Backward compatible approaches for the compression of high dynamic range videos

Mikaël Le Pendu

▶ To cite this version:

ANNÉE 2016

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Bretagne Loire

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Informatique

École doctorale Matisse

présentée par

Mikaël LE PENDU

préparée au centre de recherche INRIA Rennes - Bretagne Atlantique
et à Technicolor R&I

Backward Compatible Approaches for the Compression of High Dynamic Range Videos

Thèse soutenue à Rennes
le 17 Mars 2016

devant le jury composé de :

Céline LOSCOS
Professor, Univ. de Reims / *Présidente*

Alan CHALMERS
Professor, Univ. of Warwick / *Rapporteur*

Frédéric DUFAUX
Research Director, Télécom ParisTech / *Rapporteur*

Olivier DEFORGES
Professor, INSA de Rennes / *Examinateur*

Christine GUILLEMOT
Research Director, INRIA / *Directrice de thèse*

Dominique THOREAU
Senior Scientist, Technicolor / *Co-directeur de thèse*

à la mémoire de ma mère.

Acknowledgments

J'adresse tout d'abord mes plus sincères remerciements à Christine Guillemot et Dominique Thoreau pour m'avoir encadré pendant ces trois années. Vos remarques et votre expérience ont été particulièrement bénéfiques pour la réalisation de cette thèse.

Je tiens également à remercier les membres du jury pour le temps accordé à la lecture de mon manuscrit et pour l'intérêt porté à mes travaux. Merci donc aux Pr. Alan Chalmers et Dr. Frédéric Dufaux d'avoir accepté d'être rapporteur de ma thèse, au Pr. Olivier Déforges d'avoir accepté d'être examinateur, et enfin au Pr. Céline Loscos pour avoir présidé mon jury de thèse.

Merci aux doctorants, post-docs, stagiaires, permanents, et surtout amis de l'INRIA et Technicolor pour tous les bons moments et pour la bonne ambiance au travail comme à l'extérieur avec les mardis fous et autres after-work. Et bien sûr, merci pour tous les secrets de Polichinelle échangés en bandes organisées sous les cocotiers.

Enfin je voudrais terminer par un grand merci à toute ma famille. Je tiens en particulier à rendre hommage à ma mère pour tout l'amour qu'elle nous a toujours porté, à mon père, ma sœur et moi, et sans qui je ne serais pas arrivé jusque-là.

Contents

Résumé en Français

Contexte

La compression d'images et de vidéos est essentielle pour la télévision numérique, le streaming vidéo, l'échange de photos sur internet et bien d'autres applications. Tandis que l'évolution des technologies permet d'accroitre les capacités de stockage et de bande passante, la quantité d'images et vidéos en circulation sur internet augmente à un tel rythme qu'il reste indispensable de compresser ces données à des taux toujours plus élevés. Selon les prévisions récemment publiées par Cisco [1], le contentu vidéo qui représentait déjà 64% du trafic internet global en 2014, devrait atteindre le chiffre impressionnant de 80% en 2019. Bien que la comparaison du nouveau standard MPEG HEVC avec son prédécesseur MPEG AVC montre une division par deux du débit à qualité d'image comparable, de tels chiffres indiquent que les efforts de normalisation doivent être poursuivis.

Par ailleurs, l'amélioration des technologies de capture et d'écrans oriente la demande vers des contenus de meilleure qualité et des nouveaux formats d'image. La standardisation du format 4K, ou Ultra Haute Définition (UHD) pour la télévision est un exemple d'actualité. Mais outre l'augmentation du nombre de pixels, il existe plusieurs manières d'accroitre la qualité d'expérience pour la télévision ou le cinéma. Par exemple, la technologie d'écrans 3D a suscité un fort intérêt ces dernières années. Cet engouement a été notamment stimulé par la sortie en 3D du film “Avatar” en 2009. La capture et l'affichage vidéo à haute fréquence, ou HFR (pour High Frame Rate en anglais), constitue également une avancée vers plus de réalisme. Enfin, la qualité des images et vidéos peut être améliorée en augmentant la gamme de couleurs et de luminances pouvant être représentée par chaque pixel. Les technologies concernées sont appelées Wide Color Gamut (WCG) pour la couleur et High Dynamic Range (HDR) pour la luminance. Une image HDR peut donc contenir à la fois des régions très lumineuses tout en gardant d'autres zones très sombres. En permettant l'affichage de telles images, les dernières avancées dans le développement d'écrans HDR ont joué un rôle de catalyseur dans l'intérêt désormais porté à la création et la distribution de ce type de contenu. Tous les acteurs de la chaine de la vidéo sont alors concernés, des studios de cinéma aux fabricants de téléviseurs en passant par les chaines de télévision et les organismes de standardisation.

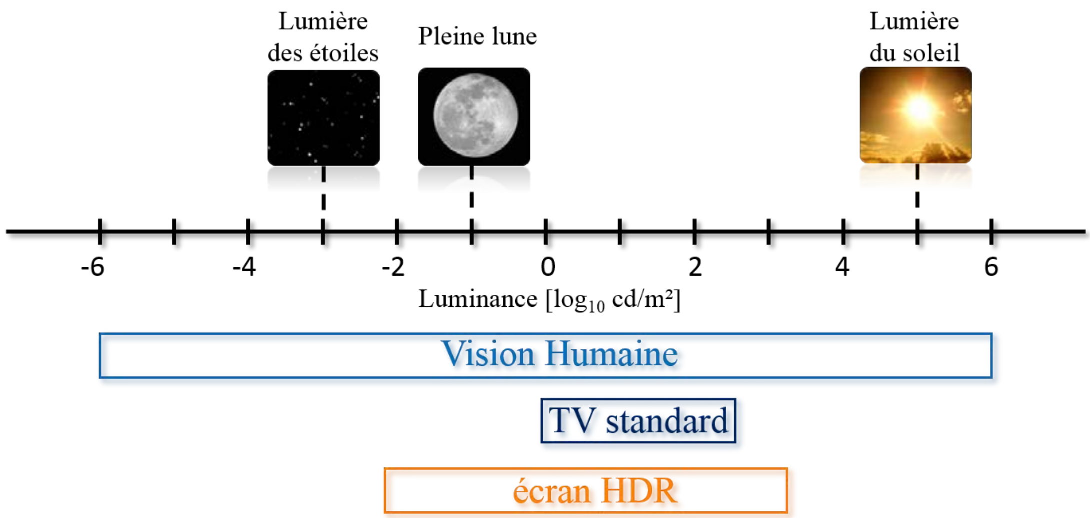


Figure 1: Gammes de luminances de la vision humaine et des écrans.

Motivations

La dynamique d'une image est caractérisée par le ratio entre les valeurs maximale et minimale de luminance, mesurées en candela par mètre carré (cd/m^2). La gamme de luminances perçue par l'œil humain se situe entre 10^{-6} et 10^6 cd/m^2, c'est à dire un ratio de 10^{12} : 1. Un écran classique, cependant, permet seulement d'atteindre des ratios d'environ 300:1. Les gammes de luminances typiques de la vision humaine et des écrans de télévision sont illustrées par la figure 1. Des algorithmes de traitement d'image appelés opérateurs de tone mapping sont alors nécessaires pour réduire la dynamique des images HDR et ainsi permettre leur affichage sur une télévision standard. Bien que la création d'images et vidéos HDR soit possible depuis longtemps par le rendu d'images de synthèse mais également en prise de vue réelle par des techniques appelées bracketing, les applications de l'imagerie HDR restaient limitées par les technologies d'affichage. L'arrivée des écrans HDR atteignant une luminances minimale d'environ 10^{-2} cd/m^2 pour une luminance maximale de 10^3 cd/m^2 ou plus, rend désormais possible la visualisation directe de ces images (i.e. sans tone mapping) pour une meilleure expérience visuelle. Mais ces nouvelles possibilités s'accompagnent de nouveaux défis quant à l'encodage et la distribution de contenu HDR à grande échelle :

- Le premier problème à résoudre est celui de la définition d'une nouvelle représentation standardisée de la lumière et la couleur en accord avec les capacités humaines. Les images traditionnelles, que l'on appellera ici LDR (pour Low Dynamic Range), représentent généralement l'information de couleur par un entier codé sur 8 bits pour chacune des composantes rouge vert et bleu (RVB). Cette représentation est adaptée aux écrans LDR classiques. En revanche, les formats d'images HDR existants n'ont pas été définis en fonction des capacités des écrans car ces formats sont antérieurs au développement des technologies d'affichage HDR. A la place, les valeurs physiques de luminance sont directement enregistrées en haute précision par des données à virgule flottante. Cela présente l'avantage de conserver toute l'information capturée, ce qui peut être très utile en particulier lors des étapes d'édition du contenu comme les corrections de couleur et des contrastes. Mais la quantité d'information stockée est très importante et peut être surdimensionnée par rapport aux capacités de la vision humaine en termes de dynamique et de précision. De plus, les méthodes de compression classiques sont conçues pour ne manipuler que des entiers. Il faut donc trouver de nouvelles manières de représenter les couleurs, ce qui inclut la définition d'un espace de couleur adapté à la perception humaine ainsi que la détermination du nombre de bits suffisant pour encoder ces données sans perte visible.

- Un second problème se pose quant à la compatibilité entre les équipements LDR et HDR. Alors que les futures télévisions seront équipées pour décoder et afficher des données HDR, ce format ne sera pas supporté par les équipements actuels. Une solution évidente consisterait à diffuser la version HDR du contenu ainsi qu'une version LDR obtenue par tone mapping sur des chaines différentes. Cette solution appelée Simulcast a été utilisée par exemple pour le passage de la télévision en définition standard (SD) à la Haute définition (HD). Cependant, le coût du Simulcast est élevé car une information similaire doit être transmise deux fois. Par ailleurs, la multiplication des nouveaux formats d'images (e.g. HD, UHD, HDR) accentue le problème puisque plusieurs types de compatibilités doivent être maintenues dans le même temps. Par conséquent, de nouvelles techniques de compression sont nécessaires afin de conserver une rétro-compatibilité à moindre coût.

Contributions

Bien que la question de la représentation des couleurs dans les formats HDR soit prise en compte, le travail de cette thèse se concentre principalement sur le second problème de rétro-compatibilité évoqué précédemment.

La première partie du travail consiste à se servir de standards de compression existants sans y apporter de modification afin d'encoder à la fois du contenu LDR et HDR par un schéma de compression simple couche. Dans cette approche, l'image HDR est d'abord convertie en LDR par un opérateur de tone mapping avant d'être compressée. Des méta-données sont également transmises avec le flux de données pour spécifier comment doit être inversé l'opérateur de tone mapping. Les anciens équipements ne décodent que l'image LDR tandis que les décodeurs HDR peuvent également lire les méta-données pour effectuer l'opération de tone mapping inverse. Le principal inconvénient de cette approche est la perte d'information liée à l'opérateur de tone mapping qui réduit le nombre de bits de la représentation et donc la précision des données. Afin de préserver la plupart des détails de l'image originale dans la version LDR malgré la perte de précision, certains opérateurs de tone mapping accentuent les contrastes localement au lieu d'appliquer la même courbe de tone mapping à tous les pixels de l'image. L'inversion de ces opérateurs locaux est cependant plus compliquée et nécessite généralement plus de méta-données. Partant du constat que le choix d'un opérateur de tone mapping influence la qualité de reconstruction de l'image HDR finale mais aussi le débit de compression, nous étudions comment optimiser cette étape pour garder une qualité d'image HDR maximale à un débit donnée.

Malgré les résultats satisfaisants obtenus par l'approche simple couche en termes d'efficacité de compression, ce type de solution n'est pas dans l'intérêt du producteur de contenu puisque la version LDR est automatiquement générée par l'encodeur et ne peut donc pas être définie par un processus artistique. Pour cette raison, la suite des travaux de cette thèse est consacrée au développement de schémas de compression scalable avec une couche LDR et une couche HDR. Dans cette approche, la version LDR est d'abord encodée dans un format standard, puis la version HDR est compressée par une méthode scalable se servant de la couche LDR décodée pour réduire les redondances, et donc la quantité d'information nécessaire pour l'encodage de la couche HDR. Les méthodes développées au cours de la thèse ont été conçues pour rester efficaces même dans le cas complexe où la version LDR du contenu a été générée par un opérateur de tone mapping local.

Dans une dernière contribution, nous cherchons à améliorer les schémas de codage scalables en prenant en compte la manière dont l'information chromatique est généralement traitée par les opérateurs de tone mapping. Ces travaux ont aussi été l'occasion d'explorer une représentation de la couleur différente de ce qui est habituellement utilisé en compression. Cette alternative, mieux adaptée à notre schéma scalable a également des avantages pour la compression d'images et vidéos HDR de manière plus générale.

Résumé par chapitre

Ce manuscrit de thèse est organisé en deux parties. La première partie contient deux chapitres présentant l'état de l'art dans les domaines de la compression d'images et vidéos et de l'imagerie HDR. La seconde partie décrit nos contributions en compression HDR :

Chapitre 1 : Ce chapitre présente les concepts fondamentaux de la perception de la lumière par l'œil humain afin de comprendre comment sont mesurées la quantité de lumière et la couleur en photométrie et en colorimétrie. Des aspects plus haut niveau du système visuel humain sont également abordés avec les représentations perceptuellement uniformes de la couleur. L'encodage couleur classique des images LDR, bien que relié à ces principes de colorimétrie et d'uniformité perceptuelle, a initialement été conçu pour être directement adapté aux technologies d'écran LDR conventionnelles. Par exemple, la correction gamma utilisée dans les images LDR pour compenser la fonction de transfert des écrans est approximativement perceptuellement uniforme jusqu'à un certain niveau de luminance. Cependant, pour la gamme de luminance considérée en imagerie HDR, des encodages perceptuels plus précis peuvent être déterminés. Enfin, nous évoquons dans ce chapitre les principaux formats d'images HDR existants ainsi que les techniques permettant la capture et l'affichage de ces images.

Chapitre 2 : Le second chapitre présente le domaine de la compression d'images et vidéos. Après avoir décrit les principaux concepts de décorrelation, quantification et codage entropique, nous expliquons comment ils sont appliqués dans les encodeurs modernes basés sur un découpage de l'image en blocs. Nous présentons notamment les principaux outils et caractéristiques du récent standard HEVC, à savoir la décomposition en quadtree pour le découpage des blocs, les outils de prédiction intra et inter image pour réduire les redondances spatiales et temporelles, les étapes de transformation et quantification, le codage entropique avec CABAC et enfin les outils de filtrage dans la boucle de codage. La question de la compression du contenu HDR est également abordée par la présentation de trois types d'architecture avec, soit une couche HDR simple, soit une couche LDR accompagnée de méta-données pour la rétro-compatibilité, soit deux couches dans le cas du codage scalable.

Chapitre 3 : Dans ce premier chapitre de contribution, nous abordons la compression HDR rétro-compatible simple couche en mettant l'accent sur les performance débit-distorsion. Le problème consiste alors à déterminer la meilleure manière de réduire la dynamique de l'image HDR afin de pouvoir la compresser par un encodeur standard de faible profondeur de bit (i.e. bitdepth) avec un minimum de distorsion pour un débit donné. Dans nos premiers travaux, nous étudions un schéma de quantification uniforme adaptée au contenu. L'adaptation est effectuée soit par blocs de l'image pris séparément, soit par images, soit par groupe d'images consécutives d'une séquence vidéo. Nous montrons que ces trois variantes de la méthode sont respectivement adaptées au codage à haut, moyen et bas débits. Dans un second temps, en considérant une adaptation au contenu par images, un schéma de quantification non-uniforme est déterminé à partir d'un modèle statistique de la chaîne de codage pour optimiser les performances débit-distorsion des approches rétro-compatibles à simple couches.

Chapitre 4 : Pour traiter le cas où les versions HDR et LDR du contenu sont toutes les deux déjà fournies, nous définissons une méthode scalable permettant d'encoder les deux versions dans des couches distinctes. Le mécanisme clé de ce schéma est appelé prédiction inter-couches et consiste à prédire le contenu de la couche HDR à partir de la couche LDR déjà encodée et décodée. Une prédiction inter-couches précise est essentielle pour décorréler les signaux LDR et HDR et donc réduire significativement le débit. Pour chaque bloc à prédire, une courbe non linéaire de tone mapping inverse est déterminée à l'aide des données LDR et HDR déjà décodées dans le voisinage du bloc. En s'adaptant à chaque bloc de l'image, cette méthode permet de traiter efficacement le cas complexe où la version LDR est générée par un tone mapping local. De plus, grâce à la définition d'un tone mapping inverse non-linéaire dont les paramètres n'ont pas besoin d'être transmis, la méthode obtient des gains de compression significatifs en comparaison avec les autres schémas locaux de prédiction inter-couches.

Chapitre 5 : Bien que la méthode de prédiction inter-couches locale soit bien adaptée lorsqu'aucune information sur le tone mapping à inverser n'est connue à priori, certaines suppositions peuvent être faites sur le lien existant entre l'information chromatique des couches LDR et HDR. Dans la plupart des cas, ce lien peut-être modélisé par une relation mathématique décrivant la manière dont les opérateurs de tone mapping traitent l'information de couleur. Partant de ce modèle et en inversant l'équation, nous déterminons une nouvelle méthode de prédiction inter-couches spécifiquement dédiée aux composantes chromatiques de l'image. La composante achromatique, appelée luma, reste prédite par la méthode locale étudiée précédemment. Nous présentons également une technique de pré-analyse permettant de déterminer automatiquement l'unique paramètre utilisé dans le modèle mathématique. Ce paramètre décrit le lien entre la saturation des couleurs dans les images HDR et LDR. Nous étudions deux versions du schéma de codage basées sur différentes représentations de la couleur pour la couche HDR. Tandis que la première méthode utilise un espace de couleur Y'CbCr, ce qui est la représentation la plus courante pour la compression vidéo, la seconde méthode définit un espace de couleur alternatif découlant directement des principes de colorimétrie décrits au premier chapitre. Outre les avantages que cela confère pour la compression de contenu HDR, les propriétés de cet espace de couleur permettent une prédiction inter-couches plus simple et plus précise des composantes chromatiques que dans le cas de l'espace Y'CbCr.

Introduction

Context

Image and video compression is an essential component of digital television, video streaming, exchange of photos on the internet, and many other applications. While the storage capacities and transmission bandwith allowed by new technologies improve year after year, the amount of image and video data in transit over the internet is growing at such a pace that not only is compression still needed, but increasingly efficient compression methods are required. In a recent white-paper [1], Cisco forecasts that video content, which already accounted for 64% of the global internet traffic in 2014, will reach an impressive 80% by 2019. Although the recent MPEG HEVC norm already divided the bitrates by two in comparison to its predecessor MPEG AVC, such figures clearly suggests that the standardization effort in video compression should be pursued further.

Additionally, the improvements on capture and display technologies pushes the demand for higher quality content and new video formats. The recent definition of a new standard format for Ultra High Definition (UHD) television (or 4K) is a striking example. But aside from increasing the number of pixels, there exist several other ways of improving the television or digital cinema experience. For instance, 3D display technology has raised a lot of interest over the last few years. In particular, the resounding success of the movie "Avatar" released in 3D in 2009 has stimulated the demand for 3D content. Another step towards realism can be reached by high frame rate (HFR) capture and display. Finally, the quality of each individual pixel can also be enhanced by using wide color gamut (WCG) and high dynamic range (HDR) technology. While wide color gamut imaging extends the color range of traditional image representations, high dynamic range imaging concerns the range of quantities of light, namely the luminance range, that each pixel can take. In other words, HDR images can contain very bright regions while keeping visible details in deep dark areas. By enabling a direct visualisation of such images, the recent advances in display technologies have been a catalyst in the broad interest shown towards production and distribution of HDR video content. All the actors across the video chain are now involved, from motion picture studios to television manufacturers, including television broadcasters and standardization bodies.

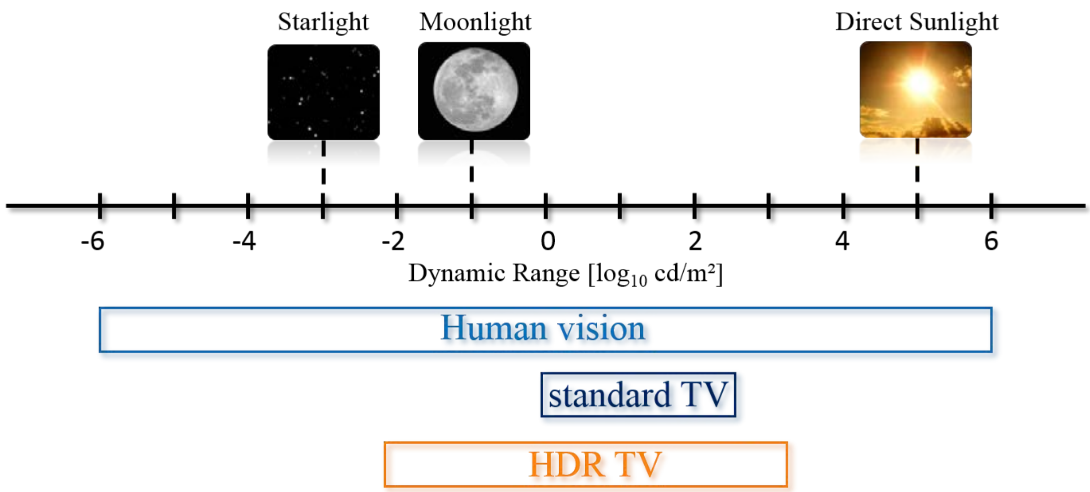


Figure 2: Dynamic ranges of human vision and displays.

Motivations

The dynamic range of an image is characterized by the ratio between the maximum and minimum luminance, measured in candela per square meter (cd/m^2). The human eye is capable of perceiving luminances ranging from 10^{-6} to 10^6 cd/m^2, giving a ratio of $10^{12}:1$. However, conventional displays only reach a contrast ratio of about 300:1. The typical dynamic ranges of human vision and television are illustrated in figure 2. Image processing algorithms called tone mapping operators (TMO) are therefore required for reducing the dynamic of the image to the displayable range allowed by standard television systems. Although the generation of High Dynamic Range images or videos has been possible for a long time with computer graphics, or even in real-world capture using bracketing techniques, the scope of application of HDR imagery was reduced by the display limitations. The arrival of HDR displays with luminances spanning from about 10^{-2} to 10^3 cd/m^2 or higher now makes it possible to visualize HDR images directly (i.e. without tone mapping), thereby enabling a better viewing experience. With these new possibilities comes new challenges regarding the encoding and distribution of HDR content on a large scale:

- A first problem to solve for encoding HDR images is the definition of new standard representations of light and color that match the human capabilities. Traditional images, referred to as Low Dynamic Range (LDR) images generally represent the color information using 8-bit integer data per RGB color component, in a way that is adapted to typical LDR displays. Conversely, existing HDR image formats are not referred to display capabilities since their development is prior to HDR display technologies. Instead, physical luminance values with high precision floating point data are stored directly. The advantage is that all the information captured is kept. This can be very helpful during the editing steps such as color grading, but it also requires the storage of a very large amount of information that may not even be visible to the human eye. Furthermore, standard compression algorithms are designed for taking integer input data. New representations of colors must then be found, including the definition of a colorspace adapted to human perception and the determination of the bitdepth that is sufficient to encode image data without visible loss.

- The next problem is the compatibility between LDR and HDR equipment. While HDR systems will be able to decode and display high bitdepth video data, this format will not be supported by conventional equipment. The obvious solution would be to broadcast the HDR version and a tone mapped version of the content on separate channels. This solution, called simulcast, was used for instance for the transition between High Definition (HD) and Standard Definition (SD) television. However, the cost of simulcast is high because similar information is transmitted twice. The problem becomes even worse with the multiplication of the new data formats (e.g. HD, UHD, HDR) since several types of compatibility are required in the same time. Therefore, new compression techniques must be found to address backward compatibility at a lower cost.

Contributions

Although the question of the representation of HDR color data is taken into account, the work presented in this thesis is mainly directed towards the second aspect of HDR compression discussed previously, namely backward compatibility.

The first part of the work consists in using existing compression standards without modification in order to encode both HDR and LDR data in a single layer. Such backward compatible schemes first perform a tone mapping of the HDR image and compress the resulting LDR image along with metadata indicating how to perform the inverse TMO. Legacy equipment merely decode the LDR image while HDR capable decoders additionally read the metadata to retrieve the original dynamic of the image. The main drawback of these methods is the loss of information inherent to tone mapping which reduces the bitdepth, and thus the image quality. In order to preserve most of the details into the LDR version despite the bitdepth reduction, some tone mapping operators enhance the contrasts locally instead of simply applying the same tone curve to each pixel of the image. Inverting those local TMOs, however, is generally more difficult and requires more metadata. From the observation that the choice of the TMO influences both the bitrate and the quality of the reconstructed HDR image, we study several ways of optimizing this tone mapping step for keeping a maximal HDR quality at a given bitrate.

Despite the satisfactory results obtained with single layer approaches in terms of compression efficiency, such solutions are not in the interest of content producers since the LDR version is automatically generated by the encoder and is therefore not defined by an artistic process. For that reason, a second part of the work is dedicated to the development of scalable encoding schemes with a LDR and a HDR layer. In this approach, the LDR version is encoded first in a standard format and the HDR version is encoded subsequently with a scalable method that makes use of the decoded LDR layer to reduce the redundancy and thus the amount of information required to encode the HDR layer. The methods developed in this thesis are designed for remaining efficient even in the challenging case where the LDR layer was generated with a local tone mapping operator.

In a last contribution, we study how to further improve the scalable encoding schemes by taking into account the way chromatic information is usually handled by tone mapping operators. This work was also an opportunity to explore an alternative color representation, better suited to our scalable compression method.

Structure of the thesis

The thesis manuscript is organized in two parts. The first part contains two chapters presenting the state of the art in the fields of image and video compression and HDR imaging. The second part describes our contributions in the HDR compression field:

Chapter 1 : This chapter first gives an introduction to the fundamental concepts of the human perception of light in order to understand how light and colors are measured in photometry and colorimetry. Higher level aspects of the human visual system are also discussed by presenting perceptually uniform representations of colors. The typical color encoding of LDR images, although linked to the principles of colorimetry and perceptual uniformity, has been designed especially to be in conformity with conventional low dynamic range display technology. For instance, the gamma correction used in the traditional color encoding for compensating the display's transfer function is approximately perceptually uniform up to a certain luminance level. For the luminance range considered in High Dynamic Range imaging, more accurate perceptual encodings are discussed. We finally describe the main HDR image formats and the techniques used to capture and display such images.

Chapter 2 : The second chapter presents the field of image and video compression. After describing the general concepts of decorrelation, quantization and entropy encoding, we explain how they are applied to block-based video codecs. In particular, we provide an overview of the main tools in the new HEVC standard. The question of the compression of HDR content is also raised and the main types of architectures for HDR coding systems are outlined in the last section of the chapter.

Chapter 3 : In this first contribution chapter, we tackle the problem of backward compatibility with single layer methods from the angle of the rate-distortion performance. This problem is then to find the best way to reduce the bitdepth of an HDR image so that it can be compressed by a conventional low bitdepth encoder with minimal distortion at a given bitrate. A first work studies the adaptation of a simple uniform quantization technique to the content of either individual blocks, frames or several consecutive frames of a sequence. We show that those variants of the method are adapted to respectively high, medium or low bitrate encoding. Secondly, assuming a frame-wise adaptation, a more advanced non-uniform quantization scheme is derived from a statistical model for optimizing the rate-distortion performance of the single layer backward compatible scheme.

Chapter 4 : Given the case where the HDR and the LDR version of the content are now both provided by the producer, we define new scalable methods that encode the two versions in separate layers. The key mechanism here is the Inter-Layer Prediction (ILP) which consists in predicting the content of the HDR layer by using the decoded LDR layer. An accurate inter-layer prediction is essential for a good decorrelation of the HDR and LDR signals, and thus a significant reduction of the bitrate. For each block to predict, a non-linear inverse tone mapping curve is determined from the already decoded neighboring LDR and HDR data. This block-wise adaptability makes the method effective in the difficult case where the LDR version was generated with a local TMO. Furthermore, thanks to the definition of a non-linear inverse tone mapping that does not require the transmission of side information, this method significantly outperforms the existing local inter-layer prediction schemes.

Chapter 5 : Although the local ILP method is particularly adapted when no prior information is known regarding the tone mapping used for generating the LDR content, the chromatic information in HDR and LDR happen to be linked, in most cases, by the same mathematical relation. By using this relation as a model describing the way TMOs treat color information, and by inverting the equation, we derive a new inter-layer prediction methods specifically for the chromatic components. The achromatic component, called luma, remains predicted with the local method. We additionally present a pre-analysis method for the automatic determination of a parameter involved in the mathematical model. This parameter describes the link between the saturation of the colors in the LDR and the HDR images. In this chapter, we study two version of the encoding scheme based on different color representations for the HDR layer. While the first method uses a Y'CbCr colorspace, which is the most common representation in the field of compression, the second method uses an alternative colorspace derived from the principles of colorimetry described in the first chapter. Aside from the advantages it offers for the compression of HDR data, this colorspace has properties that enable a more straightforward and accurate inter-layer prediction of the chromatic components than the Y'CbCr space.

Part I

Context and state of the art

Chapter 1

Background in High Dynamic Range imaging

Traditional image formats were designed by taking into account technological constraints on the display and capture devices which could only capture or reproduce a limited range of light intensities. These formats usually encode colors with three 8 bit integer values, each of them representing the intensity for a red, green or blue component. That representation is in accordance with the way cameras and displays deal with colors by either capturing or emitting red, green and blue light separately. This 8-bit per component RGB encoding, however, is no longer adapted when it comes to representing the high dynamic range of luminance that the human eye can perceive.

High Dynamic Range (HDR) imaging is the term given to capture, storage, manipulation, transmission, and display of images that more accurately represent the wide range of real-world lighting levels [2]. Different representations of colors, more directly related to physical measure of light must then be used in this field. Furthermore, understanding how the human eye perceives light and quantifying this perception is essential for allowing an efficient compression of HDR images.

This chapter returns to the fundamental concepts of the human perception of light in order to explain how the digital information in an image is related to physical light in a real scene and how High Dynamic Range imaging techniques may be used to capture, store and display a larger range of light intensities than traditional imaging.

1.1 Principles of photometry and colorimetry

This section gives a brief introduction to the concepts of photometry and colorimetry underlying the definition of the colorspaces used in both traditional and High Dynamic Range imaging.

