

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**

**MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE**

**UNIVERSITE CONSTANTINE 1  
FACULTE DES SCIENCES DE LA TECHNOLOGIE  
DEPARTEMENT D'ELECTRONIQUE**



## **Mémoire**

**En Vue de l'Obtention du Diplôme De Magistère  
Option : Traitement du Signal  
Spécialité : Traitement de L'image**

## **Thème**

---

# **COMPRESSION VIDEO DISTRIBUE UTILISANT LES CODES SOURCES ET CODES CANAL**

**Par : BENAISSA Manel**

---

Soutenu publiquement le 26/11/2014 devant le jury composé de :

---

<b>Président</b>	<b>Prof. Abdelfatah Charef</b>	Université Constantine -1
<b>Rapporteur</b>	<b>Dr. Said Benierbah</b>	Université Constantine -1
<b>Examineur</b>	<b>Prof. Fella Hachouf</b>	Université Constantine -1
<b>Examineur</b>	<b>Prof. Atef Farouki</b>	Université Constantine -1

---

**Promotion 2014/2015**

## Résumé

Le Codage vidéo distribué (DVC) est un nouveau paradigme de codage vidéo, il a été introduit et développé ces dernières années car il offre une possibilité d'avoir un encodeur plus simple comparé aux autres codecs classiques dit prédictifs. Ceci est bénéfique pour certaines applications telles que la vidéo surveillance et les réseaux de capteurs sans fil. D'après des recherches effectuées dans le domaine, il a été prouvé que l'utilisation de la sélection de modes combinés à un codage entropique au niveau des systèmes DVC augmente les capacités de compression de ces derniers, tout en réduisant leurs complexités. Dans ce mémoire, nous avons réalisé un système de codage distribué avec sélection de mode au codeur et le codage canal est remplacé par un codage entropique.

Mots-clés - codage vidéo distribués; Représentation différentielle ; classification des blocs; Sélection de mode.

## ملخص

عملية ضغط الفيديو الموزعة هو نموذج جديد في ترميز الفيديو، تم عرضه وتطويره في السنوات الأخيرة لأنه يوفر إمكانية الحصول على انكودر أبسط بالمقارنة مع غيره من الكوديكس التقليدية المذكورة. وهذا مفيداً لبعض التطبيقات مثل المراقبة بالفيديو وشبكات الاستشعار اللاسلكية. وفقاً لأبحاث أجريت في هذا المجال، تم إثبات أن استخدام انتقاء المودس على مستوى الديكودر مشتركاً مع الترميز الانتروبي يزيد من قدرة ضغط هذا الأخير، مع خفض من تعقيده. في هذه الأطروحة، سنقدم نظام ضغط الفيديو الموزعة باستعمال انتقاء المودس على مستوى الانكودر مشتركاً مع الترميز الانتروبي. و قدمنا أيضاً نظام ترميز الفيديو بالانترا وحده مع الترميز الانتروبي و هذا بدون قناة العودة و لا تقدير الحركة.

Mots-clés - codage vidéo distribués; Représentation différentielle ; classification des blocs; Sélection de mode.

## Abstract

Distributed video coding (DVC) is an emerging video coding paradigm due to its flexibility to introduce much lower encoding complexity than conventional predictive codecs, which is beneficial for some applications such as wireless video surveillance, wireless sensor networks. According to research performed in the field, it has been proven that the use of Modes selection combined to entropy coding in DVC systems increases its compression capacity and reduce its complexity. In this thesis, we will first present a DVC system with modes selection in the encoder and channel coding is replaced by entropy coding.

*Keywords—distributed video coding; differential; block classification; Mode decision; SI generation.*



# *Dédicaces*

*Je dédie ce modeste travail en signe de respect et de reconnaissance*

*A mes parents*

*Ma famille*

*Ainsi qu'à Amine*

*Et*

*A tous mes amis.*

*Puisse Dieu les garder.*

*Menel*

## *Remerciements*

Je remercie tout d'abord dieu le tout puissant (الحمد لله) pour son aide et pour le courage qu'il m'a donné afin de surmonter toutes les difficultés durant mes études.

Je remercie très particulièrement mon encadreur, **Dr. Said Benierbeh** pour son aide, ses conseils, et l'intérêt qu'il a su me porter, en me faisant part sa grande expérience pour réaliser ce mémoire.

Je tiens à exprimer ma sincère gratitude envers notre chef de laboratoire le **Prof. Mohammed Khamaja** pour le temps précieux qu'il m'a consacré et les conseils avisés qu'il m'a fourni tout au long de mon travail. Ainsi que l'ensemble de l'équipe du laboratoire qui mon aider dans mon travail.

Je ne saurais oublier de remercier l'ensemble des membres de jury d'avoir accepté de juger et d'évaluer mon travail.

Je remercie également tous les enseignants qui ont contribués à mon instruction durant mes années d'étude et en particulier le **Dr. Benia** à qui je porte un grand respect.

Sans oublier tous ceux qui m'ont aidés de loin ou de près à l'élaboration de ce travail, je leur exprime ma profonde sympathie et leur souhaite beaucoup de biens.

## Liste Des Acronymes

**AVC** Advanced Video Coding

**CIF** Common Intermediate Format for images (288 lines by 352 columns)

**CRC** Cyclic Redundancy Check

**Db** Decibel

**DC** Direct Current

**DCT** Discrete Cosine Transform

**DSC** Distributed Source Coding

**DVC** Distributed Video Coding

**GOP** Group Of Pictures

**PRISM** Puissance-efficace, Robuste, de haute compression Multimédia codage

**MSE** Mean Squared Error

**MPEG** Motion Picture Experts Group

**PSNR** Peak Signal Noise Ratio

**QCIF** Quarter Common Intermediate Format for images (144 lines by 176 columns)

**RD** Rate Distortion

**FMD-DVC** Fast Mode Decision DVC

**IOFMD-VC** Intra Only Fast Mode Decision Video Coding

# Table des matières

*List des Tableaux et Figures*

## Introduction générale

## Chapitre I : Etat de l'art du codage vidéo distribué

I.1 Introduction.....	1
I.2 Codage de Source Distribué.....	1
I.2.1 Notion de base sur la théorie de l'information.....	1
I.2.1.1 Quelques définitions et théorèmes.....	1
I.2.1.2 Codage Entropique.....	2
I.2.1.3 Codage canal.....	3
I.2.2 Définition du Codage Source Distribué .....	3
I.2.2.1 Codage de Source Distribué sans Pertes.....	5
I.2.2.2 Codage Source Distribué avec pertes.....	7
I.3 Codage Vidéo Distribué (DVC).....	8
I.3.1 Codage DVC avec codage canal.....	10
I.3.1 Architecture Stanford.....	10
I.3.2 Architecture PRISM.....	14
I.3.2 Codage DVC avec codage source .....	16
I.3.3 Architecture DLSW.....	18
I.3.4 Architecture LCLD.....	19
I.4 Conclusion .....	20

## Chapitre II : Implémentation de FMD-DVC et IOFMD-VC

II.1 Introduction.....	22
II.2 Système LCLD-DVC .....	22
II.2.1 Principe de fonctionnement .....	22
II.2.2 Remarques sur le système LCLD .....	28
II.3 Système FMD-DVC.....	28
II.3.1 Codeur FMD-DVC.....	29
II.3.2 Décodeur FMD-DVC.....	32

II.4 Système IOFMD-VC.....	34
II.4.1 Codeur IOFMD-VC.....	34
II.4.2 Décodeur IOFMD-VC.....	35
II.5 Conclusion .....	35

## **Chapitre III : Résultats de Simulation**

III.1 Introduction.....	39
II.2 Résultats de simulations.....	39
II.2.1 Critère Rate-Distorsion.....	42
II.2.2 Critère de la Qualité Visuel.....	49
II.2.3 Complexité du système.....	52
II.3 Conclusion.....	53

## **Conclusion générale**

### *Bibliographie*

# Liste Des Figures

## Chapitre I

<b>FIG I.1</b> Exemple de deux cameras sans fils et un seule receveur.....	4
<b>FIG I.2</b> Codeur vidéo Simple avec le CSD.....	4
<b>FIG I.3</b> Schéma codage-décodage de deux sources corrélées $X$ et $Y$ .....	5
<b>FIG I.4</b> Théorème S-W.....	6
<b>FIG I.5</b> Limites du théorème S-W.....	7
<b>FIG I.6</b> Codage W-Z.....	8
<b>FIG I.7</b> Principe du DVC.....	9
<b>FIG I.8</b> Codeurs vidéo avec complexité limitée. Téléphone portable, caméra sans fil, capsule endoscopique.....	9
<b>FIG I.9</b> Schéma bloc du codec DVC avec architecture Stanford.....	11
<b>FIG I.10</b> Estimation du mouvement bidirectionnel.....	13
<b>FIG I.11</b> Architecture du codeur PRISM [20].....	14
<b>FIG I.12</b> Codage par syndrome et codage entropique dans un seul bloc de $8 \times 8$ .....	15
<b>FIG I.13</b> Architecture du décodeur PRISM [20].....	16
<b>FIG I.14</b> Dual-layer Slepian-Wolf encodeur et décodeur propose dans [11].....	19

## Chapitre II

<b>FIG II.1</b> Schéma bloc du système LCLD.....	23
<b>FIG II.2</b> Forma des bits de retour.....	24
<b>FIG II.3</b> Différence entre $x$ et $y$ (suit une distribution de Laplace): $E_{max}$ représente le maximum de différences absolu.....	26
<b>FIG II.4</b> Schéma synoptique du système FMD-DVC.....	29
<b>FIG II.5</b> Représentation binaire de la carte binaire.....	30
<b>FIG II.6</b> Bloc des différences absolues entre les pixels des blocs $B_n$ et $B_{n+1}$ .....	31
<b>FIG II.7</b> Influence du choix de $Moy_{min}$ au lieu de $E_{max}$ .....	32
<b>FIG II.8</b> Procédure de décodage par l'utilisation de la carte binaire.....	33
<b>FIG II.9</b> Schéma synoptique du système IOFMD-VC.....	35

## Chapitre III

<b>FIG III.1</b> R-D des séquences Coastgard et Foreman (30fps) de RFR-1.....	42
<b>FIG III.2</b> R-D de la séquence Hall (30fps) de RFR-1.....	43
<b>FIG III.3</b> R-D des séquences Coastgard et Foreman (30fps) de RFR-2.....	44
<b>FIG III.4</b> R-D de la séquence Hall (30fps) de RFR-2.....	44
<b>FIG III.5</b> R-D des séquences Coastgard et Foreman (30fps) de FMD-DVC.....	45
<b>FIG III.6</b> R-D de la séquence Hall (30fps) de FMD-DVC.....	46
<b>FIG III.7</b> R-D des séquences Coastgard et Foreman (15fps) de FMD-DVC.....	47
<b>FIG III.8</b> R-D de la séquence Hall (30fps) avec IOFMD-VC.....	47
<b>FIG III.9</b> R-D de la séquence Foreman .....	48
<b>FIG III.10</b> Image 256 de la séquence Foreman (30 fps) avec $Q_p = 26$ .....	49
<b>FIG III.11</b> Image 23 de la séquence Hall avec $Q_p = 28$ .....	50
<b>FIG III.12</b> Image 73 de la séquence Coastgard avec $Q_p = 26$ .....	50
<b>FIG III.13</b> Image 71 de la séquence Foremane (15 fps) avec $Q_p = 22$ .....	51

## Liste Des Tableaux

### Chapitre I

**TAB II.1** Différentes représentation de  $X_n$  dans différentes bases, ainsi que les différentes valeurs de  $X_{n-1}$  au tour de  $X_n$  qui peuvent être reconstruite sans ambiguïté.....29

### Chapitre III

**TAB III.1** Paramètres de quantification de la séquence Foreman 30fps.....40

**TAB III.2** Paramètres de quantification de la séquence Coastgards 30fps.....40

**TAB III.3** Paramètres de quantification de la séquence Hall 30fps.....40

**TAB III.4** Paramètres de quantification de la séquence Foreman 15fps.....40

**TAB III.5** Résultats de simulation avec trois seuils différents.....41

**TAB III.6** Temps de Processeur (s) et  $Q_p = 34$ .....53

## ***Introduction Générale***

Les techniques de compression vidéo numérique ont joué un rôle important dans le monde des télécommunications en général et les multimédias en particulier. Cette histoire de la compression vidéo a commencé dans les années 80's où la première norme de compression vidéo est apparue sous le nom de norme H.120. Cette dernière a été suivie par H.261 [1] vers la fin des années 80's. Au début des années 90's, le Groupe d'experts Motion Picture (MPEG) de l'ISO a donné de la norme MPEG-1 [1], suivie et remplacé par la norme MPEG-2. Les normes MPEG-4 et H.264 sont apparues vers la fin des années 90's début des années 2000 avec de meilleures performances à des taux de compressions plus élevés [1]. Enfin, un nouveau standard dit *High Efficiency Video Coding (HEVC)* a été proposé en avril 2013 et sera officiellement adopté en fin 2014 comme étant le nouveau standard de compression vidéo H.265. Ce dernier est une version amélioré du H.264, il peut réaliser un taux de compression égale au double comparé à son prédécesseur et avec une qualité vidéo encore meilleure.

Tous ces standards de compression conventionnelle sont prédictifs [1] parce qu'ils exploitent la dépendance statistique des trames au niveau du codeur pour augmenter le taux de compression. Si ce genre de codage a permis d'atteindre des taux de compressions très élevées, il a aussi des inconvénients dont le principal est la complexité élevée du codeur. Chose qui le rend impraticables dans le cas où les ressources du codeur sont limitées, ce qui est le cas pour beaucoup d'applications dites 'up-Link'. C'est le cas des téléphones portables ou les caméras de surveillances sans fils à faible consommation d'énergie ou encore les réseaux de capteurs vidéos sans fils.

C'est dans ce contexte que le codage vidéo distribué (DVC) a été introduit, il consiste à coder deux sources statistiquement dépendantes de façon distribuée (indépendamment) et de les décoder conjointement. Ainsi, la complexité du codage est transférée du codeur au décodeur. Dans leurs théorème [2], Stephan et Wolf ont démontré qu'il été possible de coder deux sources statistiquement dépendantes  $X$  et  $Y$  avec un débit binaire égale à leurs entropie jointe, si elles sont conjointement décodée. Ce théorème a ensuite été complété par Winner et Ziv [3] pour le codage avec pertes, c'est sur la base de ce théorème que le codage DVC (ou codage WZ) a

été construit dans les années 90's [4].

Ainsi, plusieurs architectures DVC basées sur ces théorèmes sont apparues, telle que le système DVC proposé par l'université Stanford dans le domaine pixel en 2002 [5][6], suivi par une autre architecture basée sur le domaine de transformation, proposé par la Berkeley University dit PRISM (Puissance-efficace, robuste, de haute compression Multimédia codage) [7] [8] en 2002.

L'architecture Stanford a largement été utilisée dans la littérature pour la réalisation des systèmes DVC, dont la plus connue est le projet européen DISCOVER (DIStributedCOding for VideoSERvices) [9]. Le principe de ce codage est de générer la SI au décodeur, cette dernière est considérée comme étant la version bruitée de la trame d'origine obtenue à travers un canal virtuel modélisé par une distribution de la place, elle est ensuite corrigée à l'aide des codes correcteurs (tels que les turbo codes et LDPC codes) en utilisant un canal de retour. Même si le codeur est beaucoup plus simple dans le système de codage DVC, la complexité du codage canal et l'utilisation fréquente du canal de retour le rend relativement lent comparé au codage conventionnel. De plus, la non-connaissance de la SI au niveau du codeur a pour effet de réduire les performances qualitatives de ce type de codage, même si théoriquement il est possible d'atteindre les mêmes performances que le codage conventionnel. Ceci nous laisse penser qu'il y a d'autres méthodes plus efficaces que le codage canal pour mettre en pratique le codage DVC.

Ces dernières années, beaucoup d'études ont été réalisées dans le but de réduire la latence et augmenter les performances du codage DVC, notamment en utilisant un codage source au lieu d'un codage canal combiné à une méthode de représentation différentiel des données [10][11][12] ou en introduisant des sélections de modes au niveau du codeur ou du décodeur [13][14][15], ou encore par l'élimination du canal de retour [16]. Les résultats de ces travaux se sont pour la plus part avérés meilleurs que l'état de l'art du codage DVC (DISCOVER).

Dans ce mémoire, nous avons réalisé un système de compression distribué avec sélection de modes au codeur et sans codage canal que nous avons remplacé par un codage source moins complexe combiné à une méthode de représentation différentiel basée sur l'arithmétique modulaire. Nous avons appelé ce système FMD-DVC (Fast Mode Decision DVC). Afin de mesurer les performances de ce dernier, nous l'avons comparé à un système de codage vidéo très simple avec sélection de modes (Skip et Intra) au codeur que nous avons implémenté, nous l'avons dénommé

IOFMD-DVC (Intra Only Fast Mode Decision DVC).

Le reste de ce document se présente comme suit, le Chapitre I est dédié à l'état du codage DVC et ces différentes architectures. Le Chapitre II est dédié à la présentation et l'explication du contenu du travail que nous avons effectué. Enfin, le Chapitre III est consacré aux résultats de simulations, suivit par la conclusion générale et des perspectives.

# Chapitre I

*Eta de l'Art du DVC*

## I.1 Introduction

Dans ce chapitre, nous allons présenter l'état de l'art du Codage Vidéo Distribué (DVC). Nous commencerons par les notions de base de la théorie de l'information et les théorèmes de codage de source distribué (CSD), qui ont contribué à la naissance du DVC. Et nous concluons par la présentation des architectures DVC les plus connues basées sur le codage canal et le codage source.

## I.2 Codage de Source Distribué

Avant d'entamer notre discussion sur le codage source distribué, un préambule sur la théorie d'information est nécessaire.

