Université de Nice - Sophia Antipolis

École doctorale STIC

Sciences et Technologies de l'Information et de la Communication

THÈSE

pour obtenir le titre de

Docteur en Sciences

de l'Université de Nice - Sophia Antipolis

Mention : Automatique, Traitement du Signal et des Images

présentée par

Marie Andrée AGOSTINI

Équipe d'accueil : CReATIVe - Laboratoire I3S

NOUVELLES APPROCHES POUR LA COMPRESSION DE VIDEOS HAUTE DEFINITION - APPLICATION AU CODAGE PAR DESCRIPTIONS MULTIPLES

Thèse dirigée par Marc ANTONINI, Directeur de Recherche - CNRS

Soutenue le 26 Juin 2009 devant le jury composé de

Antonio Ortega	University of Southern California, USA	Rapporteur
Philippe Salembier	University of Catalonia, Spain	Rapporteur
Béatrice Pesquet-Popescu	Telecom'Paris-Tech, France	Rapporteur
Michel Kieffer	L2S, France	Examinateur
Joël Jung	Orange Labs, France	Examinateur
Michel Barlaud	Université de Nice-Sophia Antipolis	Examinateur
Marc Antonini	Université de Nice-Sophia Antipolis	Directeur de thèse

University of Nice - Sophia Antipolis

Graduate school STIC

Sciences et Technologies de l'Information et de la Communication

PhD THESIS

A dissertation submitted in partial satisfaction of the requirements for the degree of

Doctor of Science

Specialized in Signal and Image Processing

presented by

Marie Andrée AGOSTINI

prepared at CReATIVe - Laboratoire I3S

NEW TRENDS IN HIGH DEFINITION VIDEO COMPRESSION - APPLICATION TO MULTIPLE DESCRIPTION CODING

Thesis supervised by Marc ANTONINI, Directeur de Recherche - CNRS

Defended the 26th of June 2009 in front of

Béatrice Pesquet-Popescu	Telecom'Paris-Tech, France	Reviewer
Philippe Salembier	University of Catalonia, Spain	Reviewer
Antonio Ortega	University of Southern California, USA	Reviewer
Michel Kieffer	L2S, France	Examiner
Joël Jung	Orange Labs, France	Examiner
Michel Barlaud	University of Nice-Sophia Antipolis	Examiner
Marc Antonini	University of Nice-Sophia Antipolis	Advisor

Acknowledgements

Only my name appears as writer in the front page of this PhD thesis, but in reality, many people have contributed to it, and deserve to be thanked.

First, I would like to thank Marc Antonini, "le directeur de thèse modèle", for supervising my PhD thesis. I greatly acknowledge him for his wise advices and his involvements in our research. Without him I would not have been able to accomplish this thesis work.

Next, I would also like to express my deep thanks to Béatrice Pesquet-Popescu, Antonio Ortega, and Philippe Salembier for having accepted to review my manuscript despite the important amount of work this required. I am also extremely grateful to Michel Barlaud, Michel Kieffer, and Joel Jung for their participation in my jury. I would like to thank all of them for their useful comments.

I address also a grateful thank to Professor Michel Barlaud, who welcomed me in his research team, and guided, with Marc, my first steps as a researcher. A special thank goes to Micheline, what would we do if you were not there?

I would like to thank Ebroul Izquierdo, for accepting me as a postdoc in his team in Queen Mary, University of London.

I would also like to thanks all my lab colleagues and friends for those great years: Adrien, Alex, Aline, André, Andrew, Anis, Ariane, Arnaud, Aymen, Benoit, Céd, Changy, Christian, David, Elena, Eric D, Eric W, Estevao, Francois, Fred, Gaelle, Hala, Khaled, Lao, Laure, Laurent, Marco, Moger, Muriel, Paolo, Peter Parker, Ronald, Sandrine, Seb x 2, Sofiane, Silvia, Syl, Stéphane, Thomas, Vincenzo, Xavier, Yasmine, Zohra, the almost-lab twins V & G, and all the forgotten...

I would like to spend some more words for some of my colleagues.
I have to thank Yasmine, my "co-bureau", for her smile and all the sup-

port.

Céd, thank you for your lovely accent, and your endless jokes which have illuminated our coffee-breaks.

I want to thank the "old" CReATIVe students who are now dear friends: Moger, Peter "Vincent" Parker, Lao (thank you to helping me improve my abilities in printing-stuffs), Thomas & Muriel, Marco, and Changy (I Know you're not a CReATIVe one, sorry).

Finally, thank you Syl, without you, nothing would have been the same.

Guys, I miss you already!

A very special thank goes to my parents, to whom I owe everything.

The last words go to Romain, without Whom I would be nothing. He supports me already for nine years, and He will forever, I hope...

Thank you all!

Sophia Antipolis, July 2009

Contents

List of Figures

Introduction

Le travail présenté dans cette thèse concerne principalement la compression vidéo et la transmission de vidéos.

Contexte général

Au cours de ces trente dernières années, notre civilisation aurait produit plus d'informations que pendant les cinq mille ans qui les ont précédées. En outre, plus de la moitié de l'information qui est maintenant créée est déjà sous forme numérique. Les progrès réalisés dans les domaines du réseau, des télécommunications et du stockage numérique ne sont pas suffisants pour assimiler une telle quantité de données. Parmi toutes ces données à stocker ou à transmettre, les données multimédia ont une place croissante. En particulier, dans les domaines de la parole, de l'image et de la vidéo, les techniques numériques ont définitivement remplacé les techniques analogiques. Les données représentées par la vidéo numérique (télévision numérique haute définition, cinéma numérique, visioconférence, internet ou communications mobiles) sont particulièrement vertigineuses. Mais le problème de stockage n'est pas le seul problème lié à l'explosion de la vidéo numérique. La transmission le long de nombreux réseaux sans perdre la qualité des données est un immense challenge. Les réseaux actuels atteignent de hauts débits de transmission et supportent des débits de données suffisants pour les applications vidéo, même dans les communications mobiles par exemple. Ces faits ouvrent un tout nouveau monde de communications.

Pour toutes ces raisons, la compression est devenue une étape indispensable dans la plupart des applications liées à la vidéo numérique. Les normes de codage vidéo tels que MPEG-2 (utilisée pour le DVD et la télévision numérique), MPEG-4 et H.264, ont connu un important succès industriel. Les dernières normes MPEG-4 et H.264 (MPEG-4 / Part 10) et leurs évolutions améliorent grandement le compromis entre le débit et la qualité des vidéos, et permettent de nouvelles fonctionnalités, comme la scalabilité. La scalabilité en codage vidéo consiste à extraire, à partir d'un seul fichier vidéo compressé, plusieurs versions de cette vidéo, en fonction du type de transmission et du support de stockage. Malheureusement, le

support de la scalabilité est limitée, en raison d'un manque de flexibilité et, souvent, de la dégradation des performances. En outre, les normes ne sont pas compatibles avec la norme de codage d'image JPEG2000 basée ondelettes.

Pour ces raisons, nous avons décidé, dans la première partie de cette thèse, d'explorer le codage vidéo basé ondelettes, car la transformée en ondelettes a récemment montré son efficacité. La transformée en ondelettes 3D a attiré l'attention du monde de la recherche dans le domaine de la compression de données. La communauté a espéré pouvoir reproduire les excellentes performances qu'a atteintes sa version 2D utilisée en codage d'images fixes. De plus, l'approche ondelettes fournit un support total de la scalabilité, ce qui semble être l'un des enjeux les plus importants dans le domaine du codage et de la transmission vidéo. Bien sûr, des améliorations sont encore possibles, notamment en ce qui concerne le codage des vecteurs mouvement.

En effet, la qualité des vidéos encodées serait améliorée si le mouvement était décrit d'une façon plus précise. Dans certains cas, le poids relatif des vecteurs mouvement dans le train binaire est trop important, surtout à bas débit. Dans ce travail, le codage des vecteurs mouvement est amélioré, et le compromis débit-distorsion entre l'information de mouvement et les coefficients d'ondelettes est optimisé, afin d'accroître l'efficacité du codage vidéo.

Une autre amélioration possible du codeur vidéo basé ondelettes concerne la compensation de mouvement. L'introduction de la compensation de mouvement dans la transformée en ondelettes temporelle (et dans sa version liftée) a montré une amélioration de l'efficacité de la décomposition en sous-bandes temporelles. Mais, l'influence de certains vecteurs mouvement mal estimés sur la transformée en ondelettes compensée en mouvement peut être minimisée. Une méthode pour le calcul du schéma lifting, l'implémentation la plus adaptée de la transformée en ondelettes temporelle, est donc proposée. Les pas du schéma lifting sont étroitement adaptés au mouvement.

Dans le cadre d'un contrat industriel avec Orange Labs, la précédente méthode de quantification des vecteurs mouvement est appliquée au codeur vidéo H.264. En effet, le compromis entre l'allocation des ressources de codage des vecteurs mouvement ou des coefficients de la transformée a une grande importance quand il s'agit de l'efficacité du codage vidéo. Néanmoins, dans les standards de codage vidéo, il est en général seulement possible de choisir indirectement la façon dont le débit est partagé entre les vecteurs mouvement et les coefficients, en choisissant parmi plusieurs modes de codage disponibles pour chaque macro-bloc. Par conséquent, quand une séquence est encodée à faible ou très faible débit, une grande partie des ressources est allouée aux vecteurs mouvement. Cela suggère que, dans le cadre d'un codeur H.264, les performances pourraient être améliorées si un nouveau mode de codage avec un mouvement moins coûteux était introduit.

La deuxième partie de ce travail est dédiée à la transmission de vidéos sur

canaux bruités, qui est une question ardue en communications numériques. Le problème de l'efficacité de la transmission vidéo implique de bons taux de compression et de la robustesse en présence d'échecs du canal. Le codage source-canal conjoint a reçu un intérêt croissant de la communauté de recherche. En particulier, le codage par descriptions multiples a déjà montré de bons résultats en tant que codage source conjoint robuste. Deux grandes familles de codage par descriptions multiples existent, selon que la redondance est introduite pendant ou avant la quantification. Dans le premier type d'approches, les quantificateurs sont conçus pour produire des descriptions redondantes du même signal. Dans la deuxième catégorie, la redondance est introduite au cours de la transformation du signal. Le signal original peut être reconstruit dès que l'une de ces descriptions est disponible. La disponibilité de la deuxième description permet d'augmenter la qualité du signal reconstruit. Le codage par descriptions multiples a été récemment appliqué au codage vidéo.

Le schéma de codage par descriptions multiples présenté dans ce travail est équilibré et basé ondelettes. Il appartient à la deuxième catégorie d'approches : la redondance est introduite avant la quantification du signal. Le principal but de cette étude, réalisée dans le cadre d'un projet de recherche national, le projet ANR " blanc" ESSOR, est le décodage optimal du signal après la transmission sur des canaux bruités. En effet, le challenge au décodeur est de reconstruire un signal avec une distorsion minimale. Des estimations par maximum a posteriori de la source originale sont réalisées, basées sur la connaissance des descriptions latérales corrompues par les erreurs de transmission. L'information a priori est représentée par un modèle décrivant la distribution des coefficients d'ondelettes des images. Deux approches différentes ont été mises en oeuvre.

La dernière partie de ce travail concerne une étude sur le codage vidéo distribué, réalisée dans le cadre du projet ESSOR. Le codage de source distribué a émergé comme une technologie intéressante pour les réseaux de capteurs. Il correspond à la compression de signaux corrélés capturés par différents capteurs qui ne communiquent pas entre eux. Tous les signaux capturés sont compressés de façon indépendante et transmis à une station centrale de base, qui a la capacité de les décoder conjointement. Le codage de source distribué a été récemment appliqué au codage vidéo, ce qui conduit au codage vidéo distribué. Contrairement aux schémas classiques de codage vidéo, le codage vidéo distribué effectue un codage intra-frame d'images corrélées (sans exploiter les éventuelles corrélations entre les images à l'encodeur), et un décodage inter-frame (en exploitant les corrélations temporelles au décodeur). En d'autres termes, l'estimation de mouvement n'est plus réalisée au codeur comme dans les schémas classiques de codage, mais au niveau du décodeur. Certains travaux récents explorent l'utilisation du principe de codage par descriptions multiples pour améliorer les performances des systèmes de codage vidéo distribué, notamment en considérant

l'utilisation des descriptions multiples dans le contexte du codage de source avec information adjacente. Ici, l'accent est mis sur la construction efficace de l'information adjacente, l'estimation de certaines images de la séquence, qui est sûrement l'aspect le plus important d'un codeur vidéo distribué.

Contributions et plan

Tout d'abord, la partie I traite de codage vidéo classique. Après un non-exhaustif état de l'art (chapitre 2), le chapitre 3 présente quelques améliorations d'un codeur vidéo basé ondelettes scalable. La principale contribution de cette partie concerne le codage des vecteurs mouvement, le compromis débit-distorsion entre l'information de mouvement et les sous-bandes d'ondelettes, et le schéma lifting. Enfin, l' approche de codage avec pertes des vecteurs mouvement est appliquée au fameux standard de codage vidéo H.264, afin d'introduire un nouveau mode efficace de codage (chapitre 4).

Deuxièmement, dans la partie II, la transmission de vidéos sur des canaux bruités est explorée. La principale technique étudiée dans cette partie est le codage par descriptions multiples, dont l'état de l'art est présenté dans le chapitre 5. Ici, un focus est fait sur le décodage optimal de la description central, après la transmission sur canaux bruités (chapitre 6). Le codage vidéo distribué est également étudié dans le chapitre 7, en particulier les liens entre le codage par descriptions multiples et le codage vidéo distribué, et la construction de l'information adjacente.

Codage vidéo basé ondelettes

L'objectif est ici d'améliorer certaines parties d'un codeur vidéo basé ondelettes compensé en mouvement, présenté brièvement dans la Section 3.1. Plus précisément, une approche de codage avec pertes des vecteurs mouvement est présentée dans la Section 3.2, afin de réduire le coût du mouvement. Cette approche consiste à introduire des pertes sur les vecteurs mouvement estimés avec une grande précision sous-pixellique, tandis qu'un critère débit-distorsion est optimisé. Cette approche en boucle ouverte permet d'améliorer les performances de codage, en particulier à bas débit. Bien évidemment, l'introduction de pertes sur le mouvement a un impact sur la séquence décodée.

Dans le but d'évaluer analytiquement cet impact, un modèle débit-distorsion entre l'information de mouvement et les sous-bandes d'ondelettes est réalisé (Section 3.3). Un modèle théorique de la distorsion entrée/sortie de l'erreur de codage du mouvement a d'abord été établi, et a ensuite été amélioré par l'introduction de l'erreur de quantification des coefficients d'ondelettes et par la généralisation du modèle à plusieurs niveaux de décomposition temporelle. Ce modèle est ensuite utilisé pour répartir de manière optimale

les ressources entre les débits des vecteurs mouvement et des coefficients d'ondelettes spatio-temporelles. A cet effet, une allocation de débit basée modèle a été réalisée.

Enfin, la Section 3.4 présente une méthode nouvelle et adaptée pour la mise en oeuvre du schéma lifting, appelée schéma lifting pondéré et adapté au mouvement. Les pas du schéma lifting sont adaptés au mouvement : la fonction d'échelle originale est échantillonnée à des points calculés par un critère basé sur la norme des vecteurs mouvement. Cette méthode permet d'éviter certains artefacts de la séquence décodée causés par d'éventuelles défaillances de l'estimation de mouvement.

Application à un codeur hybride : H.264

L'approche de codage avec pertes des vecteurs mouvement présentée dans la Section 3.2 est appliquée au codeur H.264 à la Section 4.1. La clé de ce nouveau mode est le codage avec pertes des vecteurs mouvement, obtenu par la quantification. En outre, ce codage avec pertes est effectué dans un système à boucle ouverte de sorte que, alors que le résiduel compensé en mouvement est calculé avec des vecteurs mouvement de haute précision, les vecteurs mouvement sont quantifiés avant d'être envoyés au décodeur. Le niveau de quantification des vecteurs mouvement est choisi au sens débit/distorsion de manière optimisée.

Plusieurs questions théoriques (Section 4.2) doivent être traitées, afin d'améliorer les performances générales, comme, en particulier, le codage des vecteurs mouvement quantifiés, et la sélection et le codage du pas de quantification.

Le nouveau mode a été d'abord testé sur la partition 16x16 d'H.264, puis sur la partition 8x8. Pour le mode 8x8, les macro-blocs sont divisés en quatre sous-blocs de taille 8x8. Les vecteurs mouvement peuvent être quantifiés avec différents pas de quantification, les plus petites dimensions de macro-blocs peuvent alors aboutir à une mauvaise prédiction. Potentiellement, l'énergie de la prédiction du vecteur pourrait devenir plus importante que celle du vecteur lui-même, de sorte que la prédiction pourrait ne plus être une stratégie satisfaisante. Des solutions possibles sont donc analysées à la Section 4.3 pour tenir compte de l'influence de la prédiction provenant de différents pas de quantification. Ce nouveau mode de codage permet d'améliorer les performances de la référence H.264.

Codage vidéo par descriptions multiples

Le contexte de codage conjoint source-canal présenté dans ce travail est un schéma équilibré de codage par descriptions multiples pour le codage vidéo basé ondelettes au fil de l'eau. Le schéma de codage par descriptions multiples considéré est basé sur la structure générale du codeur présenté dans

la Section 3.1, et effectue une transformée en ondelettes spatio-temporelle discrète compensée en mouvement des images de la vidéo. La redondance est introduite avant la quantification, les descriptions équilibrées sont produites grâce à une allocation de débit basée sur les caractéristiques du canal. La structure générale de ce schéma est présentée dans la Section 6.1.

Dans cette thèse, un focus est réalisé sur le décodage conjoint de deux descriptions reçues au décodeur avec du bruit (Section 6.2). Deux approches de décodage optimal ont été mises en oeuvre : la première essaie d'estimer les deux descriptions générées, à partir des données reçues en sortie du canal, tandis que la seconde se concentre sur une estimation directe de la source à partir des deux descriptions bruitées, sans essayer d'estimer les descriptions latérales. Les résultats expérimentaux pour les deux approches présentent une bonne robustesse du codeur aux erreurs canal.

Codage vidéo distribué

Dans le cadre du projet de recherche ESSOR en collaboration avec d'autres laboratoires français, le codage vidéo distribué a été exploré, et est brièvement présenté à la Section 7.1. La qualité de l'information latérale est améliorée en proposant une interpolation d'image efficace à la Section 7.2. En effet, dans les schémas de codage vidéo distribué, l'extraction de l'information de mouvement est effectuée pour construire une estimée, appelée information adjacente, de certaines images de la séquence. La qualité de l'information adjacente a un fort impact sur les performances de codage du système. Une nouvelle méthode d'interpolation est donc proposée, elle réalise une estimation de mouvement bi-directionnelle et utilise une compensation de mouvement pixélique en permettant l'utilisation de vecteurs mouvement superposés. Cette technique dépasse les meilleures solutions existantes.

Chapter 1

Introduction

The works presented in this thesis mainly concern video coding and transmission.

1.1 General framework

During these last thirty years, our civilization would have produced more information than during the five thousand years that preceded them. Besides, more than half of the information that is created now is already on a digital form. Advances realized in the fields of network, telecommunications and digital storage are not sufficient to assimilate this amount of data. Among all these data to store or to transmit, multimedia data have an increasing place. In particular in the fields of speech, image and video, digital techniques have definitely replace analogical ones. The data represented by digital video (high definition digital television, digital cinema, visioconference, internet or mobile communications) are particularly vertiginous. But the problem of storage is not the only problem linked with the explosion of digital video. Transmission over many different networks without losing the quality of data is a huge challenge. Today's networks achieve high transmission speeds and support data rates sufficient for video applications, even in mobile communications for example. These facts promise a whole new world of communications.

For all of these reasons, compression has become an indispensable step in most of the applications linked with digital video. The actual standards of video coding, such as MPEG-2 (used for DVD, digital television), MPEG-4 and H.264, have known an important industrial success. The latest norms MPEG-4 and H.264 (MPEG-4 / Part 10) and their evolutions really improve the trade-off between the rate and the quality of the compressed videos, and allow new functionalities, as scalability. Scalability in video coding consists in extracting, from a single compressed video file, several versions of this video, in function of the transmission and storage support. Unfortunately, the support for scalability is limited, because of a lack of flexibility and often of a degradation in performances. Furthermore, the actual standards

are not compatible with the wavelet-based image coding norm JPEG2000.

For these reasons, it was decided, in the first part of this thesis, to explore wavelet-based video coding, since it has recently shown to be efficient. The 3D wavelet transform (WT) approach has drawn a huge attention by researchers in the data compression field. They hoped it could reply the excellent performances its two-dimensional version achieved in still image coding. Moreover, the WT approach provides a full support for scalability, which seems to be one of the most important topics in the field of multimedia delivery research. Of course, some improvements are still possible, especially in what concerns the motion vectors coding.

Indeed, the quality of the coded videos would be improved if the motion was described in a more precise way. In some cases, the relative weight of the motion vectors in the bitstream could be too important, especially at low bit-rate. In this work, the motion vectors coding is improved and the rate-distortion trade-off between motion information and wavelet coefficients optimized, in order to increase the video coding efficiency.

Another possible improvement of the wavelet-based video coder concerns the motion compensation. The inclusion of motion compensation in the temporal WT (and in its lifted version) has been shown to improve the efficiency of the temporal subband decomposition. But, the influence of some badly estimated motion vectors on the motion-compensated WT can be minimized. A method for the computation of the lifting scheme, the most suitable implementation for temporal WT, is thus proposed. The lifting steps are closely adapted to the motion.

In the framework of an industrial contract with the French national Telecom operator, Orange labs, the previous method of quantization of the motion vectors is applied to the video coder H.264. Indeed, the trade-off between the allocation of coding resources to motion vectors or to transform coefficients has a major importance when it comes to efficient video coding techniques. Nevertheless, in video coding standards, there is not much flexibility at this end: generally it is only possible to indirectly choose how the bit-rate is shared between motion vectors and coefficients by selecting one among the several available coding modes for each macroblock. Therefore, it has been noted that when a sequence is encoded at low and very low bit-rates, a large quota of resources is allocated to MVs. This suggests that, in the framework of a H.264 coder, there could be room for improvement if some new coding mode with less costly motion information is introduced.

The second part of this work is dedicated to video transmission over noisy channels, which is a tough question in digital communications. The problem of efficient video transmission involves good compression rates and effectiveness in presence of channel failures. Joint source-channel (JSC) coding has received an increasing interest in the research community. In particular, multiple description coding (MDC) has already shown good re-

sults as error-resilient joint-source coding. Two main families of MD coding schemes exist, depending on whether the redundancy is introduced during or before the quantization. In the first class of approaches, the quantizers are designed to produce two redundant descriptions of the same signal. In the second class, the redundancy is introduced during signal transformation. The original signal may be reconstructed as soon as one of the descriptions is available. The availability of the second description allows to increase the quality of the reconstructed signal. MD coding has been recently applied to video coding.

The MDC scheme presented in this work is balanced and wavelet-based. It belongs to the second class of approaches: redundancy is introduced before quantization of the signal. The main goal of this study, realized in the framework of a research national project, "ANR Projet blanc" ESSOR, is the optimal decoding of the signal after transmission over noisy channels. Indeed, the challenge at the decoder side is to reconstruct a signal with a distortion that is as small as possible. Maximum *a posteriori* estimations of the original source from the knowledge of the side descriptions corrupted by transmission errors are provided. *A priori* information is represented by a model describing the distribution of the wavelet coefficients of the frames. Two different approaches have been implemented.

The last part of this work concerns a study on distributed video coding (DVC), realized in the framework of the ESSOR project. Distributed source coding has emerged as an enabling technology for sensor networks. It refers to the compression of correlated signals captured by different sensors which do not communicate between themselves. All the signals captured are compressed independently and transmitted to a central base station which has the capability to decode them jointly. Distributed source coding has been recently brought into practice in video coding, leading to DVC. Contrary to classical video coding schemes, DVC performs intra-frame encoding of correlated frames (without exploiting any correlation between frames at the encoder), and inter-frame decoding (by exploiting the temporal frame correlation at the decoder). In other words, motion estimation is not performed anymore at the encoder as classical video coding schemes do, but at the decoder. Some recent works explore the use of the MDC principle to improve the performances of DVC schemes, especially by considering multiple descriptions in the context of source coding with side information. Here, the main focus is on the efficient construction of the side information, the estimate of some frames of the sequence, which is nearly the most important part of a distributed video coder.

1.2 Contributions and outline

First, part I deals with classical video coding. After a non-exhaustive state-of-the-art (Chapter 2), Chapter 3 presents some improvements of a scalable wavelet-based video coder. The main contribution of this part concerns motion vectors coding, rate-distortion trade-off between motion information and wavelets subbands, and the lifting scheme. Finally, the lossy motion information coding approach is applied to the famous H.264 video standard, in order to introduce a new efficient coding mode (Chapter 4).

Secondly, in Part II, video transmission over noisy channels is explored. The main technique studied in this part is multiple description coding (MDC), whose state-of-the-art is presented in Chapter 5. Here, a focus is made on the optimal decoding of the descriptions after transmission over noisy channels (Chapter 6). Distributed video coding is also explored in Chapter 7, especially the links between MDC and DVC, and the construction of the side information.

1.2.1 Wavelet-based video coding

The aim here is to improve certain parts of a motion-compensated wavelet-based (MCWT) video coder, briefly presented in Section 3.1. More precisely, an approach of lossy coding of motion vectors is introduced in Section 3.2, in order to reduce the motion cost. This approach consists in introducing losses on motion vectors estimated with a high sub-pixel accuracy, while optimizing a rate-distortion criterion. This open-loop method allows to improve the coding performances, especially at low bit-rates. Obviously, the introduction of loss in the motion has an impact on the decoded sequence.

Thus, in order to evaluate this impact analytically, a distortion-rate model between motion information and wavelet subbands is performed (Section 3.3). A theoretical input/output distortion model of the motion coding error has first been established, and has then been improved by introducing the quantization error of the wavelet coefficients and by generalizing the model to several temporal decomposition levels. This model is then used to dispatch in an optimal way the binary resources between the motion vectors and the temporal wavelet coefficients. For that purpose, a model-based bit allocation process has been performed.

Finally, Section 3.4 presents a novel and adaptive method for the implementation of the lifting scheme, called motion-adapted weighted lifting scheme. The lifting steps are closely adapted to the motion: the original mother scaling function is sampled at sampling points which are computed using a criterion based on the norm of the motion vectors. This method allows to avoid some artefacts in the decoded sequence due to some failures of the motion estimation.

1.2.2 Application to a hybrid coder: H.264

The proposed approach of lossy coding of the motion vectors presented in Section 3.2 is applied to the H.264 coder in Section 4.1. The key tool of the new mode is the *lossy coding* of motion vectors, obtained via quantization. Moreover, lossy coding is performed in an open loop system so that, while the transformed motion-compensated residual is computed with a high-precision motion vector, the motion vector is quantized before being sent to the decoder. The amount of quantization for the motion vectors is chosen in a rate/distortion optimized way.

Several theoretical issues (Section 4.2) need to be dealt with, in order to achieve a relevant overall performance improvement, as, in particular, the encoding of quantized motion vectors, and the selection and encoding of the quantization step.

The new mode has been first tested on the 16x16 partition, and then on the 8x8 partition. For the 8x8 mode, the macro-blocks (MBs) are split in four sub-blocks 8x8. The motion vectors can be quantized with different quantization steps, but the smaller dimension of MBs can handle to a wrong prediction. Potentially, the energy of the vector prediction could become more significant than the original vector, so the prediction could not be a convenient strategy anymore. Some possible solutions are therefore analyzed at Section 4.3 in order to take into account the influence of prediction coming from different quantization steps. This new coding mode allows to improve the performances of the H.264 reference.

1.2.3 Multiple description video coding

The framework of joint source-channel (JSC) coding presented in this work is a balanced multiple description coding (MDC) scheme for scan-based wavelet transform video coding. The considered MDC scheme is based on the general structure of the coder presented in Section 3.1, and performs a motion-compensated spatio-temporal discrete wavelet transform of the video frames. Redundancy is introduced before quantization, and balanced descriptions are produced thanks to a bit allocation based on the characteristics of the channel. The general structure of this scheme is presented in Section 6.1.

In this thesis, a focus is made on the joint decoding of two descriptions received at decoder with noise (Section 6.2). Two approaches of optimal decoding have been implemented: the first one tries to estimate the two generated descriptions from the received channel outputs, whereas the second one focuses on the direct estimation of the source from the two noisy descriptions, without trying to estimate the side descriptions. Experimental results for both approaches present a good robustness to the channel errors.

1.2.4 Distributed video coding

In the framework of the research project ESSOR with other French laboratories, distributed video coding has been explored, and is briefly presented in Section 7.1. The quality of the side information is improved by proposing an efficient frame interpolation in Section 7.2. Indeed, in DVC schemes, the motion information extraction is performed to build an estimate, called side information (SI), of some frames of the sequence. The quality of this SI has a strong impact on the coding performance of the system. A novel interpolation method which performs bidirectional motion estimation and uses pixelwise motion compensation by allowing overlapped motion vectors is thus proposed. This technique surpasses the best existing solutions.

Part I

A little incursion in video coding

Chapter 2

A state-of-the-art in video coding

Compression is an almost mandatory step in storage and transmission of video, since, as simple computation can show, one hour of color video at CCIR 601 resolution (576 x 704 pixels per frame) requires about 110 GB for storing or 240 Mbps for real time transmission.

On the other hand, video is a highly redundant signal, as it is made up of still images (called frames) which are usually very similar to one another, and moreover are composed by homogeneous regions. The similarity among different frames is also known as temporal redundancy, while the homogeneity of single frames is called spatial redundancy. Virtually all video encoders perform their job by exploiting both kinds of redundancy and thus using a spatial analysis (or spatial compression) stage and a temporal analysis (or temporal compression) stage.

This first chapter aims to present the specificities of video coding. First, some important generalities in terms of video coding are presented. Then, to conclude this non-exhaustive state-of-the-art, wavelet-based video coding will be introduced.

2.1 What is video coding?

In the next section, the general characteristics of a video, and the scalability property, which is a very important feature, are described. Then, a brief presentation of the classical video coding norms, from H.261 to MPEG4/H.264 will be drawn.

2.1.1 Some generalities

What are the main video formats and the main characteristics of a video? What is scalability?

2.1.1.1 The main characteristics of a video

There are several ways to sample the different color components of a digital video, which take into account the fact that the human eye is more sensitive

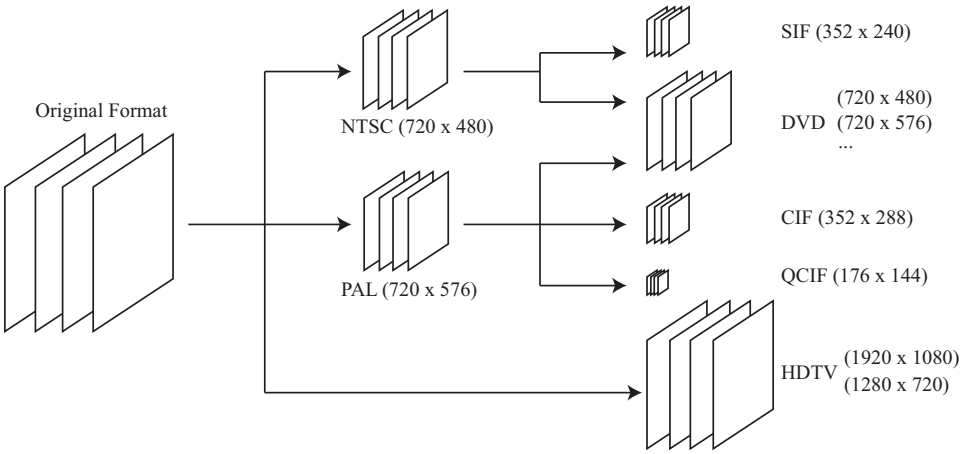


Figure 2.1: Different formats used in digital video.

to the luminance precision than to the chrominance precision. The $4:4:4$ sampling gives the same importance at all the color components, the $4:2:2$ mode only takes one chrominance sample for two pixels, and the $4:1:1$ and $4:2:0$ modes only take one sample for four pixels. The two last modes allow to obtain an image of sufficient quality for most of the applications. The $4:2:0$ mode is the most frequently used in image compression.

Figure 2.1 describes several formats of digital video. They are mainly characterized by the image dimensions (number of rows and columns) and the number of frames per second. The sampling of a analog video signal TV of type PAL (phase alternating line, 720×576 at 25 fps) or NTSC (National Television System Committee, 720×480 at 30 fps) allows to obtain a digital video at a non-compressed rate of 166 Mbps in $4:2:2$. By dividing by two the dimensions of the images along each axe, sequences at resolutions of 360×240 for the NTSC and 360×288 for the PAL are obtained. The color is then sub-sampled in $4:2:0$. The dimensions of the images have to be multiples of 16; by eliminating the four outside columns of each side of the images, the Standard Interchange Format (SIF, 352×240 at 30 fps) and the Common Intermediate Format (CIF, 352×288 at 25 fps), are obtained. They both encode non-interlaced videos at a non-compressed rate of $30,4$ Mbps. The CIF format at 30 fps ($36,5$ Mbps) and the QCIF format (quarter of CIF) are also used.

The available formats for the DVD are also derived of the PAL and NTSC. The possible resolutions go from 352×240 to 720×480 pixels for the signal derived of the NTSC, and from 352×288 to 720×576 pixels for those derived of the PAL. The 720×480 resolution is also called 525 SD (Standard Definition), and the 625 SD represents the 720×576 resolution.

For the High Definition Television (HDTV) and the digital cinema, three resolutions have been defined: the 1080i at 1920×1080 pixels, and the 720i

and 720p at 1280×720 pixels. The "i" signifies that the video is interlaced, by opposition to the "p" which signifies progressive. In the 1080i resolution, non-compressed rates of at least 1,5 Gbps are reached.

For the future, UHDTV (ultra-high definition TV) is already promising. The videos build for this format will have a maximal resolution of 7680×4320, *i.e.* 32 millions of pixels, with a frequency up to 120 fps and a sound on 22.2 channels. One minute of non-compressed UHDTV would need nearly 195 Go of storage, *i.e.* more than 40 DVD, or more than 4 Blu-ray discs double layer! With this course to high rates and resolutions, the democratization of nomad devices able to read videos increases. The offer of PDA, cellular phones and digital walk-man is very important, the price become accessible and the demand explodes.

This brief introduction shows two needs: to develop efficient video coders, allowing the transfer of high-rates contents on devices whose capacity evolves slowly, and to make these video coders fully scalable. Being able to extract of the same compressed file several different quality of the same video, in order to adapt a posteriori at the application, becomes today very interesting.

2.1.1.2 The dream feature: the Scalability

A scalable compressed bit-stream can be defined as one made up of multiple embedded subsets, each of them representing the original video sequence at a particular resolution, frame rate, quality, complexity or even in scene content. Moreover, each subset should be an efficient compression of the data it represents. Scalability is a very important feature in network delivering of multimedia (and of video in particular), as it allows to encode the video just once, while it can be decoded at different rates and quality parameters, according to the requirements of different users.

The scalable coding is often done by knowing, at the encoding, the different conditions of possible decoding. The bitstream is adaptive and could be decoded differently in function of the different configurations of the decoder. The main problem is often the definition of the basic layers. A basic layer, with a minimal description of the signal, will be transmitted in any case; the other layers allow to improve the quality or the resolution of the signal, with an increase of the decoded signal rate. According to the user needs (demanded specific rate, quality or resolution), the refinement layers should be defined in another way.

There are several forms of scalability. However, the main scalability functions are the rate scalability and the resolution scalability (spatial or temporal). The problems of the scalability in complexity or in scene content are less seen. Each type of scalability modifies a specific part of the coder, but the impact of the scalability on the coder global performances should be limited.

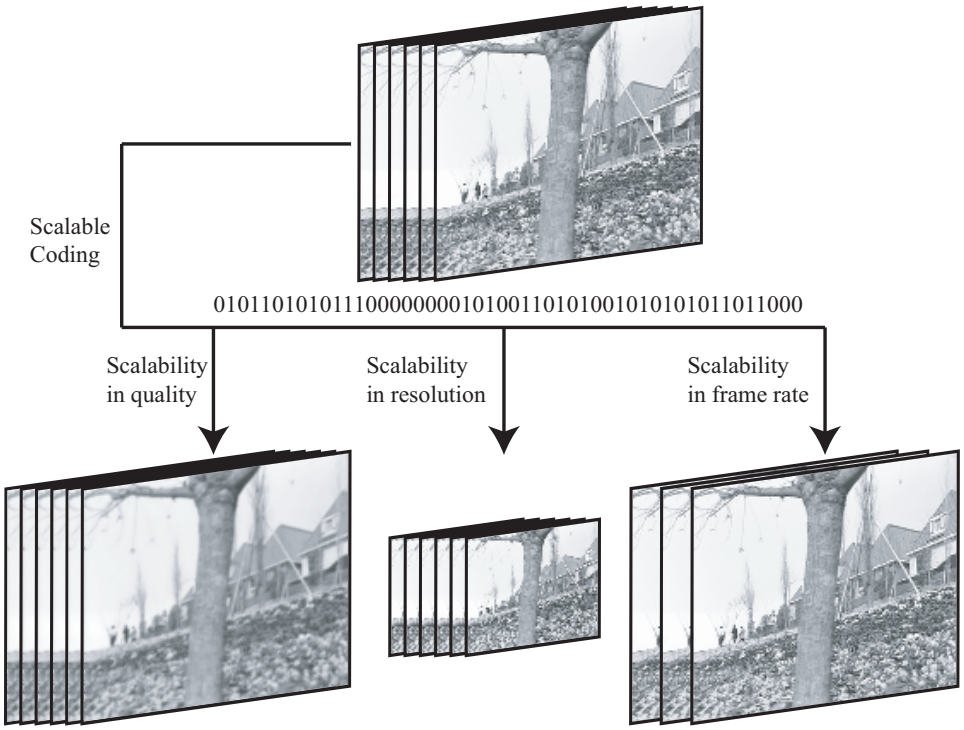


Figure 2.2: Scalability in quality, resolution, and frame rate.

Scalability in rate or in quality The rate scalability allows to choose the decoding rate of a compressed signal. The decoding of all the layers allows to obtain the compressed signal at maximal rate; if a certain number of layers are not decoded, the decoded rate decreases. The rate scalability is implemented in the coder at the step of quantization and coding. The EBCOT algorithm [Tau00] of bitplanes coding of images offers such as scalability.

This scalability must allow to directly choose the quality of the reconstructed signal, for example in terms of SNR, not in terms of rate.

Scalability in resolution (spatial or temporal) The user can desire to decode a video at a smaller size or at a smaller frame rate than the original one: this is the resolution scalability. Spatial scalability corresponds to when a video has to be decoded at a different spatial resolution than the original, and temporal scalability corresponds to when a video has to be decoded at a different frame rate. Different layers of resolution should be defined in order to know what pixels, or what images (in the case of the temporal scalability) should be decoded. The resolution scalability can be implemented by adapting the step of spatial or temporal transform when it is possible, or by adding a sub-sampling step. Most of the applications have chosen a spatial or temporal sub-sampling by 2^n.

Scalability in complexity The scalability in complexity consists in adapting the complexity of the decoding operations in function of the capacities and needs of the user. It can be linked with the other forms of scalability; for example, one of the possibilities for reducing the decoder complexity is to only decode one image over two (temporal scalability). However, the scalability in complexity may also cover other parts of the coder, as for example the quantization, the binary coding, or the motion estimation which remains the more expensive component in computations times in most of the video coders.

2.1.2 Hybrid coders

The most successful video compression schemes to date are those based on hybrid video coding. This definition refers to two different techniques used in order to exploit spatial redundancy and temporal redundancy.

2.1.2.1 The main principles

Spatial compression is indeed obtained by means of a transform based approach, which makes use of the discrete cosine transform (DCT), or its variations. Temporal compression is achieved by computing a motion-compensated prediction of the current frame and then encoding the corresponding prediction error. Of course, such an encoding scheme needs a motion estimation (ME) stage in order to find the motion information necessary for prediction.

The hybrid encoder works in three possible modes: intraframe, INTRA interframe, INTER and SKIP. In the intraframe mode, the current frame is encoded without any reference to other frames, so it can be decoded independently from the others. Intra-coded frames (also called anchor frames) have worse compression performances than inter-coded frames, as the latter benefits from motion-compensated prediction. Nevertheless they are very important as they assures random access, error propagation control and fast-forward decoding capabilities. The intra frames are usually encoded with a JPEG like algorithm, as they undergo DCT, quantization and variable length coding (VLC). The spatial transform stage concentrates signal energy in a few significative coefficients, which can be quantized differently according to their visual importance. The quantization step here is usually tuned in order to match the output bit-rate to the channel characteristics. The SKIP mode corresponds to a case where only the signalling information is sent, and the macroblock (MB) is reconstructed by copying the MB from the reference image at a position inferred from the motion vectors of the neighbors MBs.

In the interframe mode, current frame is predicted by motion compensation from previously encoded frames. Usually, motion-compensated prediction of current frame is generated by composing blocks taken at displaced

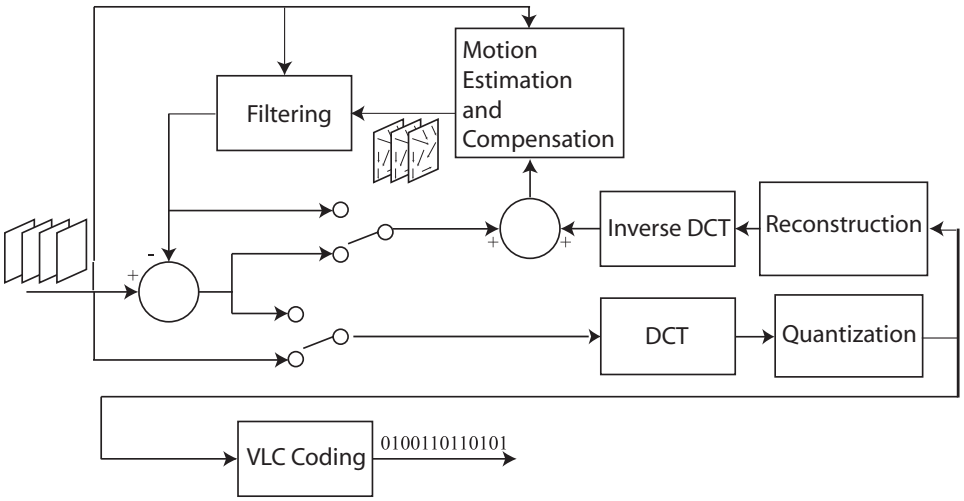


Figure 2.3: The H.261 coder.

positions in the reference frame(s). The position at which blocks should be considered is obtained by adding to the current position a displacement vector, also known as motion vector (MV). Once current frame prediction is obtained, the prediction error is computed, and it is encoded with the same scheme as intra frames, that is, it undergoes a spatial transform, quantization and entropy coding.

In order to obtain motion vectors, a motion estimation stage is needed. This stage has to find which vector better describes current block motion with respect to one (or several) reference frame(s). Motion vectors have to be encoded and transmitted as well. A VLC stage is used to this end.

All existing video coding standards (described in the following section) share this basic structure, except for some MPEG-4 features.

2.1.2.2 The main video standards

From H.261 to MPEG-2 Historically, H.261 was the first finalized norm of video coding, in 1990. Dedicated to visioconference, its allows to code the CIF and QCIF formats with a delay lower than 150 ms. Figure 2.3 shows the general scheme of the H.261 coder, which has inspired the one of all the hybrid coders until MPEG4. The motion compensation is optional, the motion estimation is done with 16x16 blocks, a forward error correction (FEC) code is included in the bitstream.

The MPEG-1 norm, finalized in 1992, is widely based on H.261, with some improvements. It was created to code videos at a rate of 1.5 Mbps, which corresponds to the lecture speed of a CD-ROM. The main evolution concerns the temporal prediction. The motion estimation is possible in half-pixelic precision. MPEG-1 works on groups of pictures (GOP) of flexible

size. It also improves the quantization, by taking into account the characteristics of the human eye, which is more sensitive to the effects of the quantization on the low frequencies. The complexity of the coding algorithm of MPEG-1 is of course higher than the one of H.261.

The works on MPEG-2, started in 1991, aimed to build a coder able to handle with rates higher than MPEG-1, from 4 to 15 Mbps, to obtain high quality videos to be exploited in on-demand video, digital television, TVHD, etc. MPEG-2 allows scalability in quality (measured by the signal-to-noise ratio), in spatial resolution, and in frequency. It is possible to combine at the maximum two scalability modes. In order to adapt at the different needs, and to avoid a unique coder with multiple possibilities but too heavy, some groups of characteristics, called profiles, are defined for some given needs. MPEG-2 can handle with all the input formats with a resolution till 16384x16384 pixels, and till 60 frames per seconds. But, the main evolution concerns the interlaced videos. In each step of coding, MPEG-2 proposes two modes: the "frame" mode where the images are treated as non-interlaced, and the "field" mode, where the specificities of the interlaced signal are taking into account. This mode allows to improve the final quality, mainly if the objects have a fast motion.

MPEG-4 and H.264 The MPEG-4 norm mainly introduces the notion of audio and video objects, in the goal of giving a little of interactivity to the users. Like the previous norms, MPEG-4 defines the syntax of the bitstream, and the structure of the decoder, and leaves free the implementation of the coder. The textures coding is done in a very similar way at in MPEG-2, with some improvements. For example, the basis version of MPEG-4 ASP (advanced simple profile) provides a quarter-pixelic accuracy for the motion estimation.

A new step of performances is nevertheless achieved with the AVC version. In 1998, the VCEG group (Video Coding Experts Group) of the ITU-T creates the H.26L project [Joi02] with the objective of increasing the coding efficiency compared to the existing norms. In 2001, VCEG and MPEG conjointly form the JVT (joint video team) and realize the MPEG-4 part.10 AVC norm (advanced video coding), also called H.264 [SWS03, SW05, MWS06]. This norm proposes several little improvements of MPEG-4 ASP, which increase a lot the coding gain. Of course, for a standard television video, H.264 is from 8 to 10 times more complex than MPEG-2, and the complexity of the decoder is also increased (4 times more complex).

First, the motion compensation can handle size of blocks varying between 16×16 and 4×4 pixels, square or rectangular, in order to be better adapted to the shape of the objects in the video. The intraframe prediction is also improved. It can be done in 8 directions. The DCT is replaced by

a separable integers transform on 4x4 blocks. During the quantization, 52 scalar non-uniform quantizers can be used. Finally, the bitstream is generated by an entropic coding. The UVLC algorithm (unified variable length coding) is used to code the headers, and the CAVLC algorithm (context adaptive variable length coding) or CABAC (context adaptive binary arithmetic coding) is used for the other parts of the bitstream.

The observed gain in performances between MPEG-4 ASP and H.264 is more than 2 dB in average. In other terms, for the same quality, the bit-rate is divided by two. Like MPEG-2, H.264 proposes several different profiles. Finally, a huge number of commercial codecs are based on MPEG-4, like DivX, XviD, WM9, QuickTime etc.

2.1.2.3 Scalability and hybrid coders

The simple scheme described in Section 2.1.2.1 does not integrate any scalability support. As shown by Table 2.1, norms and standards have been build to answer to very precise needs. However, the importance of scalability was gradually recognized in video coding standards. The earliest algorithms (as ITU H.261 norm [ITU99, Liu91]) did not provide scalability features, but as soon as MPEG-1 was released [ISO93], the standardization boards had already begun to address this issue. In fact, MPEG-1 scalability is very limited (it allows a sort of temporal scalability thanks to the subdivision in GOP. The following ISO standards, MPEG-2 and MPEG-4 [ISO00, ISO01, Sik97] increasingly recognized scalability importance, allowing more sophisticated features. MPEG-2 compressed bit-stream can be separated in subsets corresponding to multiple spatial resolutions and quantization precisions. This is achieved by introducing multiple motion compensation loops, which, on the other hand, involves a remarkable reduction in compression efficiency. For this reason, it is not convenient to use more than two or three scales.

Scalability issues were even more deeply addressed in MPEG-4, whose fine grain scalability (FGS) allows a large number of scales. This algorithm consists in defining only two quality layers, the base layer and the refinement layer, which is coded in a progressive way by bit-planes. It is possible to avoid further motion compensation loops, but this comes at the cost of a drift phenomenon in motion compensation at different scales. In any case, introducing scalability affects significantly performances. The fundamental reason is the predictive motion compensation loop, which is based on the assumption that at any moment the decoder is completely aware of all information already encoded. This means that for each embedded subset to be consistently decodable, multiple motion compensation loops must be employed, and they inherently degrade performances. An alternative approach (always within a hybrid scheme) could provide the possibility, for the local decoding loop at the encoder side, to lose synchronization with the actual decoder at certain scales; otherwise, the enhancement information at certain

Norm	typical rate	Applications
H.261	64 Kbps	Visioconference (ISDN)
MPEG-1	1,5 Mbps	Video on demand (Internet) Storage and lecture (CD-ROM) Visioconference (WAN)
MPEG-2 / H.262	1,5 Mbps 1,5 - 9,72 Mbps 10 - 20 Mbps	Storage and lecture (CD-ROM) Storage and lecture (DVD) TVHD
H.263	64 Kbps 1,5 Mbps	Visioconference (ISDN) Visioconference (WAN)
MPEG-4 & H.264	64 Kbps 56 kbps - 1 Mbps 1 Mbps 6 Mbps	Visioconference (ISDN) Video on demand (Internet) Storage and lecture (CD-ROM) TVHD

Table 2.1: Main video coding norms and their applications.

scales should ignore motion redundancy. However, both solutions degrade performances at those scales. Indeed, it is one of the main drawback of the scalable hybrid coders: some works like the ones of Li [Li01] have shown lower performances, more than 2 dB, compared to the ones of the no-scalable versions of these coders.

The SVC extension of H.264 [OSWW08] is also based on a coding by refinement layers, but it is able to define more than two quality layers. Thus, it offers a good support for the quality scalability, and the spatial and temporal scalabilities are also proposed. Furthermore, it has the advantage of being compliant with H.264 for the low layer, and to achieve almost the same performances.

The structure of the hybrid coders is very efficient, the H.264 norm is now finalized and its parameters are again and again refined. However, this structure does not exactly respond to the actual needs in terms of scalability. Even if some modifications are currently going in this way, it is really possible that an alternative structure, wavelet-based for example, could one day achieve equivalent performances while allowing a better support to the scalability. The actual research works on wavelet-based video coding aim to this goal.

2.2 Wavelets and video coding

Wavelets have been successfully used in image coding till the nineties [ABMD92]. The wavelet-based JPEG2000 coder is fully scalable and very efficient in respect to the JPEG standard based on the DCT (however, its diffusion has

been slowed because of problems of normalization and complexity). In the spatial domain, the wavelets transform aims to replace the DCT of the hybrid coders.

After a brief presentation of the most important wavelets used in compression, the main parts of a wavelet-based video coder are presented: the temporal filtering and the motion compensation, and the coding of the wavelets coefficients and the motion vectors.

2.2.1 Main wavelet transforms used in compression

Among all the wavelet transforms used in image processing [CSZ98, Mal99, UB03], most of the image and video coders only use some of the well-adapted ones. In the case of a spatial transform, long filters can be used in order to obtain a good decorrelation. For the temporal transform, shorter filters are necessary, mainly because of the presence of the motion that have to be compensated; but a too short filter will obtain worst performances. In fact, coders use mainly the following wavelet transforms:

- The Haar wavelet [Haa10], whose support is limited to two samples, is the only wavelet allowing to build an orthogonal transform, symmetric and with a finite impulsionnal response. It allows to simplify several problems in wavelet-based video coding, like the motion compensation. But it has only one null moment.

- The 5/3 wavelets have a support of 5 samples at the analysis (3 samples at the synthesis) and two null moments; however, their use is quite simple. They are bi-orthogonal symmetric wavelets very used for the motion-compensated temporal filtering.

- The 9/7 wavelets have a wider support of 9 samples for the analysis and 7 for the synthesis, and four null moment, which improves their decorrelation ability. They are bi-orthogonal, even nearly orthogonal. The 9/7 transform, very efficient, is used in several image coding schemes, as JPEG2000. Antonini et al. [ABMD92] have shown their superiority for the coding of natural images.

A more precise description of these wavelets can be found in [Mal99, Dau92] for example.

In the following, the necessary adaptations for wavelet transforms used for a temporal analysis are presented.

2.2.2 Temporal filtering, lifting schemes and motion compensation

The first application of the wavelet transform to video coding has been proposed by Karlsson and Vetterli [KV88]. The coder is scalable and presents

good performances, but suffers from the lack of motion compensation. Despite of the improvements bring by the 9/7 transform [ABMD92], by the 3D coding based on SPIHT of Kim and Pearlman [KP97], or by the motion-compensated coder of Choi and Woods [CW99a], the performances of these coders remain smaller than those of the hybrid coders in presence of motion.

2.2.2.1 The lifting scheme

Implementing a wavelet transform by a lifting scheme consists in replacing the filter banks by some operators of prediction P and update U, also called lifting steps. These operations occur after the decomposition of the signal into several components, generally two. The first component will thus be predicted thanks to the second, which will then be updated in function of the prediction error on the first component.

The polyphase transform defines several components for the signal. Generally, she consists in separating the even samples from the odd ones, but the selection criterion and the number of components can vary. The operators of prediction P must allow to obtain an approximation of a sample in function of its neighbors. This prediction allows to minimize a criterion of distortion between the sample and its prediction (for example the mean square error), but here again, this criterion can change. The operators of update U are more difficult to define by objectives properties; it can be for example desired that the update result minimizes the spectral aliasing.

Daubechies showed [DS98] that each wavelet transform implemented by a filter bank with finite impulsionnal responses can also be implemented by an equivalent lifting scheme with a finite number of decomposition levels. However, the reciprocity is not true.

For example, the implementation of the 5/3 filtering by lifting scheme is done only thanks to two lifting steps. In this case, the high- and low-frequency subbands for the first resolution level are computed by:

$$h_k(\mathbf{p}) = x_{2k+1}(\mathbf{p}) - \frac{1}{2}\left(x_{2k}(\mathbf{p}) + x_{2k+2}(\mathbf{p})\right), \tag{2.1}$$

$$l_k(\mathbf{p}) = x_{2k}(\mathbf{p}) + \frac{1}{4}\left(h_{k-1}(\mathbf{p}) + h_k(\mathbf{p})\right), \tag{2.2}$$

for each position $\mathbf{p}$ of the pixels of the image, and for each $k \in \left[2, \frac{K}{2} - 1\right]$, with K the number of frames of the sequence. $\mathbf{p}$ can be defined by the couple (p_1, p_2), with p_1 the number of the row, and p_2 the number of the column which correspond to the considered pixel. (2,2) also indicates the length of the prediction and update operators [CDSY98]. A temporal (2,2) lifting scheme is presented at Figure 2.4.

The obtained subbands $\{h_k\}$ and $\{l_k\}$ are the same than for the 5/3 filter, but the lifting scheme presents several advantages to implement a wavelet transform:

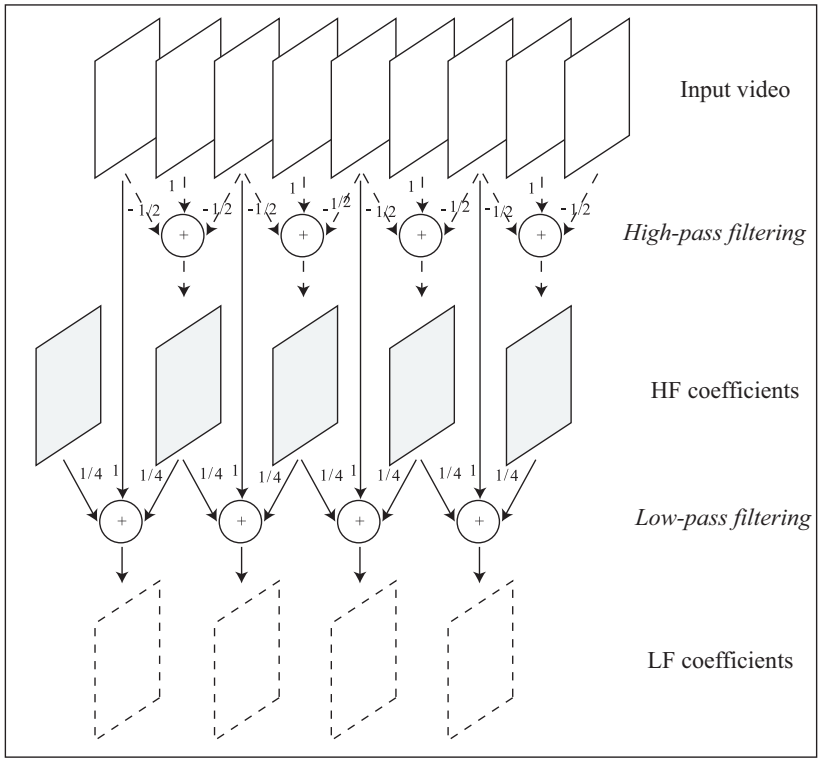


Figure 2.4: Temporal wavelet transform by (2,2) lifting scheme.

- it is more efficient in terms of memory occupation, because the calculations can be done in place. This is very interesting for the temporal filtering of a video sequence
- it is also less complex, because it needs always fewer operations than its equivalent transverse filter
- it is very easily reversible and it does not need the computation of a particular synthesis filter
- it allows to build the transforms and their inverses very easily, in function of the desired properties, without the need of the Fourier transform. For this reason, these are wavelets of second generation, by opposition to the wavelets of first generation which are issued of translations and dilatations of mother wavelets.

Nevertheless, the more important advantage of the lifting scheme for the temporal filtering of a video sequence is the possibility of modifying the prediction and update operators, even by non-linear operations such as motion compensation, without losing the reversibility property.