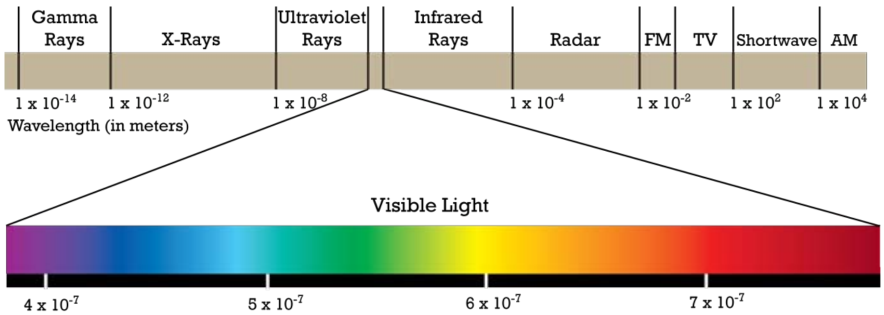


Figure 1.1: Spectrum of the visible light.

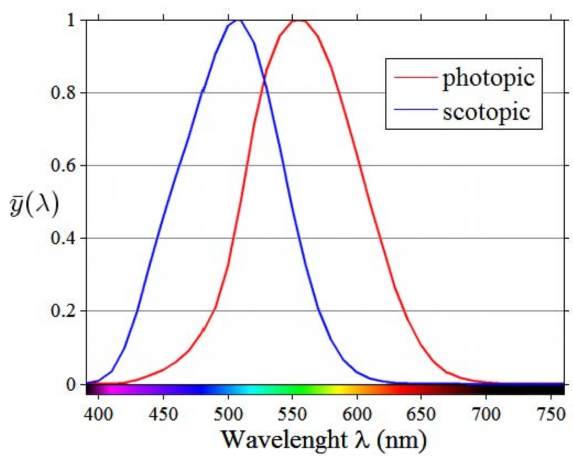


Figure 1.2: CIE standard photopic (red) and scotopic (blue) luminosity functions.

1.1.1 Light

The visible light is an electromagnetic radiation within a limited portion of the electromagnetic spectrum. As represented in figure 1.1, only the wavelengths comprised between approximately 380 and 780 nanometers can be perceived by the human eye.

In photometry, the amount of light is measured by taking into account the sensitivity of the human eye to each wavelength of the visible spectrum. The luminosity function, $\bar{y}(\lambda)$, defined by the International Commission on Illumination (CIE from the french name, Commission Internationale de l'Eclairage) is used to weight the contribution of the radiant energy at each wavelength to the overall perceived amount of light. Since the eye behaves differently in photopic (i.e. high light) or scotopic (i.e. low light) conditions, two variants of the luminosity function have been defined. They are both shown in figure 1.2.

1.1.2 Luminance

The fundamental physical measure to consider in HDR imaging is the luminance. It is a photometric measure expressed in candelas per square meter (cd/m^2) which describes the amount of light that arrives at the eye (or the camera sensor) from a given direction. In the classical definition of luminance, the luminosity function of photopic conditions is generally used for measuring the amount of light.

As such, luminance does not directly measure the intensity of a specific light source since it combines, at each point of a scene, all the direct and indirect light coming from that point to the observer. Therefore, given an image displayed on a screen and an observer, the luminance perceived from each pixel should depend on the lighting conditions of the room and the position of the observer. However, for ideal visualization conditions, it can be assumed that there is no reflection on the screen originating from external light sources. In these conditions, the light intensity that should be displayed for reproducing a given luminance level on the observer's eye can be determined. As a result, an image can be characterized by a luminance value at each pixel.

1.1.3 Chromaticity

Luminance alone only gives an achromatic information. Other quantities are necessary to describe color. In 1931, the International Commission on Illumination defined the CIE XYZ colorspace [3] which encompasses all the colors that can be perceived by the human eye with three values. For a given spectral power distribution $J(\lambda)$, the tristimulus CIE X, Y and Z values are obtained by:

$$X = \int_{380}^{780} J(\lambda) \cdot \bar{x}(\lambda) \, \mathrm{d}\lambda \tag{1.1}$$

$$Y = \int_{380}^{780} J(\lambda) \cdot \bar{y}(\lambda) \, \mathrm{d}\lambda \tag{1.2}$$

$$Z = \int_{380}^{780} J(\lambda) \cdot \bar{z}(\lambda) \, \mathrm{d}\lambda \tag{1.3}$$

where $\bar{x}$, $\bar{y}$ and $\bar{z}$ are the standard color matching functions represented in figure 1.3

Note that the colormatching function $\bar{y}$ is the photopic luminosity function. The value Y is thus equal to the luminance. Although the three values X, Y and Z contain less information than the spectral power distribution $J(\lambda)$, they are sufficient to represent all the visible colors. This is due to the way the eye perceives color. Humans with normal vision have three types of cone cells in their retina designated by the letters L, M and S, each type mostly responding to respectively long, medium and short wavelength. The stimuli of L, M and S cones are combined in the brain to form the perception of color. As a consequence, two light sources of different spectral distributions may appear to be the same color if they induce the same response for each type of cone. This phenomenon is called metamerism. In the definition of the CIE XYZ, the $\bar{x}$, $\bar{y}$ and $\bar{z}$ colormatching functions are analogous, although different to the L, M and S cones responses.

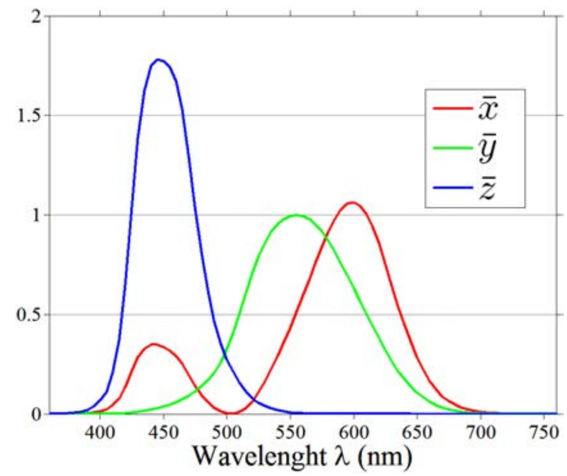


Figure 1.3: CIE standard color matching functions $\bar{x}$, $\bar{y}$ and $\bar{z}$.

A convenient way of representing color is to separate the luminance, indicating the global amount of light it contains, from the chromaticity which gives information about the hue and the saturation. For that purpose, the xyY colorspace has been derived where Y is the luminance, and x and y are the coordinates of the chromaticity diagram defined by:

$$x = \frac{X}{X+Y+Z} \tag{1.4}$$

$$y = \frac{Y}{X+Y+Z} \tag{1.5}$$

Figure 1.4 represents the CIE XYZ colorspace and the xy chromaticity diagram. Note that only the points within the colored shape, called the gamut of human vision, correspond to actual colors. The curve at the boundary of the gamut is called the spectral locus and corresponds to monochromatic light (i.e. light with a single wavelength in its spectral power distribution). The colors represented on this line are very vivid (i.e. with a high saturation) while the colors closer to the center of the gamut tend towards gray.

1.2 Perceptually Uniform representations

As stated in [4], a system is perceptually uniform if a small perturbation to a component value is approximately equally perceptible across the range of that value. In a perceptually uniform colorspace, the euclidean distance is good a measure of the perceptual difference between colors. The notion of perceptual uniformity is important in the field of compression. If a colorspace which is far from being perceptually uniform is used in a lossy compression system, too much bitrate will be allocated for encoding some color values with unnecessary precision, while other colors might appear distorted after decoding.

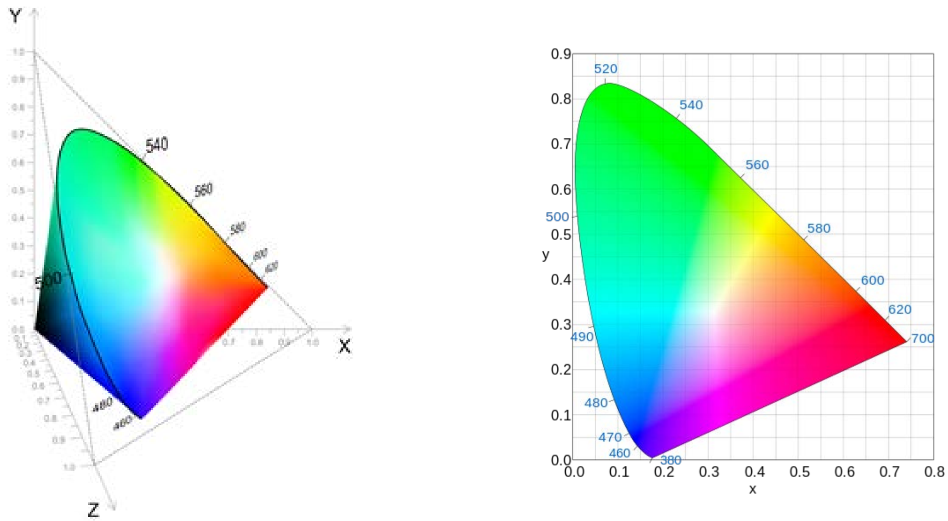


Figure 1.4: Left : 3D representation of the CIE 1931 XYZ colorspace. Right : CIE 1931 xy chromaticity diagram.

1.2.1 Perception of luminance

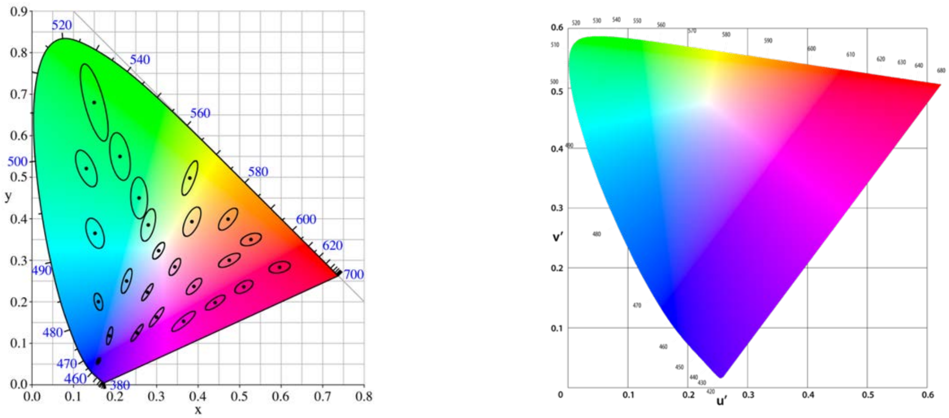
The subjective sensation of brightness is not linearly linked to the luminance. The earliest model describing the human response to a physical stimulus dates back to the experiments of Weber in 1834. He stated that the just-noticeable difference (JND) between two stimuli is proportional to the magnitude of the stimuli, which can be noted

$$\frac{\Delta I}{I} = k, \tag{1.6}$$

where ΔI is the JND (or perception threshold) for a stimuli of intensity I, and k is a constant. In the case I represents luminance, the value $\frac{\Delta I}{I}$ is called the just noticeable contrast.

Fetchner [5] later derived from this observation that the perceived intensity of a stimulus is proportional to the logarithm of its physical magnitude. This model, known as Weber-Fetchner law is very general and may apply to any physical stimulus. The validity of this law has been questioned in the case of the perception of luminance. For instance, according to Stevens law [6], the perceived brightness of a light is proportional the cube root of the luminance. However, this power law was fitted from experimental data for relatively low luminance levels and does not generalize well to higher luminance.

More recently, several experiments were conducted to find more accurate models [7, 8, 9]. These experiments were performed by placing an observer in front of a screen displaying a gray patch onto a uniform gray background of different intensity. The just noticeable difference was found by varying the luminance of the patch until it becomes indistinguishable from the background. The experiment was repeated for different background luminance values in order to find a Threshold Versus Intensity (TVI) function representing the JND as a function of luminance. The results show that equation (1.6),



(a) CIE 1931 xy diagram

(b) CIE 1976 u'v' diagram

Figure 1.5: Left : Macadam ellipses drawn on the xy chromaticity diagram. Right : CIE 1976 u'v' uniform chromaticity scale diagram (UCS).

and thus the Weber-Fetchner law, is only true for sufficiently high luminance values for which the photoreceptors (i.e. cone and rod cells in the retina) are fully responsive.

In the latter experiments, a patch of fixed intensity was used. More elaborated models have also been developed by replacing it with a sinusoidal pattern so as to evaluate the effect of spatial frequency on the visibility threshold [10, 11]. The resulting model is called contrast sensitivity function (CSF), where the contrast sensitivity is defined as the inverse of the just noticeable contrast and is given as a function of the luminance and the spacial frequency. Note that given a contrast sensitivity function, a TVI function can be derived from the peak contrast sensitivity at each luminance level:

$$TVI(I) = \frac{I}{\max_\rho(CSF(\rho, I))} \tag{1.7}$$

where ρ is the spatial frequency. The TVI function may then be used to determine a perceptually uniform encoding of the luminance. A similar method was used for example in [12] for deriving a perceptually uniform curve for HDR luminance data based on the CSF model of Daly [13].

1.2.2 CIE Uniform colorspaces

Although the XYZ and xyY colorspaces were defined by taking elementary properties of the human visual system into account, they are not perceptually uniform colorspaces. Aside from the non-linear perception of luminance, the CIE xy diagram does not give a perceptually uniform representation of the chromaticity. This is illustrated in figure 1.5(a) where Macadam ellipses [14] are drawn on the xy chromaticity diagram. Each ellipse contains all the colors that are indistinguishable from the color at the center of the ellipse. A large variation of the size and orientation of the ellipses can be observed depending on their position in the diagram. In particular, the ellipses are much bigger in the green areas than in the blue ones. Therefore, large portions of the diagram are used for representing green colors which are very close visually.

In 1976, the CIE defined a more perceptually uniform chromaticity diagram called the uniform chromaticity scale (UCS) [15]. Its coordinates u' and v' are derived from the XYZ values as follows

$$u' = \frac{4 \cdot X}{X + 15 \cdot Y + 3 \cdot Z} \tag{1.8}$$

$$v' = \frac{9 \cdot Y}{X + 15 \cdot Y + 3 \cdot Z} \tag{1.9}$$

An illustration of the UCS diagram is given in figure 1.5(b).

It must be noted that the u'v' diagram is not a complete colorspace since it only represents chromaticity without consideration for the luminance. However, the accuracy of the perception of chromaticity is dependant on the luminance level. For instance, in low light conditions the human eye has very poor color vision while luminance contrasts can still be perceived. As a consequence, luminance and chromaticity must be considered jointly for determining a perceptually uniform colorspace. The CIE $L^*a^*b^*$ (CIELAB) [16] and CIE $L^*u^*v^*$ (CIELUV) [15] colorspaces have been designed in that purpose.

The CIE additionally defined a color difference formulas in order to quantify the perceptual difference between two colors. Several successive versions have been defined. The first version noted ΔE^*_{ab} is simply the euclidean distance in the CIELAB colorspace. Since CIELAB is an approximately perceptually uniform colorspace, the euclidean distance should be a good perceptual measure of color difference. However, further experiments have shown that the CIELAB space was not as perceptually uniform as intended, especially in the saturated and in the blue regions. The current version of the formula was defined in 2000 and is noted ΔE^*_{00}. It is still based on the original L^*, a^* and b^* components but introduces several constants and transformations for solving the perceptual uniformity issues. Because of the complex expression of the ΔE^*_{00}, there is no analytic formulation of a corresponding perceptually uniform colorspace for which the euclidean distance is equal to the ΔE^*_{00}. However, there exist numerical methods for approximating such a colorspace [17].

One limitation of the CIELAB and CIELUV representations is that they are based on Steven's law [6] which models the relationship between luminance and perceived brightness as a cube root. In particular, the L^* component, called lightness, is approximately proportional to the cube root of the luminance. We have seen that Steven's law was only valid for low luminance, and therefore, the CIELAB and CIELUV colorspaces and the color difference formulas are not generalizable to HDR images.

1.3 Color encoding of digital images

In previous sections, we have seen several ways of representing colors based on the principles of colorimetry and the properties of the human visual system. However, in actual image and video formats, the standard colorspaces have been defined in accordance with technological limitations of the capture and display devices.

1.3.1 RGB color spaces

Most capture and display devices use a color representation based on red, green and blue color primaries defining a RGB colorspace. This is the most common way of defining the colors of pixels in traditional LDR images. However the definition of the red, green and blue primary colors may not be the same for every device. The RGB representation is thus called device-dependent.

In practice, the RGB primaries do not need to be defined by their complete spectral power distribution. Instead, four chromaticity points in the CIE xy diagram (or equivalently in the u'v' diagram) are sufficient. Three of those points correspond to the chromaticity of the RGB primaries and the last one is the chromaticity of the white point obtained when mixing equal intensities of red green and blue (i.e. when R=G=B). The most commonly used white point is the CIE standard illuminant D65 [18] which represents average daylight. Its xy coordinates are x_{D65}=0.31271 and y_{D65}=0.32902. Given the four chromaticity points $(x_C, y_C)_{C=R,G,B,W}$, linear transformations can be derived for the conversions between the reference CIE XYZ and the RGB colorspace as stated in [19].

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = M \times \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1.10}$$

where the matrix M is defined by

$$M = \begin{bmatrix} x_R & x_G & x_B \\ y_R & y_G & y_B \\ 1 - x_R - Y_R & 1 - x_G - Y_G & 1 - x_B - Y_B \end{bmatrix} \begin{bmatrix} C_R & 0 & 0 \\ 0 & C_G & 0 \\ 0 & 0 & C_B \end{bmatrix} = [P] \cdot [C] \tag{1.11}$$

And where $\begin{bmatrix} C_R \\ C_G \\ C_B \end{bmatrix} = [P]^{-1} \cdot \begin{bmatrix} x_W/y_W \\ 1 \\ z_W/y_W \end{bmatrix}$.

For instance, in the sRGB colorspace [20], one of the main standards for display technologies, the matrix M of conversion from RGB to XYZ is given by:

$$M = \begin{bmatrix} 0.4124 & 0.3576 & 0.1805 \\ 0.2126 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9505 \end{bmatrix} \tag{1.12}$$

Note that the luminance Y is a weighted average of the RGB values with a strong weight value of 0.7152 for the green component. This is explained by the fact that the luminosity function $\bar{y}$ shown in figures 1.2 and 1.3 peaks at a wavelength of 555 nm which corresponds to green light.

Assuming that only positive RGB values are allowed, only the colors within the triangle formed by the red, green and blue points in the xy diagram can be represented. The triangle for the sRGB colorspace, called the sRGB color gamut is illustrated in figure 1.6.

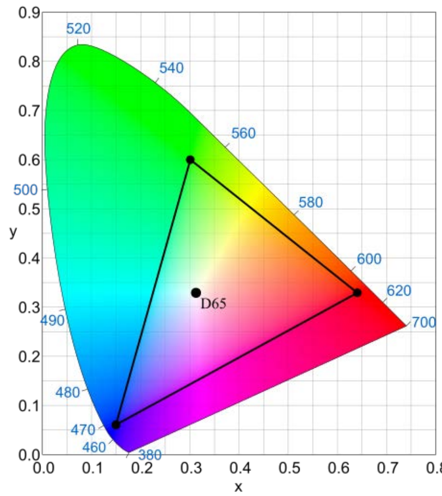


Figure 1.6: Gamut of the sRGB colorspace. The white point corresponds to the CIE standard illuminant D65 [18].

1.3.2 Gamma correction

As shown by equation (1.10) the RGB components can be computed as a linear transformation of the CIE XYZ components, and inversely the luminance Y is a linear combination of the RGB values. However, we have seen in subsection 1.2.1 that the human perception of luminance is not linear. In order to define RGB colorspaces with better perceptual uniformity, a non-linear transformation called gamma correction is used in traditional imaging and applied independently to each of the RGB components.

Originally, the gamma correction was not only designed to account for the non-linear relationship between luminance and perceived brightness, but also because of a technological constraint imposed by the cathode ray tube (CRT) monitors. The cathode ray tubes convert an input voltage V into a luminance L_v in a non linear way which is well described by a power law:

$$L_v = k \cdot V^{\gamma} \tag{1.13}$$

Although the more recent technology used in LCD displays have very different transfer functions, backward compatibility is kept by incorporating circuitry that mimics the transfer function of a CRT device.

In order to compensate for this transfer function, gamma correction is applied to the RGB components as shown in equation (1.14). Here, it is assumed that the RGB values are normalized in the $[0, 1]$ range (i.e. clipped above a luminance threshold and divided by this maximum value):

$$R' = R^{\frac{1}{\gamma}}, \quad G' = G^{\frac{1}{\gamma}}, \quad B' = B^{\frac{1}{\gamma}} \tag{1.14}$$

Gamma correction is included in the definition of standard RGB colorspaces. For instance, the sRGB colorspace is defined by its primaries as seen in subsection 1.3.1, but also by a non-linear function approximately equal to the gamma correction function of equation (1.14) with $\gamma = 2.2$.

Most LDR images are represented by their gamma corrected R'G'B' components quantized to 8 bits integers. Once displayed, a pixel with value 255 on each component will correspond to the maximum luminance of the display device. For the luminance range considered in traditional LDR display technology, the gamma correction turns out to give a fairly good approximation of the human perceptual response to luminance. However, it does not generalize well for a wider luminance range. In the field of HDR imaging other non-linearities have been determined. They are often referred to as Opto-Electrical transfer functions (OETF) for converting luminance into a perceptually uniform representation. The inverse functions are called Electro-Optical transfer functions (EOTF). The SMPTE recently proposed a new EOTF for HDR called PQ-EOTF [21, 22] based on Barten's model of the Constrast Sensitivity Function [10]. Several other candidates have been proposed in MPEG-XYZ group [23] and in ITU-R (ITU-R SG6/W6-C group) [24]. Furthermore, bitdepths higher than 8 are necessary in order to quantize the larger range of luminance values without perceived loss.

1.3.3 Y'CbCr encoding

In the field of image and video coding, the R'G'B' components are generally converted to Y'CbCr colorspace before compression. This operation separates the signal into a luma component Y' that approximates the luminance after gamma correction and two chroma components Cb and Cr containing most of the chromatic information. Assuming R'G'B' are already quantized to n bits, the general conversion formula can be expressed as follows:

$$Y' = \alpha_0 \cdot R' + \alpha_1 \cdot G' + \alpha_2 \cdot B' \tag{1.15}$$

$$Cb = \frac{Y' - B'}{2 \cdot (1 - \alpha_2)} + 2^{n-1} \tag{1.16}$$

$$Cr = \frac{Y' - R'}{2 \cdot (1 - \alpha_0)} + 2^{n-1} \tag{1.17}$$

where α_0, α_1 and α_2 are the same coefficient as in the conversion from RGB without gamma correction to luminance Y (i.e. second row of the matrix M in subsection 1.3.1). In the case of the sRGB colorspace, $\alpha_0 = 0.2126$, $\alpha_1 = 0.7152$ and $\alpha_2 = 0.0722$. The Y'CbCr values are then rounded to keep integers with a bitdepth n.

The reason of this transformation is twofold. Natural images contains a lot of redundancy between the R', G' and B' components which are all strongly correlated with the luma. The conversion to Y'CbCr removes most of the inter component correlation, thereby reducing the amount of information to encode. The second reason is perceptual and relates to the lower spatial sensitivity of the human visual system to chromatic differences than luminance differences. Splitting the signal into luma and chroma enables the spatial downsampling of the chroma components of the image. Chroma downsampling is heavily used in video coding. Several types of chroma formats exist and the most commons are:

- 4:2:0 format : both horizontal and vertical downsampling by a factor of 2 are applied to the chroma.
- 4:2:2 format : similar to 4:2:0 but only horizontal downsampling is applied.
- 4:4:4 format : No downsampling is applied.
- 4:0:0 format : Only the luma is encoded. Chromatic information is lost.

In the literature, the notation Y'CbCr is often simplified into YCbCr or YUV. However, we will keep the term Y'CbCr in order to avoid confusion between the luma Y' and the luminance Y and between the CbCr chroma components and the u*v* components of the CIE LUV colorspace or the u'v' chromaticity coordinates.

1.4 HDR imaging techniques

This section gives a brief overview of High dynamic range imaging techniques necessary for the capture and display of HDR images and presents the main HDR image file formats.

1.4.1 Capture of HDR images

High Dynamic Range images may be either rendered using 3D computer graphics techniques or captured from real world scenes. In the latter case, computational imaging techniques must be used in order to overcome the limitations of conventional camera equipment.

In a digital camera, the sensors can only capture a limited amount of light below a certain threshold defined by the exposure Value (eV) setting of the camera. For a luminance value above that threshold the sensors saturate and the pixels will appear white. And on the other side, for too low luminance values, the sensors are not accurate enough and noise will become predominant. In HDR imaging, a higher range of luminance can be captured while keeping a reasonable Signal to Noise Ratio (SNR) in dark areas by taking several pictures of the same scene with different exposures. An HDR image can then be computed by combining the LDR pictures according to their exposure Values. This process is called bracketing and can be performed in several ways:

- The easiest method called temporal bracketing consists in taking the LDR pictures one after the other [25, 26]. However, this is not adapted for videos and in the case of a fast moving scene, additional computations are required to remove the ghost artifacts that may appear in the final image[27].
- In spatial bracketing, an optical mask is superimposed onto a conventional image sensor array. The mask has a pattern with spatially varying transmittance, thereby giving different exposures to the pixels [28, 29]. This method can capture simultaneously different exposures but results in a loss of spatial resolution.

- Optical systems can be designed with multiple sensor arrays and a beam splitter which divides the light unequally between each sensor array. Such cameras enable the simultaneous capture of several LDR images with different exposures without loss of spatial resolution [30].

- Similarly, multiple camera rigs with different neutral gray filters can be used to produce HDR images.

These techniques may be seen as transitional solutions. It is highly probable that in a near future, the improvements of sensor intrinsic capabilities will make it possible to directly capture the wide range of visible light and color gamut without requiring bracketing techniques.

1.4.2 HDR image formats

There exists a variety of High Dynamic Range image formats using different representations of the colors. This subsection presents some of the most popular formats in the HDR imaging community. They often come with elementary compression algorithms which are fast to compute and generally lossless or near lossless. For that reason, these formats are well adapted to the exchange of HDR images between programs such as image editing programs or computer graphics software. But they are not intended for applications such as video broadcasting or streaming which require lower bitrates and tolerate some visible loss.

1.4.2.1 RGBE format

The RGBE format [31] is one of the first HDR image formats. It was developed in 1991 by Gregory Ward to store the HDR images generated by the Radiance rendering software [32]. It is now very well supported by most software that deal with High Dynamic Range images. The RGBE format uses a floating point representation of the linear RGB tri-stimulus values (i.e. without gamma correction), where 8 bits are used to store each of the mantissas R_m, G_m and B_m of the RGB components, and another 8 bit integer encodes a shared exponent E. From the mantissas and exponent, the RGB values are given by

$$R = \frac{R_m}{256} \cdot 2^{E-128}, \quad G = \frac{G_m}{256} \cdot 2^{E-128}, \quad B = \frac{B_m}{256} \cdot 2^{E-128} \tag{1.18}$$

In addition, the RGBE format uses run-length encoding as compression algorithm. In total, a very high dynamic range of 76 orders of magnitude (i.e. contrast ratio of $10^{76} : 1$) is covered by this representation using only 32 bits per pixel, which is more than the eye can perceive. However, the RGBE format fails to represent highly saturated colors (e.g. very high R value and low B and G values) and colors outside the gamut of the RGB colorspace. These problems are fixed in a variant of the format referred to as the XYZE format by using the CIE XYZ components instead of RGB.

1.4.2.2 OpenEXR format

OpenEXR [33] is another popular format developed by Industrial Light & Magic (ILM) for special effects, rendering and compositing. The source code was made freely available since 2002. In its basic form, OpenEXR represents the color as linear RGB values encoded in the 16-bit floating point format (i.e. half float) [34], for a total of 48 bits per pixel. The so-called half float data format was originally developed by Nvidia for its graphics cards and contains 1 bit for encoding the sign s, 5 bits for the exponent e, and 10 bits for the mantissa m. A half float value f is then given by:

$$f = \begin{cases} (-1)^s \cdot 2^{e-25} \cdot (1024 + m), & \text{if } 1 \leq e \leq 30 \\ (-1)^s \cdot 2^{-24} \cdot m, & \text{if } e = 0 \end{cases} \tag{1.19}$$

The exponent value 31 is reserved for encoding special values such as Not A Number (NaN) and the infinity. Unlike the RGBE format, negative values can be encoded, which may be useful for representing colors outside the RGB gamut. The dynamic range covered by the half float representation is lower than for the RGBE format but it is sufficient for HDR imaging applications. Furthermore, it performs a finer quantization of the RGB values which makes it perceptually lossless, even for highly saturated colors.

In addition, OpenEXR implements several compression methods including the lossless PIZ algorithm which is based on a wavelet transform followed by Huffman entropy encoding [35]. Despite the 48 bits encoding of color, the HDR images stored in the OpenEXR formats are generally smaller than in the RGBE format thanks to the PIZ compression. Note that in recent versions, a lossy compression codec was also added for allowing the user to control the balance between quality and compression rate.

1.4.2.3 TIFF format and LogLuv representation

HDR images can also be stored in the TIFF format [36]. TIFF is a very flexible format with many extensions that can represent colors in different colorspaces with several possible bitdepths. In particular, a TIFF file may contain 32 bit floating point RGB pixel data, giving 96 bits per pixel. This encoding covers nearly 79 orders of magnitude with very small quantization steps, but it requires a lot of memory and it is difficult to compress.

Alternatively, the LogLuv representation [37, 38] was developed by Gregory Ward and included in the LibTIFF library, which is the most popular implementation of the TIFF specifications. The LogLuv format uses a logarithmic encoding of the luminance in accordance with the Weber-Fetchner law of perceptual uniformity. Additionally, the CIE u'v' coordinates of the Uniform Chromaticity Scale diagram are used for representing the chromaticity. Two versions of the LogLuv encoding have been developed using either 24 or 32 bits per pixel. In the 32-bits version, the logarithm of the luminance Y is encoded as a 15 bit integer L_{15} while 8-bits values u_8 and v_8 are used for the u' and

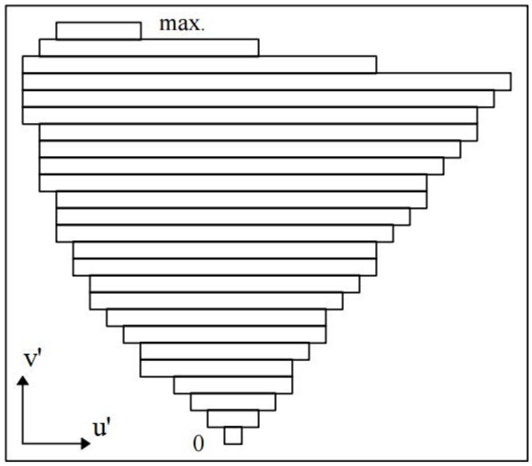


Figure 1.7: u'v' look-up table encoding. The figure was taken from [37]

v' coordinates: The remaining bit is a sign bit:

$$L_{15} = \lfloor 256(log_2 Y + 64) \rfloor \quad (1.20)$$
$$u_8 = \lfloor 410\, u' \rfloor \quad (1.21)$$
$$v_8 = \lfloor 410\, v' \rfloor \quad (1.22)$$

The coefficient 410 was chosen so that u_8 and v_8 remain in the range [0,255], knowing that the entire gamut is described by values of u' and v' between 0 and 0.62. Run length encoding is finally applied. This encoding covers a luminance values between $5.5 \times 10^{-20} cd/m^2$ to $1.9 \times 10^{19} cd/m^2$, which corresponds to 38 orders of magnitude.