### I.2.1 Notion de base sur la théorie de l'information

La théorie de l'information dans le domaine du CVD consiste à déterminer les limites (imposées par les lois physiques) de la compression et la transmission de l'information à partir d'une source.

#### I.2.1.1 Quelques définitions et théorèmes

Si l'on considère qu'une source  $X$  émet un nombre fini  $K$  de valeurs discrètes  $\{x_1, x_2, \dots, x_k\}$  avec une probabilité d'apparition de chaque symbole modélisé par  $P(x_1), P(x_2), \dots, P(x_k)$ . Ainsi,

**Définition.1 (Auto-Information)** l'auto-information  $I[x_i]$  du symbole  $x_i$ , représente la quantité d'information apporté par ce dernier,

$$I[x_i] = -\log P[x_i] \quad (1)$$

**Définition.2 (Entropie)** L'entropie d'une source est égale à la moyenne des auto-informations de son alphabet soit,

$$H(X) = \sum_{i=1}^K P(x_i) I(x_i). \quad (2)$$

**Définition.3 (Entropie Conjointe)** L'entropie conjointe de deux sources  $X$  et  $Y$  ayant respectivement un nombre fini  $K$  de valeurs discrètes  $\{x_1, x_2, \dots, x_k\}$  et  $\{y_1, y_2, \dots, y_k\}$ ,

$H(X, Y)$ , est définie la moyenne de la quantité d'information conjointe  $I(x_i, y_i)$  de chaque message élémentaire  $x_i$  et  $y_i$  des deux sources soit,

$$H(X, Y) = \sum_{i=1}^K P(x_i, y_i) I(x_i, y_i). \quad (3)$$

**Définition.4 (Entropie Conditionnelle)** L'entropie conditionnelle  $H(X | Y = y)$  est l'entropie de la variable  $X$  conditionnée par la variable  $Y$  prenant une certaine valeur  $y$ , alors  $H(X | Y)$  est le résultat d'une moyenne de  $H(X | Y = y)$  sur l'ensemble des valeurs  $y$  que  $Y$  peut prendre.

$$H(X|Y) = \sum_{i=1}^K P(x_i, y_i) \log \frac{\log P(x_i)}{P(x_i, y_i)}. \quad (4)$$

### 1.2.1.2 Codage Entropique

Le codage entropique ou codage source de manière générale consiste à transformer le message d'une source en une séquence d'information «  $I(x_i)$  » de façon à minimiser la taille du message en éliminant les redondances, tout en gardant la possibilité de retrouver le message d'origine à partir de la séquence de substitution «  $I(x_i)$  ».

Le but de cette opération est d'optimiser les ressources nécessaires à la transmission (temps, puissance, bande passante, etc.).

#### ***Théorème codage source***

Pour assurer une reproduction fiable de la source  $x_i$  à la réception (transmission sans perte), cette dernière doit être codée à un débit binaire  $R_{x_i}$  égale ou supérieure à son entropie  $H(X)$ .

$$R_{x_i} \geq H(X) \quad (5)$$

Par exemple le Codage de *Huffman* est un algorithme de compression de données sans pertes. Il utilise un codage à longueur variable pour représenter les symboles de la source. Le codage à longueur variable consiste à coder avec peu de bits les mots les plus fréquents, de façon à utiliser le minimum de bits en moyenne.

### **I.2.1.3 Codage canal**

Le codage canal a pour rôle de protéger l'information émise contre les perturbations du canal de transmission susceptibles de modifier son contenu. Il s'agit donc de rajouter de la redondance de manière à détecter et éventuellement corriger les erreurs lors de la réception si la stratégie adoptée le permet. Les conditions d'un codage correct sont déterminées par le second théorème de Shannon [17].

#### ***Théorème codage canal***

Ce théorème indique qu'il est possible de réaliser une transmission sans erreurs avec un débit inférieur ou égale à la capacité du canal.

$$R \leq C, \quad (6)$$

Où  $C$  est la capacité du canal. Pour un canal à bruit blanc gaussien elle est définie par

$$C = W \log_2 \left( 1 + \frac{P}{N_0 W} \right) \quad (7)$$

Où

1.  $W$  est la largeur de bande disponible,
2.  $\frac{N_0}{2}$  est la variance du bruit blanc,
3.  $P$  est la puissance du signal émis.

Deux types de codes peuvent être utilisés: les codes en bloc et les codes convolutifs. Des exemples de codes en blocs sont les codes (BCH), Reed-Muller (RM), les codes cycliques, et les codes de Hamming, et Reed Solomon (RS), Low Density Parity Chek (LDPC). Les codes convolutifs appelés codes en treillis tels que les Turbo codes.

### **I.2.2 Définition du Codage Source Distribué**

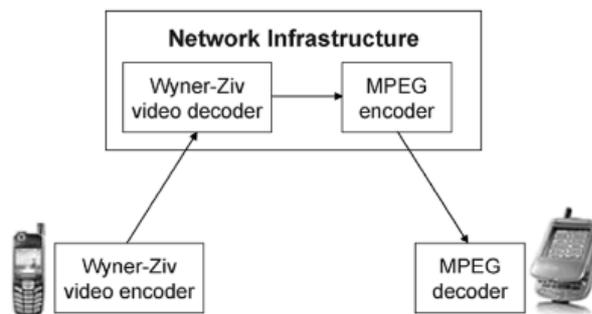
Le principe du Codage Source Distribué (CSD) est de coder deux sources (ou plus) d'informations aléatoires corrélées sans aucune communication entre elles au niveau du codeur. Ces dernières sont par ailleurs décodées de manière conjointe au niveau du décodeur, en utilisant quelques modèles de corrélation entre les sources

en question. La figure I.1 illustre un exemple de deux codeurs (cameras sans fils) indépendant et un décodeur (récepteur) commun.



**FIG I.1** Exemple de deux cameras sans fils et un seule receveur.

Le CSD est donc capable de basculer la complexité de calcul du codeur vers le décodeur. Pour exemple, la figure I.2 montre une communication entre deux téléphones réalisée par un codeur simple (WZ) et un décodeur simple (MPEG).



**FIG I.2** Codeur vidéo Simple avec le CSD.

Ainsi, un train de bits indépendants est envoyé par chaque codeur vers un seul décodeur qui effectue un décodage conjoint de tous les trains de bits reçus, en exploitant la corrélation statistiques entre eux.

Le principe du CSD prend toute son importance lorsqu'il s'agit d'une situation ou le codeur dispose de capacités limité du point de vue énergie, comme est le cas dans la vidéo de surveillance sans fil ou les systèmes de codage vidéo classiques ne sont pas bien adaptés pour une telle application, car la tâche d'estimation de

mouvements consomme beaucoup d'énergie avec un taux de calcul colossale.

Si l'idée du CSD semble très intéressante du point de vue, transfert de la complexité de codage du codeur au décodeur, il serait judicieux de connaître le débit minimum nécessaire pour la transmission de deux sources  $X$  et  $Y$ , lorsque ces dernières sont codées de façon disjointe et conjointement décodé. La réponse à cette question été apporté en 1973 par D. Slepian et J. K. Wolf [2], ces derniers ont donné la limite théorique qui assure une compression sans perte dans le cas d'une compression distribuée de deux sources statistiquement corrélées, indépendantes et identiquement distribuées (i.i.d),  $X$  et  $Y$ . Plus tard, des nouvelles limites théoriques, dans le cas de compression avec perte, ont été présentées par A. D. Wyner et J. Ziv en 1976 [3].

Les deux théorèmes de Slepian-Wolf et WZ sont présentés ci-dessous.

### I.2.2.1 Codage de Source Distribué sans Pertes

Le paradigme de codage de source distribué, est de réaliser une transmission fiable (sans perte) d'une seule séquence d'information discrète et aléatoire  $x_i$  avec un débit  $R_{x_i} < H(X)$ . Si on suppose qu'on a une paire de séquences  $x_i$  et  $y_i$  présentant une certaine corrélation. L'objectif du codage de sources distribuées est de parvenir à une compression optimale de ces deux sources, sachant que celles-ci ne communiquent pas durant le codage tel que montré dans la figure II.3.

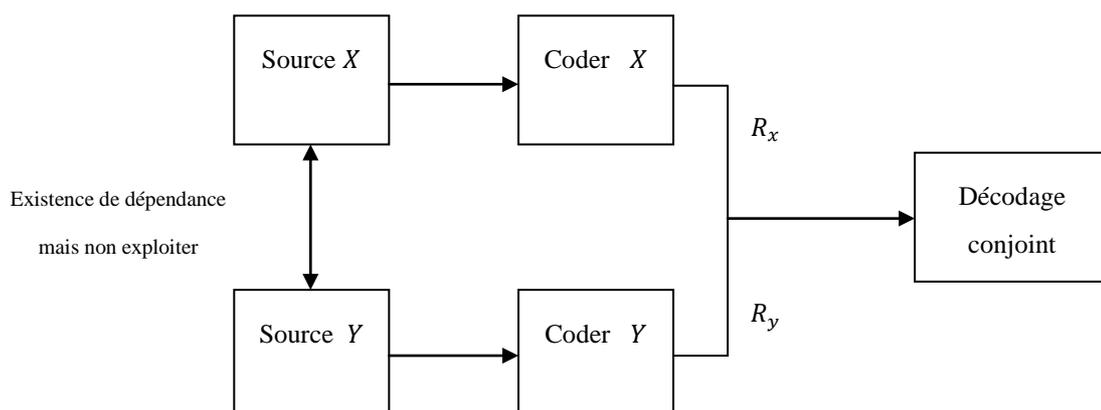
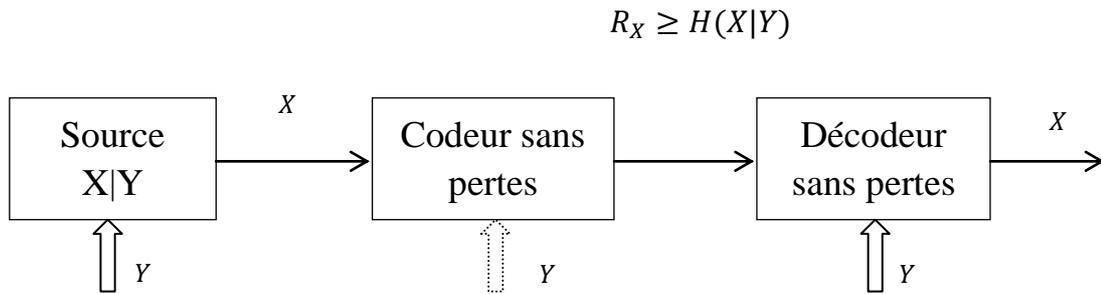


FIG I.3 Schéma codage-décodage de deux sources corrélées  $X$  et  $Y$ .

Le théorème de Slepian-Wolf [2] répond à la question suivante : comment peut-on connaître l'information redondante si les deux sources ne communiquent pas à l'émission ?

La réponse à cette question est expliquée par la figure I.4.

***Théorème de Slepian-Wolf***



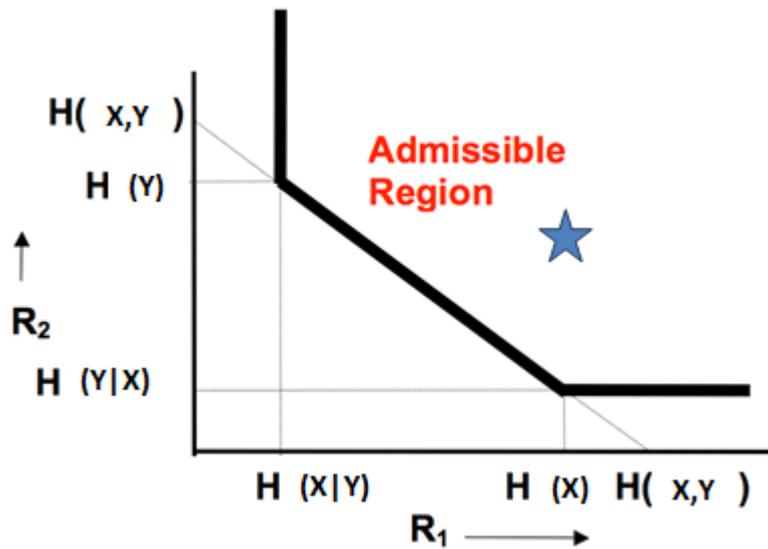
**FIG I.4** Theoreme de S-W.

Selon S-W, L'encodage de  $X$  nécessite un débit  $R_X$  tel que,

$$H(X|Y) \leq R_X \leq H(X) \tag{8}$$

Ceci est valable si le décodeur dispose de l'information adjacente  $Y$ . Ce scénario est valable pour le cas d'une compression sans perte de  $X$  avec une information adjacente  $Y$ , disponible uniquement au décodeur.

Nous noterons que  $P_{XY}(x_i, y_j)$  est la probabilité conjointe d'avoir  $x_i$  en entrée du canal et  $y_j$  en sortie, et  $P_{X|Y}(x_i/y_j)$  la probabilité d'avoir  $x_i$  en entrée du canal, en sachant qu'on a  $y_j$  en sortie.



**FIG I.5** Limites du théorème S-W.

Les résultats sont présentés sous forme de régions d'admissibilité, tracées dans le plan  $[R_X - R_Y]$ , et délimitant la zone du plan (région d'admissibilité) dans laquelle on peut envoyer les deux séquences avec fiabilité.

La région d'admissibilité, est définie par les conditions suivantes [2].

$$\begin{cases} R_{x_i} \geq H(X|Y) \\ R_{y_j} \geq H(Y|X) \\ R_{x_i} + R_{y_j} \geq H(X, Y) \end{cases} \quad (9)$$

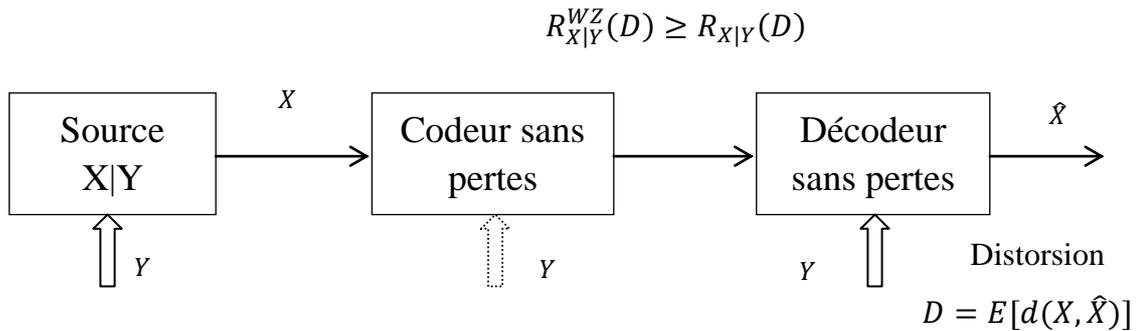
D'après la figure I.5, on constate que l'on peut transmettre la source  $x_i$  à un débit inférieur à son entropie  $R_{x_i} < H(X)$  si la source  $y_j$  est codée avec un débit respectant le critère  $R_{y_j} \geq H(Y)$ . Ainsi, malgré le fait que chaque codeur ne voit que sa propre source ( $X$  ou  $Y$ ), la corrélation entre les deux sources peut être exploitée au décodeur.

### I.2.2.2 Codage Source Distribué avec pertes

La plupart des systèmes de compression tolèrent un certain niveau de distorsion entre l'information originale et l'information reçue. Wyner et Ziv se sont intéressés dans leurs travaux [3], au cas de compression avec pertes, en définissant

la fonction débit distorsion  $R(D)$  tel que présenté dans la figure I.6.

Dans le cas de présence de l'information adjacente  $Y$  au niveau du décodeur, le théorème de Wyner-Ziv (W-Z) stipule que cette information peut être exploitée pour réduire le débit de transmission sans augmenter la distorsion,



**FIG I.6** Théorème de W-Z.

$R_{X|Y}(D)$  désigne le débit d'encodage de la source  $X$  si l'information adjacente est disponible à la fois à l'encodeur et au décodeur et  $R_{X|Y}^{WZ}(D)$  désigne le débit d'encodage de la source  $X$  si l'information adjacente est disponible uniquement au décodeur. Wyner et Ziv [3] ont démontré que,

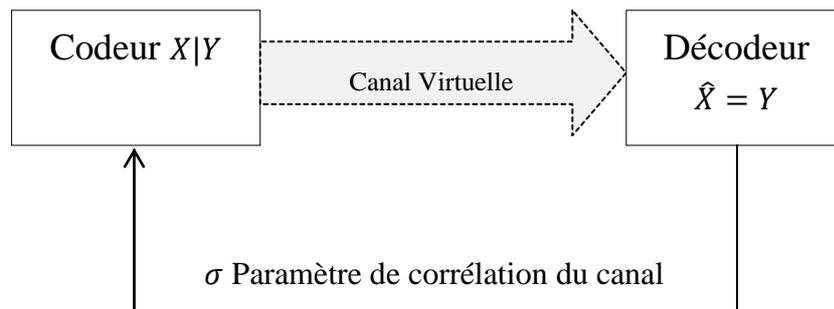
$$R_{X|Y}^{WZ}(D) \geq R_{X|Y}(D) \quad (10)$$

Où l'égalité est atteinte si les sources  $X$  et  $Y$  sont conjointement gaussiennes et si la mesure de la distorsion consiste en l'erreur quadratique moyenne.

### I.3 Codage Vidéo Distribué (DVC)

Le codage vidéo distribué, constitue un des domaines d'exploitation du CSD. La mise en œuvre pratique de ce théorème dans le domaine de la compression vidéo a été réalisée vers les années 2000 [4] par des codes correcteurs d'erreurs. Le principe est de considérer l'information adjacente  $Y$ , comme étant la sortie systématique de l'information original  $X$  à travers un canal virtuel modélisé par une

distribution de Laplace, en général tel que montré dans la figure I.7. Un codage canal est alors utilisé pour corriger la SI.