In image coding, the generalized lifting has been recently introduced

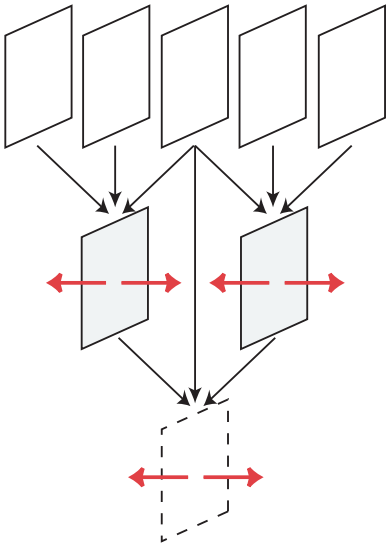


Figure 2.5: Temporal wavelet transform by (2,2) motion-compensated lifting scheme: in red, the *backward* and *forward* motion vectors.

as an extension of the classical lifting scheme to introduce more flexibility and to permit the creation of new nonlinear and adaptive transforms [BTS01, HPPP06a, HPPP06b, SS07, RS07, RSA08].

2.2.2.2 Motion compensation

The motion-compensated lifting scheme, whose first applications were presented by Pesquet-Popescu and Bottreau [PPB01], and by Secker and Taubman [ST03], are for the most based on the (2,2) lifting scheme, as seen in Figure 2.5. These coders obtain better performances than the hybrid coders till MPEG2 while being scalable.

The lifting scheme equations (2.1) and (2.2) can easily be modified to take into account the motion. $\mathbf{v}_{i+j\to i}(\mathbf{p})$ denotes a vector describing the motion of the pixel $\mathbf{p}$ of image x_{i+j} through the pixel $\mathbf{p}+\mathbf{v}_{i+j\to i}(\mathbf{p})$ of image x_i. The estimation of this vector is often based on the hypothesis that the luminance is stationary within the trajectory of each object, that is to say $x_i(\mathbf{p}+\mathbf{v}_{i+j\to i}(\mathbf{p})) \approx x_{i+j}(\mathbf{p})$.

The motion vector $\mathbf{v}_{i+j\to i}(\mathbf{p})$ will be called *backward* if j is negative, or *forward* if j is positive. Then $x_i(\mathbf{p}+\mathbf{v}_{i+j\to i}(\mathbf{p}))$ is the motion-compensated image x_i with respect to image x_{i+j}.

The equations of the (2,2) motion-compensated lifting scheme on one

resolution level are thus:

$$h_k(\mathbf{p}) = x_{2k+1}(\mathbf{p}) - \frac{1}{2}\big(x_{2k}(\mathbf{p} + \mathbf{v}_{2k+1\to 2k}(\mathbf{p})) + x_{2k+2}(\mathbf{p} + \mathbf{v}_{2k+1\to 2k+2}(\mathbf{p}))\big), \quad (2.3)$$

$$l_k(\mathbf{p}) = x_{2k}(\mathbf{p}) + \frac{1}{4}\big(h_{k-1}(\mathbf{p} + \mathbf{v}_{2k\to 2k-1}(\mathbf{p})) + h_k(\mathbf{p} + \mathbf{v}_{2k\to 2k+1}(\mathbf{p}))\big), \quad (2.4)$$

for each pixel $\mathbf{p}$, and for each $k \in \big[2, \frac{K}{2} - 1\big]$.

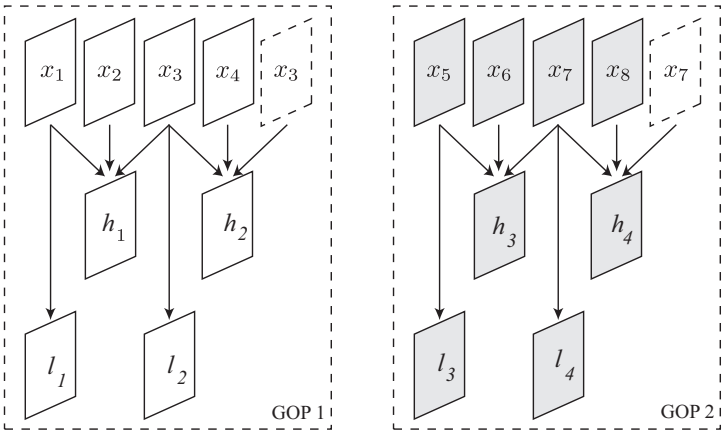
2.2.2.3 Scan-based filtering

The temporal analysis of H.264 can be done independently on the successive images of the original video, on the contrary, the wavelets filtering is a continuous process. To compute the wavelet coefficients for a given pixel, the neighbor pixels are necessary. It is impossible to filter a complete spatio-temporal signal along the temporal axis without dividing it in several parts. But, the temporal wavelet transform must not be computed independently on groups of consecutive images seen as temporal blocks (GOP). Indeed, this filtering by blocks has some bad consequences: the most important is that each group of wavelet coefficients would suffer from edge effects which would decrease the transform efficiency and thus the coder performances. The solution consists in computing the transform continuously, thanks to a scan-based filtering. Figure 2.6 compares the filtering by blocks to the scan-based filtering, for one level of decomposition, on the 8 first images of a sequence x_k. The block transform needs to obtain by symmetrization the images in dotted lines, which changes the nature of the transform. These necessary symmetrizations lead to a decrease in PSNR. Visually, it leads to flutter effects, or even to luminosity variations, which could be very bad, in particular at low rate. The scan-based transform directly compute the correct transform without symmetrization, thus the PSNR is unchanged compared to when the transform is computed over the whole sequence.

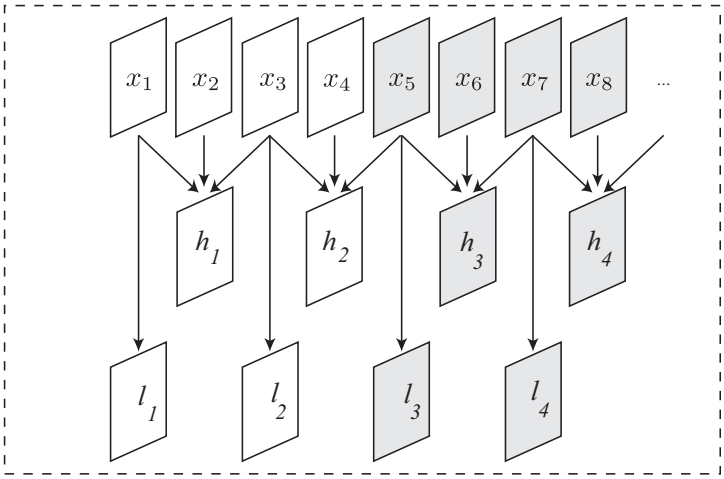
The scan-based wavelet transform was first introduced in 1994 by Vishwanath [Vis94] for the monodimensional filtering. It has then be adapted to the 2D filtering by Charbonnier *et al.* [CAB95], by Parisot *et al.* [Par03] and by Chakrabarti et al. [CVO95], as an alternative to the transform by blocks. The technique has also been successfully used in the framework of the wavelet transform by Chrysafis and Ortega [CO98].

2.2.3 Wavelet subbands coding

The wavelet subbands coding was the object of many research works. The first embedded zero-tree coder , allowing an efficient scalable coding of the



(a) Temporal blocks filtering (GOP).



(b) Scan-based temporal filtering.

Figure 2.6: Temporal analysis: comparison of the block filtering and the scan-based filtering (this example is for the (2,0) lifting scheme which will be presented in Section 3.3.1.2).

wavelet coefficients, is EZW [Sha93]. It performs a progressive coding in two layers by bit-planes; the truncation of the bits of the lowest weight is done by taking into account their impact in all the subbands. EZW offers scalability in resolution and bit-rate, and a better objective quality compared to JPEG.

The zero-tree coding algorithm has later been improved by SPIHT [SP96], which mainly benefits from a better modelization of the importance of the coefficients before truncation, and then by EZBC [HW00], which includes a contextual arithmetic coder. The performances of EZBC coder are excellent, in terms of subjective and objective quality, and in terms of scalability.

Finally, it is the EBCOT algorithm [Tau00] which has been integrated

in the standard of still images coding JPEG2000 [TM02]. The wavelet coefficients are grouped in independent codeblocks coded by bit-planes. An optimization algorithm of the rate-distortion trade-off allows to define optimal truncation points, which give, for different target bit-rates, the codeblocks that have to be kept or removed. EBCOT reaches performances which can be compared to EZBC by bringing an important flexibility in bit-rate scalability.

2.2.4 Motion vectors coding

In what concerns the motion vectors, they are coded without loss and in a non-scalable way, as in the coder proposed by Ohm or in MC-EZBC. The motion vectors adapted to an inferior spatial resolution are simply divided to correspond to the scale, for example by two for a level of scalability. The main drawback of this simplified approach is that the relative weight of the motion vectors in the bitstream is too important at low bit-rate, and, in some cases, not sufficient at high bit-rate [VGP02]. In other terms, the quality of the high-rate coded videos should be improved if the motion was described in a more precise way. The accuracy of the motion description should be chosen a priori. But, in an ideal scheme, the motion has to be coded in a scalable way, so that its accuracy corresponds to the spatial resolution and to the quality of the decoded video.

Several research works have studied the problem and different solutions have been proposed. The authors of [JO97] consider adaptive context modeling techniques to losslessly code the motion information. They study various forward/backward context selection approaches, and show that forward adaptation can result in performances improvements. In [BBFPP01], Bottreau et al. estimate the motion by an hierarchical research on two resolution levels, and transpose this hierarchy during the coding: the difference between the high-resolution motion and the low-resolution motion is coded in a progressive way. In [BFG04], Boisson et al. propose a hierarchical description of the motion and a bit-planes coding with as much truncation points as available resolutions at the decoder. In [ST04], Secker and Taubman propose a motion coding by layers after a spatial wavelet transform, the selection of the layers function of the bit-rate is done at the decoder side. The Wavelet project of the MPEG group had chosen a coding by spatial slices [MPE05].

In most of the classical coders, such as the standard H.264 [SWS03, STL04], the motion vectors are predicted spatially, then they are coded losslessly with an entropy coder. The trade-off between vector accuracy and vector size is also optimized in order to control the quality of the reconstructed video at a given bit-rate. The motion bit-rate is adjusted while modifying the parameters of the motion estimation: the size of the blocks (also as in [MDN93]) and the accuracy of the estimator are adapted to reach

the target bit-rate.

Other approaches [LW95, CW00, RR97] propose to simultaneously estimate and vector quantize the motion vectors by reinterpreting the "block-matching" algorithm as a kind of vector quantization, with different layers, and by using an entropy-constrained quantization [JFB95]. In [XXW+04], to make the best trade-off between motion and texture, the authors use a motion layer decision algorithm. Some approaches [MSAI04] use a precision limited coding (PLC) for scalable motion vectors (vectors estimated at the highest resolution and then scaled down at the decoder). In [BMV+05], the authors present a quality-scalable motion vector coding algorithm using median-based motion vector prediction, and an heuristic technique for global rate-allocation. In a wavelet-based $2D + t$ video coder, the authors of [TMTS06] encode the nodes of the quadtree, resulting from the variable size block matching, from top to bottom starting from the most significant bitplane. But some of these approaches remain very complex and their implementations could be prohibitive for some applications.

In the following chapter, some improvements of a wavelet-based motion-compensated video coder are presented. In particular, a novel approach for the coding of motion vectors has been developed. This approach will allow to control the rate-distortion trade-off between motion vectors and wavelet subbands. A motion-adapted weighted lifting scheme is also presented.

Chapter 3

Some improvements of a wavelet-based video coder

In the general framework of a wavelet-based video coder, an approach to quantize the motion vectors using a scalable and open-loop lossy coder has been developed. A theoretical distortion model of the motion coding error has then been established in order to evaluate the impact of this lossy motion coding on the decoded sequence. Thanks to the proposed theoretical distortion model, including also the subbands quantization noise, a model-based bit-rate allocation has been developed between motion vectors and wavelet subbands. Finally, an improvement of the lifting scheme by closely adapting the lifting steps to the motion is proposed.

3.1 General structure of the coder

Fully scalable, the considered video encoder is based on a lifted motion-compensated wavelet transform. Most of this coder has been developed in the thesis works of T. André [And07] and M. Cagnazzo [Cag04]. Its general structure is described in the Figure 3.1. Its main parts are described in the following sections.

3.1.1 Temporal analysis

The general scheme of the temporal analysis stage is shown in Figure 3.2. The input sequence undergoes motion estimation, in order to find the motion vectors. These latter are needed in order to perform a motion compensated wavelet transform (MCWT). Motion vectors are finally encoded and transmitted to the decoder, while temporal subbands feed the spatial analysis stage.

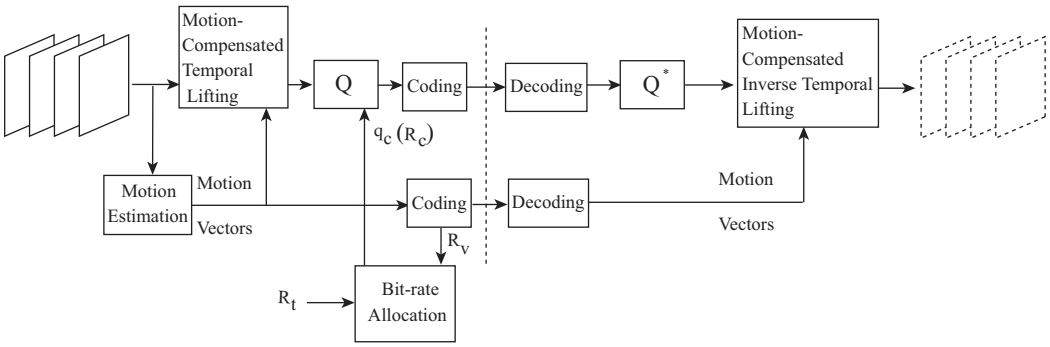


Figure 3.1: General structure of the encoder.

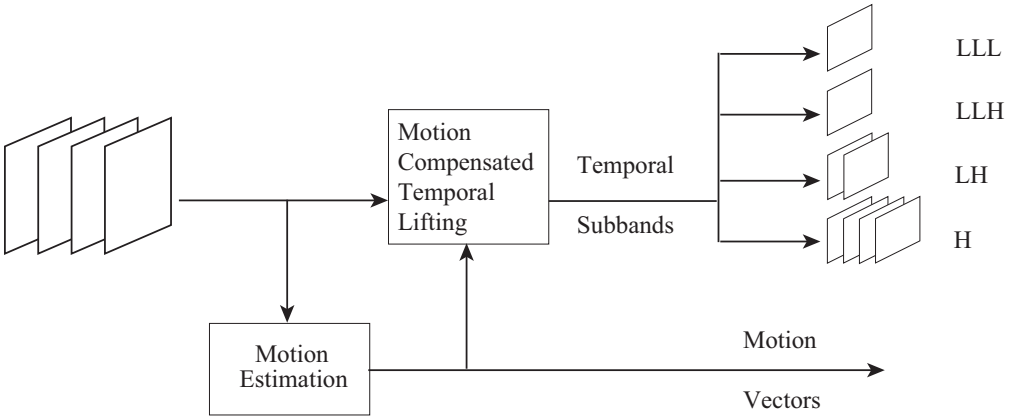


Figure 3.2: General scheme of the motion-compensated temporal analysis.

3.1.1.1 Temporal filtering

Since a video sequence can be seen as a three-dimensional set of data, the temporal transform is just a filtering of this data along the temporal dimension, in order to take advantage of the similarities between consecutive frames. This filtering is adapted to the object movements using motion compensation.

This is possible by performing the time-filtering not in the same position for all the considered frame, but by "following the pixel" in its motion. In order to do this, a suitable set of motion vectors is needed. Indeed, a set of vectors is needed for each temporal decomposition level.

A new class of filters, the so-called (N, 0) [Kon04, ACA+04], has been implemented and studied for this kind of application. This filters are characterized by the fact that the Low-Pass filter actually does not perform any filtering at all. This means, among other things, that the lowest frequency subband is just a subsampled version of the input video sequence. This has remarkable consequences as far as scalability is concerned. The temporal

analysis is thus performed by a (2,0) lifting scheme [LLL+01, ACA+04], which is obtained from the (2,2) lifting scheme by suppressing the update step.

The computation of the wavelet transform is scan-based [CO00]. This method avoids visual artefacts due to the processing of the sequence by GOP (in general a GOP is composed by 8 or 16 frames), which allows not to store in memory the complete video sequence.

3.1.1.2 Motion estimation

Motion estimation (ME) is a very important step in any video encoder. The motion estimation stage has to provide the motion vectors needed by the motion compensation stage, which, in the case of hybrid coders is the prediction stage, while in the case of WT coders is the motion compensated temporal filtering.

Many issues have to be considered when designing the ME stage. First of all, a model has to be chosen for the motion. The simplest is a block-based model, in which frames are divided into blocks. Each block of the current frame (*i.e.* the one which is being analyzed for ME) is assumed to be a rigid translation of another block belonging to a reference frame. The ME algorithm has to find which is the most similar block of the reference frame. In the considered encoder, this simple block-based approach is used, in which motion is described by two parameters, which are the two components of the 2D vector defining the rigid translation. Both Backward and Forward vectors can be computed (see Section 2.2.2.2), and a sub-pixelic interpolation can be used to obtain more accurate motion vectors. Some classical motion estimation algorithms have been implemented, as the exhaustive search, the three-steps search and the diamond search.

With respect to the chosen motion model, the ME stage has to find a set of motion parameters (e.g. motion vectors) which minimizes some criterion, as the mean square error (MSE) between current frame and motion compensated reference frame. The MSE criterion is the most widely used but is not necessarily the best possible. Indeed, a compromise between accuracy and coding cost of motion vectors should be considered. In the considered coder, a sum of squared difference (SSD) criterion based on both luminance and chrominance informations can be used [And07].

3.1.1.3 Motion information encoding

Once ME has been performed, the motion information has to be encoded. Lossless encoding is first mainly considered, so that the encoder and decoder use the same vectors, and perfect reconstruction is possible if no lossy operation is performed in the spatial stage.

Here the main problem is how to exploit the high redundancy of motion vectors. Indeed, motion vectors are characterized by spatial correlation,

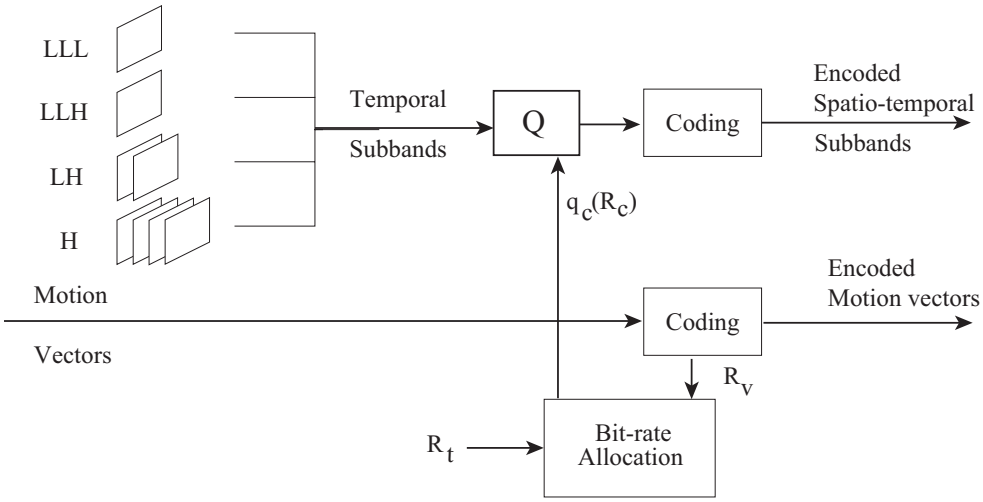


Figure 3.3: Spatial analysis: processing of the temporal subbands produced by a dyadic 3-levels temporal decomposition.

temporal correlation, and, in the case of WT video coding, the correlation among the vectors belonging to different decomposition levels. In the considered coder, the motion vectors are encoded by JPEG2000.

An important contribution of this thesis is the introduction of lossy coding of the motion vectors, in order to reduce their cost and to optimize the trade-off between the bit-rate of the wavelet subbands and the one of the motion information. This work is detailed in Section 3.2.

3.1.2 Spatial analysis

The temporal analysis stage outputs several temporal subbands: generally speaking, the lowest frequency subband can be seen as a coarse version of the input video sequence. Indeed, as long as (N, 0) filters are used, the low frequency subband is a temporally subsampled version of the input sequence (see Section 3.3.1.2). On the other hand, higher frequency subbands can be seen as variations and details which have not been catched by the motion compensation. The general scheme of the spatial analysis stage is represented in Figure 3.3.

3.1.2.1 Spatial Filtering and Encoding

The temporal subbands are processed in the spatial analysis stage, which performs a 2D DWT, producing the motion-compensated 3D WT coefficients which are then quantized and encoded. The encoding algorithm should allow good compression performances and scalability. To this end, the most natural choice appears to be JPEG2000. Indeed, this standard

provides a state-of-the-art compression and an excellent support to scalability.

Moreover, the proposed architecture allows to even provide a low frame-rate version of the original input sequence, without any further computation. This interesting result is due to the peculiar family of filters which are used for the temporal stage.

3.1.2.2 Resource Allocation

A suitable algorithm must be used in order to allocate the coding resources among the subbands. The problem is how to choose the bit-rate for each subband in order to get the best overall quality for a given total rate. This problem is addressed by a theoretic approach in order to find the optimal allocation. Moreover, a model is used in order to catch rate-distortion characteristics of subbands without a huge computational effort. Therefore this model allows to find optimal rates for subbands with a low computational cost.

More precisely, the bit-rate allocation between the wavelet subbands uses an optimal algorithm which requires the knowledge of the rate-distortion curves of each subband. A model-based approach permits to compute these curves in a precise and efficient way [CAAB04].

But a bit-rate allocation between wavelet subbands and motion information has also to be considered.

3.2 Lossy coding of motion vectors

As said previously, in a motion-compensated video coder, the bit-rate of the motion information can become proportionally too much significant compared to the one of the wavelet subbands, especially at low and very low bit-rates. The goal is thus to obtain the best motion vectors in terms of SSD, and then to find the best representation of these vectors in terms of a rate-distortion estimation. The main idea of this work is to encode the given motion vectors with losses, while keeping a good trade-off between the bit-rate and the distortion of the reconstructed sequence. The proposed approach is presented in what follows.

3.2.1 Problem statement

The wavelet transform allows an efficient decorrelation of the video sequences [Ohm94]. However, if the motion of the objects and/or the motion of the video camera is not taken into account, the decorrelation follows strictly the time axis. Consequently, points of successive images will be associated for the computation of the temporal transform whereas they did not share anything besides their coordinates in the image. This involves the

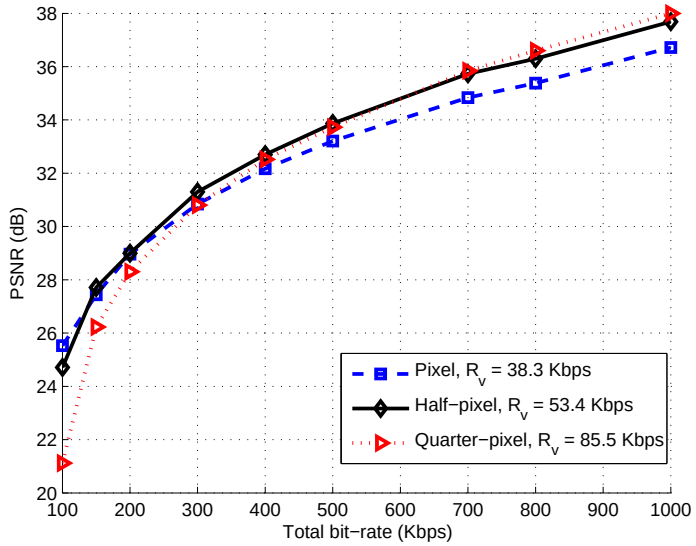


Figure 3.4: PSNR *vs* bit-rate curves with motion vectors coded losslessly to their entropy, on the sequence ERIC. Three temporal decomposition levels are computed using a lifting (2,0) and a "block matching" with diamond search algorithm. Several vectors accuracies are considered (R_v represents the motion vectors bit-rate when a lossless coding is applied). The PSNR values are computed only on the luminance component.

sub-optimality of the decorrelation, but also visual artefacts in case of a reconstruction of the sequence after a lossy coding.

Motion compensation is thus essential for an efficient decorrelation of the video sequences [Ohm94]. The motion of the sequence have to be estimated in the form of a dense vectors field. Nevertheless, a vectors field of good accuracy will necessarily be expensive to code compared to the wavelet coefficients. Moreover, wavelet-based coders apply generally the temporal transform on several decomposition levels, which increases the quantity of motion information to transmit.

Therefore, the two important questions are: how to represent and code the motion vectors? How to optimize the bit-rate trade-off between wavelet subbands and motion information?

The cost of the motion vectors depends on several parameters: the image resolution, the vectors accuracy (pixel, half-pixel, quarter pixel accuracy [Gir93]), and the size of blocks used for a "block matching" estimator [SB91]. This cost can be very significant, which is not desirable, especially at low bit-rate. Indeed, the part of the motion vectors in the total bit-rate can become too high compared with that of the temporal wavelet coefficients, and the coding could be prohibitive at low bit-rate.

This fact is illustrated on Figure 3.4, which represents PSNR (Peak Signal-Noise Ratio)/rate curves for the CIF-resolution (352x288 pixels, 30 fps) ERIC sequence and for several accuracies of motion vectors. Here, the vectors are estimated by a "block matching" method with diamond search algorithm and encoded without loss. This figure shows that, for low bit-rates, the performances of the pixelic estimator are better than those obtained with a higher accuracy (half or quarter pixel). The same kind of observations has been done on other sequences. These results highlight clearly that the representation of motion vectors is sub-optimal and involves an important cost for motion estimators of high accuracy. The challenge is thus to reduce the cost of the motion vectors, in particular for low bit-rate compression applications. Several approaches are possible and the most important are referenced in the state-of-the-art (Section 2.2.4). A method to encode the motion vectors in a lossy way is proposed, and a rate allocation based on a theoretical model. The goals are twofold: reduce the cost of the vectors, and improve the quality of the reconstructed video sequence.

3.2.2 Open loop coding of the motion vectors

The objective here is to code the motion vectors in an efficient and scalable way, while preserving the quality of the motion-compensated temporal filtering. The precision of the motion vectors is important for an efficient motion compensation and a good inter-frame decorrelation. If the motion field has a high precision, the high frequency temporal subbands will have small entropy and energy, allowing a good coding gain [CAAB04].

Therefore, in order to preserve good properties for the high frequency subbands, full precision motion vectors must be used at the encoder side for the motion compensation and the computation of the wavelet transform. Then, the quantization of the motion vectors is performed after the motion compensation. In that case, as illustrated in the Figure 3.5, the encoder is open-loop (and not closed-loop, see Figure 3.6). Then the direct and inverse temporal transforms will not use the same vectors and the perfect reconstruction of the sequence is not possible any more[1]. However, since this application deals with low and very low bit-rates, the exact reconstruction property is not required [ABMD92]. The open loop coding also allows to control the quantization noise of the wavelet subbands, and the one of the motion vectors, which is not the case in the lossless case or in a closed loop scheme.

3.2.3 Proposed motion coder

The quantization of motion vectors is detailed in Figure 3.7. Motion information is quantized with losses by a uniform scalar quantizer applied on

[1] Note that if the motion vectors are coded losslessly, the classical MCWT is retrieved.

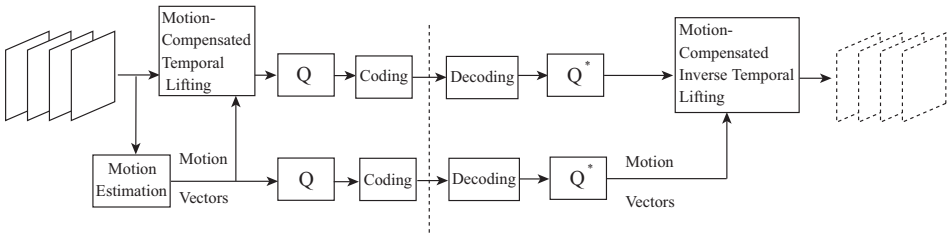


Figure 3.5: Open loop coding of motion vectors in a video coder: the wavelet subbands and the vectors are scalable, motion bit-rate can thus be perfectly adapted to the subbands bit-rate.

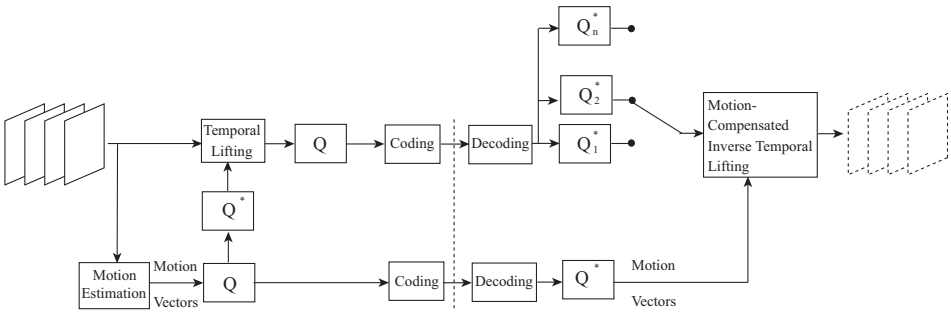


Figure 3.6: Closed loop coding of motion vectors in a video coder: only the wavelet subbands are scalable and thus decodable at the desired bit-rate. Motion vectors are not scalable: their bit-rate are fixed and cannot thus be adapted to the subbands bit-rate.

the vectors coordinates, with quantization step q_v, which controls the rate-distortion trade-off of the motion vectors. The quantized vectors are finally encoded losslessly using an EBCOT encoder [Tau00]. [2]

At the decoder side, the bitstream is decoded with EBCOT and the decoded vectors are rescaled by the quantization step q_v. Then, the motion compensation and the inverse temporal wavelet transform are done using the quantized decoded vectors.

3.2.4 Rate-distortion trade-off

The MSE of the output coded/decoded video depends on the choice of the motion bit-rate R_v as well as the wavelet coefficients bit-rate R_c. Consequently, in order to control the output distortion, it is necessary to tune jointly, and in an optimal way, the quantization of the motion and the one of the wavelet coefficients subbands. This can be done by the way of a

[2] CABAC could also be used.

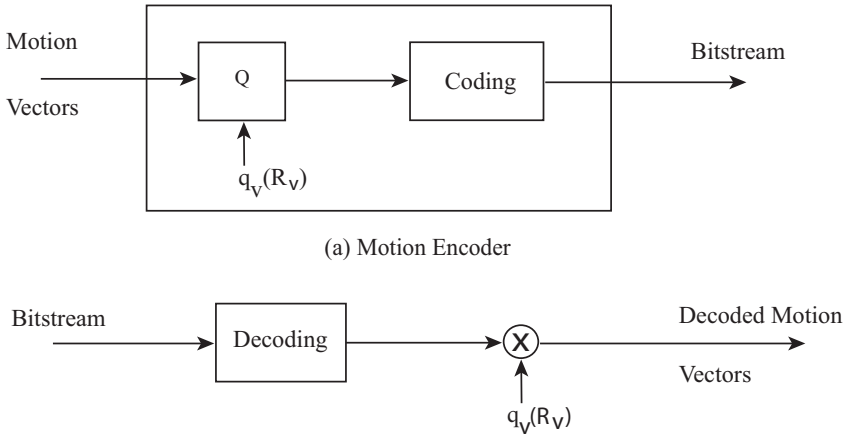


Figure 3.7: Motion vectors encoder and decoder, including the quantization.

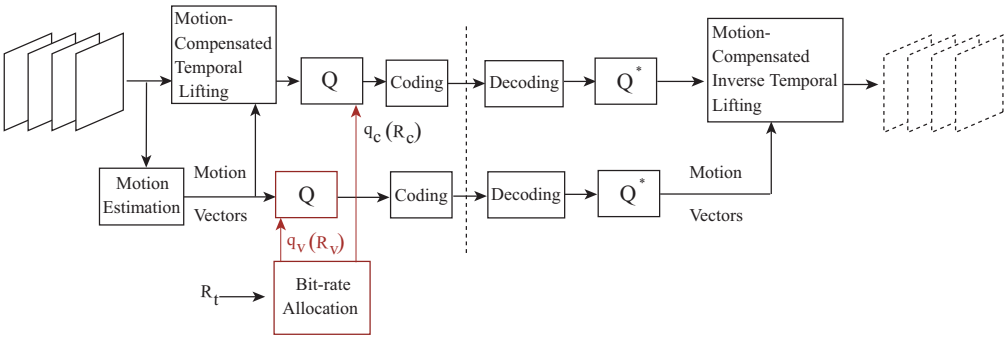


Figure 3.8: General structure of the proposed coder including the lossy coding of motion information and the bit-rate allocation.

bit allocation process, as presented in Figure 3.8. In order to decrease the complexity, this process can be performed thanks to a model. If the total bit-rate R_t can be expressed as a sum of the bit-rates R_v and R_c, on the other hand it is not trivial to obtain a model for the distortion of the reconstructed signal. For that purpose, a model for the convex-hull of the rate-distortion behavior of the proposed MCWT video codec is proposed in Section 3.3.2.

3.3 A rate-distortion model between motion information and wavelet subbands

Obviously, this lossy motion coding has an impact on the decoded sequence. This impact is then evaluated by establishing a theoretical distortion model of motion coding error, including also the subbands quantization noise. This model will allow to perform a model-based bit-rate allocation between wavelet coefficients and motion information.

3.3.1 Background and notations

Section 3.3.1.1 presents some notations and gives in Section 3.3.1.2 the properties of the (2,0) lifting scheme which will be useful for the computation of the distortion model of the quantization error.

3.3.1.1 Notations

Let $\mathbf{v}$ be a motion vector. B_k and F_k respectively denote the "backward" and "forward" motion vectors such that:

$$B_k = \mathbf{v}_{k\to k-1} \text{ and } F_k = \mathbf{v}_{k\to k+1}.$$

The quantized motion vectors are then given by:

$$\widehat{B}_k = Q\left(\mathbf{v}_{k\to k-1}\right) \text{ and } \widehat{F}_k = Q\left(\mathbf{v}_{k\to k+1}\right).$$

where $Q(.)$ stands for the quantization operator. The quantities $\eta_{B_k} = B_k - \widehat{B}_k$ and $\eta_{F_k} = F_k - \widehat{F}_k$ will denote the respective quantization noises. The "backward" (resp. "forward") motion-compensated pixels at frame k can be written as:

$$x_k^{B_{k+1}}(\mathbf{p}) = x_k\left(\mathbf{p} + B_{k+1}\left(\mathbf{p}\right)\right) = x_k^B(\mathbf{p}),$$

and

$$x_k^{F_{k-1}}(\mathbf{p}) = x_k\left(\mathbf{p} + F_{k-1}\left(\mathbf{p}\right)\right) = x_k^F(\mathbf{p}),$$

where $x_l(\mathbf{p})$ corresponds to the pixel x of frame l located at position $\mathbf{p}$. According to these notations, a motion-compensated pixel with quantized motion vector for a "backward" motion is defined as (with a similar expression for the "forward" motion):

$$\widetilde{x}_k^{\widehat{B}_{k+1}}(\mathbf{p}) = x_k\left(\mathbf{p} + \widehat{B}_{k+1}\left(\mathbf{p}\right)\right) = \widetilde{x}_k^{\widehat{B}}(\mathbf{p}).$$

In the following, $\mathbf{Pn}$ will stand for the power operator (with N the number of rows and M the number of columns for one image of the sequence):

$$\mathbf{Pn}\left(x_k\right) = \frac{1}{NM}\sum_{\mathbf{p}} x_k^2\left(\mathbf{p}\right).$$

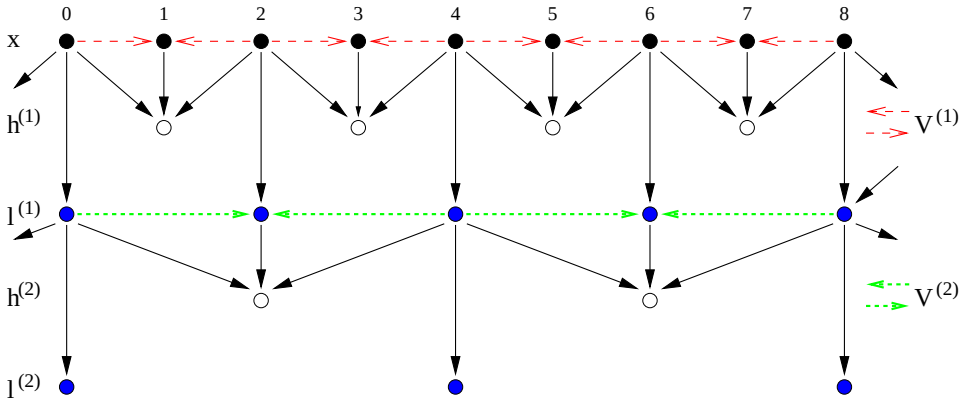


Figure 3.9: (2,0) lifting scheme on two temporal wavelet decomposition levels. $V^{(1)}$ represents the motion vectors at the first level, and $V^{(2)}$ the motion vectors at the second level.

3.3.1.2 The (2,0) motion compensated lifting scheme

The temporal analysis is performed by a (2,0) lifting scheme (see Figure 3.9), which appears to be a simple and yet interesting alternative [LLL+01, ACA+04]. It is obtained from the (2,2) lifting scheme by suppressing the update step; the low-pass filtering is then reduced to a simple temporal subsampling of the original sequence.

The (2,0) lifting scheme analysis equations on one decomposition level are the following:

$$\begin{cases} h_k^{(m)}(\mathbf{p}) = x_{2k+1}(\mathbf{p}) - \frac{1}{2}\left(x_{2k}(\mathbf{p}) + x_{2k+2}(\mathbf{p})\right) \\ l_k^{(m)}(\mathbf{p}) = x_{2k}(\mathbf{p}), \end{cases}$$

where the notation (m) stands for the resolution level. The signals h and l are respectively the high-pass and low-pass subbands.
When motion compensation is introduced in the lifting scheme, the previous analysis equations become:

$$\begin{cases} h_k^{(m)}(\mathbf{p}) = x_{2k+1}(\mathbf{p}) - \frac{1}{2}(x_{2k}^{B^{(m)}}(\mathbf{p}) + x_{2k+2}^{F^{(m)}}(\mathbf{p})) \\ l_k^{(m)}(\mathbf{p}) = x_{2k}(\mathbf{p}), \end{cases} \tag{3.1}$$

where $B^{(m)}$ and $F^{(m)}$ are the motion vectors for the resolution level m.
Then, the corresponding motion compensated synthesis equations can be written as (see Figure 3.9):

$$\begin{cases} x_{2k}(\mathbf{p}) = l_k^{(m)}(\mathbf{p}) \\ x_{2k+1}(\mathbf{p}) = h_k^{(m)}(\mathbf{p}) + \frac{1}{2}(x_{2k}^{B^{(m)}}(\mathbf{p}) + x_{2k+2}^{F^{(m)}}(\mathbf{p})). \end{cases}$$

Thus, with quantized motion vectors, the synthesis equations for one (2,0) temporal decomposition level become (the quantization error on $h_k^{(m)}$ is negligible):

$$\begin{cases} x_{2k}(\mathbf{p}) = l_k^{(m)}(\mathbf{p}) \\ \widetilde{x}_{2k+1}(\mathbf{p}) = h_k^{(m)}(\mathbf{p}) + \frac{1}{2}(\widetilde{x}_{2k}^{\widehat{B}^{(m)}}(\mathbf{p}) + \widetilde{x}_{2k+2}^{\widehat{F}^{(m)}}(\mathbf{p})). \end{cases} \tag{3.2}$$

To simplify the notations in the rest of the development, "$(\mathbf{p})$" will be removed of the equations in the computation of the model. Also, the value $\hat{x}$ will correspond to a quantized sample x.

3.3.2 Distortion model on one decomposition level

In this section, one level of temporal (2,0) MCWT decomposition is considered. A distortion model is derived (with some hypotheses presented in Section 3.3.2.1), including both the quantization noise of the motion vectors (Section 3.3.2.2) and the one of the wavelet coefficients (Section 3.3.3). The limits of the model and possible simplifications are also proposed (Section 3.3.2.3).

3.3.2.1 Hypotheses

Let assume the following hypotheses:

(i) The quantization noise, noted ϵ, is supposed to be additive [GG92]:

$$\hat{x} = x + \epsilon.$$

(ii) Consecutive frames in the video are considered stationary such that it is possible to assume that the power of a frame at time l is almost equal to the power of the compensated frame at same time, i.e.,

$$\mathbf{Pn}\left(x_{2k}^{B}\right) \approx \mathbf{Pn}\left(x_{2k}\right) \quad \text{and} \quad \mathbf{Pn}\left(x_{2k+2}^{F}\right) \approx \mathbf{Pn}\left(x_{2k+2}\right).$$

(iii) Asymptotical (or high-resolution) hypothesis is considered, assuming that a small quantization error is done on the motion vectors. This implies that, for a given resolution level, the power of a compensated frame with the original motion vectors is almost equal to the power of the same frame compensated with the quantized motion vectors:

$$\mathbf{Pn}\left(x_{2k}^{B^{(m)}}\right) \approx \mathbf{Pn}\left(\widetilde{x}_{2k}^{\widehat{B}^{(m)}}\right) \quad \text{and} \quad \mathbf{Pn}\left(x_{2k+2}^{F^{(m)}}\right) \approx \mathbf{Pn}\left(\widetilde{x}_{2k+2}^{\widehat{F}^{(m)}}\right).$$

In the same way:

$$\mathbf{Pn}\left(\widetilde{\epsilon}_{2k}^{\widehat{B}^{(m)}}\right) \approx \mathbf{Pn}\left(\epsilon_{2k}^{B^{(m)}}\right) \approx \mathbf{Pn}\left(\epsilon_{2k}\right),$$

and

$$\mathbf{Pn}\left(\widetilde{\epsilon}_{2k+2}^{\widehat{F}^{(m)}}\right) \approx \mathbf{Pn}\left(\epsilon_{2k+2}^{F^{(m)}}\right) \approx \mathbf{Pn}\left(\epsilon_{2k+2}\right),$$

with ϵ_k the subbands quantization noise.

(iv) The quantization noises between two frames or between the "backward" and "forward" motion vectors are supposed to be uncorrelated.

3.3.2.2 Modeling of the motion vectors quantization noise

The input-output distortion is given by the MSE between the original video x and the decoded video $\widetilde{x}$:

$$D_t = \underbrace{\frac{1}{K}\sum_{k=0}^{K-1} \underbrace{\mathbf{Pn}\left(x_k - \widetilde{x}_k\right)}_{\text{MSE on the frame } k}}_{\text{MSE on the sequence of size } K}, \tag{3.3}$$

where K is the size of the sequence. For one temporal decomposition level, this equation can also be rewritten as:

$$D_t = \frac{1}{K}\sum_{k=0}^{\frac{K}{2}-1}\left[\mathbf{Pn}\left(x_{2k} - \widetilde{x}_{2k}\right) + \mathbf{Pn}\left(x_{2k+1} - \widetilde{x}_{2k+1}\right)\right]. \tag{3.4}$$

Since a (2,0) lifting scheme is used, a direct sub-sampling is done on the even frames without the need of motion vectors (see Section 3.3.1.2 and Figure 3.9). Then, the distortion introduced by the quantization of the motion vectors only has an impact on the second part of (3.4) and the distortion D_t can be simplified in D_v, with

$$D_v = \frac{1}{K}\sum_{k=0}^{\frac{K}{2}-1}\mathbf{Pn}\left(x_{2k+1} - \widetilde{x}_{2k+1}\right). \tag{3.5}$$

Furthermore, by using the first equation of (3.1) and the second equation of (3.2) of the Section 3.3.1.2, and because the motion quantizer works in open loop, the following relation can be obtained:

$$\widetilde{x}_{2k+1} = x_{2k+1} - \frac{1}{2}(x_{2k}^{B^{(1)}} + x_{2k+2}^{F^{(1)}}) + \frac{1}{2}(\widetilde{x}_{2k}^{\widehat{B}^{(1)}} + \widetilde{x}_{2k+2}^{\widehat{F}^{(1)}}),$$

or also:

$$x_{2k+1} - \widetilde{x}_{2k+1} = \frac{1}{2}(x_{2k}^{B^{(1)}} - \widetilde{x}_{2k}^{\widehat{B}^{(1)}}) + \frac{1}{2}(x_{2k+2}^{F^{(1)}} - \widetilde{x}_{2k+2}^{\widehat{F}^{(1)}}). \tag{3.6}$$

Moreover, as the model is established here on one (2,0) temporal decomposition level:

$$x_{2k} - \widetilde{x}_{2k} = x_{2k} - x_{2k} = 0 \qquad \forall k,$$

since there is no motion compensation on the even frames. By combining (3.5) and (3.6), and by assuming that the "backward" and the "forward" motion vectors quantization errors are decorrelated (see Section 3.3.2.1), the distortion due to the quantization of motion vectors becomes:

$$D_v = \frac{1}{K}\sum_{k=0}^{\frac{K}{2}-1}\frac{1}{4}\mathbf{Pn}\left(x_{2k}^{B^{(1)}} - \widetilde{x}_{2k}^{\widehat{B}^{(1)}}\right) + \frac{1}{K}\sum_{k=0}^{\frac{K}{2}-1}\frac{1}{4}\mathbf{Pn}\left(x_{2k+2}^{F^{(1)}} - \widetilde{x}_{2k+2}^{\widehat{F}^{(1)}}\right).$$

By developing, the following relation is obtained:

$$\begin{aligned} D_v = \frac{1}{K}\sum_{k=0}^{\frac{K}{2}-1}\Bigg[&\frac{1}{4}\mathbf{Pn}\left(x_{2k}^{B^{(1)}}\right) + \frac{1}{4}\mathbf{Pn}\left(\widetilde{x}_{2k}^{\widehat{B}^{(1)}}\right) - \frac{1}{2}\left\langle x_{2k}^{B^{(1)}}, \widetilde{x}_{2k}^{\widehat{B}^{(1)}}\right\rangle \\ &+ \frac{1}{4}\mathbf{Pn}\left(x_{2k+2}^{F^{(1)}}\right) + \frac{1}{4}\mathbf{Pn}\left(\widetilde{x}_{2k+2}^{\widehat{F}^{(1)}}\right) - \frac{1}{2}\left\langle x_{2k+2}^{F^{(1)}}, \widetilde{x}_{2k+2}^{\widehat{F}^{(1)}}\right\rangle\Bigg], \end{aligned}$$

with the scalar product $\langle ., . \rangle$ defined as (with similar expression for the "forward" motion):

$$\left\langle x_{2k}^{B^{(1)}}, \widetilde{x}_{2k}^{\widehat{B}^{(1)}}\right\rangle = \frac{1}{M}\sum_{\mathbf{p}_1,\mathbf{p}_2} x_{2k}^{B^{(1)}}(\mathbf{p}_1) \times \widetilde{x}_{2k}^{\widehat{B}^{(1)}}(\mathbf{p}_2).$$

Assuming that the image is stationary in one GOP at time k (Section 3.3.2.1), the scalar products become:

$$\left\langle x_{2k}^{B^{(1)}}, \widetilde{x}_{2k}^{\widehat{B}^{(1)}}\right\rangle = \Gamma_{x_{2k}}(\eta_{B^{(1)}})$$

and

$$\left\langle x_{2k+2}^{F^{(1)}}, \widetilde{x}_{2k+2}^{\widehat{F}^{(1)}}\right\rangle = \Gamma_{x_{2k+2}}(\eta_{F^{(1)}}),$$

with $\Gamma_{x_{2k}}$ and $\Gamma_{x_{2k+2}}$ the autocorrelation functions of the signals x_{2k} and x_{2k+2}, $\eta_{B^{(1)}} = B^{(1)} - \widehat{B}^{(1)}$ and $\eta_{F^{(1)}} = F^{(1)} - \widehat{F}^{(1)}$, the quantization errors on "Backward" and "Forward" motion vectors.
As expressed in Section 3.3.2.1, thanks to the high bit-rate assumption, the quantization errors $\eta_{B^{(1)}}$ and $\eta_{F^{(1)}}$ on motion vectors are considered smalls. With similar expression for $\mathbf{Pn}\left(x_{2k+2}^{F^{(1)}}\right)$, one can have:

$$\mathbf{Pn}\left(x_{2k}^{B^{(1)}}\right) \approx \mathbf{Pn}\left(\widetilde{x}_{2k}^{\widehat{B}^{(1)}}\right) \approx \mathbf{Pn}\left(x_{2k}\right).$$

And thus for the motion distortion model on one (2,0) decomposition level:

$$D_v = \frac{1}{2K}\sum_{k=0}^{\frac{K}{2}-1}\left[\mathbf{Pn}\left(x_{2k}\right) - \Gamma_{x_{2k}}\left(\eta_{B^{(1)}}\right) + \mathbf{Pn}\left(x_{2k+2}\right) - \Gamma_{x_{2k+2}}\left(\eta_{F^{(1)}}\right)\right]. \tag{3.7}$$

3.3.2.3 Limits and simplification of the model

The quantization errors $\eta_{B^{(1)}}$ and $\eta_{F^{(1)}}$ correspond to shifts between two compensated frames. If there are not errors on the motion vectors, these shifts are equal to zero and thus:

$$\Gamma_{x_{2k}}\left(\eta_{B^{(1)}}\right) = \Gamma_{x_{2k}}\left(\mathbf{0}\right) = \mathbf{Pn}\left(x_{2k}\right)$$

and

$$\Gamma_{x_{2k+2}}\left(\eta_{F^{(1)}}\right) = \Gamma_{x_{2k+2}}\left(\mathbf{0}\right) = \mathbf{Pn}\left(x_{2k+2}\right).$$

Then, the distortion D_v is also equal to zero (see (3.7)) and confirms that there are no errors on the decoded video due to the quantization of the motion vectors. On the other hand, if there is a big error on the motion vectors, with a similar expression for the "forward" motion, one can write:

$$\lim_{\eta_{B^{(1)}} \to \infty} \Gamma_{x_{2k}}\left(\eta_{B^{(1)}}\right) = 0,$$

implying

$$\lim_{\eta \to \infty} D_v = \frac{1}{2K} \sum_{k=0}^{\frac{K}{2}-1} \left[\mathbf{Pn}\left(x_{2k}\right) + \mathbf{Pn}\left(x_{2k+2}\right)\right],$$

which is the asymptotical limit of the model when the motion vectors bit-rate is equal to zero.

Moreover, if it is assumed that the video sequence is stationary inside each GOP and that the "backward" and "forward" motion vectors are estimated symmetrically (that is to say that $B^{(1)} = -F^{(1)}$, see [ACA+04]), one can write:

$$\mathbf{Pn}\left(x_{2k}\right) \approx \mathbf{Pn}\left(x_{2k+2}\right)$$

and

$$\Gamma_{x_{2k}}\left(\eta_{B^{(1)}}\right) \approx \Gamma_{x_{2k+2}}\left(\eta_{F^{(1)}}\right).$$

Equation (3.7) can therefore be simplified in:

$$D_v \approx \frac{1}{K} \sum_{k=0}^{\frac{K}{2}-1} \left[\mathbf{Pn}\left(x_{2k}\right) - \Gamma_{x_{2k}}(\eta_{B^{(1)}})\right] \approx \frac{1}{K} \sum_{k=0}^{\frac{K}{2}-1} \left[\mathbf{Pn}\left(x_{2k+2}\right) - \Gamma_{x_{2k+2}}(\eta_{F^{(1)}})\right].$$

These equations mean that the knowledge of the frames x_{2k} (or x_{2k+2}) allows to estimate the distortion introduced by the motion vectors quantization, under the hypothesis presented in Section 3.3.2.1. Indeed, this distortion is simply function of the x_{2k} (or x_{2k+2}) frame powers and of the x_{2k} (or x_{2k+2}) frame autocorrelation functions.

3.3.3 Including the subbands quantization noise

When the high frequency temporal wavelet coefficient subbands ($h^{(1)}$) and the low frequency subband ($l^{(1)}$) of each GOP (of the first decomposition level) are quantized, the total distortion D_t on one decomposition level is given by (expressed from equation 3.4):

$$D_t = \frac{1}{K} \sum_{k=0}^{\frac{K}{2}-1} \mathbf{Pn}\left(x_{2k} - \widetilde{x}_{2k}\right) + \mathbf{Pn}\left(x_{2k+1} - \widetilde{x}_{2k+1}\right). \tag{3.8}$$

Let introduce $\epsilon_{h^{(1)}}$, the quantization noise of the high frequency frame $2k+1$, and ϵ_{2k} and ϵ_{2k+2}, the quantization noises of the low frequency frames $2k$ and $2k+2$.
Due to the properties of the (2,0) lifting scheme on one decomposition level (see Figure 3.9), the first term of equation 3.8 is simply equal to the low frequency subbands coding error on the image x_{2k}:

$$\mathbf{Pn}\left(x_{2k} - \widetilde{x}_{2k}\right) = \mathbf{Pn}(\epsilon_{2k}). \tag{3.9}$$

The (2,0) lifting scheme analysis equation for the high frequencies on one decomposition level becomes:

$$\widehat{h}^{(1)} = x_{2k+1} - \frac{1}{2}(x_{2k}^{B^{(1)}} + x_{2k+2}^{F^{(1)}}) + \epsilon_{h^{(1)}},$$

and the synthesis equation:

$$\widehat{\widetilde{x}}_{2k+1} = \widehat{h}^{(1)} + \frac{1}{2}(\widetilde{x}_{2k}^{\widehat{B}^{(1)}} + \widetilde{x}_{2k+2}^{\widehat{F}^{(1)}}).$$

By combining the two previous equations, the following relation is obtained:

$$\begin{aligned} x_{2k+1} - \widehat{\widetilde{x}}_{2k+1} = \frac{1}{2}\Bigg(& (x_{2k}^{B^{(1)}} - \widetilde{x}_{2k}^{\widehat{B}^{(1)}} - \widetilde{\epsilon}_{2k}^{\widehat{B}^{(1)}}) \\ & + (x_{2k+2}^{F^{(1)}} - \widetilde{x}_{2k+2}^{\widehat{F}^{(1)}} - \widetilde{\epsilon}_{2k+2}^{\widehat{F}^{(1)}})\Bigg) - \epsilon_{h^{(1)}}. \end{aligned} \tag{3.10}$$

Therefore, the total distortion D_t becomes, after combining (3.9) and (3.10):

$$\begin{aligned} D_t = \frac{1}{K} \sum_{k=0}^{\frac{K}{2}-1} \Bigg[& \frac{1}{4}(\mathbf{Pn}(x_{2k}^{B^{(1)}} - \widetilde{x}_{2k}^{\widehat{B}^{(1)}} - \widetilde{\epsilon}_{2k}^{\widehat{B}^{(1)}}) + \mathbf{Pn}(x_{2k+2}^{F^{(1)}} - \widetilde{x}_{2k+2}^{\widehat{F}^{(1)}} - \widetilde{\epsilon}_{2k+2}^{\widehat{F}^{(1)}})) \\ & + \mathbf{Pn}(\epsilon_{2k}) + \mathbf{Pn}(\epsilon_{h^{(1)}})\Bigg]. \end{aligned}$$

By developing (with similar expressions as previously for the scalar products):

$$\begin{aligned}D_t =&\frac{1}{2K}\sum_{k=0}^{\frac{K}{2}-1}[\frac{1}{2}\mathbf{Pn}(x_{2k}^{B^{(1)}})+\frac{1}{2}\mathbf{Pn}(\widetilde{x}_{2k}^{\widehat{B}^{(1)}})+\frac{1}{2}\mathbf{Pn}(\widetilde{\epsilon}_{2k}^{\widehat{B}^{(1)}})\\&-\langle x_{2k}^{B^{(1)}},\widetilde{x}_{2k}^{\widehat{B}^{(1)}}\rangle-\langle x_{2k}^{B^{(1)}},\widetilde{\epsilon}_{2k}^{\widehat{B}^{(1)}}\rangle-\langle\widetilde{x}_{2k}^{\widehat{B}^{(1)}},\widetilde{\epsilon}_{2k}^{\widehat{B}^{(1)}}\rangle+\frac{1}{2}\mathbf{Pn}(x_{2k+2}^{F^{(1)}})\\&+\frac{1}{2}\mathbf{Pn}(\widetilde{x}_{2k+2}^{\widehat{F}^{(1)}})+\frac{1}{2}\mathbf{Pn}(\widetilde{\epsilon}_{2k+2}^{\widehat{F}^{(1)}})+2\mathbf{Pn}(\epsilon_{2k})+2\mathbf{Pn}(\epsilon_{h^{(1)}})\\&-\langle x_{2k+2}^{F^{(1)}},\widetilde{x}_{2k+2}^{\widehat{F}^{(1)}}\rangle-\langle x_{2k+2}^{F^{(1)}},\widetilde{\epsilon}_{2k+2}^{\widehat{F}^{(1)}}\rangle-\langle\widetilde{x}_{2k+2}^{\widehat{F}^{(1)}},\widetilde{\epsilon}_{2k+2}^{\widehat{F}^{(1)}}\rangle\Big].\end{aligned}$$

Let us assume that the cross scalar products $\left\langle x_{2k}^{B^{(1)}},\widetilde{\epsilon}_{2k}^{\widehat{B}^{(1)}}\right\rangle$, $\left\langle\widetilde{x}_{2k}^{\widehat{B}^{(1)}},\widetilde{\epsilon}_{2k}^{\widehat{B}^{(1)}}\right\rangle$, $\left\langle x_{2k+2}^{F^{(1)}},\widetilde{\epsilon}_{2k+2}^{\widehat{F}^{(1)}}\right\rangle$ and $\left\langle\widetilde{x}_{2k+2}^{\widehat{F}^{(1)}},\widetilde{\epsilon}_{2k+2}^{\widehat{F}^{(1)}}\right\rangle$ are equal to zero (decorrelation between the corresponding data, see Section 3.3.2.1). Let assume the asymptotical hypothesis expressed in Section 3.3.2.1, in particular:

$$\mathbf{Pn}\left(\widetilde{\epsilon}_{2k}^{\widehat{B}^{(1)}}\right)\approx\mathbf{Pn}\left(\epsilon_{2k}^{B^{(1)}}\right)\approx\mathbf{Pn}\left(\epsilon_{2k}\right).$$

By using the same expressions for the scalar products $\langle x_{2k}^{B^{(1)}},\widetilde{x}_{2k}^{\widehat{B}^{(1)}}\rangle$ and $\langle x_{2k+2}^{F^{(1)}},\widetilde{x}_{2k+2}^{\widehat{F}^{(1)}}\rangle$ as in Section 3.3.2.2, D_t can be simplified as:

$$\begin{aligned}D_t=\frac{1}{2K}\sum_{k=0}^{\frac{K}{2}-1}\Big[&\mathbf{Pn}\left(x_{2k}\right)-\Gamma_{x_{2k}}\left(\eta_{B^{(1)}}\right)+\mathbf{Pn}\left(x_{2k+2}\right)-\Gamma_{x_{2k+2}}\left(\eta_{F^{(1)}}\right)\\&+\frac{5}{2}\mathbf{Pn}(\epsilon_{2k})+\frac{1}{2}\mathbf{Pn}(\epsilon_{2k+2})+2\mathbf{Pn}(\epsilon_{h^{(1)}})\Big].\end{aligned}$$

Equivalently, by introducing $\epsilon_{l^{(1)}}$ and $\epsilon_{h^{(1)}}$, the low frequency subband noise and the high frequency subband noise for the first temporal decomposition level, the previous equation can be rewritten as:

$$\begin{aligned}D_t\approx\frac{1}{2K}\sum_{k=0}^{\frac{K}{2}-1}&\Big[\mathbf{Pn}\left(x_{2k}\right)-\Gamma_{x_{2k}}\left(\eta_{B^{(1)}}\right)+\mathbf{Pn}\left(x_{2k+2}\right)-\Gamma_{x_{2k+2}}\left(\eta_{F^{(1)}}\right)\Big]\\&+\frac{1}{K}\Big[\frac{3}{2}\mathbf{Pn}(\epsilon_{l^{(1)}})+\mathbf{Pn}(\epsilon_{h^{(1)}})\Big].\end{aligned}\tag{3.11}$$

3.3.4 Total distortion model on N temporal decomposition levels

It is possible to generalize to N temporal decomposition levels the distortion model (3.11) given in the previous section. Here, the results obtained for two decomposition levels (Section 3.3.4.1), and the generalization to N decomposition levels in Section 3.3.4.2 are presented. The validation of the distortion model is presented in Section 3.3.4.3.