In the 24-bits version, the logarithm of the luminance uses only 10 bits and the u'v' chromaticity coordinates are encoded as a single 14-bit index in a 2D look-up table (LUT) representing positions in the u'v' diagram as illustrated in figure 1.7. The index zero in the LUT corresponds to the point of the visible gamut with the smallest v' value, and the next table entries are assigned left to right along the horizontal scan lines and from bottom to top. Using this encoding only 14 bits can be used to cover the full gamut. However, in the 24-bit LogLuv both the u'v' step size and the luminance step size are slightly larger than the visible threshold. Furthermore, the 10-bit log luminance encoding only covers 4.8 orders which is less than the human eye's capabilities.

1.4.3 Display of HDR images

Two solutions can be considered to display HDR images: either using directly displays with a higher dynamic range than conventional devices or reducing the dynamic range of the image so that it fits the capabilities of typical displays.

The peak luminance of conventional Liquid Crystal Display (LCD) devices is usually in the order of 300 cd/m^2, for a contrast ratio of about 300:1. In such displays a LCD panel is backlit with Cold Cathode Fluorescent Lamps (CCFL) emitting a uniform light over the entire surface of the screen. For enlarging the displayable dynamic range, Seetzen et al. proposed in [39] to replace the CCFL back-lighting by an array of modulated Light Emitting Diodes (LED) which can emit a stronger light focused on a small area, thereby enabling higher peak luminance without impacting the rendering of the black areas. This technique, often referred to as dual modulation, has been used by SIM2 to build the first commercially available HDR display, the SIM2-HDR47E, with a peak luminance of 4000 cd/m^2. More recently, Dolby announced the manufacturing of HDR displays (the pulsar series) better designed for the mass-market with a lower price and power consumption.

However, despite the recent advances in HDR display technology, reducing the dynamic range of HDR images remains a necessity for addressing the vast majority of low dynamic range output devices. This operation is called tone mapping. Over the last two decades a wide variety of Tone Mapping Operators (TMO) have been developed. They can be divided into two main categories: the global and the local TMOs. Global TMOs consist in defining a monotonously increasing function (or tone curve) and apply it to all the pixel values of the image. As an example, the TMO in [40] determines a tone curve adapted to the characteristics of a target display and takes into account properties of the Human Visual System (HVS) in order to minimize the perceptual distortion of the tone mapped image. By contrast, the photographic tone reproduction TMO [41], instead of minimizing perceptual distortion on the viewer's side, takes its inspiration from photographic techniques to produce visually pleasing images.

When a very large reduction of the dynamic range is required, global TMOs may result in important loss of details and contrast. In local tone mapping operators, a higher dynamic range reduction is made possible by a preservation of the local contrasts in the final tone mapped image. For instance, in [42] a bilateral filter [43] is used to decompose the HDR image in a high frequency subband and a low frequency subband. Only the low frequency subband is tone mapped and the details contained in the high frequency subband are added back to the tone mapped low frequency image. Alternatively, [44] first computes the gradients of the HDR image to attenuate only the gradients of large magnitude corresponding to strong edges. A low dynamic range image is then retrieved from the filtered gradients. Figure 1.8 shows examples of results for the local and global TMOs described here.

Tone mapping operators are usually applied on the luminance component Y to obtain a tone mapped luminance Y_{TMO} in the range [0,1]. In order to deal with colors, the HDR linear RGB values are then processed as defined in [45]:

$$R_{TMO} = \left(\frac{R}{Y}\right)^s \cdot Y_{TMO}, \quad G_{TMO} = \left(\frac{G}{Y}\right)^s \cdot Y_{TMO}, \quad B_{TMO} = \left(\frac{B}{Y}\right)^s \cdot Y_{TMO} \quad (1.23)$$

where s is a parameter that can be used to adjust the saturation of the tone mapped image. The value of s is generally below 1. This processing preserves well the hues of the colors of the original HDR image. The luminance of the resulting image is approximately equal (or exactly equal in the case s=1) to the originally tone mapped luminance Y_{TMO}.

(a) (b) (c) (d)

Figure 1.8: Examples of tone mapped images with different local and global tone mapping operators: (a) Display adaptive TMO [40], (b) Photographic tone reproduction [41], (c) Bilateral Filtering based TMO [42] and (d) Gradiant domain TMO [44].

Note that the tone mapping operation only consists in a reduction of the luminance range and does not account for gamma correction. In order to display the image with a conventional output device, the R_{TMO}, G_{TMO} and B_{TMO} components still need to be gamma corrected and quantized to 8 bits integers.

1.5 Conclusion

In this chapter, we introduced the basic knowledge to understand how light and colors are measured in photometry and colorimetry by taking into account the spectral sensitivity of the human eye (section 1.1). Then, in section 1.2, we discussed higher level aspects of the human visual system by presenting perceptually uniform representations of colors. We have seen in section 1.3 that the traditional encoding of colors, although linked to the principles of colorimetry and perceptual uniformity mentioned previously, has been designed especially to be in conformity with conventional low dynamic range display technology. In particular, the gamma correction used in the traditional color encoding for compensating the display's transfer function is approximately perceptually uniform up to a certain luminance level. For the luminance range considered in High Dynamic Range imaging, however, more accurate perceptual encodings can be determined. Finally the main HDR image formats and the techniques used to capture and display such images were described in section 1.4.

Chapter 2

HDR image and video compression

The development of High Dynamic Range images and videos brings new challenges regarding the storage and distribution of this extended image format. While traditional Low Dynamic Range images are represented by 8 bit integers per pixel and per color component, we have seen that higher bit-depth or even floating point values are generally required in HDR imaging to represent the full luminance range that can be perceived by the human eye. In the example of OpenEXR format, 16-bit half float data is used to encode each of the RGB components. Using the lossless PIZ compression included in OpenEXR, the storage required for a natural image may be divided by two approximately. While this might be sufficient as an intermediate format for studios, much larger compression ratios are required to enable the storage of a complete movie on a Blu-Ray, or the broadcast of HDR content for instance. For that type of applications, new methods based on the most recent and efficient lossy compression standards must be determined. From a distribution point of view, the issue of backward compatibility is also essential for the transition from legacy decoders and LDR displays to HDR technology. This aspect should also be considered in addition to the compression efficiency in the design of HDR codecs.

This chapter gives an overview of the functioning of typical compression methods through the example of the recent High Efficiency Video Coding (HEVC) standard in sections 2.1 and 2.2. Section 2.3 describes the three types of coding schemes for encoding HDR content. Such methods may be either directly based on existing compression standards or they may require the addition of new tools within standard codecs.

2.1 Image and video compression methods

2.1.1 General compression concepts

Image and video compression methods are based on the principles illustrated in figure 2.1. The first step of decorrelation consists in reducing the redundancies that exist in the content. It is usually implemented by two methods, prediction and transform which can be combined:

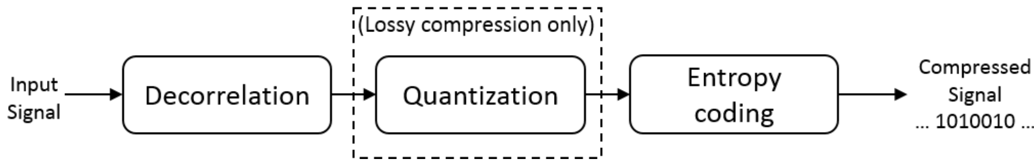


Figure 2.1: General structure of image or video compression schemes.

- The prediction step tries to guess the value of a pixel or block of pixels from the regions that have been previously decoded. The predicted signal is then subtracted from the original signal and only the residual must be transmitted by the encoder to the decoder. If an accurate prediction is found, the residual signal is close to zero and can be encoded accurately with very few bits.

- The transform represents the input signal as a weighted sum of predefined signals. The input signal is then described by the coefficients of the sum, instead of the values of the pixels. The goal of the transform is to compact most of the energy of the signal in a small number of coefficients. In the ideal case, one coefficient is equal to 1 and all the other coefficients are zero, which can be encoded very efficiently.

If both prediction and transform are used, the signal is thus represented by the transform coefficients of the prediction residual. This decorrelated signal is likely to have a lower entropy than the original signal, where entropy represents the amount of information. In information theory, the entropy H of a random variable with a finite number n of possible values, is defined by:

$$H = -\sum_{i=0}^{n} p_i \cdot log_2(p_i) \tag{2.1}$$

where p_i is the probability of the value i. By extension, this formula defines the entropy of a signal, considering the signal as a sequence of random variables whose probability distribution is defined such that the probabilities p_i are the frequency of occurrence of each value i. Since the same probability distribution is considered for every random variable, the entropy does not take into account the correlations in the signal, which explains why exploiting those correlations helps reduce the entropy. The entropy encoding step uses the low entropy property of the decorrelated signal to encode it efficiently into a series of bits.

In the case of lossy compression, the decorrelated signal is quantized (e.g. divided by a factor and rounded) to further reduce its entropy and thus reduce the amount of bits at the output of the entropy encoder. However this step causes loss of information. The quantization step may be adjusted to find a trade-off between the amount of compression and the quality of the reconstructed signal.

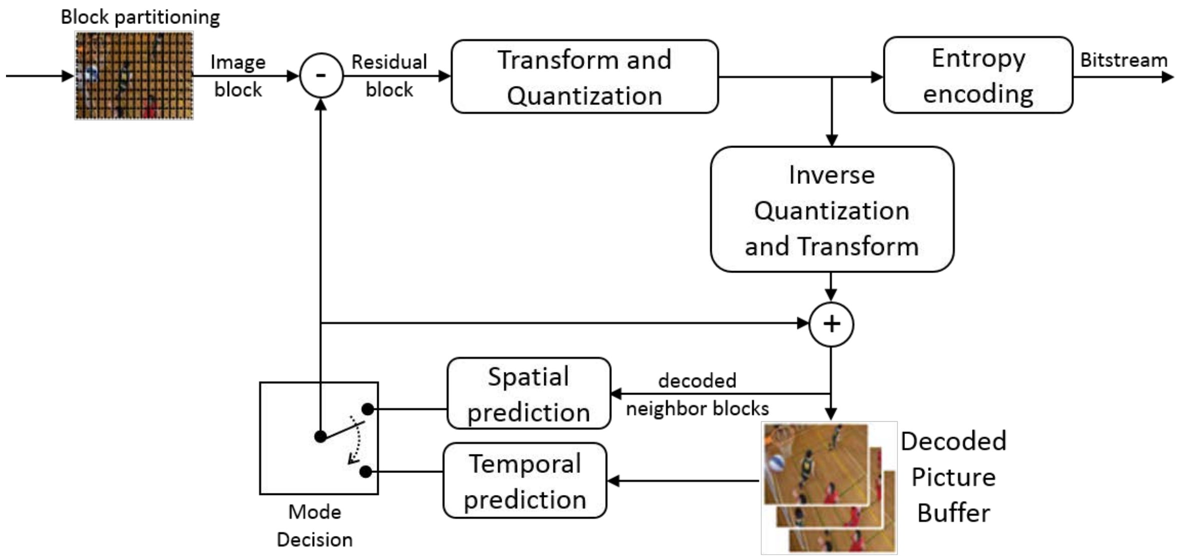


Figure 2.2: Video encoder overview.

2.1.2 Application in video coding

More specifically, in video compression, most standards follow the structure outlined in figure 2.2. Note that in general, the input image is given in the Y'CbCr colorspace with downsampled chroma components for reducing the inter-component redundancies and exploiting the properties of the human visual system.

In the encoder, the frames of the video sequence are first divided into blocks. Each block is then encoded with the techniques presented previously using either spatial or temporal prediction. While spatial predictions are based on the neighborhood of the current block in the current frame, temporal predictions make use of previously reconstructed frames which are likely to contain a region that is very similar to the current block. For each block, the encoder may perform both spatial and temporal predictions in order to determine which of the two modes minimizes the lagrangian function J defined by:

$$J = D + \lambda \cdot R \tag{2.2}$$

where D is the distortion between the original and the decoded block. It is usually measured by the mean squared error (MSE). R is the rate, that is, the number of bits required to encode the block. And λ is a fixed lagrangian multiplier value determined by the encoder to obtain the best trade-off between rate and distortion. This mode decision process is called rate distortion optimization (RDO) [46, 47]. Note that this RDO step can only be performed by the encoder since the original block is unknown on the decoder side. Therefore, the encoder must add a flag (i.e. a binary information) in the bitstream indicating which type of prediction should be used to decode the block. Finally, the encoder subtracts the block predicted with the chosen mode from the original block. The residual is then transformed, quantized and entropy encoded.

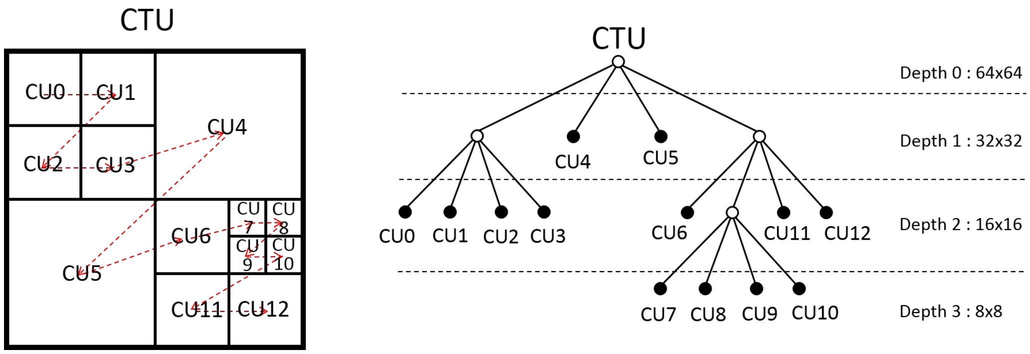


Figure 2.3: Illustration of the HEVC CU quad-tree partitioning. The CUs are processed in the Z-scanning order shown by the red arrows.

The encoder additionally computes the decoded version of the block by applying the inverse quantization and transform and adding back the prediction block. The decoded block is then added to a decoded picture buffer which can be used to perform predictions of subsequent blocks. It is essential that the encoder performs the predictions only from decoded data so that the decoder can find exactly the same predictions.

2.2 Overview of HEVC

Finalized in 2013, the HEVC standard [48] was designed with the objective of reducing the bitrate by 50% at equivalent quality level compared to its predecessor, the ITU-T H.264 / MPEG-4 Part 10 'Advanced video coding' [49], or more simply H.264 or AVC.

This section section presents an overview of the main tools in HEVC.

2.2.1 Quad-tree structure

The most notable improvement of the HEVC standard over H.264 is its quad-tree structure that allows a greater flexibility in the block partitioning of the image according to the content. Four types of blocks are defined in the HEVC partitioning structure:

- Coding Tree Unit (CTU) : Also sometimes referred to as Largest Coding Unit (LCU), it is the largest partition in HEVC. The picture to encode is first divided into a regular grid of 64x64 CTUs.

- Coding Unit (CU) : A CTU may then be divided into four Coding Units. The CU structure forms a quad-tree where the CTU is the root node. It can have four children CUs which can be themselves divided into four smaller CUs recursively. The smallest CU size defined in HEVC is 8x8 corresponding to 3 recursion levels. The processing order of the CUs follows the Z-scanning as illustrated in figure 2.3. The choice of the prediction mode (intra or inter) is decided at the CU level.

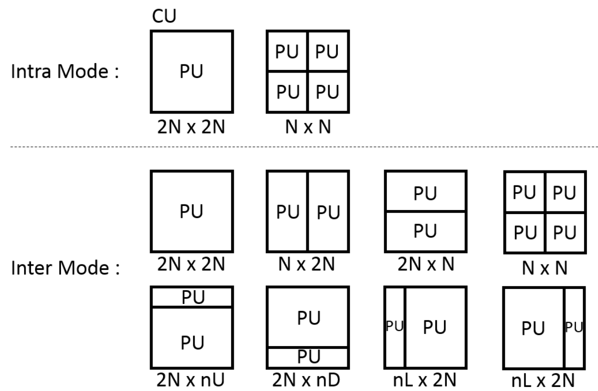


Figure 2.4: PU partitioning schemes.

- Prediction Units (PU) : Each CU contains a PU structure which can be composed of one or several PUs. One PU contains all the information required for performing a prediction (i.e. motion vector for inter, an index for intra). The maximal size of a PU is the size of the CU containing the PU structure, in that case the PU size is called 2Nx2N. Other PU partitioning schemes can be used depending on the prediction mode. They are shown in figure 2.4.

- Transform Units (TU) : Besides the PU partitioning of CUs, each CU also contains a TU structure composed of one or several TUs in which the transform and quantization steps are performed. Transform Units sizes may vary between 4x4 to 32x32. Similarly to CUs, a TU can be split recursively into smaller TUs in a quad-tree decomposition where the root is the CU containing the TU structure. The Z-scanning order is also used for processing the TUs.

2.2.2 Intra prediction

Intra prediction mode only exploits spatial correlations in the current frame. When a block of the image is predicted with intra mode, the reconstructed values of the pixels from the already encoded and decoded neighboring blocks are retrieved. Because of the Z-scanning order, when a block is being processed, its top and left neighbors are already encoded and decoded. Therefore, the reconstructed pixels on the top and on the left of the current block are known by both the encoder and decoder. In addition, in HEVC, different block sizes are possible and the neighboring blocks may thus be bigger than the current block. In this case some pixels on the top right and bottom left sides of the current blocks are also available. Figure 2.5 shows the pixels used for intra predictions. In the case the top right or bottom left pixels are not available, they are padded from the known pixels before performing the actual intra prediction.

A prediction can be derived from the neighboring pixels in several ways. The HEVC standard defines 33 directional modes in which the neighboring pixel values are propagated with a given direction. Two additional modes, DC and planar, may be used.

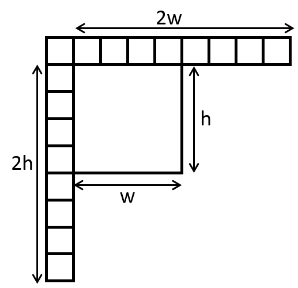


Figure 2.5: Intra prediction neighboring samples.

The DC mode consists in computing the average of the neighbor pixels while the planar mode performs a multi-directional prediction. Figure 2.6 shows the different intra modes and their associated indices in HEVC. The index of the mode chosen is transmitted for each intra-encoded PU so that the decoder can determine which direction (or DC or planar) should be used for predicting the PU. Although the index of the intra mode is stored on the PU level, the intra prediction is performed at the TU level since several TUs may lie inside a PU. In this way, the first TUs in the scanning order are fully encoded and decoded (i.e. including quantization of the transformed residual) and their reconstructed pixels can be used for predicting the next TUs. This gives more accurate results without requiring the transmission of additional information than performing the intra prediction on the entire PU in a single step.

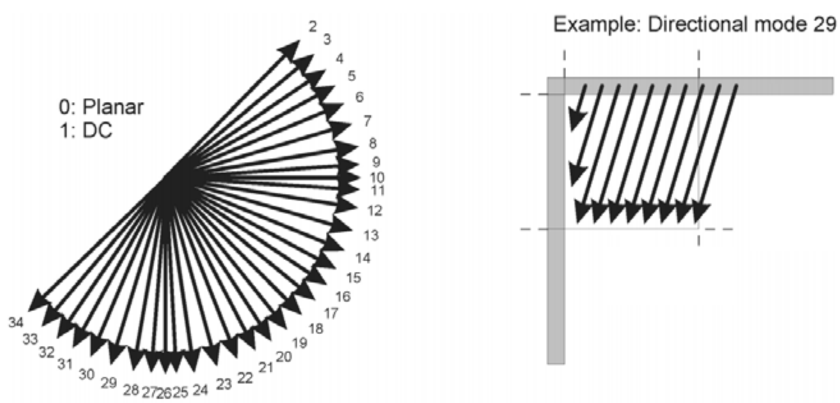


Figure 2.6: Intra prediction modes in HEVC.

Note that for encoding the chroma components, only five modes are available : the planar, DC, vertical, horizontal and direct mode (DM). DM mode simply consists is using the same mode that was used for luma. This mode is likely to give good results because of the spatial correlations that exist between luma and chroma components.

2.2.3 Inter prediction

Inter predictions use the temporal redundancy that exists in a video sequence. It introduces dependencies between frames since one or several reference frames that have been previously encoded are necessary for encoding or decoding a current frame. In HEVC several types of frames are defined:

- I frames are encoded independently of the other frames, therefore using only inta prediction mode. Since intra predictions are usually less accurate than inter, I frames have a higher coding cost than the other frames and account for a significant part of the overall bitrate.

- P frames use temporal predictions from the preceding frames in the sequence. Either intra or inter prediction modes may be used for the CUs of P frames, which improves significantly the coding efficiency.

- B frames are bi-predicted. They can use reference frames that are either before or after them in the sequence, which reduces even further the coding cost compared to I and P frames. However, when a sequence is encoded using B frames, the coding order must be different from the order in which the frames appear in the sequence.

In HEVC, a video sequence is divided into Groups Of Pictures (GOP) following one another over time. All the GOPs contain I, P and B frames ordered following the same pattern. Figure, 2.7 shows a usual example of GOP structure. The coding order, or Picture Order Count (POC), is computed automatically from the dependencies that exist between the frames of the GOP. Although the frames are reordered by the encoder, all the frames of a GOP are always encoded before those of the subsequent GOPs. However, their might be a dependency between a GOP and the one that precedes it.

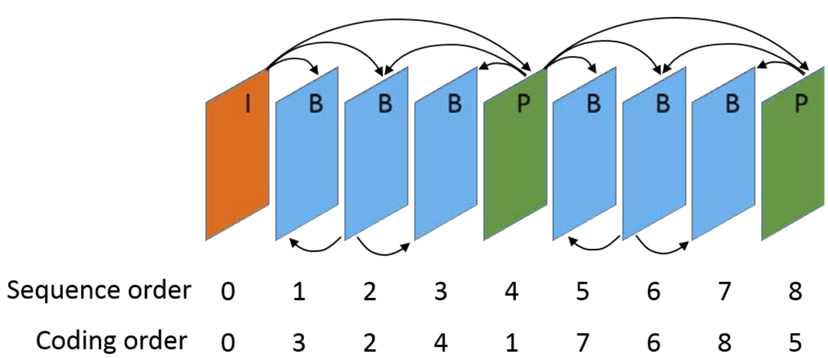


Figure 2.7: Example of GOP structure. In this example, a B frame may be used for the prediction of other B frames. This is called a hierarchical GOP structure.

Given a current frame and an associated reference frame in the sequence, the inter prediction of HEVC performs a motion compensation using a motion vector defined with a quarter pixel precision. On the encoder side, the motion vector is determined for a given Prediction Unit by motion estimation with the reference frame. It is then encoded in the bitstream along with the index of the reference frame so that the decoder can perform the same motion compensation to predict the PU. A motion vector prediction based on the previously inter-encoded neighbor PUs is performed in order to encode only the residual between the actual motion vector and the predicted one.

Additionally to normal inter prediction, HEVC defines the merge mode and the skip mode in which no motion estimation is performed. Instead, a predicted motion vector is derived and used directly for motion compensation. Note that a different vector prediction method than in normal inter mode is used. It consists in selecting one of up to five motion vector candidates taken from neighboring PUs, and transmitting only the index of the selected candidate. The difference between merge and skip is that in skip mode, the quantized transform coefficients of the block residual are not encoded. The reconstructed PU is then the same as its prediction. Both those modes are less computationally expensive on the encoder side and require less information to transmit than normal inter mode, although the prediction itself requires the transmission of an index.

2.2.4 Transform and quantization

HEVC, as well as most video compression standards, use the two-dimensional discrete cosine transform (DCT) [50] which represents the signal as a weighted sum of 2D basis functions built from the cosine function. The basis functions are represented in figure 2.8 in the case of the 8x8 DCT transform.

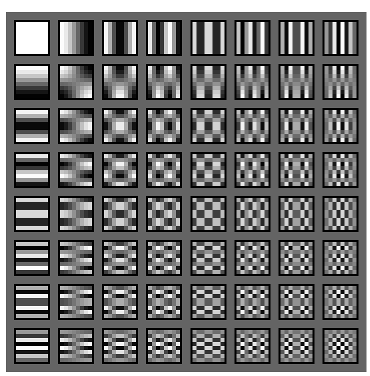


Figure 2.8: 2D DCT basis functions for the transform of a 8x8 block.

The result of the DCT is a 2D block F of transform coefficients of the same size as the original block I. Given a block size NxN, the DCT coefficients in F are expressed as

$$F(u,v) = \frac{1}{4} C_u C_v \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} I(m,n) \cos(u\pi \frac{2m+1}{2N}) \cos(v\pi \frac{2n+1}{2N}) \tag{2.3}$$

where

$$C_u = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } u = 0 \\ 1 & \text{otherwise} \end{cases} \quad \text{and } C_v = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } v = 0 \\ 1 & \text{otherwise} \end{cases} \tag{2.4}$$

The DCT is heavily used in compression because of its good energy compaction properties. In other words, it concentrates most of the energy of the signal in the top left coefficients corresponding to low spatial frequencies. In particular, the top left coefficient, $F(0,0)$ is called DC coefficient and is equal to the average value of the block (by ignoring a normalization factor). In addition, the two-dimensional DCT has the advantage of being separable, which means that it can be built by computing first the 1D DCT independently on each line and then on each column of the block. However the DCT should be applied to relatively small blocks of the image to remain efficient. In HEVC the maximum block size for applying the DCT is 32x32 (i.e. maximum TU size). Note that in the particular case of 4x4 intra predicted luma blocks, HEVC transforms the residual with the discrete sine transform (DST), which is similar to DCT and was found to give better results in that case.

For fast computations, an integer version of the DCT is performed in HEVC. The resulting coefficients are thus only integer values with a bitdepth that is sufficient not to cause any loss of information at this stage, given that the input signal is also represented by integers of a known bitdepth.

The loss of information and the consequent bitrate reduction are entirely controlled by the quantization of the coefficients. Uniform quantization is applied in HEVC, which means that a fixed quantization step size is used over the whole range of values that can be taken by the transform coefficients. The size of this step is controlled by an integer valued quantization parameter (QP) ranging from 0 to 51. It is defined such that an increase by 6 of the QP value doubles the quantization step size. Note that in the versions of HEVC allowing higher bitdepth than 8, negative QP values are allowed for very high encoding quality.

2.2.5 CABAC Entropy coding

In HEVC, a Context Adaptive Binary Arithmetic Coder (CABAC) is used for the entropy encoding of all the data in the bitstream, including the quantized transform coefficients, but also the motion vectors, the intra mode indices as well as all the syntax required for decoding. The principle of CABAC is based on arithmetic coding [51] which makes use of the probabilities of each value that the signal can take in order to encode it with a minimal number of bits. In the case of binary arithmetic coding, the signal to encode must be given as a sequence of bins, each representing a binary information. For each bin, two values, or symbols, can thus be taken (i.e. a 0 or a 1).

The binary arithmetic encoding process is illustrated in figure 2.9. The coder first splits the range [0,1] in two non-overlapping intervals whose lengths are equal to the probability of each symbol. The interval corresponding to the symbol taken by the first bin is then selected. For each subsequent bin in the signal, the selected interval is further subdivided proportionally to the symbol's probabilities and the algorithm selects the

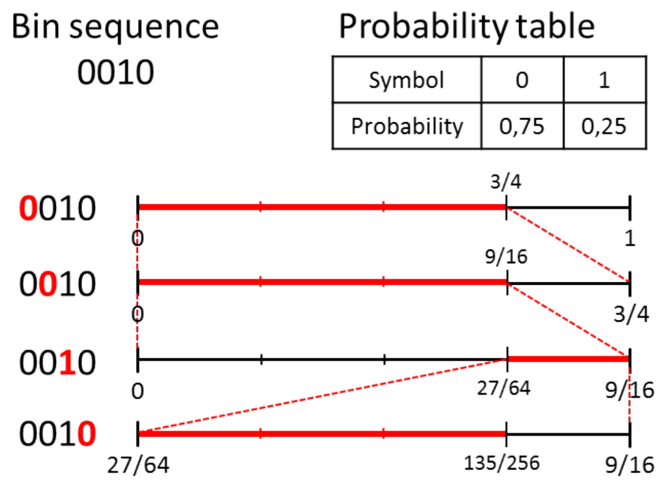


Figure 2.9: Binary arithmetic coding process. The selected sub-interval at each step is represented in red.

sub-interval corresponding to the bin's value. When the last bin is reached, the selected interval is representative of the complete sequence since any other sequence would have resulted in a different and non-overlapping interval. A single decimal number falling in that interval can then be used to encode the whole sequence. The algorithm finally outputs the minimum number of bits required to identify uniquely the interval using a binary fractional representation. At the output of the arithmetic encoder, the average number of bits per encoded bin is close to the entropy of the binary signal. Therefore, arithmetic coding is more effective when the probability of one of the symbols is close to 1.

In CABAC this process is adaptive in the sense that the probabilities are not fixed for the whole sequence but they are first initialized and then updated after each bin. If the symbol of a bin to encode is equal to the most probable symbol (MPS), the probability of the MPS increases (and thus the probability of the least probable symbol (LPS) decreases) for the encoding of the following bin. This is shown in figure 2.10. Each type of binary information that must be transmitted in HEVC is encoded with a specific CABAC context containing a probability model that is updated each time a bin is encoded using the same context. For instance, a CABAC context is defined for encoding the flags indicating the prediction mode chosen for each CU. The updating of the context's probability enables the encoder to exploit correlations between the previously encoded flags and the current one. In the extreme case where all the CUs use the same mode, the MPS quickly reaches a probability close to 1 so that the number of bits required to encode all the flags is much smaller than the number of flags.