**FIG I.7** Principe du DVC.

Etant donné que le codage vidéo numérique tolère une certaine distorsion du fait de la quantification, le principe du DVC a été mis en œuvre sur la base du théorème de Wyner-Ziv [3]. Ainsi, le rôle principal du DVC est de transférer la complexité du codeur au décodeur tout en approchant au mieux les performances du codage vidéo conventionnel. De plus en plus d'applications dans le domaine vidéo sont développées sur des terminaux sans grandes ressources énergétiques ou matérielles tel que les capsules endoscopiques, les téléphones portables, les réseaux de capteurs sans fils ou encore les cameras sans fils.



**FIG 1.8** Codeurs vidéo avec complexité limitée. Téléphone portable, caméra sans fil, capsule endoscopique.

Beaucoup de recherches ont été réalisées pour augmenter l'efficacité du DVC avec codage canal en utilisant notamment de nouveaux codes correcteurs d'erreurs, ou encore par le changement de la modélisation de la distribution du canal virtuel, ou encore par l'amélioration de la SI.

Cependant, d'autres recherches se sont penchées sur le fait d'éliminer le codage canal en le remplaçant par un codage source plus simple, ou encore en utilisant une sélection de modes au codeur. Comme ce fut le cas de notre travail.

Par ailleurs, plusieurs algorithmes de codage avec décision de mode Intra (classique) ont été présentés dans la littérature DVC. Ainsi, le codeur a la possibilité de sélectionner et de coder en Intra les blocs critiques. La décision des blocs Intra est prise pour les régions à faible corrélation entre la SI et la trame WZ correspondante. Dans tous les travaux, les blocs de la trame WZ sont classés suivant deux modes: WZ ou Intra et les blocs Intra sont utilisés directement comme information décodée finale, la SI n'est donc pas générée / utilisée pour eux.

Par exemple, dans [13], les corrélations spatiales et temporelles sont utilisées pour émettre une décision sur la nature du bloc. Dans [14], le mécanisme de sélection proposé a pour but de déterminer quel bloc dans la trame WZ est le plus intéressant à envoyer en Intra pour améliorer la qualité de la SI. Néanmoins, la méthode du choix du meilleur bloc comporte beaucoup de paramètres à calculer et augmente par ce fait la complexité du codeur. Dans [15], un codec DVC utilise trois modes de codage à savoir: Intra, Inter et WZ.

### **I.3.1 Codage DVC avec codage canal**

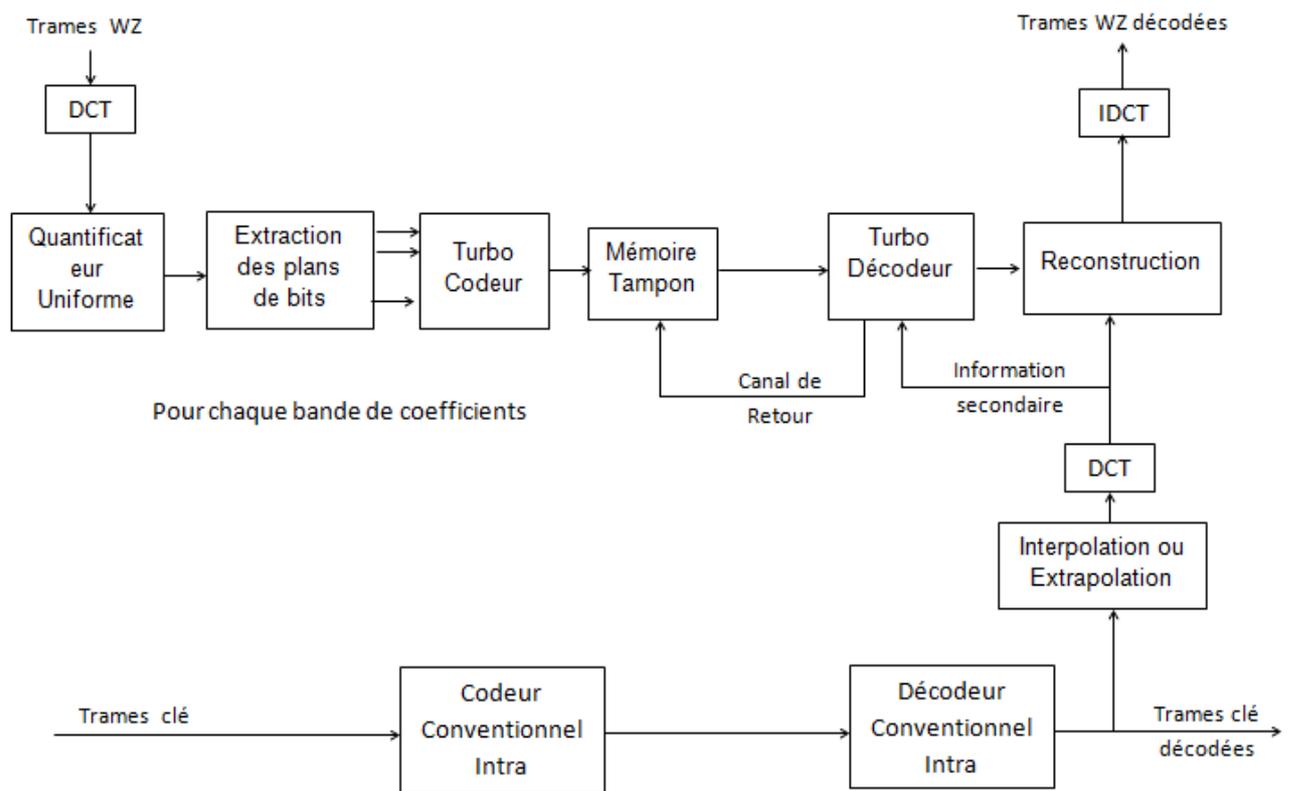
La mise en œuvre pratique du codage vidéo DVC a commencé vers 2002, à la suite des progrès importants réalisés autour du codage canal et ce en particulier après l'apparition de codes correcteurs d'erreurs avec une capacité de correction proche de la limite de Shannon, parmi lesquels on peut citer les turbo codes et les codes LDPC [18]. Les premiers systèmes DVC ont été introduits par les chercheurs de l'université de Stanford et Berkeley.

#### **I.3.1.1 Architecture Stanford**

Le système de codage vidéo basé sur l'architecture Stanford a d'abord été proposé pour le domaine pixel [5][6] et plus tard étendu au domaine de transformation DCT (Discret CosinTransform) [5][6]. La figure I.8 représente le

schéma synoptique du codec DVC Stanford [5][6]. Le système de codage distribué actuel tel qu'introduit par le projet européen DISCOVER [9], est basé sur l'architecture Stanford (c'est d'ailleurs l'architecture utilisé dans la plus part des codeurs DVC actuels).

Les éléments composant ce système sont :



**FIG I.9** Schéma bloc du codec DVC avec architecture Stanford.

### Classification de trame

La séquence vidéo est divisée en trames WZ (WZ Frames), codées par DVC, et des trames clés (Key Frames). Les trames clés sont codées en intra, en utilisant la norme H.264/AVC en mode intra. Les images clés sont périodiquement insérées, déterminant ainsi la taille du groupe d'images (GOP).

### Transformation

L'intérêt d'une transformation est que seuls un petit nombre de coefficients sont non-nuls, et peuvent être utilisés pour reconstruire l'image par transformée

inverse (IDCT) lors de la décompression. Le gain en termes de compression vient de la suppression des coefficients nuls ou proches de zéro par la quantification.

Après avoir divisé les trames WZ en blocs (généralement de 4x4), une DCT est appliquée à ces derniers. Les Coefficients de chaque bloc sont ensuite regroupés, formant des bandes (16 bandes) de coefficients, dont une bande DC et des bandes AC.

### **Quantification**

Chaque bande est uniformément quantifiée et les bits des symboles quantifiés sont regroupés, formant des plans de bits, ces derniers sont ensuite séparément turbo codé (ou codé par LDPC). Les coefficients de la bande DC sont quantifié par un quantificateur scalaire uniforme dont la plage peut aller de  $[0, 2^M)$ , ou  $M$  est le nombre de bits utilisé pour chaque coefficient. Pour ceux des bandes AC, un quantificateur à zone morte (Dead-Zone).

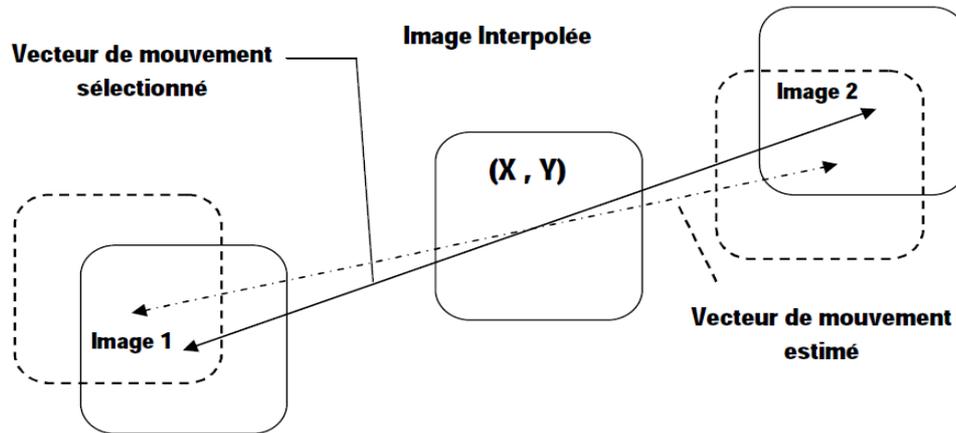
### **Codage Canal**

Le codage de chaque bande DCT commence avec le Plan de bits les plus significatifs (MSB). Le syndrome généré pour chaque plan de bits est ensuite stocké dans la mémoire tampon et envoyé en morceaux à la demande du décodeur, par l'intermédiaire du canal de retour. Les exemples de codage canal les plus utilisés sont Turbo codes et LDPC [18].

### **L'Information Adjacente (Side Information SI)**

L'étape d'estimation de la SI est importante car, plus la qualité de cette dernière est bonne (la corrélation entre les images WZ est élevée) moins de bits de parités sont nécessaires pour le décodage. La SI est générée par interpolation avec Estimation de Mouvement (ME), entre les deux trames voisines de la trame originale (Interpolation) ou en utilisant les trames précédentes (Extrapolation).

Il a été prouvé que l'interpolation est plus appropriée pour générer une meilleure estimation de la SI, cette dernière consiste à estimer l'information adjacente d'une image WZ, qui se trouve dans une position temporelle  $k$ , en se basant sur deux images de référence déjà décodées, la précédente ( $k-1$ ) et la suivante ( $k+1$ ) [19].



**FIG I.10** Estimation du mouvement bidirectionnel.

### Modélisation du Bruit de corrélation

La modélisation du canal virtuel entre la trame WZ et l'information adjacente est réalisée par une distribution de Laplace dont le paramètre  $\sigma_n$  est estimé au décodeur et envoyé au codeur par le canal de retour.

### Décodage

Une fois l'information adjacente obtenue cette dernière sera divisé en blocs auxquels une transformation DCT est appliqué suivie par la quantification des bandes DCT et leurs dispositions en plans de bits. Une fois une partie des syndromes envoyés à partir du codeur pour chaque plan de bits un décodage est opéré, dans le cas où l'opération réussie les bits sont envoyés à l'étape suivante, sinon le décodeur demande plus de syndromes au codeur à travers le canal de retour, jusqu'à ce que le décodeur utilise le critère d'arrêt.

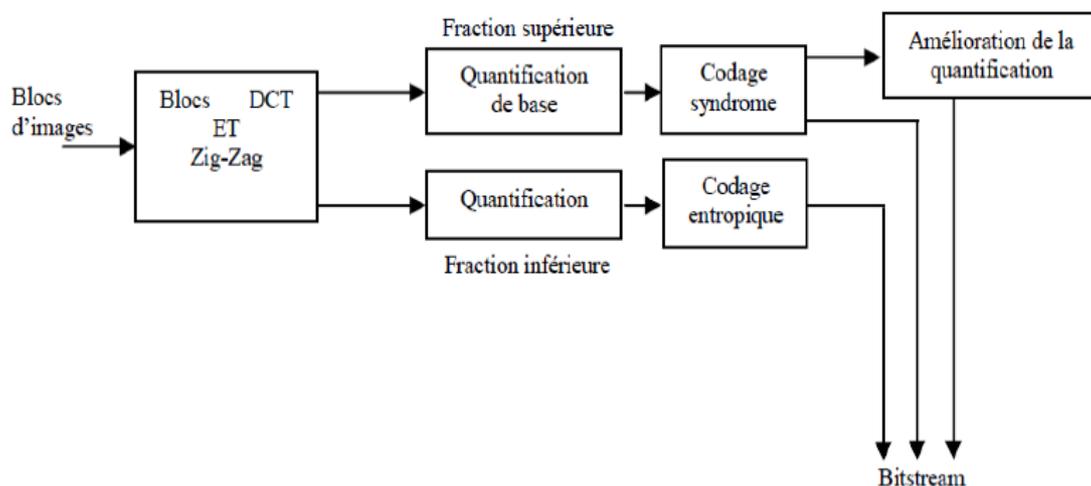
### Reconstruction

Après décodage canal, tous les plans de bits associés à chaque bande DCT sont regroupés pour former un flux de symboles décodés et quantifiés. Une fois tous symboles obtenus, il est possible de reconstruire les coefficients DCT de la partie haute fréquence de l'image a partir de l'information secondaire si les bits WZ n'ont pas été transmis [5][6].

### I.3.1.2 Architecture PRISM

L'approche proposée par l'équipe Berkeley est connu dans la littérature sous le nom PRISM (Power-efficient, Robust, high-compression, Syndrome-based Multimedia coding) [7][8], fonctionne dans le domaine de transformation et est basée sur des blocs. L'architecture de Berkeley consiste en un codage par blocs (4x4 pixels) avec un décodeur avec estimation de mouvement, la classification des blocs est de type Skip, Intra ou WZ, cette sélection est faite sur la base d'une différence entre les trames. Dans le cas des blocs WZ, les coefficients des basses fréquences (en haut à gauche du bloc) sont codés par un codage canal et leurs syndromes sont envoyés avec un code CRC, le reste des coefficients sont codés de manière conventionnelle. Au niveau du décodeur, une estimation (par compensation du mouvement) des blocs de la trame actuelle est réalisée en s'appuyant sur l'information secondaire, pour les blocs WZ, les coefficients de basses fréquences sont corrigés en utilisant les syndromes et le CRC auquel sont joints les coefficients de hautes fréquences préalablement codés d'une manière conventionnelle, il faut noter que ce système n'utilise pas de canal de retour.

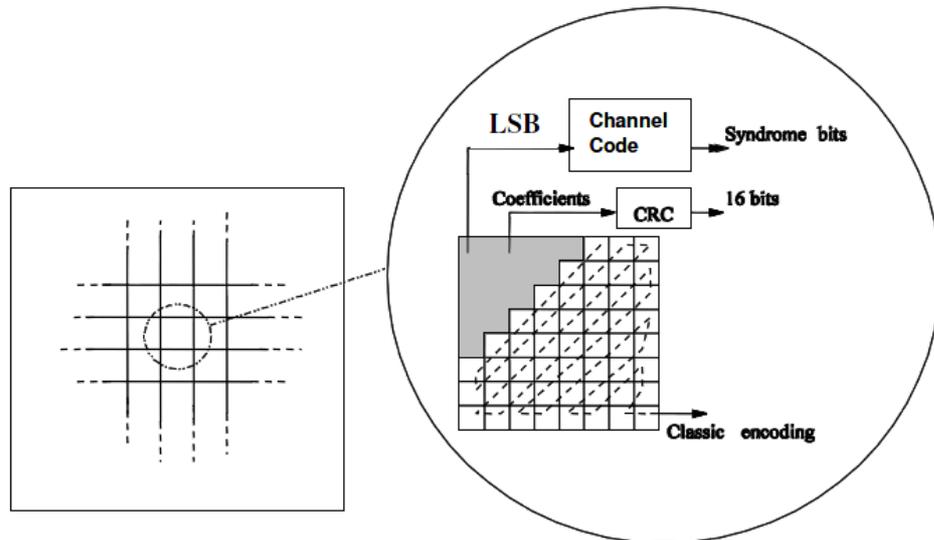
Nous allons brièvement présenter l'architecture de l'encodeur et du décodeur de la solution PRISM pour le codage vidéo distribué.



**FIG I.11** Architecture du codeur PRISM [20].

## Encodeur

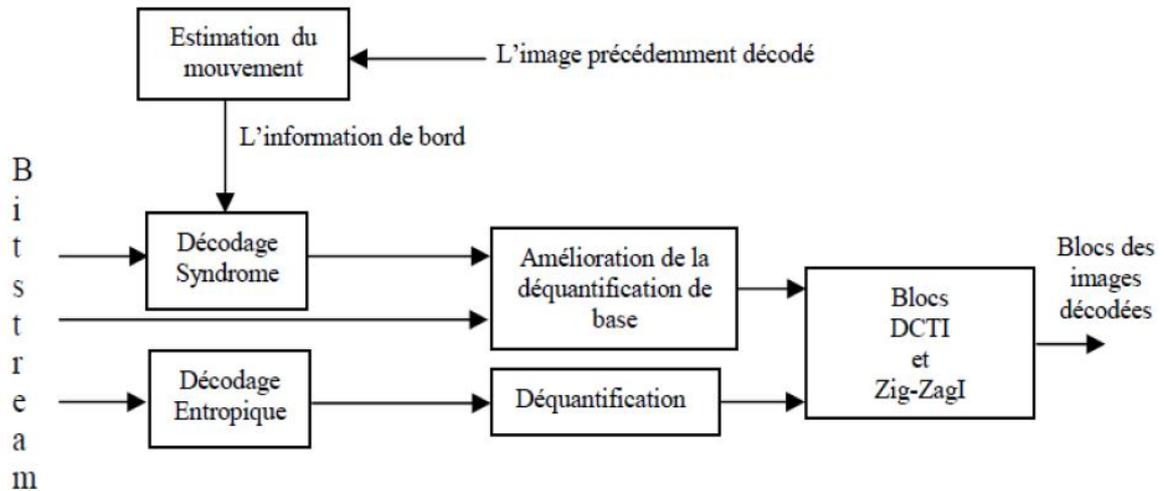
- L'image est divisée en N blocs.
- Les blocs sont classés en 3 catégories : Skip/ intra / WZ avec différents modes de syndromes basés sur les différences entre les trames.
- Pour les blocs WZ, envoyer un syndrome et un CRC (hach).