3.3.4.1 Case of two decomposition levels

The details of these computations are presented in Appendix A. Taking into account the structure of the lifting scheme (see Figure 3.9), the distortion model given by (3.3) can be expanded on two temporal decomposition levels as:

$$D_t = \frac{1}{K}\Big(\sum_{k=0}^{\frac{K}{4}-1} \big(\mathbf{Pn}\left(x_{4k} - \widetilde{x}_{4k}\right) + \mathbf{Pn}\left(x_{4k+2} - \widetilde{x}_{4k+2}\right)\big) + \sum_{k=0}^{\frac{K}{2}-1} \mathbf{Pn}\left(x_{2k+1} - \widetilde{x}_{2k+1}\right)\Big). \tag{3.12}$$

Due to the properties of the (2,0) lifting scheme on two decomposition levels (see Figure 3.9), the first term of this equation (called D_{4k}) is simply equal to the low frequency subbands coding error on the images x_{4k}:

$$D_{4k} = \sum_{k=0}^{\frac{K}{4}-1} \mathbf{Pn}(\epsilon_{4k}). \tag{3.13}$$

The second term of (3.12) (called in the following D_{4k+2}) has thus to be computed, thanks to the lifting equations of analysis and synthesis for the second decomposition level. After calculus, D_{4k+2} can be expressed as:

$$D_{4k+2} = \frac{1}{2K} \sum_{k=0}^{\frac{K}{4}-1} \Big[\mathbf{Pn}\left(x_{4k}\right) - \Gamma_{x_{4k}}\left(\eta_{B^{(2)}}\right) + \mathbf{Pn}\left(x_{4k+4}\right) - \Gamma_{x_{4k+4}}\left(\eta_{F^{(2)}}\right) + \frac{1}{2}\mathbf{Pn}(\epsilon_{4k}) + \frac{1}{2}\mathbf{Pn}(\epsilon_{4k+4}) + 2\mathbf{Pn}(\epsilon_{h^{(2)}}) \Big]. \tag{3.14}$$

Finally, the last term called D_{2k+1} have to be computed, which gives:

$$D_{2k+1} = \frac{1}{2K} \sum_{k=0}^{\frac{K}{2}-1} \Big[\mathbf{Pn}\left(x_{2k}\right) - \Gamma_{x_{2k}}\left(\eta_{B^{(1)}}\right) + \mathbf{Pn}\left(x_{2k+2}\right) - \Gamma_{x_{2k+2}}\left(\eta_{F^{(1)}}\right) + \frac{1}{2}\mathbf{Pn}(\epsilon_{2k}) + \frac{1}{8}\mathbf{Pn}(\epsilon_{4k}) + \frac{1}{8}\mathbf{Pn}(\epsilon_{4k+4}) + \frac{1}{2}\mathbf{Pn}(\epsilon_{h^{(2)}}) + 2\mathbf{Pn}(\epsilon_{h^{(1)}}) \Big]. \tag{3.15}$$

And by adding (3.13), (3.14) and (3.15), the total distortion model on two decomposition levels is obtained by:

$$\begin{aligned} D_t \approx & \frac{1}{2K}\sum_{k=0}^{\frac{K}{2}-1}\left[\mathbf{Pn}\left(x_{2k}\right)-\Gamma_{x_{2k}}\left(\eta_{B^{(1)}}\right)+\mathbf{Pn}\left(x_{2k+2}\right)-\Gamma_{x_{2k+2}}\left(\eta_{F^{(1)}}\right)\right] \\ & +\frac{1}{2K}\sum_{k=0}^{\frac{K}{4}-1}\left[\mathbf{Pn}(x_{4k})-\Gamma_{x_{4k}}(\eta_{B^{(2)}})+\mathbf{Pn}(x_{4k+4})-\Gamma_{x_{4k+4}}(\eta_{F^{(2)}})\right] \\ & +\frac{1}{K}\left[\frac{1}{2^2}\mathbf{Pn}(\epsilon_{l^{(2)}})+\sum_{i=1}^{2}\frac{1}{2^i}\mathbf{Pn}(\epsilon_{h^{(i)}})\right], \end{aligned}$$

with $l^{(2)}$ the low frequency subband and $h^{(i)}$ the high frequency subband at the i^{th} decomposition level.

3.3.4.2 Case of N decomposition levels

The approximation of the total input/output distortion D_t can be easily generalized to several temporal decomposition levels.

In order to simplify the notations, let us write:

$$\eta_{B^{(n+1)}}=\eta_{B^{(n+1)}_{2^{N-n}k+2^{N-n-1}}},$$

and

$$\eta_{F^{(n+1)}}=\eta_{F^{(n+1)}_{2^{N-n}k+2^{N-n-1}}}.$$

Then, using the results of Section 3.3.4.1, it is possible to have an expression of the total distortion:

$$\begin{aligned} D_t \approx & \frac{1}{2K}\sum_{n=0}^{N-1}\sum_{k=0}^{\frac{K}{2^{N-n}}-1}\Big[\mathbf{Pn}\left(x_{2^{N-n}k}\right)-\Gamma_{x_{2^{N-n}k}}(\eta_{B^{(n+1)}}) \\ & +\mathbf{Pn}\left(x_{2^{N-n}k+2^{N-n}}\right)-\Gamma_{x_{2^{N-n}k+2^{N-n}}}(\eta_{F^{(n+1)}})\Big] \\ & +\frac{1}{K}\left[\frac{1}{2^N}\mathbf{Pn}(\epsilon_{l^{(N)}})+\sum_{i=1}^{N}\frac{1}{2^i}\mathbf{Pn}(\epsilon_{h^{(i)}})\right], \end{aligned} \tag{3.16}$$

with N the number of levels, $l^{(N)}$ the low frequency subband at the lowest resolution level N and $h^{(i)}$ the high frequency subband at the i^{th} decomposition level.

Obviously, it can be seen that in (3.16) it is possible to separate the motion vectors noise from the wavelet coefficients noise, and to write:

$$D_t = D_v + D_c,$$

with the motion distortion:

$$D_v \approx \frac{1}{2K} \sum_{n=0}^{N-1} \sum_{k=0}^{\frac{K}{2^{N-n}}-1} \Big[\mathbf{Pn}\left(x_{2^{N-n}k}\right) - \Gamma_{x_{2^{N-n}k}}(\eta_{B^{(n+1)}}) + \mathbf{Pn}\left(x_{2^{N-n}k+2^{N-n}}\right) - \Gamma_{x_{2^{N-n}k+2^{N-n}}}(\eta_{F^{(n+1)}}) \Big], \tag{3.17}$$

and the temporal subbands distortion:

$$D_c = \frac{1}{K}\left[\frac{1}{2^N}\mathbf{Pn}(\epsilon_{l^{(N)}}) + \sum_{i=1}^{N}\frac{1}{2^i}\mathbf{Pn}(\epsilon_{h^{(i)}})\right].$$

Besides, it is known that D_t depends on the motion information and the wavelet subbands quantization steps, respectively q_v and q_c. Since there is a direct link between the quantization step and the bit-rate, the distortion model of (3.16) is also a function of the motion bit-rate R_v and of the set of the M subbands bit-rates $\mathbf{R_c} = \{R_i\}_{i=1}^{M}$. The quantity R_i corresponds to the bit-rate of the temporal subband i among the M subbands.[3] Thus, one can write:

$$D_t = D_t(R_v, \mathbf{R_c}).$$

This model could be easily derived for other lifting schemes, as the (2,2). In this framework, a focus is done on the (2,0) lifting scheme, since it is an efficient alternative for video coding [LLL+01].

3.3.4.3 Validity of the proposed model

The input/output distortion model has been defined as a function of the motion and the wavelet coefficients quantization steps. An example of the distortion model of (3.16) as a function of R_v and R_c is plotted in 3D in Figure 3.10 for two temporal decomposition levels of the video sequence FOREMAN.

In order to validate the proposed approach, the input/output experimental rate-distortion behavior of the CIF sequence FOREMAN and SD sequence CITY is compared in Figure 3.11 to the theoretical models. The motion noise and the subbands noise models are first validated separately. The sequences have been decomposed on two temporal decomposition levels with a (2,0) lifting scheme: in (a), the motion vectors estimated with a block-matching

[3] The subband bit-rate value R_c can also be defined as:

$$R_c = \sum_{i=1}^{M} a_i R_i.$$

The quantity a_i is simply the fraction of total pixels in the i-th subband.

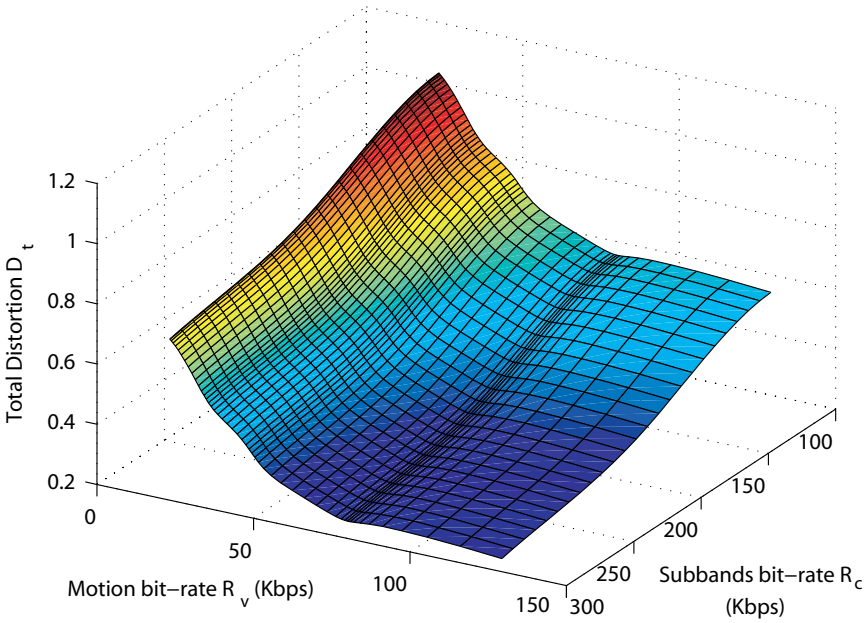
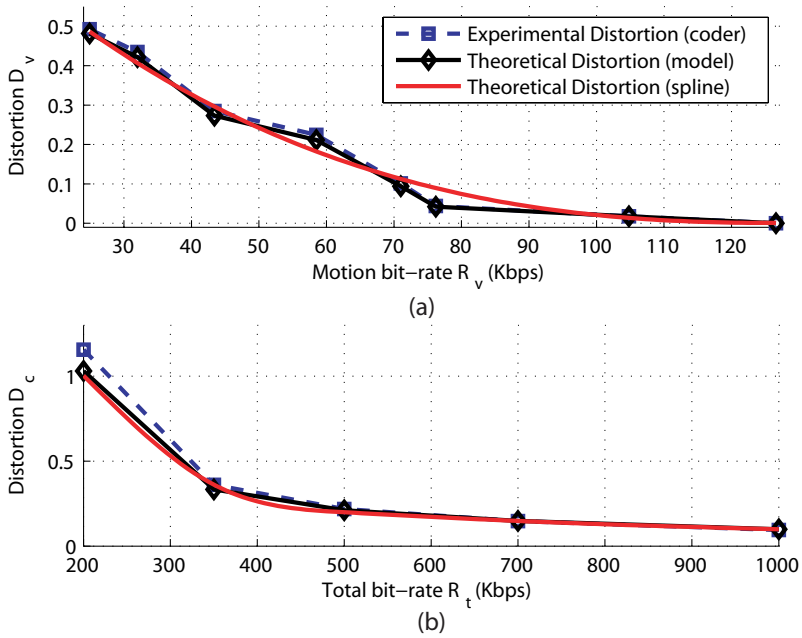


Figure 3.10: Typical behavior of the distortion model D_t. This result was obtained on the CIF video FOREMAN decomposed on 2 temporal levels, with quarter-pixel motion vectors.

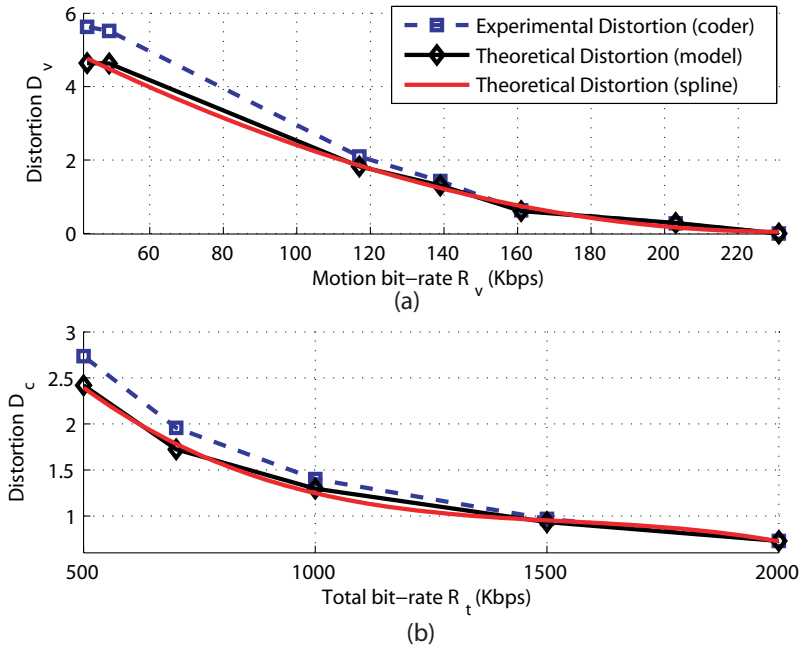
algorithm at, respectively, quarter-pixel and half-pixel precision are coded with different quantization steps q_v (i.e. with different motion bit-rates R_v) and the wavelet subbands are coded losslessly; in (b), the motion information is not quantized whereas the subbands are coded with losses. The input/output experimental rate-distortion behavior of the quantized CIF sequences FOREMAN and CITY has also been compared in TABLE 3.2 to the theoretical model of the motion quantization noise. The errors in % between theory and experimentation are also presented. The sequences have been decomposed on one or two temporal decomposition levels with a (2,0) lifting scheme: the "Backward" and "Forward" motion vectors estimated with a block-matching algorithm with blocks of fixed size 16x16 (respectively at a quarter-pixel and a half-pixel accuracy), are coded with different quantization steps q_v (i.e. with different motion bit-rates R_v) and the wavelet subbands are coded losslessly.

Then, figures 3.12(a) and 3.12(b) show the results using the global formula of (3.16) for FOREMAN at 500 Kbps and for CREW at 1500 Kbps, on two decomposition levels and for different motion bit-rates R_v: here, both wavelet subbands and, respectively, quarter-pixel and half-pixel vectors are quantized. Some points of the model are computed, and smoothing B-splines are used to approximate the curves.

The results of TABLE 3.2, figures 3.11, 3.12(a) and 3.12(b) show that



(a) Foreman.



(b) city.

Figure 3.11: Experimental rate-distortion curves and their approximations using the theoretical distortion model. The simulations have been done on the CIF Foreman and SD city sequences, decomposed on two temporal levels and with, respectively, quarter-pixel and half-pixel motion vectors. (a) Subbands coded losslessly and motion vectors quantized with losses at different bit-rates R_v, (b) motion vectors coded losslessly and wavelet coefficients quantized with losses at different bit-rates R_c (for visibility reasons, this curve is plotted in function of $R_t = R_v + R_c$).

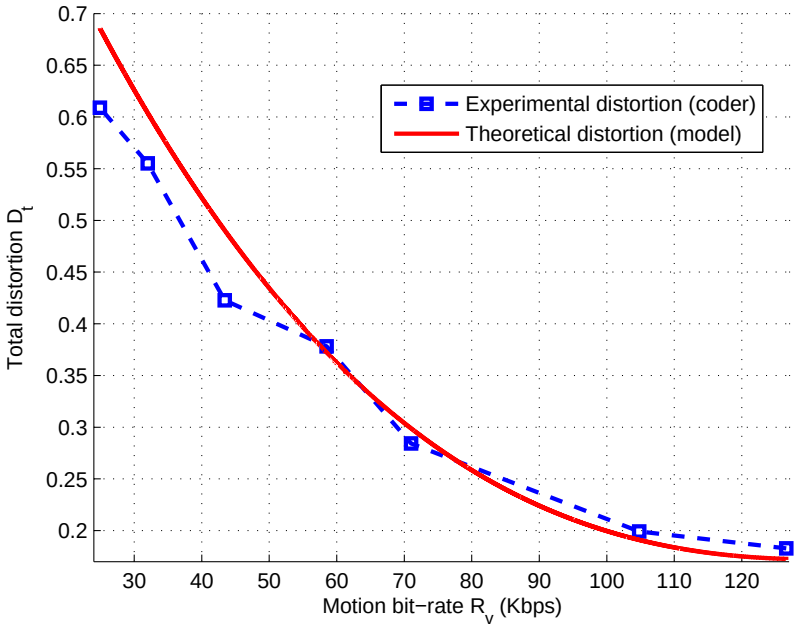
Foreman, 1 level (Kbps)	**79**	**65**	**50**	**49**	**43**	**41**	**40**	**39**
Th- distortion	0	0.113	0.14	0.19	0.21	0.22	0.22	0.24
Exp- distortion	0	0.11	0.13	0.18	0.19	0.20	0.21	0.21
Errors (%)	0	2.7	3.6	4.7	5	8.7	9.3	11.8
City, 1 level (Kbps)	**125**	**108**	**93**	**79**	**62**	**43**	**25**	**19**
Th- distortion	0	0.172	0.23	0.73	1.28	1.33	1.43	1.52
Exp- distortion	0	0.17	0.22	0.72	1.15	1.18	1.26	1.32
Errors (%)	0	1.1	4.3	1.9	10	11.2	11.8	12
Foreman, 2 levels (Kbps)	**126**	**104**	**76**	**71**	**58**	**43**	**32**	**24**
Th- distortion	0	0.02	0.04	0.09	0.19	0.25	0.41	0.56
Exp- distortion	0	0.0202	0.041	0.10	0.22	0.28	0.45	0.49
Errors (%)	0	1.1	2.3	6	10.4	10.7	10	12.1
City, 2 levels (Kbps)	**231**	**203**	**161**	**139**	**117**	**115**	**49**	**43**
Th- distortion	0	0.29	0.61	1.31	1.83	2.72	4.63	4.64
Exp- distortion	0	0.28	0.63	1.43	2.1	3.21	5.52	5.63
Errors (%)	0	3.1	4	8.3	13.3	15.8	16.3	17.5

Table 3.1: Experimental rate-distortion points and their approximations using the theoretical distortion model. The simulations have been done on the FOREMAN and CITY sequences decomposed on one or two temporal levels with respectively quarter-pixel and half-pixel motion vectors. Subbands are coded losslessly and motion vectors are quantized with losses at different bit-rates R_v. The errors (in %) between theory and experimentation are also presented.

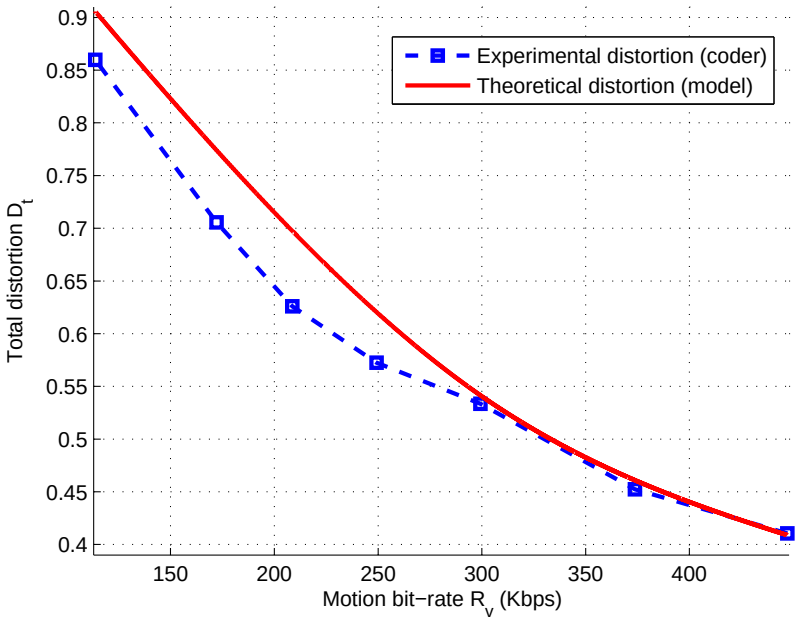
the theoretical and experimental points are very close. The curves follow the same progression. Indeed, less than 10% of error on average between theory and experimentation is observed. Therefore, the proposed theoretical distortion model for the coding error provides a good approximation. It can be nevertheless noticed that the model gives a better approximation at high bit-rate, which is relevant since some asymptotic hypothesis have been done; but the results at low bit-rates are also convincing.

3.3.5 Model-based bit-rate allocation

The particularity of the proposed coder is to quantize with losses both the motion vectors and the temporal wavelet coefficients. Obviously, it is then necessary to dispatch in an optimal way the bit budget R_t between motion and wavelet coefficients in order to minimize the total distortion $D_t(R_v, R_c)$. This can be done in an efficient way thanks to a model-based bit-rate allocation algorithm using the proposed distortion model of (3.16) as described in this section.



(a) FOREMAN, $R_t = 500$ Kbps.



(b) CREW, $R_t = 1500$ Kbps.

Figure 3.12: Experimental rate-distortion curve and its approximation using the theoretical distortion model for a total bit-rate R_t, on FOREMAN and CREW decomposed on two temporal levels with motion vectors estimated with a quarter-pixel accuracy (size of blocks 16x16). In these experiments, both motion and wavelet coefficients are quantized with losses.

3.3.5.1 Optimization problem

The temporal analysis used here is a dyadic one-dimensional decomposition on N levels resulting in $M = N + 1$ subbands: N high-frequency and 1 low-frequency. The problem arises of assigning the coding resources to the subbands, so that either the distortion is minimized for a given target bit-rate, or the bit-rate is minimized for a given target quality [ACAB07].

The bit-rate allocation problem (P) consists in finding the value of R_v and the set $\mathbf{R_c} = \{R_i\}_{i=1}^{M}$ which minimizes the distortion $D_t(R_v, \mathbf{R_c})$ given by the model of (3.16). Thus, the problem (P) can be written as:

$$(P) \begin{cases} \min\limits_{R_v, \mathbf{R_c}} D_t(R_v, \mathbf{R_c}) \\ \text{under constraint } R_v + \sum_{i=1}^{M} a_i R_i = R_t \ . \end{cases}$$

This problem can be easily solved using the Lagrange multipliers. To that purpose, the following differentiable and convex functional is introduced:

$$J_\lambda(R_v, \mathbf{R_c}) = D_t(R_v, \mathbf{R_c}) - \lambda(R_v + \sum_{i=1}^{M} a_i R_i - R_t).$$

The optimal rate allocation vector $\mathbf{R}^* = [R_v^*, \{R_i^*\}_{i=1}^{M}]$ is then obtained by minimizing $J_\lambda(R_v, \mathbf{R_c})$. In the hypothesis of differentiability and by imposing the zero-gradient condition, the optimal rate allocation vector must verify the following set of equations:

$$\begin{cases} \frac{1}{a_i} \frac{\partial D_t(R_v, \mathbf{R_c})}{\partial R_i} \Big|_{R_v(\lambda), \mathbf{R_c} = \{R_i(\lambda)\}_{i=1}^{M}} = \lambda, \forall i \in \{1, ..., M\} \\ \frac{\partial D_t(R_v, \mathbf{R_c})}{\partial R_v} \Big|_{R_v(\lambda), \mathbf{R_c} = \{R_i(\lambda)\}_{i=1}^{M}} = \lambda. \end{cases}$$

where the value of $R_i(\lambda)$ is the bit-rate of the i-th subband which corresponds to a slope λ on the rate-distortion curve. Note that $\lambda \leq 0$ since the rate-distortion curve is decreasing. Then, the constrained bit-rate allocation problem consists in finding the slope value λ^* such that:

$$R_v(\lambda^*) + \sum_{i=1}^{M} a_i R_i(\lambda^*) = R_t.$$

Simple algorithms exist which allow to find λ^*, among which the bisection method, the Newton method, the Golden Section method, and the Secant method can be mentioned. These algorithms usually converge after 3 to 6 iterations, and their complexity is negligible if compared to the other parts of video coder such as motion estimation and compensation.

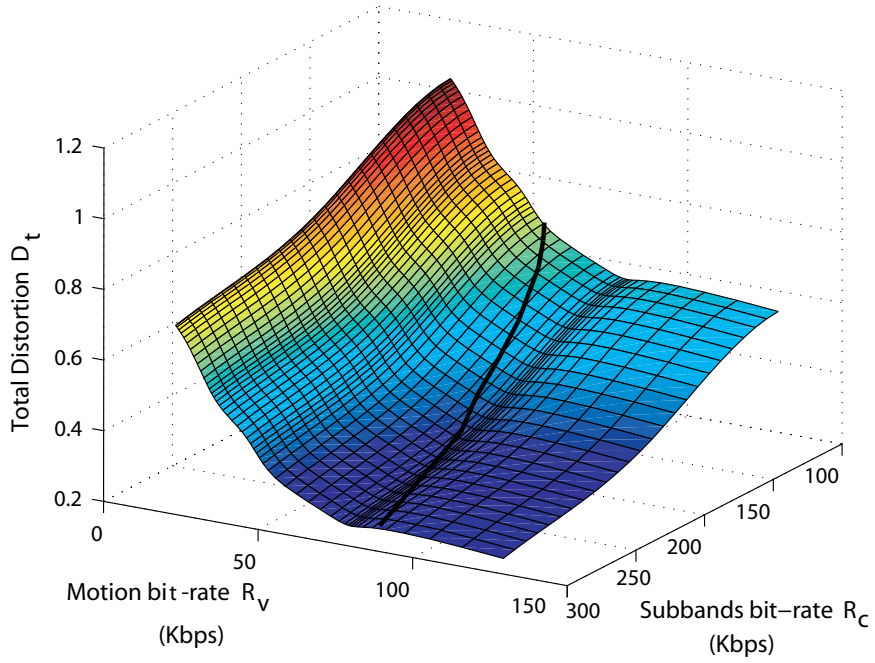


Figure 3.13: Results on the optimization of the functional $J_\lambda(R_v, \mathbf{R_c})$ for the CIF video FOREMAN decomposed on two temporal levels and for motion vectors estimated at a quarter-pixel precision. The solid line corresponds to the evolution of the distortion $D_t(R_v^*, \mathbf{R_c^*})$, solution of the minimization of $J_\lambda(R_v, \mathbf{R_c})$.

3.3.5.2 Bit allocation algorithm

Therefore, the proposed bit allocation algorithm works as follows:

1. $\lambda = \lambda_{init}$
2. For each value of the set $\mathbf{R_c}$, find the value $R_v^*(\lambda, \mathbf{R_c})$ that minimizes the criterion $J_\lambda(R_v, \mathbf{R_c})$
3. Find the set of values $\{R_i^*\}_{i=1}^M$ that minimizes the criterion $J_\lambda(R_v, \mathbf{R_c})|_{R_v = R_v^*(\lambda, \mathbf{R_c})}$
4. If $R_v^*(\lambda, \mathbf{R_c}) + \sum_{i=1}^M a_i R_i^* = R_t$ then stop, else change the value of λ and go to step 2.

3.3.6 Performances

In this section is presented the evaluation of the proposed model-based bit allocation in the framework of MCWT video coding. The experiments (Section 3.3.6.1) and the evaluation of the coder performances (Section 3.3.6.2)

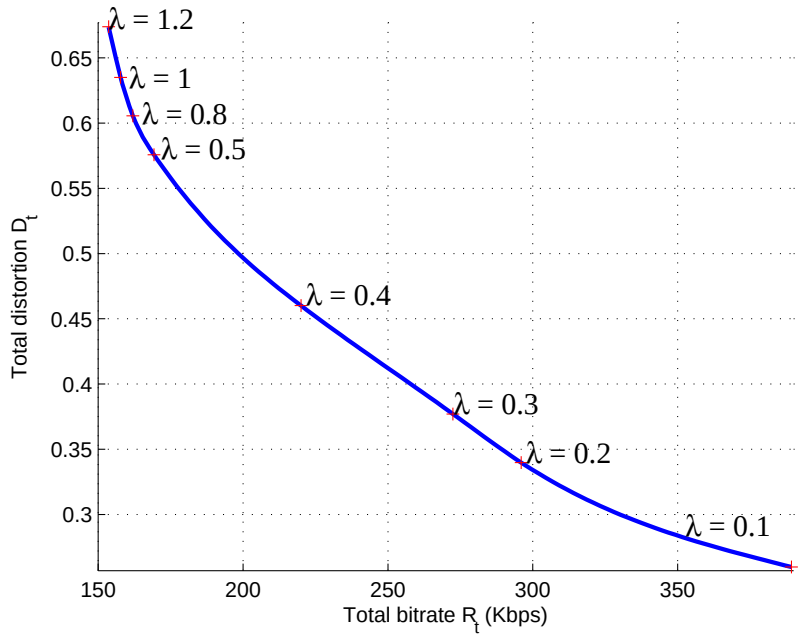


Figure 3.14: Results on the optimization of the functional $J_\lambda(R_v, \mathbf{R_c})$ for the CIF video FOREMAN decomposed on two temporal levels and for motion vectors estimated at a quarter-pixel accuracy: behavior of $D_t(R_v^*, \mathbf{R_c^*})$ in function of $R_t = R_v^* + R_c^*$.

are done on the CIF (352x288 pixels) video FOREMAN, the SD (704x576 pixels) video CITY and the HD 720p (1280x720 pixels) video JETS, at 30 fps for the three sequences.

3.3.6.1 Behavior of the bit-rate allocation

In Figure 3.13 is plotted the distortion $D_t(R_v, \mathbf{R_c})$ for different bit-rates (3D mesh plot). The solid line shows the optimal distortions $D_t(R_v^*, \mathbf{R_c^*})$ in function of R_v^* and $\mathbf{R_c^*}$, obtained by the minimization of the functional $J_\lambda(R_v, \mathbf{R_c})$ for various R_t. Figure 3.14 shows the behavior of the convex-hull $D_t(R_v^*, \mathbf{R_v^*})$ in function of $R_t = R_v^* + R_c^*$. These experiments were carried out on the sequence FOREMAN decomposed on two temporal levels and for motion vectors estimated at a quarter-pixel accuracy (the size of the blocks for the block-matching algorithm is 16x16 pixels). The Figure 3.14 shows the behavior of the optimal total distortion obtained with the bit allocation algorithm.

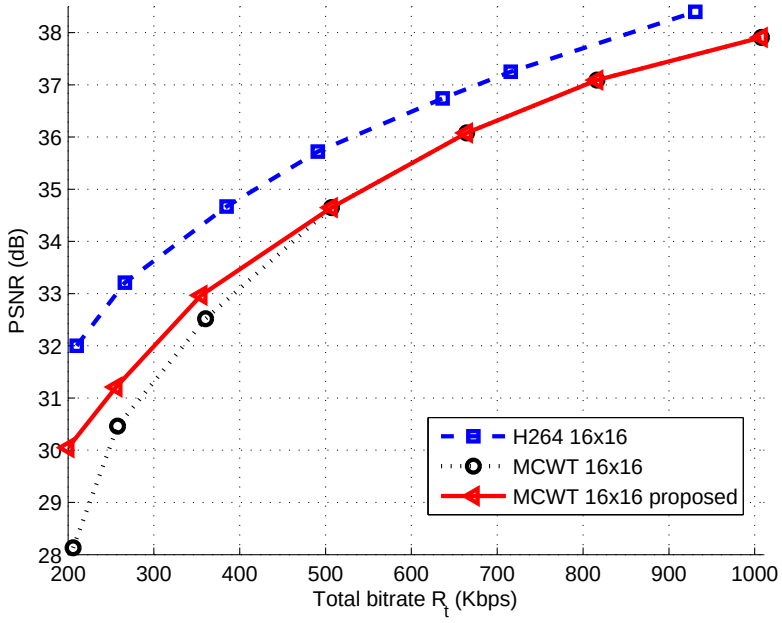
3.3.6.2 Evaluation of the performances

The performances of the proposed coder, in a scalable version, obtained by quantizing the motion vectors and applying the bit allocation algorithm, are compared to those obtained when the motion vectors are coded losslessly. All the experiments are done on three different sequences of different formats, with different motion vectors accuracies and for three (2,0) temporal decomposition levels. The motion vectors accuracy is varying according to the video format. Here, motion vectors with a quarter-pixel accuracy for the CIF sequence FOREMAN, motion vectors with a half-pixel accuracy for the SD sequence CITY and motion vectors with a pixel accuracy for the HD $720p$ sequence JETS have been chosen.

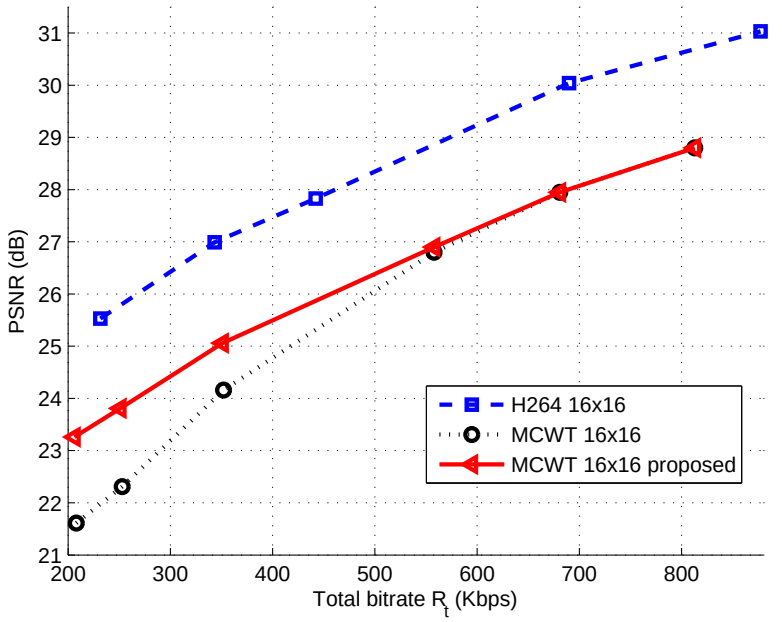
Figures 3.15(a), 3.15(b) and 3.16(a) present the improvement obtained when using the proposed open loop coding of motion vectors and bit-rate allocation (triangular markers). These results are compared to the ones obtained when motion vectors are coded losslessly (circular markers, for same target bit-rates R_t, R_v = 143.4 Kbps for FOREMAN, R_v = 309.4 Kbps for CITY, and R_v = 766.5 Kbps for JETS). The results obtained with the H.264 coder are also presented, with a block size of 16x16 and quarter-pixel accuracy for the motion vectors (square markers). At Figure 3.16(b), results for a block size of 8x8 are presented, for FOREMAN, and for the same three coders (R_v = 430 Kbps). These curves represent the input-output PSNR in dB (computed only on the luminance-component) as a function of the target bit-rate R_t. It appears that using the optimal motion and subbands bit-rates allows to improve the quality of the decoded sequence up to 4 dB for JETS for example. These results show that the proposed approach of optimal bit-rate allocation gives satisfactory results. The performances also get closer to the ones of H.264.

Figure 3.17 presents a visual comparison on three images extracted from the video sequence FOREMAN coded at a total bit-rate R_t = 150 Kbps. For the images on the left hand side, the motion was coded losslessly at R_v =143.4 Kbps and for the images on the right hand side, the motion was coded with losses at R_v^* =48.5 Kbps using the bit-rate allocation. From these results, it is clear that optimizing the rate-distortion trade-off between motion information and wavelet coefficients improve the results of coding/decoding. Indeed, when no allocation is done, blocking effects are very important and color information could be quite inexistent. Furthermore, in some cases (see for example the image 3.17(c)), the loss of information is almost total. This phenomenon is amplified specially at low bit-rates.

Figure 3.18 presents visual results for the sequence JETS. For the images on the left hand side, the vectors are not quantized (motion bit-rate R_v = 766.5 Kbps) and the total bit-rate R_t is equal to 1.3 Mbps. On the right hand side, the bit allocation approach is used (R_v^* = 357.5 Kbps) in order to obtain nearly the same PSNR as for the lossless case; the result-

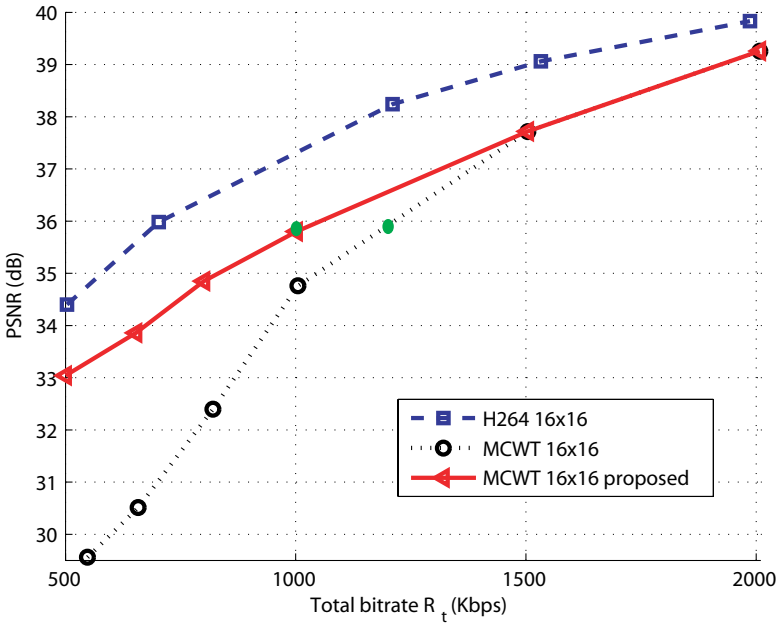


(a) FOREMAN, 16x16.

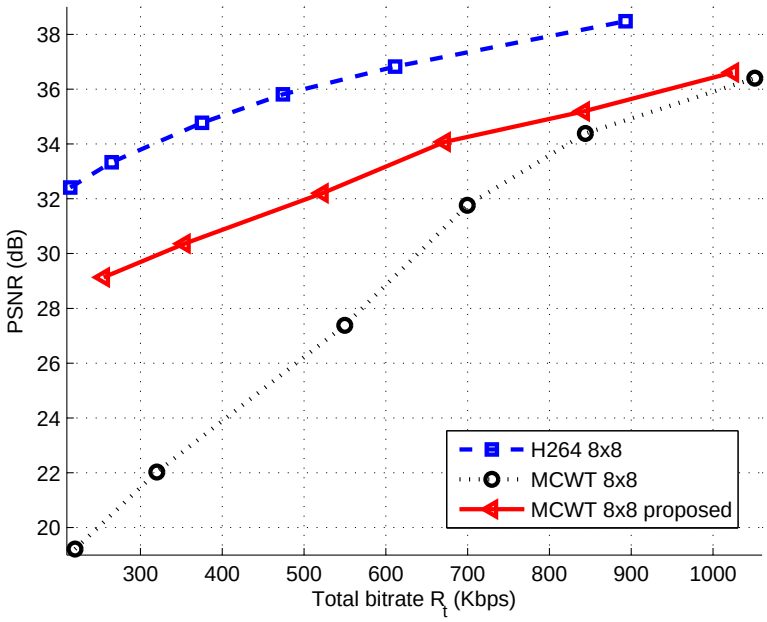


(b) CITY, 16x16.

Figure 3.15: Performance comparison between the proposed approach (triangular markers), the one which consists in coding without losses the motion vectors (circular markers), and the H.264 baseline coder (square markers); block size of 16x16.



(a) JETS, 16x16.



(b) FOREMAN, 8x8.

Figure 3.16: Performance comparison between the proposed approach (triangular markers), the one which consists in coding without losses the motion vectors (circular markers), and the H.264 baseline coder (square markers); for different block sizes.

Figure 3.17: Decoded FOREMAN at 200 Kbps ; images 20, 45 and 54; (a), (c), (e): half-pixel accuracy with vectors coded losslessly ($R_v = 143.4$ Kbps) ; (b), (d), (f): half-pixel accuracy with the bit allocation approach and motion vectors quantized at $R_v^* = 48.5$ Kbps.

ing sequence is encoded at a total bit-rate $R_t = 1$ Mbps (see green circular markers at Figure 3.16(a)). It can be noticed that the quality of the images is slightly the same, but 300 Kbps are saved for the total bit-rate. Consequently, controlling in an optimal way the binary resources between motion information and wavelet coefficients makes it possible to improve the quality of the reconstructed video sequence, even if losses are introduced on the motion vectors. Moreover, it is also possible to decrease the total birate of the coding (and thus to increase the compression ratio), with slightly the same quality at the reconstruction.

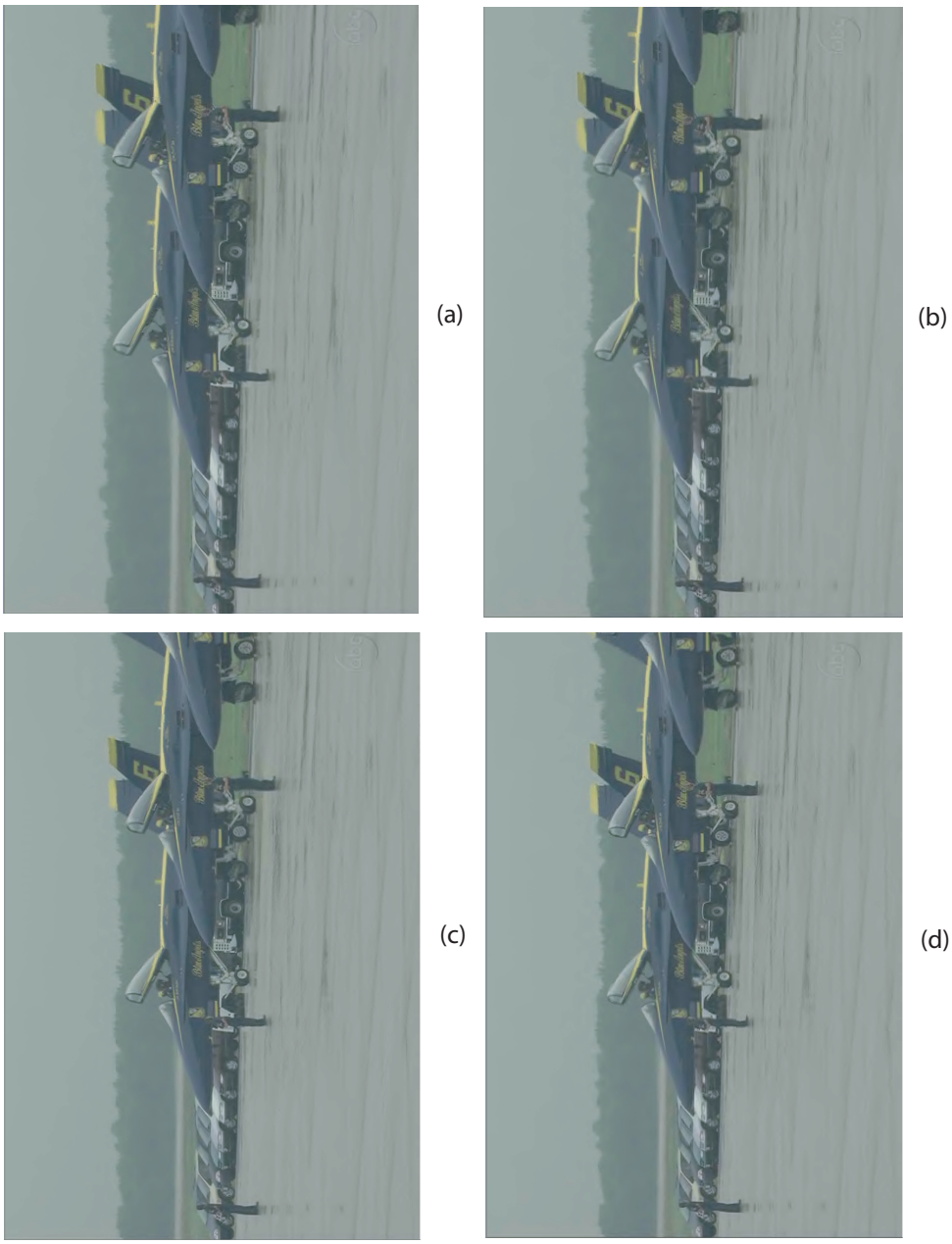


Figure 3.18: Reconstructed video sequences JETS, images 14 and 59. (a), (c), (e): pixel accuracy with vectors coded losslessly, $R_t = 1.3$ Mbps ($R_v = 766.5$ Kbps) ; (b), d), (f): pixel accuracy, $R_t = 1$ Mbps, the bit allocation approach is used and the motion vectors are quantized at $R_v^* = 357.5$ Kbps; PSNR = 35.9 dB for the two experiments (corresponding to the green circular markers at Figure 3.16(a)).

3.4 Motion-adapted weighted lifting scheme

In order to increase the performances of the wavelet-based video coder, the influence of some badly estimated motion vectors on the motion-compensated wavelet transform has to be minimized. A novel and adaptive method for the implementation of the lifting scheme is thus proposed.

3.4.1 Problem statement

The inclusion of motion compensation in the lifting steps has been shown to improve the efficiency of the temporal subband decomposition. While the application of a lifting scheme instead of a "classical" filtering implementation (obtained by convolution) reduces the computational complexity, the inclusion of motion compensation facilitates temporal subband decomposition along motion trajectories. This reduces the wavelet coefficient energy in both subbands leading to a more efficient compression.

Nevertheless, there is still room for improvement. Some works have been done to reduce the ghosting artefacts where the motion estimation have failed. Pesquet-Popescu et al. [PPB01] proposed a method which uses energy in the high-pass subband to improve the motion estimation/compensation process in the lifting based implementation of temporal Haar wavelet transform. Trappe et al. [TZL99] also proposed an approach which modifies the predictor's performances of the wavelet transform at scene changes. Due to their dependence on predictive feedback, this method does not deal with scalability. Mehrseresht et al. presented in [MT03] a method for reducing artefacts in the low-pass temporal frames: they adaptively weight the lifting update steps according to the energy in the corresponding high-pass temporal subband.

An improvement of the lifting scheme by closely adapting the lifting steps to the motion is performed, in the framework of scalable MCWT. Assuming that a vector with a high norm should correspond to a rapid motion or a badly estimated motion, the lifting coefficients should be changed to take into account the effectiveness of the motion estimation. Indeed, a badly estimated motion could generate wavelet coefficients of high energy (expensive coding cost). To this end, the original mother scaling function is sampled at sampling points computed using a criterion based on the norm of the motion vectors, leading to an irregular sampling (the sampling index $n \in \mathbb{R}$). This method allows to decrease the influence of a badly estimated motion on the wavelet subbands, by minimizing the value of the filter in this case. As all the useful data to compute the norms at the decoder side are the motion vectors, which are coded and included in the bitstream, no side information is needed for decoding. Thus, no bit waste is carried out.

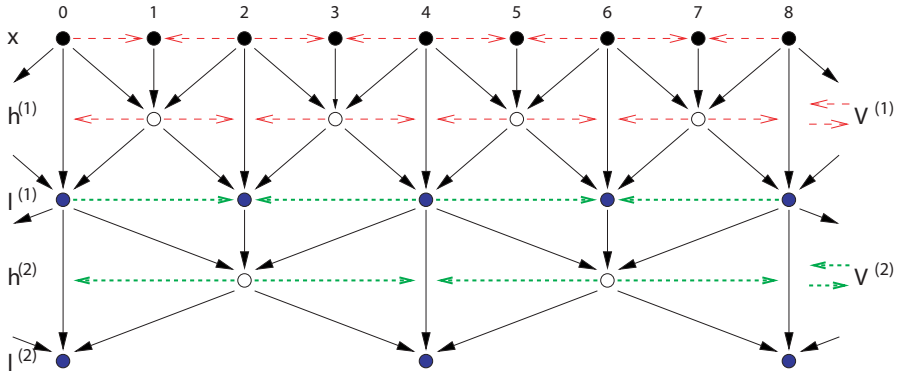


Figure 3.19: The (2,2) analysis lifting scheme for two levels of temporal wavelet decomposition.

3.4.2 (2,2) Motion-Compensated lifting scheme

Let's first recall the expressions of the (2,2) lifting scheme, and then present the link with the polyphase matrix.

3.4.2.1 Equations of the (2,2) lifting scheme

Let $(x_n)_K$ be a K-images sequence. Let denote by $v_{i+j\to i}(\mathbf{p})$ a motion vector of pixel at spatial location $\mathbf{p}$ in frame $i+j$ that displaces this pixel to new location $\mathbf{p}+v_{i+j\to i}(\mathbf{p})$ in frame i. Estimation of this vector is usually based on the assumption of constant image intensity along motion trajectory, i.e. $x_i(\mathbf{p}+v_{i+j\to i}(\mathbf{p}))\approx x_{i+j}(\mathbf{p})$. The motion vector $v_{i+j\to i}(\mathbf{p})$ is a forward (respectively backward) motion vector if j is negative (respectively positive). Thus, $x_i(\mathbf{p}+v_{i+j\to i}(\mathbf{p}))$ is the motion-compensated prediction of x_{i+j} using image x_i as reference. With these notations, the equations of the analysis motion-compensated (2,2) lifting scheme, which give the high-pass and low-pass subbands, respectively $h_k(\mathbf{p})$ and $l_k(\mathbf{p})$, are the following:

$$\begin{cases} h_k(\mathbf{p}) & = x_{2k+1}(\mathbf{p}) - \frac{1}{2}\Big(x_{2k}\big(\mathbf{p}+v_{2k+1\to 2k}(\mathbf{p})\big) \\ & \quad +x_{2k+2}\big(\mathbf{p}+v_{2k+1\to 2k+2}(\mathbf{p})\big)\Big) \\ l_k(\mathbf{p}) & = x_{2k}(\mathbf{p}) + \frac{1}{4}\Big(h_{k-1}\big(\mathbf{p}+v_{2k\to 2k-1}(\mathbf{p})\big) \\ & \quad +h_k\big(\mathbf{p}+v_{2k\to 2k+1}(\mathbf{p})\big)\Big) \end{cases}$$

Figure 3.19 shows the (2,2) analysis motion-compensated lifting scheme on two wavelet temporal decomposition levels. The synthesis equations are

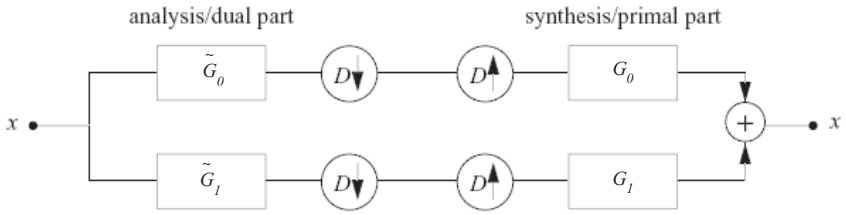


Figure 3.20: General filter bank.

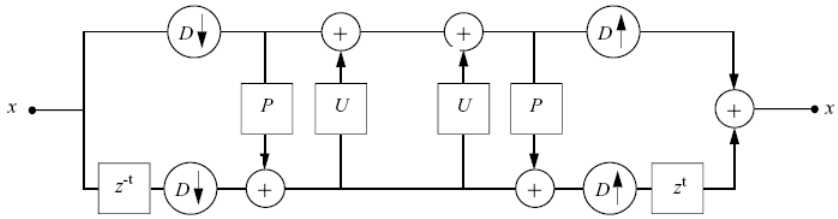


Figure 3.21: The corresponding lifting scheme.

given by:

$$\begin{cases} x_{2k}(\mathbf{p}) & = l_k(\mathbf{p}) - \frac{1}{4}\Big(h_{k-1}\big(\mathbf{p} + v_{2k\to 2k-1}(\mathbf{p})\big) \\ & \quad + h_k\big(\mathbf{p} + v_{2k\to 2k+1}(\mathbf{p})\big)\Big) \\ x_{2k+1}(\mathbf{p}) & = h_k(\mathbf{p}) + \frac{1}{2}\Big(x_{2k}\big(\mathbf{p} + v_{2k+1\to 2k}(\mathbf{p})\big) \\ & \quad + x_{2k+2}\big(\mathbf{p} + v_{2k+1\to 2k+2}(\mathbf{p})\big)\Big) \end{cases}$$

3.4.2.2 Link with the polyphase matrix

For each implementation of a filter, it exists an equivalent lifting-based implementation [DS98]. Let define a filter g_0 as:

$$g_0(z) = h_e\left(z^2\right) + z^{-1}h_o\left(z^2\right),$$

with

$$h_e(z) = \sum_k g_{02k}z^{-k},$$

the filter of even coefficients, and

$$h_o(z) = \sum_k g_{02k+1}z^{-k},$$

the filter of odd coefficients. Then, the synthesis polyphase matrix P of Figure 3.20 can be written as:

$$P(z) = \begin{pmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{pmatrix}.$$

The lifting scheme (2,2) corresponds to the biorthogonal 5/3 filter bank. In this case, the low pass and high pass analysis filters are:

$$\widetilde{g}_0(z) = -\frac{1}{8}z^{-2} + \frac{1}{4}z^{-1} + \frac{3}{4} + \frac{1}{4}z - \frac{1}{8}z^2$$

and

$$\widetilde{g}_1(z) = -\frac{1}{2}z^{-2} + z^{-1} - \frac{1}{2}.$$

The analysis high pass filter is the interpolating 3-taps filter. Then, by construction, the lowpass synthesis filter is also interpolating, i.e., $g_0(2k) = \delta_k \forall k$ (this means that the filter is zero in all even location except 0). In that case, the synthesis polyphase matrix can be re-written as [DS98]:

$$P(z) = \begin{pmatrix} 1 & g_e(z) \\ h_o(z) & 1 + h_o(z)g_e(z) \end{pmatrix}.$$

This matrix can be also written in terms of $U(z)$ and $P(z)$ [KS00], which gives (according to the notations - see Figure 3.21):

$$P(z) = \begin{pmatrix} 1 & U(z) \\ P(z) & 1 + U(z)P(z) \end{pmatrix}.$$

The $U(z)$ and $P(z)$ operators can easily be identified using the knowledge of the synthesis filters g_0 and g_1.

Furthermore, the analysis polyphase matrix can be found using the perfect reconstruction equation [DS98]:

$$P(z)\widetilde{P}(z^{-1})^T = I, \tag{3.18}$$

and the Cramer's rule:

$$adj\,(A)\ A = A\,adj(A) = \det(A)I, \tag{3.19}$$

where $adj\,(A)$ is the adjugate of A (the transpose of the "cofactor matrix" of A). Let suppose that $P(z)$ is known, the equation (3.19) involves:

$$P(z)\,adj\,(P(z)) = I,$$

since $\det(P(z)) = 1$. From equations (3.18) and (3.19) the matrix $\widetilde{P}(z^{-1})^T$ can be identified as:

$$\begin{aligned} \widetilde{P}(z^{-1})^T &= adj\,(P(z)) \\ &= \begin{pmatrix} 1 + U(z)P(z) & -U(z) \\ -P(z) & 1 \end{pmatrix}. \end{aligned}$$

Then, the update $U(z^{-1})$ and prediction $P(z^{-1})$ operators, which give the analysis equations of the lifting scheme, are given by:

$$\begin{aligned}\widetilde{P}(z) &= \begin{pmatrix} 1+U(z^{-1})P(z^{-1}) & -P(z^{-1}) \\ -U(z^{-1}) & 1 \end{pmatrix} \\ &= \begin{pmatrix} \widetilde{h}_e(z) & \widetilde{g}_e(z) \\ \widetilde{h}_o(z) & \widetilde{g}_o(z) \end{pmatrix} \\ &= \begin{pmatrix} \widetilde{h}_e(z) & \widetilde{g}_e(z) \\ \widetilde{h}_o(z) & 1 \end{pmatrix}. \end{aligned} \tag{3.20}$$

It is then easy from the knowledge of the coefficients of the filter bank to obtain the prediction and update steps of the lifting scheme, using the analysis polyphase matrix, or, in an equivalent way, the synthesis polyphase matrix, as it will be seen in the following.