The data that are not binary (e.g. motion vectors, coefficients,...) must be binarized before being encoded with CABAC. For instance, numerical values may be converted to bins with a fixed length representation. Each bin may then be encoded using a specific

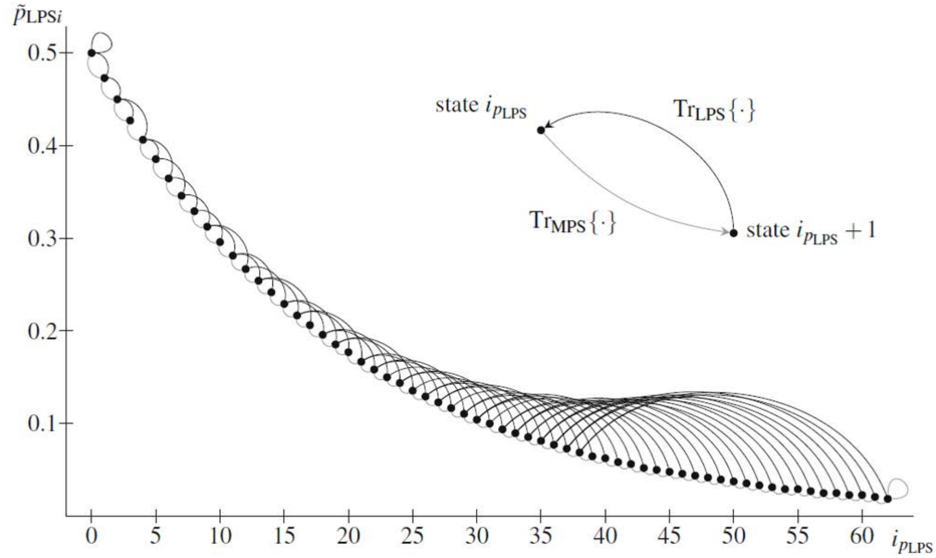


Figure 2.10: CABAC states update. $\tilde{p}_{LPS_i}$ is the probability of the least probable symbol (LPS) at a given state of the CABAC context. It decreases according to the arrow Tr_{MPS} when a most probable symbol (MPS) is encoded. Conversely, when the least probable symbol is encoded, $\tilde{p}_{LPS_i}$ increases according to Tr_{LPS}.

CABAC context. In practice, however, in a fixed length representation, only the most significant bins may still have correlations to exploit. The least significant bins are usually encoded using equal probabilities between the two symbols. Each of these bins will then cost one bit on average.

2.2.6 In-Loop Filtering

When a sufficiently strong quantization is applied, several types of artifacts may appear in the reconstructed image that may significantly affect of the perceived quality. For instance, because of the block structure of the codec, discontinuities at the block bound-

Figure 2.11: Left : An image with block artifacts. Right : An example of ringing artifacts caused by the strong edges in the letters.

aries may become apparent. In addition, the quantization in the DCT domain causes ringing artifacts, also known as Gibbs phenomenon, that appear near strong edges in the image. Examples of blocking and ringing artifacts are shown in figure 2.11 For removing such artifacts, two filters were adopted in the HEVC standard: the deblocking filter [52] for reducing block artifacts, and the sample adaptive offset (SAO) [53], that was designed against ringing artifacts.

These filters are called in-loop because they are included in the coding loop so that the picture buffer used for further predictions contains already filtered data. Therefore, in-loop filters may indirectly improve the predictions. Another advantage of applying the filters in the coding loop is that the filter's parameter may be adjusted on the encoder side to minimize the mean square error between the original and the reconstructed image. The parameters must then be transmitted to the decoder. In particular, the SAO requires the transmission of several parameters on the CTU level.

2.3 HDR compression schemes

The video compression standards have been designed for encoding and decoding LDR images given in a gamma corrected 8-bit format. HDR images are usually represented by floating point data in linear RGB or XYZ colorspaces. In order to use the existing video compression standards for encoding HDR content, a first conversion to integers is thus necessary. However, a uniform quantization of the linear data will results in a very inefficient bit allocation because of the highly non-linear human perception of luminance. The use of a HDR perceptual curve is then required before quantizing the values to obtain integer data. Furthermore, even with a perceptually uniform encoding, a higher bitdepth than 8 is necessary for ensuring that the conversion does not produce visible artifacts. Another important constraint to consider in HDR compression is the backward compatibility with legacy decoder and display systems. In this section, we categorize the HDR compression schemes into three types of architectures addressing different levels of backward compatibility.

2.3.1 Single HDR layer

The most straightforward scheme consists in encoding a single HDR layer as illustrated in figure 2.12. The High Dynamic Range image or video is first converted to a rather perceptually uniform colorspace. For instance, a process very similar to the way typical LDR images are encoded is often used: a HDR perceptual curve, analogous to the gamma correction used for LDR images, is first applied to the linear RGB components independently. The resulting perceptually encoded RGB values are then rounded to high bit-depth integers (e.g. 10, 12, 16 bits) and converted to Y'CbCr colorspace. An encoder that supports high bitdepth integers is then used to compress the converted HDR data. In the HEVC standard for instance, although the main profile is limited to 8-bit integers, Range-Extension (RExt) profiles were defined that can take up to 16 bit input data.

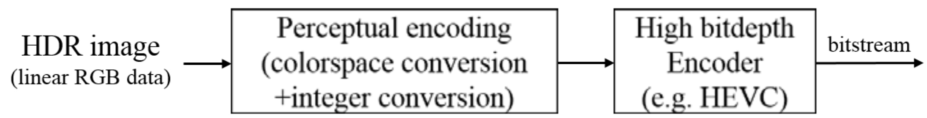


Figure 2.12: Single HDR layer compression scheme.

The main drawback of this scheme is that it does not address backward compatibility since only an HDR version is encoded. Furthermore, a high bitdepth decoder is required to read the bitstream. Legacy equipment are therefore not able to read and display the HDR content encoded with this method.

2.3.2 Backward compatible single layer

In backward compatible single layer schemes, as shown in figure 2.13, a tone mapped and gamma corrected version of the HDR image is encoded along with metadata indicating how to perform the inverse tone mapping that retrieves the original tones of the HDR image. Alternatively, in [54, 55, 56], an HDR perceptual curve (e.g. logarithm, PU curve [12], etc) is applied to the original HDR data before the tone mapping. In this case gamma correction is no longer needed. Conventional devices can ignore the metadata and only decode the image in a suitable format for LDR displays. An HDR decoding system can additionally use the metadata to perform inverse tone mapping and retrieve the HDR image.

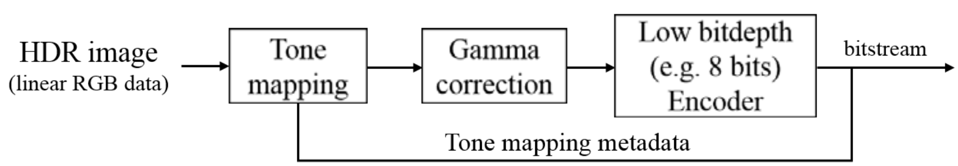


Figure 2.13: Single layer backward compatible compression scheme.

Several examples of backward compatible schemes using a single layer exist in the literature. They mainly differ by the tone mapping operator used. In this context, the choice of the tone mapping operator may be driven by two purposes:

- The coding performance : Since the tone mapped image is quantized to low bit-depth integers, some information is lost before applying the actual compression. In particular, when the original HDR image contains large regions of smoothly varying luminance, the reconstructed image (i.e. after tone mapping, quantization and inverse tone mapping) may contain banding artifacts, as illustrated in figure 2.14. This is a very common problem particularly for outdoor scenes where the sky covers a large part of the image. In [54, 55], Mai et al. have shown that an appropriate global TMO can minimize this loss and thus increase the quality of the decoded HDR image.

Figure 2.14: Example of banding artifact. For the illustration, the effect was simulated by quantizing the image to 5-bit integers.

- The artistic intent : The initial purpose of tone mapping operators remains to keep details in both dark and bright areas of the image while preserving a visually pleasing image that fits the artistic intent of the producer.

However, those two goals are not always compatible and a trade-off must be found. In [57] for instance, the photographic tone reproduction TMO [41] is used and its parameters are optimized for the compression performance. Note that a lesser degree of backward compatibility can still be addressed by focusing only on the coding performance, with no consideration for the appearance of the LDR image. This image can be decoded by a regular decoder but requires further processing for a better viewing experience on a LDR display.

It should be noted in addition that such compression schemes are often limited to global tone mapping operators that only require the transmission of a tone curve in order to perform the inverse tone mapping on the decoder side. The local tone mapping operator developped in [58] is an exception, though. The inverse tone mapping process can be performed from the tone mapped image only without requiring additional information.

2.3.3 Two layer scalable encoding

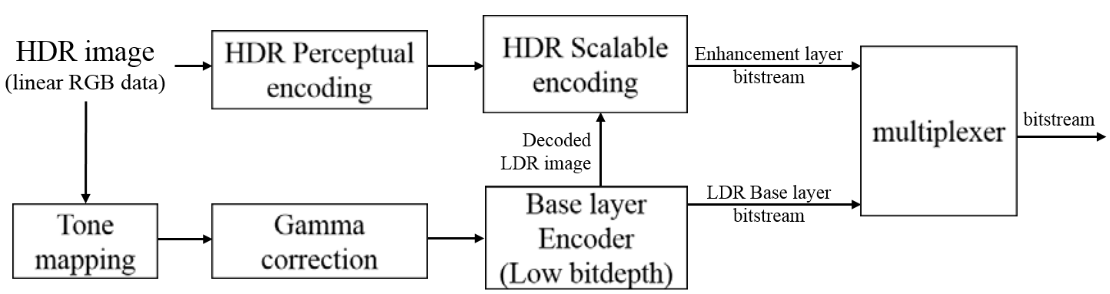


Figure 2.15: HDR Scalable encoding scheme.

Scalable HDR encoding schemes enable full backward compatibility with more freedom on the choice of the tone mapping operator used for generating the LDR version, and no limitation on the quality of the finally decoded HDR version. The principle of HDR scalable encoding is depicted in figure 2.15. A tone mapped LDR version of the image is first encoded as a base layer in a conventional low bitdepth format. The decoded LDR layer is then used by a scalable HDR encoder that exploits the redundancies between the LDR and the HDR data to encode efficiently the HDR image. The two layers are finally combined in a single bitstream. A conventional decoder can only read the data corresponding to the LDR image while a HDR scalable decoder will additionally decode the enhancement layer to obtain the HDR image.

Although the choice of the TMO may still influence the coding performance, the loss of precision in the LDR layer can be compensated by encoding the enhancement layer with sufficient quality. HDR scalable schemes require higher bitrate than a single layer encoding of the HDR image, since both a LDR and the HDR version of the image are encoded. However, thanks to the strong correlation between the two versions, scalable schemes may significantly increase the compression performance in comparison to an independent encoding of both versions.

In HDR scalable schemes, the enhancement layer can take different forms. In [59] and [60] for instance, both the TMO and the inverse TMO require a low-pass filtered version of the luminance channel. This low frequency luminance channel is then transmitted as an enhancement layer. In this case, however, this second layer can be merely considered as metadata since it only contains information required for inverting the tone mapping process specified by the encoder. As a result, the backward compatibility remains only partially adressed in the sense that the LDR image automatically generated by the encoder might not fit the artistic intent of the producer. In the case where an arbitrary TMO is used to generate the LDR version, a more generic approach is required.

To this end, Ward and Simmons, developed the JPEG-HDR format [61], a scalable encoding method based the JPEG standard. In the JPEG-HDR codec, the enhancement layer is the logarithm of the ratio between the LDR and the HDR images. It is encoded on 8-bits and compressed in the JPEG format as well as the LDR base layer. When the file is read in regular software, the metadata is ignored and only the LDR image is decoded. JPEG-HDR compliant software additionally reads the ratio image and reconstructs the HDR image. In [62], Mantiuk et al. automatically compute an inverse tone curve based on the HDR image to encode and the decoded LDR image. The curve is encoded and used to predict the HDR image from its decoded LDR version. The prediction residual is then filtered to remove invisible noise and it is quantized to be finally compressed with a standard 8-bit MPEG encoder. Compared to the ratio image in Ward and Simmons method [61], the residual image is easier to compress thanks to the decorrelation obtained by the prediction scheme. Since the prediction consists in applying a tone curve on the whole image, it is very well suited for global TMOs. The principles developed in both [61] and [62] are becoming increasingly popular. For instance, similar prediction methods have been included in different profiles of the upcoming JPEG-XT standard [63]. In particular, the Profile A of JPEG-XT corresponds to the JPEG-HDR method. Furthermore, several scalable compression schemes have been developed based on the principles of either ratio image or global inverse tone curve

[64, 65]. Such approaches have the advantage of requiring only legacy low bit depth codecs without modification. Therefore, they are rather simple to implement.

However, more flexibility and potentially better decorrelation can be obtained by including an inter-layer prediction step in the coding loop of a compression image or video standard. In [66, 67, 68] for instance, the enhancement layer is directly the perceptually encoded HDR image. A high bitdepth version of the H.264 standard was modified by defining an inter-layer prediction (ILP) mode in addition to the existing intra and inter modes in order to encode efficiently an HDR image from the decoded LDR image. Because of the block structure of the H.264 standard, the properties of the ILP can be adapted locally for a better decorrelation, especially in the case where a local TMO was used for generating the LDR layer.

2.4 Conclusion

In this chapter, we have introduced the general principles underlying any image or video compression method, and how they are implemented in the recent HEVC standard. Although those principles remain the same for the compression of High Dynamic Range content, new techniques must be found in order to keep backward compatibility with existing standard decoders and conventional display devices. The direct encoding of an HDR image using high bitdepth profiles of an existing standard is not backward compatible. However, we have seen two other types of compression schemes, using either one or two layers, that can address this problem. The work presented in this thesis manuscript falls within these two categories and enable different levels of backward compatibility. While the next chapter studies the optimization of the compression performance in single layer schemes, the two following chapters are dedicated to inter-layer prediction in a HDR scalable context.

Part II

Contributions

Chapter 3

Bit-depth reduction for single layer compression

Existing image and video codecs were originally designed for relatively low bitdepth input data. In the recent HEVC standard, the main profile only supports 8-bit input data. Although extended versions can support a bitdepth of up to 16 bits, their use can be restricted because of the increased implementation and computational cost. In particular, the 16-bit profiles are clearly not intended for being implemented in mass-market decoders. However, 10 or 12 bits might become a new standard for television. The recent ITU-R Recommendation BT.2020 (Rec.2020) [69] for Ultra HD television defines a bitdepth of either 10 or 12 bits, in addition to the specification of a new RGB colorspace with extended gamut compared to the Rec.709 [70] colorspace for HD television. It is therefore likely that 10 or 12 bits HEVC decoders will be implemented for modern television systems. From the initial floating point representation of the HDR image data (i.e. 16 or 32 bits), a lossy conversion to lower bitdepth integers (e.g. 10, 12 bits) is then required in addition to the perceptual encoding of luminance. Moreover, for addressing backward compatibility without resorting to a rather complex scalable compression scheme, further bit-depth reduction to 8 bits is required.

In the literature, several attempts have been made to encode HDR content using high bit depth versions of a H.264 or HEVC codec after a minor reduction of the data precision. For instance, [71] presents a modified LogLuv transform where the minimum and maximum luminances of the frames are used to map the floating point numbers to 14 bit integers for the luma channel. The resulting frame-wise adapted values are then encoded using H.264/AVC compression. In [72], the performance gain of HEVC over H.264/AVC is studied for sequences of floating point images previously converted to 10 and 12 bits per component.

In this chapter, the focus is set on the coding performance of single layer schemes where the bitdepth of the HDR content is first reduced by a tone mapping operator. In our approach, the tone mapped images at the input of the encoder are not tuned for artistic purposes. The tone mapping step is then rather seen as quantization. Given floating point input images, we show in section 3.1 that a preliminary conversion to high bitdepth integers can be performed without loss and can be considered as an approximate perceptual encoding. From this integer image, we study in section 3.2 the effects of various uniform quantization schemes on the compression performance. Then, section 3.3 is dedicated to the definition of the optimal tone curve (or equivalently, non-uniform quantizer) for the rate-distortion performance of single layer backward compatible schemes.

3.1 Approximate logarithmic encoding

3.1.1 Floating point bit pattern

HDR content is generally encoded in floating point data representing physical luminance (or linear RGB) values. In particular, we have seen in chapter 1 that the OpenEXR format uses half-float data based on a sign bit, a 5-bit exponent e and a 10-bit mantissa m. A half float value f is then given by:

$$f = \begin{cases} (-1)^s \cdot 2^{e-25} \cdot (1024+m), & \text{if } 1 \leq e \leq 30 \\ (-1)^s \cdot 2^{-24} \cdot m, & \text{if } e = 0 \end{cases} \tag{3.1}$$

And the exponent value 31 encodes NaN or infinity values.

As mentioned in [73], a piecewise linear approximation of the logarithm function is obtained by taking the bit pattern of a positive floating point value (i.e. the concatenation of the exponent bits and the mantissa bits) and by interpreting the bits as an unsigned integer. For instance, in the case of positive half-float numbers, the integer representation of the bit pattern gives a value i defined by

$$i = 2^{10}.e + m \tag{3.2}$$

Therefore,

$$e = (i - m).2^{-10} \tag{3.3}$$

According to equation (3.1), in the case $e > 0$, we have:

$$f = 2^{(i-m)\cdot 2^{-10}-25} \cdot (1024+m), \tag{3.4}$$

$$f = 2^{i\cdot 2^{-10}-25} \cdot \frac{1024+m}{2^{m\cdot 2^{-10}}}, \tag{3.5}$$

$$f = 2^{i\cdot 2^{-10}-25} \cdot 1024 \cdot \frac{1+m\cdot 2^{-10}}{2^{m\cdot 2^{-10}}}. \tag{3.6}$$

Knowing that for $0 \leq m < 2^{10}$:

$$1 \leq \frac{1+m\cdot 2^{-10}}{2^{m\cdot 2^{-10}}} \leq 1.0615,$$

the following approximation can be derived :

$$f \approx 2^{i\cdot 2^{-10}-15}$$

It follows :

$$i \approx 2^{10} \cdot (log_2(f) + 15) \tag{3.7}$$

Assuming an input HDR image in the OpenEXR format, this approximate logarithmic encoding has the advantage that it does not require any computation since it is the way OpenExr's half float values are stored internally. It is also advantageous in lossless or near lossless compression and has been used in this context, for example, in [74] or in the PIZ algorithm from OpenExr. Moreover, a logarithmic encoding of luminance values is in accordance to Weber law of perceptual uniformity. Although more accurate models of the human perception of luminance have been determined, we will consider in this chapter that this encoding is approximately perceptually uniform. Here, the input images are supposed to contain only positive values. Hence, the encoding of the sign bit is not considered. The converted data is then defined by 15-bits integers. We also assume that the floating-point image do not contain NaN or infinity values. The maximum exponent is then 30 and the integer i is in the range $[0, 31743]$.

3.1.2 YCbCr conversion

For an efficient compression of the 15-bits RGB components obtained, a conversion to Y'CbCr colorspace is additionally performed. Assuming that the input RGB colorspace is defined by the Rec.709 chromaticities and given that the maximum value is 31743, the Y'CbCr values are defined by:

$$Y = w(0.2126R + 0.7152G + 0.0722B), \tag{3.8}$$

$$Cb = \frac{wB - Y}{1.8556} + \frac{2^{15} - 1}{2}, \tag{3.9}$$

$$Cr = \frac{wR - Y}{1.5748} + \frac{2^{15} - 1}{2}, \tag{3.10}$$

with $w = \frac{2^{15}-1}{31743}$.

3.2 Adaptive Uniform quantization

In this section, we study the impact on coding performance of a uniform adaptive quantization scheme applied to the 15-bit Y'CbCr image before compression with an HEVC encoder. The complete encoding scheme is depicted in figure 3.1. Given a target bitdepth n equation (3.11) describes the quantization of a pixel value x. The inverse quantization step is given in equation (3.12).

$$x' = x - x_{min}, \qquad \text{if } x_{max} - x_{min} \leq 2^n - 1 \tag{3.11a}$$

$$x' = \frac{(x - x_{min}) \cdot (2^n - 1)}{x_{max} - x_{min}}, \qquad \text{otherwise} \tag{3.11b}$$

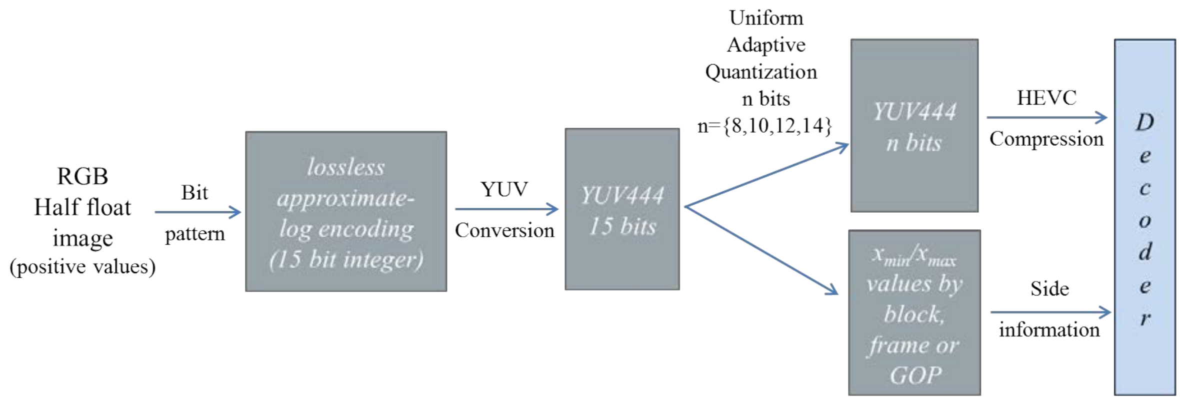


Figure 3.1: Adaptive Uniform Quantization scheme for floating point HDR image encoding.

$$x_{dec} = x' + x_{min}, \qquad \text{if } x_{max} - x_{min} \leq 2^n - 1 \tag{3.12a}$$

$$x_{dec} = \frac{x' \cdot (x_{max} - x_{min})}{2^n - 1} + x_{min}, \qquad \text{otherwise} \tag{3.12b}$$

Where x_{min} and x_{max} are respectively the minimum and maximum values in the region containing the pixel. It might be either a block, a frame or a GOP. The quantized and decoded values are denoted x' and x_{dec}.

Similarly to the Adaptive LogLuv [71], the minimum and maximum values are used to adapt the mapping to the data. However, we distinguish two cases. When the dynamic of the data is superior to that of the target bit depth, it is necessary to downscale the values. Otherwise, when the data has a sufficiently low dynamic to fit into the target bit depth, no re-scaling is applied. If the same formula were used in this case, the data would be unnecessarily upscaled.

The quantization schemes tested differ by the target bitdepth used and by the regions (i.e. blocks, frames or complete GOPs) in which minimum and maximum values are defined.

3.2.1 Block, frame and GOP wise variants

Three variants of the method are proposed using either a 16x16 block, a frame, or a Group Of Pictures (GOP) as the basic unit. As shown in figure 3.1, minimum and maximum values of all the sequence's regions (i.e. blocks, frames or GOPs) have to be transmitted to the decoder to be able to perform the inverse quantization.

The advantage of the block-wise method is that the dynamic is more likely to be small in a block than in the whole image or in a GOP. As a result, the quantization is less severe for low target bit depths. Moreover, if the input image contains some pixels with extremely high or low values, only the blocks containing those pixels will be affected. In counterpart, the coherence between blocks is not preserved. This property is not well suited to the HEVC encoder which tries to estimate blocks of pixels by a prediction based on spatial or temporal neighboring pixels. The discontinuities between the quantized blocks will bias those predictions and thus degrade the compression performance. In addition, more minimum and maximum data must be encoded along with the HEVC bit stream. The frame-wise variant is better to keep the effectiveness of intra predictions, but does not preserve temporal coherence.

In order to improve the performance of temporal predictions, a GOP-wise method is also proposed. In this method, the first and last frames of the GOP are intra coded and are used as reference pictures in a hierarchical coding scheme for the prediction of the frames in-between. The last frame of a GOP corresponds to the first frame of the next one so that only one I picture is needed at the transition. However, as the quantization applied to each GOP may be different, using directly the last frame of a GOP as a reference frame for the next one would break the temporal coherence. Instead, the first GOP is coded, decoded and rescaled to its original dynamic. The last decoded frame is then quantized with the minimum and maximum values of the next GOP before being used as a reference. This process is illustrated in figure 3.2. Note that for the experiment, the coding cost of the first frame in the second GOP is not taken into account for the calculation of the total bit rate because it was already encoded in the previous one.

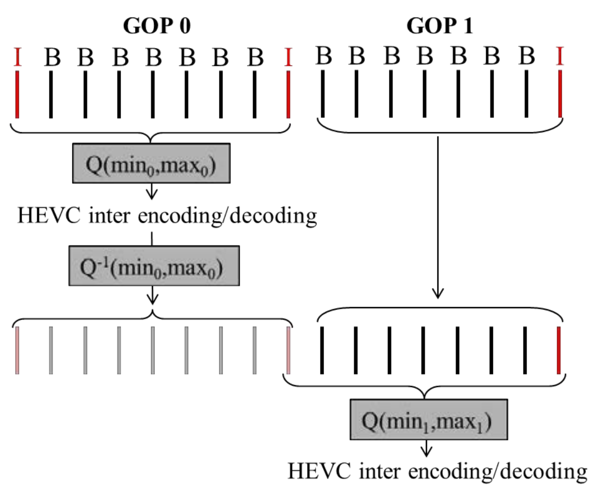


Figure 3.2: The process used for the GOP-wise method. This method keeps the temporal coherence inside a GOP and between two GOPs to better exploit temporal predictions. The quantization and inverse quantization steps are referred to as Q and Q^{-1} respectively, and min_i and max_i are the minimum and maximum values in the GOP i.

3.2.2 Encoding of the quantization parameters

The information of minimum and maximum values is necessary to perform the inverse quantization and must be transmitted to the decoder. In the frame and GOP-wise methods, the number of bits needed to store those two values is negligible compared to the total coding cost. But it is not the case for the block-wise method. A simple way to reduce the bit length of the quantization parameters was used here. For the minimum value, the 15 bits are directly transmitted for each block. Then the difference *delta* between the maximum and minimum is computed. According to equation (3.12), if $delta \leq 2^n - 1$ (where n is the target bitdepth), the exact value of *delta* is not needed. In this case, only the $15 - n$ most significant bits of *delta* are transmitted. They are all equal to zero. Otherwise at least one of these bits is non-zero and we transmit all the bits. On the decoder side, we first read the $15 - n$ most significant bits, if they are all zeros, equation (3.12a) applies, otherwise the remaining n bits are read and equation (3.12b) applies.

3.2.3 Results

The tests were performed on 17 frames of the 1920x1080 sequence "Tunnel video clip" shown in figure 3.3. For HEVC encoding, we used the HM range extension in [75] which enables YUV 4:4:4 input since no subsampling was performed on the chroma channels. Two GOPs of size 8 were used for the tests with inter predictions. Each curve is constructed by encoding the sequence under varying Quantization Parameters (QP) ranging from 0 to 50. Negative QP values have also been used in figure 3.4 to compare 14 bit frame-wise and 12 bit block-wise methods in the context of near lossless compression. The PSNR levels obtained without HEVC compression are also presented in table 3.1. It represents the maximum quality reachable by each variant of our coding scheme.

We also implemented the frame adaptive LogLuv transform from [71] with 14 bits for luma and 8 bits for chroma. In the article, the transform was applied before H264/AVC encoding. Instead, we used HEVC intra so as to be able to compare it with our quantization method. The results are presented in the form of bit rate distortion curves where

Figure 3.3: HDR sequence "Tunnel video clip".

Method	8 bits	10 bits	12 bits	14 bits
GOP-wise	65.32	77.27	87.68	95.38
Frame-wise	66.87	78.75	89.98	95.38
Block-wise	85.21	93.31	95.32	95.38

Table 3.1: PSNR levels (dB) obtained without HEVC compression. Only YUV conversion and adaptive quantization are applied.

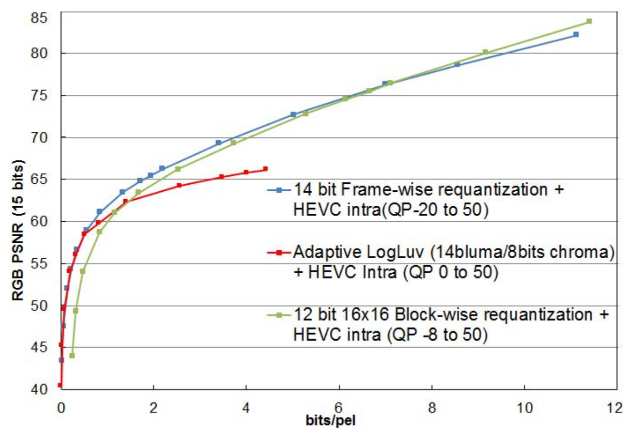


Figure 3.4: Comparison between 14 bit frame-wise, 12 bit block-wise and frame adaptive LogLuv in [71].

the bit rate is computed in average number of bits per pixel and the quality metric is the PSNR computed on the 15 bits integer RGB components.

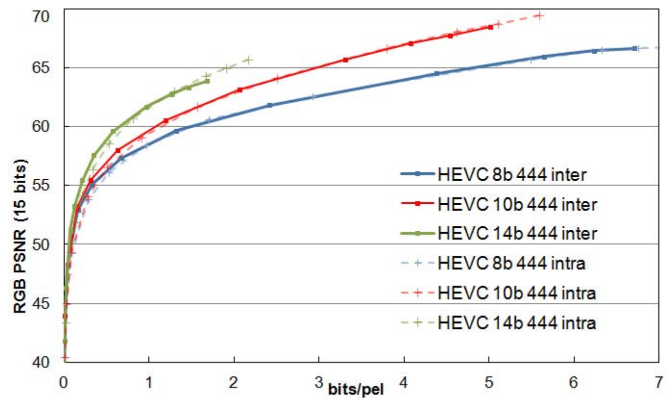
In figure 3.4, the 14 bit frame-wise method is compared to the adaptive LogLuv transform which also modifies the bit depth of the data on a per frame basis. A significant improvement over the latter method is observed, especially at high bit rates and PSNR. For low QP values ranging from 0 to 4, a 52% average rate improvement is observed using the Bjontegaard metric [76].

In the same figure, we plot the rate-distortion curve of the 12 bit block-wise method. It outperforms the 14 bit frame-wise method at very high bit rates and PSNR. However the loss of coherence between blocks and the coding cost of the quantization parameters significantly degrades its performance at lower bit rates.

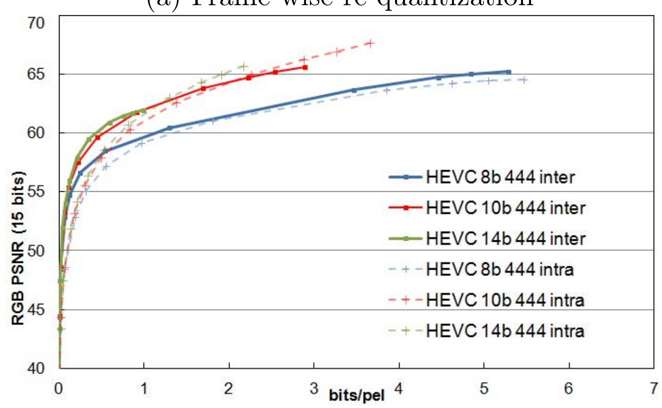
In figure 3.5, inter and intra coding are compared at different bit depths for GOP-wise and frame-wise variants. For both of these variants, the best results are obtained with a target bit depth of 14, which is the highest bitdepth tested.