**FIG I.12** Codage par syndrome et codage entropique dans un seul bloc de  $8 \times 8$

Codage d'un bloc de WZ:

- Calculer la DCT pour chaque bloc.
- Créer des syndromes pour les coefficients de basses fréquences.
- Créer un CRC pour ces coefficients.
- Coder de manière conventionnelle (codage entropique) les coefficients des hautes fréquences.
- La position des coefficients de faibles et de hautes fréquences est fonction de la force de corrélation entre la trame actuelle et précédente.



**FIG I.13** Architecture du décodeur PRISM [20].

### ***Décodeur***

Pour chacun des N blocs WZ dans l'image

- La recherche du mouvement de chaque bloc de l'information adjacente au niveau du décodeur par rapport à la trame (n-1).
- La Correction des coefficients DCT (de basses fréquences) à l'aide des syndromes.
- Vérifie si le CRC est correct.
- Maintient le premier prédicteur qui fournit un CRC correct.
- Joindre le reste des coefficients de hautes fréquences qui ont été codé de manière conventionnelle.

### **I.3.2 Codage DVC avec codage source**

Si l'utilisation des codes correcteurs d'erreurs performants tels que LDPC ou les turbos codes, a permis de réaliser des systèmes DVC avec des codeurs simples, ceci implique aussi que les décodeurs sont assez complexes. La cause étant principalement due à ces mêmes codes qui nécessitent des algorithmes de décodage très complexes, d'autant plus que l'EM est réalisé au décodeur. Sans oublier le problème du canal de retour qui est fréquemment utilisé durant la phase de décodage (en moyenne 5 requêtes) [11] pour demander d'avantages de syndromes

au codeur. D'autant plus que pour réduire ce nombre de requêtes, une estimation préalable du paramètre de corrélation  $\sigma$  du bruit dans le canal virtuel est nécessaire, ce qui induit encore plus de complexité au niveau du décodeur.

C'est pour toutes ces raisons que des chercheurs se sont intéressés à des moyens alternatifs au codage canal, pour réaliser des systèmes DVC moins complexes. Ils se sont penchés sur la question de remplacer le codage canal par un codage source plus simple. Parmi lesquelles nous citerons les travaux réalisés par P. Aparna et S. David [21][22]. Ces derniers ont réalisé un système DVC qui combine à la fois un codage source et un codage canal en plus d'une classification de blocs au codeur. Le principe de ce système est d'utiliser une classification des blocs au codeur (en Intra, Skip ou Inter) en utilisant le paramètre du canal virtuel  $\sigma$  estimé. Ensuite une DCT est appliquée aux blocs Intra, suivit par une Quantification et un codage source représenté par un codage Huffman. Les blocs Skip ne sont pas envoyés et les blocs Inter sont codés en utilisant un codage canal et un codage source. Le concept du codage canal est appliqué en utilisant le paramètre  $\sigma$  afin de déterminer le nombre de syndromes nécessaires à la correction de la SI. Sauf que dans ce cas seuls les syndromes des coefficients DCT les plus significatifs (20%) sont calculés et codés en utilisant un codage arithmétique, les autres coefficients (80%) sont quantifiés et codés par un codage Huffman.

Une autre approche (mais le principe n'est pas très différent) basé sur l'architecture PRISM fut adopté par M. Calhau [23]. Elle consiste aussi à utiliser la classification des blocs (en Intra, Skip et Inter) au niveau du codeur et un codage entropique. Le système de [23] utilise la mesure de la MSE et la variance  $\sigma$  afin de réaliser la classification. Pour les blocs Inter, le nombre des syndromes est calculé en utilisant  $\sigma$  et ces derniers sont envoyés après avoir été codés en utilisant un codage à longueur variable appelé (Last,Run,Depth,Path) EntropyCoding [23] accompagné d'un code CRC de longueur fixe (16 bits). Les blocs Intra sont quant à eux codés en utilisant le H.263+ [23].

Tel qu'en témoignent les travaux précédents, il existe plusieurs façons de mettre en pratique le codage vidéo distribué avec un codage source à la place du codage canal. Plusieurs travaux ont été réalisés pour optimiser le codage source, comme est le cas de [24] qui a proposé une version améliorée du codage arithmétique basé sur l'utilisation d'une machine à état finit.

Pour nos travaux, nous allons nous intéresser à deux systèmes DVC sans codage canal que nous expliquerons en détails car ils nous serviront de référence par rapport à notre travail. Le premier a été réalisé par G. Huchet et D. Wang en 2010. Ces derniers ont remplacé les codes correcteurs d'erreurs par un codage source combiné à une méthode de représentation différentielle de données [12], associé à une méthode de codage guidée par les coefficients DC présenté en [10]. Le second a été proposé en 2013, par G. Qiu, S. Gao et G. Tu [11]. Ces derniers se sont appuyés sur les travaux réalisés par leurs prédécesseurs pour créer un système DVC basé sur des blocs avec sélection de modes au décodeur. Le codage canal est remplacé par le codage source combiné à une méthode de représentation de données différentielle [12].

Sachant que nous avons choisi le système de [11] basé sur l'utilisation de la sélection des modes comme référence par rapport à notre travail, et puisque nous avons choisi d'appliquer cette sélection au codeur. Nous commencerons par la présentation des systèmes proposés dans [10] et nous terminerons avec celui de [11].

### **I.3.2.2 Architecture DLSW (Dual-Layer Slepian-Wolf)**

Dans [10], les auteurs ont proposé un nouveau type de systèmes DVC (figure I.14), ce dernier est fondé sur une méthode de représentation adaptative de la source [12] et un codage arithmétique des coefficients DCT, ceux ci viennent remplacer le codage canal complexe comme les codes LDPC ou turbo. Comparé aux autres systèmes DVC traditionnels utilisant une architecture Stanford, la complexité de décodage est réduite et le nombre de requêtes à travers le canal de retour est réduit à 2 par trame WZ (tandis-que pour les systèmes DVC conventionnels ce nombre est généralement supérieur à 5 requêtes).

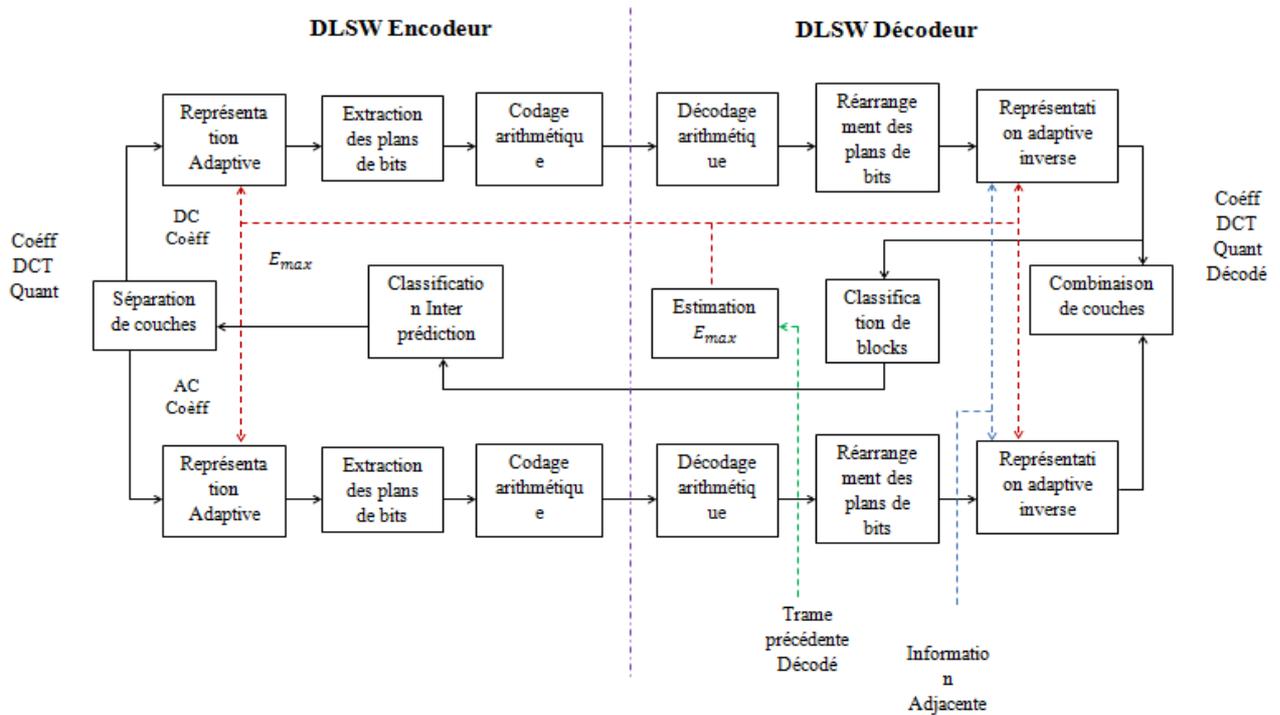


FIG I.14 Dual-layer Slepian-Wolf encodeur et décodeur propose dans [10].

Comme illustré dans la figure ci-dessus, l'idée principale de ce codec est de diviser les coefficients quantifiés DC et AC de tous les blocs de la trame WZ (et l'information adjacente SI) en deux couches distinctes.

La couche des coefficients DC est transmise et décodé en premier, ceci permet au codeur de savoir si tel ou tel bloc nécessite une correction ou pas. Donc, le coefficient DC agit comme un syndrome du bloc en question.

Ainsi, s'il s'avère que le coefficient DC du bloc WZ est le même que celui du bloc de la SI, le reste des coefficients AC sont forts probablement similaires et il n'est par conséquent pas nécessaire de les transmettre. Dans le cas contraire, ces derniers sont transmis après avoir subi une réadaptation différentielle [12] et un codage entropique.

### I.3.2.3 Architecture LCLD (Low-Complexity Low-Delay)

Ce système a été proposé par G. Qiu, S. Gao et G. Tu [11] de l'académie des sciences département d'électronique en s'appuyant sur des études effectuées préalablement en [10].

L'architecture du système proposé par cette équipe est différente de celle de Stanford basée sur le codage canal. En effet ce système utilise un codage source au lieu du codage canal, en plus d'une nouvelle méthode de codage basée sur une réadaptation de l'information source en utilisant le principe de l'arithmétique modulaire [11].

Nous prendrons ce système comme référence pour le reste de nos travaux, c'est la raison pour laquelle nous allons expliquer son principe de fonctionnement de manière plus détaillée dans le chapitre suivant.

## **I.4 Conclusion**

Nous avons introduit le codage vidéo de façon générale et présenté de manière plus précise le codage vidéo distribue à travers ce chapitre. Si les techniques de codage traditionnelles telles que MPEG.X et H.26X sont actuellement les plus performantes du point de vue compression,elles ne sont pas bien adaptées aux nouveaux besoins technologiques en raison de la complexité de calcul au niveau du codeur. C'est pour remédier à cela que le DVC est apparu, car il permet de transférer la complexité du codeur vers le décodeur. Beaucoup de modèles DVC ont été proposé, néanmoins, les plus populaires d'entre eux sont : Stanford et PRISM. Le modèle de Stanford a été quant à lui choisit pour servir de base à la réalisation de la plu part des systèmes DVC actuels notamment le projet européen DISCOVER, car il est très efficace et fournit une meilleure génération de l'information adjacente. Néanmoins, la complexité du codage canal utilisée par cette architecture induit non seulement un décodeur complexe mais aussi un temps de traitement des données plus lent, sans oublier l'utilisation fréquente du canal de retour au cours du décodage. C'est pour cette raison que d'autres architectures DVC sans codage canal ont été proposé et c'est sur la base de ces recherches que nous avons réalisé notre système DVC sans codage canal. Dans le chapitre qui suit nous allons présenter le travail que nous avons réalisé dans le domaine DVC.

# Chapitre II

*Présentation de notre travail*

## II.1 Introduction

En [10][11], les auteurs ont proposé un schéma DVC, sans codage canal. Comparé aux schémas traditionnels basés sur l'architecture Stanford, le décodeur est beaucoup moins complexe et le nombre de requêtes vers le canal de retour est restreint à une par trame WZ (à noter que les schémas DVC utilisant LDPC ou Turbo-Codes nécessite en moyenne 5 requêtes).notre travail est l'implémentation de ce système pour voir ses caractéristiques et modifications possibles.

Dans ce chapitre nous commencerons par réaliser le système proposé dans [11] dit LCLD-DVC (Low-Complexity Low-Delay-DVC). Auquel, nous introduirons quelques modifications qui consistent à :

- Remplacer le paramètre  $E_{max}$  par  $Moy_{min}$  (nous expliquerons ces paramètres ultérieurement).
- Introduire une classification de blocs au codeur.

Nous appellerons ce système FMD-DVC (Fast Modes Decision DVC). Nous présenterons aussi un système dit IOFMD-DV (Intra Only Fast Modes Decision Video Coding) qui nous servira de comparateur par rapport à FMD-DVC.

## II.2 Système LCLD-DVC

Ce système a été proposé par G.Qiu, S.Gao et G.Tu [11] en s'appuyant sur des études effectuées préalablement en [10]. L'architecture du système proposé par cette équipe est différente de celle de Stanford ; basée sur le codage canal. En effet ce système utilise un codage source au lieu du codage canal, en plus d'une nouvelle méthode de codage basée sur une réadaptation de l'information source en utilisant le principe de l'arithmétique modulaire [12]. Nous prendrons ce système comme référence pour le reste de nos travaux, c'est la raison pour laquelle nous allons expliquer son principe de fonctionnement de manière plus détaillée.

### II.2.1 Principe de fonctionnement

L'architecture de ce système est assez simple et ne nécessite pas beaucoup de ressources, la figure II.1 représente le schéma bloc de ce système :

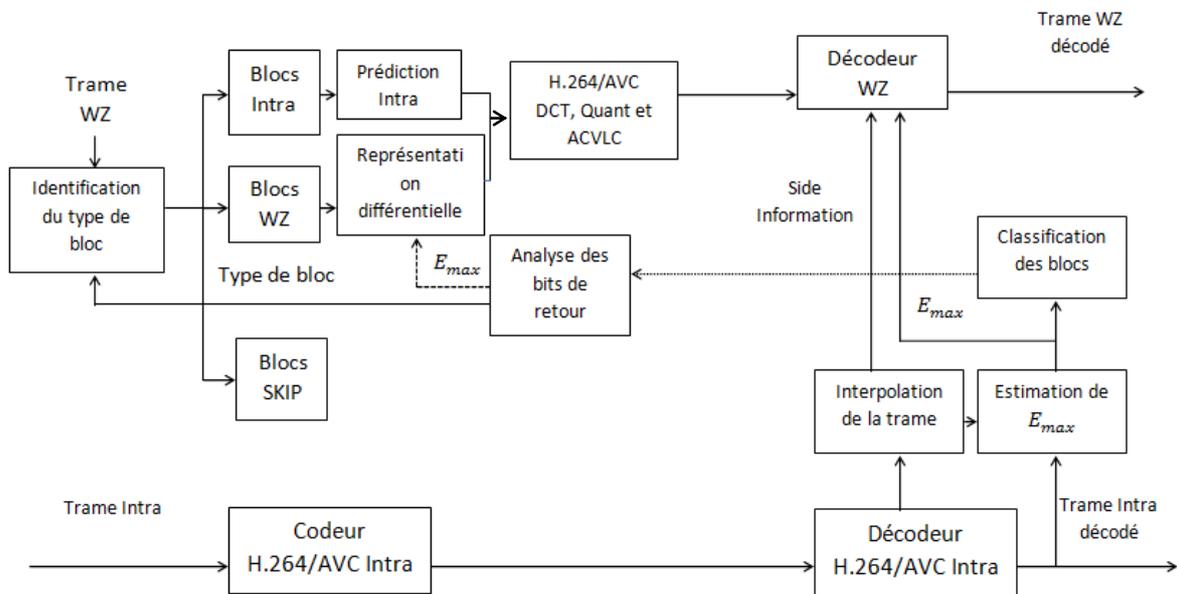


FIG II.1 Schéma bloc du système LCLD

L'architecture du système LCLD [11] est basée sur les blocs. Ainsi, une fois la séquence vidéo divisée en trames Intra et WZ. Les trames Intra sont codées par un Encodeur H.264 et les trames WZ sont divisées en blocs de 4x4 et classifiées comme étant Intra, WZ ou Skip, en utilisant deux seuils Skip et Intra.

Nous avons retenu que l'essentiel de ce système est basé sur,

- ✓ La classification des blocs de la trame WZ.
- ✓ La représentation différentielle des données des blocs WZ en utilisant l'arithmétique modulaire.

