 1: Complexite du calcul de prédiction

(OPE) = opération élémentaire= Différence + valeur absolue + accumulation

(x) = multiplication

(≠) avec algorithme simplifié 3 pas (Algorithme JAIN ref 5).

Les résultats obtenus en terme de rapport signal à bruit (S/B) sur trois séquences types de visioconférence, et pour un débit de codage de 300Kbit/s, sont les suivants:

MODELE	SEQ 1	SEQ 2	SEQ 3
avec compensation sans filtrage	38.98	36.97	36.20
avec compensation et filtrage	39.78	37.82	37.32
hybride inter/intra (2 classes)	38.12	36.52	35.38
hybride "meilleures prédictions"(8 classes)	38.82	37.05	36.28

TAB 2 Comparaison des résultats du (S/B) (en DB) (images 360 x 288 , 10 ou 15 Hz)

En conclusion, cette étude montre les avantages, en termes de complexité, des méthodes à "meilleures prédictions", dans lesquelles l'estimation d'un bloc est calculée à partir du bloc de l'image codée précédente situé à la même adresse (dans le domaine transformé).

Elle montre également qu'avec la compensation de mouvement qui oblige à revenir dans le domaine image pour les calculs des vecteurs déplacements, les opérations très simples de filtrage dans le plan transformé (multiplications) deviennent beaucoup plus délicates dans le plan image (pseudo-convolutions).

Enfin, elle suggère que des solutions de compromis (avantages de la compensation de mouvement et du filtrage dans le domaine transformé) pourraient être obtenues en recherchant un algorithme adroit de compensation dans le plan transformé ou plus probablement en utilisant un schéma de codage à trois TCD.

-REFERENCES-

-1- J.Guichard et G.Eude
"Codage intra-image en transformée en cosinus"
CESTA 86

-2- J.Roese, W.Pratt and J.Robinson
"Interframe cosine transform image coding"
IEEE Computer Trans. C23 (1974)

-3- J.Guichard and G.Eude
"Intra and inter frame transform coding for the transmission of moving pictures"
Proc. ICC 86

-4- S.Ericsson
"Motion compensated hybrid coding at 50 kbit/s"
IEEE (ASSP) March 1985

-5- J.R.Jain and A.K.Jain
"Displacement measurement and its application in interframe image coding"

-6- W.Chen
"Scene adaptive coder"
Proc. ICC 81