For the frame-wise method, almost no difference is observed between inter and intra in figure 3.5 (a). This is due to the loss of temporal coherence caused by this method.

When the GOP-wise method is used, the temporal coherence is kept. As a result, inter frame predictions perform better than intra frame predictions. This is visible in figure 3.5 (b) especially in the 14 bit version. However, when high PSNR levels are reached, inter and intra encoding have similar rate-distortion performance.



(a) Frame-wise re-quantization



(b) GOP-wise re-quantization

Figure 3.5: Rate-distortion curves for HEVC intra and inter (QP 0-50) after : frame-wise method (a) and GOP-wise method (b).

3.2.4 Conclusion on adaptive uniform quantization

In this work, we have reduced the bitdepth of the original data so that it can be supported by existing compression standards such as HEVC. We have seen that the best rate distortion performance is obtained by encoding the image directly in the highest possible bit depth. This is explained by the fact that the quantization applied before the HEVC compression is a lossy operation. At low bitrates, the loss caused by the HEVC encoding is predominant. That is why the rate distortion performance does not depend too much on the target bitdepth of the quantization at low bitrates. However, this is no longer true for higher bitrates, since the error of the prior quantization becomes predominant over the error caused by HEVC.

For enabling very high quality encoding while using a low bitdepth encoder, we have seen that the best solution was to perform the adaptive quantization by blocks. However, this method gives bad results at low and medium bitrates because of the loss of spatial coherency and the large overhead induced by the encoding of the minimum and maximum values of each block. Conversely, the best solution for low bitrate encoding with a low bitdepth encoder is to perform the adaptive quantization only by GOPs which preserves spatial and temporal coherency.

3.3 Rate-distortion optimized tone curve

In the previous section, the study was limited to a fairly simple uniform quantization scheme which linearly maps the HDR data from the range $[x_{min}, x_{max}]$ to the range $[0, 2^n - 1]$ for a target bitdepth n. In this section, we will see how to improve the overall rate distortion performance by applying a non-uniform quantization scheme.

In the literature, several attempts have been made at reducing the bit-depth using a global TMO which consists in applying a non linear curve (called compressor), followed by uniform quantization. Applying the inverse quantization and the inverse curve (called expander) to the encoded and decoded LDR image expands the data to its original dynamic. This approach, often referred to as companding is equivalent to non-uniform quantization. As an example, the method in [57] iteratively optimizes the parameters of the photographic TMO from [41] in order to minimize the HDR reconstruction error. In [77], an approximation of the data distribution based on Gaussian mixture models (GMM) is used to build compressor and expander curves that approach the results obtained by the iterative Lloyd-Max algorithm [78]. The latter algorithm is used, for example in [79], in the context of HDR compression. It aims at finding the optimal quantizer in terms of distortion, but does not consider further encoding of the quantized image. In [55], Mai et al. define a segment based curve that minimizes the data loss caused by both the tone mapping and the encoder error. However, all these methods only focus on the distortion without taking into account the rate of the encoded LDR image. In [80], Chou et al. added an entropy constraint to the Lloyd-Max algorithm to take the rate into account, but this method remains iterative and computationally expensive. Moreover, they assume that the quantized image is only entropy encoded without loss.

In our method, both rate and distortion are optimized. The image is first tone mapped with a global invertible TMO and then encoded with HEVC. Based on a statistical model of the complete HDR compression scheme and assumptions on the rate of the encoded LDR image, a closed form solution is found for the optimal tone curve in the sense of rate distortion performance.

The rest of the section is organized as follows. In subsection 3.3.1, the statistical model of the complete compression scheme is presented and the problem to solve is posed. A closed form solution is given in subsection 3.3.2. Then, the implementation of our optimized tone mapping is described in subsection 3.3.3.

3.3.1 Statistical model

The problem consists in minimizing an objective function of the form $D + \lambda_r.R$, where D is the total distortion after reconstruction of the HDR image, R is the rate of the encoded LDR image and λ_r is a Lagrangian multiplier that is adjusted to obtain the optimal rate distortion performance.

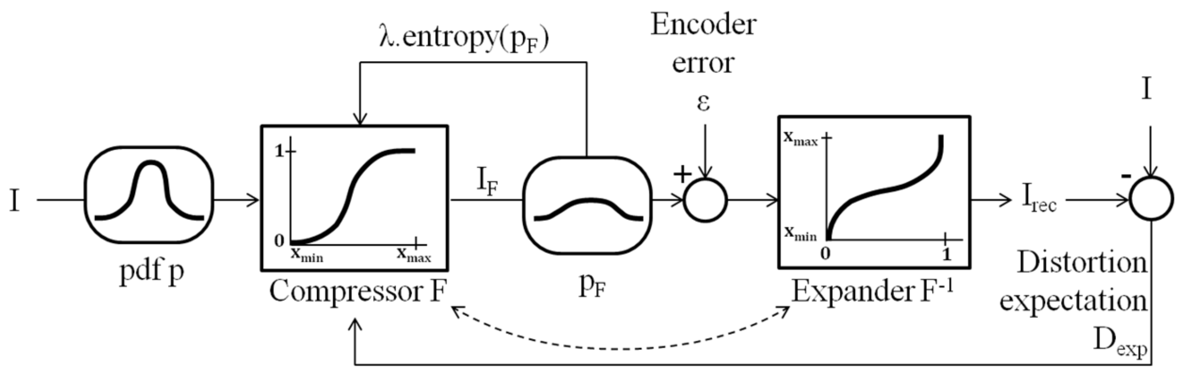


Figure 3.6: Statistical model of the HDR compression scheme.

In order to find a closed form solution to this problem, we define a statistical model of our compression scheme. It is illustrated in figure 3.6. In this model, we consider that the pixels have real values (not necessarily integers). The input image I has a probability density function (p.d.f.) p and its minimum and maximum pixel values are respectively x_{min} and x_{max}.

First, a function F that we call compressor function is applied to the pixel values. F is defined on the interval $[x_{min}, x_{max}]$ such that $F(x_{min}) = 0$ and $F(x_{max}) = 1$. We assume that F is a continuous and strictly monotonous function. These properties ensure that F has an inverse function F^{-1} (called expander). After applying the function F, no data is lost since F is mathematically invertible. We call I_F the obtained image and p_F its probability density function. Note that in a real implementation, the operation of tone mapping with a curve is equivalent to applying a compressor function such as F followed by uniform quantization.

Then a random variable ($\mathcal{E}$) is added to each pixel value. It models both the quantization error from the tone mapping and the encoder error. Here we suppose that the distribution of these random variables does not depend on the position or the value of the pixel. It has zero mean, and a variance σ^2.

Finally the expander F^{-1} is applied to reconstruct the image I_{rec}. Based on this model, the total mean square error (MSE) of the reconstructed image can be estimated by its statistical expectation value D_{exp}. Given an image I and an encoder with fixed quality settings (e.g. fixed QP), we make the assumption that the rate R of the encoded image is proportional to the entropy of I_F. Thus, minimizing $D_{exp} + \lambda_r.R$ is equivalent to minimizing $D_{exp} + \lambda.entropy(I_F)$, where λ is another Lagrangian multiplier.

3.3.2 Closed Form Solution

3.3.2.1 Expression of the cost

Considering the mean square error as distortion metric, for an input value x in the original image, the distortion $D(x)$ is given by :

$$D(x) = (x - F^{-1}(F(x) + \mathcal{E}))^2$$

For small values of $\mathcal{E}$, the following approximation can be done :

$$\begin{aligned}
D(x) &\approx (x - F^{-1}(F(x)) + \mathcal{E} \cdot F^{-1\prime}(F(x)))^2 \\
D(x) &\approx (\mathcal{E} \cdot F^{-1\prime}(F(x)))^2 \\
D(x) &\approx \frac{\mathcal{E}^2}{F'(x)^2}
\end{aligned}$$

Thus, the expected distortion for the value x is :

$$E(D(x)) = \frac{var(\mathcal{E})}{F'(x)^2} = \frac{\sigma^2}{F'(x)^2}$$

And the total mean distortion is :

$$D_{exp} = \int_{x_{min}}^{x_{max}} p(x) \cdot E(D(x)) \, \mathrm{d}x = \int_{x_{min}}^{x_{max}} p(x) \cdot \frac{\sigma^2}{F'(x)^2} \, \mathrm{d}x \tag{3.13}$$

The entropy H_F of the tone mapped image I_F after applying the compressor function F is :

$$H_F = -\int_0^1 p_F(y) \cdot log_2(p_F(y)) \, \mathrm{d}y$$

Given that F is strictly increasing, we have $p_F(y) = \dfrac{p(F^{-1}(y))}{F'(F^{-1}(y))}$. Thus,

$$H_F = -\int_0^1 \frac{p(F^{-1}(y))}{F'(F^{-1}(y))} \cdot log_2\left(\frac{p(F^{-1}(y))}{F'(F^{-1}(y))}\right) \mathrm{d}y$$

Applying the substitution $y = F(x)$, we obtain :

$$\begin{aligned}
H_F &= -\int_{x_{min}}^{x_{max}} \frac{p(x)}{F'(x)} \cdot log_2\left(\frac{p(x)}{F'(x)}\right) \cdot F'(x) \, \mathrm{d}x \\
H_F &= -\int_{x_{min}}^{x_{max}} p(x) \cdot log_2\left(p(x)\right) \, \mathrm{d}x + \int_{x_{min}}^{x_{max}} p(x) \cdot log_2\left(F'(x)\right) \, \mathrm{d}x
\end{aligned}$$

As a result,

$$H_F = H + \int_{x_{min}}^{x_{max}} p(x) \cdot log_2\left(F'(x)\right) \, \mathrm{d}x \tag{3.14}$$

where H is the entropy of the original image I. The expression of the total cost to minimize is then :

$$Cost = \int_{x_{min}}^{x_{max}} \left(\frac{\sigma^2 \cdot p(x)}{F'(x)^2} + \lambda \cdot p(x) \cdot log_2(F'(x)) \right) \mathrm{d}x + \lambda \cdot H \tag{3.15}$$

3.3.2.2 Resolution

Applying the Euler Lagrange equation gives the following necessary condition for the optimal function F with respect to the cost:

$$\frac{\mathrm{d}}{\mathrm{d}x} \left[\frac{-2 \cdot \sigma^2 \cdot p(x)}{F'(x)^3} + \frac{\lambda \cdot p(x)}{\ln(2) \cdot F'(x)} \right] = 0 \tag{3.16}$$

In our problem, the function F must be strictly increasing and such that $F(x_{min}) = 0$ and $F(x_{max}) = 1$. Therefore, in addition to the condition (3.16), the derivative F' must also satisfy the two conditions:

$$\begin{cases} \forall x \in [x_{min}, x_{max}],\ F'(x) > 0 & \text{(3.17a)} \\ \displaystyle\int_{x_{min}}^{x_{max}} F'(x)\,\mathrm{d}x = 1 & \text{(3.17b)} \end{cases}$$

Let $\lambda' = \dfrac{\lambda}{2\ln(2) \cdot \sigma^2}$. The Euler Lagrange condition (3.16) can be simplified by removing the differential operator and introducing a constant:

$$(3.16) \Leftrightarrow \exists c \in \mathbb{R},\ \forall x \in [x_{min}, x_{max}],\ \frac{-p(x)}{F'(x)^3} + \lambda' \cdot \frac{p(x)}{F'(x)} = c$$

- By taking $c = 0$, we find a first solution $F'(x) = \sqrt{\frac{1}{\lambda'}} > 0$. However, the condition (3.17b) is only satisfied in the particular case where $\lambda' = (x_{max} - x_{min})^2$. In that case, the optimal compressor F^* is a linear function:

$$F^*(x) = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.18}$$

  In order to find the optimal solution in the general case, we must solve the following equation with respect to $F'(x)$:

$$\frac{-p(x)}{F'(x)^3} + \lambda' \cdot \frac{p(x)}{F'(x)} = c \tag{3.19}$$

  with $c \neq 0$. Because of the condition (3.17a), only the positive solutions must be considered. (3.19) can be rewritten in the form of a cubic equation:

$$F'(x)^3 + a \cdot F'(x)^2 - b = 0 \tag{3.20}$$

  where $a = -\lambda' \cdot \dfrac{p(x)}{c}$ and $b = -\dfrac{p(x)}{c}$.

- Let us consider the case where $c < 0$. We know that $\lambda' > 0$ and $p(x) > 0$. Therefore, $a > 0$ and $b > 0$. We have shown in appendix A.1 that in these conditions, the cubic equation (3.20) has exactly one positive solution:

$$F'(x) = \begin{cases} \sqrt[3]{m + \sqrt{m^2 - n^2}} + \sqrt[3]{m - \sqrt{m^2 - n^2}} - \sqrt[3]{n}, & \text{if } m > n \\ \dfrac{a}{3} \cdot \left(2\cos\left(\dfrac{\arccos\left(\frac{27b}{2a^3} - 1\right)}{3} \right) - 1 \right), & \text{otherwise} \end{cases} \tag{3.21}$$

where $\begin{cases} a = -\lambda' \cdot \dfrac{p(x)}{c} = -\dfrac{\lambda}{2\ln(2)\cdot\sigma^2} \cdot \dfrac{p(x)}{c}, & b = -\dfrac{p(x)}{c} \\ m = \dfrac{b}{2} - \dfrac{a^3}{27}, & n = \dfrac{a^3}{27} \end{cases}$

Furthermore, it can be proven that there exists a value of c such that F' satisfies the condition (3.17b) only if $\lambda' \in]0, (x_{max} - x_{min})^2[$. This is shown in Annex A.2. As a result, if $\lambda' \geq (x_{max} - x_{min})^2$, equation (3.21) does not correspond to an actual solution of the problem.

- For $c > 0$, both a and b are negative. In this case, we determined in appendix A.1 that the cubic equation (3.20) only has positive roots if $b \geq \dfrac{4a^3}{27}$. By replacing a and b by their expressions, we find that this condition is equivalent to $\lambda' > \sqrt[3]{\dfrac{27c^2}{4p(x)^2}}$ for all x. As a result, for λ' sufficiently large, the equation (3.19) has positive solutions F' when c is negative. The integral F of the resulting functions F' then satisfies the Euler-Lagrange condition. However, this is a necessary but not sufficient condition for minimizing the cost in equation (3.15). Here, we will admit that none of those solutions found with $c > 0$ correspond to the global minimum for the cost. They are only stationary points.

As a result, three cases can be distinguished:

- if $\lambda' > (x_{max} - x_{min})^2$, there is no function that minimize the cost in (3.15). This is discussed in the next subsection.
- If $\lambda' = (x_{max} - x_{min})^2$, the optimal function is given by equation (3.18).
- If $\lambda' < (x_{max} - x_{min})^2$, the optimal function is given by:

$$F^*(x) = \int_{x_{min}}^{x} F^{*\prime}(t)\,\mathrm{d}t \tag{3.22}$$

where $F^{*\prime}$ is defined by equation (3.21) in which c is such that $F^*(x_{max}) = 1$.

However, in the last case, although we know that c exists, we cannot determine its analytical expression. To solve this problem, let us introduce a new parameter λ_0 and define the function S'_{λ_0} as the unique positive solution to the following equation:

$$\forall x \in [x_{min}, x_{max}], \frac{-p(x)}{S'_{\lambda_0}(x)^3} + \lambda_0 \cdot \frac{p(x)}{S'_{\lambda_0}(x)} = -1 \tag{3.23}$$

This is the same equation as (3.19) where λ' was replaced by the new parameter λ_0, and c was fixed arbitrarily to -1. A negative value of c was chosen here so that the equation has a positive solution as stated in equation (3.21). Therefore, similarly to equation (3.21), we have:

$$S'_{\lambda_0}(x) = \begin{cases} \sqrt[3]{m+\sqrt{m^2-n^2}} + \sqrt[3]{m-\sqrt{m^2-n^2}} - \sqrt[3]{n}, & \text{if } m > n \\ \frac{a}{3} \cdot \left(2\cos\left(\frac{\arccos\left(\frac{27b}{2a^3}-1\right)}{3} \right) - 1 \right), & \text{otherwise} \end{cases} \tag{3.24}$$

$$\text{where } a = \lambda_0 \cdot p(x), \qquad b = p(x), \qquad m = \frac{b}{2} - \frac{a^3}{27} \quad \text{and} \quad n = \frac{a^3}{27}$$

By definition, S'_{λ_0} is positive and satisfies the Euler Lagrange condition (3.16). Now, let us define F by:

$$F(x) = \frac{S_{\lambda_0}(x)}{S_{\lambda_0}(x_{max})} \tag{3.25}$$

where $S_{\lambda_0}(x) = \int_{x_{min}}^{x} S'_{\lambda_0}(t)\,\mathrm{d}t$. We want to prove that F is the optimal function for a given value of λ'.

- $F'(x) = \frac{S'_{\lambda_0}(x)}{S_{\lambda_0}(x_{max})} > 0$. Therefore, condition (3.17a) is satisfied.
- By definition, $F(x_{min}) = 0$ and $F(x_{max}) = 1$. Therefore, condition (3.17b) is also satisfied.
- Moreover, from equation (3.25) we know that $S'_{\lambda_0}(x) = F'(x) \cdot S_{\lambda_0}(x_{max})$. By replacing $S'_{\lambda_0}(x)$ by this expression in equation (3.23), we obtain:

$$\frac{-p(x)}{F'(x)^3} + \lambda_0 \cdot S_{\lambda_0}(x_{max})^2 \cdot \frac{p(x)}{F'(x)} = -S_{\lambda_0}(x_{max})^3 \tag{3.26}$$

  Therefore, if $\lambda' = \lambda_0 \cdot S_{\lambda_0}(x_{max})^2$, F' solves the equation (3.19) with $c = -S_{\lambda_0}(x_{max})^3 < 0$. The Euler Lagrange condition (3.16) is thus satisfied. We show in appendix A.3 that for any λ' in $[0, (x_{max} - x_{min})^2[$, there is a unique value λ_0^* such that $\lambda' = \lambda_0^* \cdot S_{\lambda_0^*}(x_{max})^2$. The consequence is that for any value of the parameter λ', and thus for any couple of parameters (λ, σ) in the original formulation, if a solution to the minimization problem exists, we can find a value λ_0^* such that the function F defined by the equations (3.24) and (3.25) is optimal. Note that as λ' approaches $(x_{max} - x_{min})^2$, the value of λ_0^* tends towards infinity. And in that case, the optimal compressor function F^* tends towards the linear solution in equation (3.18) found in the particular case where $\lambda' = (x_{max} - x_{min})^2$.

3.3.2.3 Discussion

We have found that, in the particular case where $\lambda' > (x_{max} - x_{min})^2$ (i.e. $\lambda > 2\ln(2) \cdot \sigma^2 \cdot (x_{max} - x_{min})^2$), there is no solution to the equation. However in the context of rate distortion optimization, we do not need to solve the problem for any value of the Lagrangian parameter λ. Instead, the value of λ should be chosen in order to optimize the rate-distortion function. In our statistical model, the rate distortion function would represent the entropy H_F as a function of the distortion expectation D_{exp}. So, ideally, knowing the p.d.f p of the source signal, the variance σ^2 of the error ϵ, and given a target distortion D^t_{exp}, the parameter λ should be chosen to minimize H_F with the constraint that $D_{exp} \leq D^t_{exp}$.

Now, if the equation has no solution (i.e. for $\lambda' > (x_{max} - x_{min})^2$), it means that for any function F that satisfies the conditions (3.17a) and (3.17b), we can find another function which also satisfies those conditions and which gives a lower cost. This is only possible because the set of functions satisfying (3.17a) is an open set. The cost can thus become arbitrarily small by choosing a compressor function F arbitrarily close to the boundary of that open set : in other words, a function F such that for at least one value x, $F'(x)$ is arbitrarily close to zero. In that case, the distortion expectation D_{exp} is arbitrarily large and the entropy H_F is arbitrarily large and negative (the entropy can be negative for a continuous probability density function). Therefore, in the rate distortion optimization problem, given a finite value D^t_{exp} of the target distortion expectation, the constraint $D_{exp} \leq D^t_{exp}$ cannot be satisfied. As a result, for any finite value of the target distortion expectation D^t_{exp}, the optimal value of λ is such that $\lambda \leq 2\ln(2) \cdot \sigma^2 \cdot (x_{max} - x_{min})^2$ since for higher values of λ, the distortion expectation D_{exp} would become arbitrarily large and thus greater than D^t_{exp}.

For this reason, the case $\lambda > 2\ln(2) \cdot \sigma^2 \cdot (x_{max} - x_{min})^2$ is not considered in what follows. Therefore, it can be assumed that there is always a value $\lambda_0^* \geq 0$ such that the function defined in equation (3.25) is the optimal compressor function.

3.3.3 Implementation

In this subsection, we describe an encoding method for HDR images using the theoretical results described previously.

Our complete encoding scheme is depicted in figure 3.7. The input image is given in a floating point format and represents luminance values at each pixel. Since the human perception of luminance is not linear, a perceptual curve is first applied to the image before performing any compression. For the sake of simplicity in the implementation, the resulting perceptually encoded image is first converted to high bit depth integers (by uniform quantization). For a sufficiently high bit-depth, the loss caused by this operation is not visible. From the perceptually encoded image, the distribution of the pixels' values is estimated as explained in subsection 3.3.3.1. Given a lagrangian parameter λ_0, the tone mapping curve is computed in the form of a Lookup Table (LUT) as described in subsection 3.3.3.2. This LUT is applied to the image in order to obtain a low bit depth LDR image suitable for a typical encoder. An HEVC encoding is then performed with a given quantization parameter (QP). We explain in subsection 3.3.3.3 that the optimal lagrangian parameter λ_0^* for the tone mapping curve can be well approximated as a function of the QP parameter that will be used in the encoder. In addition to the bitstream of the tone mapped image, the estimated p.d.f. must also

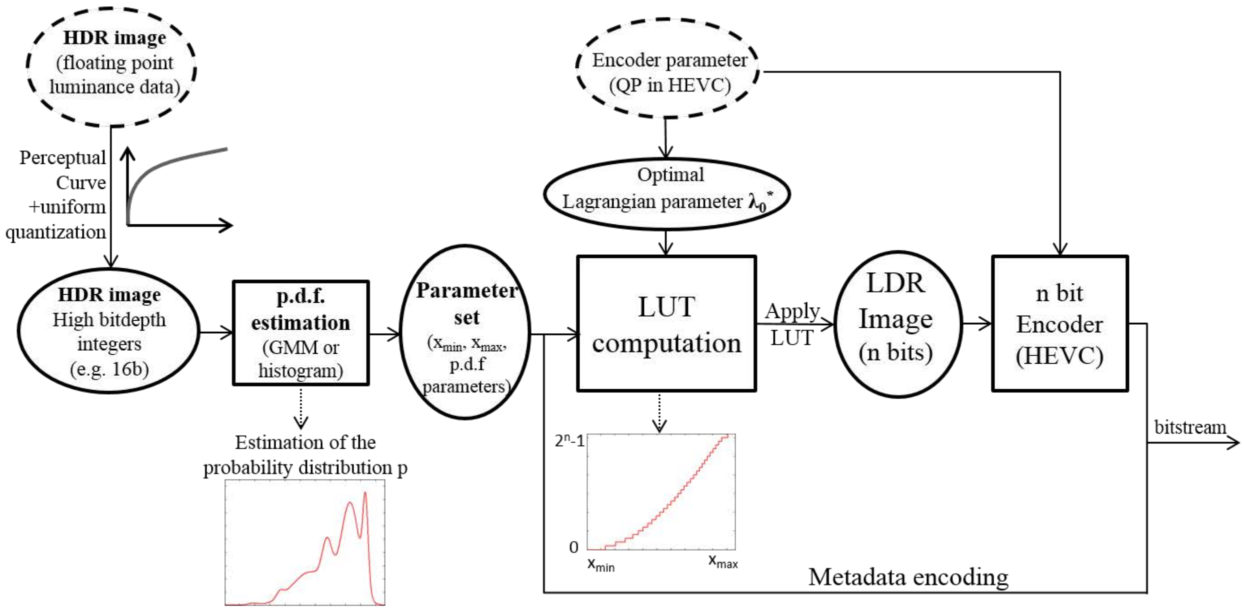


Figure 3.7: Overview of the HDR compression scheme based on the Rate Distortion optimized tone mapping.

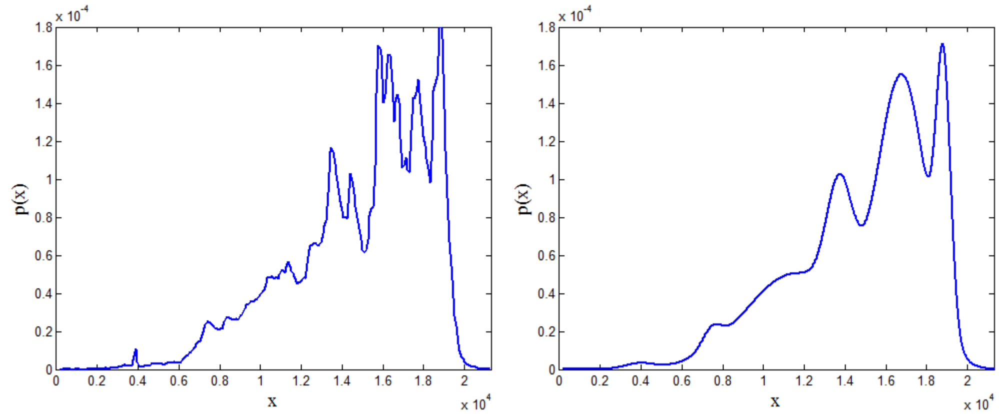
be sent to the decoder as metadata so that it can perform the inverse tone mapping. The lagrangian parameter does not need to be transmitted since it can be directly found from the QP which is already known by the HEVC decoder.

3.3.3.1 Model of the probability density function

As shown in subsection 3.3.2.2, the optimal tone mapping curve depends on the probability density function p of the pixel values in the original image. It can be easily estimated by computing the histogram of the image. However, since the decoder must be able to compute the inverse tone mapping curve, the histogram must be transmitted. In order to transmit only a few side information to the decoder, the histogram should thus contain a limited number of bins. In our implementation, the interval $[x_{min}, x_{max}]$ was divided into k intervals (i.e. bins) of equal size $\frac{x_{max} - x_{min}}{k}$. At each value x in the bin $B(x)$, the p.d.f. of an image I is then expressed as

$$p(x) = \frac{\sum_{i=1}^{N} \delta(I(i), B(x))}{N} \cdot \frac{k}{x_{max} - x_{min}} \tag{3.27}$$

where N is the number of pixels in the image, $I(i)$ is the value of the pixel i, and $\delta(I(i), B(x))$ is equal to 1 if $I(i) \in B(x)$ and 0 otherwise. Since the function p is constant inside each bin, only k values in addition to the values x_{min} and x_{max} are sufficient to describe the p.d.f. and need to be transmitted. In our implementation, the number of bins k was fixed to 250.



(a)histogram ($k = 250$) (b)GMM ($m = 6$)

Figure 3.8: example of p.d.f. estimation for the image *Marché* with both the histogram and the GMM method.

In order to reduce even further the amount of side information, a second version was implemented where the p.d.f. is modelled by a Gaussian mixture model (GMM) instead of the histogram. A GMM is a weighted sum of several Gaussians. The model parameters are the variance v_j, the mean value μ_j and the weight α_j of each Gaussian j in the mixture model. The p.d.f. is thus given by

$$p(x) = \sum_{j=1}^{m} \frac{\alpha_j}{\sqrt{2\pi \cdot v_j}} \cdot exp\left(\frac{(x-\mu_j)^2}{2v_j}\right) \tag{3.28}$$

where m is the number of Gaussians used in the model. The Expectation Maximization algorithm (EM) from [81] was used in order to determine the GMM that best fits the histogram of the image. In our implementation, the number m of Gaussian functions was fixed to 6. We did not observe significant changes in the probability density function by increasing this value.

For both methods, an example of estimated probability density functions is shown in figure 3.8.

3.3.3.2 Computation of a Lookup Table

As it is explained in section 3.3.2, the optimal compressor function F^* can be computed from equations (3.24) and (3.25). Given a value of λ_0, we first tabulate the function S'_{λ_0} of equation (3.24) at every integer value x from x_{min} to x_{max}. The function S_{λ_0} can then be approximated by numerical integration. The final tone curve is then computed as a Lookup Table (LUT) from the following equation:

$$LUT(x) = \left[(2^n - 1) \cdot \frac{S_{\lambda_0}(x)}{S_{\lambda_0}(x_{max})}\right] \tag{3.29}$$

where n is the bitdepth of the LDR image and the brackets represent the rounding operation. The factor $2^n - 1$ and the rounding operation are necessary for the quantization to n bits integers but it has no impact on the overall shape of the tone curve.

The tone mapping operation only consists in applying this LUT to every pixel of the perceptually encoded HDR image. The image obtained is then compressed with an LDR encoder. The parameters used for the construction of the tone mapping curve (i.e. x_{min}, x_{max} and the p.d.f. parameters) must also be transmitted as side information.

On the decoder side, the first step is to decode the LDR image and the model parameters. Knowing the parameters, the equations (3.27) (or (3.28) for the GMM version) and (3.24) can be computed. From the tabulated function S'_{λ_0}, the inverse of the integral can be computed numerically to obtain an inverse tone mapping LUT. Finally, the inverse LUT is applied to the decoded LDR image to reconstruct the HDR image.

3.3.3.3 Determination of the Lagrangian Parameter

In the previous section, all the computations were based on the parameter λ_0. This value must be optimized with respect to the rate distortion performance of the complete HDR compression scheme. For a given encoder (e.g. HEVC [48], MPEG-4 H264/AVC [49], JPEG2000 [82]), the optimal value of λ_0 depends on the quantization performed internally by the encoder. In practice however, the internal quantization step is generally not specified directly in the encoder's configuration, but it is computed from a quality setting. In HEVC for instance, for a given bitdepth n at the input of the encoder, the quantization step Q_n is a function of the integer valued QP setting:

$$Q_n = X \cdot 2^{\frac{QP-6\cdot(n-8)}{6}} \tag{3.30}$$

where X is a constant factor.