3.4.3 Motion-adapted lifting scheme

In this section is presented the method for adapting the lifting steps to the motion, and it is applied to the (2,2) and (2,0) lifting schemes.

3.4.3.1 Adapting the lifting steps to the motion

The main objective is to compute the filter coefficients associated to the scaling function by taking into account the motion. By definition, the coefficients of the low-pass analysis filter $\widetilde{g}_0$ are given by ([ABMD92]):

$$\widetilde{g}_0(n) = \int_{-\infty}^{+\infty} 2^{-\frac{1}{2}} \phi(\frac{1}{2}x)\widetilde{\phi}(x-n)dx, \tag{3.21}$$

and equivalently, those of the low-pass synthesis filter g_0 by:

$$g_0(n) = \int_{-\infty}^{+\infty} 2^{-\frac{1}{2}} \widetilde{\phi}(\frac{1}{2}x)\phi(x-n)dx, \tag{3.22}$$

where $n \in \mathbb{Z}$ are the values on which the filters are sampled, and ϕ is the mother scaling function. Here, a method to adapt the sampling index n to the norm of the motion is proposed.

Let define a motion vector v by $v = (v_x, v_y, v_z)$. Assuming that $v_z = 1$ (v_z represents the time component), the norm of the vector v is given by:

$$||v|| = \sqrt{v_x^2 + v_y^2 + 1}$$

Let denote by $||\overline{v}||$ the average norm of the motion vectors and by n the original sampling index. Remember that it is assumed that a vector with a high norm should correspond to a rapid motion or a badly estimated motion.

Then, in this case, the coefficient of the filters should be small such that the motion has a small impact on the filtering result. On the contrary (small norm), the coefficients of the filters should be sufficiently high to take into account the effectiveness of the motion estimation. Finally, if the norm is equal to $||\overline{v}||$, classical coefficients are used instead. This can easily be done by taking into account the characteristics of the mother scaling function and by sampling this function according to the motion vector norms, the new sampling index n' thus becomes:

$$n' = \begin{cases} n + \frac{||v|| - ||\overline{v}||}{||v_{\max}|| - ||\overline{v}||} & \text{if} \quad ||v|| > ||\overline{v}|| \\ n - \frac{||v|| - ||\overline{v}||}{||v_{\min}|| - ||\overline{v}||} & \text{if} \quad ||v|| < ||\overline{v}|| \\ n & \text{if} \quad ||v|| = ||\overline{v}|| \end{cases} .$$

Note that this new sampling is irregular and is signal-adapted. But, as the motion vectors are entirely transmitted, the norm is easily computable at the decoder side and there is no bit waste. This approach also allows to preserve the scalability.

Then, the computation of the new coefficients of filters $\widetilde{g}_0$ and g_0 is done with the help of equations (3.21) and (3.22) at sampling values n'. On the other hand, the new coefficients of $\widetilde{g}_1$ are then computed thanks to the following relation:

$$\widetilde{g}_1(n') = (-1)^{n'} g_0(-n' + 1).$$

Moreover, the $\widetilde{g}_0$ and $\widetilde{g}_1$ coefficients have to satisfy the following conditions of normalization, which must be taken into account for the final computation of the new lifting equations:

$$\sum_{n'} \widetilde{g}_0(n') = 1 \text{ and } \sum_{n'} \widetilde{g}_1(n') = 0.$$

Finally, the identification of the two polyphase matrices given in equation (3.20) permits to obtain the new values of the lifting operators P and U adapted to the motion vector norms.

3.4.3.2 Case of the (2,2) lifting scheme

Let denote by w_{g_b} and w_{g_f} the new lifting coefficients (respectively "backward" and "forward") corresponding to the filter $\widetilde{g}_1$; in the same way, w_{h_b} and w_{h_f} are the new lifting coefficients corresponding to $\widetilde{g}_0$. One can also set

$$w_g = w_{g_b} + w_{g_f},$$

and

$$w_h = w_{h_b} + w_{h_f}.$$

The equations of the analysis motion-compensated and motion-adapted (2,2) lifting scheme become:

$$\begin{cases} h_k(\mathbf{p}) = w_g \cdot x_{2k+1}(\mathbf{p}) - \Big(w_{g_b} \cdot x_{2k}\big(\mathbf{p} + v_{2k+1\to 2k}(\mathbf{p})\big) \\ +w_{g_f} \cdot x_{2k+2}\big(\mathbf{p} + v_{2k+1\to 2k+2}\,(\mathbf{p})\,\big)\Big) \\ l_k\,(\mathbf{p}) = x_{2k}\,(\mathbf{p}) + \frac{1}{2\cdot w_h \cdot w_g}\Big(w_{h_b} \cdot h_{k-1}\big(\mathbf{p} + v_{2k\to 2k-1}(\mathbf{p})\big) \\ +w_{h_f} \cdot h_k\big(\mathbf{p} + v_{2k\to 2k+1}\,(\mathbf{p})\,\big)\Big) \end{cases}$$

Reversing these equations permits to obtain the synthesis equations with motion-adapted lifting steps:

$$\begin{cases} x_{2k}\,(\mathbf{p}) = l_k\,(\mathbf{p}) - \frac{1}{2\cdot w_h \cdot w_g}\Big(w_{h_b} \cdot h_{k-1}\big(\mathbf{p} + v_{2k\to 2k-1}(\mathbf{p})\big) \\ +w_{h_f} \cdot h_k\big(\mathbf{p} + v_{2k\to 2k+1}\,(\mathbf{p})\,\big)\Big) \\ x_{2k+1}\,(\mathbf{p}) = \frac{h_k(\mathbf{p})}{w_g} + \frac{1}{w_g}\Big(w_{g_b} \cdot x_{2k}\big(\mathbf{p} + v_{2k+1\to 2k}(\mathbf{p})\big) \\ +w_{g_f} \cdot x_{2k+2}\big(\mathbf{p} + v_{2k+1\to 2k+2}\,(\mathbf{p})\,\big)\Big) \end{cases}$$

These equations give perfect reconstruction at the decoder side, since it is a property of the lifting scheme.

3.4.4 Case of the (2,0) lifting scheme

The original equations of the (2,0) lifting scheme can be found in Section 3.3.1.2 (see also Figure 3.9 in this same section).

With the same ponderations as previously, the equations of the analysis motion-compensated and motion-adapted (2,0) lifting scheme are:

$$\begin{cases} h_k(\mathbf{p}) = w_g.x_{2k+1}(\mathbf{p}) - \Big(\widetilde{g}_b.x_{2k}\big(\mathbf{p} + v_{2k+1\to 2k}(\mathbf{p})\big) \\ +\widetilde{g}_f.x_{2k+2}\big(\mathbf{p} + v_{2k+1\to 2k+2}\,(\mathbf{p})\,\big)\Big) \\ l_k\,(\mathbf{p}) = x_{2k}\,(\mathbf{p}) \end{cases}$$

And the synthesis equations:

$$\begin{cases} x_{2k}\,(\mathbf{p}) = l_k\,(\mathbf{p}) \\ x_{2k+1}\,(\mathbf{p}) = \frac{h_k(\mathbf{p})}{w_g} + \frac{1}{w_g}\Big(\widetilde{g}_b.x_{2k}\big(\mathbf{p} + v_{2k+1\to 2k}(\mathbf{p})\big) \\ +\widetilde{g}_f.x_{2k+2}\big(\mathbf{p} + v_{2k+1\to 2k+2}\,(\mathbf{p})\,\big)\Big) \end{cases}$$

Foreman (Kbps)	**2000**	**1500**	**1000**	**500**	**300**	**240**
MC (2,2) lifting	40.64	39.24	36.98	32.67	24.94	19.13
Motion-adapted lifting steps	40.97	39.37	37.23	32.75	25.16	19.25
MC (2,0) lifting	40.75	39.49	37.09	33.54	30.93	29.72
Motion-adapted lifting steps	40.99	39.82	37.31	33.65	31.15	29.89
City (Kbps)	**4500**	**4000**	**3500**	**3000**	**2500**	**2000**
MC (2,2) lifting	35.64	35.02	34.06	33.39	32.32	30.89
Motion-adapted lifting steps	35.87	35.26	34.25	33.53	32.47	31
MC (2,0) lifting	35.62	35.04	34.06	33.39	32.36	30.99
Motion-adapted lifting steps	35.88	35.24	34.26	33.56	32.51	31.11
Jets (Kbps)	**6000**	**5500**	**4500**	**3500**	**2700**	**2000**
MC (2,2) lifting	41.74	41.49	41.15	40.21	38.32	34.33
Motion-adapted lifting steps	41.92	41.66	41.35	40.3	38.48	34.51
MC (2,0) lifting	41.76	41.47	41.11	40.23	39.66	38.49
Motion-adapted lifting steps	41.91	41.66	41.27	40.39	39.83	38.65

Table 3.2: PSNR (dB) results for different target bit-rates in Kbps for the sequences FOREMAN, CITY (both on three decomposition levels, with quarter-pixel motion vectors) and JETS (two levels, half-pixel vectors): comparison between the classical (2,2) and (2,0) motion-compensated (MC) lifting scheme and the (2,2) and (2,0) lifting with motion-adapted lifting steps.

3.4.5 Some results

The experiments have been done with the (2,2) and (2,0) lifting schemes, on three different sequences of different formats and heterogeneous motion types: on CIF sequence FOREMAN, on SD sequence CITY (both with quarter-pixel motion vectors and three levels of temporal decomposition) and on HD *720p* sequence JETS (with half-pixel motion vectors and two levels of temporal decomposition). In table 3.2, PSNR results for these sequences are presenting, at different bit-rates: the first rows present the results with classical motion-compensated lifting schemes and the second rows present the results of the proposed approach (with the lifting equations presented in Section 3.4.3.2 and 3.4.4). In table 3.3, for FOREMAN, are also presented some results using measures of subjective quality: the structural similarity index (SSIM [WBSS04]) and the noise quality measure (NQM [DVKG$^+$00]). The first rows present the results for a classical ap-

SSIM	**2000**	**1500**	**1000**	**500**
MC (2,2) lifting	0.952	0.942	0.929	0.8582
Motion-adapted lifting steps	0.955	0.9435	0.933	0.8593
NQM	**2000**	**1500**	**1000**	**500**
MC (2,2) lifting	31.83	30.66	29.3	22.76
Motion-adapted lifting steps	32.08	30.813	29.48	22.874

Table 3.3: SSIM and NQM (dB) results for different target bit-rates in Kbps for FOREMAN, on three decomposition levels, with quarter-pixel motion vectors: comparison between the classical and the proposed approach.

proach, and the second rows for the proposed approach.

These results show that using motion-adapted lifting steps into a motion-compensated lifting scheme allows to increase the performances of the video coder up to 0.3 dB. This approach also increases the subjective quality of the decoded sequences: the two measures of subjective quality give better results.

3.5 Conclusion

It is a matter of fact that it is necessary to globally optimize the rate-distortion trade-off between motion information and wavelet coefficients in MCWT video coders. To this end, an approach to introduce losses on motion vectors estimated with a high sub-pixel accuracy has been proposed, in order to perform a bit-rate allocation to optimally distribute binary resources between motion information and wavelet subbands. The proposed approach of lossy motion coding uses an uniform scalar quantizer and an encoding performed in open loop, and has been presented in [AAAB05b, AAAB05a]. This method has been applied to the standard video coder H.264, as described in the following chapter.

To evaluate the impact of the losses introduced when quantizing both the motion information and the wavelet coefficients, a theoretical input/output distortion model including the motion [AAAB06] and the subbands coding errors [AA06] has been developed. This model has been derived for several temporal decomposition levels and validated on several video sequences. Experimental validation of the proposed model has provided very good results. Furthermore, this model has permitted to derive an efficient model-based bit-rate allocation algorithm to dispatch the binary resources between motion vectors and wavelet coefficients, which has been presented in [AAB06, AAB07]. The proposed model-based approach allows to find

analytically for a given target bit-rate, the optimal rates for the motion vectors and the wavelet coefficients in order to have a minimal input/output distortion at decoding. The accuracy of the motion vectors is thus variable. This approach decreases the computational complexity and the cost of the coding and improves the coder performances. Experimental results on CIF, SD and HD sequences show a significative improvement in term of PSNR for the decoded video compared to a standard approach without optimal bit allocation between motion information and wavelet subbands.

A motion-adapted lifting scheme has also been introduced in the MCWT video coder [AA07]. This new approach allows to increase the performances of the coder, while preserving the scalability, and especially to avoid some wrong effects due to a bad motion estimation. To this purpose, the lifting steps have been closely adapted to the norm of the motion. The scaling function is sampled in an irregular way according to the value of this norm, and the new weighted lifting steps are computed. This approach does not introduce a bit waste, since all the useful information (motion vectors) is already transmitted at the decoder side, and obviously allows to improve the whole coder performances.

Chapter 4

Application to a hybrid coder: H.264

In the framework of an industrial contract with Orange labs, the lossy coding approach of motion vectors presented in Section 3.2 has been applied to the standard video coder H.264. The effectiveness of a new *coding mode*, based thus on the *quantization of motion vectors* (QMV) is studied. This new coding mode is introduced in an H.264 [WSBL03] implementation called JM [H26] (version 11.0 KTA 1.4). It is derived for the different partitions of H.264, and brings some theoretical issues.

4.1 A New Coding Mode

After a brief presentation of the goal of this study and a brief description of the new mode, the cost function of the new coding mode is derived. For precisions on the H.264 architecture, the reader can refer to Section 2.1.2.1.

4.1.1 Problem statement and motivations

The goal here is to introduce a new motion compensated mode in the framework of rate-distortion optimized video coding, by applying the result of Section 3.2, obtained for a wavelet-based coder, to the hybrid coder H.264, whose general structure is presented at the Figure 4.1. In particular, this mode will be inserted into the JM [H26] implementation of the H.264 standard.

Let begin by a little study. In Figure 4.2 are reported the average macroblock (MB) rate and distortion for several coding modes in a H.264 coder, with RD optimization, quarter-pixelic motion estimation, and all typical modes enabled. These operation points have been obtained on the sequence CITY; for other sequences similar results have been obtained. By doing this simple quantitative study, one can see that there is a significant gap between the low-cost, high-distortion SKIP mode and the relatively higher-cost, low-distortion INTER 16x16 mode (while INTRA and lossless IPCM modes are far more expensive and usually not suitable in low bit-rate context). Thus,

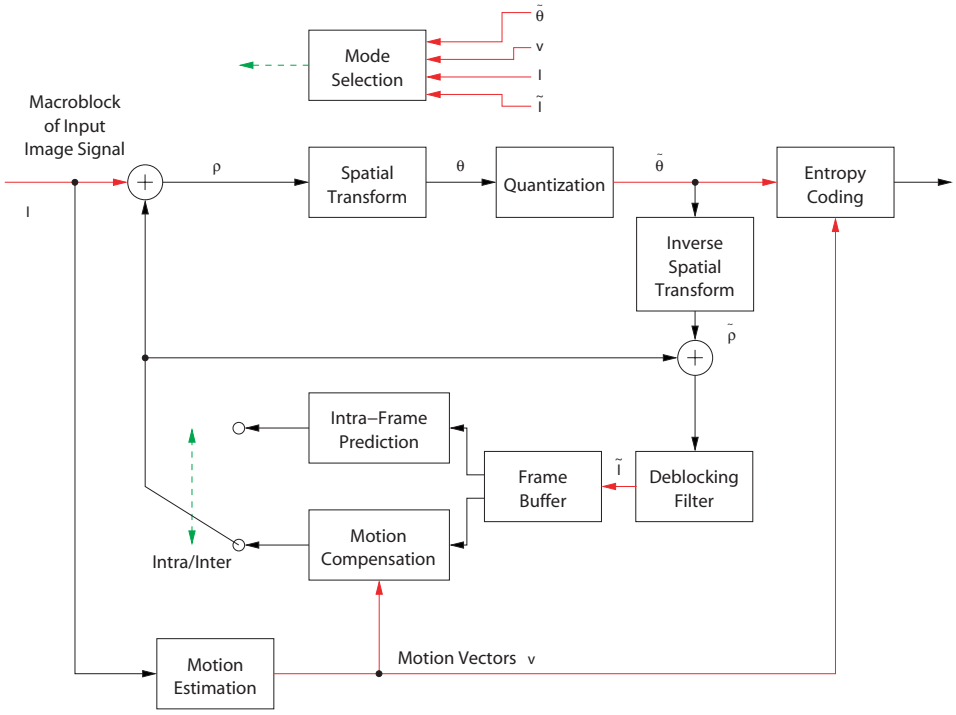


Figure 4.1: A simplified scheme of the H.264 encoder

it could be interesting to introduce a new mode with a behavior that is in some way intermediate between the INTER and the SKIP modes: on a hand, the new mode should have a coding cost lower than the INTER's but higher than the SKIP's; on the other hand it should achieve a distortion definitely smaller than the one of the SKIP mode. In this way, good video quality even at low bit-rates with moderate-to-complex motion content could be achieved.

The key tool to achieve this target is the *lossy coding* of MVs, obtained via quantization. Moreover, this lossy coding is performed in an *open loop system* so that, while the transformed motion-compensated residual is computed with a high-precision motion vector (MV), and sent to the decoder, the MV is quantized before being sent to the decoder. This will reduce the coding rate, but can also increase the distortion as the motion-compensated MB computed from the quantized MVs will be used as MB prediction instead of the original motion-compensated MB. However, as explained in the following of this chapter, the amount of quantization for the MVs is chosen in a rate-distortion optimized way. As the main new tool of this mode is the quantization of MVs, it is called *quantized motion vector* mode (QMV).

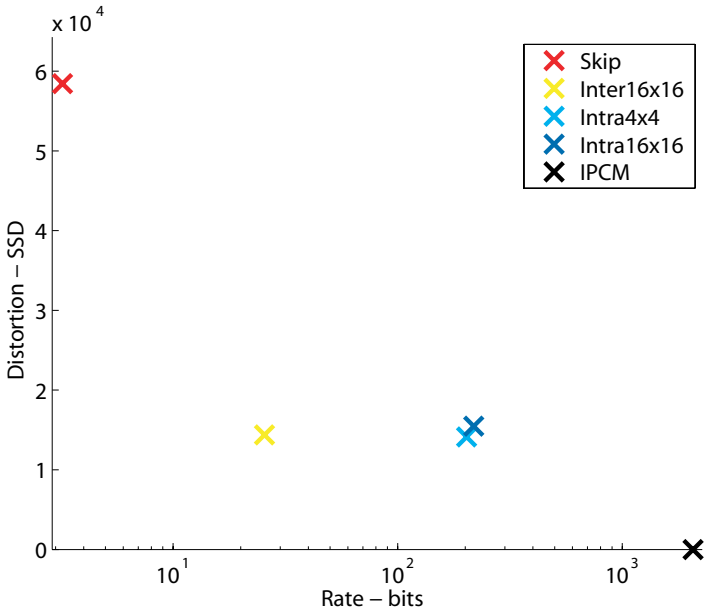


Figure 4.2: Average operation points of H.264 modes, sequence CITY.

4.1.2 General description of the new mode

The new coding mode is quite simple: a relatively accurate (*i.e.* non-quantized) MV is computed by classical motion estimation, and is used in order to compute the motion-compensated residual, which is then transformed, quantized and sent to the output (after entropy coding), like in all hybrid coders. The difference with a standard INTER mode is that the MV is quantized before being sent to the decoder, and thus, at the reconstruction stage, the motion compensated prediction of the current block is *not* obtained using the original vector, but using its de-quantized version. This process amounts to a simple scalar uniform quantization of its components with a quantization step q_v. The problem of efficient selection and encoding of this quantization step is differed in Section 4.2.2. Of course, the encoder must control the distortion caused by this quantization, which is accomplished by computing this distortion at the encoder side, with a process depicted in Figure 4.3.

4.1.3 Notations

Before computing the cost function for the new coding mode, let define some useful notations. In Figure 4.1, I denotes the original current macroblock, $\tilde{I}$ its reconstructed version, ρ the residual of the spatial prediction of I (and $\tilde{\rho}$ its quantized version), θ the transform coefficients, $\tilde{\theta}$ the corresponding quantized version, and $\mathbf{v}$ the MV. $I_{\mathrm{REF}}(\mathbf{v})$, $\rho(\mathbf{v})$ and $\theta(\mathbf{v})$ should denote respectively the motion compensated prediction of the MB, the mo-

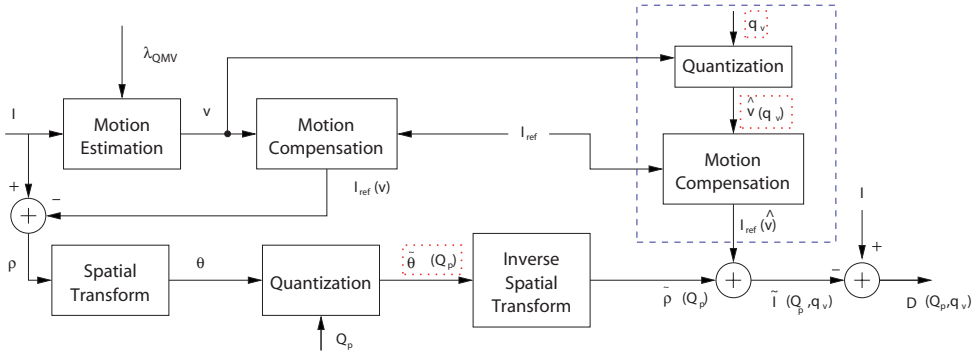


Figure 4.3: The QMV coding mode.

tion compensated residual, and its transform, computed using the MV $\mathbf{v}$. This figure shows a simplified scheme of an H.264 encoder. In red is highlighted the information needed by the mode selection module, which in this simplified scheme switches between INTER and INTRA predictions.

In a cost function computation, the distortion, *i.e.* the p-norm of the error between the original MB and its reconstructed version should be computed:

$$D = \left\| I - \tilde{I} \right\|^p. \tag{4.1}$$

The parameter p is usually equal to 1 or 2.

In details, the motion estimation is performed by using some function of the so called *displaced frame difference* (DFD) as distortion measure, without considering the quantized coefficients of the residual. For each candidate vector $\mathbf{v}$ in the search set V^*, one can have:

$$D_{\mathrm{DFD}}(\mathbf{v}) = \|I - I_{\mathrm{REF}}(\mathbf{v})\|^p.$$

In the following, Q_p and q_v will denote the quantization step of, respectively, the residual and the motion vectors, λ_{mode}, λ_{ME}, and λ_{QMV} the lagrangian parameters, $P[\cdot]$ the (reversible) prediction operator, $T[\cdot]$ the transform operator.

4.1.4 Cost function of the QMV mode

The new coding mode is described by specifying how to compute its cost function, since this stage simulates the operation of the encoder and the decoder. The cost functions for the classical modes of H.264 are presented in Appendix B.

First of all, a motion estimator similar to the one in INTER mode is needed. A "high-precision" vector $\mathbf{v}$ is computed in this stage, according to the following equation:

$$\mathbf{v}(\lambda_{QMV}) = \arg\min_{\mathbf{v}\in V^*} D_{\mathrm{DFD}}(\mathbf{v}) + \lambda_{QMV} R(\mathbf{v}).$$

This equation differs from (B.5) of Appendix B in:

- the search set V^* can be in principle finer that the one used in (B.5); for example $1/16th$ pixel or even finer precision can be used in principle;
- the lagrangian parameter λ_{QMV} has not necessarily the same value as λ_{ME}

These differences derive from the fact that in principle the vector $\mathbf{v}_{\text{OL}}$ has to have the best possible precision (*i.e.* have to minimize the distortion D_{DFD}), since the optimization between the cost and the distortion provided by the vector is not obtained in the ME stage, but in the reconstruction stage by quantization of the MVs.

However, for a preliminary implementation, it is reasonable to neglect these differences. In other words, in a first moment the same search set as for the usual mode can be used, or maybe just extend it to the eighth-pixel precision. The λ_{QMV} parameter can be set to 0 or to λ_{ME}, as discussed in Section 4.2.3.

Once the vector $\mathbf{v}$ has been obtained, it is used to compute the motion compensated residual. With a notation similar to the one used for the INTER mode in Appendix B, one can have:

$$\begin{aligned}\rho_{QMV}(\mathbf{v}) &= I - I_{\text{REF}}(\mathbf{v}) \\ \theta_{QMV}(\mathbf{v}) &= T\left[\rho_{QMV}(\mathbf{v})\right] \\ \tilde{\theta}_{QMV}(Q_p) &= \text{round}\left(\frac{\theta_{QMV}(\mathbf{v})}{Q_p}\right).\end{aligned}$$

The residual $\tilde{\theta}_{QMV}$ is sent to the decoder, so it must be considered in order to compute the coding rate and the reconstruction distortion. Then, it is reconstructed by:

$$\tilde{\rho}_{QMV}(Q_p) = T^{-1}\left[Q_p.\tilde{\rho}_{QMV}\right].$$

The quantization of $\mathbf{v}$ is performed as scalar quantization of its components, even though a vector quantization could be envisaged. The following development is made for an assigned value of the quantization step q_v. The quantized and the motion compensated prediction are computed as:

$$\begin{aligned}\widehat{\mathbf{v}}(q_v) &= \text{round}\left(\frac{\mathbf{v}}{q_v}\right), \\ \tilde{I}(Q_p, q_v) &= I_{\text{REF}}(q_v \cdot \widehat{\mathbf{v}}(q_v)) + \tilde{\rho}_{QMV}(Q_p).\end{aligned}$$

The resulting distortion is thus:

$$\begin{aligned}D(Q_p, q_v) &= \left\| I - \tilde{I}(Q_p, q_v) \right\|^p \\ &= \|I - I_{\text{REF}}(q_v \cdot \widehat{\mathbf{v}}(q_v)) - \tilde{\rho}_{QMV}(Q_p)\|^p \\ &= \|\rho(\widehat{\mathbf{v}}(q_v)) - \tilde{\rho}_{QMV}(Q_p)\|^p.\end{aligned} \tag{4.2}$$

As far as the rate is concerned, the techniques proposed for encoding the quantized vectors are described in Section 4.2.1, and how to select and encode the quantization step q_v is presented in Section 4.2.2; here it will suffice to write down:

$$R(Q_p, q_v) = R[\tilde{\theta}_{QMV}(Q_p)] + R[\widehat{\mathbf{v}}(q_v)] + R_{\text{mode}}.$$

The cost function for the QMV mode is:

$$J_{\mathsf{QMV}}(Q_p, q_v, \lambda_{\text{mode}}) = D(Q_p, q_v) + \lambda_{\text{mode}} R(Q_p, q_v), \tag{4.3}$$

with the distortion and rate previously defined. For some assigned q_v, Q_p and λ_{mode}, one can compute the cost function for the new mode. This value should be compared with the cost function of the other modes. Obviously, the QMV mode will be selected if its cost function is less than the other.

Figure 4.3 summarizes also the mode selection procedure for the QMV case. This is to be compared to the scheme for the INTER mode, shown in Figure B.1. In particular, the motion estimation is the same, but the lagrangian parameter and the search set can possibly change. The computation of the motion compensated residual is very similar to the INTER case, but for a possible motion compensation with a MV at arbitrary precision. The main difference is about the motion compensated prediction, which is computed with a set of quantized vectors. This new part of the scheme is highlighted in blue. As in Figure B.1, the rate estimation module is not reported explicitly, but its inputs are highlighted with a red box: one can remark that the estimated MV $\mathbf{v}$ is not sent to the encoder, while its quantized (and hopefully cheaper in terms of coding resource) version $\widehat{\mathbf{v}}$ is used for the rate computation, together with the quantized transform of the residual, $\tilde{\theta}_{QMV}$, and the information about the selected quantization step.

4.2 Theoretical issues

Several theoretical issues need to be dealt with, in order to achieve a relevant overall performance improvement when the new mode is introduced. In this section are listed some critical points and the envisaged solutions. However, the reader has to have clear in mind that more efficient strategies can be conceived, only after that a simplified implementation of the new mode will be included into the video codec.

The main theoretical issues concern:

1. the encoding of quantized MVs;
2. the selection and the encoding of the quantization step q_v;
3. the open loop motion estimation parameters;
4. the extension to other modes than 16x16.

These issues are discussed in the following subsections.

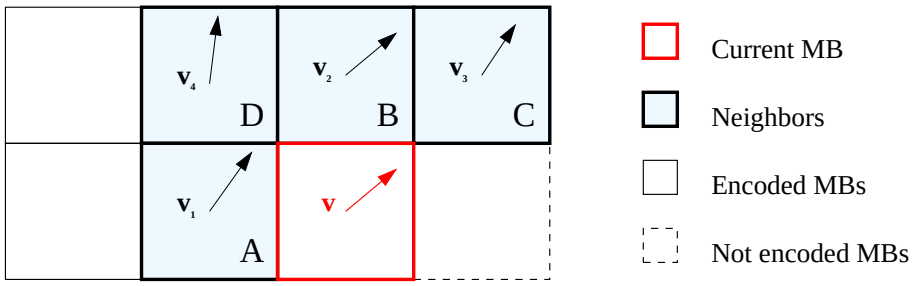


Figure 4.4: Neighborhood used for coding MVs in the QMV mode.

4.2.1 *Coding of quantized motion vectors*

The coding of the quantized MVs is a very important problem in this study. In fact, it is well known that the existing motion-compensated modes in H.264 perform a very efficient MV coding. Nevertheless, at low bit-rates, and for complex-motion sequences, the motion information constitutes a large part of the total bit-rate. The new mode has been introduced with the target of a substantial reduction of the motion information rate, without affecting too much the distortion. Hopefully, the proposed mode should allow a finer choice of the coding strategy, exploring the intermediate solutions between the two "extreme points" represented by the INTER modes (relatively high cost of motion information, low distortion) and the SKIP mode (very low cost of motion information but relatively high distortion). The new QMV mode can be seen as a generalized mode which can be particularized as INTER if q_v is set to its minimum value (*i.e.* the ME precision). On the other hand the SKIP mode is not perfectly equivalent to a QMV mode with $q_v = \inf$, since in the SKIP mode $Q_p = \inf$ as well, while this is not true for the QMV mode. At this end, it is very important to have en effective strategy for MV coding. The objective is to achieve an efficiency comparable to the one of standard modes.

In order to gain some insight on the MV coding problem, the coding technique used in H.264 is briefly described. In this standard, the MVs are predicted and the prediction error is encoded with the entropy coder, CAVLC or the more efficient CABAC (used here, see Section 2.1.2.2 for more details). The predictor is based on a suitable neighborhood, shown in Figure 4.4. Each component of $\mathbf{v}$ is predicted as the median among the same component of the vector in the neighborhood. However, the neighborhood can change according to the availability of MVs in adjacent MBs. An adjacent MB cannot be provided with MV if it is INTRA-coded, or if the current MB is near the image borders. In particular the main rules are the following:

- if all the vectors in A, B and C are available, all of them constitute the neighborhood;

- if the MV in C is not available, D's vector is used instead of it;
- if in B and C there are not available vectors, only A's vector is used;
- if none of these rules applies, the prediction is the null vector.

When this coding technique has to be extended to the case where the QMV are included, two problems arise:

1. how to code the vectors for the QMV mode;
2. how to code INTER MVs when some of the neighbors are QMV.

Let consider the first case. One has to code:

$$\widehat{\mathbf{v}} = \text{round}\left(\frac{\mathbf{v}}{q_v}\right),$$

which is the index corresponding to the de-quantized vector $(q_v \cdot \widehat{\mathbf{v}})$. Then the neighborhood is considered according to the usual rules for H.264. The vector prediction is computed using the *de-quantized* values for all the available vectors (QMV MB or ordinary INTER MB). In facts, INTER vectors can be seen as motion vectors quantized with a q_v equal to the precision (usually, quarter pixel). If $\widehat{\mathbf{v}}_1$, $\widehat{\mathbf{v}}_2$ and $\widehat{\mathbf{v}}_3$ are the de-quantized vectors for MBs A, B and C respectively (supposing that all of them are available), the predictor $\hat{\mathbf{v}}$ of the current vector is defined as:

$$\hat{\mathbf{v}} = \text{median}(\widehat{\mathbf{v}}_1, \widehat{\mathbf{v}}_2, \widehat{\mathbf{v}}_3)$$

Then the vector prediction is quantized using the same step selected for the current MB, and the prediction error $\epsilon(\widehat{\mathbf{v}})$ is sent to the entropic coder:

$$\epsilon(\widehat{\mathbf{v}}) = \text{round}\left(\frac{\mathbf{v}}{q_v}\right) - \text{round}\left(\frac{\hat{\mathbf{v}}}{q_v}\right). \tag{4.4}$$

The encoding strategy will be successful if small values of ϵ are achieved with high probability (or more precisely, if the entropy $H(\epsilon)$ of the prediction error is minimized). Of course, if not all the vectors of MBs A, B and C are available, the neighborhood is formed according to the standard rules of H.264 previously described.

Now the coding of MVs for INTER modes is considered. This technique must be updated because the neighbors of an INTER MB could be QMV. An INTER MB can be considered as a QMV MB with a quantization step equal to the motion estimation resolution. So the same technique used for the QMV MBs can simply be applied, that is prediction from de-quantized vectors and possible re-quantization with the appropriate quantization step. One can explicitly remark that, when none of the neighbors of the current MV is a QMV MB, this technique becomes equivalent to the ordinary MV coding of H.264.

4.2.2 Selection and encoding of the quantization step q_v

The selection and encoding of the quantization step for the current MV are other very important steps in this study, and they are linked problems. The study of these problems is presented in Section 4.2.2. In section 4.2.2.2, a solution to find the range of the quantization steps is described.

4.2.2.1 Quantization step selection strategies

The quantization step should be chosen from a large set of values, and represented finely. However, one cannot spend too many bits to code this information into the bit-stream, since the target of the QMV mode is to reduce the cost of motion information.

Let introduce the formal problem. The function $J_{\mathsf{QMV}}(Q_p, q_v, \lambda_{\text{mode}})$ is defined as in equation (4.3). For the rest of this section, Q_p and λ_{mode} are considered fixed, and are dropped from the expression of J_{QMV}. One would like to find:

$$q_v^* = \arg\min_{q_v \in \mathbb{R}} J_{\mathsf{QMV}}(q_v)$$

and use it as quantization step for the MV. However, if q_v^* is simply varying in $\mathbb{R}$, its coding cost would be too high, and so the QMV mode would not be competitive with classical modes. Note that q_v^* is expected to be found with good precision, by evaluating J_{QMV} in a set of points, called S_Q, and then using some numerical algorithm to find the minimum of the function. In conclusion, once q_v^* has been found, this value must be optimally coded in order to have a competitive coding cost for the mode.

A first solution is to look for the minimum of J_{QMV} for q_v varying in S_Q. In this case the coding cost of the quantization step would be limited to $\approx \log_2 N$, where N is the cardinality of S_Q. However this strategy does not remove all the difficulties: N is a model parameter that should be fixed *a priori*; this can be critical, because if it is too large, it increases too much the coding cost of the mode, while if it is too small, it can be difficult to select good values of q_v. A first solution to find the range of the quantization steps will be presented in Section 4.2.2.2.

More efficient solutions can be envisaged if a *double-pass* coding strategy is allowed. In a first scanning of the current slice, the values of $J^k_{\mathsf{QMV}}(q_v)$, where $k \in \{1, 2, \ldots, K\}$ is the MB index, are gathered. Then one tries to represent in an efficient way the whole vector

$$\mathbf{q}_v^* = \{q_v^*(1), q_v^*(2), \ldots q_v^*(k), \ldots\},$$

where $q_v^*(k)$ is the best step for the k-th MB given by the computation of $J^k_{\mathsf{QMV}}(q_v)$. The advantage is that the coding cost of the vector $\mathbf{q}_v^*$ is shared among all the MB of the slice. More precisely, the signal to be coded is now this vector $\mathbf{q}_v^*$. Several solutions to represent it can be envisaged:

"Oracle" strategy: The encoder uses the optimal vector $\mathbf{q}_v^*$ for the slice, but no bit is accounted for its coding cost. This gives an upper bound of the achievable performance of the QMV mode, and it corresponds to the case of an extremely efficient coding of $\mathbf{q}_v^*$ (or, to the case of an "oracle" decoder, able to know the q_v used for each MB). In other words, in this case one should have $R(q_v) \approx \log_2 |S_Q|$, but in fact $R(q_v) = 0$.

"Minsum" strategy: A single value of q_v is used for the whole slice, namely the one minimizing $\sum_k J^k_{\mathsf{QMV}}(q_v)$. In this way the coding cost of q_v is practically negligible, since it is shared among all the MBs of the slice: $R(q_v) \approx \log_2 |S_Q|/K$.

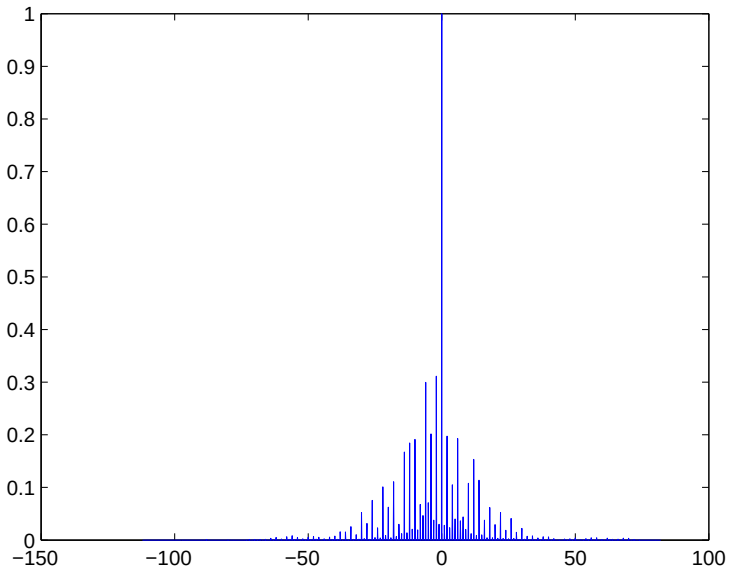
Of course, once q_v has been selected according to one of these strategies, it must be coded, but a fine representation can be used, since its cost is shared by all the K MBs of the slice. Then, in a second pass, the effective quantization of the MVs is done by using the chosen quantization steps, according to the considered strategy.

In any case, before choosing the final strategy to select and encode q_v it seems reasonable to start by implementing a relatively simple solution, as the one that approximates each $q_v^*(k)$ with the value minimizing $\sum_k J^k_{\mathsf{QMV}}(q_v)$. At the same time, all the relevant statistics on J_{QMV} have to be collected. Once these data are known, one can decide if it is worth switching to more complex techniques for representing the quantization steps, as the unconstrained vector quantization, or maybe the trellis coded quantization.

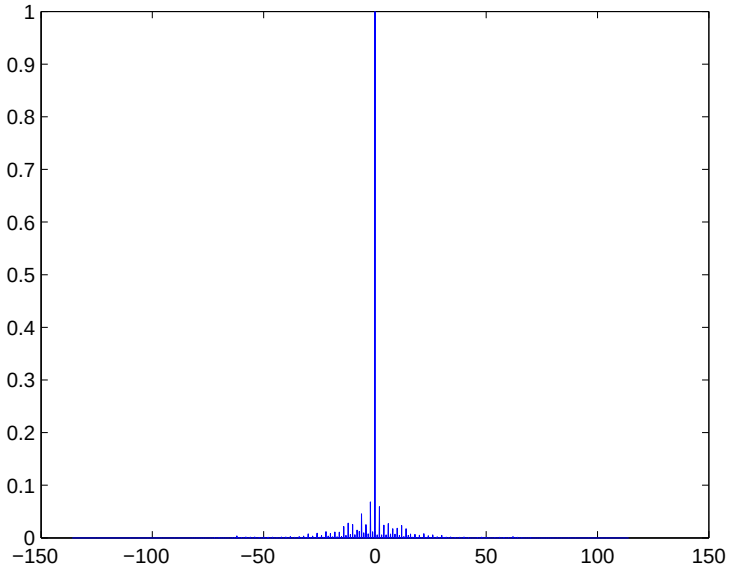
4.2.2.2 A strategy to decide the quantization step range

To further improve the performances of the new mode, the test range of the quantization steps q_v is studied. The process of quantization consists in subdivide a priori the range of variation of a signal in a finite number of intervals (levels), so the choice of the intervals size has a sensible effect. In this case, the signal is represented by the MVs values. The range of assumed values for some video sequences has been studied: the video sequences are coded with H.264 standard, with the different available coding modes. Some curves of the normalized distribution of MVs are reported, for the sequence *city CIF*, coded with the INTER 16x16 mode (Figure 4.5(a)) and the INTER 8x8 mode (Figure 4.5(b)).

Obviously, the most assumed values are around zero, and the dynamic of values changes between the different coding modes, because the resolution of the motion description changes. With the simple description of the INTER 16x16 mode (one MV per MB), big values are possible, and with the more complex description of INTER 8x8 (4 MVs per MB), big values become less probable. Thus, a Lloyd-Max solution based on an analytical model of the MVs distribution could be used to find the optimal range S_Q of the quantization steps q_v. Basing this algorithm on a model of the pdf of MVs



(a) CITY coded with INTER 16x16 mode.



(b) CITY coded with INTER 8x8 mode.

Figure 4.5: Distribution of MVs (component v_x) for the sequence CITY CIF.

will thus make this range be finer on the small values, and coarser on big values of MVs. It will be not necessary to transmit the dictionary: only the parameters of the model will be transmitted, and the dictionary will be re-computed at the decoder side.

For complexity reasons, a first approach, simpler, has been chosen. Here, the maximum variation for a MV has to be evaluated, in order to give more precision to the values around zero, and to quantize the bigger values by zero. The two components, v_x and v_y, are independently coded:

$$\mathbf{v} = (v_x, v_y).$$

One can evaluate a different maximum variation for each component:

$$q_{x_{max}} = 2 \cdot max(|v_{x_{max}}|, |v_{x_{min}}|),$$
$$q_{y_{max}} = 2 \cdot max(|v_{y_{max}}|, |v_{y_{min}}|),$$

and then the bigger one is chosen:

$$q_{v_{max}} = max(q_{x_{max}}, q_{y_{max}}).$$

In conclusion, a range S_Q for the q_v steps can be obtained, variable between 0 and $q_{v_{max}}$.

4.2.3 The open loop motion estimation parameters

Some study will be devoted to the open-loop motion estimation parameters. In principle, the same function as for the ordinary motion estimation in JM can be used. However as observed in Section 4.1, in general the proposed approach could require an estimation with arbitrary precision, and a lagrangian parameter λ_{QMV} not necessarily equal to λ_{ME}.

In any case, a preliminary experimental study is necessary to have a better comprehension of the relationships among the parameters of QMV modes (q_v and λ_{QMV}) and the other optimization related parameters (Q_p, λ_{ME} and λ_{mode}).

Some simple configurations are first tested, defined by the value of λ_{QMV} and by the precision of the ME (*i.e.* the search set V^*)

1. $\lambda_{QMV} = 0$ and the precision is settled to the maximum available value (eighth-pixel in this implementation of JM);

2. $\lambda_{QMV} = 0$ and the precision is settled to the usual value for H.264, *i.e.* the quarter pixel;

3. $\lambda_{QMV} = \lambda_{\text{ME}}$ and the precision is settled to the usual value for H.264, *i.e.* the quarter pixel.

The first configuration should provide the most costly vector, which on the other hand, minimize the DFD energy. This is coherent with the idea that the rate-distortion trade-off MVs should be tuned only by the q_v parameter. The third configuration on the other hand, provide the same vector used for the INTER mode. The rationale behind this approach is to have the INTER mode to appear as a special case of the QMV mode: this would happen when q_v is equal to the ME resolution. The second configuration is halfway between the first and the third. Experimental results obtained with these configurations confirm that the third configuration is the most suitable one.

4.2.4 The extension to other modes

In the JM implementation of H.264, all the motion compensated modes from INTER16x16 to INTER4x4 are available. Notwithstanding an extension of the QMV modes to finer block sizes is conceptually straightforward, the implementation can require a considerable effort, and, above all, analyzing the effect of the insertion of too many new modes at once is for sure quite difficult.

For this reason, the QMV mode has first been implemented for the 16x16 block size, and been inserted in a coder where only the INTER16x16 are allowed among the motion-compensated modes. This initial configuration provided more easily interpretable experimental data. On the basis of these results, the implementation of QMV modes for finer block sizes, like the 8x8, has been proceeded, as described in the following section. In any case one has to keep in mind that the QMV modes should show their benefits above all in the case of smaller blocks, *i.e.* when the coding cost of motion information becomes more important.

4.3 Extension to the QMV 8x8 coding mode

For the QMV 8x8 mode, the 16x16 MBs are split in four sub-blocks 8x8. The MVs can be quantized with different q_v ("Oracle" case), thus the smaller dimension of MBs can handle to a wrong prediction. MVs are predicted from a suitable neighborhood and the prediction error is entropically encoded. The predictor is obtained from the median of the de-quantized vectors for the MBs of neighborhood. Each vector is de-quantized with his optimal step, so if the steps are very different, then the reconstruction levels of the MVs have a different precision, and some disparities in the motion can appear, especially with small dimensions of MBs. More these levels are different, more the error on prediction of the current sub-macroblock will increase because the predictor does not use the original version of MV but the de-quantized version. In the QMV8x8 mode, this perturbation is not negligible, and it could augment when the size of sub-macroblocks decreases.

Potentially, the error on vector prediction could become more significant

than the original vector, so the prediction could not be a convenient strategy. The prediction error can be denoted as:

$$\epsilon(\widehat{\mathbf{v}}) = \widehat{\mathbf{v}} - \tilde{\widehat{\mathbf{v}}},$$

with $\widehat{\mathbf{v}}$ the quantized motion vector, and $\tilde{\widehat{\mathbf{v}}}$ its prediction. The energy of prediction error for a frame can be considered:

$$\sigma_\epsilon^2 = \sum \|\epsilon(\widehat{\mathbf{v}})\|^2,$$

and it is compared with the energy of the quantized vectors for a frame:

$$\sigma_v^2 = \sum \|\widehat{\mathbf{v}}\|^2.$$

The expected result is that residual energy is smaller than vector energy, so transmitting the prediction error is more convenient than transmitting the vectors. But, if $\sigma_\epsilon^2 \geq \sigma_v^2$, the prediction gives no gain, and it's better to directly transmit the vectors.

This problem can increase the total bit-rate. The distortion is not impacted, because it is computed from transform coefficients, so the increase of bit-rate will deteriorate the performances of the QMV encoder compared to those of the traditional H.264 encoder. In the next sections, some possible solutions in order to take into account the influence of prediction coming from different quantization steps are analyzed. A first solution is to introduce a criterion in order to choose if the transmission of prediction error is more convenient than the transmission of original vectors. Another solution is to adapt the prediction according to the quantization steps of neighbor vectors.

4.3.1 Switch on the prediction of the quantized vectors

In order to evaluate the importance of the prediction, two different values of J_{QMV} have to be considered in the first pass. Indeed, for each q_v in the testing set, one can compute:

$$J_{\mathsf{QMV},pred}(q_v) = D_{pred}(q_v) + \lambda_{\text{mode}} R_{pred}(q_v),$$

which gives the best quantization step $q^*_{v,pred}$ in the case of prediction, and

$$J_{\mathsf{QMV},vect}(q_v) = D_{vect}(q_v) + \lambda_{\text{mode}} R_{vect}(q_v),$$

which gives $q^*_{v,vect}$. Then, the two different J_{QMV}, evaluated with the respective best q^*_v, are compared. At the second pass, if the minimum of the two J_{QMV} corresponds to $J_{\mathsf{QMV},pred}$, computed with the rate of residual error, the second pass is not changed and $q^*_{v,pred}$ is used. Otherwise, a "not-predictive" coding is done, *i.e.*in the second pass $J_{\mathsf{QMV},vect}$ is computed with $q^*_{v,vect}$ and

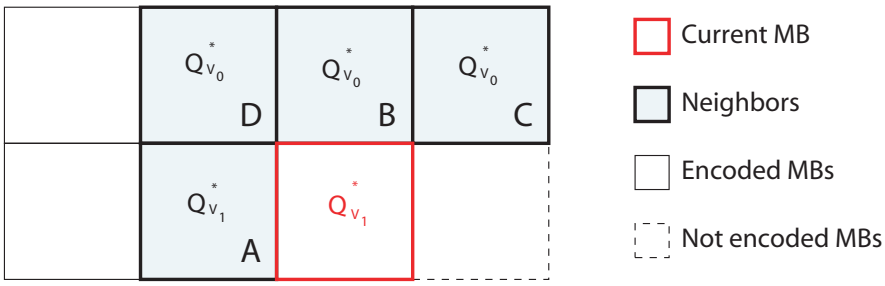


Figure 4.6: The current MB (in red) and its neighborhood for quantized MVs prediction.

the rate obtained from the encoding of quantized MVs.

It should be remarked that, according to the q_v selection strategy of Section 4.2.2.1, it exists two different way to compare J_{QMV}. For the "Oracle" case, $J_{\mathsf{QMV},pred}(q^*_{v,pred})$ and $J_{\mathsf{QMV},vect}(q^*_{v,vect})$ are compared for each MB, because they can have different best q^*_v and the minimum between the predictive or not-predictive J_{QMV} is chosen. For the "Minsum" case, the best q^*_v is chosen for the whole frame, so two global J_{QMV} have to be estimated and then compared:

$$J_{\mathsf{QMV},pred} = \sum_k J^k_{\mathsf{QMV},pred}(q^*_{v,pred}),$$

and

$$J_{\mathsf{QMV},vect} = \sum_k J^k_{\mathsf{QMV},vect}(q^*_{v,vect}),$$

with k the number of the MB.

4.3.2 *Adaptive Prediction constrained on Qv values*

If the neighborhood is composed of quantized MVs with different quantization steps q_v, as in the "Oracle" case, the prediction error on the current quantized motion vector could have a significant energy, when the description of the motion is very precise. In order to reduce this energy, only the MVs that have been quantized with the same precision are used for the prediction. For an example, let consider Figure 4.6, where the current MB has a quantization step equal to q_{v_1}. Normally, for its prediction, a median with MBs A, B, C would be computed. If the constrain on q_v values is considered, only the MV of MB A will be used, because its quantization step is equal to q_{v_1}. If the current MB has quantization step q_{v_0}, the MVs of B and C will be used instead, and so on.

These improvements allow to obtain interesting results for the new QMV coding mode.

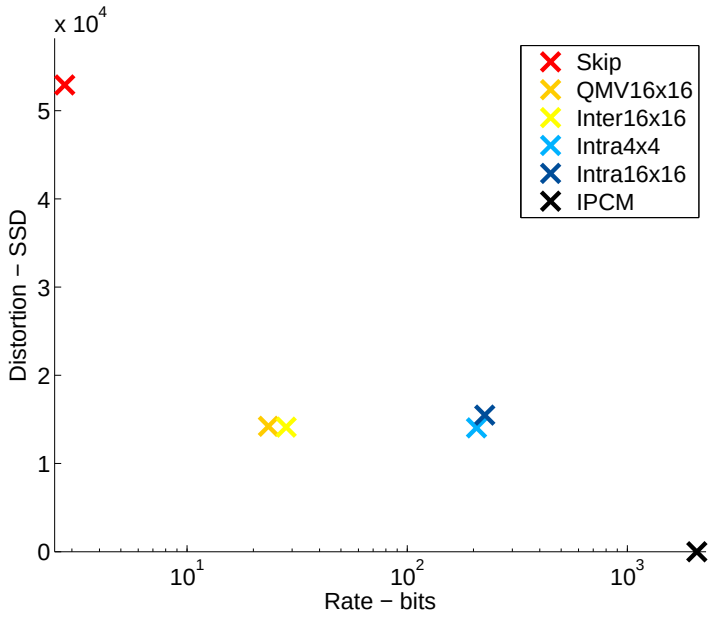


Figure 4.7: Operation points with the new QMV 16x16 mode, sequence CITY.

4.4 Some experimental results

The QMV mode has been implemented over the H.264/AVC JM software (v.11.0 KTA 1.4 [H26]), called H.264 in the following, with all typical modes and with 1/8-pel motion estimation enabled. Results for the 16x16 and 8x8 partitions, and results when both the 16x16 and 8x8 partitions are enabled, are presented. All of these results have been validated at the decoder. Both strategies described in Section 4.2.2 and approaches presented in Section 4.3 have been jointly considered. Different kinds of results are presented.

4.4.1 Operation points

In a first test, the sequence CIF CITY is coded with the new encoder, with all modes enabled, and with the same conditions than at Figure 4.2. The average operation points of the encoding modes are computed. The results are shown in Figure 4.7 for the Minsum mode. As expected, the new QMV mode has a behavior intermediate between the SKIP and the INTER modes, while remaining close to INTER mode. Similar results have been obtained for other sequences.

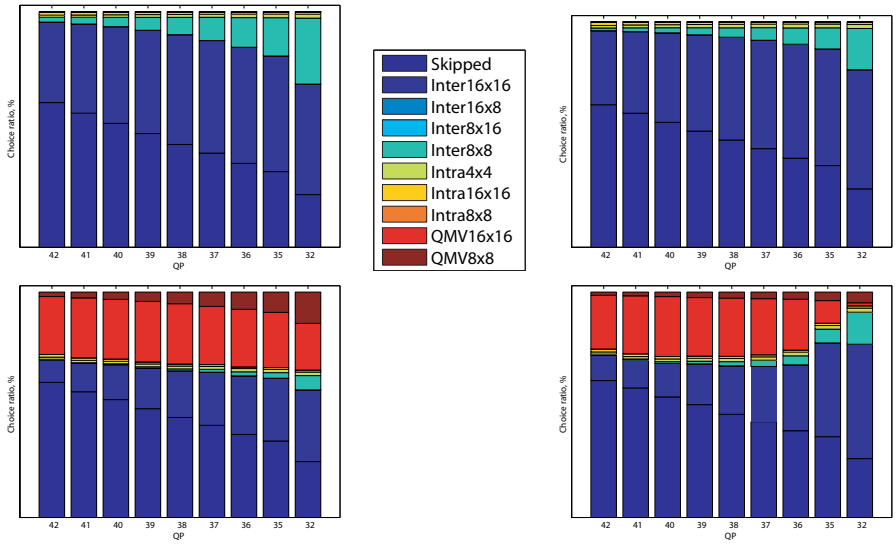


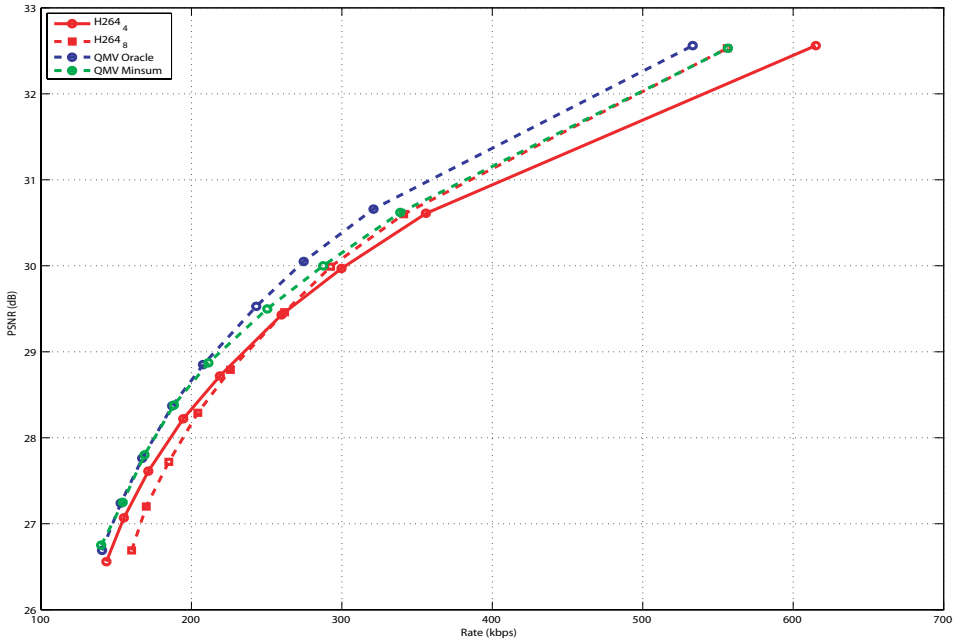
Figure 4.8: Mode distribution, 16x16 and 8x8 enabled, *tempete*. First row: H.264 + 1/4-pel, H.264 + 1/8-pel; second row: QMV Oracle, QMV Minsum.