L'opération de classification tout comme la représentation différentielle des blocs WZ nécessite la connaissance préalable d'un paramètre appelé : Maximum de différences ( $E_{max}$ ), ce paramètre est calculé au niveau du décodeur pour chaque bloc 4x4. Supposant que  $b_{n-1}$  est un bloc dans la trame Intra précédente  $X_{n-1}$  et  $b_{n+1}$  le bloc correspondant dans la trame Intra suivante  $X_{n+1}$  au niveau du décodeur. Supposant aussi que  $b_n$  est le bloc de la SI  $Y_n$  créé par interpolation des trames Intra précédemment évoqué. Sachant que  $P_i^n$  ( $i = 1 \dots 16$ ) est le  $i^{\text{ème}}$  pixel dans le bloc  $b_n$  de la SI et  $P_i^{n-1}$  et  $P_i^{n+1}$  ceux de  $b_{n-1}$  et  $b_{n+1}$ , le maximum de différences  $E_{max}$  est calculé comme suit:

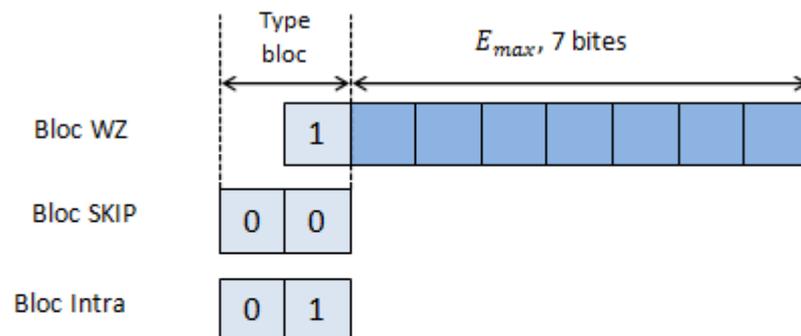
$$\begin{cases} E_1 = \max(|P_i^n - P_i^{n-1}|), i = 1 \dots 16 \\ E_2 = \max(|P_i^n - P_i^{n+1}|), i = 1 \dots 16 \\ E_{max} = \max(E_1, E_2) \end{cases} \quad (11)$$

Ainsi,  $E_{max}$  représente le maximum de différences entre les pixels du bloc dans la SI et les trames Intra précédentes et suivantes. Une fois ce paramètre déterminé, l'opération de classification est réalisée en utilisant deux seuils prédéterminés dit, seuil Intra et seuil Skip. Tel que,

$$\begin{cases} E_{max} \leq E_{TH_{SKIP}}, & \text{le bloc n'est pas envoyé} \\ E_{TH_{SKIP}} < E_{max} \leq E_{TH_{WZ}}, & \text{le bloc est classé WZ} \\ E_{max} > E_{TH_{WZ}}, & \text{le bloc est classé Intra} \end{cases} \quad (12)$$

Il est clairement défini que lorsque le  $E_{max}$  est inférieur ou égal à un seuil Skip, ce dernier est laissé telquel. Lorsque ce dernier est entre le seuil skip et Intra, il est jugé comme étant WZ. Enfin, il est jugé Intra s'il est supérieur au seuil Intra [11].

Le maximum des différences  $E_{max}$  est d'abord calculé au niveau du récepteur et utilisé pour faire une classification des blocs (SKIP, WZ ou Intra) ces informations ainsi que la valeur de  $E_{max}$  sont ensuite envoyées au codeur par le canal de retour en utilisant le format suivant :



**FIG II.2** Forma des bits de retour

Le type de chaque bloc (4x4) est représenté par un code variable qui contient 3 valeurs, dans le cas où le 1<sup>er</sup> bit est 0 alors le bloc est de type SKIP ou Intra et la valeur de  $E_{max}$  n'est pas transmise. Si le bit suivant est 1 alors le bloc est Intra, par contre si c'est 0 alors il est classé SKIP. Autrement, si le premier bit est 1 alors le bloc est WZ et la valeur de  $E_{max}$  est codée sur 7 bits [11].

Une fois la classification terminée au niveau du codeur, une représentation différentielle des blocs WZ est réalisée, une prédiction Intra est opérée pour les blocs Intra et ces données sont codées par H.264/AVC (DCT, Quantification et CAVLC).

### II.2.1.1 Représentation différentielle des blocs WZ se fait comme suit

Comme dans les schémas de codage vidéos traditionnelles où une grande compression est obtenue suite à la réduction de l'énergie de la trame transmise (trame résiduelle), qui est la différence entre la trame originale et celle obtenue par interpolation (SI). Le but de cette opération est de diminuer l'énergie de la trame WZ (dans ce cas le bloc WZ), en utilisant l'arithmétique modulaire [12].

En s'appuyant sur les résultats pratiques obtenus, il a été constaté de manière générale qu'en comparant les coefficients DCT de l'image originale et ceux de l'information adjacente. Les différences entre les bits les moins significatifs sont plus importants que ceux des bits les plus significatifs, il est ainsi possible de réaliser un taux de compression important au niveau des bits les plus significatifs car ils contiennent moins d'erreurs.

Soit  $X$  l'image source et  $Y$  l'information adjacente. Il est possible d'avoir des valeurs très proches de  $X$  et  $Y$  dans une base décimale tel-que  $X = 63$  et  $Y = 64$ , mais leurs représentations binaire est très différente,

$$X = (63)_{10} = (00111111)_2$$

$$Y = (64)_{10} = (01000000)_2$$

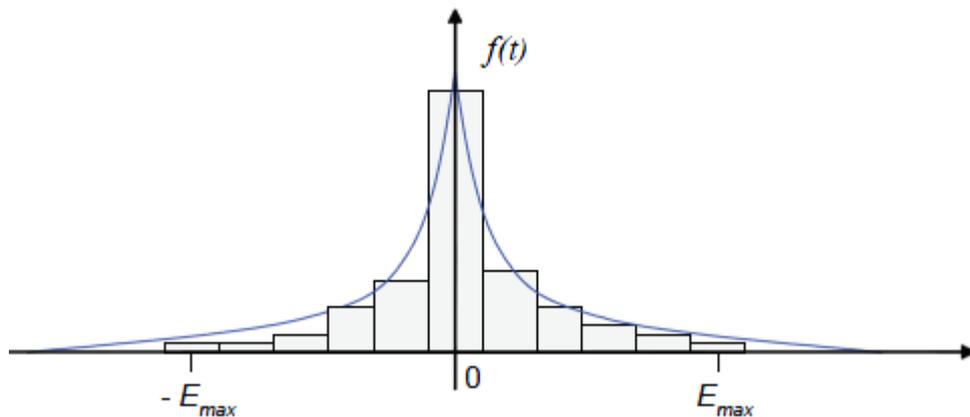
La réadaptation de ces coefficients en codage Gray est venue résoudre ce problème, car le fait de changer un seule bit d'une valeur a la valeur suivante diminue la différence entre les bits, ce qui a pour effet de de réduire le nombre de bits à corriger et donc augmente la compression.

$$X = (63)_{10} = (00100000)_2$$

$$Y = (64)_{10} = (01100000)_2$$

Dans la même optique de réadaptation des coefficients pour augmenter la compression, l'idée de diminuer le nombre de digits par coefficient est apparue. Car en s'appuyant sur le fait que les bits les plus significatifs sont très semblables, il est possible de les remplacer par ceux de l'information adjacente et de n'envoyer par conséquent que les bits les moins significatifs [12].

Cette réadaptation est réalisée sur la base d'un calcul de la différence maximale ( $E_{max} = |X - Y|$ ), entre  $X$  l'information source et  $Y$  l'information latérale. Ce dernier est calculé au niveau du décodeur et envoyé au codeur, à noter que la différence entre  $X$  et  $Y$  suit une distribution de Laplace comme le présente la Figure II.3.



**FIG II.3** Différence entre  $X$  et  $Y$  (suit une distribution de Laplace):  $E_{max}$  représente le maximum de différences absolu.

Ainsi, si l'on considère  $E_{max} = \max|X - Y|$  alors  $X_{tr}$  (qui est une représentation numérique de  $X$  en base  $B$ ) [12].

$$X_{ntr} = X_n \bmod B \quad (13)$$

Et

$$B = 2(E_{max} + 1) \quad (14)$$

Cette représentation est utilisé dans le but de diminuer l'énergie du signal  $X_n$ , et donc diminuer le nombre de bits nécessaires pour le coder et augmenter la compression, donc pour une petite valeur de  $B$  et une large valeur de  $X_n$ , la valeur à coder  $X_{ntr}$  est relativement faible. Cette dernière est facilement reconstituée au niveau du décodeur [12].

Tel-que

$$X_n = Y_n + E_{max} - (Y_{ntr} - X_{ntr} + E_{max}) \bmod B \quad (15)$$

Et

$$Y_{ntr} = Y_n \bmod B \quad (16)$$

Afin d'illustrer ce concept, considérons l'exemple suivant :

Si  $X = 63$  et  $E_{max} = 3$ , alors  $B = 8$ . Le tableau II.1 représente les différentes valeurs possibles de  $X_{n-1}$  au tour de  $X_n$  qui peuvent être reconstruites sans ambiguïté connaissant  $E_{max}$ .

	$X - E_{max}$	$X - 2$	$X - 1$	$X$	$X + 1$	$X + 2$	$X + E_{max}$
$(X)_{10}$	60	61	62	63	64	65	66
$(X)_8$	74	75	76	77	100	101	102
$X_{tr}$	4	5	6	7	0	1	2
Représentation Binaire	100	101	110	111	000	001	010
Représentation Gray	110	111	101	100	000	001	011

**TAB II.1** Différentes représentation de  $X_n$  dans différentes bases, ainsi que les différentes valeurs de  $X_{n-1}$  au tour de  $X_n$  qui peuvent être reconstruite sans ambiguïté.

Dans ce cas si  $60 \leq X_{n-1} \leq 66$ , alors  $X_n$  peut être correctement retranscrit en utilisant les équations (15,16) précédentes [12].

Le nombre de bits nécessaire pour la représentation binaire de  $X_{ntr}$  est  $N = \lceil \log_2(B) \rceil$  Sachant que  $\log_2(B)$  n'est pas un nombre entier, il a été prouvé dans

[12] qu'il été possible de représenter  $X_{n_{tr}}$  en utilisant  $N - 1$  bits, dans le cas où la valeur de ce dernier est inférieur ou égale à un certain seuil  $S = 2^N - B - 1$ . Autrement, dans le cas où  $X_{n_{tr}}$  est supérieur, elle doit être représentée en  $N$  bits [12].

Une fois cette opération terminée et comme est le cas pour les blocs Intra, une DCT et une Quantification sera appliqué aux blocs WZ suivit par un codage CAVLC [25].

### II.2.2 Remarques sur le système LCLD

Le système LCLD-DVC présente beaucoup d'avantages par rapport à celui de Stanford, car en plus d'un décodeur plus simple (sans codage canal) et une utilisation réduite du canal de retour (une seule requête par trame WZ), il présente un meilleur gain en termes de compression vidéo ainsi qu'un temps de traitement de données très réduit. Etant donné que  $E_{max}$  est estimé au niveau du décodeur entre la SI  $Y_n$  (au lieu de la trame originale  $X_n$ ) et les trames clés précédemment décodées, des erreurs d'évaluation peuvent apparaître. Ainsi la classification faite au niveau du décodeur n'est pas forcément la même au niveau du codeur. Il est donc possible d'avoir un bloc classé WZ au décodeur alors qu'au niveau du codeur ce dernier est SKIP. De plus, dans certains cas les valeurs de la SI sont très différentes de celles dans la trame WZ original. Ceci induit une large valeur de  $E_{max}$  et par conséquent, une grande valeur de  $B$ , ce qui veut dire que  $X_{n_{tr}}$  est presque égale à  $X$  et dans ce cas il n'y a pas de compression, tel que montrée dans cet exemple. Si l'on considère que  $X = 54$  et  $Y = 258$ , alors  $E_{max} = 104$  et  $B = 210$ . Ainsi,  $X_{tr} = 54$ , et donc  $X_{tr} = X$ , ainsi il n'y a pas de compression.

### II.3 Système FMD-DVC

Tout comme le schéma proposé par [11], notre système est à base de blocs et le codage canal est remplacé par un codage source [25] combiné à une méthode de représentation différentielle de la source [12]. Une méthode de classification de blocs est utilisée au niveau du codeur et du décodeur. L'architecture de notre schéma est présentée dans la Fig.II.3.



$$\begin{cases} Moy_{n-1} = \frac{(\sum_i (|P_i^n - P_i^{n-1}|))}{16}, i = 1 \dots 16 \\ Moy_{n+1} = \frac{(\sum_i (|P_i^n - P_i^{n+1}|))}{16}, i = 1 \dots 16 \\ Moy_{min} = \text{Min}(Moy_{n-1}, Moy_{n+1}), \end{cases} \quad (17)$$

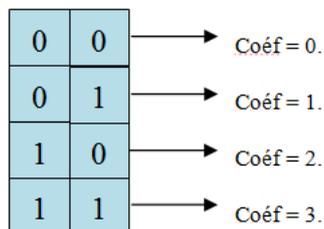
Où,  $Moy$  est la moyenne des différences absolues entre les pixels  $P_i^n$  ( $i = 1 \dots 16$ ) du bloc  $B_n$  ceux des blocs  $B_{n-1}$  et  $B_{n+1}$  ( $P_i^{n-1}$  et  $P_i^{n+1}$ ). Après le calcul de  $Moy_{n-1}$  et  $Moy_{n+1}$ , nous prendrons la plus petite valeur entre elles ( $Moy_{min}$ ). C'est cette dernière qui sera utilisée pour classifier les différents blocs (4x4) en utilisant deux seuils  $E_{TH\_skip}$  et  $E_{TH\_Intra}$ . Tel que,

$$\begin{cases} Moy_{min} \leq E_{TH\_skip}, & \text{le bloc est SKIP} \\ E_{TH\_skip} < Moy_{min} \leq E_{TH\_Intra}, & \text{le bloc est WZ} \\ Moy_{min} > E_{TH\_Intra}, & \text{le bloc est Intra} \end{cases} \quad (18)$$

A noter que la classification ne nécessite que de simples opérations arithmétiques, ce qui préserve la simplicité du codeur. Une fois la classification terminée, une carte binaire avec différents coefficients représentant la nature de chaque bloc est construite tel que :

- Coefficient=0 : Le bloc est Skip par rapport à l'image précédente.
- Coefficient=1 : Le bloc est Skip par rapport à l'image suivante.
- Coefficient=2 : Le bloc est Intra.
- Coefficient=3 : Le bloc est WZ.

La carte binaire est codée en utilisant un codage à longueur fixe de sorte que la trame envoyée est décrite par la figure II.4,



**FIG II.5** Représentation binaire de la carte binaire.

## Remarque Importante sur le choix de $Moy_{min}$

### A. D'un point de vue classement

Nous avons choisi de remplacer le paramètre  $E_{max}$  par  $Moy_{min}$  car dans la plupart des cas, un bloc est jugé Intra à cause de trois ou quatre pixels (en moyenne), ou parfois moins comme le montre l'exemple de la figure II.5,

$$\begin{bmatrix} 0 & 0 & 3 & 4 \\ 1 & 1 & 1 & 1 \\ 2 & 32 & 19 & 2 \\ 0 & 59 & 53 & 3 \end{bmatrix}$$

FIG II.6 Bloc des différences absolues entre les pixels des blocs  $B_n$  et  $B_{n+1}$

Si l'on considère que le seuil Intra est fixé à 50, alors il n'y a que deux pixels dans le bloc dont la différence est supérieure au seuil. D'autre part, si l'on utilise le paramètre  $Moy_{min}$ , ce dernier est dans cet exemple égale à 11. Ce bloc sera donc classé Skip au lieu de Intra, dans le cas où nous avons fixé le seuil Skip à 12.

Ainsi, le nombre des blocs Intra (couteux) sera nettement moins important avec l'utilisation de  $Moy_{min}$  à la place de  $E_{max}$ . Ceci explique notre choix de remplacer le paramètre de classification.

### B. D'un point de vue représentation différentielle

La question de savoir si le choix de  $Moy_{min}$  à la place de  $E_{max}$  a une influence sur la représentation différentielle est maintenant posé. La réponse à cette question est oui, car le fait de réduire l'intervalle de différence entre  $X$  et  $Y$  de :  $[Y - E_{max}, Y + E_{max}]$  à  $[Y - Moy_{min}, Y + Moy_{min}]$ , sachant que  $Moy_{min} \leq E_{max}$ , introduit forcément des erreurs mais que l'on estime «tolérables», la figure II.7 montre un exemple du résultat de l'utilisation de  $Moy_{min}$ . Sachant que,  $[E_{Th_{Skip}}, E_{Th_{Intra}}] = [5, 10]$ .