In our experiments, the 8 bit version of HEVC was used for encoding the image. Therefore, we had to determine a law giving the optimal value λ_0^* as a function of the QP parameter. For that purpose, several images were encoded with our method over a large range of QPs and λ_0 values. For each image, at each QP, the encoding was performed several times by varying the value of λ_0. Given a QP value, the Rate Distortion (RD) point obtained with the optimal λ_0 is on the convex hull of the set of all the achievable RD points as illustrated by figure 3.9. Thus, the optimal Lagrangian parameter could be determined by selecting the best RD point at each QP.

The operation was performed for several images and an exponential law was fitted to the experimental data to derive a formula for λ_0^*. We observed that for very high bitrates (i.e. low QP), the rate distortion curves have an inflection point. A different formula was then obtained for low QP values to account for this phenomenon. The final formula is then:

$$\lambda_0^* = \begin{cases} 2^{\frac{-0.357\cdot QP_n^2+16.628\cdot QP_n+34.388}{QP_n+5.95}}, & \text{if } QP_n \leq 10 \\ 2^{0.412\cdot QP_n+5.991}, & \text{otherwise} \end{cases} \tag{3.31}$$

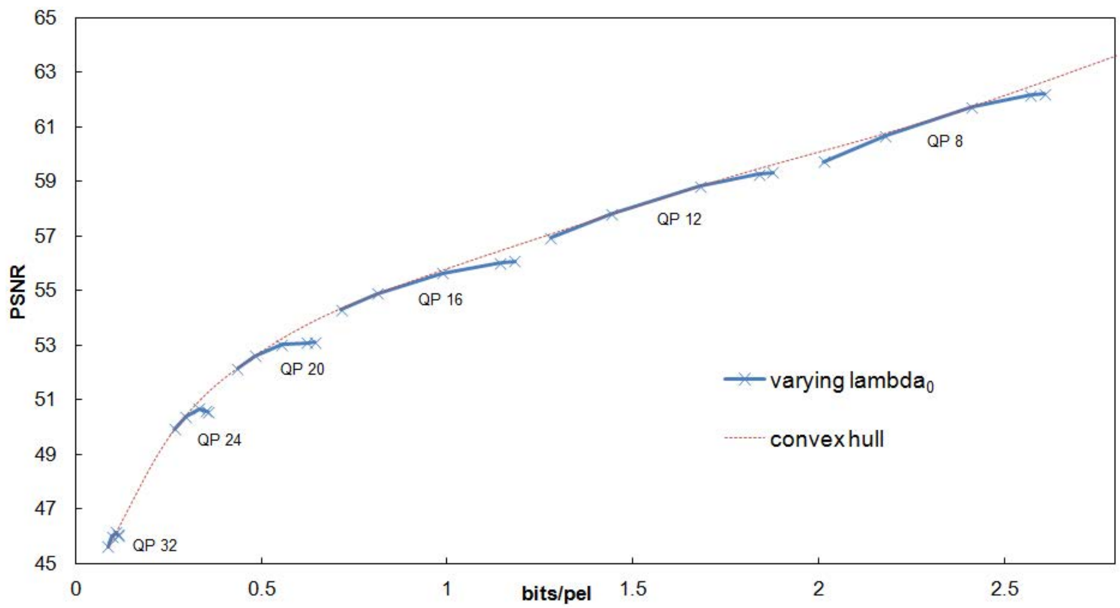


Figure 3.9: Lagrangian parameter optimization

where $QP_n = QP + 6 \cdot (n - 8)$ and n is the input bitdepth of the HEVC encoder (i.e. the target bitdepth of our tone mapping operator). Although the experiment was performed only with $n = 8$, the formula was generalized for any bitdepth by taking into account the fact that the internal quantization step in HEVC is dependent on the bitdepth as shown in equation (3.30). The term $6 \cdot (n - 8)$ was then added to the QP in order to compensate for the adjustment of the quantization performed in HEVC in the case $n > 8$.

Note that in theory, the optimal Lagrangian parameter may not only depend on the encoder's quantization but also on the distribution of the pixels' values in the image (i.e. the histogram). For the images tested, however, we have observed that the same law gave nearly optimal results although they had rather different histograms.

3.3.4 Experimental Results

For the experiments, only a luma channel was tone mapped and encoded using the HEVC standard. The joint quantization of luma and chroma components is not considered here.

The input images were originally in the OpenEXR half float RGB format [33]. For this experiment, the approximate logarithmic encoding described in section 3.1 was performed to convert the RGB floating point data to 15 bit integers. For the sake of simplicity, this encoding is considered as perceptually uniform in what follows. The obtained log RGB values were then converted to YUV using the BT.709 primaries. Only the luma channel Y was compressed with our method by tone mapping to 8 bits and encoding with HEVC using the YUV 4:0:0 chroma format and 8 bit input bitdepth.

(a) Péniches (b) Mongolfière

(c) Cracheur (d) Marché

(e) Désert (f) Tunnel

Figure 3.10: Tone mapped version of the HDR images used for our experiment. For the tone mapping, our algorithm was used with the parameter $\lambda_0 = 0$.

Our distortion measure is the PSNR computed on the 15 bit integer luma channel reconstructed by applying the inverse tone curve to the 8 bit decoded image. Thus, the peak signal value used in the PSNR formula is 32767. Note that this 15 bit data is proportional to the logarithm of luminance. Therefore, we obtain a better visual indicator than a PSNR applied directly to luminance values.

The results are shown for the input images *Péniches*, *Mongolfière*, *Cracheur*, *Marché*, *Désert* and *Tunnel*. Figure 3.10 shows a tone mapped version of those images with our algorithm for $\lambda_0 = 0$. Our method was compared to the distortion optimized tone mapping developed by Mai et al. [55]. In our implementation their curve is applied to the log encoded 15 bit data. Therefore, their logarithm function is not applied again. Similarly to the histogram based version of our method, the histogram bin size in Mai et al's tone mapping was chosen in order to have 250 segments between the minimum and the maximum pixels values x_{min} and x_{max}. Note that in these conditions, Mai et al's method is identical to our histogram based version for the particular case where $\lambda_0 = 0$. This is in accordance with the fact that only the distortion is taken into account in [55] which corresponds exactly to the case $\lambda_0 = 0$. In addition to Mai et al's method, we also compared our optimized tone curve to a linear mapping of the HDR values from the range $[x_{min}, x_{max}]$ to the 8 bit range $[0, 255]$, which corresponds to the frame-wise version of the quantization algorithm in section 3.2. This is equivalent to the extreme case $\lambda_0 = +\infty$.

The resulting rate distortion curves are shown in figure 3.11. The curves were generated by varying the QP value from 0 to 32 in the HEVC encoder configuration. The transmission cost of the additional parameters was not included in the bitrate. However, it can be considered as negligible. For instance, in the histogram based version of our method, 250 histogram values must be encoded in addition to x_{min} and x_{max}. Assuming that the histogram values are encoded with a precision of 16 bits (which is sufficient in practice) and given that the dimension of the encoded images is 1920x1080, the overhead is still less than 0.002 bits per pixel.

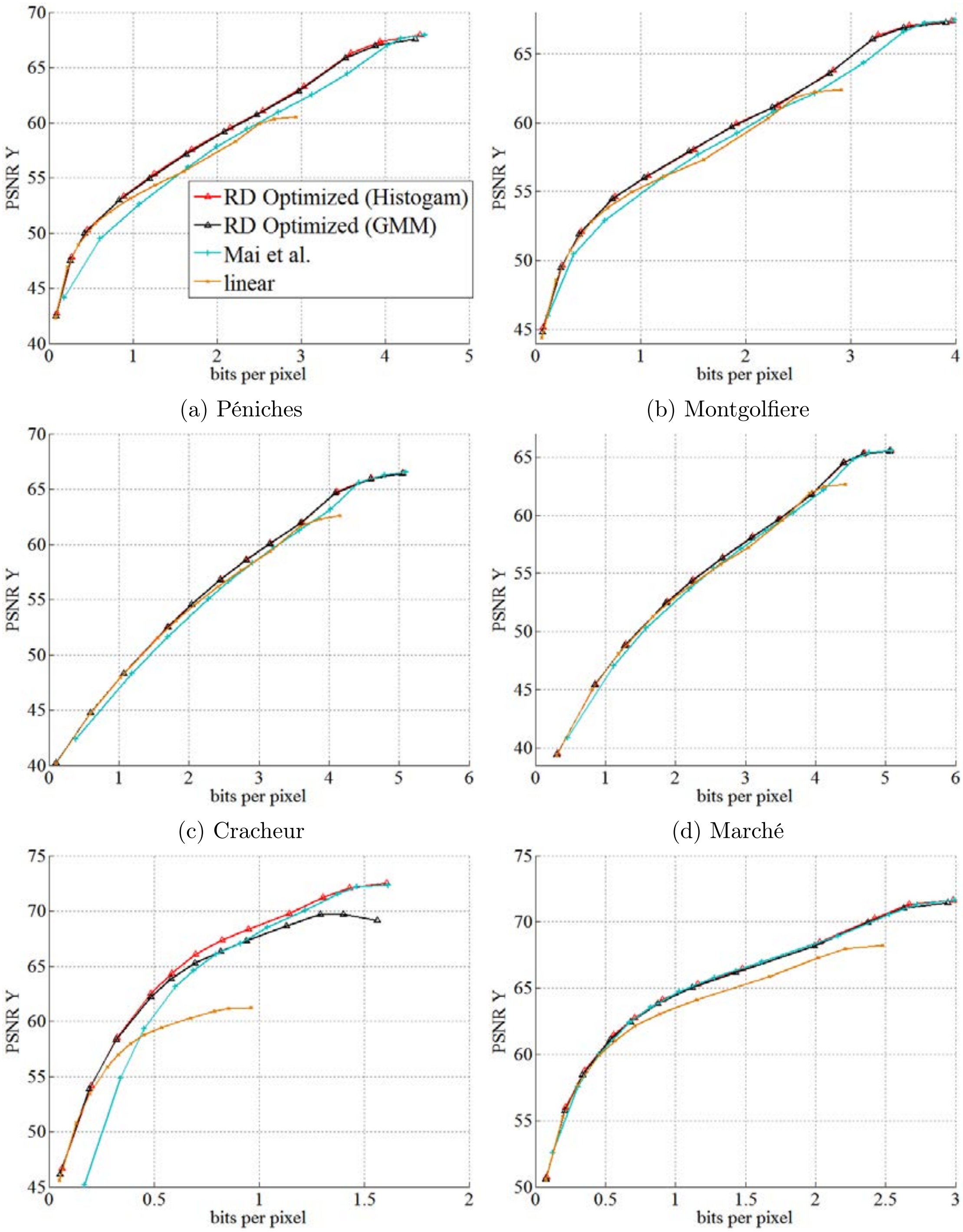
Table 3.2 shows the bit rate savings obtained in comparison to Mai et al's method using Bjontegaard metric [76] for two different QP ranges and for both versions of our method. High bitrates are computed for QPs from 0 to 16 and low bit rates are computed for QPs from 16 to 32. The bit rate savings with respect to the linear method are given in table 3.3

	with histogram		with GMM	
Image	Low bitrates	High bitrates	Low bitrates	High bitrates
Péniches	-34.6%	-9.12%	-34.4%	-8.24%
Mongolfière	-18.0%	-6.85%	-17.9%	-6.90%
Cracheur	-11.4%	-4.40%	-11.4%	-4.31%
Marché	-7.11%	-2.41%	-7.14%	-2.37%
Désert	-40.7%	-8.88%	-42.2%	+0.87%
Tunnel	-4.81%	-0.65%	-4.25%	-1.36%
Average	**-19.4%**	**-5.39%**	**-19.5%**	**-3.72%**

Table 3.2: Bjontegaard rate gains of both versions of our rate distortion optimized TMO with respect to Mai et al's TMO [55]. QPs from 16 to 32 are used for low bitrates and QPs from 0 to 16 are used for high bitrates.

As expected, for high bitrates (i.e. at low QP values and thus low λ_0), the results of our method are close to those of Mai et al. For lower bit rates, our method gives better results thanks to the rate distortion optimization technique. However as the bitrate decreases, the rate distortion performance of the linear mapping method gets closer to that of our rate distortion optimized method. This was also expected since at low bitrates (i.e. high QPs), our optimal tone curve tends to become linear.

The image *Marché* gives the lowest gains. This can be explained by the fact that the histogram of this image is more "uniform" than those of the other images. In this case, the tone curves obtained by all the methods are similar and close to linear. In contrast, the image *Désert* which has a highly non-uniform histogram shows the highest gains. For high bitrates, the gains with respect to the linear method could not be computed because the rate distortion curves are too different.



(a) Péniches (b) Montgolfiere

(c) Cracheur (d) Marché

(e) Désert (f)Tunnel

Figure 3.11: Rate-Distortion results for each HDR test images.

	with Histogram		with GMM	
Image	Low bitrates	High bitrates	Low bitrates	High bitrates
Péniches	-2.78%	-15.1%	-2.14%	-14.0%
Mongolfière	-1.64%	-10.9%	-1.62%	-11.1%
Cracheur	-0.54%	-4.16%	-0.58%	-4.05%
Marché	-0.21%	-3.18%	-0.22%	-3.09%
Désert	-4.40%	-	-4.37%	-
Tunnel	-1.79%	-18.3%	-1.16%	-16.2%
Average	**-1.89%**	**-10.3%**	**-1.68%**	**-9.69%**

Table 3.3: Bjontegaard rate gains of both versions of our rate distortion optimized TMO with respect to the linear TMO. QPs from 16 to 32 are used for low bitrates and QPs from 0 to 16 are used for high bitrates. A hyphen indicates that the gain could not be computed.

Note that in most cases, similar results are obtained with the histogram and the GMM based versions of our method. However, it can be observed that the histogram based version gives slightly better results. This is explained by the fact that a histogram models the distribution of the pixels' values more accurately than a GMM which contains only a few parameters. The difference is particualrly noticeable for the image *Désert* because the histogram has very narrow peaks that are not well represented by the GMM model. It explains why the GMM method performs poorly at high bitrates for that image. Another advantage of the histogram method is that it is faster to compute compared to the GMM which requires a time consuming Expectation Maximization (EM) operation for determining the gaussian parameters.

3.4 Conclusion

In this chapter, we have explored several ways to encode HDR images using a low bitdepth encoder with a particular attention to the overall compression performance. In a preliminary work, we have considered a regional adaptation of a uniform quantization scheme where the regions could be either blocks, frames or GOPs. While a block-wise adaptation enables a reduction of the bitdepth with very low distortion, it is only advantageous in conditions of high bitrate and near lossless encoding because of the high transmission cost of side information and the loss of both temporal and spatial coherency. Larger regions (i.e. frames or GOPs) for the adaptation are better suited to lower bitrate applications.

In a second part, we have studied a different level of adaptation by performing non-uniform quantization in a rate distortion optimized manner. Given the image distribution and the quality setting of the encoder (i.e. QP in HEVC), we have determined the optimal tone curve (or compressor function) that should be applied to the image before the encoding in a low bitdepth integer format. Although a linear tone curve (and thus a uniform quantization) turns out to be close to optimal at low bitrates, substantial coding gains were obtained at high bitrates with our non-uniform quantization compared to uniform quantization. In this second work, however, only a luma channel was encoded. Although the same method could be applied independently to each component, this approach may not be optimal because of the large differences in the data distribution of luma and chroma. An independent quantization of the components might then result in allocating too much bitrate for chroma. The joint quantization of luma and chroma components remains a question for future research.

The methods presented in this chapter only encode a single LDR layer. Such schemes impose restrictions on the tone mapping operation since this step results in a loss of information that degrades the quality of the finally reconstructed HDR image. Scalable methods remove this limitation and enable more artistic freedom by encoding a second layer containing that missing information. Such methods are developed in the next chapters.

Chapter 4

Local Inter-Layer prediction for scalable schemes

In the previous chapter, only a LDR layer was transmitted along with a few metadata indicating how to invert the tone mapping process and recover the original dynamic of the image. We will now focus on scalable HDR encoding methods where the base layer is a low dynamic range version of the image that may have been generated by an arbitrary Tone Mapping Operator (TMO). No restriction is imposed on the TMO, which can be either global or local, so as to fully respect the artistic intent of the producer.

We have presented in chapter 2 several HDR compression schemes with two layers. While the methods proposed in [59] and [60] are limited to a specific TMO, other approaches in [61] or [62] may be applied with an arbitrary TMO. For simplicity, the latter methods use legacy low bit-depth encoders for the compression of both the LDR base layer and the enhancement layer. The consequence is that the decorrelation step between LDR and HDR data is performed outside of the encoder and cannot fully take advantage of the great flexibility of modern video coding techniques. More flexible scalable methods with a LDR base layer and a HDR enhancement layer can be designed by including the mechanism of Inter-Layer Prediction (ILP) in the core of an encoder. For example, in modern compression standards such as H.264/AVC [49] or HEVC [48], complex block splitting schemes are used. The implementation of an ILP method into these standards can benefit from the block structure to adapt its properties to each block. Another advantage of this approach is the possibility to choose dynamically between the ILP mode and the regular inter or intra modes with rate distortion optimization.

Several examples of such HDR scalable methods have been proposed in the literature. In [83] and [84], only global ILP is performed, similarly to the decorrelation method proposed in [62]. In [66, 68, 67], the authors implemented a local ILP method in an H.264/AVC encoder. For each block (e.g. macroblock in the case of H.264), a linear relationship between the decoded LDR and the HDR block to encode is determined. In this case, scale and offset parameters must be signalled to the decoder. Such approaches are particularly well adapted to the case of local TMOs but they are limited by the high transmission cost of the scale and offset parameters.

The methods developed in this chapter endeavors to improve the performance of HDR scalable compression techniques especially in the challenging case where the LDR base layer was generated by a local tone mapping operator. For that purpose, new inter-layer prediction tools applied on a block-wise basis are introduced and implemented in the HEVC standard. In a first version of the proposed ILP scheme the parameters required for the prediction do not need to be transmitted, unlike the existing approach in [66, 68, 67]. Both the encoder and decoder can determine those parameters using the neighboring pixels of the current block. Since no additional data has to be encoded for the block, our method is not limited to a simple linear prediction. Instead, a linear spline model is determined which can take into account the possible non-linearity of the TMO even in small blocks. We additionally present an improved version in which a further scaling operation, requiring the transmission of a single parameter, is applied to the prediction block in order to increase the robustness in complex cases. Based on an efficient Rate-Distortion Optimization scheme, this prediction adjustment method substantially improves the compression performance. In order to compare the efficiency of our local ILP method with the state of the art, we also implemented in HEVC the linear ILP method, where the slopes and offsets are transmitted. For a fair comparison, the encoding of the parameters follows the method proposed by Garbas and Thoma [68], which is highly optimized for rate and distortion. It also contains more advanced prediction tools for the slope and offset parameters than the other existing local ILP methods in [66] and [67].

The chapter is organized as follows. An overview of the complete scalable coding scheme is depicted in section 4.1. Our inter layer prediction method is then described in section 4.2. In section 4.3, we present the prediction adjustment method. Further details on our HEVC implementation are given in section 4.4. Finally, the experimental results are presented in section 4.5.

4.1 Overview of the compression scheme

The diagram in figure 4.1 describes our compression scheme. The original HDR image is represented by absolute luminance values in cd/m^2. The human perception of luminance being non-linear, an Opto-Electrical Transfer function (OETF) and a quantization to integers is applied first to generate a perceptually uniform HDR signal suitable for compression. In this paper, we used the PQ-OETF function from [21, 22] which takes input luminance values of up to 10000 cd/m^2 and outputs 12 bit integers. This curve has been derived from Barten's Contrast Sensitivity Function (CSF) so that the quantization to 12 bit integers does not produce any visible loss when the inverse curve is applied to retrieve absolute luminance data. In our scheme, the PQ-OETF curve is applied independently to the HDR R, G and B channels.

The base layer is computed by a Tone Mapping Operator followed by gamma correction. The result is quantized to 8 bit integers to be encoded by a regular HEVC encoder. After a conversion to YUV 420 format, the 12 bit HDR layer is encoded using our modified version of an HEVC encoder that also takes the decoded LDR base layer as

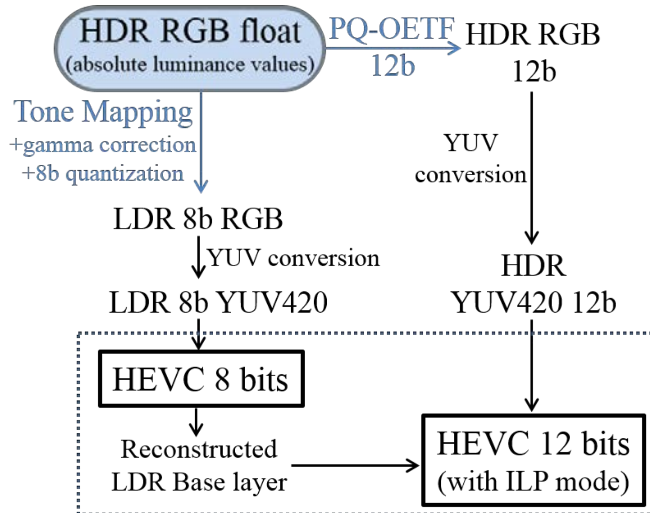


Figure 4.1: Diagram of the HDR scalable compression scheme. The dotted line indicates the parts of the diagram corresponding to our encoder taking the LDR and HDR layers as input in the YUV format.

input. In addition to the existing HEVC modes (i.e. intra, inter), our modified version contains the Inter Layer Prediction mode presented in the following sections.

4.2 Template based Inter Layer Prediction

The main particularity of our inter layer prediction scheme described in this section, is that it does not require the encoding of additional parameters. Note that in the improved version presented in section 4.3, a single parameter is transmitted for the refinement of the existing prediction. Throughout the chapter, we use the notations given in figure 4.2. The goal of the ILP is to determine a prediction for the current block Y_u^B. The first step consists in learning an inverse tone mapping curve from the template Y_k^T and the collocated template X_k^T in the LDR layer. The LDR block X_k^B collocated to Y_u^B is then inverse tone mapped using the curve estimated on the template to predict the HDR version. On the decoder side, all the LDR base layer is known. Moreover, since the template Y_k^T is in the causal zone of the current block, its decoded samples are also known. Hence, both the encoder and the decoder can determine the same curve using the decoded LDR and HDR pixel values in the templates.

4.2.1 Linear spline model

In our method, we propose to model the inverse tone mapping curve by a linear spline composed of three segments. The obtained inverse curve is therefore piecewise linear. As a result, it can take into account the possible non-linearity of the tone mapping while remaining simple to compute. Unlike in [55], where the authors also determine a

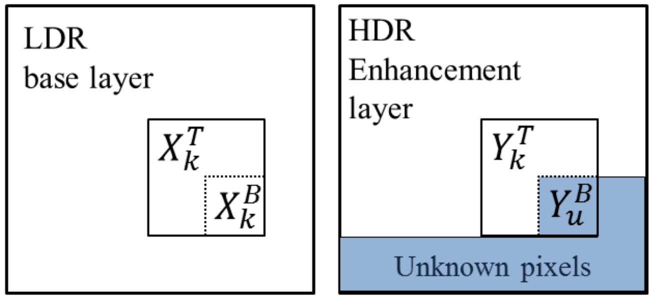


Figure 4.2: HDR and LDR layer notations. The letters k, u, B and T stand for 'known', 'unknown', 'block' and 'template' respectively.

piecewise linear tone curve, we derive a different curve for each block. Moreover, the parameters do not need to be transmitted to the decoder. Consequently, the higher number of parameters in this model compared to a simple linear model, as defined in [66, 68, 67], does not imply any additional cost.

The curve learning process consists in fitting a linear spline to the point cloud formed by the pixels' LDR and HDR values in X_k^T and Y_k^T respectively, as illustrated in figure 4.3.

Given l_{min} and l_{max}, the minimum and maximum LDR values in X_k^T, we must first define two interior knots k_1 and k_2 such that $l_{min} \leq k_1 \leq k_2 \leq l_{max}$. This step is detailed in the next subsection. A linear spline with three segments can then be defined by the following function:

$$f_\alpha(x) = \alpha_0 + \alpha_1 \cdot x + \alpha_2 \cdot (x - k_1)_+ + \alpha_3 \cdot (x - k_2)_+ \tag{4.1}$$

where $(z)_+ = \begin{cases} z & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases}$

By definition of f_α, the curve obtained is piecewise linear in such a way that two consecutive lines intersect exactly at the knot in between (i.e. k_1 or k_2).

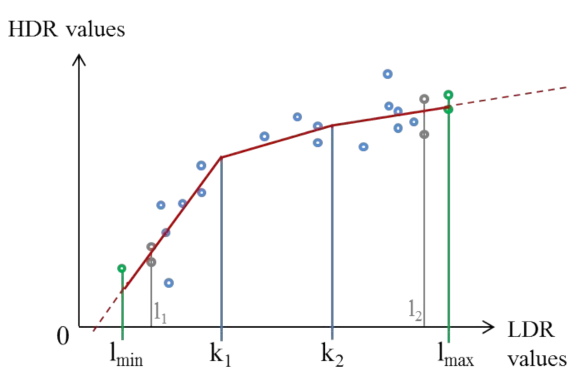


Figure 4.3: Example of linear spline fitted to the data points.

The vector of coefficients α in f_α can be fitted to the data by solving the standard least squares problem:

$$\hat{\alpha} = \underset{\alpha}{argmin} \left(\sum_{i=1}^{n} (f_\alpha(x_i) - y_i)^2 \right) \tag{4.2}$$

where n is the number of pixels in the training data (i.e. the number of pixels in the template), and x_i and y_i are the values of a pixel i in X_k^T and Y_k^T respectively.

Finally, a prediction $\widehat{Y_u^B}$ for the block Y_u^B is given by $\widehat{Y_u^B} = f_{\hat{\alpha}}(X_k^B)$. Note that, in addition to the increased accuracy compared to a simple linear regression, this model also has better extrapolation properties. For instance, X_k^B may contain values above l_{max}. In this case, the last segment of the spline will be considered. Therefore, it is more likely to be representative of the points with a high LDR value than a single linear model fitted to the whole range $[l_{min}, l_{max}]$.

4.2.2 Determination of the knots

A simple default choice for the knots k_1 and k_2 consists in splitting the interval $[l_{min,} l_{max}]$ in three equal parts as in the example of figure 4.3. We obtain

$$k_1 = l_{min} + \frac{1}{3} \cdot (l_{max} - l_{min}), \tag{4.3}$$

$$k_2 = l_{min} + \frac{2}{3} \cdot (l_{max} - l_{min}). \tag{4.4}$$

However, this strategy does not ensure that each segment of the spline has enough data points to be fitted to. For example, it is possible that l_1, the second smaller LDR value in X_k^T, is greater than k_1. In that case, k_1 is set to l_1 so that there are two distinct LDR values in the data in the interval $[l_{min}, k_1]$. The same method is used if $l_2 < k_2$, where l_2 is the second higher LDR value in X_k^T.

Furthermore, if the number n_{LDR} of distinct LDR values in X_k^T is small, each of the three fitted lines rely on too few data points. In order to improve robustness, when $n_{LDR} < 8$, a simple linear regression is performed instead, which is equivalent to solving (4.2) with $k_1 = l_{min}$ and $k_2 = l_{max}$.

4.3 Prediction adjustment factor

In some complex cases, local illumination variations can be difficult, if not impossible, to predict using only the neighborhood of the current block. For this reason, we also introduced a method that rescales the values of the prediction block initially obtained with the previous method. This time, the computation of the scaling parameters is performed using the original block Y_u^B that is unknown to the decoder. As a result, it is then necessary to transmit additional data for each prediction block.

The first step consists in determining a scaling factor $\hat{s}$ and an offset $\hat{o}$. Then, the adjusted prediction block $\widetilde{Y_u^B}$ will be obtained with the following operation :

$$\widetilde{Y_u^B} = \hat{o} + \hat{s} \cdot \widehat{Y_u^B} \tag{4.5}$$

The parameters must be chosen to minimize the mean squared error between the original block and the new prediction : $\|Y_u^B - \widetilde{Y_u^B}\|_2^2$. The optimal parameters can be determined with a linear regression. However, for reasons of transmission cost, we prefer to encode only a scaling factor. Therefore, the offset must be estimated only from data known by the decoder (i.e. the pixel data in $\widehat{Y_u^B}$, the scaling factor $\hat{s}$ already decoded). Note that in a compression scheme based on the DCT transform such as HEVC, the difference between the estimated offset $\hat{o}$ and the optimal offset o will be indirectly encoded as the DC coefficient of the transformed prediction residual $Y_u^B - \widetilde{Y_u^B}$. That is why it is preferable in our case to only estimate the offset without explicitly encoding the difference $o - \hat{o}$ so as not to interfere with the quantization and encoding of the DC coefficient that is already optimized. For instance, in HEVC, the quantization of the DC coefficient is dependent on the QP parameter and advanced tools such as rate distortion optimized quantization (RDOQ) might also influence the encoding of this value.

4.3.1 Offset Estimation

In the general case, we know that given a slope s, the optimal offset o is given by :

$$o = \overline{Y_u^B} - s \cdot \overline{\widehat{Y_u^B}} \tag{4.6}$$

where $\overline{Y_u^B}$ is the mean value of the original block Y_u^B, and $\overline{\widehat{Y_u^B}}$ is the mean value of the initially predicted block $\widehat{Y_u^B}$. But $\overline{Y_u^B}$ is unknown to the decoder. Making the assumption that the initial template based prediction gives a good approximation of the mean value of the block, we can replace $\overline{Y_u^B}$ by $\overline{\widehat{Y_u^B}}$ in equation (4.6) to obtain the estimated offset $\hat{o}$. Given the encoded and decoded scaling factor $\hat{s}$, we obtain :

$$\hat{o} = (1 - \hat{s}) \cdot \overline{\widehat{Y_u^B}} \tag{4.7}$$

From this definition of $\hat{o}$, examples of possible regression lines for different values of the parameter $\hat{s}$ are shown in figure 4.4. All the possible lines pass through the point of coordinates $(\overline{\widehat{Y_u^B}}, \overline{\widehat{Y_u^B}})$. Hence, for any value of $\hat{s}$, the mean value of the prediction block (and thus the DC coefficient) will remain unchanged after the adjustment (i.e. $\overline{\widetilde{Y_u^B}} = \overline{\widehat{Y_u^B}}$).

4.3.2 Adjustment factor computation

From equations (4.5) and (4.7), we know that :

$$\|Y_u^B - \widetilde{Y_u^B}\|_2^2 = \|Y_u^B - (1 - \hat{s}) \cdot \overline{\widehat{Y_u^B}} - \hat{s} \cdot \widehat{Y_u^B}\|_2^2$$

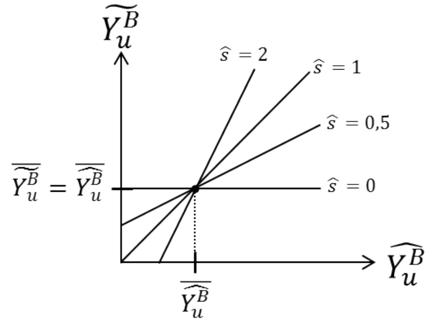


Figure 4.4: Examples of regression lines constrained by the offset estimation.