4.4.2 Mode distribution

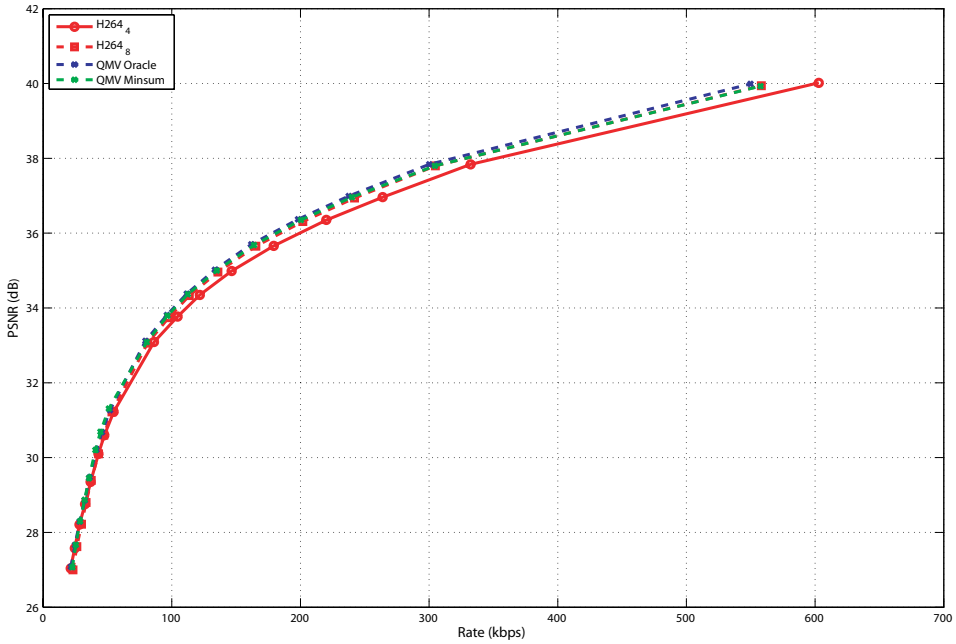
In Figure 4.8, the mode distribution for the 4 encoders (H.264 codec at 1/4-pel and 1/8-pel motion estimations, and new coding mode with both strategies "Oracle" and "Minsum", partitions 16x16 and 8x8 enabled, motion estimated at the 1/8-pel precision), for the sequence CIF TEMPETE is reported. The set of available q_v values is $S_Q = \{\frac{1}{8}, \frac{2}{8}, \frac{3}{8}, \frac{4}{8}, \frac{5}{8}, \frac{6}{8}, \frac{7}{8}, 1\}$, weighted by the MVs dynamic. In the "Oracle" case, the QMV mode has almost always replaced the INTER mode. This is reasonable since "Oracle" chooses the best q_v for each MB. When the more realistic "Minsum" strategy is used, the QMV mode is frequently chosen at low bit-rates (*i.e.* large Q_p). When the available bit-rate increases, the INTER mode is chosen more frequently. Once again, similar distributions have been observed for other sequences and other sets of parameters.

4.4.3 Coding performances

In order to assess the rate-distortion performances of the new codec, the sequence SD CITY is encoded with the new encoder, with only the 16x16 partition enabled. The performances in terms of PSNR are computed for the four configurations and are presented at Figure 4.9(a). The sequence CONTAINER is also encoded with the new encoder, with only the 8x8 partition enabled. The results are shown in Figure 4.9(b). As expected, the new mode has better results. Similar comments can be made about the rate-distortion curves for the sequences CIF TEMPETE and SD SOCCER, with

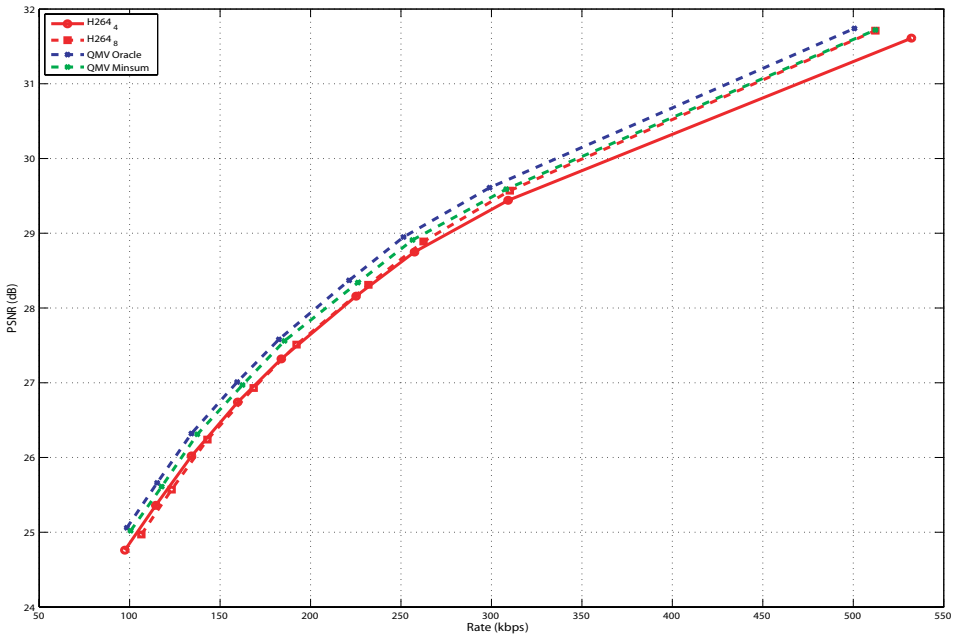


(a) CITY SD @ 30 fps, 16x16 coding modes.

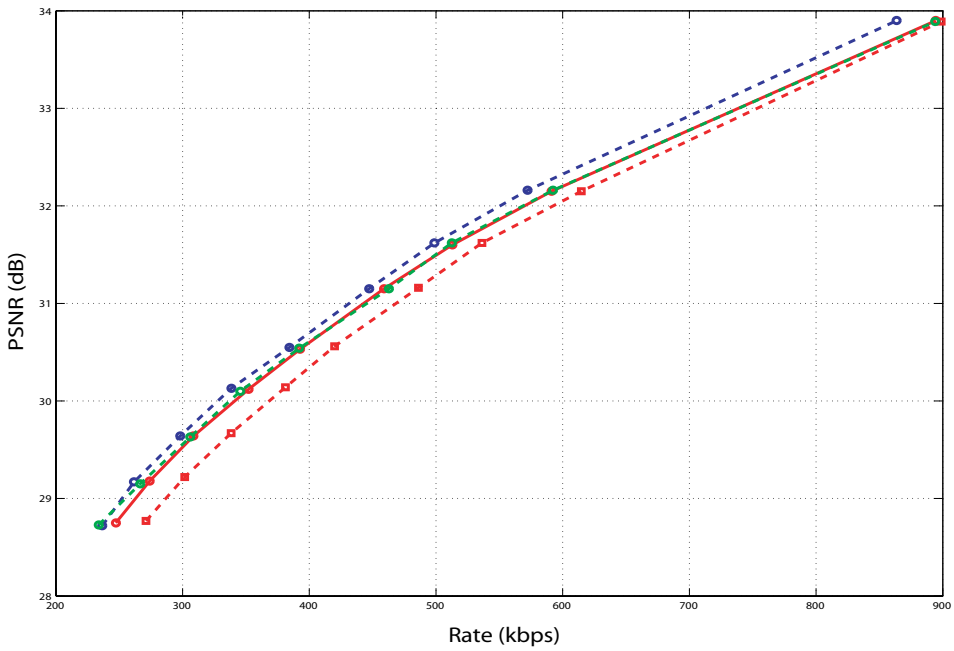


(b) CONTAINER CIF @ 30 fps, 8x8 coding modes.

Figure 4.9: Rate-distortion performances of the QMV coding mode.



(a) TEMPETE CIF @ 30 fps.



(b) SOCCER SD @ 30 fps.

Figure 4.10: Rate-distortion performances of the QMV coding mode, partitions 16x16 and 8x8 enabled.

Strategy	**Oracle**		**Minsum**	
Total rate	Low	Medium	Low	Medium
FOREMAN, H264 $\frac{1}{4}$pel	-3,79	-3,78	-3,03	-0,07
FOREMAN, H264 $\frac{1}{8}$pel	-13,50	-12,84	-12,82	-9,50
TEMPETE, H264 $\frac{1}{4}$pel	-6,74	-6,62	-3,87	-4,20
TEMPETE, H264 $\frac{1}{8}$pel	-8,21	-6,26	-5,44	-3,83
MOBILE, H264 $\frac{1}{4}$pel	-7,14	-8,49	-4,22	-3,38
MOBILE, H264 $\frac{1}{8}$pel	-8,03	-6,00	-5,15	-0,87

Table 4.1: Per cent rate savings given by the Bjontegaard metric for the QMV mode at different rates (Q_p from 36 to 39), 16x16 and 8x8 enabled.

both partitions 16x16 and 8x8 enabled, shown in Figure 4.10. Similar results have been obtained for other sequences.

Then, with the same 4 encoders and with both the 16x16 and 8x8 partitions enabled, several luminance-only CIF video sequences at 30 frames per second, FOREMAN, TEMPETE, and MOBILE, are compressed. Q_p takes the values 32 to 42, in order to check the behavior of the coders from low to medium bit-rates. In Table 4.1 are reported the per cent rate savings of QMV modes with respect to the two H.264 coders over these sequences, using the Bjontegaard metric [Bjo01, PJ07], as recommended by the VCEG and JVT standardization groups. Two rate intervals are considered: low (corresponding to Q_p ranging from 39 to 42) and medium (Q_p from 36 to 39) rates. The QMV encoders improve the performances with respect to H.264, and widely with respect to H.264 at the 1/8-pel precision. Indeed, high-resolution MVs are not worth at low rates, where the standard encoder has the best performance, while at high rates, high-resolution MVs can be afforded. However, with QMV mode, 1/4-pel performances and 1/8-pel performances are improved, here when the 8x8 and 16x16 partitions are enabled. Indeed, the MVs rate and the MVs precision are adapted, thanks to the quantization step by slice or even by MB, this precision becomes variable and is optimized according to the complexity of the motion. On the contrary, with the classical implementation of H.264, the precision of the MVS is fixed (1/4- or 1/8-pel). The "Oracle" coder has normally slightly better performances than the "Minsum" one, and even better than those obtained by the 1/4-pel and 1/8-pel H.264 coders.

4.5 Conclusion

Even though H.264 has excellent rate-distortion performances, some intuitions suggest that they can be improved using a more flexible motion coding.

A new coding mode based on MVs quantization has been proposed, in the framework of an industrial contract with Orange labs [ACA+07, CAA+07]. In order to insert this technique in the highly optimized H.264 encoder, some problems regarding the choice of the quantization step and the encoding of quantized MVs have been solved [MCA09]. For the 8x8 coding mode, some issues due to the high precision of the motion have also been resolved. The experimental results show that this new coding mode brings a non-negligible gain. In fact, the best performances are widely better than those of the H.264 1/4 or 1/8-pel coder [CAC+09].

The first part of this work has dealt with video coding, by using both of the main existing coding techniques: the wavelet-based and the hybrid techniques. But the transmission of the resulting encoded video sequences over communication channels is also a great challenge. This is the subject of the second part of this thesis.

Part II

Transmissions of videos over noisy channels

Chapter 5

Multiple description coding: the state-of-the-art

In the last decade, the use of mobile and multimedia communications has seen an enormous increase, with the wireless channels considered as a transport medium for various types of multimedia information. Due to the high bit rates involved with multimedia, the scarcity of wireless bandwidth, the time-varying characteristics of the channel, and the power limitations of wireless devices, multimedia communications, specially the wireless ones, are a tremendous challenge.

In order to have a coding scheme robust to transmissions over noisy channels, multiple description coding (MDC) is mainly explored in the second part of this thesis. An overview of the literature is thus drawn in this chapter. The main theoretical results and the main approaches of MDC are presented. Techniques where MDC is applied to video are also presented, and a focus on the decoding of multiple descriptions ends this state-of-the-art.

5.1 The theoretical principles of multiple description coding

The MD problem was posed by Gersho, Witsenhausen, Wolf, Wyner, Ziv and Ozarow at the September 1979 IEEE Information Theory Workshop as a generalization of Shannon's problem of source coding with fidelity criterion [Sha59]. This problem can be briefly posed as: "If a source is described by two different descriptions, which are the quality limitations of these descriptions taken apart and jointly?". It is summarized in Figure 5.1. The difficulty in such a problem is that good individual descriptions must be close to the process, and necessarily must be highly dependent. Thus, after the reception of the first description, the second description will contribute little extra information. On the other hand, two independent descriptions must be far apart and thus cannot in general be individually good.

The first theoretical results of multiple descriptions appear in 1980 and

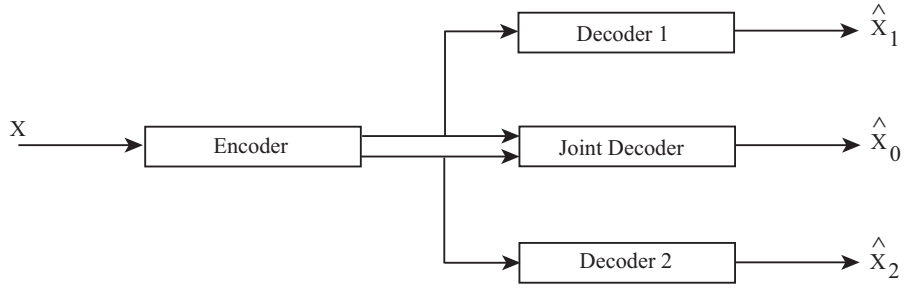


Figure 5.1: Classical scheme of MDC with two descriptions.

try to characterize the set of achievable quintuples $(R_1, R_2, D_1, D_2, D_0)$, with R_1 and R_2 the bit-rates of the side descriptions, D_1 and D_2 their distortions, and with D_0 the central distortion. They were proposed by Witsenhausen [Wit80], Ozarow [Oza80], Wolf, Wyner and Ziv [WWZ80], and El Gamal and Cover [GC82].

In [Wit80], Witsenhausen presented a lower bound for side distortion when a memoryless binary symmetric source is considered with a Hamming distance (i.e., probability of error) as distortion. In [WWZ80], Wolf, Wyner and Ziv also considered the binary symmetric memoryless source and the Hamming distance as distortion. They proved that, if $(R_1, R_2, D_0, D_1, D_2)$ is achievable, then $R_1 + R_2 \geq 2 - h(D_0) - h(D_1 + 2D_2)$ and $R_1 + R_2 \geq 2 - h(D_0) - h(2D_1 + D_2)$, where

$$h(\lambda) = \begin{cases} 0, & \lambda = 0 \\ -\lambda log_2\lambda - (1-\lambda)log_2(1-\lambda), & 0 < \lambda \leq 1/2 \\ 1, & \lambda > 1/2. \end{cases} \tag{5.1}$$

When $R_1 = R_2 = 1/2$, $D_0 = 0$ and $D_1 = D_2$, the rate-distortion bound implies that $1 - h(D_1) \leq R_1 = 1/2$, or $D_1 \geq 0.11$. However, the theorem of Wolf, Wyner and Ziv, yields $h(3D_1) \geq 1$, or, $D_1 \geq 1/6$. The authors conclude that the bound under the Shannon assumptions is defined by the tangents to the hyperbola at the two points where it cuts the coordinate axis.

Works by El Gamal and Cover [GC82] have shown that, under the Shannon assumptions, all points above the hyperbola are achievable. The hyperbola is known as the achievable rate region of (R_1, R_2) pairs as a function of the distortion vector $D = (D_1, D_2, D_0)$ (see Figure 5.2), for a memoryless source and a single-letter fidelity criterion as proved in the following theorem.

Theorem Let $X_1, X_2, ...$ be a sequence of i.i.d. finite alphabet random variables drawn according to a probability mass function $p(x)$. Let $d_i(.,.)$

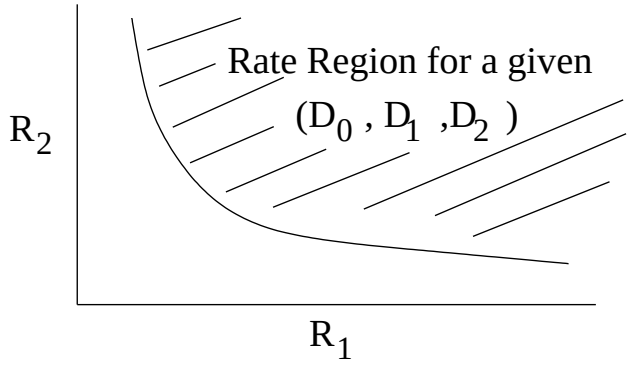


Figure 5.2: Achievable rate region of (R_1, R_2) pairs as a function of the distortion vector $D = (D_1, D_2, D_0)$.

be bounded. An achievable rate region for distortion $D = (D_1, D_2, D_0)$ is given by the convex hull of all (R_1, R_2) such that

$$R_1 > I(X; \hat{X}_1),$$

$$R_2 > I(X; \hat{X}_2),$$

$$R_1 + R_2 > I(X; \hat{X}_1, \hat{X}_2, \hat{X}_0) + I(\hat{X}_1; \hat{X}_2),$$

with I the mutual information, for some probability mass function $p(\hat{x}, \hat{x}_0, \hat{x}_1, \hat{x}_2) = p(x)p(\hat{x}_0, \hat{x}_1, \hat{x}_2|x)$ such that

$$D_0 \geq E\{d_0(X; \hat{X}_0)\},$$

$$D_1 \geq E\{d_1(X; \hat{X}_1)\},$$

$$D_2 \geq E\{d_2(X; \hat{X}_2)\}.$$

Ozarow in [Oza80] considered the case where the sources are Gaussian and the distortion is the squared-error criterion. The achievable rate region (the hyperbola) derived in [GC82], was proved to be, in fact, the rate-distortion region for the source. This is done by obtaining the converse theorem. This theorem states that the achievable set of quintuples $(R_1, R_2, D_1, D_2, D_0)$ is given by the set of points satisfying $D_1 \geq e^{-2R_1}$, $D_2 \geq e^{-2R_2}$ and $D_0 \geq e^{-2(R_1+R_2)} \frac{1}{1-(\sqrt{(1-D_1)(1-D_2)}-\sqrt{D_1 D_2 - e^{-2(R_1+R_2)}})^2}$.

Witsenhausen and Wyner [WW81] have obtained an outer bound for the case of a binary symmetric source with the Hamming distortion and have compared it in one case to the achievable region of [GC82], but the bounds exceed the achievable point.

Berger and Zhang in [BZ83] defined $d = inf\{D : (\frac{1}{2}, \frac{1}{2}, D, D, 0)$ is achievable$\}$

for memoryless binary sources. They proved that $d = (\sqrt{2} - 1)/2$.

Ahlswede in [Ahl85] proved the tightness of the hyperbola bound in [GC82] on a case of no excess rate at D_0 ($R_1 + R_2 = R(D_0)$), for the binary symmetric memoryless source with an error frequency distortion criterion. Zhang and Berger in [ZB87] disprove the conjecture that the achievable rate region given in [GC82] coincided with the rate-distortion region in case of binary symmetric source with Hamming distortion measure.

An important special case of the MD problem was presented in [EC91] and is known as the problem of successive refinement of information or multiresolution (MR). The successive refinement problem is a special case of the MD problem in which there is no constraint on $E\{d(X, \hat{X}_1)\}$ and in which $R_2 = R(D_2)$ and $R_1 + R_2 = R(D_0)$ is required. In this article, a necessary and sufficient condition is derived, such that, the rate-distortion problem is successively refinable. The result follows from the tightness of the achievable region established by El Gamal and Cover [GC82] for the no excess rate sum case [Ahl85].

At this moment there is a potential for applications of MD source codes in speech and video coding over packet-switched networks where packet losses can result in a degradation in signal quality. One of the first practical coder design for multiple descriptions appears in the context of speech coding. In 1981, Jayant and Christensen [Jay81, JC81] consider MD coding of DPCM speech for combating speech coding degradation due to packet losses.

Different approaches of multiple description image coders can now be found. Multiple description transform coding, on which this work is based, is first presented. Multiple description coders can be also based on multiple description quantization, or on forward error correcting (FEC) codes.

5.2 Multiple description transform coding

In the multiple description transform coding (MDTC) approach, linear transforms are used to introduce a controlled amount of correlation among the transformed coefficients.

The MDTC of a source x uses a decorrelating transform T_1 (e.g. KLT, DCT, DWT ...), and transform the result (which can be quantized) with an invertible, discrete transform $T_2 : \mathbb{C}^n \to \mathbb{C}^m$. When all components are received, the reconstruction process is to exactly invert the transform. If some components are lost, they are estimated from the received components using the statistical correlation introduced by the correlating transform.

The MDTC system focuses on the search for optimal redundancy rate-distortion points by designing the correlating transform T_2. Two different MDTC methodologies can be found, which are described below. The first one, called square-transform based uses non-overlapping linear transforms while the second MDTC methodology, called frame-based, uses overlapping

transforms.

5.2.1 Square-transform based MDTC

The square-transform based MDTC was pioneered by Wang, Orchard, and Reibman [WOR97, WOR98]. A $N \times N$ block linear transform has been designed to introduce a controlled amount of correlation between the transform coefficients. In this way, if one of the descriptions is lost, the other one can be statistically estimated using the introduced dependencies. These works also address several issues related to the optimality of the transforms used for encoding, and it is shown that non-orthogonal transforms perform better than orthogonal transforms in terms of redundancy rate-distortion gain.

In [OWVR97], Orchard *et al.* discuss MD coding of two dimensional Gaussian vectors using transform techniques. This work introduces a performance metric called *redundancy rate-distortion function*, where the redundancy rate is defined as the number of extra bits required to match a given coding distortion, compared to a single description coding (SDC) system. Indeed, the performance of a MDC system can be measured with three parameters: the bit rate, the coding distortion, and the reconstruction distortion. The coding distortion refers to the error between the original signal and the decoded one from all descriptions, while the reconstruction distortion is defined as the error under a given channel loss profile. With conventional SDC, the goal is to maximize the coding efficiency which is equivalent to minimize the bit rate for a given coding distortion, or vice versa. With MDC, in order to reduce the reconstruction distortion, the coder must introduce a certain amount of correlation among separate descriptions, which will reduce the coding efficiency compared to that achievable by SDC.

In [GK98, GKAV98, GK01], the authors generalize the construction proposed in [OWVR97] by dealing with arbitrary N-dimensional vectors, and by expanding the set of transforms which are considered.

In [CW99b], the authors developed a MD encoder that generates multiple descriptions by splitting the coefficient blocks of a conventional LOT-based encoder (Lapped Orthogonal Transform). A maximally smooth image-recovery method is developed as part of the MDC decoder, which can recover the original signal from an incomplete set of coefficient blocks.

In [JO99a, JO99b], the authors propose a two stage transform design technique for MDTC. The motivation is that protection properties of a MDTC system can be characterized by which descriptions are correlated (structure), and by what extent they are correlated (magnitude). While the magnitude information cannot, in general, be quantified for specific redundancy and distortion constraints, the structural information can be inferred from specific channel conditions. Consequently, the structure design will find admissible transforms using a scaling-rotation factorization and the mag-

nitude design will search for the optimal transform from these admissible transforms.

5.2.2 Frame-based MDTC

The frame-based MDTC was pioneered by Goyal, Kovacevic, and Vetterli, the descriptions are constructed by separately describing the N coefficients of an overcomplete $N \times K$ tight frame expansion [GKV98]. Here, a linear transform from $\mathbb{R}^k$ to $\mathbb{R}^n$, followed by scalar quantization, is used to generate n descriptions of a k-dimensional source. The n descriptions are such that a good reconstruction can be computed from any k descriptions. In [GKAV98], preliminary image communication experiments are presented using the methods of [GK98, GKV98]. In [CMW99], a POCS-based (Projection Onto Convex Sets) algorithm for consistent reconstruction from MD of overcomplete expansions is developed. Consistent reconstructions have smaller expected squared error distortion than inconsistent reconstructions [GVT98]. The authors construct the frame from two complete transform bases. Since such transforms are usually efficient to compute, they can perform the reconstruction much faster than with previous methods.

Some authors dedicated their work to the construction and analysis of filter banks for MDTC or more generally for image coding and transmission over erasure channels. For example, in [BDV00], a windowed Fourier method is used for a MDC based on overcomplete expansions. In [YR00, DSV02b], the authors designed biorthogonal filter banks for MD coding of Gaussian sources, with the difference that in [YR00], they use the correlating transform before quantization, and in [DSV02a], the quantization step is performed before the transform. In [Mar84] oversampled block transform like the Discrete Fourier Transform (DFT) codes have been considered for MDC. Filter bank frame expansions have also been studied to achieve resilience to erasure [KDG02, DKG01, GKV99, MC00]. In [MG03] two channel oversampled filter banks (OFBs) and tree-structured oversampled filter banks which implement frame decomposition are considered. Tree-structured OFBs provide a natural framework for unequal loss protection.

Several of the approaches mentioned above involve the design of specific transforms or quantizers that have to match with the desired level of protection. In these schemes, adapting to changing network conditions would entail changing the transforms and/or quantizers. The last approach of MDTC tries to overcome this limitation.

5.2.3 MDC using explicit redundancy

This MDC approach exploits the natural correlation between symbols for reconstruction. This approach is similar to the square-transform based MDTC approaches above, except that the transform is not actively designed. An

example is the multiple description via polyphase transform (MDPT) developed by Jiang and Ortega [JO99a]. MDPT is an extension to SPIHT coder [SP96] by separating zerotrees into polyphase components. The principles of embedded zerotree wavelet (EZW) coding introduced by Shapiro in [Sha93] are used. Rogers *et al.* [RC98] propose to rearrange bits at the output of one configuration of the SPIHT coder, in such a way that the loss of one packet results in an error that does not propagate beyond the image region contained in that packet.

Miguel, Mohr and Riskin proposed a scheme using SPIHT in a generalized multiple description framework [MMR99], called MD-SPIHT. In [MR00], the authors extend the unequal loss protection framework of MD-SPIHT by adding more redundancy to the region of interest (ROI) than to other parts of the image. In this way, they present an efficient scheme for protecting a ROI.

In [JO99a, MMR99], explicit redundancy is introduced, so that each sample in the input (for example each wavelet coefficient) is transmitted more than once and coded with different accuracy each time. This strategy has the drawback of leading to transmission of more samples than initially present in the source, and thus inefficiency in the case of error-free transmission.

In [SO00], Sagetong and Ortega demonstrate how these explicit redundancy techniques have the additional advantage of providing very simple mechanisms for adaptation to changing network conditions. The key observation is that the level of redundancy can be selected by determining the number of times a given sample is transmitted, and how many bits should be used for each of the redundant representations. In this paper, the authors show how a bit allocation problem can be defined, where the goal is to choose the best distribution of redundancy for a given packet loss rate. They provide techniques to solve this problem and show how different loss rates require different levels of redundancy. Note that by using bit allocation to determine the level of redundancy, not only the encoder can adjust itself in a simple manner, but in addition the decoder can handle packets with different levels of redundancy without requiring any significant changes to its structure. More specifically, the MDC technique used here generates the various descriptions through a polyphase transform. For example, this polyphase-based MDC will divide a scalar source into even and odd samples, and will compress each sample using two different quantization scales (coarse and fine). Then this approach will transmit groups of samples where a set of coarsely quantized odd samples is combined with a set of finely quantized even samples, and vice versa. The decoder operates by gathering the available information for each sample and then selecting for each polyphase component its highest quality copy to be used in the decoding; the remaining copies are discarded. In [SO01], the authors improve the system by using a *priority scaling factor* to introduce redundancy in each description.

The coders proposed above are designed for ideal MDC channel environments, where the channels are independent and data on each channel is either completely lost or received intact. In a packet network environment these ideal conditions may not hold true: packet losses can be correlated and only partial data (of either description) may be received at the decoder.

Besides, one can remark that the last MD coders are joint source-channel coders that use redundancy adaptability to be adapted to changing network conditions. The existing methods using this approach are dedicated mostly to ideal MDC channel environment, and the last ones to packet lossy channels. For example, in [Per04], Pereira presents a MDC scheme who adapts the explicit redundancy to changing networks transmission. A bit allocation allows the automatic adjustment of the encoder, with no changes needed at the decoder.

5.3 Multiple description quantization

The second family of MDC approaches is the multiple description quantization, which can be scalar or vector quantization.

5.3.1 Scalar quantization

First, multiple description scalar quantization (MDSQ) was pioneered by Vaishampayan in [Vai93]. MDSQ proposes that the rate of the descriptions can be traded of against the side distortions. The quantizer is obtained by a standard scalar quantizer followed by an index assignment that splits the signal into two descriptions. In this way, it sends information from each sample over both channels. This design problem is posed as an optimization problem and necessary conditions for optimality are derived in [Vai93]. Unlike a single channel scalar quantizer, the performance of a MD scalar quantizer is dependent on the index assignment. The author addresses the problem of index assignment and describes two families of index assignment matrices in which the maximal distortion between two indices sharing a description is minimized.

Let us briefly describe the MDSQ. A (M_1, M_2)-level multiple description scalar quantizer maps the source sample x to the reconstruction levels $\hat{x}^0, \hat{x}^1, \hat{x}^2$ that take values in the codebooks, $\hat{\chi}^0 = \{\hat{x}^0_{i,j}, (i,j) \in \mathcal{C}\}, \hat{\chi}^1 = \{\hat{x}^1_i, i \in \mathcal{I}_1\}$ and $\hat{\chi}^2 = \{\hat{x}^2_j, i \in \mathcal{I}_2\}$, respectively, where $\mathcal{I}_1 = \{1, 2, ..., M_1\}, \mathcal{I}_2 = \{1, 2, ..., M_2\}$ and $\mathcal{C}$ is a subset of $\mathcal{I}_1 \times \mathcal{I}_2$. An MDSQ can be broken into two side encoders, $f_1 : \mathbb{R} \to \mathcal{I}_1$ and $f_2 : \mathbb{R} \to \mathcal{I}_2$ which select the indexes i and j, respectively, and three decoders, $g_0 : \mathcal{C} \to \mathbb{R}$ (central decoder), $g_1 : \mathcal{I}_1 \to \mathbb{R}$ and $g_2 : \mathcal{I}_2 \to \mathbb{R}$ (side decoders), whose outputs are the reconstruction levels with indexes ij, i and j from the codebooks $\hat{\chi}^0, \hat{\chi}^1$, and $\hat{\chi}^2$, respectively. The rate of the encoder f_m is given by $R_m = log_2 M_m \, bpss, m = 1, 2$. The two encoders impose a partition $\mathcal{A} = \{A_{i,j}, (i,j) \in \mathcal{C}\}$ on $\mathbb{R}$, where

$A_{i,j} = \{x : f_1(x) = i, f_2(x) = j\}$. The MDSQ is completely described by $\mathcal{A}, \hat{\chi}^0, \hat{\chi}^1$, and $\hat{\chi}^0$. The encoder is refereed as $= (f_1, f_2)$, the decoder as $= (g_0, g_1, f_2)$, $\mathcal{A}$ as the central partition, and the elements of $\mathcal{A}$ as the central cells. The determination of the central partition is crucial. Several methods by which this can be done are presented in [FV87]. If both indexes are received, the central decoder g_0 is used to reconstruct the source sample. On the other hand, if only $i(j)$ is received, then side decoder $g_1(g_2)$ is used to reconstruct the sample. The central and side MSEs that can be achieved are determined by the index assignment.

Vaishampayan and Domaszewicz in [VD94] extended the work in [Vai93] to entropy constrained quantizers. They also used variable length codes (VLCs) instead of fixed length codes. With VLCs, better performances are achieved, however, they are very sensitive to errors (due to synchronization problems). In [GGF02], the authors analyse the MDC system and evidence the most appropriate form of redundancy one should introduce in the context of VLC compressed streams in order to fight against de-synchronization when impaired by channel noise.

In [VB98], Vaishampayan and Batllo present an asymptotic analysis of the MDSQ of [Vai93]. Specifically, expressions are derived for the average side and central distortions and for entropy when the number of quantization levels is large. In this work, they compare the distortion product D_0D_1 of the optimum level-constrained quantizer for a unit-variance Gaussian source with the one on the converse theorem. From the converse theorem, it can be shown that the multiple description rate-distortion bound at large rates is given approximately by $D_0D_1 \approx \frac{1}{4}2^{-4R}$. The performance of the optimum level-constrained quantizer is given by $D_0D_1 \approx \frac{3\pi^2}{16}2{-}4R$ and of the optimum entropy-constrained quantizer by $D_0D_1 \approx \frac{\pi^2e^2}{144}2{-}4R$. These important results show that for MDSQ both the side and the central distortion attain the optimal exponential rate of decay ($D_0 \approx 2^{-2R}, D_1 \approx 2^{-2R}$). The only sub-optimality of MDSQ at high rates is due to the use of a scalar quantizer which partitions the space into cubic regions instead of an ideal vector quantizer that would optimally partition the space into spheres (see Section 5.3.2).

Jafarkhani and Tarokh in [JT99] constructed MD trellis coded quantizers.

In [GGP01], the authors consider the usage of multiple description uniform scalar quantization (MDUSQ) for robust and progressive transmission of images over unreliable channels. They develop an index assignment which allows to improve the rate-distortion performance against previous proposed index assignments in the context of progressive and embedded bit streams. Thus, the MDUSQ proposed is well adapted for non stationary (varying bandwidth) communication environments.

5.3.2 Vector quantization

An other possibility of MD quantization is vector quantization. Vaishampayan in [Vai91] describes an iterative algorithm similar to the generalized Lloyd algorithm that minimizes the Lagrangian of the rates and expected distortions R_1, R_2, D_1, D_2, D_0 and applied it to the optimization of multiple description vector quantizers. Non-balanced MD vector quantization was studied by Fleming and Effros [FE99], including more than two descriptions. This paper presents a new practical algorithm, based on a ternary tree structure, for the design of both fixed and variable rate multiple description vector quantizers for an arbitrary number of channels.

Some works propose the design of MD lattice vector quantizers (MDLVQ) [SVS99, VSS01, DSV02a]. The work in [DSV02a] has the particularity of considering asymmetric MD contrary to the former where the considered MD are always symmetric. In [GKK00, GKK02], a method is introduced for a two channels MD coding that generalizes the MDLVQ developed in [SVS99]. This last one uses a fine lattice Λ and a coarse sublattice Λ'. The former uses the index assignment of [SVS99] and a coarse lattice Λ. With the slight increase in complexity, the convex hull of the operating points is improved.

A MDLVQ is a triplet $\mathcal{Q} = (\Lambda, \Lambda', l)$. Λ is a lattice, and Λ' is a sublattice that is geometrically similar to Λ [CS98]. Each lattice point $\lambda \in \Lambda$ gets mapped by l to a pair of sublattice points $(\lambda'_{red}, \lambda'_{green})$ that uniquely identifies λ, *i.e.*, l must be an injection:

$$\Lambda \overset{1-1}{\longleftrightarrow} l(\Lambda) \subset \Lambda' \times \Lambda'.$$

l is referred to as the *index assignment*, and the pair of points in the image assigned by l of a point λ are referred to as *red* and *green* descriptions.

The amount of redundancy in a lattice quantizer is controlled by $N = |\Lambda/\Lambda'|$, the index of Λ' in Λ. Given any sublattice point $\lambda' \in \Lambda'$, it is required that l is such that the total number of distinct lattice points $\lambda \in \Lambda$ for which Λ' is used to describe λ is exactly N, *i.e.*,

$$|\{\lambda : \pi_{red}(l(\lambda)) = \lambda'\}| = |\{\lambda : \pi_{green}(l(\lambda)) = \lambda'\}| = N,$$

where, $\pi_{red}(\pi_{red}, \pi_{green}) = \pi_{red}$, and similarly for π_{green}. Lattice points are labeled with pairs of sublattice points (it is these sublattice points that actually get transmitted over each channel), and that each sublattice point is used exactly N times. The larger is N, the higher the uncertainty about the original lattice point when one of the channels fail.

A key property of good index assignments l is that the set of central cells that share a given label must be as localized in space as possible, in order to achieve low distortion in the case of channel failure. This is analogous to the idea that for a scalar quantizer the *spread* of a side cell must be minimized

[Vai93].

For real world sources such as speech and video, it is important to exploit the correlation in order to build efficient coders. MD quantizers can be used efficiently for sources with memory by using standard decorrelating transforms. Batllo and Vaishampayan present in [BV97] an orthogonal MDTC followed by MDSQ: the quantizers are applied to sources with memory. In [SRVN98, SRVN00] Servetto, Ramchandran, Vaishampayan and Nahrstedt use the MDC in [BV97] to design a wavelet based image coder. Some of the most successful wavelet coders [Sha93, SP96, CO97, LRO97] derive their high coding performance from their ability to identify sets of coefficients with different statistics within image subbands, and then coding each of these sets with respect to an appropriate statistical model. Since these sets typically are image dependent, this information is not known a priori, and therefore must be somehow conveyed to the decoder, explicitly [Sha93, SP96], or implicitly [CO97, LRO97]. These schemes are thus particularly well-adapted to MDC.

5.4 MDC based on forward error correcting codes

The previous methods of MDC introduced redundancy at the source coding level. The methods presented in [ABE$^+$96, DD96] propose to use channel coding techniques to add redundancy to the transmitted signal. The information flow to be coded is thus supposed to be organized in a hierarchical way and segmented in layers of decreasing importance. These layers can then be protected by error correcting codes of also decreasing redundancy. Unlike others MDC approaches, these ones achieve MD property without modifying the source coding algorithm. Rather, correlation is reintroduced into the transmitted bitstream by applying different amounts of error protection to the sections of the bitstream produced by the source coder, and then combining these sections into equally important descriptions.

Mohr *et al.* propose the use of error correcting codes of different strengths applied to different portions of a progressive bitstream such as that generated by SPIHT coder [MRL99a, MRL99b]. As the source coding and the channel coding are done separately in this kind of schemes, it is thus necessary to use bit-rate allocation techniques between the two kinds of coding. This can be very complex, two methods allowing to obtain an optimal partition of the flow minimizing the average distortion at the decoder side are described in [MRL99a, MRL99b, PR99]. In these MD FEC systems, the reduction of distortion associated with any description actually depends on how many other descriptions are received.

5.5 Multiple description video coding

The MDC approaches can be applied to video coding. The main motivation for the use of MDC in this domain is its ability to assure a minimum quality without the need of the retransmission of lost packets. This particularity makes MDC really interesting for interactive and real-time applications, where a second packet transmission would be unacceptable.

The work of Vaishampayan [VJ99] can be noticed: the predictive MD system was applied along with transform coding to construct an inter-frame balanced MD video coder based on the H.263 standard. Apostolopoulos [Apo99, Apo00a] shows that MD coding and path diversity provide improved reliability in systems with multiple paths with equal or unequal bandwidths. Reibman, Jafarkhani, Wang, Orchard and Puri [RHW+99] have proposed MD video coders which use motion-compensated predictions. In [YWK00, Apo00a], a temporal sub-sampling produces two descriptions separately coded by a predictive coder. The redundancy thus comes from the sub-optimality of the prediction which is done in less correlated frames.

More recently, the works of Reibman [Rei02, RJW+02, WRL05] can be found, who propose a MD video coder based on a bit-rate allocation. Wang and Lin, in [WL02], predict each image between two images in different descriptions. A similar approach is presented in [KKL01]. Heng, Apostolopoulos and Lim [HAL06] propose a MDC adaptive method taking into account the characteristics of the network and the video. MD approaches based on FEC codes (see Section 5.4) have also been applied to video, as in [WFI05, EKKS07].

Several problems of communication over network does not allow the implementation of the source/channel separation theorem. Moreover, some constraints as real-time communications or a tight control of the channel load can forbid packets retransmission. In these conditions, only joint source/channel coding can allow to increase robustness. McCanne *et al.* [MV95, SVJ97] have proposed the use of joint source/channel coding for multicast video transmission on heterogeneous network. In their approach, each receiver can dynamically choose the local network capacity by adjusting the quality of the received video. In a network as Internet, several descriptions can be send to a receiver along different paths [Apo00b]. An approach of robust peer-to-peer streaming has been presented by Padmanabhan, Wang and Chou [PJC03], based on a construction algorithm of several distribution trees in order to introduce a diversity in the used network paths, and combined with a MDC which introduces redundancy in the transmitted data. An other system to transmit a video over a peer-to-peer network is presented in [ATC07]. It is based on an architecture of MDC which adapts the number of descriptions, their type and their quantity of redundancy in function of the network state.

MDC has also been used in $t+2D$ video coding with motion-compensated

temporal filtering. In [PAB02, Per04], Pereira, Antonini and Barlaud use a scan-based 3D wavelet transform, allowing to automatically adapt the amount of added redundancy dispatched on the different descriptions according to the error characteristics of the underlying channel. Van der Schaar and Turaga [dST03] propose to build two descriptions on a motion-compensated dyadic temporal decomposition by duplicating the temporal approximation subband in each description. Comas, Singh, Ortega and Marqués [CSOM03] propose an unbalanced MDC system (one description encoded at high rate, the other encoded at low rate). In [TPPdS04, Til05, TPPP07], Tillier, Pesquet-Popescu and Van der Schaar present a MD coder based on a three-band Haar scheme (the redundancy is introduced at the source by subsampling the sequence).

5.6 Extension to N descriptions

The MDC problem can be extended at N channels, with $2^N - 1$ decoders. This generalization has been studied by Witsenhausen [Wit80] for the case where the source has a entropic finite rate and where a lossless communication is required, whatever the number of received descriptions. The author concluded that for a certain value of $k, 0 < k < N$, if any k (or fewer) channel breaks down, $R = \frac{1}{N-k}$ is the rate required to obtain error-free operation. Similar results are presented in a more general framework in [Wit81], and a case with three channels and seven decoders has been studied by Zhang et Berger in [ZB87].

More recently, in [GKK00], the extension of the algorithm in [SVS99] provides a technique for more than two descriptions. Venkataramani, Kramer and Goyal have found bounds on the achievable performance region for MD coding with more than two descriptions [VKG01]. Puri, Pradhan et Ramchandran [PPR02a, PPR02b] also managed to characterize the performances of the problem of N descriptions coding. For that, they applied the results presented in [PPR01] to the problem of N descriptions symmetric coding by using the structure of coding proposed by Albanese *et al.* [ABE+96] and detailed in Section 5.4.

Berger-Wolf and Reingold in [BWR02] found an index assignment and a performance bound for MD scalar quantization for more than two descriptions. The index problem is formulated as a combinatorial optimization problem of arranging numbers in a matrix to minimize the maximum difference between the largest and the smallest number in any row or column. In the case of two descriptions transmitted at equal rates, the bounds (lower and upper bound) coincide, thus giving an optimal algorithm for the index assignment problem. In the case of three or more equal channels, the bounds are within a multiplicative constant.

In [PA05], the authors propose a MDC method for N channels where the redundancy estimation applied to each description is estimated based

on the channel information. The MDC method for N channels permits to construct a multi-channel adaptive allocation codec. The proposed codec is well suited for wideband mobile communications where the channel can be modeled as the superposition of a discrete number of paths.

5.7 Optimal decoding of noisy descriptions

MDC schemes are particularly well adapted to channels encountering packet losses [LMWA05]. These channel models are representative of transmission of multimedia contents over wired links or over wireless links with classical protocol stacks, *e.g.*, RTP/UDP/IP over 802.11 MAC and PHY layers, where any error in a transmitted packet results in the loss of that packet, and when retransmission is not possible as in the case of broadcasting. When a link is broken, all of the symbols or packets passing through that channel are lost; when it is functioning properly, the symbols are transmitted error free.

Nevertheless, for wireless applications, there is currently some research effort in the development of *permeable* protocol stacks allowing transmission errors to reach the upper protocol stacks [LDJF04, JSX05, MMLB$^+$07, MLKD08], drastically reducing the amount of lost packets. In these protocols, robust source decoders may be put at work to correct erroneous packets. The study of the efficiency of MDC schemes for channels introducing errors at the bit level are thus particularly interesting. First results have been proposed in [Sri99, KAM01, GGF02, LWK06] for theoretical sources and in [CLKD06] for video frames compressed using a motion-compensated oversampled filterbank.

The decoding of the resulting noisy descriptions are a tough problem, however a very reduced amount of research is dedicated to it, especially in case of video transmission. The authors of [GGF02] propose a soft decoding procedure of MD for Gauss-Markov sources, where they use the source coder model and merge the soft information of the two descriptions in order to take into account the inter-symbol correlation. In [SO03], the authors propose a recovery algorithm based on sending MD of the source and using a deterministic distance measure to find the most likely estimate for the lost data, knowing the received data and the side information. This approach is applied for erasure recovery in predictive coding schemes, but one can imagine to apply it to error-prone channels. In [LWK06], in the case of mixed internet and wireless channels, the authors present an Entropy-Constrained Multiple Description Trellis-Coded Quantizer (EC-MDTCQ) combined with a Variable-Length Coder (VLC). An iterative decoding is performed, where only iterations between the VLCs and the ECMDTCQ decoder is needed, providing a reduced complexity and good performances. In [PAB03b], the authors propose a wavelet-based video MD coder. In order to provide synchronization and minimize the error propagation in the

case of channel errors, each spatio-temporal subband is divided into blocks. Then, arithmetic coding is performed on each block independently. For error detection, when the number of coded coefficients is known, it is possible to verify if the arithmetic coder stops correctly. In case of error, the decoder is synchronized to start at the beginning of next block.

In the following chapter, approaches for optimal decoding of descriptions transmitted over noisy channels are presented. The multiple description video coding scheme considered here is transform-based and uses explicit redundancy, and more precisely, it is based on the motion compensated wavelet-based video coder previously described in Chapter 3. Only the two balanced descriptions case will be considered.

Chapter 6

Optimal multiple description decoding

In the framework of a national research project, "ESSOR" [ess09], in collaboration with L2S research team, transmissions over noisy channels have been studied, and in particular multiple description video coding. MDC without side information taken into account for the time being is considered here. A focus is done on the transmission of several descriptions over noisy channels. The crucial problem of this kind of schemes is the optimal decoding of the source, based on the noisy received descriptions. Two decoding algorithms are proposed here. A first one tries to estimate the two generated descriptions from the received channel outputs. A second focuses on the direct estimation of the source from the two noisy descriptions, without trying to estimate the single descriptions. Simulation results show a good robustness of the proposed decoding schemes against transmission errors.

6.1 Structure of the considered multiple description coder

The approaches of decoding of the source presented in the following may be applied to many MD coding schemes. Though, the MD coding scheme considered here is based on the previous video coder presented in Section 3.1, and thus performs a motion-compensated spatio-temporal DWT of the video frames. Redundancy is introduced before quantization, the balanced descriptions being produced thanks to a bit allocation based on the characteristics of the channel.

6.1.1 General structure

The joint source channel (JSC) coding scheme used here corresponds to the class of JSC where the redundancy is introduced before source coding as proposed in [GKD07] (see the Figure 6.1). Here, the joint encoder consists in (see the Figure 6.2):

- a spatio-temporal DWT of the source data used to generate the different balanced descriptions by duplication (the wavelet coefficients are

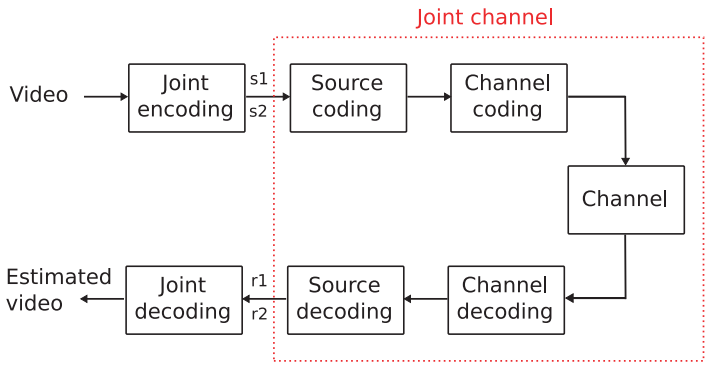


Figure 6.1: Proposed JSC coding scheme where the redundancy is introduced on the quantized wavelet coefficients during the bit allocation step (joint encoding box).

repeated in both the descriptions), as in [PAB03b]. In this case, the source pdf can be modeled by a generalized Gaussian [PAB03a];

- a model-based bit allocation, which dispatches the bits across the different subbands and the different descriptions, according to an information on the channel noise, and thus on the needed redundancy [PAB03b]. No predictive feedback is used;
- a scalar or vector quantization, followed by a fixed-length coding.

6.1.2 The bit allocation

The considered MDC scheme, based on the thesis work of M. Pereira [PAB03b], focus on the special case where a transmitter and a receiver are linked by two channels of equal capacity. Thus, this MDC scheme is a balanced MDC (BMDC). A BMDC framework generates descriptions of equal rate and importance. Explicit redundancy is introduced, so that each wavelet coefficient is transmitted more than once and coded with a different accuracy each time. The DWT is performed and then, the wavelet coefficients are repeated in both the descriptions. When a subband is finely coded in one description, the algorithm forces it to be coarsely coded in the other, as seen in Figure 6.3.

6.1.2.1 The bit allocation problem

The problem of the bit allocation is thus to find, for a given redundancy between the descriptions, the combination of scalar quantizers [SG88, Ort00] across the various wavelet coefficients subbands that will produce the minimum total central distortion D_0, while satisfying constraints on the side bit rates R_1 and R_2, and on the side distortions D_1 and D_2. This problem can

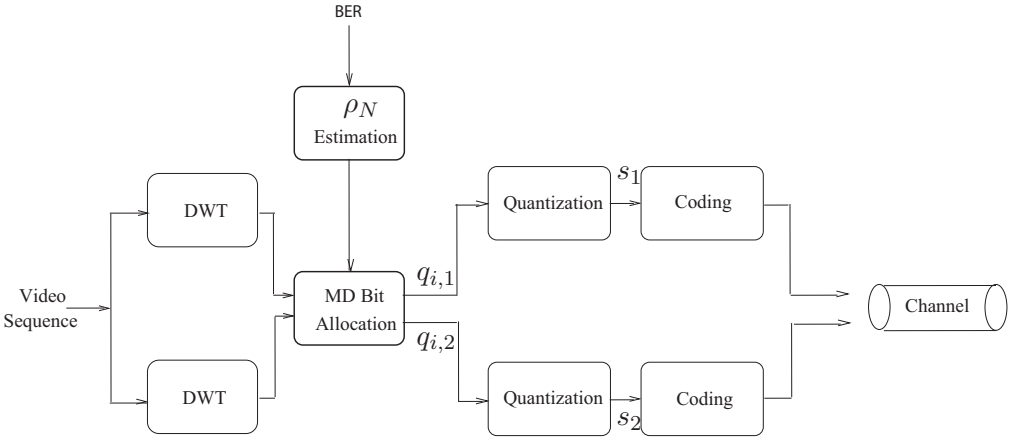


Figure 6.2: General coding scheme.

be resumed as:

$$(P)\begin{cases} \textbf{min } D_0\left(R_1, R_2\right) \\ \textbf{under the constraints } R_1 \leqslant \frac{R_T}{2} \text{ and } R_2 \leqslant \frac{R_T}{2} \\ \text{and } D_1(R_1) \leqslant D_m \text{ and } D_2(R_2) \leqslant D_m \end{cases}$$

with R_T the target output bit rate and D_m the maximal side distortion imposed, and

$$R_j = \sum_{i=1}^{N} a_i R_{i,j}\left(\tilde{q}_{i,j}\right), \forall j \in \{1,2\}, \tag{6.1}$$

with N the total number of wavelet subbands, $\tilde{q}_{i,j} = \frac{q_{i,j}}{\sigma_{i,j}}$, $q_{i,j}$ the quantization step and $\sigma_{i,j}$ the variance of the i-th subband. The parameter a_i in (6.1) is the size of the i-th subband divided by the size of the sequence, and $R_{i,j}(\tilde{q}_{i,j})$ is the output bit rate in bits per sample for the i-th subband.
One can also have:

$$D_j(R_j) = \sum_{i=1}^{N} \Delta_i w_i \sigma_{i,j}^2 D_{i,j}\left(\tilde{q}_{i,j}\right), \forall j \in \{1,2\}, \tag{6.2}$$

with $D_{i,j}\left(\tilde{q}_{i,j}\right)$ the quantization distortion of the i-th subband of description j, Δ_i an optional weight for frequency selection and w_i the weights of the

filter bank [B.U96] for the i-th subband.
Let us define the central distortion for the i-th subband as (see [Per04]):

$$D_{i,0}\left(\tilde{q}_{i,1},\tilde{q}_{i,2}\right)=\frac{1}{\sigma_{i,0}^2}\frac{1}{\rho_N+1}\left[\min\left(\sigma_{i,1}^2 D_{i,1}\left(\tilde{q}_{i,1}\right),\sigma_{i,2}^2 D_{i,2}\left(\tilde{q}_{i,2}\right)\right)\right.$$
$$\left.+\rho_N\times\max\left(\sigma_{i,1}^2 D_{i,1}\left(\tilde{q}_{i,1}\right),\sigma_{i,2}^2 D_{i,2}\left(\tilde{q}_{i,2}\right)\right)\right], \quad (6.3)$$

where ρ_N is a weighting parameter, called the *redundancy parameter* which returns an information on the channel noise. The redundancy parameter domain is $[0,1]$, $\rho_N=0$ is used when the channel is noiseless, and $\rho_N=1$ is used when a very noisy channel is expected. The amount of redundancy, *i.e.*, the importance of the redundant subbands, have to depend on the channel model. It determines the intermediate redundancies, and implicitly the intermediate values of the ρ_N parameter. An example of computation of this parameter will be presented in Section 6.2.2.2.

6.1.2.2 The functional to optimize

The Lagrangian functional J for the constrained optimization problem is then given by:

$$J=D_0+\sum_{j=1}^{2}\lambda_j F(R_j)+\sum_{j=1}^{2}\mu_j P(D_j), \quad (6.4)$$

with λ_j and μ_j some Lagrangian multipliers, and

$$D_0=\sum_{i=1}^{N}\Delta_i w_i \sigma_{i,0}^2 D_{i,0}\left(\tilde{q}_{i,1},\tilde{q}_{i,2}\right),$$

with $\sigma_{i,0}$ the variance of the i-th subband of the central description, and $D_{i,0}\left(\tilde{q}_{i,1},\tilde{q}_{i,2}\right)$ the central distortion of the i-th subband.

The constraint F_j on the bit rate can be expressed as, for the different descriptions $j=1,2$:

$$F_j=\left(\sum_{i=1}^{N}a_i R_{i,j}\left(\tilde{q}_{i,j}\right)-R_T/2\right),\forall j\in\{1,2\}. \quad (6.5)$$

Because balanced descriptions are wanted, a penalty on the side distortions is needed. By considering a constraint $x<0$, the penalty can be written as:

$$P(x)=\left(\frac{|x|+x}{2}\right)^2. \quad (6.6)$$

If the constraint is verified then $x<0$ and $P(x)=0$. Otherwise, $x\geqslant 0$ and $P(x)=x^2$. Considering the side distortions D_1, D_2 defined by (6.2), the constraint is:

$$(D_j-D_M)\leqslant 0. \quad (6.7)$$

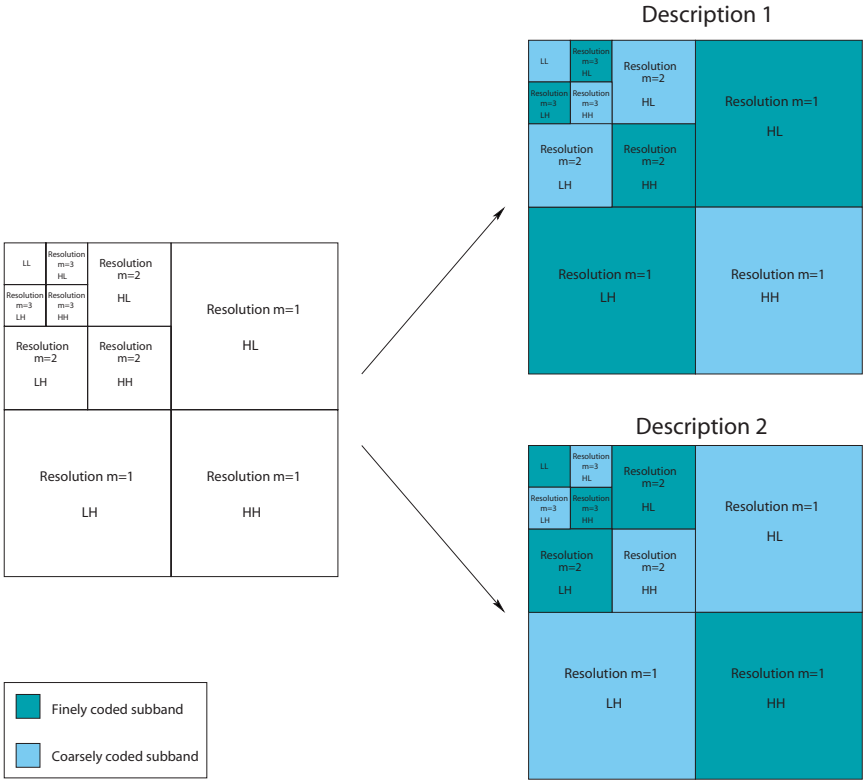


Figure 6.3: Example of division of the wavelet subbands between primary subbands (finely coded) and redundant subbands (coarsely coded) in the two descriptions.