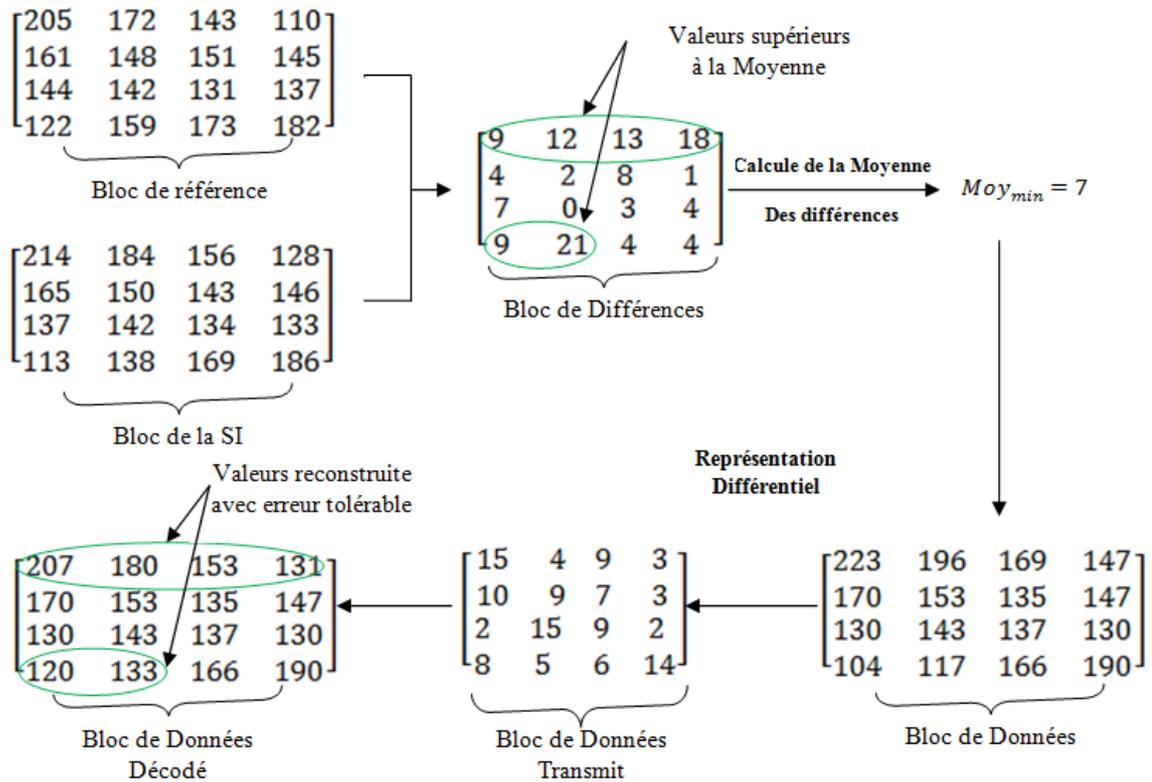


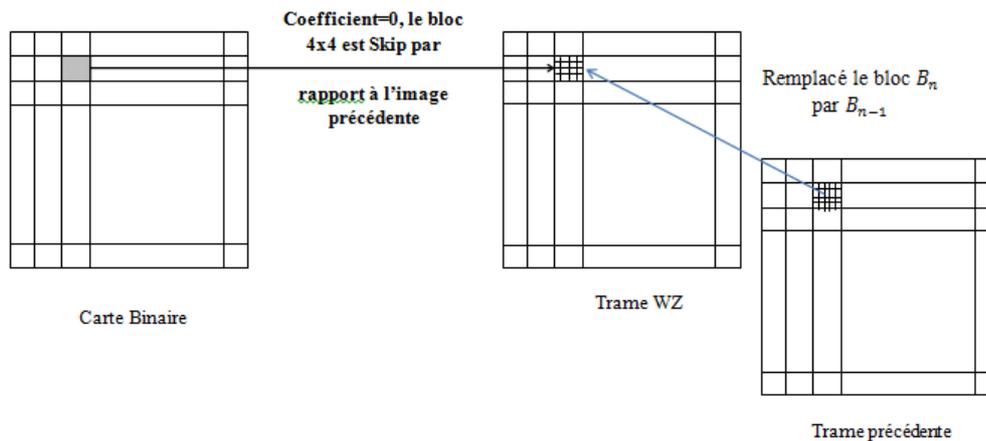
FIG II.7 Influence du choix de  $Moy_{min}$  au lieu de  $E_{max}$ .

### II.3.2 Décodeur FMD-DVC

Une fois la carte binaire reçue, l'opération de décodage est réalisée à travers les étapes suivantes :

1. En premier lieu, l'information adjacente  $Y$  est créée en utilisant l'interpolation entre les trames clés Intra, précédemment décodées.
2. Une opération de classification de blocs est réalisée au niveau du décodeur de la même façon que celle effectuée au codeur, mais cette fois la trame WZ est remplacée par l'information adjacente et les trames précédentes et suivantes sont remplacées par les trames clés.
3. Une fois l'opération de classification terminée, l'opération de vérification des blocs est lancée. Cette dernière consiste à utiliser la carte binaire pour vérifier si un bloc donné a été correctement estimé. Tel que, si un bloc  $B_n$  a été classé de la même façon au codeur et au décodeur, ce dernier a été correctement estimé. Dans

le cas contraire une correction est nécessaire.



**FIG II.8** Procédure de décodage par l'utilisation de la carte binaire.

### II.3.2.1 Correction des blocs mal estimés

La correction des blocs varie selon leurs natures, il existe 3 cas possibles,

- Si un bloc  $B_n$  est classé Intra au codeur et pas au décodeur, un coefficient sera envoyé du décodeur au codeur pour lui indiquer de retransmettre le bloc  $B_n$  codé en Intra.
- Si un bloc  $B_n$  est classé WZ au codeur et pas au décodeur un coefficient indiquant sa nature ainsi que la valeur  $Moy_{min}$  de ce dernier est transmis du décodeur au codeur, afin que celui-ci soit corrigé en utilisant la méthode de représentation différentielle [12] de la source (tel que nous nous l'avons expliqué précédemment).
- Enfin, si un bloc  $B_n$  est codé Skip au codeur et pas au décodeur, ce dernier sera remplacé par un des blocs  $B_{n-1}$  ou  $B_{n+1}$  le plus proche en utilisant la carte binaire. De sorte que,
  - Si le coefficient dans la carte binaire est 0, le bloc  $B_n$  sera remplacé par le bloc  $B_{n-1}$ .

- Si le coefficient dans la carte binaire est 1, le bloc  $B_n$  sera remplacé par le bloc  $B_{n+1}$ .

Les blocs classés Intra au décodeur sont retransmis par le codeur, où une DCT, une Quantification et un codage entropique leurs sont appliquées.

Si la méthode de représentation différentielle [12], utilisée dans le système de [11] repose sur le paramètre  $E_{max}$ . Dans notre cas ce dernier est remplacé par  $Moy_{min}$ . Une fois cette opération terminée et comme c'est le cas pour les blocs Intra, une DCT et une Quantification sera appliquée aux blocs WZ suivit par un codage source, qui dans le cadre de notre travail est représenté par un CAVLC (Context Adaptive Variable Length Coding) [25]. C'est ainsi que ce termine le décodage de la trame WZ.

## II.4 Système IOFMD-VC

Bien que ce système ne s'inscrit par vraiment dans l'optique du codage vidéo distribué, car il ne contient pas de mode WZ. Nous avons néanmoins décidé de le présenter car il bénéficie de deux grands avantages, qui sont : une extrême SIMPLICITE et une grande RAPIDITE de traitement de données.

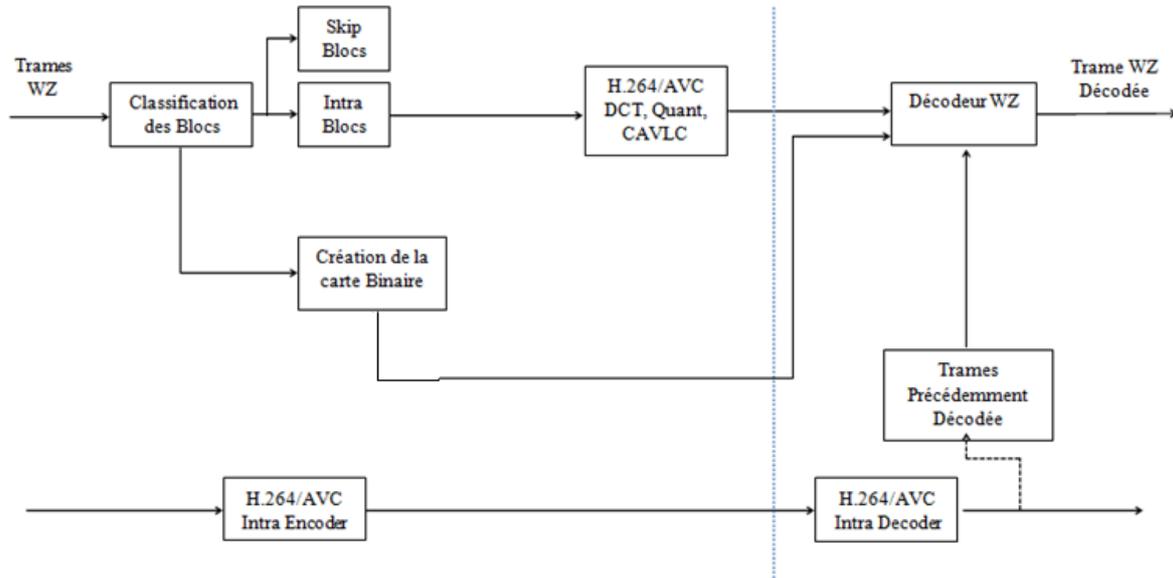
Il est utilisé dans cette thèse à des fins de comparaison avec le system FMD-DVC pour mettre en valeur l'effet d'ajouter un autre mode WZ qui est aussi simple et qui permettra d'améliorer la performance de codage.

Le schéma synoptique de ce système est présenté ci-dessous,

### II.4.1 Codeur IOFMD-VC

Le codeur de ce système est extrêmement simple, car il n'y a qu'une seule opération effectuée qui est celle de la classification des blocs. Tout comme le système précédant, une classification des trames est d'abord réalisée au niveau du codeur. Les trames Intra sont codées par H.264/AVC et les trames Inters sont divisées en blocs (4x4) et classifiées en deux types : Intra et Skip, par rapport à un seul seuil  $E_{Th-Intra}$  en utilisant le paramètre  $Moy_{min}$ , tel que présenté précédemment.

Les blocs Intra sont alors codés par codage source [25] et envoyés, une carte binaire contenant des coefficients indiquant la position et la nature de chaqu'un est aussi envoyée après avoir de même été codée par codage source.



**FIG II.9** Schéma synoptique du système IOFMD-VC

## II.4.2 Décodeur IOFMD-VC

L'opération de décodage est aussi simple que celle du codage. Une fois la carte binaire ainsi que les blocs Intra reçus et décodés. Chaque bloc sera placé dans sa position respective dans la trame Inter en utilisant la carte binaire.

Ainsi, si un coefficient indique que le bloc  $B_n$  est de nature Skip, ce dernier sera remplacé par un des blocs les plus proches  $B_{n-1}$  ou  $B_{n+1}$  dans les trames Intra décodées. Si ce dernier est Intra, il sera remplacé par le bloc Intra décodé correspondant.

## II.4 Conclusion

Dans ce chapitre nous avons présenté notre travail de façon détaillée, nous avons commencé par la présentation du schéma LCLD de [11], auquel nous avons introduit une sélection de modes au codeur (sans augmenter sa complexité). Ainsi que le remplacement de  $E_{max}$  par  $Moy_{min}$ , ce qui a eu pour effet de réduire le nombre de bloc Intra et WZ. Enfin, nous avons terminé avec le schéma IOFMD-VC, qui nous servira pour comparer les performances de notre système.

# Chapitre III

## *Résultats de simulation*

### III.1 Introduction

Dans ce chapitre, nous commencerons par présenter les différents résultats que nous avons obtenus. Nous comparerons ces résultats à ceux d'autres systèmes de codage vidéo. Cette comparaison sera réalisée suivant trois critères qui sont,

- ✓ La courbe Distorsion-Rate (R-D).
- ✓ La qualitévisuelle.
- ✓ La complexité du système.

Nous conclurons par une synthèse des différents résultats obtenus.

### III.2 Résultats de simulations

Nous avons choisis de réaliser nos tests sur trois séquences vidéo de 297 trames chacune, à savoir «Hall», «Coastguard», et "Foreman". Les trois séquences sont d'une résolution QCIF avec une fréquence de 30 images par seconde (fps), celles-ci sont classées selon la vitesse de leurs mouvements de la plus faible («Hall»), à la plus rapide ("Foreman"). En passant par celles avec un mouvement moyen («Coastguard»). Nous avons calculé les résultats de la composante de luminance, pour chacune des séquences. De plus, nous avons réalisé des résultats pour une séquence ("Foreman") avec une fréquence de 15 images par seconde, ce qui signifie que le mouvement est plus sévère comparé aux séquences avec 30 fps. Les résultats des systèmes DISCOVER, H.264 Baseline Intra Only et H.264 Baseline GOP : IPIP auxquels nous avons comparé nos résultats, ont été extraits à partir de la source [10] après les avoir confirmé au près de leurs sites respectifs. Ceux du système RFR ont été extrait de leurs travaux réalisé dans [11]. Enfin, notez que pour la séquence Foreman (15fps), les résultats du système H.264 sont en mode H.264/AVC Intra only et H.264/AVC Zero Motion.

Nos résultats ont été obtenus par l'utilisation des paramètres suivants,

$Q_p$	38	36	34	30	26	24
-------	----	----	----	----	----	----

**TAB III.1** Paramètres de quantification de la séquence Foreman 30fps.

$Q_p$	38	36	34	32	28	24
-------	----	----	----	----	----	----

**TAB III.2** Paramètres de quantification de la séquence Coastgards 30fps.

$Q_p$	38	36	32	30	28
-------	----	----	----	----	----

**TAB III.3** Paramètres de quantification de la séquence Hall 30fps.

$Q_p$	36	34	30	28	22
-------	----	----	----	----	----

**TAB III.4** Paramètres de quantification de la séquence Foreman 15fps.

Le paramètre de quantification des blocs Intra dans la trame WZ a été fixé à,  $Q_I = 30$ .

### **Choix des seuils Skip e Intra**

D'après les tests que nous avons effectués nous avons décidé de choisir les seuils suivants,

- [30-65] : Pour les systèmes RFR-1 et RFR-2.
- [30-70] : Pour le système FMD-DVC.
- [10] : Pour le IOFMD-VC (ce système comporte deux modes Skip et Intra, d'où la nécessité d'un seuil)

Car nous avons jugé que les performances des systèmes que nous avons réalisé sont plus réalistes pour les seuils ci-dessus. Pour exemple, nous avons testé les performances du système FMD-DVC pour trois seuils différents et ce pour les trois séquences que nous avons choisis. Les résultats de ces tests ont été réalisé avec une quantification de  $Q_p = 26$  et sont exposés dans le tableau III.1. Nous avons également cité le nombre moyen (par trame) des blocs classé Intra ( $Nb_I$ ), Skip ( $Nb_S$ ), ou WZ ( $Nb_{wz}$ ),

Seuils Séquences	[10-25]	[30-70]	[40-90]
Hall	PSNR=38.60 Rate=369.04 $Nb_I=2$ $Nb_S=1568$ $Nb_{WZ}=15$	PSNR=38.57 Rate=368.53 $Nb_I=0$ $Nb_S=1582$ $Nb_{WZ}=1$	PSNR=38.56 Rate=368.51 $Nb_I=0$ $Nb_S=1583$ $Nb_{WZ}=0$
Coastgard	PSNR=35.56 Rate=538.65 $Nb_I=23$ $Nb_S=1406$ $Nb_{WZ}=177$	PSNR=35.51 Rate=534.6 $Nb_I=0$ $Nb_S=1570$ $Nb_{WZ}=13$	PSNR=35.49 Rate=534.37 $Nb_I=0$ $Nb_S=1578$ $Nb_{WZ}=5$
Foreman	PSNR=35.5 Rate=398.30 $Nb_I=32$ $Nb_S=1396$ $Nb_{WZ}=187$	PSNR=35.36 Rate=391.62 $Nb_I=0$ $Nb_S=1564$ $Nb_{WZ}=19$	PSNR=35.34 Rate=391.40 $Nb_I=0$ $Nb_S=1576$ $Nb_{WZ}=7$

**TAB III.5** Résultats de simulation avec trois seuils différents.

D'après les résultats de ce tableau, il est possible de constater que les seuils que nous avons choisis sont mieux adaptés pour le concept WZ, car le fait de choisir des seuils inférieurs donnerait, certes une meilleure qualité, mais aussi un cout plus important. D'autre part, le choix de seuils plus élevés donnerait un nombre de blocs WZ très réduit (parfois nul), au risque de sortir du contexte de codage WZ. Cependant, il est possible de choisir n'importe quelle valeur pour ces seuils, suivant la nature de la situation.

### III.2.1 Critère Rate-Distorsion

La courbe R-D est réalisée en utilisant une mesure de la Distorsion par PSNR (Peak-Signal-to-Noise Ratio) en fonction du débit binaire. Le PSNR en (dB) est représenté par,

$$PSNR = 10 \log_{10} \frac{(255)^2}{MSE} \quad (19)$$

Où, MSE est l'erreur quadratique moyenne entre l'image originale et l'image décodée.

#### III.2.1.1 Implémentation du système LCLD-DVC

Nous commençons donc notre travail par l'implémentation du système LCLD-DVC (RFR) de [11] (nous l'appellerons DVC Conventionnel avec  $E_{max}$  ou RFR-1). Les figures ci-dessous représentent les résultats de simulation en termes de R-D obtenues.