The optimal scaling factor in the least square sense is then given by :

$$\underset{s}{argmin} \sum_{i=1}^{n} (y_i - (1-s) \cdot \overline{\widehat{Y_u^B}} - s \cdot x_i)^2 \tag{4.8}$$

where x_i and y_i are respectively the values of a pixel i of the initially predicted block $\widehat{Y_u^B}$ and the original block Y_u^B, and n is the number of pixels in the blocks. By differentiating the sum in equation (4.8) with respect to s and by setting the derivative to zero, we find the following expression for the optimal value s :

$$s = \frac{\overline{xy} - \overline{x} \cdot \overline{y}}{\overline{x^2} - \overline{x}^2} \tag{4.9}$$

where

$$\overline{xy} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i \cdot y_i, \qquad \overline{x^2} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i^2,$$

$$\overline{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i = \overline{\widehat{Y_u^B}} \quad \text{and} \qquad \overline{y} = \frac{1}{n} \cdot \sum_{i=1}^{n} y_i = \overline{Y_u^B}.$$

We can note that we obtain exactly the same expression for s than in a regular linear regression without the constraint on the offset in equation (4.7).

4.3.3 Adjustment factor encoding

Knowing the optimal parameter s, it must then be encoded. First, we determine a predictor s_{pred} for the value of s. Since the input values in $\widehat{Y_u^B}$ are already determined by a prediction method based on the neighborhood, we consider that there is no more spatial correlation to be exploited. Hence, we consider that in the general case $Y_u^B \approx \widehat{Y_u^B}$, and thus $s \approx 1$. Therefore, we take $s_{pred} = 1$ and the prediction error on s is then $s - s_{pred} = s - 1$.

The prediction error is then quantized. A quantization step of $\frac{1}{8}$ was found to give good results. We obtain :

$$s_Q = [(s-1) \cdot 8] \tag{4.10}$$

The operator [.] represents a rounding to the closest integer. Finally, we obtain the value of the parameter $\hat{s}$, by performing the inverse quantization :

$$\hat{s} = 1 + \frac{s_Q}{8} \tag{4.11}$$

Knowing $\hat{s}$ from equation (4.11), we can also compute the parameter $\hat{o}$ using equation (4.7). Only the value s_Q must be transmitted by entropy coding so that the decoder can perform the same operations.

Although the adjusted prediction is necessarily better than the initial prediction with respect to the mean square error (MSE), the transmission cost of s_Q may degrade the global rate distortion performance of the encoder. Therefore, it can be better in some cases not to perform the adjustment for a given block. From the equations (4.7) and (4.11), we can observe that when $s_Q = 0$, we have $\hat{s} = 1$, $\hat{o} = 0$, and thus $\widetilde{Y_u^B} = \widehat{Y_u^B}$. In this particular case, our prediction adjustment method has no effect. As a result, there is no need to transmit an additional flag specifying if the adjustment must be performed for the current block. By setting s_Q to 0 on the encoder side, a low number of bits is required and the decoder can interpret that the initial prediction block must not be modified.

More generally, a complete rate distortion optimization (RDO) scheme can be used on the encoder side by testing the encoding of the block when s_Q is replaced by any integer value s'_Q between 0 and s_Q. The encoder can choose the value of s'_Q giving the lowest rate distortion cost.

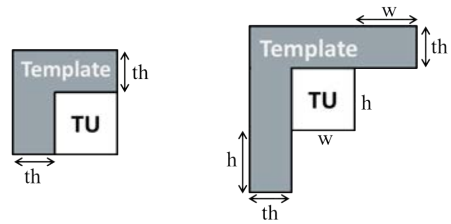
4.4 HEVC Implementation

Our encoder and decoder implementation is based on the HM Range Extension [75] which supports 12 bit input data.

ILP is considered as a prediction mode in addition to the existing HEVC intra and inter modes. For each Coding Unit (CU), the encoder determines the best mode, that is signaled to the decoder by a flag. The prediction algorithm itself, as described in previous sections, is performed on the Transform Unit (TU) level and only 2Nx2N Prediction Unit (PU) size was used. The same operation is performed on each of the three channels independently.

In a first implementation, we defined L-shaped templates with the same width and height as the current TU and a thickness th, as shown in figure 4.5(a). For the sake of simplicity, the template thickness th is independent of the TU size and it is fixed to 4 pixels, which corresponds to the minimum partition size in HEVC.

In some cases, the HEVC scanning order enables the use of decoded pixels on the below left and above right parts of the current block. We have studied how these pixels can improve the predictions by testing a second version of the algorithm. In this version,



(a) Simple template (b) Extended template

Figure 4.5: Template shapes.

we used the template shape of figure 4.5(b) that extends the block area by its width w to the top right and by its height h to the bottom left of the block. The thickness is kept to 4 pixels in the implementation. When some parts of the template are unavailable (e.g. current block on the top or left side of the image), the ILP is computed from the remaining available parts.

For the implementation of the prediction adjustment method, we used the full RDO scheme that tests all the integer values of s'_Q between 0 and s_Q. In order to keep reasonable encoding complexity, the rate distortion cost computation does not require a complete encoding and decoding of the block (i.e. including the DCT transform of the prediction residual, quantization, entropy coding of the coefficients, etc). Instead, the cost J is defined by the following equation :

$$J = SATD(\widetilde{Y_u^B}) + \lambda \cdot R \tag{4.12}$$

where $SATD(\widetilde{Y_u^B})$ is the sum of absolute transformed differences between the adjusted prediction and the original block. R is the number of bits required to encode the value s'_Q, and λ is a lagrangian multiplier value already defined in the HM [75]. The implementation of the SATD existing in the HM and based on the Hadamard transform was used. Therefore :

$$SATD(\widetilde{Y_u^B}) = \sum_{i=1}^{w \cdot h} |Had_i(Y_u^B - \widetilde{Y_u^B})| \tag{4.13}$$

where Had_i is the coefficient of the Hadamard Transform at position i, and w and h are the width and height of the blocks.

For the purpose of comparison with existing local inter layer prediction schemes, we also implemented the ILP method in which a simple linear regression is performed between the current original HDR block and the collocated LDR block in the base layer. The method from Garbas and Thoma [68] was chosen for the encoding of the resulting slope and offset parameters. This is, to our knowledge, the most advanced method from that point of view. In this method, a rate distortion optimization is performed by varying both the slope and offset parameters around the values obtained by linear regression. Similarly to the RDO scheme used in our prediction adjustment method, we estimated the RD cost using only the Hadamard transformed prediction residual and the entropy coded parameters. However, in spite of the fast cost computation, this RDO

(a) (b) (c) (d) (e) (f)

Figure 4.6: Tone mapped versions of the image *Memorial*. (a), (b) and (c) represent the HDR image at varying exposures with simple gamma correction for visualization on regular displays. (d), (e) and (f) are the results of the tone mapping operators mantiuk06 [85], fattal02 [44] and pattanaik00 [86] respectively.

scheme remains very complex because of the large number of combinations tested for the offset and scale parameters. For this reason, we also implemented a faster version of the original algorithm without RDO.

4.5 Experimental Results

For our experiment, we have used five HDR test images : *Marché* (1920x1080), *Montgolfière* (1920x1080), *Forest path* (2048x1536), *Memorial* (512x768), and *mpi atrium* (1024x672). *Marché* and *Montgolfière* are taken from sequences produced by Binocle and Technicolor within the framework of the french collaborative project NEVEx. They have been graded on a SIM2 HDR display. *Forest path* and *mpi atrium* are freely available from [87] and *Memorial* is available from [88]. Since those images are originally given in relative luminance, we multiplied them by a constant in order to convert to absolute luminance data. For each image, the SIM2 display was used to determine an appropriate constant. Note that all the test images have been generated with multiple exposures. Their dynamic range, expressed as the ratio between the luminance of the brightest and the darkest points, are reported in table 4.1. Note that we have chosen images representing a large variety of dynamic ranges, from *Forest path*, which has an almost standard dynamic range, to *Memorial*, with a very high contrast ratio.

Image	Dynamic Range
Marché	1:30,000
Montgolfière	1:64,000
Forest Path	1:600
Memorial	1:140,000
mpi atrium	1:8,000

Table 4.1: Dynamic Range of the test images.

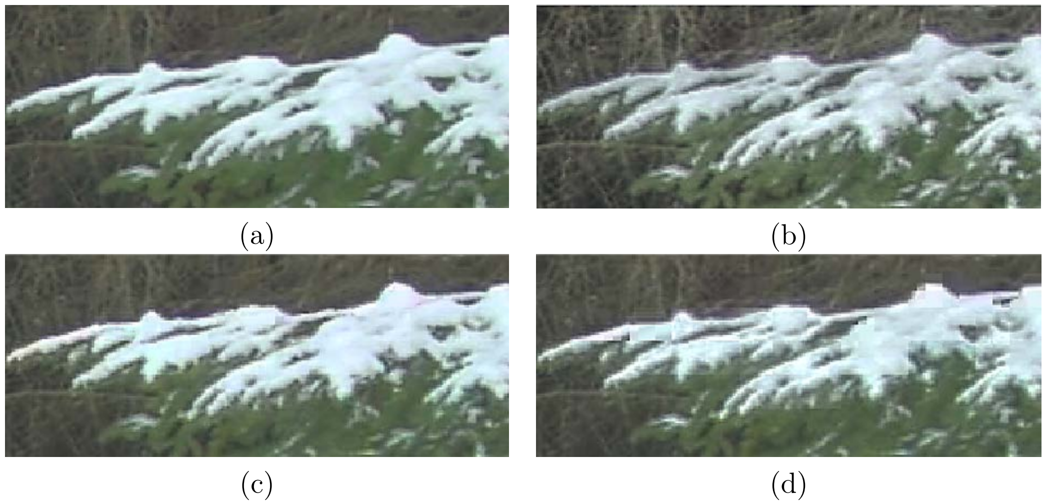
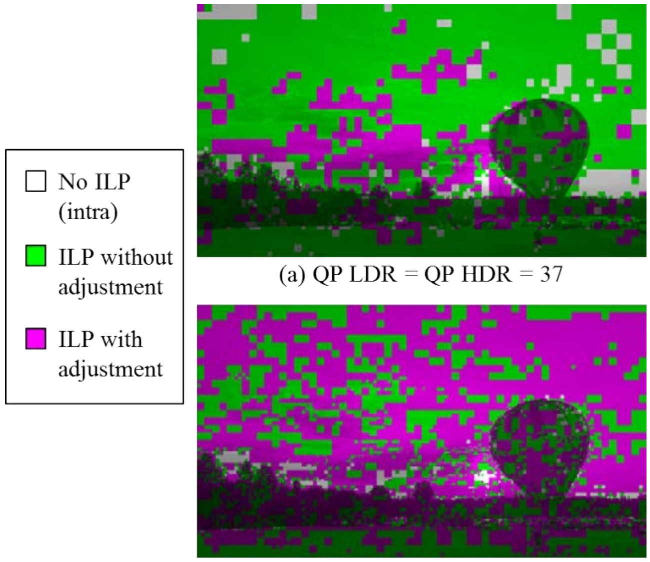


Figure 4.7: Prediction results on a detail of the *Forest path* image. (a) Original HDR layer. (b) LDR layer generated with mantiuk06 [85]. (c) our HDR prediction. (d) HDR prediction with Garbas and Thoma's method. For (c) and (d), QP 22 was used for both the LDR and HDR layers. For the sake of illustration, HDR images are represented with a simple gamma correction.

For each HDR image, several LDR versions were generated with TMOs from the pfstmo library [89]. In particular, the two local TMOs developed by Mantiuk et al. in [85] and by Fattal et al. in [44] and the global TMO of Pattanaik et al. [86] were used. Their respective implementations in the pfstmo library are referred to as mantiuk06, fattal02 and pattanaik00. An example of the results produced by these TMOs for the image *Memorial* is given in figure 4.6. For both the HDR and the LDR layers, the RGB colorspace is defined by the standard BT.709 primaries. For the sake of simplicity, we did not consider the case where the colorspaces of each layer are different.

In order to show the benefits of each aspect of the method, simulations were performed with three different versions. In the first version, only the template based prediction with simple templates is included. A second version uses the extended templates of figure 4.5(b) instead of the simple templates. Finally, a third version additionally includes the prediction adjustment method presented in section 4.3.

An example of prediction results in our HEVC implementation is shown in figure 4.7. The LDR base layer in figure 4.7(b) was generated with the local TMO mantiuk06. Figure 4.7(c) represents the prediction results with our template based method (without adjustment factor), while figure 4.7(d) is the prediction obtained with our HEVC implementation of the blockwise linear prediction [68]. Note that for figure 4.7(d), the linear regression is computed directly on the current block and the slope and offset parameters are transmitted for each block. Thus, a higher bitrate is necessary to perform this prediction than that of figure 4.7(c). Moreover, since HEVC adaptively splits the image via quadtree decomposition using a rate-distortion criterion, the extra cost of the parameters prevents the encoder from splitting the image into very small blocks.



(b) QP LDR = QP HDR = 22

Figure 4.8: Mode usage for our template based ILP with extended templates and adjustment factor.

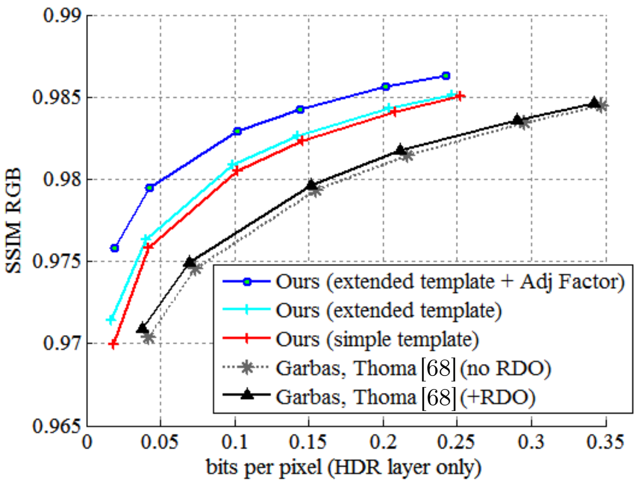
Conversely, our template based method fully takes advantage of the block splitting of HEVC because no additional information is transmitted. This explains why there are less block artifacts in 4.7(c).

An example of the usage of each mode in our method with both extended templates and prediction adjustment is shown in figure 4.8. When the LDR layer is encoded in high quality (i.e. low QP), most blocks perform ILP with linear adjustment. With low quality LDR encoding, the adjustment is less used by the encoder. In both cases, the ILP method is generally preferred over HEVC intra prediction except in areas that are either flat or clipped in the LDR version.

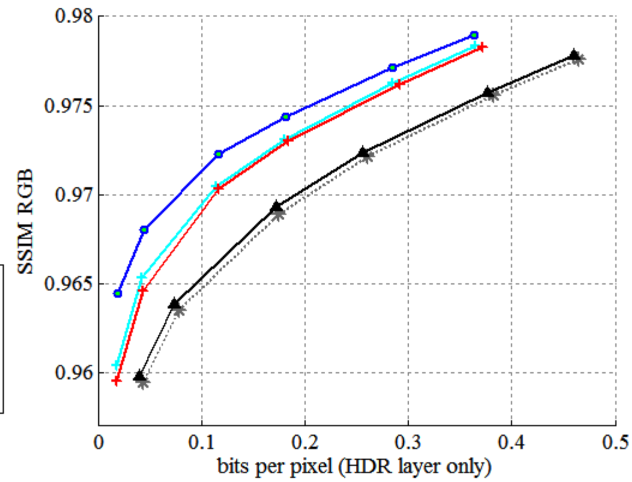
4.5.1 Rate Distortion Results

In order to compare our template based ILP with Garbas and Thoma's ILP method, we computed rate distortion curves by encoding the HDR layer using the QP values 22, 23, 25, 27, 32 and 37. Those simulations were performed for four different base layer qualities obtained by encoding the tone mapped image with QP values of 22, 27, 32 and 37 respectively. For the evaluation of the distortion, the SSIM metric was used [90]. It estimates the visibility of errors in the structure of the image more accurately than the peak signal to noise ratio (PSNR). This is an important property in our case because different types of artifacts may be produced by the methods to compare. Since the PQ-OETF curve used in our framework can be considered as perceptually uniform, the SSIM was computed on the 12 bit perceptually encoded RGB values.

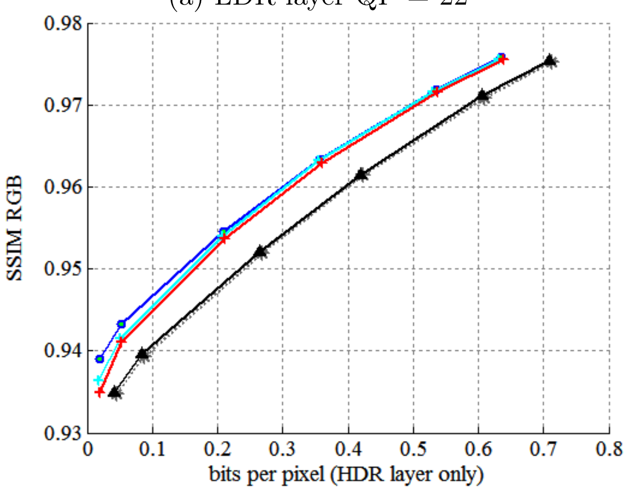
Rate Distortion curves for the image *mpi atrium* using a LDR layer computed with



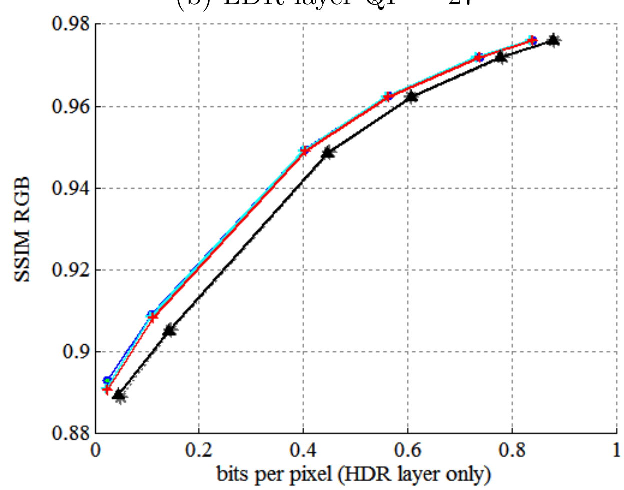
(a) LDR layer QP = 22



(b) LDR layer QP = 27



(c) LDR layer QP = 32



(d) LDR layer QP = 37

Figure 4.9: Rate Distortion curves for *mpi atrium* with mantiuk06 TMO [85]. (a), (b), (c) and (d) show respectively the results when the LDR layer is encoded at QP=22, QP=27, QP=32 and QP=37. Each curve is generated for QP values of 22, 23, 25, 27, 32, and 37 for the HDR layer.

the local TMO mantiuk06 are shown in figure 4.9. Only the bitrate of the HDR layer is shown here, since the bitrate of the base layer is independent of the ILP method. From this example, we can observe very significant gains obtained by our template based non-linear method compared to the linear one, especially when the LDR layer is encoded in high quality (e.g. QP=22 for figure 4.9(a)).

For all the tested images and TMOs, the coding gains for the HDR layer were computed with the Bjontegaard Delta Rate (BD-Rate) metric [76] and are presented in tables 4.2 to 4.4. The reference method for the comparison is the linear ILP with Garbas and Thoma's encoding with RDO. In the tables, negative values indicate a gain while

Method	LDR QP	*Marché*	*Montgolfière*	*Forest path*	*Memorial*	*mpi atrium*	**Average**
Garbas,Thoma [68] without RDO (fast)	22	9.5	19.3	10.7	6.2	*9.8*	**11.1**
	27	4.6	7.7	8.6	4.8	*7.6*	**6.6**
	32	2.3	3.9	5.6	3.0	*3.8*	**3.7**
	37	1.1	1.5	3.0	0.8	*1.7*	**1.6**
linear splines simple templates	22	-76.8	-42.9	-70.9	-40.2	*-44.0*	**-55.0**
	27	-57.3	-36.2	-59.2	-31.6	*-41.2*	**-45.1**
	32	-35.9	-27.4	-38.3	-24.1	*-30.3*	**-31.2**
	37	-18.9	-18.1	-25.4	-16.0	*-21.4*	**-20.0**
linear splines extended templates	22	-82.3	-50.5	-75.8	-44.8	*-50.5*	**-60.8**
	27	-61.5	-41.5	-64.6	-35.6	*-46.4*	**-49.9**
	32	-39.2	-30.8	-42.3	-26.9	*-33.3*	**-34.5**
	37	-21.3	-20.0	-27.9	-18.0	*-23.1*	**-22.0**
linear splines extended templates +adjustment factor	22	-87.4	-68.7	-87.2	-64.7	*-68.4*	**-75.3**
	27	-62.6	-48.7	-73.3	-46.2	*-58.6*	**-57.9**
	32	-39.3	-31.4	-45.5	-28.2	*-36.0*	**-36.1**
	37	-20.9	-20.6	-26.8	-17.0	*-22.5*	**-21.5**

Table 4.2: BD-Rate with local mantiuk06 tone mapping [85] (in %) where Garbas and Thoma's method with RDO [68] is used as the reference method in the computation of the Bjontegaard metric. Only the rate and distortion of the HDR layer is considered. The column in italic corresponds to the results presented in figure 4.9.

positive values indicate a loss in compression performance. In addition to the three versions of our algorithm, we also computed the BD-Rate for the fast version of Garbas and Thoma's ILP that does not include the time consuming rate distortion optimization scheme. This fast version was added to the results in order to show the compression performance of a method with an encoding complexity that is comparable to that of our template based methods. More detail on the complexity is given in subsection 4.5.3. In the case where the LDR layer is encoded in low quality (e.g QP=32, QP=37) only a small performance loss of about 2-4% is observed for this fast version compared to the version with full RDO. The complex RDO scheme, might be justified only when a high quality LDR layer is available.

As far as our template based linear spline methods are concerned, it is worth noting that all the BD-Rate values are negative, meaning that they always perform better than linear ILP. In most cases, the highest gains are observed when a high quality base layer is available. Furthermore, it can be seen that the rate distortion performance is increased by both the extended templates and the prediction adjustment factor methods. However, as the LDR layer is encoded in lower qualities, the three versions of our method tend to give similar results. This is clearly visible in the example of figure 4.9.

Regarding the extended template version with prediction adjustment, when combining the results with all the TMOs and all the QPs for the LDR layer, we obtain an average bitrate saving of 47% on the HDR layer bitrate compared to the linear ILP with Garbas and Thoma's method. Even in the worst case tested, the gain of this method is still 13.2%.

For the sake of comparison with non scalable schemes, we also present in table 4.5

Method	LDR QP	*Marché*	*Montgolfière*	*Forest path*	*Memorial*	*mpi atrium*	**Average**
Garbas,Thoma [68] without RDO (fast)	22	10.6	19.3	16.4	8.3	12.6	**13.4**
	27	5.2	10.6	9.7	7.1	7.4	**8.0**
	32	2.4	4.7	6.1	3.7	3.2	**4.0**
	37	1.4	1.5	4.8	1.5	1.9	**2.2**
linear splines simple templates	22	-65.1	-50.2	-15.8	-15.5	-49.3	**-39.2**
	27	-54.7	-29.7	-25.2	-17.8	-44.4	**-34.4**
	32	-37.2	-25.3	-27.5	-18.1	-25.1	**-26.6**
	37	-23.8	-20.1	-24.4	-14.2	-12.8	**-19.1**
linear splines extended templates	22	-78.2	-61.2	-40.8	-34.5	-61.1	**-55.2**
	27	-64.0	-39.1	-44.3	-31.2	-52.7	**-46.3**
	32	-41.3	-30.9	-34.5	-24.5	-28.9	**-32.0**
	37	-26.8	-22.2	-26.7	-17.8	-14.8	**-21.7**
linear splines extended templates +adjustment factor	22	-89.1	-70.5	-83.0	-65.8	-83.3	**-78.3**
	27	-68.1	-49.0	-84.7	-53.9	-64.9	**-64.1**
	32	-42.9	-33.0	-55.1	-37.0	-32.3	**-40.1**
	37	-26.7	-21.6	-35.1	-22.5	-13.2	**-23.8**

Table 4.3: BD-Rate with local fattal02 tone mapping [44] (in %) where Garbas and Thoma's method with RDO [68] is used as the reference method in the computation of the Bjontegaard metric. Only the rate and distortion of the HDR layer is considered.

Method	LDR QP	*Marché*	*Montgolfière*	*Forest path*	*Memorial*	*mpi atrium*	**Average**
Garbas,Thoma [68] without RDO (fast)	22	4.9	9.9	12.2	4.5	10.0	**8.3**
	27	2.8	5.0	7.3	2.5	5.7	**4.7**
	32	1.2	2.7	5.6	1.0	3.3	**2.7**
	37	1.2	1.1	4.4	1.1	2.4	**2.0**
linear splines simple templates	22	-68.4	-54.2	-89.2	-36.0	-73.9	**-64.3**
	27	-48.8	-35.3	-57.8	-29.1	-52.0	**-44.6**
	32	-28.2	-22.3	-34.5	-18.9	-28.0	**-26.4**
	37	-17.5	-16.4	-22.3	-12.0	-16.7	**-17.0**
linear splines extended templates	22	-73.6	-55.5	-92.5	-46.6	-80.0	**-69.6**
	27	-52.4	-38.3	-59.2	-35.5	-55.2	**-48.1**
	32	-30.5	-24.0	-38.5	-21.9	-30.6	**-29.1**
	37	-20.0	-18.5	-25.9	-14.1	-18.8	**-19.5**
linear splines extended templates +adjustment factor	22	-76.0	-59.5	-94.3	-54.5	-82.9	**-73.5**
	27	-52.4	-39.5	-57.1	-40.5	-54.8	**-48.9**
	32	-30.4	-24.5	-39.3	-22.8	-30.5	**-29.5**
	37	-19.8	-18.7	-25.5	-14.5	-18.0	**-19.3**

Table 4.4: BD-Rate with global pattanaik00 tone mapping [86] (in %) where Garbas and Thoma's method with RDO [68] is used as the reference method in the computation of the Bjontegaard metric. Only the rate and distortion of the HDR layer is considered.

the average BD-Rate of each scalable method with respect to Simulcast and single layer HDR encoding schemes. Simulcast consists in encoding both the LDR and HDR layers independently using a regular HEVC encoder (i.e. without inter-layer prediction). In the single layer scheme, the LDR layer is not encoded and therefore, only the bitrate of the HDR layer is taken into account. Nonetheless, for Simulcast as well as all the tested scalable methods, the sum of the bitrates of both layers is used. In order to

Method	Average Gain vs Simulcast	Average Loss vs single layer HDR
Garbas,Thoma [68] with RDO	-47.5%	39.2%
Garbas,Thoma [68] without RDO (fast)	-46.8%	41.0%
linear splines simple templates	-51.6%	29.0%
linear splines extended templates	-52.2%	27.3%
linear splines extended templates +adjustment factor	-53.3%	24.2%

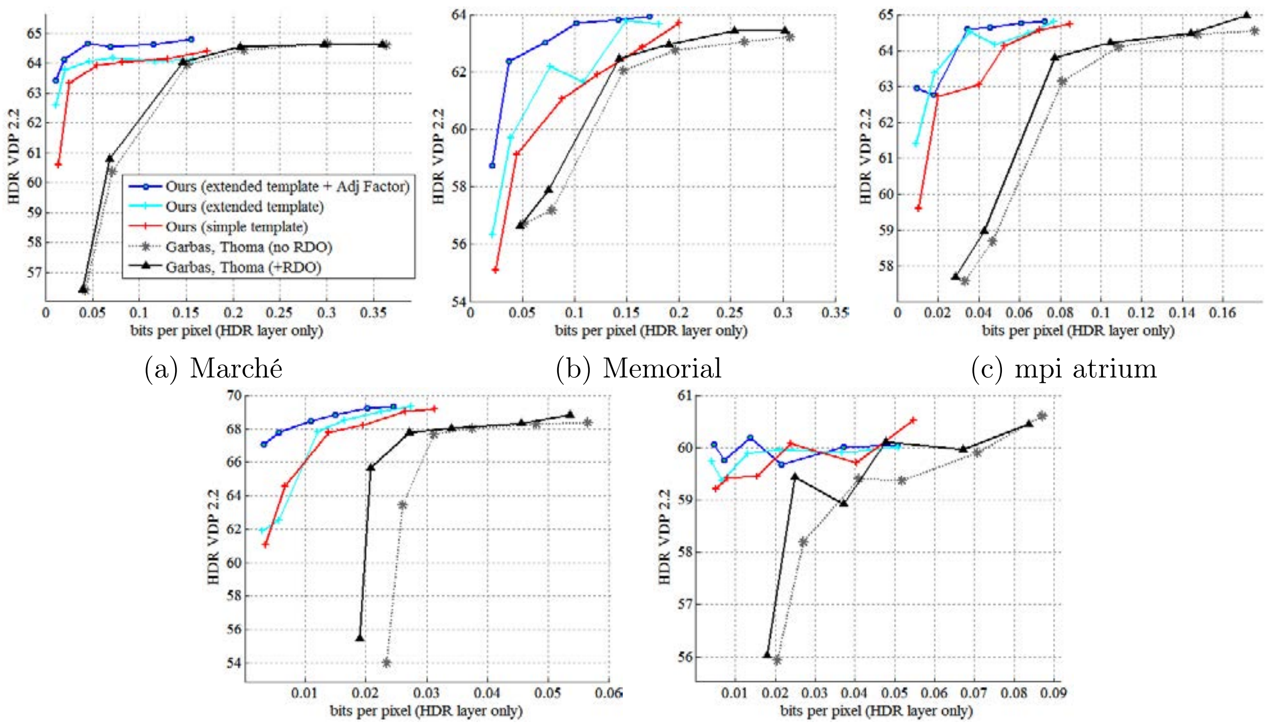
Table 4.5: Average BD-Rate for all the images and TMOs with respect to simulcast and single layer HDR encoding. For each method, the same QP was used for both layers.

obtain approximately the same quality for the HDR and LDR layers, this comparison was performed by using the same QP for both layers (except in the single layer method) and the QP values 22, 27, 32 and 37 were used. The results confirms that our method based on extended templates and prediction adjustment factor outperforms all the other scalable schemes. It reaches a 53.3% gain over Simulcast on average. The average loss compared to single layer encoding is reduced to 24.2%, which is reasonable considering the complex and highly non-linear relationships between the LDR and HDR images used in our simulation.