The penalty P_j can thus be expressed as:

$$P_j = \left[\frac{|D_j(R_j) - D_M| + (D_j(R_j) - D_M)}{2} \right]^2, \forall j \in \{1, 2\}. \tag{6.8}$$

Considering the central distortion given by (6.3), the bit rate constraint (6.5), and the side distortion penalty (6.8), the Lagrangian functional (6.4) can be expressed as:

$$\begin{aligned} J &= \sum_{i=1}^{N} \Delta_i w_i \sigma_{i,0}^2 D_{i,0}\left(\tilde{q}_{i,1}, \tilde{q}_{i,2}\right) \\ &+ \sum_{j=1}^{2} \lambda_j \left(\sum_{i=1}^{N} a_i R_{i,j}\left(\tilde{q}_{i,j}\right) - R_T/2 \right) \\ &+ \sum_{j=1}^{2} \mu_j \left[\frac{|D_j(R_j) - D_M|}{2} + \frac{(D_j(R_j) - D_M)}{2} \right]^2 \end{aligned}$$

6.1.2.3 The solution

Finally, the resolution of the following system (with first order conditions) gives the optimal sets of quantization steps $\{\tilde{q}_{i,1}\}$, $\{\tilde{q}_{i,2}\}$, for a given ρ_N, and for the i-th subband:

$$\begin{cases} \dfrac{\partial D_{i,1}}{\partial R_{i,1}}(\tilde{q}_{i,1}) = \dfrac{-\lambda_1 a_i}{\Delta_i w_i \sigma_{i,1}^2 \left(\dfrac{C_{i,1}}{1+\rho_N} + \mu_1 E_1 \right)} & (\mathbf{a}) \\ \dfrac{\partial D_{i,2}}{\partial R_{i,2}}(\tilde{q}_{i,2}) = \dfrac{-\lambda_2 a_i}{\Delta_i w_i \sigma_{i,2}^2 \left(\dfrac{C_{i,2}}{1+\rho_N} + \mu_2 E_2 \right)} & (\mathbf{b}) \\ \sum_{i=1}^{N} a_i R_{i,1}(\tilde{q}_{i,1}) - R_T/2 = 0 & (\mathbf{c}) \\ \sum_{i=1}^{N} a_i R_{i,2}(\tilde{q}_{i,2}) - R_T/2 = 0 & (\mathbf{d}) \end{cases}$$

With the $C_{i,j}$ parameter defined as:

$$C_{i,j} = \begin{cases} 1, & \text{if } \min(\sigma_{i,1}^2 D_{i,1}(\tilde{q}_{i,1}), \sigma_{i,2}^2 D_{i,2}(\tilde{q}_{i,1})) = \sigma_{i,j}^2 D_{i,j}(\tilde{q}_{i,j}), \forall j \in \{1,2\} \\ \rho_N, & \text{otherwise.} \end{cases}$$

and the E_j parameter computed from:

$$E_j = \begin{cases} 2 \times \left(D_j(R_j) - D_M \right), & \text{if } D_j(R_j) > D_M, \forall j \in \{1,2\} \\ 0 & \text{otherwise.} \end{cases}$$

More details can be found in [PAB03b] or [Per04].

6.2 Multiple description decoding

In this thesis, contrary to the work of M. Pereira, the focus is not done in the coder side of the MDC scheme, but on the decoding part. As said in Section 5.7, the decoding of descriptions transmitted over error prone channels is a crucial problem and has not been so much explored in video transmission. In order to optimally decode the central description, two approaches have been implemented: the first one tries to construct the central description by first evaluating the two side descriptions from the received channel outputs, whereas the second one focuses on the direct estimation of the central description from the two noisy side descriptions, without trying to estimate these descriptions.

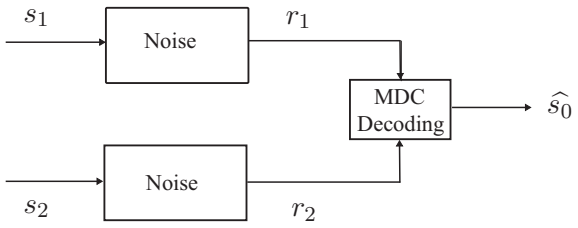


Figure 6.4: General scheme of the decoding of two noisy descriptions.

6.2.1 Problem statement

At the decoder, the challenge is to reconstruct a 'central' signal with central distortion D_0 as small as possible, using the knowledge of the two side descriptions. Figure 6.4 illustrates the problem. s_1 and s_2 represent the quantized source descriptions at emitter (quantization values issued from an optimal codebook $\boldsymbol{C}$ of M symbols before binary coding), and r_1 and r_2 represent the observed noisy descriptions at the receiver. The signal $\widehat{s_0}$ is the reconstructed signal obtained by decoding the two received descriptions (here with the system described in Section 6.1.2). In the following sections, two algorithms for optimal decoding are described.

6.2.2 Decoding using a Model-Based MAP and a decision approach

The problem of choosing the best description at decoder (resumed in Figure 6.5), for each symbol (*i.e.* here the quantized wavelet coefficients), can be seen as a *maximum a posteriori* (MAP) estimation problem, consisting in determining:

$$(s_1^*, s_2^*) = \arg\max_{s_0} \; p(s_1, s_2|r_1, r_2),$$

or equivalently by:

$$(s_1^*, s_2^*) = \arg\min_{s_1, s_2} \; -\log\left[p(s_1, s_2|r_1, r_2)\right].$$

Let recall that (according to Bayes' rule):

$$p(s|r) = \frac{p(s)p(r|s)}{p(r)},$$

or even:

$$p(s|r) \sim p(s)p(r|s),$$

where $\sim$ stands for proportional to, and since $p(r)$, the density of the observed value, is not taken into account in the optimization. In an equivalent way, the joint conditional density can be written as:

$$p(s_1, s_2|r_1, r_2) \sim p(s_1, s_2|r_2)p(r_1|s_1, s_2, r_2)$$

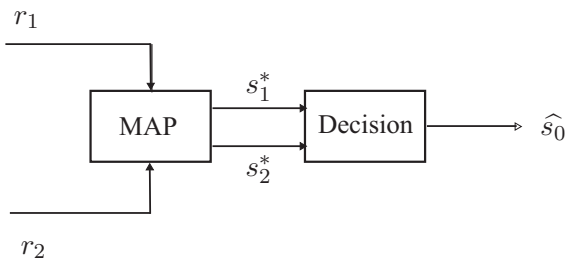


Figure 6.5: First proposed decoding approach: model-based MAP and decision approach.

Moreover,

$$p(r_1|s_1, s_2, r_2) = p(r_1|s_1),$$

since r_1, knowing s_1, can be supposed to not directly depend on the value of s_2 or the value of r_2. Thus,

$$\begin{aligned} p(s_1, s_2|r_1, r_2) &\sim p(s_1, s_2|r_2)p(r_1|s_1) \\ &\sim p(s_1, s_2)p(r_2|s_1, s_2)p(r_1|s_1) \\ &\sim p(s_1, s_2)p(r_1|s_1)p(r_2|s_2). \end{aligned}$$

The criterion to minimize is then given by:

$$(s_1^*, s_2^*) = \underset{s_1,s_2}{argmin} - \log\left[p(s_1, s_2)p(r_1|s_1)p(r_2|s_2)\right]. \tag{6.9}$$

The minimization of this criterion provides two optimal values s_1^* and s_2^*, for each coefficient, and two distortions called D_1 and D_2 with, classically:

$$D_i = \sum_{j\in\mathbf{C}} p(r_i|s_j) \int_{s_i^*-\frac{q}{2}}^{s_i^*+\frac{q}{2}} (x - s_i^*)^2 p_S(x)dx,$$

where q is the quantization step, and $p_S(x)$ is the probability density function (pdf) of the source signal (*i.e.* the wavelet subbands) from which the two descriptions s_1 and s_2 are obtained. Note that in the case where the noise is introduced by a channel assumed to be memoryless, $\sum_{j\in\mathbf{C}} p(r_i|s_j)$ corresponds to the sum of the transition probabilities over all the possible inputs. In that case, D_i corresponds to the distortion introduced by the source quantizer and the channel.
Then, $\widehat{s_0}$, the optimal reconstruction value at decoding, is set to s_1^* or s_2^* according to the values of D_1 and D_2 and using the following rule:

$$\begin{cases} \widehat{s_0} = s_1^* \text{ if } \min(D_1, D_2) = D_1 \\ \widehat{s_0} = s_2^* \text{ else if.} \end{cases}$$

The evaluation of the term $p(s_1, s_2)$ is presented in what follows.

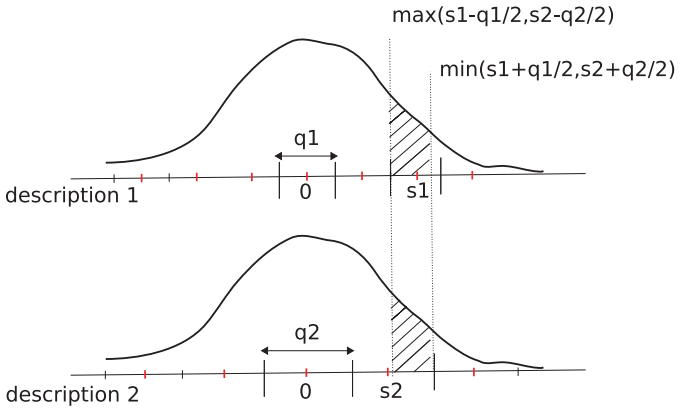


Figure 6.6: Interval containing the source data quantized by s_1 in description 1 and by s_2 in description 2.

6.2.2.1 *Evaluation of* $p(s_1, s_2)$

In the case of uniform scalar quantization, it is possible to give an expression of $p(s_1, s_2)$ in function of the quantization steps and of the source pdf. As it can be seen in Figure 6.6, this density can be evaluated as:

$$p(s_1, s_2) = \int_{\max(s_1-\frac{q_1}{2}, s_2-\frac{q_2}{2})}^{\min(s_1+\frac{q_1}{2}, s_2+\frac{q_2}{2})} p_S(x)dx \quad \text{with } (s_1, s_2) \in \mathbf{C}_1 \times \mathbf{C}_2,$$

where $\mathbf{C}_i$ is the quantization codebook for the description i.
This relation holds if $\max(s_1-\frac{q_1}{2}, s_2-\frac{q_2}{2})$ is smaller than $\min(s_1+\frac{q_1}{2}, s_2+\frac{q_2}{2})$. If it is not the case, the value of $p(s_1, s_2)$ is set to zero.

Since the function $p(s_1, s_2)$ depends on a 'min' and a 'max', four different cases have to be considered. Let first define the quantity $\Delta_S = s_1 - s_2$.

- **Case 1:** $\max(s_1-\frac{q_1}{2}, s_2-\frac{q_2}{2}) = s_1-\frac{q_1}{2}$ and $\min(s_1+\frac{q_1}{2}, s_2+\frac{q_2}{2}) = s_1+\frac{q_1}{2}$. That is to say $\Delta_S \geq \frac{1}{2}(q_1 - q_2)$ and $\Delta_S < \frac{1}{2}(q_2 - q_1)$. These conditions hold for $q_2 \geq q_1$. Then:

$$p(s_1, s_2) = \int_{s_1-\frac{q_1}{2}}^{s_1+\frac{q_1}{2}} p_S(x)dx,$$

- **Case 2:** $\max(s_1-\frac{q_1}{2}, s_2-\frac{q_2}{2}) = s_1-\frac{q_1}{2}$ and $\min(s_1+\frac{q_1}{2}, s_2+\frac{q_2}{2}) = s_2+\frac{q_2}{2}$. That is to say $\Delta_S \geq \frac{1}{2}(q_1 - q_2)$ and $\Delta_S > \frac{1}{2}(q_2 - q_1)$. These conditions hold for all q_1 and q_2. Then:

$$p(s_1, s_2) = \int_{s_1-\frac{q_1}{2}}^{s_2+\frac{q_2}{2}} p_S(x)dx,$$

- **Case 3:** $\max(s_1-\frac{q_1}{2}, s_2-\frac{q_2}{2}) = s_2-\frac{q_2}{2}$ and $\min(s_1+\frac{q_1}{2}, s_2+\frac{q_2}{2}) = s_1+\frac{q_1}{2}$. That is to say $\Delta_S \leq \frac{1}{2}(q_1 - q_2)$ and $\Delta_S < \frac{1}{2}(q_2 - q_1)$. These conditions hold for all q_1 and q_2. Then:

$$p(s_1, s_2) = \int_{s_2-\frac{q_2}{2}}^{s_1+\frac{q_1}{2}} p_S(x)dx,$$

- **Case 4:** $\max(s_1-\frac{q_1}{2}, s_2-\frac{q_2}{2}) = s_2-\frac{q_2}{2}$ and $\min(s_1+\frac{q_1}{2}, s_2+\frac{q_2}{2}) = s_2+\frac{q_2}{2}$. That is to say $\Delta_S \leq \frac{1}{2}(q_1 - q_2)$ and $\Delta_S > \frac{1}{2}(q_2 - q_1)$. These conditions hold for $q_1 \geq q_2$. Then:

$$p(s_1, s_2) = \int_{s_2-\frac{q_2}{2}}^{s_2+\frac{q_2}{2}} p_S(x)dx,$$

6.2.2.2 Channel model

To compute (6.9), one has to evaluate $p(r_i|s_i)$, $i = 1, 2$. Let define $\mathbf{u}(s)$ as a function which represents the source s after quantization, fixed-length M-bit binary indexation, and BPSK signalling. Let $\{-1, 1\}^M$ be the set of all values which may be taken by $\mathbf{u}(s)$. Then one has:

$$\begin{aligned} p(r_i|s_i) &= \sum_{\mathbf{u}\in\{-1,1\}^M} p(r_i, \mathbf{u}|s_i) \\ &= \sum_{\mathbf{u}\in\{-1,1\}^M} p(r_i|\mathbf{u}, s_i)\, p(\mathbf{u}|s_i). \end{aligned} \tag{6.10}$$

In (6.10), $p(\mathbf{u}|s_i)$ is directly determined from the quantization, indexation, and modulation of s_i, *i.e.*,

$$p(\mathbf{u}|s_i) = \begin{cases} 1 & \text{if } \mathbf{u} = \mathbf{u}(s_i) \\ 0 & \text{else.} \end{cases}$$

Thus, (6.10) simplifies to:

$$p(r_i|s_i) = p(r_i|\mathbf{u}(s_i), s_i).$$

For what concerns the channel output, $\mathbf{u}(s_i)$ provides as much information on r_i as s_i does ($s_i \rightarrow \mathbf{u}(s_i) \rightarrow r_i$ forms a Markov chain), thus, one finally gets:

$$p(r_i|s_i) = p(r_i|\mathbf{u}(s_i)). \tag{6.11}$$

Assuming that the channel is zero-mean white Gaussian with noise variance σ^2, (6.11) becomes:

$$p(r_i|s_i) = \left(2\pi\sigma^2\right)^{-M/2} \exp\left(-\frac{|r_i - \mathbf{u}(s_i)|^2}{2\sigma^2}\right). \tag{6.12}$$

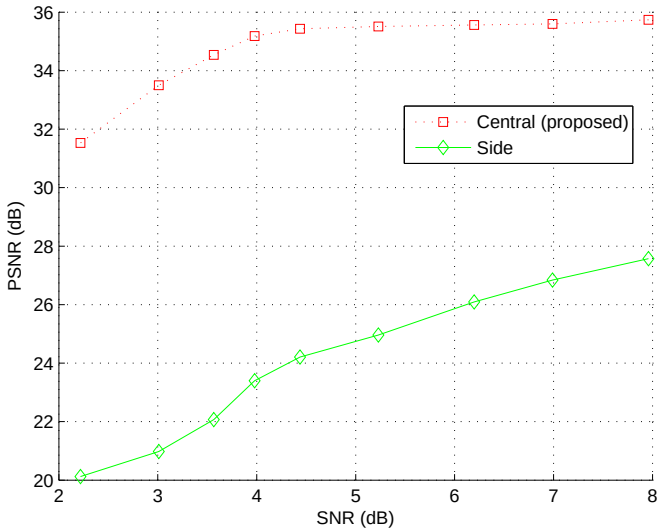


Figure 6.7: First approach: PSNR comparisons for FOREMAN between the side noisy description and the central description, bit-rate $R_t = 2$ Mbps.

It can be noted that, for such a channel, the redundancy parameter ρ_N is computed as:

$$\rho_N = 1 - \frac{B\, log_2(1 + \frac{S}{N})}{2},$$

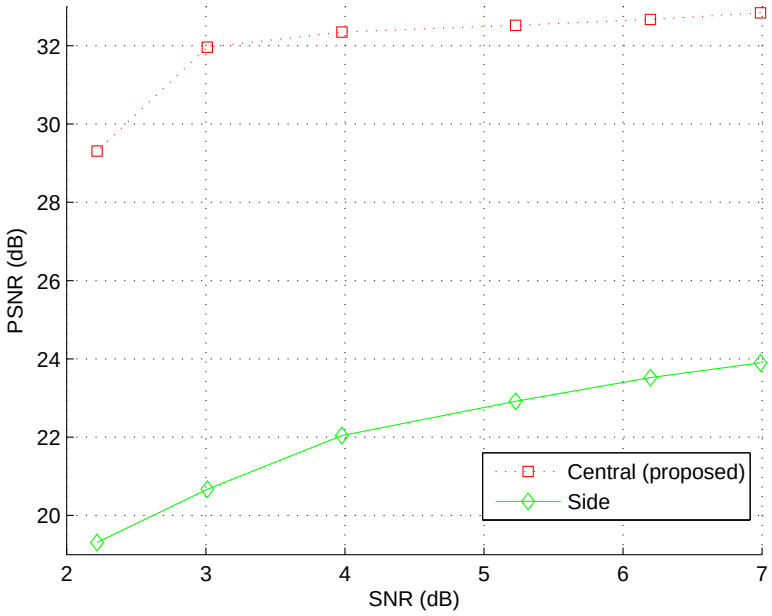
with B the channel bandwidth in symbol/s, and $\frac{S}{N}$ the SNR (where S is the received signal power and N is the AWGN power within the channel bandwidth). The details of computation for this expression can be found in [Per04].

Of course, other types of channel could also be considered, but the focus here is only done on this special case.

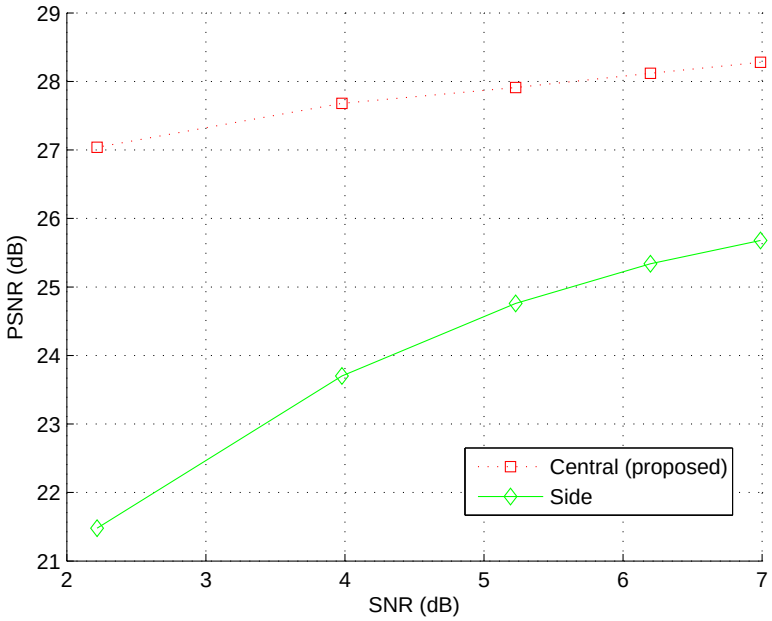
6.2.2.3 Experimental results of the first approach

The experiments have been done in different conditions, on the CIF sequences FOREMAN and ERIK, and on the SD sequence CITY, with 3 temporal decomposition levels, and with quarter-pixelic motion vectors. Each wavelet coefficient is quantized with a scalar quantization, encoded, and transmitted using BPSK signalling over an AWGN channel, with a noise variance σ^2. Headers and motion information are assumed noise free.

Figure 6.7 presents some PSNR comparisons between the side noisy description and the central description obtained with the proposed algorithm, for FOREMAN. PSNR values are presented for a bit-rate of 2 Mbps, and for different SNR values, induced by the channel noise. Figures 6.8(a) and 6.8(b) present the same kind of results for the sequences ERIK ($R_t = 2$



(a) ERIK.



(b) CITY.

Figure 6.8: First approach: PSNR comparisons for ERIK ($R_t = 2$ Mbps) and CITY ($R_t = 2.5$ Mbps) between the side noisy description and the central description.

Foreman 1.5 Mbps	**Side**	**Central**	**Foreman 2 Mbps**	**Side**	**Central**
$SNR = 3$	20.67	30.34	$SNR = 3$	20.98	33.50
$SNR = 7$	23.74	33.72	$SNR = 7$	26.84	35.6
Erik 1.5 Mbps	**Side**	**Central**	**Erik 2 Mbps**	**Side**	**Central**
$SNR = 2$	19.03	28.23	$SNR = 2$	19.31	29.31
$SNR = 6$	24.15	31.32	$SNR = 6$	22.92	32.67
City 2.5 Mbps	**Side**	**Central**	**City 2.8 Mbps**	**Side**	**Central**
$SNR = 4$	23.7	27.68	$SNR = 4$	24.12	28.21
$SNR = 7$	25.68	28.28	$SNR = 7$	25.88	28.35

Table 6.1: First approach: PSNR (dB) comparisons between the side description and the central description obtained with the first approach; for the sequences FOREMAN, ERIK and CITY (on three (2,0) decomposition levels, with quarter-pixel motion vectors), for different bit-rates, and with different values of SNR.

Mbps) and CITY ($R_t = 2.5$ Mbps). The gain in PSNR is very important: the performances in respect with the noisy side description can be improved up to 11 dB.

Some interesting visual results are then presented. Figure 6.9(a) shows, for a SNR equal at 3 dB, reconstructed images of FOREMAN, at $R_t = 2$ Mbps, for the noiseless images, the noisy side description and for the central description, obtained by applying the proposed MAP decoding algorithm. Figures 6.9(b) and 6.9(c) show the same results for ERIK ($SNR = 4$ dB, $R_t = 2$ Mbps) and CITY ($SNR = 4$ dB, $R_t = 2.5$ Mbps). The central descriptions (j) are better preserved, thanks to the proposed algorithm. The noiseless images are almost retrieved.

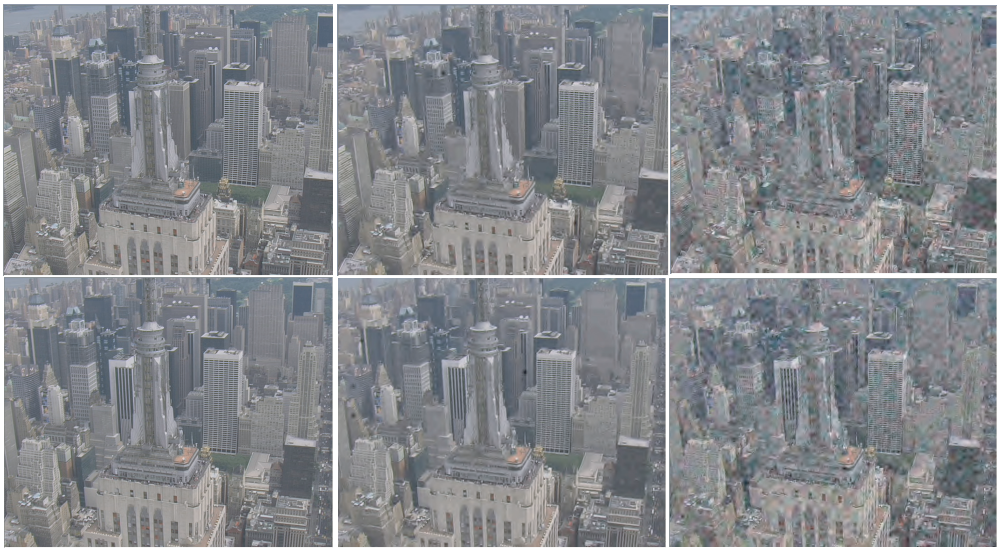
In Table 6.1 are also summarized the PSNR comparisons between the side description and the central description obtained by applying the first approach, for different values of SNR, with the same coding parameters than previously for FOREMAN, ERIK and CITY. The results are really good, the PSNR of the central description is always widely higher than the one of the side description.

6.2.3 Decoding by a direct estimation of the central description

A different method of decoding can be considered, based on a direct evaluation of the value $\widehat{s}_0$ of the central description.

(a) FOREMAN, $SNR = 3$ dB, $R_t = 2$ Mbps, images 13 and 117.

(b) ERIK, $SNR = 4$ dB, $R_t = 2$ Mbps, images 13 and 44.



(c) CITY, $SNR = 4$ dB, $R_t = 2.5$ Mbps, images 18 and 44.

Figure 6.9: First approach: visual results, (i) noiseless images, (j) central description, (k) side description.

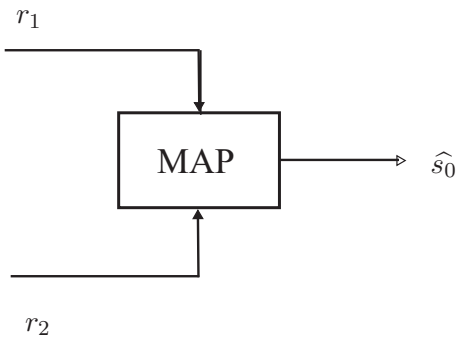


Figure 6.10: Second proposed decoding approach: direct estimation of the central description.

6.2.3.1 Estimation of the solution

The estimate $\widehat{s}_0$ of the central description is obtained from the probability of s_0 knowing the channel outputs r_1 and r_2 (see Figure 6.10):

$$\widehat{s}_0 = \arg\max_{s_0} p\left(s_0|r_1, r_2\right), \tag{6.13}$$

and can be expressed as:

$$\widehat{s}_0 = \arg\max_{s_0} \sum_{s_1,s_2} p\left(s_0, s_1, s_2|r_1, r_2\right).$$

Using Bayes' rule, one can have:

$$\widehat{s}_0 = \arg\max_{s_0} \sum_{s_1,s_2} \frac{p(r_1, r_2|s_0, s_1, s_2)p(s_0, s_1, s_2)}{p(r_1, r_2)},$$

and:

$$\widehat{s}_0 = \arg\max_{s_0} \sum_{s_1,s_2} p(r_1, r_2|s_1, s_2)p(s_1, s_2|s_0)p(s_0),$$

and finally, since the two channels are independent:

$$\widehat{s}_0 = \arg\max_{s_0} \sum_{s_1,s_2} p\left(r_1|s_1\right) p\left(r_2|s_2\right) p\left(s_1, s_2|s_0\right) p\left(s_0\right). \tag{6.14}$$

$p(s_0)$ is the pdf of the source (*i.e.* in the case of wavelet subbands, a generalized Gaussian [PAB03a]). $p(r_i/s_i)$ are the transition probabilities of the channel. $p(s_1, s_2/s_0)$ is computed thanks to the values of quantization of the side descriptions. More precisely:

$$\begin{cases} p(s_1, s_2/s_0) = 1 \text{ if } s_0 \in P_1 \cap P_2 \\ p(s_1, s_2/s_0) = 0 \text{ if not,} \end{cases} \tag{6.15}$$

with P_1 and P_2 the quantization intervals associated to s_1 and s_2.

6.2.3.2 Channel model

In the same channel conditions as in the previous sections, the same expression as in 6.2.2.2 is used for the channel model:

$$p\left(r_i|\mathbf{u}\left(s_i\right)\right) = \left(2\pi\sigma^2\right)^{-M/2} \exp\left(-\frac{\left|r_i - \mathbf{u}\left(s_i\right)\right|^2}{2\sigma^2}\right).$$

Then, this equation and (6.15) may be combined in (6.14) to get the cost function

$$J\left(s_0\right) = p(s_0) \sum_{s_1,s_2} \left(\exp\left(-\frac{\left|r_1 - \mathbf{u}\left(s_1\right)\right|^2 + \left|r_2 - \mathbf{u}\left(s_2\right)\right|^2}{2\sigma^2}\right) .p\left(s_1, s_2|s_0\right)\right), \tag{6.16}$$

which maximization leads to the estimation of s_0.
As s_i is totally determined by s_0 and by the quantization step chosen for the description i, one can write: $s_i = Q_i(s_0)$, and thus:

$$J\left(s_0\right) = \exp\left(-\frac{\left|r_1 - \mathbf{u}\left(Q_1(s_0)\right)\right|^2 + \left|r_2 - \mathbf{u}\left(Q_2(s_0)\right)\right|^2}{2\sigma^2}\right) .p\big(Q_1(s_0), Q_2(s_0)|s_0\big).p(s_0).$$

6.2.3.3 Experimental results of the second approach

As for the first approach, the experiments have been done in different conditions, using a Gaussian channel with a noise variance σ^2 in order to add noise at the 2 descriptions, on the CIF sequences FOREMAN and ERIK, and on the SD sequence CITY, with the same coding parameters.

Figure 6.11 presents some PSNR comparisons between the side noisy description and the central description obtained with the proposed algorithm of decoding of the central description, for FOREMAN. PSNR values are presented for a bit-rate of 2 Mbps, and for different SNR values. Figures 6.12 and 6.13 present the same kind of results for the sequences ERIK ($R_t = 2$ Mbps) and CITY ($R_t = 2.5$ Mbps). The gain in PSNR is here again important: the performances in respect with the noisy side description can be improved up to 9 dB.

Visual results are also presented. Figure 6.14(a) shows, for a SNR equal at 3 dB, reconstructed images of FOREMAN, at $R_t = 2$ Mbps, for the noiseless images, the noisy side description and for the central description, obtained by applying the second approach. Figures 6.14(b) and 6.14(c) show the same results for ERIK ($SNR = 4$ dB, $R_t = 2$ Mbps) and CITY ($SNR = 4$

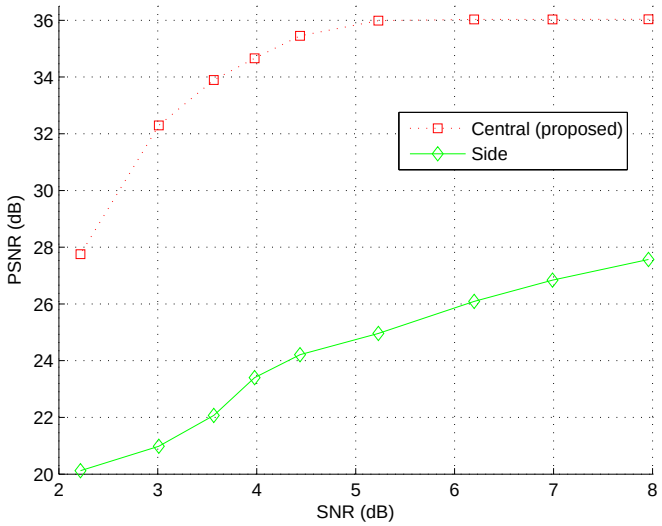


Figure 6.11: Second approach: PSNR comparisons for FOREMAN between the side noisy description and the central description, bit-rate $R_t = 2$ Mbps.

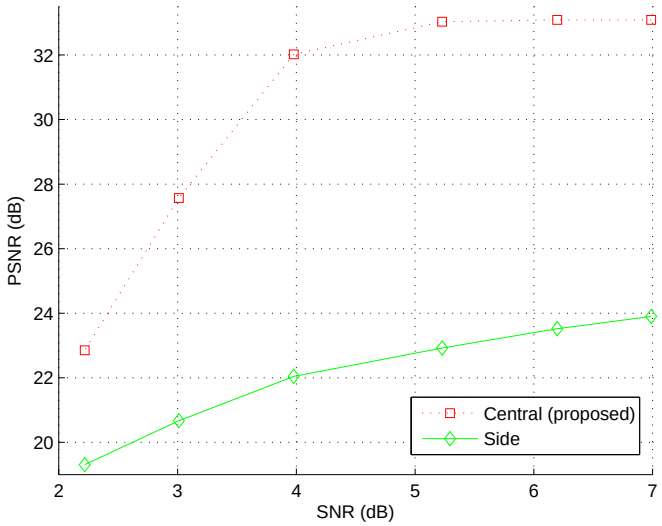


Figure 6.12: Second approach: PSNR comparisons for ERIK between the side noisy description and the central description, bit-rate $R_t = 2$ Mbps.

dB, $R_t = 2.5$ Mbps). Here again, the central descriptions (j) are better preserved, thanks to the second approach.

In Table 6.2 are summarized the PSNR comparisons between the side

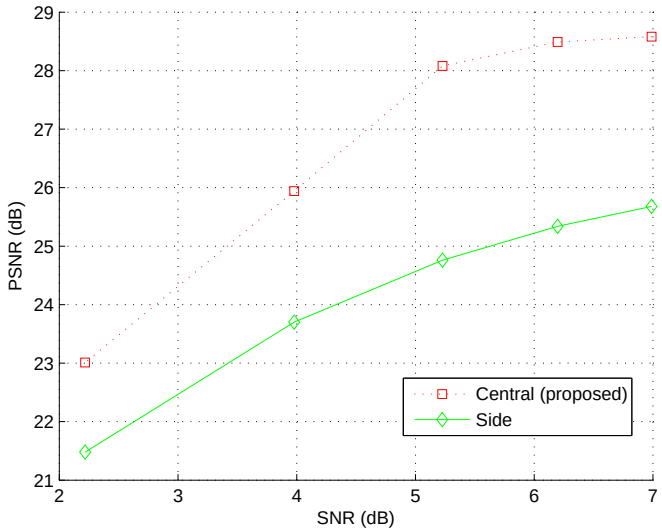


Figure 6.13: Second approach: PSNR comparisons for CITY between the side noisy description and the central description, bit-rate $R_t = 2.5$ Mbps.

description and the central description obtained by applying the second approach of decoding, for different values of SNR, with the same coding parameters than previously for FOREMAN, ERIK and CITY. The results are good, the PSNR of the central description is always higher than the one of the side description.

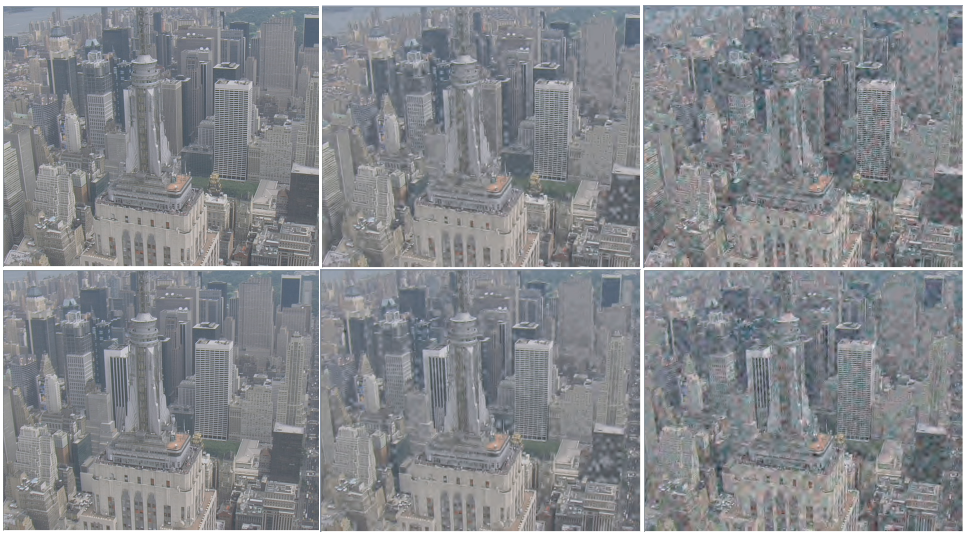
6.2.4 Comparison between the two approaches

Let's do a brief comparison between the two proposed approaches of optimal decoding. Table 6.3 presents some results of central descriptions obtained with the two approaches, for the sequences FOREMAN, ERIK and CITY, with the same coding parameters. The noiseless references are presented into parentheses. For low channel noises, the second approach gives better results than the first one. On the contrary, for higher noises, the first approach allows a much better reconstruction of the central description.

This behaviour can be explained by the fact that the first approach better describes the source and the MDC scheme, thanks to the joint density $p(s_1, s_2)$. It can be however remarked that the first approach is more complex in terms of time computation, especially because of the calculus of the joint density of the two descriptions.

(a) FOREMAN, $SNR = 3$ dB, $R_t = 2$ Mbps, images 13 and 117.

(b) ERIK, $SNR = 4$ dB, $R_t = 2$ Mbps, images 13 and 44.



(c) CITY, $SNR = 4$ dB, $R_t = 2.5$ Mbps, images 18 and 44.

Figure 6.14: Second approach: visual results, (i) noiseless images, (j) central description, (k) side description.

Foreman 1.5 Mbps	**Side**	**Central**	**Foreman 2 Mbps**	**Side**	**Central**
$SNR = 3$	20.67	29.13	$SNR = 3$	20.98	32.39
$SNR = 7$	23.74	34.00	$SNR = 7$	26.84	36.03
Erik 1.5 Mbps	**Side**	**Central**	**Erik 2 Mbps**	**Side**	**Central**
$SNR = 2$	19.03	20.59	$SNR = 2$	19.31	22.85
$SNR = 6$	24.15	31.74	$SNR = 6$	22.92	33.03
City 2.5 Mbps	**Side**	**Central**	**City 2.8 Mbps**	**Side**	**Central**
$SNR = 4$	23.7	25.94	$SNR = 4$	24.12	26.58
$SNR = 7$	25.68	28.58	$SNR = 7$	25.88	28.61

Table 6.2: Second approach: PSNR (dB) comparisons between the side description and the central description obtained with the first approach; for the sequences FOREMAN, ERIK and CITY (on three (2,0) decomposition levels, with quarter-pixel motion vectors), for different bit-rates, and with different values of SNR.

Foreman 1.5 Mbps (34.05)	**Approach 1**	**Approach 2**	**Foreman 2 Mbps** (36.07)	**Approach 1**	**Approach 2**
$SNR = 3$	30.34	29.13	$SNR = 3$	33.50	32.39
$SNR = 7$	33.72	34.00	$SNR = 7$	35.60	36.03
Erik 1.5 Mbps (31.80)	**Approach 1**	**Approach 2**	**Erik 2 Mbps** (33.10)	**Approach 1**	**Approach 2**
$SNR = 2$	28.23	20.59	$SNR = 2$	29.31	22.85
$SNR = 6$	31.32	31.74	$SNR = 6$	32.67	33.03
City 2.5 Mbps (28.61)	**Approach 1**	**Approach 2**	**City 2.8 Mbps** (28.67)	**Approach 1**	**Approach 2**
$SNR = 4$	27.68	25.94	$SNR = 4$	28.21	26.58
$SNR = 7$	28.28	28.58	$SNR = 7$	28.35	28.61

Table 6.3: PSNR (dB) comparisons for the central description between the two approaches of decoding (the noiseless references are presented into parentheses); for the sequences FOREMAN, ERIK and CITY (on three (2,0) decomposition levels, with quarter-pixel motion vectors), for different bit-rates, and with different values of SNR.

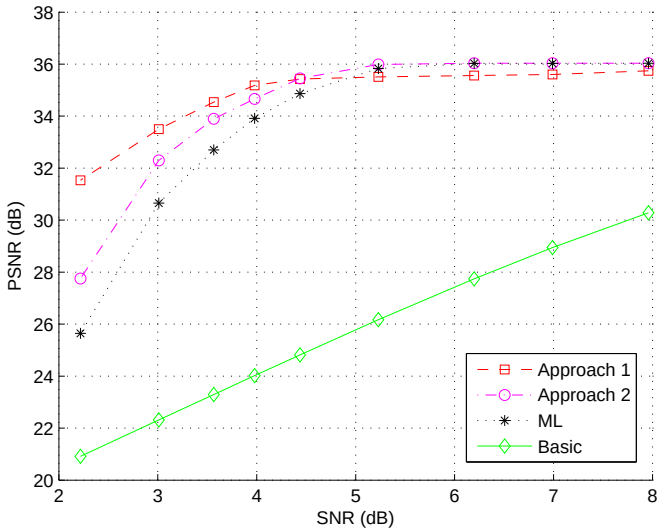


Figure 6.15: Performances comparison between the proposed approaches and the ML estimations, with the results in the basic case, for FOREMAN, at 2 Mbps.

6.2.5 *Comparison with some other methods*

In this section is presented a comparison, for FOREMAN (encoded using three temporal decomposition levels and a motion-compensation with quarter-pixelic motion vectors), between the two proposed decoding approaches, and some state-of-the-art methods.

Figure 6.15 presents curves for the PSNR of the obtained central description using the proposed decoding algorithms, the Maximum Likelihood (ML) estimator, and the basic reconstruction of the description, for different values of SNR on the channel. The ML estimator can be expressed as:

$$\widehat{s}_0 = \arg\max_{s_0} p\left(r_1, r_2 | s_0\right),$$

and after calculus and with the same hypothesis assumed in Section 6.2.3.2:

$$\widehat{s}_0 = \arg\min_{s_0} \left(|r_1 - \mathbf{u}\left(Q_1(s_0)\right)|^2 + |r_2 - \mathbf{u}\left(Q_2(s_0)\right)|^2 \right).$$

The basic reconstruction of the central description corresponds to a hard decision on the values of s_1 and s_2: the less quantized description is chosen.

The PSNR of the central descriptions obtained with the proposed approaches is higher than those obtained with the ML algorithm (up to 6 dB) or with the basic reconstruction (up to 11 dB), especially in the case of a high channel noise. Figure 6.16 shows some frames resulting from the central descriptions, for FOREMAN at $R_t = 2$ Mbps, with a SNR of 3 dB. The

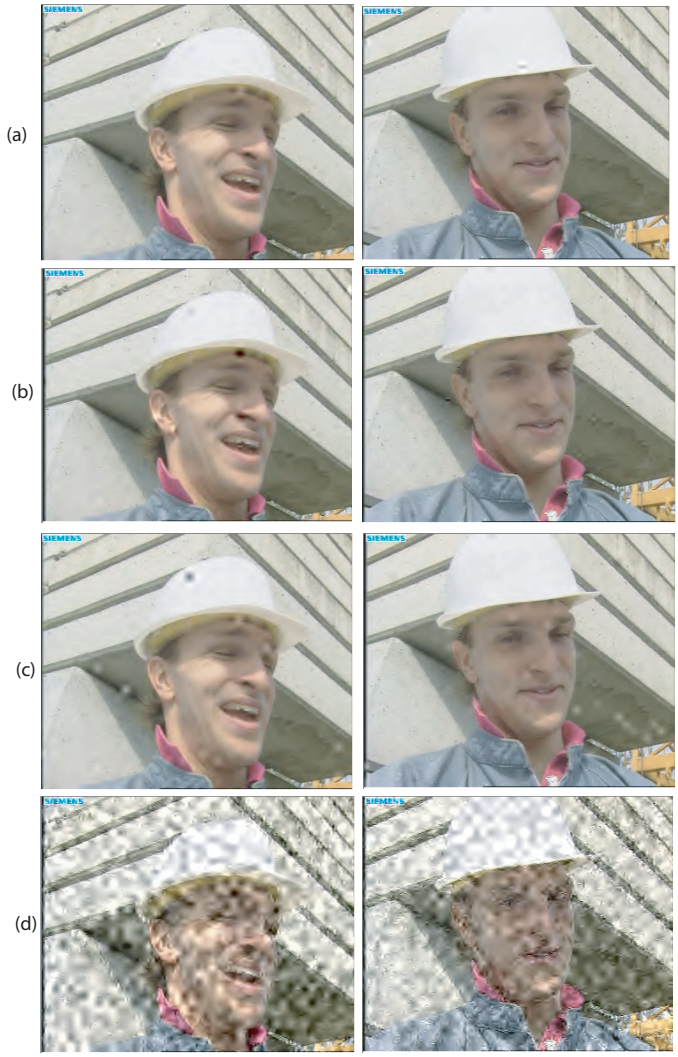


Figure 6.16: Visual comparisons, FOREMAN, $SNR = 3dB$, $R_t = 2$ Mbps, images 13 and 117, (a) central description obtained with the first proposed approach, (b) central description obtained with the second proposed approach, (c) central description obtained with ML, (d) "basic" central description.

central descriptions (a) and (b) obtained with the MAP algorithm are much better reconstructed, and much closer to the signal that would be obtained without channel noise, compared to the central description produced by the ML algorithm or the basic reconstruction.

6.3 Conclusion

An optimal decoding of MD encoded video sequences when compressed data are transmitted over channels introducing noise at the bit level has been proposed. The MD coding scheme includes a scan-based DWT and a bit allocation that dispatches the redundancy between the different descriptions. The amount of redundancy depends on the channel characteristics.

Two approaches have been explored [AAC+08]: the estimation of the two generated descriptions from the received channel outputs [AA08], and the direct estimation of the source from the two noisy descriptions [AAK09]. Experimental results for the both approaches are interesting. Even with descriptions strongly degraded by channel noise, the signal estimated using the proposed approaches is of good visual quality, close to the one of the original signal without channel noise. The results obtained are much better than those obtained with the ML algorithm.

Chapter 7

Distributed video coding

Distributed Video Coding (DVC) is a new and very interesting paradigm in video coding which proposes to move the computation complexity from the encoder to the decoder. This can be useful in many industrial applications such as video compression on mobile devices, multi-sensor system, etc ... This approach has been explored in the framework of a national project, ESSOR [ess09], in collaboration with other research teams.

After drawing a brief state-of-the-art, an efficient method of frame interpolation for Wyner-Ziv video coding is proposed.

7.1 A brief state-of-the-art

Distributed source coding (DSC) has emerged as an enabling technology for sensor networks. It refers to the compression of correlated signals captured by different sensors which do not communicate between themselves. All the signals captured are compressed independently and transmitted to a central base station which has the capability to decode them jointly. Video compression has been recast into a distributed source coding framework leading to distributed video coding (DVC) systems targeting low coding complexity and error resilience. A comprehensive survey of first DVC solutions can be found in [GARRM05]. The use of multiple description techniques could also be interesting in order to make the DVC system robust to transmission noise.

7.1.1 Distributed Source Coding: Theoretical Background

DSC finds its foundation in the seminal Slepian-Wolf (SW) [SW73] and Wyner-Ziv (WZ) [WZ76] theorems. In this section, the principles of Slepian-Wolf and Wyner-Ziv coding as well as the corresponding rate bounds are first reviewed.

7.1.1.1 Slepian-Wolf coding

Let X and Y be two binary correlated memoryless sources to be losslessly encoded. General set-up for Slepian-Wolf coding is presented at 7.1(a). If the two coders communicate, it is well known from Shannon's theory that the minimum lossless rate for X and Y is given by the joint entropy $H(X,Y)$. Slepian and Wolf have established in 1973 [SW73] that this lossless compression rate bound can be approached with a vanishing error probability for long sequences, even if the two sources are coded separately, provided that they are decoded jointly and that their correlation is known to both the encoder and the decoder. The achievable rate region is thus defined by $R_X \geq H(X|Y)$, $R_Y \geq H(Y|X)$ and $R_X + R_Y \geq H(X,Y)$, where $H(X|Y)$ and $H(Y|X)$ denote the conditional entropies between the two sources. This region is shown in Figure 7.1(b). Let us consider the particular case where Y is available at the decoder, and has been coded without information on the source X at its entropy rate $H(Y)$. According to the SW theorem, the source X can be coded losslessly at a rate arbitrarily close to the conditional entropy $H(X|Y)$ (which is function of the innovation of X given Y), if the sequence length tends to infinity. The minimum total rate for the two sources is thus $H(Y) + H(X|Y) = H(X,Y)$. This set-up where one source is transmitted at full rate (e.g., $R_Y = H(Y)$) and used as side information (SI) to decode the other one (implying $R_X = H(X|Y)$ or reciprocally) corresponds to one of the corner points of the SW rate region. This operating point is called the asymmetric case and corresponds to the point A (and reversely point B) in Figure 7.1(b).

7.1.1.2 Lossy coding of correlated sources

Lossy distributed data compression represents the extension of the Slepian-Wolf setup to the case of reconstruction of the sources under a fidelity criterion.

Let $\{(X_k, Y_k)\}_{k=1}^{\infty}$ be a sequence of independent drawings of the correlated pair $(X_k, Y_k) \sim q(x,y)$. The streams $\{X_k\}$ and $\{Y_k\}$ are encoded separately, the outputs of the encoders being binary sequences at rates R_1 and R_2 bits per input symbol respectively. The decoder observes the encoded streams jointly, and produces the reconstruction sequences $\{\hat{X}_k\}$ and $\{\hat{Y}_k\}$.

Let $d(X, \hat{X})$ be the chosen measure of the distortion between two random variables. The achievable rate region $\mathcal{R}$ under the fidelity criterion is defined as the set of pairs $(R_1(D_1), R_2(D_2))$ that allow $E[d(X, \hat{X})] \leq D_1$ and $E[d(Y, \hat{Y})] \leq D_2$. The Slepian-Wolf theorem defines the achievable rate region for the important case $D_1 = D_2 = 0$.

Suppose now that the source stream $\{Y_k\}$ could be directly available at the joint decoder: this specification of the general setup is known as Wyner-Ziv coding. The achievable rate region $\mathcal{R}$ is defined as the set of rates

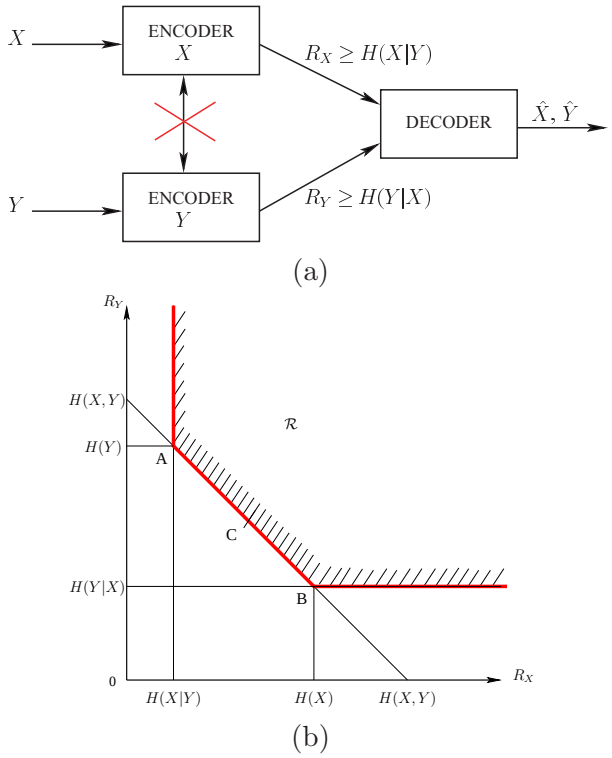


Figure 7.1: Distributed coding of statistically dependent i.i.d. discrete random sequences X and Y. Set-up (a); Achievable rate region (b).

$R_{WZ}(D)$ that allow the joint decoder to provide an estimate $\{\hat{X}_k\}$ such as $E[d(X, \hat{X})] \leq D$.

This result can be generalized to the N-dimensional case [CT91] and also to the case of any jointly ergodic source [Cov75].

7.1.1.3 Wyner-Ziv coding

Wyner and Ziv focus in [WZ76] on the determination of the rate-distortion function $R^*_{X|Y}(D)$ for the setup of lossy coding of a source with side information at the decoder, as depicted in Figure 7.2.

Let $\{(X_k, Y_k)\}_{k=1}^n$ be the sequence of independent drawings of the correlated pair $(X_k, Y_k) \sim q(x, y)$, over the alphabets $\mathscr{X}$ and $\mathscr{Y}$. The encoding process of the sequence $\{X_k\}$ produces a binary stream at the rate of R bits per input symbol. The decoder output is the sequence $\{\hat{X}_k\}_{k=1}^n$ that takes values on the reproduction alphabet $\hat{\mathscr{X}}$. The measure of fidelity of the reconstruction is evaluated as:

$$D = E[d(X, \hat{X})],$$

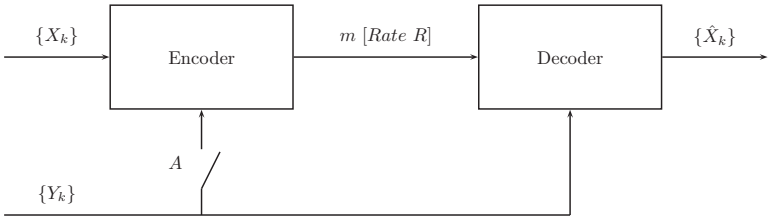


Figure 7.2: Coding of a source with side information.

where $d(X, \hat{X})$ is the measure of the distortion. The rate-distortion function represents the minimum rate at which the system can operate fulfilling the requirements on the fidelity of the reconstruction; it is defined as

$$R^*_{X|Y}(D) = \min_{R(D)\in\mathcal{R}} R(D),$$

where $\mathcal{R}$ is the achievable rate region.
Consider first the case when the switch A of Figure 7.2 is closed (case when the side information at the encoder and at the decoder): it is known from the classical Shannon theory that the rate-distortion function is given by

$$R_{X|Y}(D) = \min_{p(\hat{x}|x,y):E[d(X,\hat{X})]\leq D} I(X; \hat{X}|Y).$$

Consider now the case when the switch A is open (case when the side information is available at the decoder only). Define $\{Z_k\}_{k=1}^n$ as an auxiliary sequence of random variables $Z \in \mathscr{Z}$, such as the joint distribution $p(x, y, z)$ forms the Markov chain

$$Z \longrightarrow X \longrightarrow Y$$

(the variables Z and Y are conditionally independent, given X). The encoder and the decoder are defined by the mappings

$$\begin{aligned} f_E &: \mathscr{X}^n \longrightarrow \{1, 2, \ldots, 2^{nR}\}; \\ f_D &: \mathscr{X}^n \times \{1, 2, \ldots, 2^{nR}\} \longrightarrow \hat{\mathscr{X}}^n. \end{aligned}$$

The decoder function f_D can be thought as the composition of two functions $f_1 \circ f_2$; they are defined as the mappings

$$\begin{aligned} f_2 &: \mathscr{Y}^n \times \{1, 2, \ldots, 2^{nR}\} \longrightarrow \mathscr{Z}^n; \\ f_1 &: \mathscr{Y}^n \times \mathscr{Z}^n \longrightarrow \mathscr{Z}^n. \end{aligned}$$

The Wyner-Ziv theorem gives the general expression of the rate-distortion function for the lossy coding of a source with side information at the decoder as

$$R^*_{WZ}(D) = \min_{p(z|x)p(\hat{x}|y,z):E[d(X,\hat{X})]\leq D} \left[I(X; Z) - I(Y; Z)\right].$$

The achievability proof for the Wyner-Ziv theorem is based on a random coding argument, which does not provide any insight to the effective construction of good practical codes.

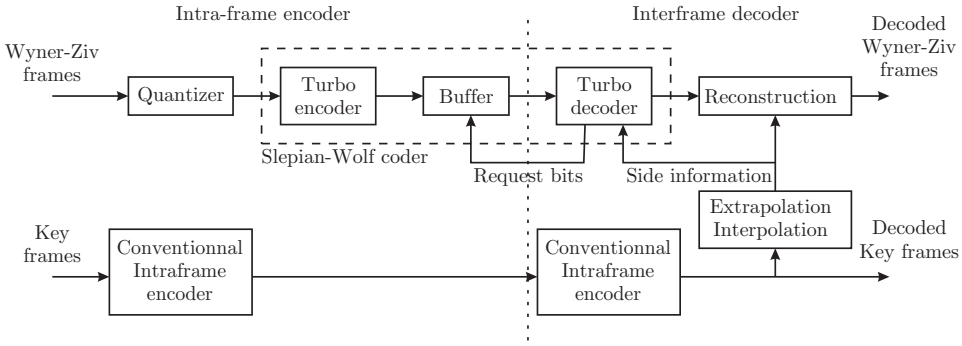


Figure 7.3: Pixel-domain Wyner-Ziv video coder.

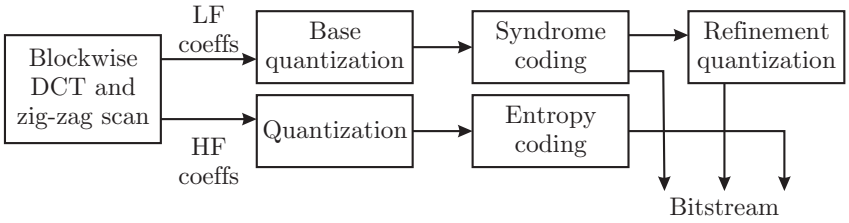
7.1.2 Video coders based on DSC

Video coders involving concepts from coding with side information present only at the decoder have been proposed very recently [ASG03, PR03]. One of the interesting property of these coding schemes is that they deport part of the encoding complexity of traditional video coders to the decoder.

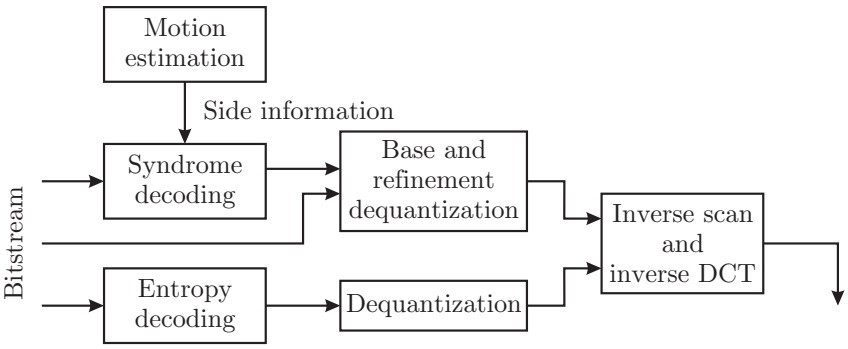
7.1.2.1 Wyner-Ziv video coding

In the first Wyner-Ziv video coder [ASG03], the regularly-place key frames (reference frames) are intra-encoded using a traditional video coder, such as H264. The remaining frames are Wyner-Ziv encoded in the pixel domain. Each pixel is quantized and the quantization indexes are then Slepian-Wolf encoded using a rate-compatible punctured turbo code (RCPT). The decoder generates the side information using the previously decoded key frames and Wyner-Ziv frames. Then, the turbo decoder combines the side information and the received parity bits (see Figure 7.3). If the decoding succeeds, the Wyner-Ziv frame is reconstructed using minimum mean squared-error reconstruction from the estimated output of the quantizer and the side information. If not, more bits are requested, requiring thus a feedback channel.

A first improvement of the previous scheme has been proposed in [ARSG04]: encoding is done in a transform domain, instead of the original pixel domain. A group of pixels are transformed using a DCT transform. Each subband is separately Wyner-Ziv encoded using again a RCPT. For high bit-rates, this scheme performs 2 dB better than the pixel-domain Wyner-Ziv scheme. A further improvement is the use at decoder-side of motion-estimated side information. When the motion is not perfectly smooth, the decoder has to be helped to get the best motion-estimated side information. In [WCO05], the authors propose a video coding framework based on Wyner-Ziv coding principles, in order to achieve efficient and low-complexity scalable coding. Based on a standard predictive coder, as base layer, the proposed Wyner-Ziv scalable coder can achieve good coding efficiency, by selectively exploiting



(a) The PRISM encoder.