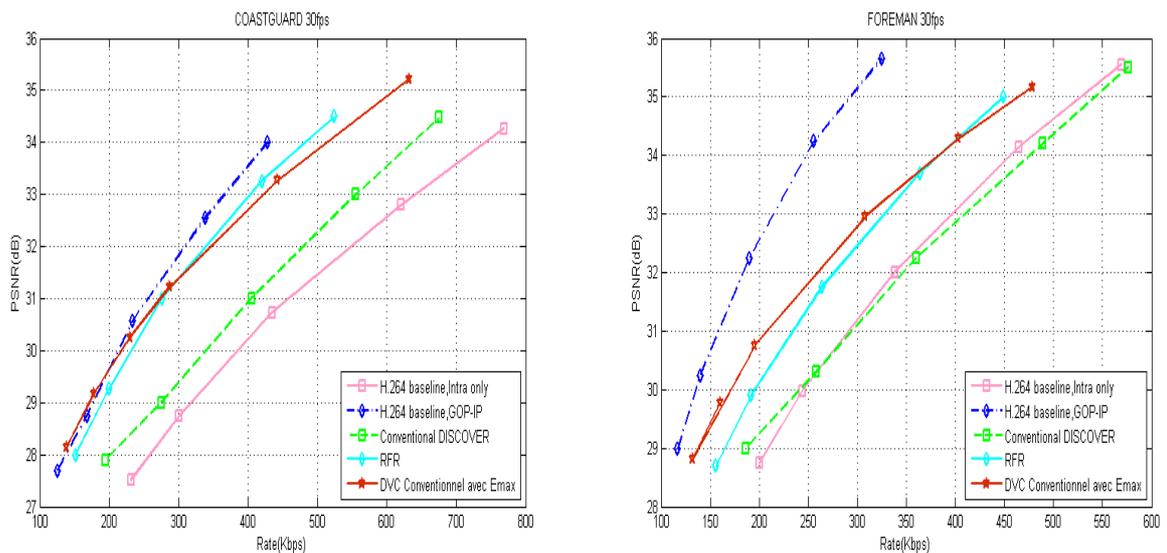
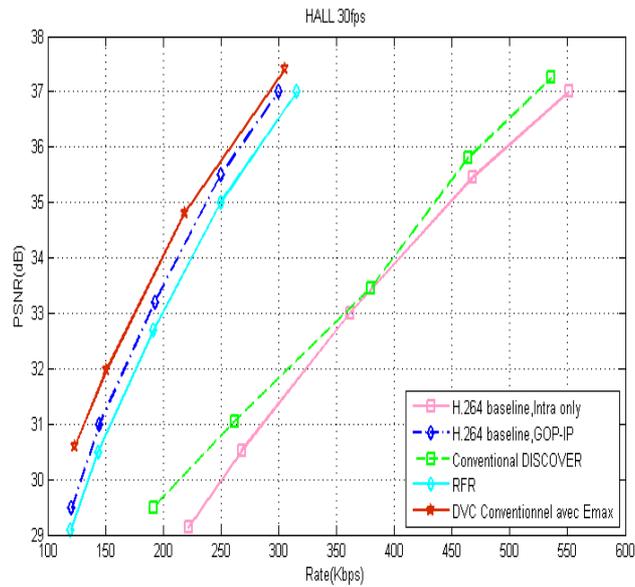


FIG III.1 R-D des séquences Coastgard et Foreman (30fps) de RFR-1.



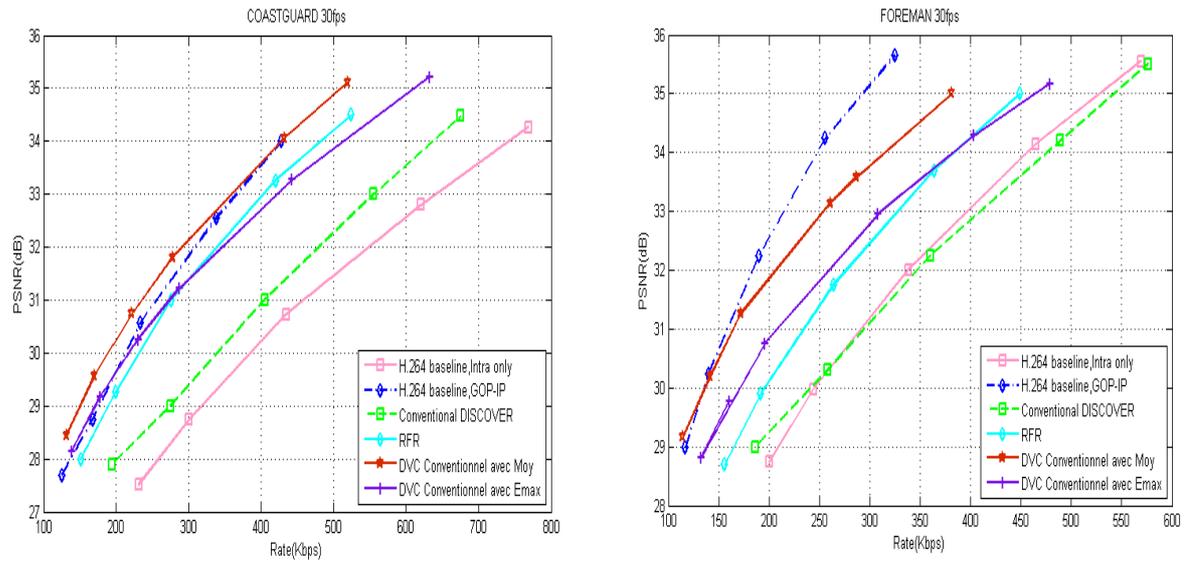
**FIG III.2** R-D de la séquence Hall (30fps) de RFR-1.

Les figures III.1 et III.2 représente la courbe R-D des trois séquences Foreman, Coastgard et Hall du système RFR-1.

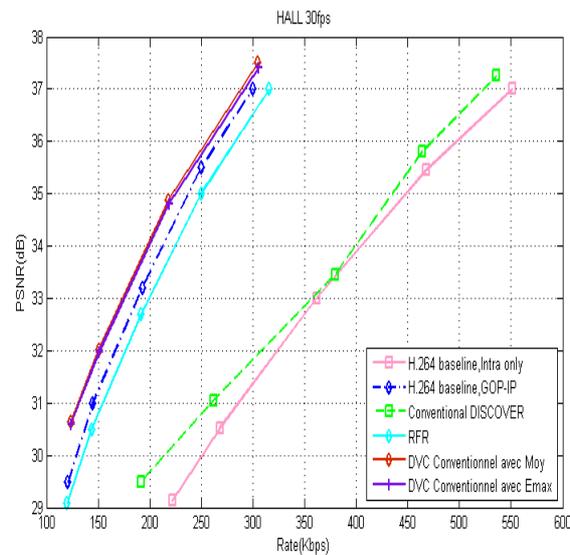
D'après les courbes obtenues, il est clair que les résultats de notre système ne sont pas identiques à celles de [11], ceci est due au fait qu'elles ne sont pas forcément implémentés de la même façon, car nous les avons créé nous-même. Cependant, les résultats sont très proches de celles de [11], pour les séquences Coastgard et Foreman et sont même légèrement supérieurs dans certains cas comme pour la séquence de Hall.

### III.2.1.2 Remplacement du paramètre $E_{max}$ par $Moy_{min}$

La première modification que nous avons introduit consiste à remplacer le critère de sélection  $E_{max}$  par  $Moy_{min}$  (chapitre II), nous avons dénommé ce système DVC Conventiennel avec  $Moy_{min}$  (ou RFR-2). Les figures ci-dessous, seront dédiées à la comparaison entre les résultats obtenues par l'utilisation des deux critères de sélection  $E_{max}$  et  $Moy_{min}$ .



**FIG III.3** R-D des séquences Coastgard et Foreman (30fps) de RFR-2.



**FIG III.4** R-D de la séquence Hall (30fps) de RFR-2.

D'après les figures III.3 et III.4, les résultats que nous avons obtenues avec le critère  $Moy_{min}$  montre une net amélioration des performances du système RFR-2 comparé à celles de RFR-1 pour les séquences a fort et moyen mouvement qui peut atteindre 1dB de Gain. Par ailleurs, elles demeurent très proches de celles de RFR-1 pour la séquence à faible mouvement avec un Gain d'environ 0.1 dB. Ces résultats s'expliquent par le fait que pour les séquences à fort et moyen mouvement, les

différences entre les pixels d'un bloc à un autre peuvent changer de manière brusque et très rapidement. Tel que nous l'avons expliqué dans le chapitre précédent, ces changements peuvent se produire sur un petit nombre de pixels dans un bloc (parfois un ou deux pixels) tandis que les autres valeurs des pixels ne changent pas trop. Ainsi, l'utilisation du critère  $E_{max}$  dans ce cas aurait pour effet de produire beaucoup de blocs Intra coûteux (parfois un bloc est classé Intra à cause d'un seul pixel). Alors que l'utilisation du critère  $Moy_{min}$  aurait tendance à produire plus de blocs Skips et WZ (peu coûteux).

Le gain que nous avons obtenu nous vient donc essentiellement du débit binaire économisé en utilisant  $Moy_{min}$  au lieu de  $E_{max}$ . Ceci est confirmé par la séquence à faible mouvement, car les pixels ne changent pas beaucoup d'un bloc à un autre et donc la moyenne des différences n'est pas très loin de celle du maximum de différences. Et le classement des blocs demeure pratiquement le même dans les deux cas.

### III.2.1.3 Introduction de la sélection de modes au codeur

La deuxième modification que nous avons réalisée consiste à introduire la sélection des modes au niveau du codeur, nous avons appelé ce système FMD-DVC (Fast Mode Decision DVC). Les résultats de ce système sont présentés ci-dessous

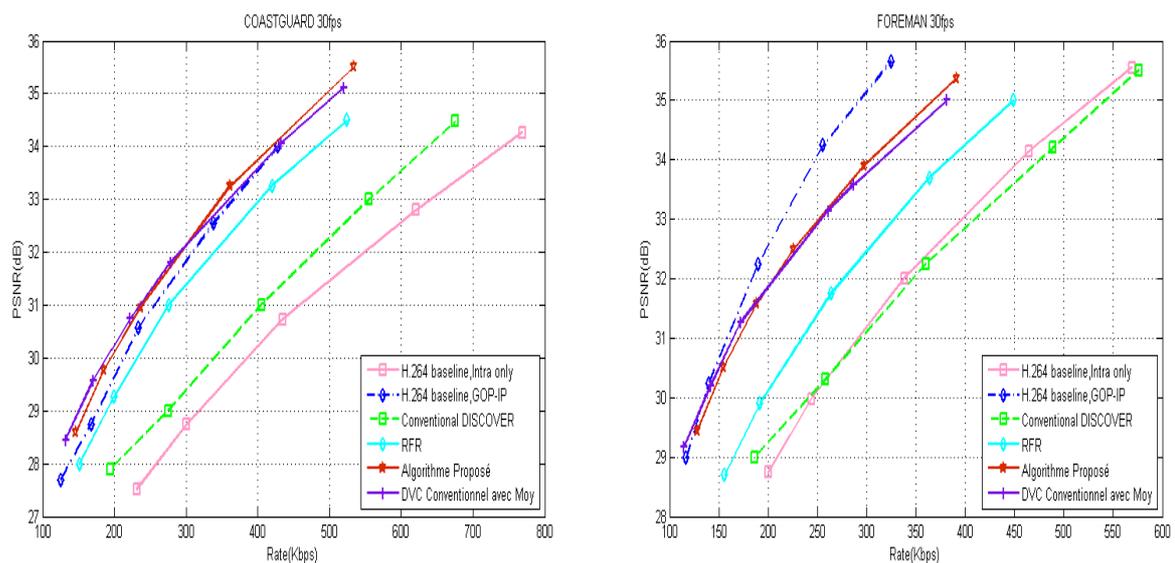
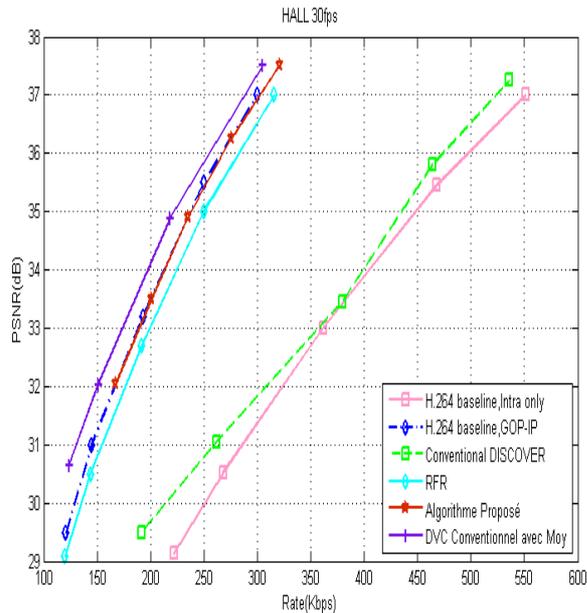


FIG III.5 R-D des séquences Coastgard et Foreman (30fps) de FMD-DVC.



**FIG III.6** R-D de la séquence Hall (30fps) de FMD-DVC.

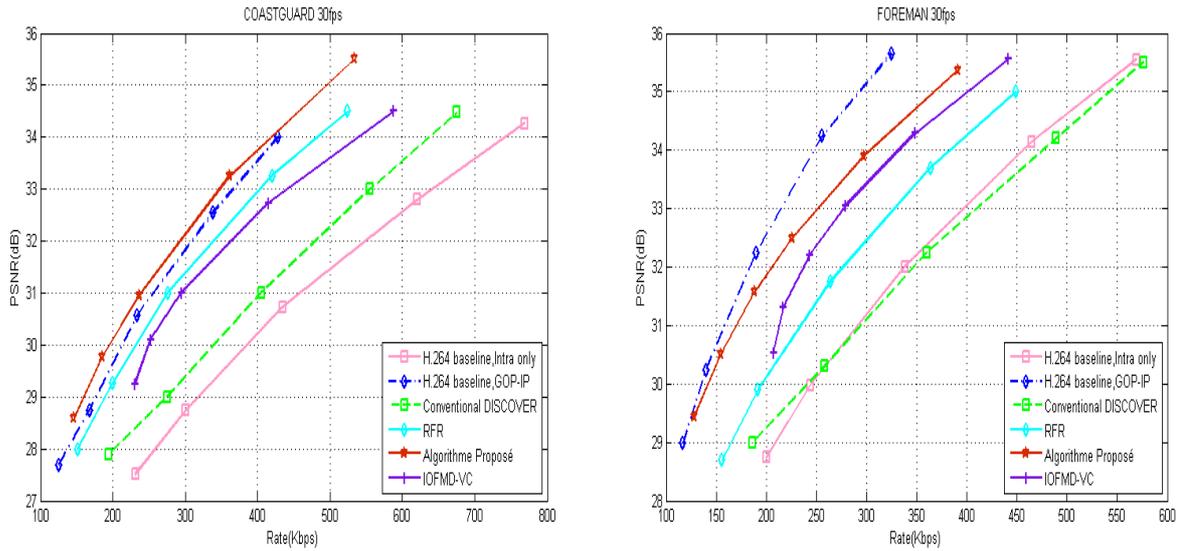
D'après les figures III.5 et III.6, l'introduction de la sélection des modes au codeur apporte une légère amélioration des performances d'environ 0.3 dB de Gain par rapport à RFR-2 pour les séquences à mouvement rapide (Foreman) et moyen (Coastgard), mais pour la séquence à mouvement long (Hall), les performances de ce système sont légèrement inférieures.

Mais il faudrait noter que pour la sélection des modes au codeur, la carte binaire comprenant le type du bloc est comptabilisée dans le débit binaire total, tandis que pour RFR, cette dernière est transmise par le canal de retour et n'est donc pas comptabilisée. Ainsi, si l'on déduit le coût de cette carte qui est d'environ 49Kbits/s, il serait possible d'avoir un Gain d'environ 0.5 dB pour toutes les séquences. Ainsi, la sélection des modes au codeur est avantageuse dans le cas où la séquence est à moyen ou rapide mouvement comme est le cas de Coastgards et Foreman.

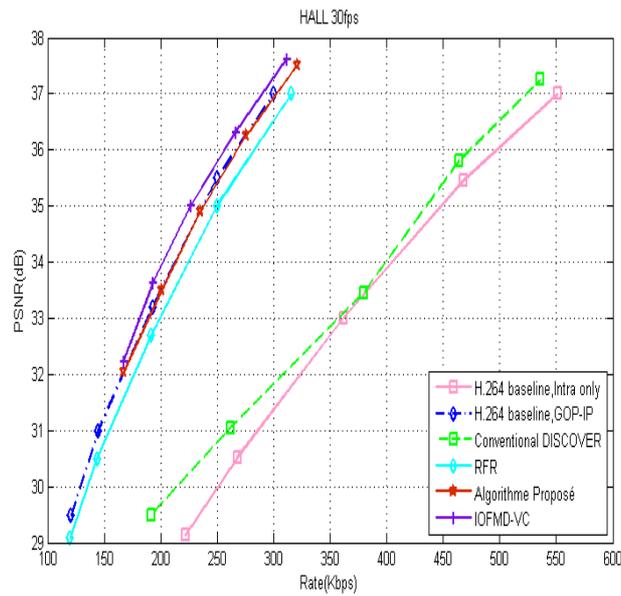
Contrairement aux séquences à mouvement long (Hall), où la qualité de la SI est nettement meilleure et ne nécessite pas vraiment plus d'information du codeur pour la corriger. Cette supposition est donc confirmée par les résultats obtenus.

### III.2.1.4 Comparaison des résultats de FMD-DVC avec IOFMD-VC

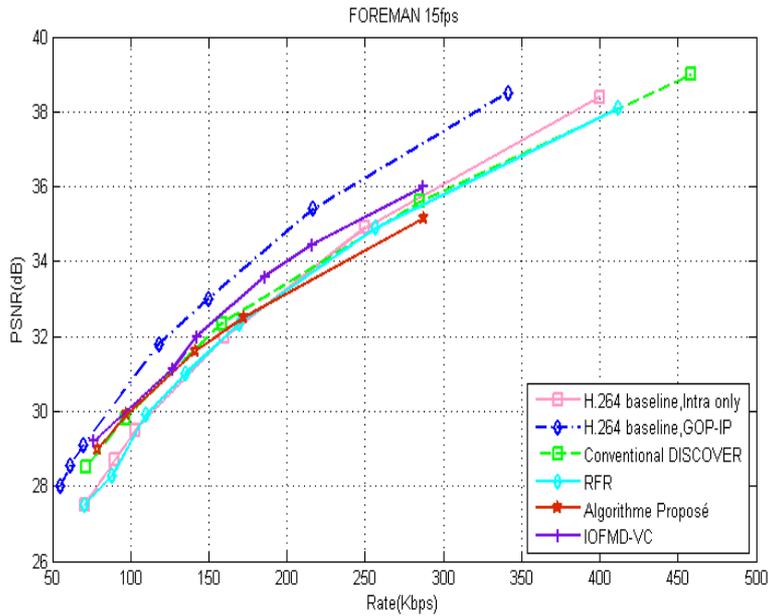
Enfin, les dernières figures R-D seront consacrées à l'exposition des résultats obtenus par le système IOFMD-VC comparé à ceux de FMD-DVC,



**FIG III.7** R-D des séquences Coastgard et Foreman (30fps) avec IOFMD-VC.



**FIG III.8** R-D de la séquence Hall (30fps) avec IOFMD-VC.



**FIG III.9** R-D de la séquence Foreman (15fps)

D'après les figures ci-dessus, nous constatons que les performances du système IOFMD-VC sont inférieures à celles de FDM-DVC d'environ 1,5 dB, pour les séquences à fort et moyen mouvement (Foreman et Coastgard à 30fps). Néanmoins, les performances de ce dernier (IOFMD-VC) sont légèrement supérieures (jusqu'à 0,7 dB) à celles de FMD-DVC pour les séquences à faible mouvement (Hall avec 30fps) ainsi que pour les séquences à mouvement extrêmement rapide (Foreman avec 15fps).