(b) The PRISM decoder.

Figure 7.4: Functional diagram of the PRISM coder.

the high quality reconstruction of the previous frame in the enhancement layer coding of the current frame.

In [FYMPP07], a scheme of distributed coding of sequences transmitted through error-prone channels is presented, and in [YFMPP07], the authors present a method for determining the theoretical compression bound of this kind of coder, by taking into account the amount of motion and the transmission channel conditions.

7.1.2.2 PRISM coding

PRISM, for PoweR-effIcient Syndrome-base Multimedia coding, proposed in [PR03] is one of the first video coder involving concepts of coding with side information at the decoder. No time-consuming motion prediction is performed at the encoder side. The motion-prediction is done at the decoder side, searching for the block that provides the best side information. The functional diagram of the encoder is represented in Figure 7.4(a). As for classical video coders, each picture is divided into blocks of 8×8 or 16×16 pixels. These blocks are classified in several classes according to the variance of their innovation. Then a decorrelating transform is applied, followed by a scalar quantization of the obtained DCT coefficients. The step size of this first quantization corresponds to the desired reconstruction quality. The

quantized coefficients of low-frequency are then syndrome encoded using a trellis code. In order to verify the Wyner-Ziv decoding performs well, a CRC is evaluated for the sequence of base scalar quantized coefficients. This CRC is transmitted and checked at decoder side. A refinement quantization is then used to improve the reconstruction of the transformed coefficients. Then, the HF coefficients are INTRA-encoded, *i.e.*, zig-zag scanned and entropy-coded using run-length Huffman codes.

At decoder side (see Figure 7.4(b)), the bitstream is read. Several *candidates for the side information* are generated first. In [PR03], a standard H263+ half-pixel motion prediction is considered, but several other motion-prediction techniques could be used to provide better candidate for the side information. Then, the *syndrome decoding* is performed using each of the candidates for the side information. Once the side information has been identified, it is used with the quantized codeword sequence to get the best estimate of the LF DCT coefficients. Using the refinement bits and the INTRA-encoded coefficients, the whole set of DCT transformed coefficients are available. Then, an inverse DCT transform is performed. In [TO07], in a PRISM-based coder, the authors propose a rate-distortion analysis for a maximum likelihood method of motion estimation at the decoder.

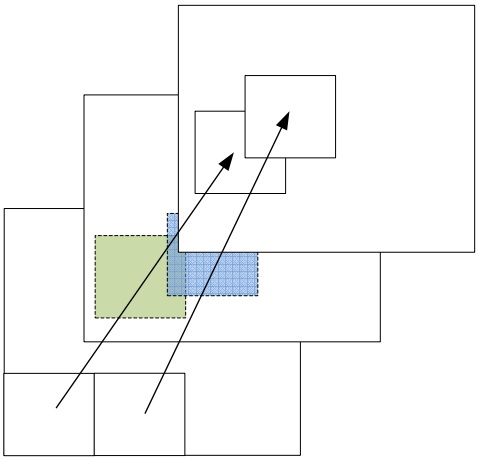
The performance of PRISM is between the H.263+ working only in INTRA mode and in INTER mode. Nevertheless, on channel loosing packet, its robustness to packet losses is much higher. A scalable extension of PRISM, proposed in [TMRT06], aims at providing SNR, spatial, and temporal scalability. At high bit-rate, this scalable extension outperforms H.263+.

7.1.2.3 Improving the side information

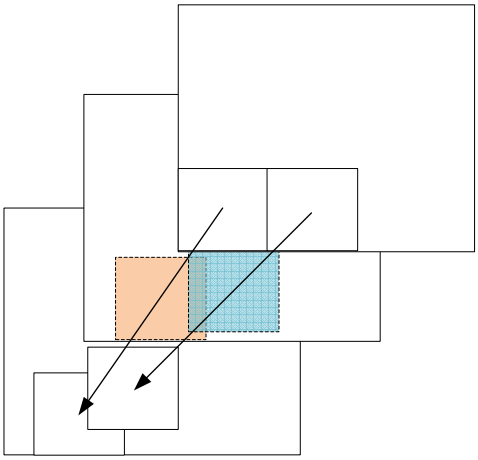
One of the main problem of Wyner-Ziv video coding is the generation of the best side information at decoder side. The transmission of hash sequences, as in [ARG04] or CRCs, as in [PR03], helps the decoder to select the best among several side information candidates, obtained, e.g., with various motion vectors. Nevertheless, this requires the transmission of additional bits. Better models, including 3D models of the scene to compress may help to improve the efficiency of the side information construction [MGM07].

The main existing solutions for the construction of the SI are recalled here. In the following, the frames are denoted by X_j (where j is the temporal index). Let us assume that the frame to estimate is X_{2i+1}. The interpolation methods will use two KFs: the backward frame X_{2i} and the forward frame X_{2i+2}.

The basic approach of the reconstruction of SI from two neighbor KFs is the **averaging** method. It assumes that there is no motion between pixels of neighboring frames. Under this assumption, a simple solution of frame interpolation for DVC scheme (studied in [ABP05]) only consists in averaging the two key frames: $X_{2i+1} = \frac{1}{2}(X_{2i} + X_{2i+2})$.



(a) Forward interpolation.



(b) Backward interpolation.

Figure 7.5: Classical interpolation tools.

Another approach is based on **Forward and Backward motion estimation**. It assumes that the motion between the frames X_{2i} and X_{2i+1} is equal to the motion between the frames X_{2i+1} and X_{2i+2}. Then the motion estimation between X_{2i} and X_{2i+2} can be used to interpolate the frame X_{2i+1}. A block matching algorithm can be used to find the best block match of target block $b_{k,l}$ centered at the coordinates (k, l) of KF X_{2i} in the next KF, X_{2i+2}. The parameters that characterize the estimation technique are the block size, the matching criterion, the search range and the precision. Given that the best matching of block $b_{k,l}$ of X_{2i} in X_{2i+2} is $f_{m,n}$ with a block motion of $\vec{w_f} = (m-k, n-l)$, the linear projection of these two blocks

onto the frame X_{2i+1} can be calculated as $c = \frac{b+f}{2}$ where c is centered at the location $(\frac{m+k}{2}, \frac{l+n}{2})$. Figure 7.5(a) describes the forward motion estimation between X_{2i} and X_{2i+2} and their linear projection on X_{2i+1}. When the forward motion vectors are projected on the frame X_{2i+1}, overlapping and uncovered areas will usually appear. The overlapping areas correspond to the multiple motion vectors which pass through a unique pixel, whereas uncovered areas correspond to the absence of the motion trajectory for these pixels. A similar calculation can be done for the backward motion estimation (see Figure 7.5(b)), where the aim is to find the block $b_{m',n'}$ in X_{2i} which is the best estimation of block $f_{k',l'}$ in X_{2i+2}. Given a backward motion of $\vec{w_b} = (m' - k', l' - n')$, the candidate block c of X_{2i+1} can be calculated similarly as in the forward case $c = \frac{b+f}{2}$, where in this case c is centered at the location $(\frac{m'+k'}{2}, \frac{l'+n'}{2})$.

In [ABP05], the authors use a **Motion compensation using rigid motion vectors**. Besides forward and bidirectional motion estimation, they use a spatial motion smoothing algorithm to eliminate motion outliers. After finding the forward motion vectors for non-overlapping blocks in the frame X_{2i+1}, the proposed scheme uses weighted vector median filters, which maintain the spatial coherence of the motion field by looking for candidate motion vectors in neighboring blocks. An extension of this method can be found in [ABP06]. This approach is used in the well-known Discover coder [dis07].

In [CMPP00], the authors present a differential motion estimation method for the construction of the SI, based on a pel-recursive motion estimation algorithm, which can improve the couple of motion vectors fields used to produce the interpolation. This work shows the interest of using a dense motion vector field in the framework of DVC.

7.1.3 Multiple descriptions for robust Wyner-Ziv coding

This part focuses on the use of the MDC principle to improve the performances of distributed video coding schemes, especially by considering multiple descriptions in the context of source coding with side information. Indeed, most of the DSC systems deal with the compression efficiency problem without taking into account the robustness of the system to the transmission noise. A DSC system, optimal for the compression, is however very sensible to the performances of the different source coders. Indeed, if one of the source coders doesn't work, the total performances of the compression system are very deteriorated. On the other hand, the MDC systems deal with this problem of robustness and allow to obtain good rate-distortion performances, even in a presence of a transmission noise.

In [RAG05], Rane, Aaron and Girod propose a systematic lossy error protection of video waveforms using multiple embedded Wyner-Ziv video descriptions. In [WVC04], the authors use the correlation between the de-

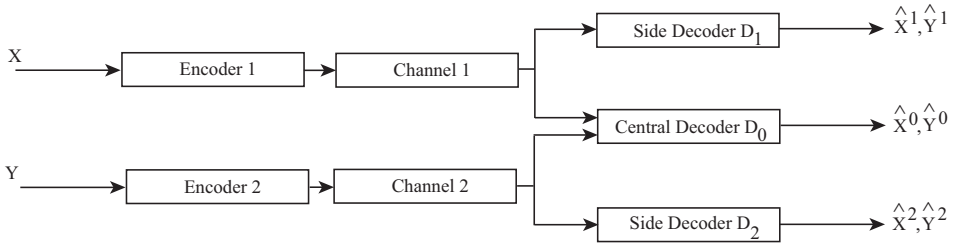


Figure 7.6: Block diagram of a robust distributed source coding scheme based on multiple description.

scriptions in each subband and potentially error corrupted description as side information at the decoder to formulate the MD decoding as a Wyner-Ziv decoding problem. That makes the system more robust to channel error corruption. In [SJA04], the authors propose a video encoding algorithm that prevents the indefinite propagation of errors in predictively encoded video. This is accomplished by periodically transmitting a small amount of additional information, called coset information, to the decoder, as opposed to the popular approach of periodic insertion of intra-coded frames. Chen, in [Che05], proposes an application of MDC in distributed image coding for a novel construction of Wyner-Ziv codec. Correlated multiple descriptions of the image can be used as correlated sources and coded parity bits can be transmitted as side information for Wyner-Ziv decoder.

Links are being also highlighted between DSC and MDC in the paper of Chen and Berger [CB08]. Results obtained in the framework of over-complete signal representations can be interesting for DSC when the data are send on noisy channels (works of Labeau *et al.* [LCK$^+$05]). In the works of Chen and Berger [CB08] and Saxena *et al.* [SNR06], the problem of encoding two correlated sources X and Y using a multiple description distributed coding scheme is considered. The quantizers for both encoders are optimized in order to get a good compromise between distortion when information from both sources are available and when one encoded source is missing, see Figure 7.6. Consider that $(\widehat{X}^0, \widehat{Y}^0)$ denotes the reconstructed values for (X, Y) produced by the central decoder D_0 and that $(\widehat{X}^1, \widehat{Y}^1)$ and $(\widehat{X}^2, \widehat{Y}^2)$ correspond to the reconstruction of (X, Y) when only the channel 1 (respectively 2) is working. In [SNR06], the aim is to minimize, for a given rate allocation R_1 and R_2 the quantity:

$$E\Big[\big(\alpha_0 d(X,\widehat{X}^0) + (1-\alpha_0) d(Y,\widehat{Y}^0)\big) + \lambda_1\big(\alpha_1 d(X,\widehat{X}^1) + (1-\alpha_1) d(Y,\widehat{Y}^1)\big) + \lambda_2\big(\alpha_2 d(X,\widehat{X}^2) + (1-\alpha_2) d(Y,\widehat{Y}^2)\big)\Big], \tag{7.1}$$

where $\alpha_n \in [0,1]$, $n = 0,1,2$, governs the relative importance of the sources X and Y at the n-th decoder (Figure 7.6). The optimization of Equation (7.1) may be performed using techniques which are very similar to those used in standard multiple description coding schemes. Pradhan *et al.* [PCR03] explore the duality between source coding with side information at the decoder and channel coding with side information at the encoder.

A recent work, presented by Crave *et al.* in [CGPPT08], proposes an extension to a classical MDC scheme, with systematic lossy description coding, where the original sequence is separated into two subsequences, one being classicaly coded, and the other being coded with a Wyner-Ziv (WZ) encoder. This leads to having a systematic lossy Wyner-Ziv coding of every other frame of each description. This error control approach can be used as an alternative to automatic repeat request (ARQ) or forward error correction (FEC).

7.2 Efficient construction of the side information for Wyner-Ziv Video Coding

DVC efficiency strongly depends on the quality of the side information construction at the decoder. The SI construction consists of computing a frame estimation, for example with an interpolation between two existing frames. This SI is corrected at the decoder by the parity bits sent by the WZ encoder. Coding efficiency strongly depends on the quality of the interpolation method. Indeed, the better the SI, the lower is the bit-rate required for the correction. In the framework of the national ESSOR project, a novel interpolation method which performs bidirectional motion estimation and uses pixelwise motion compensation by allowing overlapped motion vectors has been performed.

The proposed solution, which improves the side information quality and the coding performances, is introduced in the following.

7.2.1 Proposed interpolation method

After briefly presenting the principles of motion estimation, the focus is done on the proposed method: the bidirectional interpolation.

7.2.1.1 Forward and Backward motion estimation

A block matching algorithm can be used to find the best block match of target block b of KF X_{2i} in the next KF, $X_{2(i+1)}$. The parameters that characterize the estimation method are the block size, the matching criterion, the search range and the precision. Given that the best matching of block b of X_{2i} in $X_{2(i+1)}$ is f with a motion vector $\vec{w_f}$, the linear projection of these two blocks onto the frame X_{2i+1} can be calculated as $c = \frac{b+f}{2}$ where

c is centered at the location $b + 1/2\vec{w_f}$. A sketch of forward motion estimation between X_{2i} and $X_{2(i+1)}$ and their linear projection on X_{2i+1} can be found in Figure 7.5(a). When the forward motion vectors are projected on the frame X_{2i+1} under the assumption of linear velocity of the motion vectors, overlapping and uncovered areas will appear. The overlapping areas correspond to the multiple motion vectors which pass through a unique pixel, whereas uncovered areas correspond to the lack of motion trajectory for these pixels. A similar calculation can be done for the backward motion estimation (see Figure 7.5(b)), where the aim is to find the block b in X_{2i} which is the best estimation of block f in $X_{2(i+1)}$. Given a motion vector $\vec{w_b}$, the candidate block c of X_{2i+1} can be calculated similarly as in the forward case $c = \frac{b+f}{2}$, where in this case c is centered at $f + 1/2\vec{w_b}$.

7.2.1.2 Bidirectional interpolation

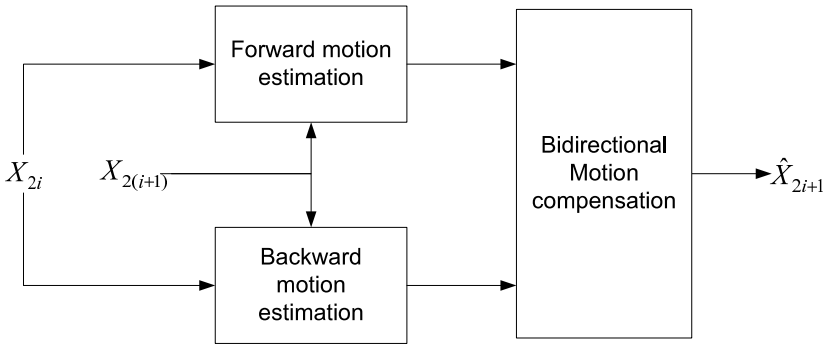
Forward and backward motion vectors ($\vec{w_f}$,$\vec{w_b}$) are calculated between two key frames as explained in the previous section. The assumption that it exists a linear motion between the key frames and the interpolated frames is done. Hence $1/2\vec{w_f}$ and $1/2\vec{w_b}$ are used for the motion compensation step. After the calculation of the forward and the backward motion compensation, the proposed bidirectional frame interpolation step is applied as follows:

Let $p_i(x, y)$ be the pixel value of the i'th frame in the coordinates of x and y. $\mathcal{C}(p_{2i+1}(x, y))$ is defined as the set $\mathcal{C}$ of motion compensated blocks that passes through the pixel $p_{2i+1}(x, y)$. Then the interpolated pixel value yields:

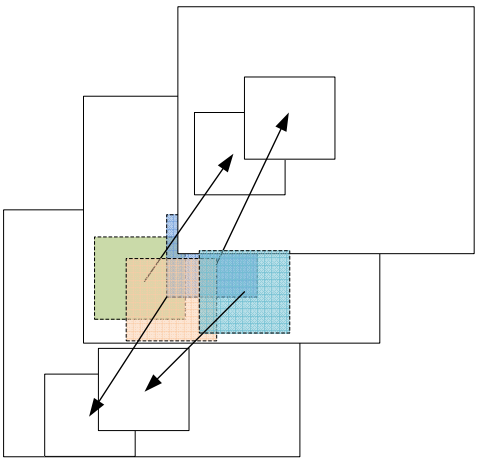
$$\hat{p}_{2i+1}(x,y) = \begin{cases} \frac{1}{|\mathcal{C}|}\sum_{i=1}^{|\mathcal{C}|} c_i, & \text{if} |\mathcal{C}| > 0, \\ 0.5 \times (p_{2i}(x,y) + p_{2i+2}(x,y)), & \text{else} \end{cases} \tag{7.2}$$

where $|\mathcal{C}|$ is the number of members in set $\mathcal{C}$. Hence, if the set $\mathcal{C}$ is not an empty set, which corresponds to at least one motion vector passes through the pixel value $p_{2i+1}(x, y)$, then an averaging of the corresponding pixel values in the motion compensated blocks of the set $\mathcal{C}$ is done. Otherwise, a simple averaging of the pixel values in previous and next KFs is performed, without any motion compensation. The block diagram of the ESSOR interpolation method and the visualization of the bidirectional estimation can be found in Fig. 7.7.

Therefore, the major difference of the proposed method from existing methods in [GARRM05, ABP06] is that the motion compensation is not based on non-overlapping block matching of the SI frame, but a pixel-wise interpolation. Contrary to the non-overlapped block matching approach in [ABP05], the proposed interpolation allows overlapped block matching and a pixel-by-pixel estimation using real bidirectional motion vectors between consecutive KFs is done in the final step. While small values of overlapped step size result in a more smoothing operation of a pixel value using a set



(a) Block diagram.



(b) Forward-Backward interpolation.

Figure 7.7: Proposed interpolation method.

of motion compensation, big values of overlapped size results in pixels with no motion vectors which corresponds simple averaging at the limit. In the experiments, a ratio of 1/2 is fixed between the overlapped step size and the block size which gives satisfactory results.

7.2.2 Experimental results

In order to evaluate the proposed interpolation method, QCIF resolution sequences with 15 fps, such as FOREMAN, NEWS and HALL are used, for the first 75 frames. Even frames are selected as KFs and their quantized version is available at the decoder, and the odd frames are interpolated from the KFs. The results of the proposed method are compared with the ones of average frame interpolation (Avg) and of the best interpolation methods proposed in [ABP05, ABP06] used in the Discover coder, available online at

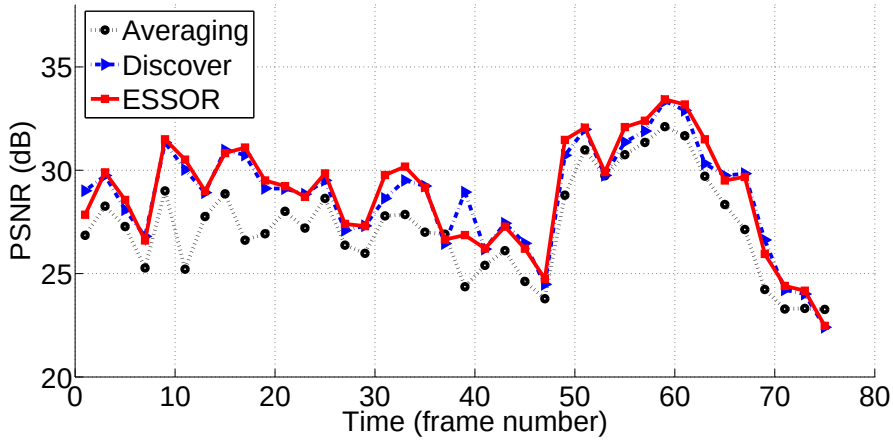


Figure 7.8: PSNR quality of each interpolated SI frame of FOREMAN sequence, where KFs are quantized, for the three interpolation methods.

[dis07]. In all the experiments, fixed block size of 8×8 pixels, a search range of ± 16, a step size of 4 pixels for the overlapped blocks, and an integer pixel precision for the forward and the backward motion estimation are used. The step size determines the shift of the blocks for calculating the next motion vector, hence MVs are calculated for the overlapped blocks in every 4 pixels in height and width. For the generation of the interpolation, three different KF types are used: lossless coding of KFs, JPEG-2000 coding of KFs with different visual qualities, and H.264 intra-coding of KFs with different visual qualities. An interpolation error analysis is first presented.

7.2.2.1 Interpolation error analysis

In this section, the side information is generated using non-degraded reference frames. The proposed method (ESSOR) is compared with the Discover approach and the average of the two reference frames (see Section 7.1.2.3 for these approaches). Here, the behavior of the SI error for the different methods is analyzed.

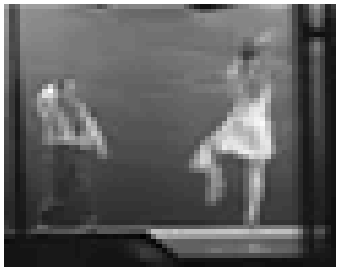
Figure 7.8 represents the evolution of the PSNR of the side information along the time for QCIF FOREMAN test sequence. These plots show that when the motion activity is not important, ESSOR method outperforms the others. This can be explained by the fact that this technique presents a smoothing property. In case of high motion activity, Discover builds an SI of higher quality than ESSOR.

In Figure 7.9, zooms on the different side informations for the third frame of NEWS test sequence are represented. Error images are also shown.

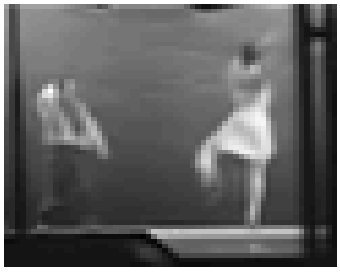
Looking at these figures, one can clearly see the smoothed aspect of ESSOR estimation, while the SI of Discover presents some blocking artefacts.

(a) Original frame.

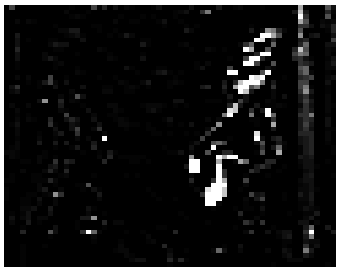
(b) Zoom on original frame.



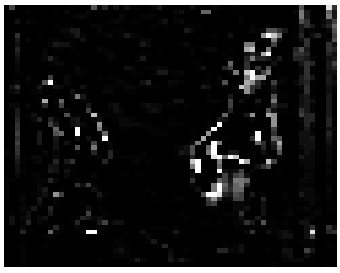
(c) Zoom on Discover interpolation.



(d) Zoom on ESSOR interpolation performance.



(e) Zoom on Discover interpolation error.



(f) Zoom on ESSOR interpolation error.

Figure 7.9: Interpolation performance of the News sequence, frame 3, zooming on the center of the frame.

7.2.2.2 Lossless Key Frames

In this section again, the side information is generated using non-degraded reference frames. The proposed method (ESSOR) is compared with the Discover approach, the basic interpolation method, and the averaging method. Experimental results are presented in Table 7.1. One can see that the ESSOR approach outperforms the Discover solution up to 1.04 dB.

Sequence	Avg	[[ABP05]]	[[ABP06]]	ESSOR
NEWS	39.76	39.80	39.83	**40.27**
FOREMAN	27.86	29.42	29.79	**29.90**
HALL	37.84	38.57	38.69	**39.73**

Table 7.1: Performance of frame interpolation methods in PSNR for lossless Key Frames.

Average KF Distortion	30.9 dB	39 dB	43.4 dB
Averaging	30.53	35.78	37.17
Discover	30.72	36.43	37.94
ESSOR	**30.93**	**37.13**	**38.88**

Table 7.2: Performance of HALL sequence when KFs are coded as JPEG-2000 frames with mean PSNR values 30.9 dB, 39 dB, and 43.4 dB.

Average KF Distortion	29.5 dB	37 dB	41.5 dB
Avg	29.48	35.71	38.01
Discover	29.49	35.74	38.04
ESSOR	**29.59**	**35.99**	**38.40**

Table 7.3: Performance of NEWS sequence when KFs are coded as JPEG-2000 frames with mean PSNR values 29.5 dB, 37 dB, and 41.5 dB.

7.2.2.3 Lossy Key Frames coded with JPEG-2000

While the Discover approach consists in using discrete cosine transform (DCT) based method, in the ESSOR project, the adopted DVC scheme is based on the discrete wavelet transform (DWT). Indeed, the intra coder is chosen to transmit the KFs is JPEG-2000 [jpe00]. This section provides the results obtained by this setup, and a comparison is given with the existing methods. Three different levels of quantization are produced, for HALL and NEWS, at Tables 7.2 and 7.3. One can see that the results of the proposed approach surpass the ones of the two other tested approaches.

7.2.2.4 Lossy Key Frames coded with H.264 Intra

In practical video coding contexts, the KFs are compressed, and the available KFs are not lossless anymore. In many coding schemes in the literature [AAD+07], the coder used to encode the KFs is H.264 intra [MWS06]. In this section, the proposed interpolation is compared to the Discover one, in case of H.264 intra transmission of the KFs. Three different quantization

Average KF Distortion	30.9 dB	34.3 dB	40 dB
Avg	29.9	33.31	36.53
Discover	30.05	33.73	37.30
ESSOR	**30.27**	**34.10**	**38.02**

Table 7.4: Performance of HALL sequence when KFs are coded as H-264 frames with mean PSNR values 30.9 dB, 34.3 dB, and 40 dB.

Average KF Distortion	29.3 dB	34.34 dB	40.7 dB
Averaging	29.614	33.47	37.64
Discover	29.616	33.49	37.72
ESSOR	**29.704**	**33.64**	**37.96**

Table 7.5: Performance of NEWS sequence when KFs are coded as H-264 intra frames with mean PSNR values 29.3 dB, 34.34 dB, and 40.7 dB.

levels corresponding to low, medium, and high bit-rates are used. The experimental results are presented for HALL and NEWS sequences in 7.4 and 7.5. The KFs average PSNR values are given in the first row of the table. For each quantization levels, the average PSNR values obtained with the ESSOR approach are compared with the ones obtained by Discover and by the average method. The results show an improvement of the performance in average PSNR of 0.5 dB compared to the Discover approach. Please note that, for low PSNR values of the KF coding, the interpolation methods can slightly surpass the average PSNR value of the KFs because the motion activity is really low.

7.3 Conclusion

DVC appears to be a very promising tool in what concerns video transmission. In the framework of the ESSOR project, an improvement of the state-of-the-art methods of frame interpolation for DVC has been performed. This work has been presented in [CDC09, MCD+08]. The SI construction is performed thanks to a frame estimation, based on an interpolation between two existing frames. The proposed method is based on a bidirectional motion compensation using pixelwise estimation and allowing overlapped motion vectors, and allows to obtain side information of higher quality than the existing approaches.

Chapter 8

General conclusion

This thesis have dealt in the first place with video coding, based on motion-compensated wavelet transform, but also by using the well-known standard H.264. Then, transmissions of video over noisy channels have been studied, especially coding techniques as multiple description coding and distributed video coding.

8.1 Video coding

8.1.1 Contributions

In the general framework of video coding, motion-compensated wavelet-based video coding has first been explored. Some improvements of a wavelet-based video coder have been proposed. This coder is fully scalable, and based on a lifted motion-compensated wavelet transform. The bit-rate allocation between the wavelet subbands uses an optimal algorithm which requires the knowledge of the rate-distortion curves of each subband. The JPEG2000/EBCOT coder is used for the coding of the spatio-temporal subbands, in order to be JPEG2000-compliant.

More precisely, the main improvement concerns the motion vectors encoding. The cost of the motion information could become too much significant in the total bit-rate, especially at low and very-low bit-rates. In this thesis, a method to optimally allocate resources between motion information and wavelet subbands in the rate-distortion sense has been proposed. To this way, motion vectors of high precision have been quantized with losses in a scalable way. A uniform scalar quantization, performed in open-loop, is used. Indeed, in order to preserve good properties for the high frequency subbands, full precision motion vectors must be used at the encoder side for the motion compensation and the computation of the wavelet transform. The quantization step controls the rate-distortion trade-off of the motion vectors.

In order to evaluate the impact of this lossy coding of the motion information, a theoretical distortion model of the motion coding error has been established. This model is simply function of the frame powers and

of the frame autocorrelation functions. The subbands quantization noise has been included to this model, which has been generalized at several levels of temporal decomposition. This distortion model allows to perform a model-based bit-rate allocation between the motion vectors and the wavelet subbands, in order to optimally dispatch the resources in the total bit-rate. This approach allows to improve the whole coder performances, especially at low bit-rates.

Some improvements of the lifting scheme have also been done. The influence of some badly estimated motion vectors on the motion-compensated wavelet transform has been minimized, by proposing a novel and adaptive method for the implementation of the lifting scheme. Indeed, the lifting steps have been closely adapted to the motion vectors norm. This method allows to increase the coder performances, and, moreover, no loss in bit-rate is carried out, since all the useful data, *i.e.* the motion vectors, is already transmitted to the decoder side.

In the framework of an industrial contract with the French national Telecom operator, Orange labs, the approach of lossy coding of motion vectors has been applied to the actual video coding standard H.264. A new H.264 coding mode has thus been defined. Indeed, a more flexible motion coding could improve the general performances. Some problems regarding the choice of the quantization step and the encoding of quantized MVs have been solved, and also about the high precision and the prediction of the motion vectors, especially in the 8x8 case. The new coding mode allows to improve the performances compared to the classical modes of H.264.

8.1.2 Perspectives

Of course, this work opens a lot of perspectives. First, in terms of general performances, a more efficient motion estimator could widely improve the wavelet-based video coder, for example with a motion described by blocks of variable-length. However, the JPEG2000 compatibility would be more difficult to reach.

For further researches on the motion-adapted weighted lifting scheme, it could be possible to extend the proposed approach to other criteria, as, for example, the correlation between the motion vectors.

Some improvements can also be performed for the new coding mode integrated in the H.264 standard. Further improvements are expected when the proposed technique will be extended to cover the cases where motion information is even more important, as it happens when sub-blocks of 4x4 pixels are enabled. It could also be interesting to perform the motion estimation on a real grid, that is to say a grid not anymore fixed at 1/4-pel or 1/8-pel precisions. In the same time, a gain in performances could be achieved if any arbitrary real value can be chosen as MV quantization step. As already said, the Lloyd-Max algorithm to represent the quantization step values could be

used. A relationship could maybe be found between the optimal quantization step for the motion vectors and the current quantization step for the coefficients. Different strategies for finding the best motion quantization step could be tested, more efficient than "Oracle" or "Minsum", as the use of quad-trees to divide the MB. Moreover, the open loop structure could be used to implement an efficient MV-scalable video coder. Finally, the last version of H.264, KTA, has to be used to obtain better performances.

8.2 Transmissions over noisy channels

8.2.1 Contributions

Video transmissions over noisy channels have been studied in the framework of the national research project ESSOR. In particular, in the general framework of multiple description coding, two approaches of optimal decoding have been proposed. The MD coder considered here is based on the motion-compensated wavelet transform coder presented in the first part of this work. Redundancy is introduced before quantization, and at the encoder side, a bit allocation based on the characteristics of the channel produces the balanced descriptions. The descriptions are then transmitted over noisy channels. In order to optimally decode the central description, two approaches have been implemented: the first one tries to estimate the two generated descriptions from the received channel outputs, whereas the second one focuses on the direct estimation of the source from the two noisy descriptions. These approaches are based on the knowledge of the density of the source and on the noise probabilities. Even with an important channel noise, the quality of the central descriptions reconstructed by both of the approaches are close to the one of the original signal.

In the framework of distributed video coding, in collaboration with other French laboratories, an improvement of the actual methods of frame interpolation has been performed, in order to increase the quality of the side information, and thus of the coding performances. The proposed method is based on a bidirectional motion compensation using pixelwise estimation and allowing overlapped motion vectors. It allows to better improve the quality of the side information compared to some state-of-the-art methods.

8.2.2 Perspectives

With the increase of multimedia communications, transmissions over noisy channels will be of great interest in the next years. Thus, to increase the compression efficiency of the proposed MDC scheme, a product code can be used to code the descriptions in a fixed-length way. Then, after indexation, an entropy-coding step may be considered. Dependence between the coefficients would be to consider for the MAP algorithms. Iterative decoding

techniques, such as those presented in [LWK06], may then be applied. In order to reduce the errors on the position of the vectors of the product code, transcoding of these vectors could be used. Finally, the noise corrupting the motion vectors has also to be taken into account. Of course, it could be interesting to introduce in the MD coder the method of lossy coding of the motion vectors previously proposed.

For what concerns the proposed frame interpolation in the framework of DVC, it could be interesting to implement shot detection methods in order to improve the performance of the interpolation method.

For further works, it would be interesting to integrate the proposed MDC scheme in a general DVC scheme, in order to make it robust to channel failures.

Conclusion générale

Cette thèse a porté en premier lieu sur le codage vidéo, basé sur la transformée en ondelettes compensée en mouvement, mais aussi en utilisant la norme bien connue H.264. Ensuite, la transmission de vidéos sur canaux bruités a été étudiée, en particulier les techniques de codage par descriptions multiples et le codage vidéo distribué.

Codage vidéo

Contributions

Dans le contexte général du codage vidéo, le codage par transformée en ondelettes compensée en mouvement a d'abord été exploré. Quelques améliorations d'un codeur vidéo basé ondelettes ont été proposées. Ce codeur est entièrement scalable, et est basé sur une transformée en ondelettes liftée et compensée en mouvement. L'allocation de débit entre les sous-bandes d'ondelettes utilise un algorithme optimal qui nécessite la connaissance des courbes débit/distorsion de chaque sous-bande. Le codeur JPEG2000/EBCOT est ensuite utilisé pour le codage des sous-bandes spatio-temporelles, afin d'être compatible JPEG2000.

Plus précisément, la principale amélioration concerne le codage des vecteurs mouvement. Le coût d'information de mouvement peut devenir trop important dans le débit total, en particulier à bas et à très bas débit. Dans cette thèse, une méthode pour répartir de manière optimale les ressources entre l'information de mouvement et les sous-bandes d'ondelettes dans le sens débit/distorsion a été proposée. Pour cela, des vecteurs mouvement de haute précision ont été quantifiés avec pertes d'une manière scalable. Une quantification scalaire uniforme, effectuée en boucle ouverte, est utilisée. En effet, dans le but de préserver des bonnes propriétés pour la sous-bande haute fréquence, des vecteurs mouvement de haute précision doivent être utilisés à l'encodeur pour la compensation de mouvement et le calcul de la transformée en ondelettes. Les pas de quantification contrôle le compromis débit/distorsion des vecteurs mouvement.

Afin d'évaluer l'impact de ce codage avec pertes de l'information de mouvement, un modèle théorique de distorsion de l'erreur de codage du mouve-

ment a été mis en place. Ce modèle est simplement fonction de la puissance des images et des fonctions d'autocorrélation des images. Le bruit de quantification des sous-bandes a été inclus à ce modèle, qui a été généralisé à plusieurs niveaux de décomposition temporelle. Ce modèle de distorsion permet de réaliser une allocation de débit optimale entre les vecteurs mouvement et les sous-bandes d'ondelettes, afin d'optimiser la répartition des ressources dans le débit total. Cette approche permet d'améliorer l'ensemble des performances du codeur, surtout à bas débit.

Des améliorations du schéma lifting ont également été réalisées. L'influence de certains vecteurs mouvement mal estimés sur la transformée en ondelettes compensée en mouvement a été réduite, par une nouvelle méthode adaptée de mise en oeuvre du schéma lifting. En effet, les pas du schéma lifting ont été étroitement adaptés à la norme des vecteurs mouvement. Cette méthode permet d'augmenter les performances du codeur, et, de surcroît, aucune perte en débit n'est effectuée, puisque toutes les données utiles, c'est-à-dire les vecteurs mouvement, sont d'ores et déjà transmises au décodeur.

Dans le cadre d'un contrat industriel avec Orange Labs, la méthode précédente de codage avec pertes des vecteurs mouvement a été appliquée à la norme de codage vidéo H.264. Un nouveau mode de codage H.264 a ainsi été défini. En effet, une plus grande souplesse dans le codage du mouvement pourrait améliorer les performances. Certains problèmes en ce qui concerne le choix du pas de quantification et le codage des vecteurs mouvement quantifiés ont été résolus, et aussi à propos de la haute précision et de la prédiction des vecteurs mouvement, en particulier dans le cas 8x8. Le nouveau mode de codage permet d'améliorer les performances par rapport aux modes classiques d'H.264.

Perspectives

Bien sûr, ce travail ouvre de nombreuses perspectives. Tout d'abord, en termes de performances générales, un estimateur de mouvement plus efficace pourrait largement améliorer le codeur vidéo basé ondelettes, par exemple, avec un mouvement décrit par des blocs de longueur variable. Toutefois, la compatibilité JPEG2000 serait plus difficile à atteindre.

Pour de plus grandes recherches sur le schéma lifting pondéré adapté au mouvement, il pourrait être possible d'étendre l'approche proposée à d'autres critères comme, par exemple, la corrélation entre les vecteurs mouvement.

Certaines améliorations peuvent également être réalisées pour le nouveau mode de codage intégré dans le standard H.264. D'autres améliorations sont prévues lorsque la technique proposée sera étendue à des cas où l'information de mouvement est encore plus importante, comme c'est le cas lorsque des sous-blocs de 4x4 pixels sont activés. Il pourrait également être intéressant de réaliser l'estimation de mouvement sur un réseau à grille réelle, c'est-à-

dire une grille non plus fixée aux précisions au quart ou au huitième de pixel. En parallèle, un gain de performances pourrait être atteint si n'importe quelle valeur réelle pouvait être choisie comme pas de quantification des vecteurs mouvement. Comme dit précédemment, l'algorithme Lloyd-Max pourrait être utilisé pour représenter les valeurs de ces pas de quantification. Une relation pourrait être trouvée entre les pas de quantification optimaux des vecteurs mouvement et ceux des coefficients. Différentes stratégies pour trouver le meilleur pas de quantification des vecteurs mouvement pourrait être testées, plus efficace que "Oracle" ou "Minsum", comme l'utilisation de quad-trees pour diviser les macro-blocs. En outre, la structure en boucle ouverte peut être utilisée pour mettre en place un efficace codeur vidéo scalable. Enfin, la dernière version de H.264, KTA, doit être utilisée pour obtenir de meilleures performances.

Transmissions sur des canaux bruités

Contributions

Les transmissions vidéo sur canaux bruités ont été étudiées dans le cadre du projet national de recherche ESSOR. En particulier, dans le contexte général du codage par descriptions multiples, deux approches optimales de décodage ont été proposées. Le codeur par descriptions multiples considéré ici est basé sur le codeur basé ondelettes compensé en mouvement présenté dans la première partie de ce travail. La redondance est introduite avant quantification, et à l'encodeur, une allocation de débit fondée sur les caractéristiques du canal produit les deux descriptions équilibrées. Ces descriptions sont ensuite transmises sur des canaux bruités. Dans le but de décoder de manière optimale la description centrale, deux approches ont été mises en oeuvre : la première essaie d'estimer les deux descriptions générées à partir des données reçues après transmission canal, tandis que la seconde se concentre sur une estimation directe de la source à partir des deux descriptions bruitées. Ces approches sont fondées sur la connaissance de la densité de la source et sur les probabilités de bruit. Même avec un bruit important, la qualité des descriptions centrales reconstruites par les deux approches est très proche de celle du signal original.

Dans le cadre du codage vidéo distribué, en collaboration avec d'autres laboratoires français, une amélioration des méthodes actuelles d'interpolation d'image a été réalisée, afin d'accroître la qualité de l'information adjacente, et donc des performances de codage. La méthode proposée est basée sur une compensation de mouvement bi-directionnelle et utilise une estimation pixélique du mouvement. Elle autorise les vecteurs mouvement superposés. Elle permet également de mieux améliorer la qualité de l'information adjacente par rapport aux méthodes de l'état de l'art.

Perspectives

Avec l'accroissement des communications multimédia, les transmissions sur canaux bruités seront d'un grand intérêt dans les prochaines années. Ainsi, pour accroître l'efficacité de la compression du codeur par descriptions multiples proposé, un code produit pourrait être utilisé pour coder les descriptions avec un code à longueur fixe. Puis, après indexation, une étape de codage entropique pourrait être considérée. Des techniques de décodage itératif, telles celles présentées dans [LWK06], pourraient être appliquées. Afin de réduire les erreurs sur la position des vecteurs issus du code produit, le transcodage de ces vecteurs pourrait être réalisé. Enfin, le bruit qui corrompt les vecteurs mouvement est aussi à prendre en compte. Naturellement, il pourrait être intéressant d'introduire dans le codeur par descriptions multiples la méthode de codage avec pertes des vecteurs mouvement également proposée.

En ce qui concerne l'interpolation d'image dans le contexte du codage vidéo distribué, il pourrait être intéressant de mettre en oeuvre des méthodes de shot detection en vue d'améliorer les performances de la méthode d'interpolation.

Pour de plus amples travaux, il serait intéressant d'intégrer le codeur par descriptions multiples proposé dans un schéma général de codage vidéo distribué, afin de le rendre robuste aux échecs du canal.

Appendix A

Distortion model on two decomposition levels

Taking into account the structure of the lifting scheme (see figure 3.9 in Section 3.3.1.2), the distortion model given by equation (3.3) of Section 3.3.2.2 can be expanded on two temporal decomposition levels as:

$$D_t = \frac{1}{K}\Big(\sum_{k=0}^{\frac{K}{4}-1} \big(\mathbf{Pn}\left(x_{4k} - \widetilde{x}_{4k}\right) + \mathbf{Pn}\left(x_{4k+2} - \widetilde{x}_{4k+2}\right)\big) + \sum_{k=0}^{\frac{K}{2}-1} \mathbf{Pn}\left(x_{2k+1} - \widetilde{x}_{2k+1}\right)\Big). \tag{A.1}$$

Due to the properties of the (2,0) lifting scheme on two decomposition levels (see figure 3.9), the first term of this equation (called D_{4k}) is simply equal to the low frequency subbands coding error on the images x_{4k}:

$$D_{4k} = \sum_{k=0}^{\frac{K}{4}-1} \mathbf{Pn}(\epsilon_{4k}). \tag{A.2}$$

The second term of (A.1) (called in the following D_{4k+2}) has thus to be computed, thanks to the lifting equations of analysis and synthesis for the second decomposition level:

$$\widehat{h}_{4k+2}\left(\mathbf{p}\right) = x_{4k+2}\left(\mathbf{p}\right) - \frac{1}{2}(x_{4k}^{B^{(2)}} + x_{4k+4}^{F^{(2)}}) + \epsilon_{h^{(2)}},$$

and

$$\widehat{\widetilde{x}}_{4k+2}\left(\mathbf{p}\right) = \widehat{h}_{4k+2}\left(\mathbf{p}\right) + \frac{1}{2}(\widehat{\widetilde{x}}_{4k}^{\widehat{B}^{(2)}} + \widehat{\widetilde{x}}_{4k+4}^{\widehat{F}^{(2)}}).$$

Just as previously, by combining the two previous equations, and by using the notations of Section 3.3.1.1, one can write:

$$x_{4k+2}\left(\mathbf{p}\right) - \widehat{\widetilde{x}}_{4k+2}\left(\mathbf{p}\right) = \frac{1}{2}(x_{4k}^{B^{(2)}} - \widetilde{x}_{4k}^{\widehat{B}^{(2)}} - \widetilde{\epsilon}_{4k}^{\widehat{B}^{(2)}}) + \frac{1}{2}(x_{4k+4}^{F^{(2)}} - \widetilde{x}_{4k+4}^{\widehat{F}^{(2)}} - \widetilde{\epsilon}_{4k+4}^{\widehat{F}^{(2)}}) - \epsilon_{h^{(2)}}.$$

By assuming that the crossed scalar products are equal to zero (Section 3.3.3), with the same notations and hypothesis as previously, and while proceeding in the same way as in Section 3.3.3, D_{4k+2} is expressed as:

$$D_{4k+2} = \frac{1}{2K} \sum_{k=0}^{\frac{K}{4}-1} \Big[\mathbf{Pn}\left(x_{4k}\right) - \Gamma_{x_{4k}}\left(\eta_{B^{(2)}}\right) + \mathbf{Pn}\left(x_{4k+4}\right) - \Gamma_{x_{4k+4}}\left(\eta_{F^{(2)}}\right) + \frac{1}{2}\mathbf{Pn}(\epsilon_{4k}) + \frac{1}{2}\mathbf{Pn}(\epsilon_{4k+4}) + 2\mathbf{Pn}(\epsilon_{h^{(2)}}) \Big]. \tag{A.3}$$

Finally, the last term called D_{2k+1} has to be computed. One can write, by proceeding as previously and by using the figure 3.9:

$$x_{2k+1}(\mathbf{p}) - \widehat{\widetilde{x}}_{2k+1}(\mathbf{p}) = \frac{1}{2}\left((x_{2k}^{B^{(1)}} - \widehat{\widetilde{x}}_{2k}^{\widehat{B}^{(1)}}) + (x_{2k+2}^{F^{(1)}} - \widehat{\widetilde{x}}_{2k+2}^{\widehat{F}^{(1)}}) \right) - \epsilon_{h^{(1)}}. \tag{A.4}$$

It is important to note here that, because of the properties of the (2,0) lifting scheme on two decomposition levels, $\widehat{x}_{2k+2}$ is in fact equal to $\widehat{\widehat{x}}_{4k+2}$, and, thus, $\widehat{\widetilde{x}}_{2k+2}^{\widehat{F}^{(1)}}$ is equal to $\widehat{\widetilde{x}}_{4k+2}^{\widehat{F}^{(1)}}$, for even frames. And for odd frames, $\widehat{\widetilde{x}}_{2k}^{\widehat{F}^{(1)}}$ is equal to $\widehat{\widetilde{x}}_{4k+2}^{\widehat{F}^{(1)}}$, but by taking into account the properties of symmetry of the motion vectors (see Section 3.3.2.3), only the computation for the even case will be considered.
Thus, as previously (with the analysis and synthesis equations of the (2,0) lifting scheme at the second decomposition level), one can have:

$$\widehat{\widetilde{x}}_{4k+2}\left(\mathbf{p}\right) = x_{4k+2}\left(\mathbf{p}\right) - \frac{1}{2}(x_{4k}^{B^{(2)}} - \widetilde{x}_{4k}^{\widehat{B}^{(2)}} - \widetilde{\epsilon}_{4k}^{\widehat{B}^{(2)}}) - \frac{1}{2}(x_{4k+4}^{F^{(2)}} - \widetilde{x}_{4k+4}^{\widehat{F}^{(2)}} - \widetilde{\epsilon}_{4k+4}^{\widehat{F}^{(2)}}) + \epsilon_{h^{(2)}}.$$

Thanks to the high-rate assumption (see Section 3.3.2.1):

$$x_{4k}^{B^{(2)}} \approx \widetilde{x}_{4k}^{\widehat{B}^{(2)}},$$

and

$$x_{4k+4}^{F^{(2)}} \approx \widetilde{x}_{4k+4}^{\widehat{F}^{(2)}}.$$

Consequently, by replacing in (A.4), one can obtain:

$$x_{2k+1}(\mathbf{p}) - \widehat{\widetilde{x}}_{2k+1}(\mathbf{p}) = \frac{1}{2}(x_{2k}^{B^{(1)}} - \widetilde{x}_{2k}^{\widehat{B}^{(1)}} - \widetilde{\epsilon}_{2k}^{\widehat{B}^{(1)}}) + \frac{1}{2}x_{2k+2}^{F^{(1)}} - \frac{1}{2}\widetilde{x}_{4k+2}^{\widehat{F}^{(1)}} - \frac{1}{4}(\widetilde{\epsilon}_{4k}^{\widehat{B}^{(2)}})^{\widehat{F}^{(1)}} - \frac{1}{4}(\widetilde{\epsilon}_{4k+4}^{\widehat{F}^{(2)}})^{\widehat{F}^{(1)}} - \frac{1}{2}\widetilde{\epsilon}_{h_{4k+2}}^{\widehat{F}^{(1)}} - \epsilon_{h^{(1)}}.$$

Always with the asymptotical hypothesis, it is assumed that:

$$\mathbf{Pn}\left((\widetilde{\epsilon}_{4k}^{\widehat{B}^{(2)}})^{\widehat{F}^{(1)}}\right) \approx \mathbf{Pn}\left(\widetilde{\epsilon}_{4k}^{\widehat{B}^{(2)}}\right) \approx \mathbf{Pn}(\epsilon_{4k}),$$

$$\mathbf{Pn}\left((\widetilde{\epsilon}_{4k+4}^{\widehat{F}^{(2)}})^{\widehat{F}^{(1)}}\right) \approx \mathbf{Pn}\left(\widetilde{\epsilon}_{4k+4}^{\widehat{F}^{(2)}}\right) \approx \mathbf{Pn}(\epsilon_{4k+4}),$$

and

$$\mathbf{Pn}\left(\widetilde{\epsilon}_{h_{4k+2}}^{\widehat{F}^{(1)}}\right) \approx \mathbf{Pn}(\epsilon_{h^{(2)}}).$$

Moreover, one can also write that $x_{4k+2} = x_{2k+2}$, because of the properties of the (2,0) lifting scheme on two decomposition levels (see figure 3.9), and when k is even; and, thus, $\widetilde{x}_{4k+2}^{\widehat{F}^{(1)}} = \widetilde{x}_{2k+2}^{\widehat{F}^{(1)}}$. The distortion D_{2k+1} can be expressed as (with similar notations as previously and by considering that the crossed scalar products are equal to zero):

$$D_{2k+1} = \frac{1}{2K}\sum_{k=0}^{\frac{K}{2}-1}\Big[\mathbf{Pn}\left(x_{2k}\right) - \Gamma_{x_{2k}}\left(\eta_{B^{(1)}}\right) + \mathbf{Pn}\left(x_{2k+2}\right) - \Gamma_{x_{2k+2}}\left(\eta_{F^{(1)}}\right) + \frac{1}{2}\mathbf{Pn}(\epsilon_{2k}) + \frac{1}{8}\mathbf{Pn}(\epsilon_{4k}) + \frac{1}{8}\mathbf{Pn}(\epsilon_{4k+4}) + \frac{1}{2}\mathbf{Pn}(\epsilon_{h^{(2)}}) + 2\mathbf{Pn}(\epsilon_{h^{(1)}})\Big]. \tag{A.5}$$

Finally, by adding (A.2), (A.3) and (A.5), the total distortion model on two decomposition levels can be computed by:

$$\begin{aligned} D_t \approx & \frac{1}{2K}\sum_{k=0}^{\frac{K}{2}-1}\Big[\mathbf{Pn}\left(x_{2k}\right) - \Gamma_{x_{2k}}\left(\eta_{B^{(1)}}\right) + \mathbf{Pn}\left(x_{2k+2}\right) - \Gamma_{x_{2k+2}}\left(\eta_{F^{(1)}}\right)\Big] \\ & + \frac{1}{2K}\sum_{k=0}^{\frac{K}{4}-1}\Big[\mathbf{Pn}(x_{4k}) - \Gamma_{x_{4k}}(\eta_{B^{(2)}}) + \mathbf{Pn}(x_{4k+4}) - \Gamma_{x_{4k+4}}(\eta_{F^{(2)}})\Big] \\ & + \frac{1}{K}\Big[\frac{1}{2^2}\mathbf{Pn}(\epsilon_{l^{(2)}}) + \sum_{i=1}^{2}\frac{1}{2^i}\mathbf{Pn}(\epsilon_{h^{(i)}})\Big], \end{aligned}$$

with $l^{(2)}$ the low frequency subband and $h^{(i)}$ the high frequency subband at the i^{th} decomposition level.

Appendix B

Cost function for H.264 coding modes

In this appendix, some examples of the computation of the cost function for some H.264 coding modes are given. This is a useful reference for the new coding mode. The main notations used here have been defined in Section 4.1.3.
As in Section 4.1.4, for each mode, the *distortion* between the original MB and its reconstructed version should be computed. Moreover, the coding cost of the mode should be computed. For the non-motion compensated modes, it amounts to the cost of transmitting the quantized transform coefficients representing the current MB, plus the signalling of the selected mode. For the motion-compensated modes, the motion information cost has to be added.

In the standard, no constraint is given about the mode selection; indeed, only the decoder is specified, and in particular it must be able to recognize and decode any mode, but *how* to select the mode is left to the encoder implementation.

In the following, the cost function of several modes is computed. The results are given as function of a set of parameters, namely the quantization step Q_p and the lagrangian parameters λ_{mode} and λ_{ME} (the latter will be defined in the following). Therefore, Q_p, λ_{mode} and λ_{ME} can be thought as inputs of the RD-optimization stage. However, some experimental relationships exist among these parameters, so that one can consider Q_p as system input, and then obtain the values for λ_{ME} and λ_{mode} from it [WSJ+03].

B.1 Cost function for the INTRA mode

Let us consider the INTRA mode. In H.264 there exist several spatial prediction techniques and two spatial transform options, so there is a considerable

number of INTRA modes. For each of these modes, we have:

$$\begin{aligned}\rho &= P[I] \\ \theta &= T[\rho] \\ \tilde{\theta} &= \text{round}\left(\frac{\theta}{Q_p}\right) \\ \tilde{I}(Q_p) &= P^{-1}\left[T^{-1}\left(\tilde{\theta}\right)\right],\end{aligned}$$

Now the distortion and the rate for any INTRA mode can be computed:

$$\begin{aligned}D(Q_p) &= \left\|I - \tilde{I}(Q_p)\right\|^p \\ R(Q_p) &= R(\tilde{\theta}) + R_{\text{mode}} \\ J_{\text{INTRA}}(Q_p, \lambda_{\text{mode}}) &= D(Q_p) + \lambda_{\text{mode}} R(Q_p)\end{aligned}$$

One can note that the R_{mode} term accounts for the signalling of the chosen transform and prediction scheme.

B.2 Cost function for the INTER16x16 mode

The computation of the cost function for the motion compensated modes is more complex than the one for the INTRA mode, since, in principle, the motion vector selection should be performed in an optimal way as well.
For each candidate motion vector $\mathbf{v}$ in the search set V, the motion compensated residual $\rho(\mathbf{v})$, its transform $\theta(\mathbf{v})$, the quantized transform $\tilde{\theta}$, and finally the reconstructed residual $\tilde{\rho}$ should be computed:

$$\begin{aligned}\rho(\mathbf{v}) &= I - I_{\text{REF}}(\mathbf{v}) \\ \theta(\mathbf{v}) &= T\left[\rho(\mathbf{v})\right] \\ \tilde{\theta}(Q_p, \mathbf{v}) &= \text{round}\left(\frac{\theta(\mathbf{v})}{Q_p}\right) \\ \tilde{\rho}(Q_p, \mathbf{v}) &= T^{-1}\left[Q_p \tilde{\theta}(\mathbf{v})\right]\end{aligned}$$

The reconstructed residual is used by the decoder to produce the reconstructed image $\tilde{I}$, by adding it to the motion compensated prediction. So:

$$\begin{aligned}\tilde{I}(Q_p, \mathbf{v}) &= I_{\text{REF}}(\mathbf{v}) + \tilde{\rho}(Q_p, \mathbf{v}) \\ D(Q_p, \mathbf{v}) &= \left\|I - \tilde{I}(Q_p, \mathbf{v})\right\|^p \\ &= \|I - I_{\text{REF}}(\mathbf{v}) - \tilde{\rho}(Q_p, \mathbf{v})\|^p \\ &= \|\rho(\mathbf{v}) - \tilde{\rho}(Q_p, \mathbf{v})\|^p \qquad \text{(B.1)}\end{aligned}$$

The rate contribution is given by the coding cost of the motion vector, the coding cost of the quantized transform coefficients of the residual, and the coding cost of the mode:

$$R(Q_p, \mathbf{v}) = R(\mathbf{v}) + R(\tilde{\theta}(Q_p, \mathbf{v})) + R_{\text{mode}} \tag{B.2}$$

Thus, the cost function associated to the mode INTER16x16 *and* to the vector $\mathbf{v}$ is:

$$J_{\text{INTER}}(Q_p, \lambda_{\text{mode}}, \mathbf{v}) = D(Q_p, \mathbf{v}) + \lambda_{\text{mode}} R(Q_p, \mathbf{v}), \tag{B.3}$$

where D and R are provided by (B.1) and (B.2) respectively.
In conclusion, the motion estimation should performed by searching the argument minimizing (B.3):

$$\mathbf{v}^*(Q_p, \lambda_{\text{mode}}) = \arg\min_{\mathbf{v}} J_{\text{INTER}}(Q_p, \lambda_{\text{mode}}, \mathbf{v}) \tag{B.4}$$

Finally, the cost function for the INTER mode and the vector $\mathbf{v}^*$ is:

$$J^*_{\text{INTER}}(Q_p, \lambda_{\text{mode}}) = J_{\text{INTER}}(Q_p, \lambda_{\text{mode}}, \mathbf{v}^*)$$

This way to compute $\mathbf{v}^*$ and J^*, even though optimal, is hardly, if ever, used in practice, because it is far too complex [SW98]: for each *candidate vector* in the search set, the whole coding/decoding process should be simulated in order to find the best possible vector. Since the number of candidate vector is easily very large, this approach is unfeasible, all the more because there exist suboptimal motion estimation techniques which allow to largely reduce the computational complexity without affecting too much final performances. These techniques split the cost function evaluation in two phases: first, a simplified *motion estimation* is performed, in order to find a "good"[1] motion vector. Then, the actual impact of this vector (and only of it) on final rate and distortion is computed.
The best vector is computed as solution of the lagrangian problem:

$$\mathbf{v}^*(\lambda_{\text{ME}}) = \arg\min_{\mathbf{v} \in V} D_{\text{DFD}}(\mathbf{v}) + \lambda_{\text{ME}} R(\mathbf{v}) \tag{B.5}$$

where $R(\mathbf{v})$ is the rate for encoding the motion vector $\mathbf{v}$, and the lagrangian parameter λ_{ME} has to be considered as an input. The reader can observe that now, in order to find $\mathbf{v}^*$, only (B.5) has to be evaluated for each $\mathbf{v}$ instead of the complex calculations in (B.1) and (B.2).
The cost function computation proceeds as follows.

$$\begin{aligned}
\rho(\mathbf{v}^*(\lambda_{\text{ME}})) &= I - I_{\text{REF}}(\mathbf{v}^*(\lambda_{\text{ME}})) \\
\theta(\mathbf{v}^*(\lambda_{\text{ME}})) &= T\left[\rho(\mathbf{v}^*(\lambda_{\text{ME}}))\right] \\
\tilde{\theta}(Q_p, \lambda_{\text{ME}}) &= \text{round}\left[\frac{\theta(\mathbf{v}^*(\lambda_{\text{ME}}))}{Q_p}\right] \\
\tilde{\rho}(Q_p, \lambda_{\text{ME}}) &= T^{-1}\left[Q_p \tilde{\theta}(Q_p, \lambda_{\text{ME}})\right].
\end{aligned}$$

[1] The "best" vector is the one in (B.4).