Tel que nous l'avons expliqué précédemment, les résultats de IOFMD-VC sont meilleurs pour les séquences à long mouvement car il n'y a pas de grandes différences entre les images consécutives. Ainsi, l'utilisation d'une sélection de modes au codeur permet d'envoyer juste les blocs où se produit un changement (Intra), sans pour autant dégrader la qualité de la vidéo. Par ailleurs, dans le cas des séquences à moyen et rapide mouvement les performances de FMD-DVC sont meilleures du fait qu'il y ait beaucoup de changements entre les images et l'utilisation d'un troisième mode (WZ) ceci est nécessaire pour améliorer les résultats. Enfin, pour les séquences à très fort mouvement les performances de l'IOFMD-VC sont meilleures car tout simplement, le mouvement est tellement rapide que la différence entre les trames consécutives devient extrêmement grande et de ce fait, les blocs

Intra sont prédominants par rapport aux blocs Skip et WZ. Ce qui explique aussi que les performances du système H.264/AVC Intra only sont meilleures que celles de DISCOVER.

### III.2.2 Critère Qualité Visuelle

Nous avons choisi d'utiliser ce critère afin de comparer les performances des systèmes que nous avons réalisé à savoir FMD-DVC et IOFFMD-VC, car si le PSNR reste le critère phare pour mesurer la qualité de la compression. Il peut s'avérer dans certains cas que l'aspect visuel de la vidéo ne soit pas forcément dans la même optique. Nous avons donc pris des échantillons d'images pour chacun des séquences que nous avons utilisé et ceux avec un certain pas de quantification. Les figures ci-dessous représente la Qualité Visuelle des six séquences avec,

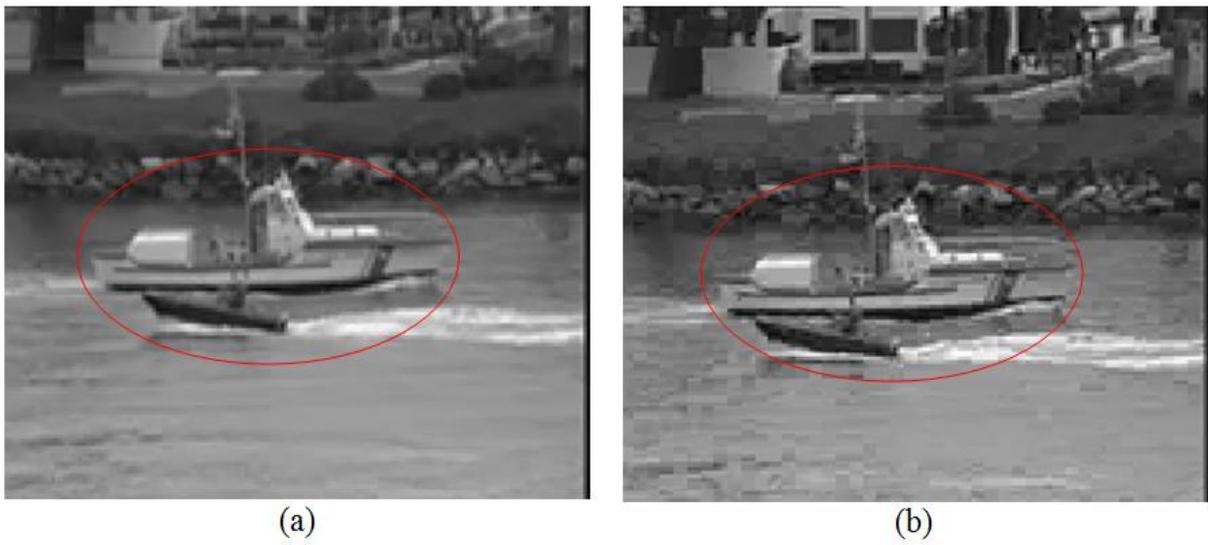


**FIG III.10** Image 256 de la séquence Foreman (30 fps) avec  $Q_p = 26$ . (a) FMD-DVC, (b) IOFFMD-VC.



**FIG III.11** Image 23 de la séquence Hall avec  $Q_p = 28$ .

(a) FMD-DVC, (b) IOFMD-VC.



**FIG III.12** Image 73 de la séquence Coastgardavec  $Q_p = 26$ .

(a) FMD-DVC, (b) IOFMD-VC.



**FIG III.13** Image 71 de la séquence Foremane (15 fps) avec  $Q_p = 22$ .

(a) FMD-DVC, (b) IOFMD-VC.

D'après ces images, on pourrait observer deux zones distinctes. L'une étant la zone de mouvement et l'autre, la zone statique c'est en se basant sur ces deux zones que l'on peut classer les performances de chaque système. Le cercle en rouge dans les images précédentes délimite la zone de mouvement critique.

Ainsi, si l'on observe la zone statique de l'image, il est clair que le système FMD-DVC présente une meilleure qualité visuelle (image plus lisse) que celle de IOFMD-VC (image contrasté), même s'il existe des filtres pour gommer cet aspect de bloc par bloc. D'autre part, concernant les zones de mouvement (et plus spécialement de fort mouvement), on constate que le système IOFMD-VC présente un meilleur aspect que celui de FMD-DVC.

Enfin, les images sélectionnées représentent des parties critiques de la séquence vidéo et non pas toute la séquence en question. La différence entre ces deux systèmes réside dans le fait que l'un utilise une estimation du mouvement et l'autre pas. Pour le FMD-DVC, l'utilisation de l'EM produit une image lisse (bonne estimation) dans le cas où le mouvement n'est pas important (ou pour les images statiques) et un effet de bloc (de taille plus ou moins importante) si le mouvement est conséquent (zone a grand changement), d'autant plus si la correction de ces erreurs n'est pas suffisante.

D'autre part, ce problème existe aussi dans le cas de l'IOFMD-VC (effet de blocs) mais a plus petite échelle. Ceci s'explique par le fait que d'abord, les blocs sont de taille moins importante (4x4) mais aussi car la valeur du seuil de classification est assez petite (10).

Ce qui veut dire que si le changement moyen d'un bloc à un autre est inférieur à 10 (ce qui correspond à moins de 5% de différence d'un bloc à un autre, sachant que le changement max entre deux pixels est de 255) ce bloc ne change donc pas vraiment, dans le cas contraire ce dernier est codé en Intra. Ainsi, si le système FMD-DVC présente une meilleure qualité visuelle dans le cas des zones à faible et moyen mouvement, l'IOFMD-VC est quant à lui plus résistant dans le cas des zones à fort mouvement, ceci est d'ailleurs confirmé par ces performances élevées (en termes de R-D) dans le cas de la séquence de Foreman avec 15fps.

### III.2.3 Complexité du système

Le rôle de ce critère est de mesurer la complexité du système, cette mesure consiste à calculer le temps (en secondes) que met le processeur pour coder et décoder une séquence vidéo de 300 trames à une fréquence de 30fps. Nous avons réalisé ces tests pour les trois séquences Foreman, Coastguard et Hall pour un pas de quantification  $Q_p = 34$ .

Système Séquence	H.264 Intra Only	Le schéma proposé	Le schéma de référence avec $Moy_{min}$	IOFMD-VC
Foreman	69.8	223.8	247.7	39.9
Coastguard	81.55	213	267.21	46.15
Hall	58.8	258.8	177.4	33.4
Temps de calcul Moyen	$T_{réf} = 74$	234.8 $3.2 \times T_{réf}$	228.7 $3 \times T_{réf}$	41.78 $0.5 \times T_{réf}$

**TAB III.6** Temps de Processeur (s) et  $Q_p = 34$ .

En s'appuyant sur le temps de calcul moyen de chaque système comparé au H.264 Intra Only (dans notre machine), nous pouvons constater que le temps de calcul des systèmes RFR-2 et FMD-DVC est trois fois plus important. Ceci peut sembler important mais demeure relativement raisonnable comparé à celui de [12] qui est presque dix fois plus important.

La simplicité du système IOFMD-VC est encore une fois confirmé lorsqu'on voit le temps de calcul de ce dernier qui est presque deux fois moins important que celui de H.264 Intra.

Enfin, malgré le fait que nous ayons introduit la sélection des modes au codeur pour le système FMD-DVC. Le temps de calcul de ce dernier reste très proche de celui de RFR-2, ce qui signifie que le codeur n'est pas plus complexe car la procédure de classification repose sur de simples opérations arithmétiques de soustraction et de comparaison.

### **III.3 Conclusion**

Nous avons dédié ce chapitre à la présentation et la discussion de nos résultats de simulation en se basant sur trois critères de comparaisons entre les performances les systèmes que nous avons réalisés et ceux de référence. Les résultats obtenus par notre système démontrent une amélioration des performances, comparées au système de référence, ceci confirme notre hypothèse concernant le fait de remplacer  $E_{max}$  par  $Moy_{min}$ . Ainsi que l'introduction de la sélection des modes au codeur, même si cette dernière est moins efficace lorsqu'il s'agit d'une séquence a mouvement long.

## ***Conclusion Générale***

Si l'état de l'art du codage DVC dit DISCOVER fondé sur l'architecture Stanford est une référence dans le domaine, ses capacités restent pour le moins limitées et même si le codeur est plus simple, l'application du codage canal et l'utilisation fréquente du canal de retour ont pour effet, d'augmenter la complexité et la latence du décodeur.

La pluparts des recherches effectuées jusque-là tourne autour de la réduction du nombre de requête entre le décodeur et le codeur et ce en calculant préalablement le nombre de syndromes minimum nécessaire pour la correction de la SI par exemple. Cependant, des chercheurs ont proposé d'autres alternatives au codage canal et ont proposé des systèmes DVC avec un codage source beaucoup plus simple et moins couteux (en termes de matériel) que le codage canal. Et une nouvelle façon de corriger la SI a été proposé par [10][11][12].

En s'appuyant sur ces recherches nous avons proposé un schéma DVC dit FMD-DVC basé sur les blocs, sans codage canal et une seule requête par trame WZ du décodeur au codeur. Nous avons aussi utilisé une classification des blocs au codeur en s'appuyant sur des seuils prédéfinis. Nous avons aussi utilisé un système dit IOFMD-VC avec sélection de modes très simple afin d'évaluer les performances de notre système.

Les résultats de simulations obtenu s par les deux systèmes sont meilleurs que ceux proposé par [12] pour les séquences de 30fps et sont légèrement inférieurs pour la séquences à 15fps.

Aussi, nous avons constaté que le IOFMD-VC en plus d'être beaucoup plus simple, est plus efficace que le FMD-DVC pour les séquences de 30fps à long mouvement et les séquences à 15fps. Ce qui le rend très attrayant pour les applications de réseaux de capteurs vidéo sans fil par exemple.

Nous avons ainsi réalisé un système DVC beaucoup plus simple et rapide que ceux basés sur l'architecture Stanford avec le codage canal mais aussi avec de meilleures performances RD. Tout en gardant le principe de simplification du codeur.

## ***Perspectives***

En vue des différentes lectures et recherches que nous avons effectué dans le domaine DVC, et en tirant conclusion de l'efficacité de la méthode proposée par

[12] et des améliorations apportées par nous-même a cette dernière. Une perspective de mettre en pratique ces systèmes s'avère être la meilleure voie pour améliorer nos connaissances dans le domaine et se confronter à la réalité du terrain.

## ***BIBLIOGRAPHIE***

- [1] Iain E. Richardson "*The H.264 Advanced Video Compression Standard*". A John Wiley and Sons, Ltd., Publication. United Kingdom, 2010.
- [2] J. D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," IEEE Trans. Inf. Theory, vol. IT-19, pp. 471–480, Jul. 1973.
- [3] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," IEEE Trans. Inf. Theory, vol. IT-22, no. 1, pp. 1–10, Jan. 1976.
- [4] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed Video Coding," in Proc. IEEE, Special Issue on Video Coding and Delivery, vol. 93, pp. 71–83, Jan. 2005. Invited Paper.
- [5] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video," in Proc. Asilomar Conference on Signals and Systems, Pacific Grove, California, Nov. 2002
- [6] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed Video Coding," in Proc. IEEE, Special Issue on Video Coding and Delivery, vol. 93, pp. 71–83, Jan. 2005. Invited Paper.
- [7] R. Puri and K. Ramchandran, "PRISM: A New Robust Video Coding Architecture Based on Distributed Compression Principles," 40<sup>th</sup> Allerton Conference on Communication, Control and Computing, Allerton, USA, Oct. 2002.
- [8] R. Puri, A. Majumdar and K. Ramchandran, "PRISM: A Video Coding Paradigm with Motion Estimation at the Decoder," IEEE Trans. On Image Processing, vol. 16, no. 10, pp. 2436 - 2448, Oct. 2007.
- [9] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret. "The DISCOVER codec: Architecture, Techniques and Evaluation". In Picture Coding Symposium (PCS'07), 2007.
- [10] G. Huchet and D. Wang, "Distributed video coding without channel codes," in Proc. IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, Shanghai, China, Mar. 2010.

- [11] G.Qiu, S.Gao and G.Tu “*Low-Complexity Low-Delay Distributed Video Coding*,” in Proc. IEE, International Conference on Digital Signal Processing (DSP), Fira, Grèce, July 2013.
- [12] G. Huchet, J.-Y.Chouinard, D. Wang, and A. Vincent, “*Adaptive Source Representation for Distributed Video Coding*,” in Proc. IEEE International Conference on Image Processing, Cairo, Egypt, Nov. 2009.
- [13] A. Trapanese, M. Tagliasacchi, S. Tubaro, J. Ascenso, C. Brites, and F. Pereira, “*Embedding a Block-based Intra Mode in Frame-based Pixel Domain Wyner-Ziv Video Coding*”,*International Workshop on Very Low Bitrate Video Coding*, Sardinia, Italy, September 2005.
- [14] J. Ascenso, F. Pereira “*Low Complexity Intra Mode Selection For Efficient Distributed Video Coding*” , IEEE. ICME 2009.
- [15] T. Clerckx, A. Munteanu, J. Cornelis, and P. Schelkens, “*Distributed Video Coding with Shared Encoder/Decoder Complexity*”, IEEE ICIP, San Antonio, TX, USA, September 2007.
- [16] Abdul Rehman, “*Rate-distortion optimized rate allocation for pixel-domain Wyner-Ziv video coding without feed-back channel*,” Master Thesis, Technische Universitat Munchen Institute for Media Technology, Felsennel kenanger 0980937 Munich, Sep .2009
- [17] Shannon, C.E. (1948), “*A Mathematical Theory of Communication*”, *Bell System Technical Journal*, 27, pp. 379–423 & 623–656, July & October, 1948.
- [18] J. Sun “*An Introduction to Low Density Parity Check (LDPC) Codes*” Wireless Communication Research Laboratory Lane Dept. of Comp. Sci. and Elec. Engr. West Virginia University. 2003
- [19] J. Ascenso, C. Brites, F. Pereira, “*Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding*”, in: Proceedings of EURASIP Conference on Speech and Image Processing, Multimedia Communications and Services(EC- SIPMCS), Smolenice, Slovak Republic, June 2005.
- [20] R. Puri and K. Ramchandran. PRISM: “*An Uplink-Friendly Multimedia Coding Paradigm*”. In IEEE International Conference on Acoustics, Speech, and Signal Processing, Proceedings.(ICASSP '03), volume 4, pages 856–9, April 2003.

- [21] Aparna P, Sumam David S., ''*Efficient Distributed Video Coding based on principle of syndrome coding*'', International Journal of Signal and Imaging Systems Engineering, Special Issue on Image and Data Compression Applications, Inderscience Publishers, Vol. 4, No,4, pp. 212-219, 2011.
- [22] Aparna P, Sumam David S., ''*Low Complexity Distributed Video Coding with Golay Codes*'', International Journal of Machine Learning and Computing vol. 2, no. 4, pp. 466-470, August 2012
- [23] M. Calhau, ''*Low Complexity Video Coding for Sensor Networks*'', Master Thesis. Engenharia Electrotécnica e de Computadores. Universidade Técnica de Lisboa.
- [24] Y. Pang, ''*Easier Encoding-Decoding Distributed Arithmetic Coding For Slepian-Wolf Coding*'', Springer-Verlag Berlin Heidelberg F.L Wang et al. (Eds): CMSP 2012, CCIS 346, pp. 523-530, 2012.
- [25] D. Marpe, H. Schwarz, G. Blättermann, G. Heising, T. Wiegand, ''*Context-Based Adaptive Binary Arithmetic Coding in JVT/H.26L*'', IEEE International Conference on Image Processing (ICIP), Rochester, September 2002.