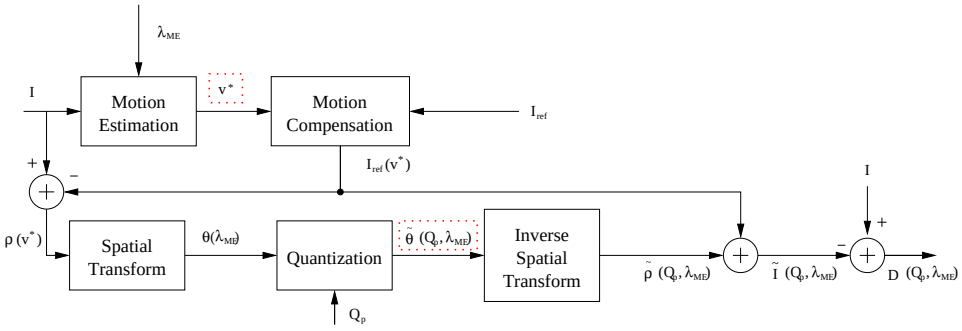


Figure B.1: Mode selection for the INTER16x16 mode

The final distortion is:

$$\begin{aligned} D(Q_p, \lambda_{\text{ME}}) &= \|I - I_{\text{REF}}(\mathbf{v}^*(\lambda_{\text{ME}})) - \tilde{\rho}(Q_p, \lambda_{\text{ME}})\|^p \\ &= \|\rho(\mathbf{v}^*(\lambda_{\text{ME}})) - \tilde{\rho}(Q_p, \lambda_{\text{ME}})\|^p, \end{aligned}$$

while the final rate is:

$$R(Q_p, \lambda_{\text{ME}}) = R(\mathbf{v}^*(\lambda_{\text{ME}})) + R(\tilde{\theta}(Q_p, \lambda_{\text{ME}})) + R_{\text{mode}}$$

In conclusion, the mode cost function is:

$$J_{\text{INTER}}(Q_p, \lambda_{\text{mode}}, \lambda_{\text{ME}}) = D(Q_p, \lambda_{\text{ME}}) + \lambda_{\text{mode}} R(Q_p, \lambda_{\text{ME}}).$$

Fig. B.1 presents a scheme summarizing this mode selection procedure. The lagrangian motion estimation produce a single vector $\mathbf{v}^*$ which is used for the computation of the reconstructed motion compensated residual $\tilde{\rho}$. Then this quantity can be used to compute the resulting distortion at the decoder. In order to not make too complex the scheme, the modules for the rate estimation is not inserted; however the input of the entropic coder (which would provide the encoding rate), *i.e.* the motion vector $\mathbf{v}^*$ and the quantized transform of the residual, $\tilde{\theta}$, is highlighted in red.

B.3 Other coding modes

H.264 provides some other coding modes. Actually, there exist several variants of the INTER16x16 mode. In these variants the MB is split in two or four sub-blocks (in this case, each of the four sub-blocks can be further split into 2 or 4 blocks). According to the sub-block size, these modes are called INTER16x8, INTER8x16, and so on. Of course, these divisions increase the coding cost, because a new motion vector is needed for each sub-block. However the distortion is hopefully reduced, so it makes sense to perform a lagrangian competition among the INTER modes.

While these modes provide a smaller distortion at the cost of a higher coding rate, the SKIP mode explores the RD curve at the opposite side. When this mode is used, only the signalling information is sent, and the MB is reconstructed by copying the MB from the reference image at a position inferred from the motion vectors of the neighbors MBs. This mode has an extremely low coding cost, but the reconstructed quality cannot be very good. However, this mode is extremely effective for low-activity areas and can dramatically improve performances at low bit-rates and/or for low motion videos.

Bibliography

[AA06] M. A. Agostini and M. Antonini. Theoretical model of the coding error in MCWT video coders. In *Proc. of IEEE International Conference on Image Processing (ICIP)*, Atlanta, USA, October 2006.

[AA07] M. A. Agostini and M. Antonini. Motion-adapted weighted lifting scheme for MCWT video coders. In *Proc. Picture Coding Symposium (PCS 2007)*, Lisboa, Portugal, November 2007.

[AA08] M. A. Agostini and M. Antonini. Multiple description video decoding using MAP. In *Proc. IEEE International Conference on Image Processing (ICIP)*, San Diego, CA, USA, October 2008.

[AAAB05a] M. A. Agostini, T. André, M. Antonini, and M. Barlaud. Codage scalable des vecteurs mouvement pour la compression de vidéos basée ondelettes. In *Conférence Francophone sur la COmpression et REprésentation des Signaux Audiovisuels (CORESA 2005)*, pages 261–266, Rennes - France, November 2005.

[AAAB05b] M. A. Agostini, T. André, M. Antonini, and M. Barlaud. Scalable motion coding for video coders with lifted MCWT. In *Proc. of International Workshop on Very Low Bit-rate Video-coding (VLBV)*, Costa Rei, Sardinia - Italy, September 2005.

[AAAB06] M. A. Agostini, T. André, M. Antonini, and M. Barlaud. Modeling the motion coding error for MCWT video coders. In *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toulouse, France, May 2006.

[AAB06] M. A. Agostini, M. Antonini, and M. Barlaud. Model-based bit allocation between wavelet subbands and motion information

in MCWT video coders. In *Proc. of EUropean SIgnal Processing COnference (EUSIPCO)*, Florence, Italy, September 2006.

[AAB07] M. A. Agostini, M. Antonini, and M. Barlaud. Allocation optimale de débit entre le mouvement et les coefficients d'ondelettes pour la compression vidéo basée ondelettes. In *Colloque GRETSI 2007 (Traitement du Signal et des Images)*, Troyes - France, September 2007.

[AAC+08] M. A. Agostini, M. Antonini, O. Crave, M. Kieffer, A. Roumy, and V. Toto-Zarasoa. Robust DSC schemes: intermediate report. *ESSOR project report 2.4*, july 2008.

[AAD+07] 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)*, Lisboa, Portugal, Nov. 2007.

[AAK09] M. A. Agostini, M. Antonini, and M. Kieffer. MAP estimation of multiple description encoded video transmitted over noisy channels. In *Submitted to IEEE International Conference on Image Processing (ICIP)*, Cairo, Egypt, November 2009.

[ABE+96] A. Albanese, J. Blomer, J. Edmonds, M. Luby, and M. Sudan. Priority encoding transmission. *IEEE Trans. Inform. Theory*, 42, 1996.

[ABMD92] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using wavelet transforms. *IEEE Trans. on Image Processing*, 1(2):205–220, April 1992.

[ABP05] J. Ascenso, C. Brites, and F. Pereira. Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding. In *EURASIP Conference on Speech and Image Processing, Multimedia Communications and Services*, Smolenice, Slovak Republic, June 2005.

[ABP06] J. Ascenso, C. Brites, and F. Pereira. Content adaptive Wyner-Ziv video coding driven by motion activity. In *Image Processing, 2006 IEEE International Conference on*, Atlanta, GA, October 2006.

[ACA+04] T. André, M. Cagnazzo, M. Antonini, M. Barlaud, N. Bozinović, and J. Konrad. (N,0) motion-compensated lifting-based wavelet transform. In *Proc. IEEE Intern. Conf. on Acoustics, Speech and Signal Processing*, Montreal, Canada, May 2004.

[ACA+07] M. A. Agostini, M. Cagnazzo, M. Antonini, G. Laroche, and J. Jung. State-of-the-art: study of the quantization for the coding of motion vectors in h.264 video coder. *Orange-labs project report 2*, september 2007.

[ACAB07] Thomas André, Marco Cagnazzo, Marc Antonini, and Michel Barlaud. JPEG2000-compatible scalable scheme for wavelet-based video coding. *EURASIP Journal on Image and Video Processing*, 2007:Article ID 30852, 11 pages, 2007.

[Ahl85] R. Ahlswede. The rate-distortion region for multiple descriptions without excess rate. IT-31(6):721–726, November 1985.

[And07] Thomas André. *Codage vidéo scalable et mesure de distorsion entropique*. PhD thesis, Université de Nice Sophia-Antipolis, France, 2007.

[Apo99] J. G. Apostolopoulos. Error-resilient video compression via multiple state streams. In *International Workshop on Very Low Bitrate Video Coding (VLVB)*, pages 168–171, Kyoto, Japan, oct 1999.

[Apo00a] J. G. Apostolopoulos. Error-resilient video compression through the use of multiple states. In *Proc. International Conference on Image Processing*, volume 3, pages 352–355, sept 2000.

[Apo00b] J. G. Apostolopoulos. Reliable video communication over lossy packet networks using multiple state encoding and path diversity. In *Visual Communications and Image Processing*, volume 4310, pages 392–409, oct 2000.

[ARG04] A. Aaron, S. Rane, and B. Girod. Wyner-Ziv video coding with hash-based motion compensation at the receiver. In *IEEE ICIP*, Singapore, 2004.

[ARSG04] A. Aaron, S. Rane, E. Setton, and B. Girod. Transform-domain Wyner-Ziv codec for video. In *Proc. SPIE Visual Communications and Image Processing conf.*, volume 0, San Jose, CA, 2004.

[ASG03] A. Aaron, E. Setton, and B. Girod. Towards practical Wyner-Ziv coding of video. In *Proc. IEEE ICIP*, volume 0, Barcelona, 2003.

[ATC07] E. Akyol, A. M. Tekalp, and M. R. Civanlar. A flexible multiple description coding framework for adaptive peer-to-peer

video streaming. *IEEE Journal of Selected Topics in Signal Processing*, 1:231–245, 2007.

[BBFPP01] V. Bottreau, M. Benetiere, B. Felts, and B. Pesquet-Popescu. A fully scalable 3D subband video codec. In *Proceedings of IEEE International Conference on Image Processing*, volume 2, pages 1017–1020, Thessaloniki, Greece, october 2001.

[BDV00] R. Balan, I. Daubechies, and V. Vaishampayan. The analysis and design of windowed Fourier frame based multiple description source coding schemes. 46(7):2491–2537, 2000.

[BFG04] G. Boisson, E. François, and C. Guillemot. Accuracy-scalable motion coding for efficient scalable video compression. In *IEEE International Conference on Image Processing*, pages 1309–1312, Singapore, october 2004.

[Bjo01] G. Bjontegaard. Calculation of average PSNR differences between RD-curves. In *VCEG Meeting*, Austin, USA, April 2001.

[BMV+05] J. Barbarien, A. Munteanu, F. Verdicchio, Y. Andreopoulos, J. Cornelis, and P. Schelkens. Motion and texture rate-allocation for prediction-based scalable motion vector coding. *Signal Processing: Image Communication*, 20(4):315–342, April 2005.

[BTS01] N. V. Boulgouris, D. Tzovaras, and M. G. Strintzis. Lossless image compression based on optimal prediction, adaptive lifting, and conditional arithmetic coding. *IEEE Transactions on Image Processing*, 10(1):1–14, jan 2001.

[B.U96] B.Usevitch. Optimal bit allocation for biorthogonal wavelet coding. pages 387–395, Snowbird, UT, April 1996.

[BV97] J. Batllo and V. Vaishampayan. Asymptotic performance of multiple description transform codes. 43(2):703–707, March 1997.

[BWR02] T. Berger-Wolf and E. Reingold. Index assignment for multichannel communication under failure. 48(10):2656–2668, October 2002.

[BZ83] T. Berger and Z. Zhang. Minimum breakdown degradation in binary source encoding. IT-29(6):807–814, November 1983.

[CAA+07] M. Cagnazzo, M. A. Agostini, M. Antonini, G. Laroche, and J. Jung. Project description: study of the quantization for the coding of motion vectors in h.264 video coder. *Orange-labs project report 1*, august 2007.

[CAAB04] M. Cagnazzo, T. André, M. Antonini, and M. Barlaud. A model-based motion compensated video coder with JPEG2000 compatibility. In *IEEE Intern. Conf. on Image Processing*, pages 2255–2258, Singapore, October 2004.

[CAB95] P. Charbonnier, M. Antonini, and M. Barlaud. Implantation d'une transformée en ondelettes 2D dyadique au fil de l'eau. *Rapport Contrat CNES/TBS, (896/95/CNES/1379/00)*, October 1995.

[CAC+09] S. Corrado, M. A. Agostini, M. Cagnazzo, G. Laroche, J. Jung, and M. Antonini. Improving H.264 performances by quantization of motion vectors. In *Proceedings of Picture Coding Symposium*, Chicago, USA, May 2009.

[Cag04] Marco Cagnazzo. *Wavelet transform and three-dimensional data compression*. PhD thesis, Université de Nice Sophia-Antipolis, France, 2004.

[CB08] J. Chen and T. Berger. Robust distributed source coding. *IEEE Trans. Information Theory*, 54(8):3385–3398, 2008.

[CDC09] M. A. Agostini C. Dikici, T. Maugey and O. Crave. Efficient frame interpolation for Wyner-Ziv video coding. In *Proc. of SPIE Electronical Imaging, Visual Communications and Image Processing conference (VCIP)*, San Jose, USA, January 2009.

[CDSY98] R. Calderbank, I. Daubechies, W. Sweldens, and B.-L. Yeo. Wavelet transforms that map integers to integers. *Applied and Computational Harmonic Analysis*, 5(3):332–369, 1998.

[CGPPT08] Olivier Crave, Christine Guillemot, Béatrice Pesquet-Popescu, and Christophe Tillier. Distributed temporal multiple description coding for robust video transmission. *EURASIP J. Wirel. Commun. Netw.*, 2008(1):1–9, 2008.

[Che05] C. W. Chen. Multiple description image coding for mobile wireless applications. In *IEEE Communications Society, Palm Beach Chapter*, apr 2005.

[CLKD06] J. C. Chiang, C. M. Lee, M. Kieffer, and P. Duhamel. Robust video transmission over mixed IP - wireless channels using motion-compensated oversampled filterbanks. In *Proceedings of ICASSP*, 2006. submitted.

[CMPP00] M. Cagnazzo, T. Maugey, and B. Pesquet-Popescu. A differential motion estimation method for image interpolation in distributed video coding. In *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 0, apr 2000.

[CMW99] P. Chou, S. Mehrotra, and A. Wang. Multiple description decoding of overcomplete expansions using projections onto convex sets. Snowbird, UT, 1999.

[CO97] C. Chrysafis and A. Ortega. Efficient context-based entropy coding lossy wavelet image compression. Snowbird, UT, March 1997.

[CO98] C. Chrysafis and A. Ortega. Line based, reduced memory, wavelet image compression. In *Data Compression Conference*, pages 398–407, Snowbird, USA, march 1998.

[CO00] C. Chrysafis and A. Ortega. Line based, reduced memory, wavelet image compression. *IEEE Transactions on Image Processing*, 9(3):378–389, 2000.

[Cov75] T. Cover. A proof of the data compression theorem of slepian and wolf for ergodic sources. *IEEE Trans. Inform. Theory*, 22:226–268, 1975.

[CS98] J. Conway and N. Sloane. *Sphere Packing, Lattices and Groups*. 3rd edition edition, 1998.

[CSOM03] D. Comas, R. Singh, A. Ortega, and F. Marqués. Unbalanced multiple description video coding with rate-distortion optimization. *EURASIP Special Issue Multimedia Signal Processing*, 1:81–90, January 2003.

[CSZ98] C. Chui, J. Spring, and L. Zhong. Integer wavelet transforms. Technical Report ISO/IEC JTC/SC29/WG1 N169, Teralogic Inc., Geneva, switzerland, march 1998.

[CT91] T. M. Cover and J. M. Thomas. *Elements of Information Theory*. Wiley, New–York, 1991.

[CVO95] C. Chakrabarti, M. Vishwanath, and R.M. Owens. A survey of architectures for the discrete and continuous wavelet transforms. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Detroit, USA, may 1995.

[CW99a] S. J. Choi and J.W. Woods. Motion-compensated 3-D subband coding of video. *IEEE Transaction on Image Processing*, 8(2):155–167, February 1999.

[CW99b] D.-M Chung and Y. Wang. Multiple description image coding using signal decomposition and reconstruction based on lapped orthogonal transforms. 9(6):895–908, September 1999.

[CW00] L. A. Da Silva Cruz and J. W. Woods. Adaptive motion vector quantization for video coding. In *IEEE Intern. Conf. on Image Processing*, pages 867–870, vol 2, Vancouver, Canada, September 2000.

[Dau92] I. Daubechies. *Ten lectures on wavelets*. SIAM, Philadelphia, PA, 1992.

[DD96] G. M. Davis and J. M. Danskin. Joint source and channel coding for image transmission over lossy packet networks. *Applications of Digital Image Processing XIX (A. G. Tescher, ed.), SPIE*, 2847, 1996.

[dis07] Discover, http://www.discoverdvc.org, 2007.

[DKG01] P. Dragotti, J. Kovacevic, and V. Goyal. Quantized oversampled filter banks with erasure. Snowbird, UT, march 2001.

[DS98] I. Daubechies and W. Sweldens. Factoring wavelet transforms into lifting steps. *J. Fourier Anal. Appl.*, 4(3):245–267, 1998.

[dST03] M. Van der Schaar and D. S. Turaga. Unconstrained motion compensated temporal filtering (UMCTF) framework for wavelet video coding. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Hong Kong, apr 2003.

[DSV02a] S. Diggavi, N Sloane, and V. Vaishampayan. Asymmetric multiple description lattice vector quantizers. 48(1):174–191, January 2002.

[DSV02b] P. Dragotti, S. Servetto, and M. Vetterli. Optimal filter banks for multiple description coding: Analysis and synthesis. 48(7):2036–2052, July 2002.

[DVKG+00] N. Damera-Venkata, T.D. Kite, W.S. Geisler, B.L. Evans, and A.C. Bovik. Image quality assessment based on a degradation model. *IEEE Transactions on Image Processing*, 9(4):636–650, avril 2000.

[EC91] W. Equitz and T. Cover. Sucessive refinement of information. 37(2):269–275, March 1991.

[EKKS07] A. El Essaili, S. Khan, W. Kellerer, and E. Steinbach. Multiple description video transcoding. In *Proc. IEEE International Conference on Image Processing ICIP 2007*, oct 2007.

[ess09] National research project ESSOR "ANR projet blanc", http://www.lss.supelec.fr/ essor/wiki/doku.php, 2009.

[FE99] M. Fleming and M. Effros. Generalized multiple-description vector quantization. Snowbird, Utah, March 1999.

[FV87] N. Farvardin and V. Vaishampayan. Optimal quantizer design for noisy channels: An approach to combined source-channel coding. 33(6):827–838, November 1987.

[FYMPP07] J. Farah, C. Yaacoub, F. Marx, and B. Pesquet-Popescu. Distributed coding of video sequences transmitted through error-prone channels. In *Proc. of IEEE 4th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications*, volume 0, mar 2007.

[GARRM05] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero. Distributed video coding. *Proc. of the IEEE*, Vol. 93(71):pp 71 – 83, Jan. 2005.

[GC82] A. Gamal and T. Cover. Achievable rates for multiple description. IT-28(6):851–857, November 1982.

[GG92] A. Gersho and R. M. Gray. Vector quantization and signal compression. *Kluwer Academic Press/Springer*, 1992.

[GGF02] T. Guionnet, C. Guillemot, and E. Fabre. Soft decoding of multiple descriptions. pages 26–29, Lausanne, Switzerland, August 2002.

[GGP01] T. Guionnet, C. Guillemot, and S. Pateux. Embedded multiple description coding for progressive image transmission over unreliable channels. pages 7–10, Thessalonique, Greece, October 2001.

[Gir93] B. Girod. Motion-compensating prediction with fractional-pel accuracy. *IEEE Transactions on Communications*, 41:604–612, 1993.

[GK98] V. Goyal and J. Kovacevic. Optimal multiple description transform coding of gaussian vectors. Chicago, IL, October 1998.

[GK01] V. Goyal and J. Kovacevic. Generalized multiple description coding with correlating transforms. 47(6):2199–2224, September 2001.

[GKAV98] V. Goyal, J. Kovacevic, R. Arean, and M. Vetterli. Multiple description transform coding of image. Chicago, USA, October 1998.

[GKD07] Avi Gabay, Michel Kieffer, and Pierre Duhamel. Joint source-channel coding using real BCH codes for robust image transmission. *IEEE Transactions on Image Processing*, 16(6):1568–1583, June 2007.

[GKK00] V. Goyal, J. Kelner, and J. Kovacevic. Multiple description lattice vector quantization: Variations and extensions. Snowbird, UT, March 2000.

[GKK02] V. Goyal, J. Kelner, and J. Kovacevic. Multiple description vector quantization with a coarse lattice. 48(3):781–788, March 2002.

[GKV98] V. Goyal, J. Kovacevic, and M. Vetterli. Multiple description transform coding: robustness to erasures using tight frame expansions. page 408, Cambridge, MA, August 1998.

[GKV99] V. Goyal, J. Kovacevic, and M. Vetterli. Quantized frame expansions as source-channel codes for erasure channels. Snowbird, UT, march 1999.

[GVT98] V. Goyal, M. Vetterli, and N. Thao. Quantized overcomplete expansions in $\mathbb{R}^n$: Analysis, synthesis, and algorithms. 44(1):16–31, January 1998.

[H26] JM 11.0 H.264/AVC reference software, http://iphome.hhi.de/suehring/tml/.

[Haa10] A. Haar. Zur Theorie der orthogonalen Funktionensysteme. *Mathematische Annalen*, pages 331–371, 1910.

[HAL06] B. A. Heng, J. G. Apostolopoulos, and J. S. Lim. End-to-end rate-distortion optimized MD mode selection for multiple description video coding. *EURASIP Journal on Applied Signal Processing*, 2006:Article ID 32592, 12 pages, 2006.

[HPPP06a] H.J.A.M. Heijmans, G. Piella, and B. Pesquet-Popescu. Adaptive wavelets for image compression using update lifting: Quantisation and error analysis. *International Journal of Wavelets, Multiresolution and Information Processing (IJWMIP)*, 1(4):41–65, 2006.

[HPPP06b] H.J.A.M. Heijmans, G. Piella, and B. Pesquet-Popescu. Combining seminorms in adaptive lifting schemes and applications to image analysis and compression. *Journal of Mathematical Imaging and Vision*, 25(2), 2006.

[HW00] S. Hsiang and J. Woods. Embedded image coding using zeroblocks of subband / wavelet coefficients and context modeling. In *Proceedings of IEEE International Symposium on Circuits and Systems*, pages 662–665, Geneva, Switzerland, may 2000.

[ISO93] ISO/IEC JTC1. *ISO/IEC 11172-2: Coding of Moving Pictures and Associeted Audio for Digital Sotrage Media at up to about 1.5 Mbit/s*, 1993.

[ISO00] ISO/IEC JTC1. *ISO/IEC 13818-2: Generic Coding of Moving Pictures*, 2000.

[ISO01] ISO/IEC JTC1. *ISO/IEC 14496-2: Coding of audio-visual objects*, April 2001.

[ITU99] ITU-T. *Recommendation H.261, Video codec for audiovisual services at $p \times 64$ kbits/s*, March 1999.

[Jay81] N. Jayant. Subsampling of a DPCM speech channel to provide two "self-contained" half-rate channels. 60(4):501–509, April 1981.

[JC81] N. Jayant and S. Christensen. Effects of packet losses in waveform coded speech and improvements due to an odd-even sample-interpolation procedure. COM-29(2):101–109, February 1981.

[JFB95] R. L. Joshi, T. R. Fischer, and R. H. Bamberger. Lossy encoding of motion vectors using entropy-constrained vector quantization. In *IEEE Intern. Conf. on Image Processing*, pages 3109, vol 3, Washington, USA, 1995.

[JO97] Wenqing Jiang and Antonio Ortega. Forward/backward adaptive context selection with applications to motion vector field encoding. In *IEEE Intern. Conf. on Image Processing*, Santa Barbara, CA, USA, october 1997.

[JO99a] W. Jiang and A. Ortega. Multiple description coding via polyphase transform and selective quantization. 1999.

[JO99b] W. Jiang and A. Ortega. Multiple description coding via scaling-rotation transform. In *Proc. of ICASSP 1999*, march 1999.

[Joi02] Joint Video Team of ISO/IEC MPEG and ITU-T VCEG. *Joint Committee Draft, JVT-C167*, May 2002.

[jpe00] ISO/IEC FCD 15444-1: JPEG 2000 final comitee draft version 1.0, http://www.jpeg.org/fcd15444-1.htm, 2000.

[JSX05] H. Jenka, T. Stockhammer, and Wen Xu. Permeable-layer receiver for reliable multicast transmission in wireless systems. In *Wireless Communications and Networking Conference, 2005 IEEE*, volume 3, pages 1805–1811Vol.3, 13-17 March 2005.

[JT99] H. Jafarkhani and V. Tarokh. Design of successively refinable trellis coded quantizers. 45(5):1490–1497, July 1999.

[KAM01] N. Kamaci, Y. Altunbasak, and R. Mersereau. Multiple description coding with multiple transmit and receive antennas for wireless channels: The case of digital modulation. pages 3272–3276, San Antonio, TX, November 2001.

[KDG02] J. Kovacevic, P. Dragotti, and V. Goyal. Filter bank frame expansions with erasures. 48:1439–1450, June 2002.

[KKL01] C.-S. Kim, R.-C. Kim, and S.-U. Lee. Robust transmission of video sequence using double-vector motion compensation. *IEEE Trans. Circuits Syst. Video Technol.*, 11:1011–1021, 2001.

[Kon04] J. Konrad. Transversal versus lifting approach to motion-compensated temporal discrete wavelet transform of image sequences: equivalence and tradeoffs. In *Proc. SPIE Visual Communications and Image Processing*, January 2004.

[KP97] B.-J. Kim and W.A. Pearlman. An embedded wavelet video coder using three-dimensional set partitioning in hierarchical trees (SPIHT). In *Proceedings Data Compression Conference*, pages 251–260, march 1997.

[KS00] J. Kovacevic and W. Sweldens. Wavelet families of increasing order in arbitrary dimensions. *IEEE Transactions on Image Processing*, 9(3):480–496, 2000.

[KV88] G. Karlsson and M. Vetterli. Three-dimensional subband coding of video. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 1100–1103, New York, USA, april 1988.

[LCK+05] F. Labeau, J. C. Chiang, M. Kieffer, P. Duhamel, L. Vandendorpe, and B. Mack. Oversampled filter banks as error correcting codes: theory and impulse correction. *IEEE Transactions on Signal Processing*, 53(12):4619–4630, 2005.

[LDJF04] L. A. Larzon, M. Degermark, L. E. Jonsson, and G. Fairhurst. The lightweight user datagram protocol (UDP-Lite). Technical Report RFC 3828, The Internet Society, 2004.

[Li01] W. Li. Overview of fine granularity scalability in MPEG-4 video standard. *IEEE Transactions on Circuits for Video Technology*, 11(3):301 – 317, march 2001.

[Liu91] M. Liu. Overview of the P * 64 kbits/s video coding standard. *Commun. ACM*, 34(4):60–63, April 1991.

[LLL+01] L. Luo, J. Li, S. Li, Z. Zhuang, and Y.-Q. Zhang. Motion compensated lifting wavelet and its application in video coding. In *Proc. of IEEE International Conference on Multimedia and Expo*, pages 481–484, Tokyo, Japan, 2001.

[LMWA05] J. N. Laneman, E. Martinian, G. W. Wornell, and J. G. Apostolopoulos. Source-channel diversity for parallel channels. *IEEE Trans. Inform. Theory*, 51(10):3518–3539, 2005.

[LRO97] S. LoPresto, K. Ramchandran, and M. Orchard. Image coding based on mixture modeling of wavelet coefficients and a fast estimation-quantization framework. Snowbird, UT, March 1997.

[LW95] Y. Y. Lee and J. W. Woods. Motion vector quantization for video coding. *IEEE Transactions on Image Processing*, 4(3):378–382, 1995.

[LWK06] M. H. Larsen, C. Weidmann, and M. Kieffer. Iterative decoding of entropy-constrained multiple description trellis-coded quantization. In *Proceedings of GLOBECOM*, 2006.

[Mal99] S. Mallat. *A wavelet tour of signal processing.* Academic Press, San Diego, USA, 1999.

[Mar84] T. Marshall. Coding of real-number sequences for error correction: A digital signal processing problem. *IEEE JSAC*, 2(2):381–392, March 1984.

[MC00] S. Mehrotra and P. Chou. On optimal frame expansions for multiple description quantization. Sorrento, Italy, June 2000.

[MCA09] G. Laroche J. Jung M. Cagnazzo, M. A. Agostini and M. Antonini. Motion vector quantization for efficient low bit-rate video coding. In *Proc. of SPIE Electronical Imaging, Visual Communications and Image Processing conference (VCIP)*, San Jose, USA, 2009.

[MCD+08] T. Maugey, O. Crave, C. Dikici, M.A. Agostini, and M. Kieffer. Single source video coding based on the DSC paradigm: intermediate report. *ESSOR project report 2.2*, october 2008.

[MDN93] F. Moscheni, F. Dufaux, and H. Nicolas. Entropy criterion for optimal bit allocation between motion and prediction error information. In *Proc. SPIE - Int. Soc. Opt. Eng.*, volume 2094, pages 235–242, 1993.

[MG03] R. Motwani and C. Guillemot. Tree-structured oversampled filter banks as joint source-channel codes: Applications to image transmission over erasure channels. *Under Review with IEEE Transactions on Signal Processing*, March 2003.

[MGM07] M. Maitre, C. Guillemot, and L. Morin. 3D model-based frame interpolation for distributed video coding of static scenes. *IEEE Trans. Image Processing*, 2007.

[MLKD08] C. Marin, Y. Leprovost, M. Kieffer, and P. Duhamel. Robust header recovery based enhanced permeable protocol layer mechanism. In *Proceedings SPAWC 2008*, July 2008.

[MMLB+07] M.G. Martini, M. Mazzotti, C. Lamy-Bergot, J. Huusko, and P. Amon. Content adaptive network aware joint optimization of wireless video transmission. *IEEE Communications Magazine*, pages 84–90, jan 2007.

[MMR99] A. Miguel, A. Mohr, and E. Riskin. SPIHT for generalized multiple description coding. volume 3, pages 842–846, 1999.

[MPE05] MPEG Wavelet Video AhG, Poznan MPEG 73th meeting. *Wavelet codec reference document and software manual. Doc. N7334*, july 2005.

[MR00] A. Miguel and E. Riskin. Protection of regions of interest against data loss in a generalized multiple description framework. Snowbird, UT, March 2000.

[MRL99a] A. Mohr, E. Riskin, and R. Ladner. Generalized multiple description coding through unequal loss protection. In *Proc. International Conference on Image Processing ICIP 99*, 1999.

[MRL99b] A. Mohr, E. Riskin, and R. Ladner. Graceful degradation over packet erasures channels through forward error correcting. Snowbird, UT, March 1999.

[MSAI04] M. Mrak, N. Sprljan, G. C. K. Abhayaratne, and E. Izquierdo. Scalable generation and coding of motion vectors for highly scalable video coding. In *Picture Coding Symposium*, San Francisco, USA, 2004.

[MT03] N. Mehrseresht and D. Taubman. Adaptively weighted update steps in motion compensated lifting based on scalable video compression. In *IEEE Intern. Conf. on Image Processing*, pages 771–774, Barcelona, Spain, 2003.

[MV95] S. McCanne and M. Vetterli. Joint source/channel coding for multicast packet video. In *Proc. IEEE International Conference on Image Processing ICIP 1995*, volume 1, pages 25–28, oct 1995.

[MWS06] D. Marpe, T. Wiegand, and G.J. Sullivan. The H.264/MPEG4 advanced video coding standard and its applications. *IEEE Communications Magazine*, pages 134–143, august 2006.

[Ohm94] J-R. Ohm. Three dimensional subband coding with motion compensation. *IEEE Trans. On Image Processing*, 3(5), November 1994.

[Ort00] A. Ortega. *Compressed Video over Networks*, chapter Variable Bit-rate Video Coding, pages 343–382. M. Sun and A. Reibman Eds, Marcel Dekker, New York, NY, 2000.

[OSWW08] Tobias Oelbaum, Heiko Schwarz, Mathias Wien, and Thomas Wiegand. Subjective performance evaluation of the SVC extension of H.264/AVC. In *Proceedings of IEEE International Conference on Image Processing (ICIP'08)*, San Diego, USA, october 2008.

[OWVR97] M. Orchard, Y. Wang, V. Vaishampayan, and R. Reibman. Redundancy rate-distortion analysis of multiple description coding using paiwise correlating transforms. 1997.

[Oza80] L. Ozarow. On a source-coding problem with two channels and three receivers. 59(10):1909–1921, December 1980.

[PA05] M. Pereira and M. Antonini. Multiple description coding with automatic redundancy control for video transmission over wireless networks. In *EUSIPCO, XIII European Signal Processing Conference*, Antalya, Turquey, sept 2005.

[PAB02] M. Pereira, M. Antonini, and M. Barlaud. Channel adapted multiple description coding scheme using wavelet transform. In *IEEE International Conference on Image Processing (ICIP)*, Rochester, NY, USA, sep 2002.

[PAB03a] C. Parisot, M. Antonini, and M. Barlaud. 3D scan-based wavelet transform and quality control for video coding. *EURASIP Journal on Applied Signal Processing, Special issue on Multimedia Signal Processing*, 2003(1):521–528, jan 2003.

[PAB03b] M. Pereira, M. Antonini, and M. Barlaud. Multiple description image and video coding for wireless channels. *EURASIP Special Issue on Recent Advances in Wireless Video*, 18(10):925–945, 2003.

[Par03] Christophe Parisot. *Allocations basées modèles et transformées en ondelettes au fil de l'eau pour le codage des images et des vidéos*. PhD thesis, Université de Nice Sophia-Antipolis, France, 2003.

[PCR03] S.S. Pradhan, J. Chou, and K. Ramchandran. Duality between source coding and channel coding and its extension to the side information case. *IEEE Transactions on Information Theory*, 49(5):1181–1203, may 2003.

[Per04] M. Pereira. Multiple description image and video coding for noisy channels. *PhD Thesis*, 2004.

[PJ07] S. Pateux and J. Jung. An Excel add-in for computing Bjontegaard metric and its evolution. In *VCEG Meeting*, Marrakech, MA, 2007.

[PJC03] V. N. Padmanabhan, H. J.Wang, and P. A. Chou. Resilient peer-to-peer streaming. In *Proc. IEEE International Conference on Network Protocols*, pages 16–27, oct 2003.

[PPB01] B. Pesquet-Popescu and V. Bottreau. Three-dimensional lifting schemes for motion compensated video compression. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1793–1796, Salt Lake City, USA, 2001.

[PPR01] R. Puri, S. S. Pradhan, and K. Ramchandran. (n,k) source channel erasure codes : can parity bits also refine quality? In *Proc. Conference on Information Sciences and Systems (CISS)*, march, 2001.

[PPR02a] R. Puri, S. S. Pradhan, and K. Ramchandran. N-channel multiple descriptions : theory and constructions. 2002.

[PPR02b] R. Puri, S. S. Pradhan, and K. Ramchandran. N-channel symmetric multiple descriptions : new rate regions. 2002.

[PR99] R. Puri and K. Ramchandran. Multiple description source coding using forward error correction (FEC) codes. Pacific Groove, CA, October 1999.

[PR03] R. Puri and K. Ramchandran. PRISM: A video coding architecture based on distributed compression principles. *Technical report, EECS Department, University of California, Berkeley*, (UCB/ERL M03/6), 2003.

[RAG05] S. Rane, A. Aaron, and B. Girod. Error-resilient video transmission using multiple embedded Wyner-Ziv descriptions. In *International Conference on Image Processing (ICIP)*, volume 2, Genova, Italy, sep 2005.

[RC98] J. Rogers and P. Cosman. Robust wavelet zerotree image compression with fixed length packetization. 5(5):105–107, May 1998.

[Rei02] A. Reibman. Optimizing multiple description video coders in a packet loss environment. In *Packet Video Workshop*, Pittsburgh, USA, mar 2002.

[RHW+99] A. Reibman, H., Y. Wang, M. Orchard, and R. Puri. Multiple description video coding using motion-compensated prediction. In *IEEE International Conference on Image Processing (ICIP)*, Kobe, Japan, oct 1999.

[RJW+02] A. R. Reibman, H. Jafarkhani, Y. Wang, M. T. Orchard, and R. Puri. Multiple- description video coding using motion-compensated temporal prediction. *IEEE Trans. Circuits Syst. Video Technol.*, 12:193–204, 2002.

[RR97] Shankar L. Regunathan and Kenneth Rose. Motion vectors quantization in a rate-distortion framework. In *Proc. IEEE Intern. Conf. on Image Processing*, volume 2, pages 21–24, Washington DC, USA, 1997.

[RS07] Julio Rolon and Philippe Salembier. Generalized lifting for sparse image representation and coding. In *Proc. of Picture Coding Symposium*, Lisbon, Portugal, 2007.

[RSA08] Julio Rolon, Philippe Salembier, and Xavier Alameda. Image compression with generalized lifting and partial knowledge of the signal pdf. In *IEEE proc. of International Conference of Signal Processing*, San Diego, USA, 2008.

[SB91] G. J. Sullivan and R. L. Baker. Rate-distortion optimized motion compensation for video compression using fixed or variable size blocks. In *Proc. of IEEE Global Telecomm*, pages 85–90, 1991.

[SG88] Y. Shoham and A. Gersho. Efficient bit allocation for an arbitrary set of quantizers. 36(9):1445–1453, 1988.

[Sha59] C. Shannon. Coding theorems fos a discrete source with fidelity criterion. 7(4):142–163, March 1959.

[Sha93] J.M. Shapiro. Embedded image coding using zerotrees of wavelets coefficients. *IEEE Transactions on Signal Processing*, 41(12):3445–3462, december 1993.

[Sik97] T. Sikora. MPEG digital video-coding standard. September 1997.

[SJA04] A. Sehgal, A. Jagmohan, and N. Ahuja. Wyner-Ziv coding of video: an error-resilient compression framework. *IEEE Transactions on Multimedia*, 6(2):249–258, 2004.

[SNR06] A. Saxena, J. Nayak, and K. Rose. On efficient quantizer design for robust distributed source coding. In *Proceedings of Data Compression Conference*, pages 63–72, mar 2006.

[SO00] P. Sagetong and A. Ortega. Optimal bit allocation for channel-adaptive multiple description coding. San Jose, CA, January 2000.

[SO01] P. Sagetong and A. Ortega. Analytical model-based bit allocation for wavelet coding with applications to multiple description coding and region of interest. Tokyo, Japan, August 2001.

[SO03] R. Singh and A. Ortega. Erasure recovery in predictive coding environment using multiple description coding. *Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology*, 34(1-2):9–28, may 2003.

[SP96] A. Said and W. Pearlman. A new, fast and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 6(3):243–250, juin 1996.

[Sri99] M. Srinivasan. Iterative decoding of multiple descriptions. In *Proceedings of Data Compression Conference*, 1999.

[SRVN98] S. Servetto, K. Ramchandran, V. Vaishampayan, and K. Nahrstedt. Multiple description wavelet based image coding. In *International Conference on Image Processing*, Chicago, USA, October 1998.

[SRVN00] S. Servetto, K. Ramchandran, V. Vaishampayan, and K. Nahrstedt. Multiple description wavelet based image coding. *Transactions on Image Processing*, 9(5):813–826, 2000.

[SS07] J. Solé and P. Salembier. Generalized lifting prediction optimization applied to lossless image compression. *IEEE Signal Processing Letters*, 14(10):1–14, oct 2007.

[ST03] A. Secker and D. Taubman. Lifting-based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression. *IEEE Transaction on Image Processing*, 12(12):1530–1542, 2003.

[ST04] A. Secker and D. Taubman. Highly scalable video compression with scalable motion coding. *IEEE Transactions on Image Processing*, 13(8):1029–1041, 2004.

[STL04] G. J. Sullivan, P. Topiwala, and A. Luthra. The H.264/AVC advanced video coding standard : Overview and introduction to the fidelity range extensions. *SPIE Conference on Applications of Digital Image Processing XXVII, special Session on Advances in the New Emerging Standard : H264 / AVC*, 5558:454–474, 2004.

[SVJ97] S.McCanne, M. Vetterli, and V. Jacobson. Low-complexity video coding for receiver- driven layered multicast. *IEEE Journal on Selected Areas in Communications*, 15:983–1001, 1997.

[SVS99] S. Servetto, V. Vaishampayan, and N. Sloane. Multiple description lattice vector quantization. Snowbird, UT, 1999.

[SW73] D. Slepian and J. K. Wolf. Noiseless coding of correlated information sources. *IEEE Trans. on Information Theory*, Vol. 19:pp 471–480, July 1973.

[SW98] Gary J. Sullivan and Thomas Wiegand. Rate-distortion optimization for video compression. 15:74–90, November 1998.

[SW05] G.J. Sullivan and T. Wiegand. Video compression-from concepts to the H.264/AVC standard. *Proceedings of the IEEE*, 93(1):18–31, january 2005.

[SWS03] R. Schäfer, T. Wiegand, and H. Scharwtz. The emerging H.264/AVC standard. *EBU Technical Review*, 2003.

[Tau00] D. Taubman. High performance scalable image compression with EBCOT. *IEEE Transactions on Image Processing*, 9(7):1158–1170, july 2000.

[Til05] C. Tillier. Scalabilité et robustesse dans le codage vidéo à base d'ondelettes. *PhD Thesis*, june 2005.

[TM02] D. Taubman and M.W. Marcellin. JPEG2000: standard for interactive imaging. *Proceedings of the IEEE*, 90(8):1336–1357, august 2002.

[TMRT06] M. Tagliasacchi, A. Majumdar, K. Ramchandran, and S. Tubaro. Robust wireless video multicast based on a distributed source coding approach. *Signal Processing*, 86(11):3196–3211, 2006.

[TMTS06] M. Tagliasacchi, D. Maestroni, S. Tubaro, and A. Sarti. Motion estimation and signaling techniques for 2D+t scalable video coding. *EURASIP Journal on Applied Signal Processing*, 2006:Article ID 57308, 21 pages, 2006. doi:10.1155/ASP/2006/57308.

[TO07] I.H. Tseng and A Ortega. Rate-distortion analysis and bit allocation strategy for motion estimation at the decoder using maximum likelihood technique in distributed video coding. In *Proc. of IEEE ICIP*, volume 0, 2007.

[TPPdS04] C. Tillier, B. Pesquet-Popescu, and M. Van der Schaar. Unconstrained motion compensated temporal filtering (UMCTF) framework for wavelet video coding. In *Proc. of EUropean Signal Processing COnference (EUSIPCO)*, Vienna,Austria, sep 2004.

[TPPP07] C. Tillier, T. Petrisor, and B. Pesquet-Popescu. A motion-compensated overcomplete temporal decomposition for multiple description scalable video coding. *Special issue of the International Journal on Image and Video Processing (IJIVP) on "Wavelets in Source Coding, Communications, and Networks"*, pages 2325–2384, january 2007.

[TZL99] W. Trappe, H. Zheng, and K. J. Ray Liu. Adaptive lifting coding scheme for video scene changes. In *IEEE proc. of MSP99*, pages 321–326, Copenhagen, Denmark, september 1999.

[UB03] M. Unser and T. Blu. Mathematical properties of the JPEG2000 wavelet filters. *IEEE Transactions on Image Processing*, 12(9):1080–1090, september 2003.

[Vai91] V. Vaishampayan. Vector quantizer design for diversity systems. In *CISS*, 1991.

[Vai93] V. Vaishampayan. Design of multiple description scalar quantizers. 39(3):821–834, 1993.

[VB98] V. Vaishampayan and J. Batllo. Asymptotic analysis of multiple description quantizers. 44(1):278–283, January 1998.

[VD94] V. Vaishampayan and J. Domaszewicz. Design of entropy-constrained multiple description scalar quantizers. 40(1):245–250, 1994.

[VGP02] J. Viéron, C. Guillemot, and S. Pateux. Motion compensated 2D+t wavelet analysis for low rate fgs video compression. In *Proc. of Tyrrhenian Intern. Workshop on Digital Comm.*, Capri, Italy, september 2002.

[Vis94] M. Vishwanath. The recursive pyramid algorithm for the discrete wavelet transform. *IEEE Transactions on Signal Processing*, 42(3):673–676, march 1994.

[VJ99] V. Vaishampayan and S. John. Interframe balanced-multiple description video compression. In *IEEE International Conference on Image Processing (ICIP)*, Kobe, Japan, oct 1999.

[VKG01] R. Venkataramani, G. Kramer, and V. Goyal. Bounds on the achievable region for certain multiple description coding problems. Washington, DC, 2001.

[VSS01] V. Vaishampayan, N. Sloane, and S. Servetto. Multiple description vector quantization with lattice codebooks: Design and analysis. 47(5):1718–1734, 2001.

[WBSS04] Z. Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, april 2004.

[WCO05] Huisheng Wang, N.-M. Cheung, and Antonio Ortega. A framework for adaptive scalable video coding using wyner-ziv techniques. *EURASIP Journal on Applied Signal Processing*, pages 1–18, 2005.

[WFI05] G. Wang, S. Futemma, and E. Itakura. FEC-based scalable multiple description coding for overlay network streaming. In *Proc. CCNC Consumer Communications and Networking Conference 2005 Second IEEE*, pages 406–410, jan 2005.

[Wit80] H. Witsenhausen. On source networks with minimal breakdown degradation. 59(6):1083–1087, July-August 1980.

[Wit81] H. Witsenhausen. Minimizing the worst-case distortion in channel splitting. 60(8):1979–1983, 1981.

[WL02] Y. Wang and S. Lin. Error-resilient video coding using multiple description motion compensation. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(6):438–452, june 2002.

[WOR97] Y. Wang, M. Orchard, and A. Reibman. Multiple description image coding for noisy channels by paring transform coefficients. 1997.

[WOR98] Y. Wang, M. Orchard, and A. Reibman. Optimal pairwise correlating transform for multiple description coding. Chicago, IL, October 1998.

[WRL05] Y. Wang, A. R. Reibman, and S. Lin. Multiple description coding for video delivery. *Proceedings of the IEEE*, 93(1):57–70, 2005.

[WSBL03] Thomas Wiegand, Gary J. Sullivan, Gisle Bjøntegaard, and Ajay Luthra. Overview of the H.264/AVC video coding standard. 13(7):560–576, july 2003.

[WSJ+03] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan. Rate-constrained coder control and comparison of video coding standards. 13(7):688–703, july 2003.

[WVC04] M. Wu, A. Vetro, and C. W. Chen. Multiple description image coding with distributed source coding and side information. In *SPIE Multimedia Systems and Applications VII*, dec 2004.

[WW81] W. Witsenhausen and A. Wyner. Source coding for multiple descriptions II: A binary source. 60(10):2281–2292, December 1981.

[WWZ80] J. Wolf, A. Wyner, and J. Ziv. Source coding for multiple descriptions. 59(8):1417–1427, October 1980.

[WZ76] A. Wyner and J. Ziv. The rate-distortion function for source coding with side information at the receiver. *IEEE Trans. on Information Theory*, Vol. 22:pp 1–11, Jan. 1976.

[XXW+04] R. Xiong, J. Xu, F. Wu, S. Li, and Y. Zhang. Layered motion estimation and coding for fully scalable 3D wavelet video coding. In *IEEE Intern. Conf. on Image Processing*, pages 2271–2274, vol 4, Singapore, 2004.

[YFMPP07] C. Yaacoub, J. Farah, F. Marx, and B. Pesquet-Popescu. Performance analysis of a distributed video coding system - application to broadcasting over an error-prone channel. In *Proc. of EUSIPCO*, volume 0, 2007.

[YR00] X. Yang and K. Ramchandran. Optimal subband filter banks for multiple description coding. 46(7):2477–2490, November 2000.

[YWK00] J. Wen Y. Wang, S. Wenger and A. K. Katsaggelos. Error resilient video coding techniques. *IEEE Signal Processing Magazine*, 17:61–82, 2000.

[ZB87] Z.Zhang and T. Berger. New results in binary multiple description. IT-33(4):502–521, July 1987.

Publications

Journal papers

1. **M. A. Agostini**, M. Cagnazzo, S. Corrado, J. Jung and M. Antonini, *A new H.264 coding mode based on motion vector quantization*, In preparation for IEEE Transactions on Circuits and Systems for Video Technology.

International conferences with review comitee

1. **M. A. Agostini**, M. Antonini and M. Kieffer, *MAP Estimation of Multiple Description Encoded Video Transmitted over Noisy Channels*, To appear in IEEE International Conference on Image Processing, Cairo, Egypt, November 2009.
2. S. Corrado, **M. A. Agostini**, M. Cagnazzo, G. Laroche, J. Jung and M. Antonini, *Improving H.264 performances by quantization of motion vectors*, in Proceedings of Picture Coding Symposium, Chicago, USA, May 2009.
3. C. Dikici, T. Maugey, **M. A. Agostini** and O. Crave, *Efficient frame Interpolation for Wyner-Ziv Video Coding*, in Proceedings of SPIE Electronical Imaging, Visual Communications and Image Processing conference, San Jose, USA, January 2009.
4. M. Cagnazzo, **M. A. Agostini**, G. Laroche, J. Jung and M. Antonini, *Motion vector quantization for efficient low bit-rate video coding*, in Proceedings of SPIE Electronical Imaging, Visual Communications and Image Processing conference, San Jose, USA, January 2009.
5. **M. A. Agostini**, and M. Antonini, *Multiple description video decoding using MAP*, in Proceedings of IEEE International Conference on Image Processing, San Diego, USA, October 2008.
6. **M. A. Agostini**, and M. Antonini, *Motion-adapted weighted lifting scheme for MCWT video coders*, in Proceedings of Picture Coding Symposium, Lisboa, Portugal, November 2007.

7. **M. A. Agostini**, and M. Antonini, *Theoretical model of the coding error in MCWT video coders*, in Proceedings of IEEE International Conference on Image Processing, Atlanta, USA, October 2006.

8. **M. A. Agostini**, M. Antonini, and M. Barlaud, *Model-based bit allocation between wavelet subbands and motion information in MCWT video coders*, in Proceedings of EUropean SIgnal Processing COnference, Florence, Italy, September 2006.

9. **M. A. Agostini**, T. André, M. Antonini, and M. Barlaud, *Modeling the motion coding error for MCWT video coders*, in Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Toulouse, France, May 2006.

10. **M. A. Agostini**, T. André, M. Antonini, and M. Barlaud, *Scalable motion coding for video coders with lifted MCWT*, in Proceedings of International Workshop on Very Low Bit-rate Video-coding, Sardinia, Italy, September 2005.

National conferences with review comitee

1. **M. A. Agostini**, M. Antonini, and M. Barlaud, *Allocation optimale de débit entre le mouvement et les coefficients d'ondelettes pour la compression vidéo basée ondelettes*, in Proceedings of Colloque GRETSI, Troyes, France, September 2007.

2. **M. A. Agostini**, T. André, M. Antonini and M. Barlaud, *Codage scalable des vecteurs mouvement pour la compression de vidéos basée ondelettes*, in Proceedings of CORESA Conference, Rennes, France, november 2005.

Research reports

1. T. Maugey, O. Crave, C. Dikici, **M. A. Agostini**, and M. Kieffer, *Single source video coding based on the DSC paradigm: intermediate report*, ESSOR project report 2.2, October 2008.

2. **M. A. Agostini**, M. Antonini, O. Crave, M. Kieffer, A. Roumy, and V. Toto-Zarasoa, *Robust DSC schemes: intermediate report*, ESSOR project report 2.4, July 2008.

3. M. Antonini, **M. A. Agostini**, and M. Kieffer, *State of the art of single source video coding based on the DSC paradigm*, ESSOR project report 1.2, August 2007.

4. M. Cagnazzo, **M. A. Agostini**, M. Antonini, G. Laroche, and J. Jung, *Project description: study of the quantization for the coding of motion vectors in H.264 video coder*, Orange-labs project report 2, September 2007.

5. **M. A. Agostini**, M. Cagnazzo, M. Antonini, G. Laroche, and J. Jung, *State-of-the-art: study of the quantization for the coding of motion vectors in H.264 video coder*, Orange-labs project report 1, August 2007.

6. **M. A. Agostini**, *Etude du codage des vecteurs mouvement pour les codeurs vidéo de nouvelle génération*, Master thesis, September 2005.

Seminars

1. **M. A. Agostini** and M. Antonini, *Multiple description decoding for video transmission over noisy channels*, Popsud Seminar, INRIA Sophia-Antipolis, France, 18 September 2008.

2. **M. A. Agostini**, M. Antonini, and M. Barlaud, *Codage Avec Pertes des Vecteurs Mouvement et Allocation Optimale de Débit pour la Compression Vidéo basée Ondelettes*, GdR ISIS seminar, ENST Paris, France, 2 May 2007.

Résumé

La problématique principale de cette thèse est la compression de masses de données vidéo haute résolution. Nous proposons un schéma de compression vidéo par transformée en ondelettes compensée en mouvement. Plus précisément, dans le but de réduire le coût des vecteurs mouvement parfois trop élevé dans ce type de schéma, nous avons développé une approche de quantification avec pertes de ces vecteurs, permettant d'adapter leur précision tout en respectant le compromis débit / distorsion. Cette approche permet d'améliorer considérablement les performances du codeur, spécialement à bas débit. Pour modéliser l'influence de l'introduction de perte sur l'information de mouvement, nous avons établi un modèle théorique de distorsion de l'erreur de codage, et, enfin, nous avons réalisé une allocation de débit optimale basée modèle entre les vecteurs mouvement et les coefficients d'ondelettes.

Pour éviter certains artefacts dus à une mauvaise estimation du mouvement, nous avons ensuite amélioré le schéma lifting utilisé pour la transformée en ondelettes par une approche novatrice : les coefficients du schéma lifting sont adaptés à la norme des vecteurs mouvement.

Notre méthode de quantification des vecteurs mouvement a par ailleurs été appliquée au codeur H.264, la norme actuelle de compression vidéo pour la Haute Définition.

Enfin, nous avons travaillé sur le Codage par Descriptions Multiples, une approche de codage conjoint source / canal pour la compression robuste de vidéos utilisée dans la transmission sur des canaux de communication bruités. Nous avons développé un codeur vidéo robuste, par des approches de Codage par Descriptions Multiples dans le domaine transformé. Une allocation de débit est réalisée au codeur pour répartir le débit des coefficients d'ondelettes entre les différentes descriptions, en fonction des paramètres du canal. Plus particulièrement, pour reconstruire au mieux la vidéo en sortie du canal, nous avons réalisé des approches de décodage optimal, basées sur la connaissance des densités de probabilités des sous-bandes des différentes descriptions, sur un modèle de canal et sur des probabilités a posteriori. En parallèle, le codage de source vidéo distribué a également été exploré.

Mots-clés : codage d'images et de vidéo, transformée en ondelettes, allocation de débit, compromis débit-distorsion, modèle de distorsion, codage par descriptions multiples, codage vidéo distribué.

Abstract

The framework of the thesis is a wavelet-based video coder. Fully scalable, this video encoder is based on a lifted motion-compensated wavelet transform. The first challenge was to reduce the cost of the motion vectors, which can be prohibitive at low bit-rates, by quantizing with losses the vectors. This method has been applied to the H.264 coder. The goal is to find the optimal bit-rates for the motion vectors and for the temporal wavelet coefficients in order to minimize the total distortion. A theoretical distortion model has thus been established, and an optimal bit-rate allocation has been realized.

The influence of some badly estimated motion vectors on the motion-compensated wavelet transform has also been minimized. The steps of the lifting scheme have been closely adapted to the energy of the motion.

To deal with the problems of efficient video transmission over noisy channels, Multiple Description Coding (MDC) has been explored. The framework is a balanced MDC scheme for scan-based wavelet transform video coding. A focus is done on the joint decoding of descriptions received at decoder and corrupted by noise. The challenge is to reconstruct a central signal with a distortion as small as possible using the knowledge of the probability density function of the descriptions, by two different algorithms. Distributed video coding has also been explored.

Keywords: image and video coding, wavelet transform, bit allocation, trade-off rate-distortion, distortion model, multiple description coding, distributed video coding.