Additionally, if only the bitrate is considered, we have measured that when the same QP is used for both layers, the bitrate of the HDR layer represents on average about 17% of the total bitrate for the methods of Garbas and Thoma. For the three versions of our method, this ratio drops to approximately 10%.

4.5.2 Quality assessment with HDR-VDP 2.2

In addition to the rate distortion gains computed with the SSIM metric, the quality has also been assessed using HDR-VDP 2.2 [91] which is a perceptual metric specifically designed for HDR images. Before computing the quality index, the decoded Y'CbCr images were converted back to RGB and the inverse PQ-OETF curve was applied in order to retrieve absolute luminance values. Then, the version 2.2.1 of the metric was used to estimate the perceived quality of the decoded image with a quality index between 0 and 100, where 100 is reached when there is no visible difference with the original HDR image. The HDR-VDP 2.2 metric requires several parameters in order to take the conditions of visualization into account. For the experiment, we set the angular resolution of the image to 30 pixels per degree, which is a plausible value for the visualization with a standard resolution computer display. The predefined default



(a) Marché (b) Memorial (c) mpi atrium

(d) Forest Path (d) Montgolfière

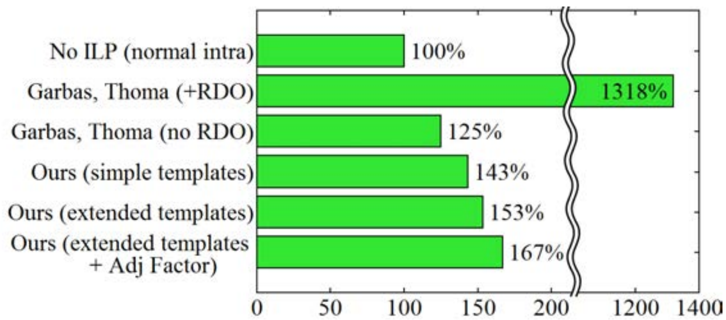
Figure 4.10: HDRVDP 2.2 Rate Distortion results for each image with fattal02 TMO [44]. Each curve is generated for QP values of 22, 23, 25, 27, 32, and 37 for the HDR layer and QP=22 for the LDR layer.

values were used for the other parameters (e.g. surrounding luminance, peak sensitivity, etc).

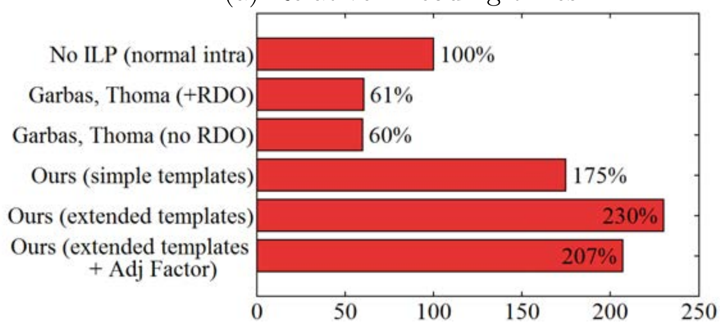
For each of the five tested HDR images, the resulting rate distortion curves are shown in figure 4.10. The HDR encoding was performed with a base layer generated with fattal02 TMO and encoded with LDR QP=22. Unlike the SSIM, the use of the HDR-VDP 2.2 metric may result in curves showing random behavior, which makes their interpretation more difficult. However, it clearly confirms that our template based methods perform better than Garbas and Thoma's method either with or without RDO. Moreover, on the whole, the ranking of the three versions of our algorithm remains consistent with the observations made with the SSIM metric, which confirms that both the extended templates and the prediction adjustment factor methods increased the visual quality at a given bitrate.

4.5.3 Complexity

An illustration of the complexity of the different ILP methods is given in figure 4.11. It shows the mean encoding and decoding times of the HDR layer as a percentage of the computation times of a normal HEVC encoding and decoding of the 12 bit HDR image.



(a) Relative Encoding times.



(b) Relative Decoding times.

Figure 4.11: Encoding and decoding times relatively to intra (no ILP) compression

Regarding the encoding times, figure 4.11(a) clearly shows the complexity added by the rate distortion optimization in [68]. Encoding with this method is 13.2 times as long as intra and 10.5 times as long as the same method without RDO. Our method increases the HEVC intra complexity by 43% to 67% depending on the version.

On the decoder side, figure 4.11(b) shows that our methods are between 1.75 and 2.3 times as complex as intra decoding. It should be noted, however, that no particular optimization was performed in our implementation, and floating point arithmetic operations were used extensively. Contrary to what could be expected, the use of the prediction adjustment factor method reduced the decoding times. The complexity added by the prediction adjustment is more than compensated by the fact that fewer AC coefficients have to be decoded.

4.6 Conclusion

We have presented a new scalable compression scheme for high dynamic range images, where the base layer is a corresponding low dynamic range version. An arbitrary and possibly local TMO may be used for generating the LDR version, thus enabling a complete artistic control over the content creation process. Our method adapts well to the case of local TMOs thanks to a template based piece-wise linear inter-layer prediction method that does not require the transmission of any additional parameter. In an advanced version of the method, the robustness of the predictions have been further improved by applying a linear scaling to the initially predicted block. Thanks to an efficient encoding of the scaling parameter, the overall compression performance is increased by this prediction adjustment method. Moreover, our HEVC based implementation enables the use of extended templates giving even higher performance. Our experiments have shown significant coding gains on the HDR layer compared to the state of the art local ILP methods that are limited to linear prediction and that require the encoding of the scaling and offset parameters for each block.

In the presented scalable coding scheme, the same method was used for the luma and the chroma components. Note that an adapted prediction strategy for the inter-layer prediction of chroma could further improve the coding efficiency. This aspect is studied in detail in the next chapter in addition to the definition of an alternative to the Y'CbCr colorspace for the coding of HDR images.

Chapter 5

Color Inter-layer prediction for scalable schemes

In the previous chapter, we have introduced efficient prediction techniques for the scalable compression of HDR images with a LDR base layer. The methods presented were especially developed for the prediction of the achromatic luma component. In this chapter, the focus is set on the numerical representation and the prediction of color information of the HDR enhancement layer, given the decoded LDR base layer.

As regards the representation of color, the use of Y'CbCr color-spaces prevails in the compression of digital images and videos. However, for HDR images, a Y'CbCr encoding including chroma down-sampling may introduce several types of distortions identified by Poynton et al. in a recent work [92]. The main reason is that the chroma CbCr components are not well decorrelated from the luminance. Although the same problems already occurred with LDR images, the artifacts were much less visible. Therefore, in the context of the distribution of High Dynamic Range content, other color encoding schemes are emerging, based on the CIE 1976 Uniform Chromaticity Scale (i.e. u'v' color coordinates). The CIE u'v' coordinates have no correlation with luminance which enables a better separation of the chromatic and achromatic signals than a Y'CbCr color-space, and thus, a better down-sampling of the color signal. The downside, however, is that perceptual uniformity is not fully satisfied because of the loss of color sensitivity of the human eye at low luminance levels. The LogLuv TIFF image format [38] was the first attempt at using this color representation for encoding images. Thereafter, more advanced compression schemes based on the MPEG standard also used the u'v' color representation [62, 93, 71, 68]. In [92], Poynton et al. proposed a modification of the CIE u'v' to take into account the lower accuracy of the human perception of color in dark areas. In their modified version, below a luminance threshold, the chromaticity signal is attenuated towards gray proportionally to the luma. This method avoids encoding with too much precision the color noise that may appear in dark areas.

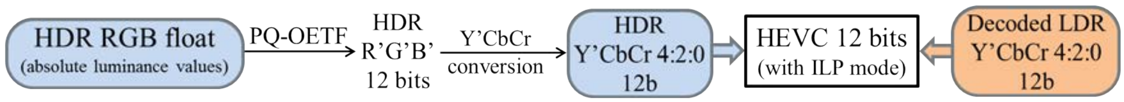
Inter-layer prediction in the context of HDR scalability amounts to inverting the tone mapping operator (TMO) that was used to generate the LDR image of the base layer. Generic inter-layer prediction methods were presented in the previous chapter, based on the assumption that the HDR and the LDR signals are well correlated. This assumption holds for the luma components of both layers which mainly differ by the non-linearities caused by the tone mapping on one hand and by the differences between the LDR gamma correction and the HDR perceptual encoding on the other hand. However, more complicated relationships may exist between the LDR and HDR chromatic components, especially if the HDR colors are encoded with a u'v' representation instead of the classical Y'CbCr used in LDR. In [62], the authors observed that many TMOs have little impact on the CIE u'v' color coordinates of the pixels in an image. The LDR layer u'v' components are then computed from the completely decoded LDR image and are used for predicting the color of the HDR enhancement layer. We show in this chapter under which circumstances this assumption is valid and how to generalize it to a broader range of tone mapping operators. For that purpose, we exploit general knowledge in the field of tone mapping and more precisely in the way the color information is handled. A very well-known method for generalizing any Tone Mapping Operator to color images was developed by Schlick [94]. In [45], Tumblin and Turk then improved this method by adding a parameter for a better control of the saturation of colors in the tone mapped image. Later, several other popular TMOs in [85, 44, 42] used the same color correction method.

In this chapter, a model for predicting the color of the HDR image from the decoded LDR image and HDR luma channel is introduced. The model is derived from the color correction equations of Tumblin and Turk [45]. Since this color correction method is based on a saturation parameter that might be unknown to the encoder, we developed a pre-analysis method that automatically determines the most suitable parameter value given the original HDR and LDR pair of images. This parameter is then transmitted as meta-data and used for performing predictions. We developed two versions of our scalable encoding scheme using either Y'CbCr or u'v' representation for the HDR layer. The color inter-layer prediction equations are derived for both color-spaces. For a fair comparison, we use the modified u'v' coordinates proposed in [92] which are more perceptually uniform than the original u'v' representation. In order to keep complete backward compatibility, the LDR layer is encoded in Y'CbCr in both encoding schemes.

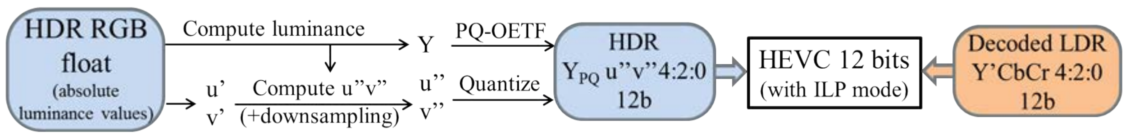
The remainder of the chapter is organized as follows. The two encoding schemes are presented in detail in section 5.1. Then, the color model based on the color correction of Tumblin and Turk is explained in section 5.2. From this model, we derive in section 5.3 the prediction equations of the chromatic components for both encoding schemes. The pre-analysis step which automatically determines the model's saturation parameter is also developed in section 5.4. Finally, our experimental results are presented in section 5.5.

5.1 Overview of the scalable HDR compression schemes

This section presents the two versions of our compression scheme based on either the Y'CbCr or the CIE u'v' representation.



(a) scheme based on Y'CbCr



(b) scheme based on u'v'

Figure 5.1: Overview of the Y'CbCr and u'v' scalable compression schemes.

5.1.1 Y'CbCr compression scheme

In the Y'CbCr scheme, illustrated in figure 5.1(a) the same color encoding as in chapter 4 is used. The PQ-OETF is applied to the linear R,G and B components independently and the resulting $R'G'B'$ components are converted to Y'CbCr color-space using the standard conversion matrix from the ITU-R BT-709 recommendations [70]. Then, the chroma channels Cb and Cr are downsampled and the image is sent to a modified version of HEVC including our inter-layer prediction mode.

5.1.2 CIE u'v' based compression scheme

In the second scheme, shown in figure 5.1(b), the true luminance Y is computed and the PQ-OETF is applied only to this achromatic component to form the luma channel Y_{PQ}. Then, the CIE u'v' color coordinates are computed from the linear RGB values. The modification proposed in [92] is applied in our scheme. The modified u'v' components are noted u"v" and are computed with the following formula:

$$
\begin{aligned}
u'' &= (u' - u'_r) \cdot \frac{Y_{PQ}}{max(Y_{PQ}, Y_{th})} + u'_r \\
v'' &= (v' - v'_r) \cdot \frac{Y_{PQ}}{max(Y_{PQ}, Y_{th})} + v'_r
\end{aligned}
\tag{5.1}
$$

where u'_r and v'_r are the u'v' coordinates of the standard D65 illuminant [18] : $u'_r = 0.1978$ and $v'_r = 0.4683$. And Y_{th} is a threshold on the luma Y_{PQ} that we set to 1000 which corresponds to an absolute luminance value of 4.75 cd/m^2. Thanks to this modification, the quantization of the color is coarser in dark regions that may contain invisible color noise. In the decoding process, the u'v' coordinates are retrieved by performing the inverse operations.

The two color channels are formed by quantizing the u"v" pixel values. In [92], the authors determined that quantizing those values to only 9 bits integers did not produce any visible artifact. However, they did not consider the HEVC based compression of

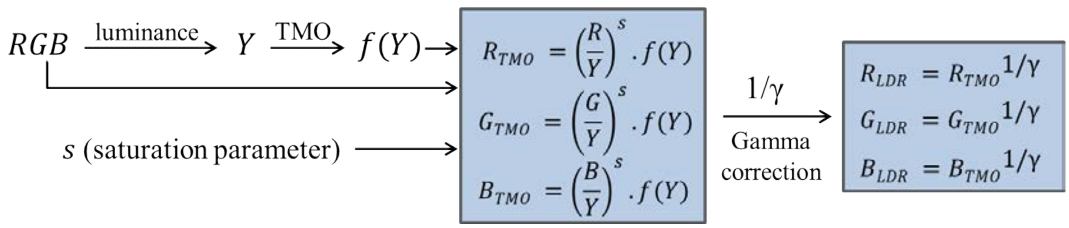


Figure 5.2: Color model of Tumblin and Turk [45]

both chromatic and achromatic signals. In practice, the quantization step of u" and v" should be chosen depending on the luma bitdepth in order to have a good bitrate allocation between luma and chromaticity. From our experiments, we have found that quantizing the chromaticity signal to 1 bit less than the luma bitdepth gave a reasonable tradeoff. Thus, 11 bits integers are used for the chromaticity. Knowing that the values of u" and v" are comprised between 0 and 0.62, we apply a factor of 3302 to obtain quantized values u''_Q and v''_Q in the range $[0, 2^{11} - 1]$, as

$$u''_Q = [3302 \cdot u''], \qquad v''_Q = [3302 \cdot v''] \ , \tag{5.2}$$

where [.] represents the rounding to the nearest integer.

Similarly to the Y'CbCr scheme, the chromatic components u''_Q and v''_Q are downsampled. In order to keep compatibility with typical LDR encoding schemes, the LDR layer is encoded in the Y'CbCr 420 format (i.e. Y'CbCr with both horizontal and vertical chroma downsampling).

5.1.3 Modified HEVC for Scalability

Our modified HEVC version contains an Inter-layer prediction mode in addition to the existing intra and inter modes. The mode decision is made at the Coding Unit (CU) level. As regards the inter-layer prediction of the luma channel, the ILP method presented in chapter 4 is used for both schemes. The method locally determines inverse tone mapping curves on a per-block basis for predicting the HDR data from the decoded LDR version. As a result, our ILP method is not limited to the case of a LDR layer generated with a global TMO.

For both encoding schemes, the inter-layer prediction equations of the chromatic components were derived by assuming that the base layer was generated with a TMO which applies the color correction of Tumblin and Turk [45]. More details on this color correction method are given in the next section and our prediction equations are presented in section 5.3.

5.2 Tone mapping color model

The color correction method used by Tumblin and Turk for generalizing any TMO to color images is illustrated in figure 5.2. In this method, the TMO f, that can be either global or local, is first applied to the luminance Y. The tone mapped RGB components are then obtained based on a saturation parameter s, the tone mapped luminance $f(Y)$ and the ratio between the HDR RGB components and the original luminance Y. Since the tone mapping is performed on linear RGB values, a further gamma correction is required. The final gamma corrected LDR RGB components are then expressed by:

$$C_{LDR} = \left(\frac{C}{Y}\right)^{\frac{s}{\gamma}} \cdot f(Y)^{\frac{1}{\gamma}} \tag{5.3}$$

with $C = R,G,B$.

With this color correction, the hue of the original HDR image is well preserved in the tone mapped image and the saturation of the LDR image can be manually adjusted with the parameter s. These properties are essential for preserving the artistic intent in the LDR version. As a result, even if the LDR images in the base layer were not generated using this color correction explicitly, we can still assume that the same properties are satisfied. For most content, the model in equation (5.3) gives a good approximation of the relation between the chromatic information in the HDR and LDR images. The next section describes how the HDR chroma components can be predicted by the model from the LDR base layer for both u"v" and Y'CbCr schemes developed in the chapter.

5.3 Color inter-layer prediction

5.3.1 Prediction of CIE u'v' values

In the original definition of the CIE standard, the u'v' color coordinates can be computed from the CIE XYZ values by :

$$\begin{aligned} u' &= \frac{4 \cdot X}{X + 15 \cdot Y + 3 \cdot Z} \\ v' &= \frac{9 \cdot Y}{X + 15 \cdot Y + 3 \cdot Z} \end{aligned} \tag{5.4}$$

Since the linear RGB components can be expressed as a linear combination of X, Y and Z, we can write :

$$\begin{aligned} u' &= \frac{a_0 \cdot R + a_1 \cdot G + a_2 \cdot B}{b_0 \cdot R + b_1 \cdot G + b_2 \cdot B} \\ v' &= \frac{c_0 \cdot R + c_1 \cdot G + c_2 \cdot B}{b_0 \cdot R + b_1 \cdot G + b_2 \cdot B} \end{aligned} \tag{5.5}$$

where the coefficients a_0 to c_2 are fixed values depending on the chromaticities of the HDR RGB color-space. In the case of BT-709 RGB [70], the values of the coefficients are :

$$\begin{array}{lll} a_0 = 1.650 & a_1 = 1.430 & a_2 = 0.722 \\ b_0 = 3.661 & b_1 = 11.442 & b_2 = 4.114 \\ c_0 = 1.914 & c_1 = 6.436 & c_2 = 0.650 \end{array}$$

From the model described in equation (5.3) we can directly determine :

$$f(Y)^{\frac{1}{s}} = {R_{LDR}}^{\frac{\gamma}{s}} \cdot \frac{Y}{R} = {G_{LDR}}^{\frac{\gamma}{s}} \cdot \frac{Y}{G} = {B_{LDR}}^{\frac{\gamma}{s}} \cdot \frac{Y}{B} \tag{5.6}$$

Thus,

$$\begin{cases} {R_{LDR}}^{\frac{\gamma}{s}} & = {G_{LDR}}^{\frac{\gamma}{s}} \cdot \frac{R}{G} \\ {B_{LDR}}^{\frac{\gamma}{s}} & = {G_{LDR}}^{\frac{\gamma}{s}} \cdot \frac{B}{G} \end{cases} \tag{5.7}$$

Now, let us rewrite equation (5.5) as :

$$u' = \frac{(a_0 \cdot \frac{R}{G} + a_1 + a_2 \cdot \frac{B}{G})}{(b_0 \cdot \frac{R}{G} + b_1 + b_2 \cdot \frac{B}{G})} \tag{5.8}$$

A similar equation can be found for v'. By multiplying both the numerator and the denominator by ${G_{LDR}}^{\frac{\gamma}{s}}$ in equation (5.8) and by using equation (5.7), we obtain a prediction value u'_{pred} for u' based only on the LDR RGB and the model parameters γ and s. The expression of v'_{pred} is obtained the same way :

$$\begin{aligned} u'_{pred} &= \frac{a_0 \cdot {R_{LDR}}^{\frac{\gamma}{s}} + a_1 \cdot {G_{LDR}}^{\frac{\gamma}{s}} + a_2 \cdot {B_{LDR}}^{\frac{\gamma}{s}}}{b_0 \cdot {R_{LDR}}^{\frac{\gamma}{s}} + b_1 \cdot {G_{LDR}}^{\frac{\gamma}{s}} + b_2 \cdot {B_{LDR}}^{\frac{\gamma}{s}}} \\ v'_{pred} &= \frac{c_0 \cdot {R_{LDR}}^{\frac{\gamma}{s}} + c_1 \cdot {G_{LDR}}^{\frac{\gamma}{s}} + c_2 \cdot {B_{LDR}}^{\frac{\gamma}{s}}}{b_0 \cdot {R_{LDR}}^{\frac{\gamma}{s}} + b_1 \cdot {G_{LDR}}^{\frac{\gamma}{s}} + b_2 \cdot {B_{LDR}}^{\frac{\gamma}{s}}} \end{aligned} \tag{5.9}$$

Hence, given the ratio between the parameters γ and s, and the decoded LDR data, we can directly predict the HDR u' and v' color components by applying the standard $u'v'$ conversion equation (5.5) to the decoded LDR RGB values raised to the power $\frac{\gamma}{s}$.

This is a generalized version of Mantiuk et al's color predictions in [62] that considered $u' = u'_{LDR}$ and $v' = v'_{LDR}$ where u'_{LDR} and v'_{LDR} are computed from the linearized LDR RGB values (i.e. ${R_{LDR}}^{\gamma}$, ${G_{LDR}}^{\gamma}$ and ${B_{LDR}}^{\gamma}$). Our prediction is equivalent in the particular case where $s = 1$. Note that in [68], Garbas and Thoma also predict the HDR layer u'v' from the LDR layer u'v' but the gamma correction is not taken into account in the computation of u'_{LDR} and v'_{LDR}. In this case, the prediction is thus equivalent to taking $\frac{\gamma}{s} = 1$, which is far from optimal in general since typical γ values are 2.2 or 2.4 while s usually does not exceed 1. Figure 5.3 shows an example of color predictions produced by [62] and [68]. The base layer in figure 5.3(b) was tone mapped from the original HDR image in 5.3(a) by the TMO [85]. This TMO explicitly uses the color correction in equation (5.3) and the parameters $s = 0.6$ and $\gamma = 2.2$ were chosen. Garbas and Thoma's color predictions [68] result in too low saturation as shown in figure 5.3(d). Better results are obtained in figure 5.3(c) by Mantiuk et al's predictions [62] which take the gamma correction into account. However, the colors are still less saturated than in the original image because the parameter s used in the TMO was less than 1. In our method, the saturations of the original HDR image can be recovered by using the actual values of the parameters γ and s in equation (5.9).

(a) (b) (c) (d)

Figure 5.3: u'v' prediction results on a frame of the *Market3* sequence. (a) Original HDR image. (b) Tone mapped image with the TMO in [85] using $s = 0.6$ and $\gamma = 2.2$. (c) HDR color prediction from [62] (i.e. assuming $s = 1$ and $\gamma = 2.2$). (d) HDR color prediction from [68] (i.e. assuming $\frac{\gamma}{s} = 1$).

Since our compression scheme is based on the modified version u"v" of the CIE u'v' coordinates, the predictions u''_{pred} and v''_{pred} are formed with equation (5.1) using u'_{pred}, v'_{pred} and the decoded HDR luma. Finally, u''_{pred} and v''_{pred} are multiplied by 3302 and rounded to the nearest integer, as in equation (5.2), to predict the quantized values u''_Q and v''_Q .

5.3.2 Prediction in Y'CbCr

In the case where the HDR layer is encoded in Y'CbCr color-space, a different prediction scheme is necessary. Unlike the u'v' coordinates, the Cb and Cr chroma components cannot be predicted directly. First, we must predict the HDR RGB values in the linear domain. Then, the PQ-OETF curve [21] must be applied to the predicted RGB values before computing the chroma prediction.

For the derivation of prediction equations of the RGB components, let us first define X_r and X_b as the ratios between the components :

$$X_r = \frac{R}{G}, \qquad X_b = \frac{B}{G} \tag{5.10}$$

From the model given by equation (5.3), we have :

$$\begin{aligned} R_{LDR} &= \left(\frac{R}{Y}\right)^{\frac{s}{\gamma}} \cdot f(Y)^{\frac{1}{\gamma}} = {X_r}^{\frac{s}{\gamma}} \cdot \left(\frac{G}{Y}\right)^{\frac{s}{\gamma}} \cdot f(Y)^{\frac{1}{\gamma}} \\ &= {X_r}^{\frac{s}{\gamma}} \cdot G_{LDR} \end{aligned} \tag{5.11}$$

The ratios X_r and X_b can thus be found using only the LDR RGB components :

$$X_r = \left(\frac{R_{LDR}}{G_{LDR}}\right)^{\frac{\gamma}{s}}, \qquad X_b = \left(\frac{B_{LDR}}{G_{LDR}}\right)^{\frac{\gamma}{s}} \tag{5.12}$$

Using the ratios X_r and X_b, and the luminance expressed as a linear combination of the RGB components, we can derive the following equations :

$$\begin{aligned} Y &= \alpha_0 \cdot R + \alpha_1 \cdot G + \alpha_2 \cdot B \\ &= (\alpha_0 \cdot X_r + \alpha_1 + \alpha_2 \cdot X_b) \cdot G \end{aligned} \tag{5.13}$$

Thus,

$$\begin{aligned} G &= \frac{Y}{\alpha_0 \cdot X_r + \alpha_1 + \alpha_2 \cdot X_b} \\ R &= X_r \cdot G \\ B &= X_b \cdot G \end{aligned} \tag{5.14}$$

where the coefficients α_0, α_1 and α_2, depend on the RGB color-space used. For the BT-709 color-space, $\alpha_0 = 0.2126$, $\alpha_1 = 0.7152$ and $\alpha_2 = 0.0722$.

However, the true luminance Y is not known in the Y'CbCr scheme. Only an approximation $\widetilde{Y}$ is obtained when the inverse PQ-OETF curve, which we denote PQ^{-1}, is applied to the luma channel Y'. The predicted RGB values can then be obtained by applying equation (5.14) and by replacing Y by $\widetilde{Y} = PQ^{-1}(Y')$. This can be inaccurate particularly in very saturated regions where one of the components is close to zero. It has been observed experimentally that better results are obtained by approximating the PQ-OETF function by a power law in the expression of $\widetilde{Y}$:

$$\begin{aligned} \widetilde{Y} &\approx \left(\alpha_0 \cdot R^{\frac{1}{p}} + \alpha_1 \cdot G^{\frac{1}{p}} + \alpha_2 \cdot B^{\frac{1}{p}}\right)^p \\ &\approx \left(\alpha_0 \cdot X_r^{\frac{1}{p}} + \alpha_1 + \alpha_2 \cdot X_b^{\frac{1}{p}}\right)^p \cdot G \end{aligned} \tag{5.15}$$

Finally the approximation for the green component G is given by :

$$G \approx \frac{\widetilde{Y}}{\left(\alpha_0 \cdot {X_r}^{\frac{1}{p}} + \alpha_1 + \alpha_2 \cdot {X_b}^{\frac{1}{p}}\right)^p} \tag{5.16}$$

Note that for $p = 1$, this is equivalent to the previous approximation (i.e. $\widetilde{Y} \approx Y$). Examples of prediction results are shown in figure 5.4 with varying values of p. From our experiments, we have found that using $p = 4$ gave good results in most situations.

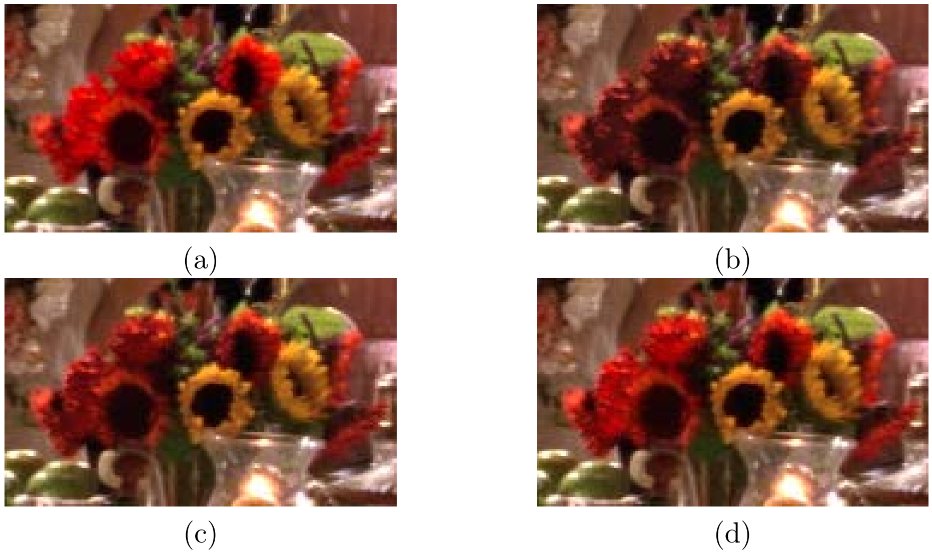


Figure 5.4: Y'CbCr prediction results on a detail of a frame in the *StEM* sequence. (a) Original HDR image. (b), (c) and (d) : prediction images with respectively $p = 1$, $p = 2$ and $p = 4$.

In order to improve our predictions in dark areas, we used in our implementation a slightly modified version of the ratios X_r and X_b :

$$\widetilde{X_r} = \left(\frac{R_{LDR} + \epsilon}{G_{LDR} + \epsilon}\right)^{\frac{\gamma}{s}}, \quad \widetilde{X_b} = \left(\frac{B_{LDR} + \epsilon}{G_{LDR} + \epsilon}\right)^{\frac{\gamma}{s}} \tag{5.17}$$

where ϵ is a small value fixed to 1% of the maximum LDR value (i.e. $\epsilon = 2.55$ for a 8 bit LDR layer). Compared to the theoretical result in equation (5.12), this prediction of X_r and X_b reduces the noise in dark regions where the ratios $\frac{R_{LDR}}{G_{LDR}}$ and $\frac{B_{LDR}}{G_{LDR}}$ might be too sensitive to small color errors caused by the lossy compression of the LDR layer. Equation (5.17) also avoids the risk of division by zero. The actual HDR RGB prediction is then computed from the decoded LDR RGB components and the decoded HDR luma Y' using the following equation :

$$\begin{aligned} G_{pred} &= \frac{PQ^{-1}(Y')}{\left(\alpha_0 \cdot \widetilde{X_r}^{\frac{1}{p}} + \alpha_1 + \alpha_2 \cdot \widetilde{X_b}^{\frac{1}{p}}\right)^p} \\ R_{pred} &= \widetilde{X_r} \cdot G_{pred} \\ B_{pred} &= \widetilde{X_b} \cdot G_{pred} \end{aligned} \tag{5.18}$$

The Cb and Cr components are finally predicted by applying back the PQ-OETF to R_{pred}, G_{pred} and B_{pred} and by computing the chroma.

5.3.3 Implementation details

In both the Y'CbCr and the u"v" encoding schemes, the prediction of the chromatic components is based on the decoded LDR RGB components and the HDR luma. In our implementation, for a given block in the image, the luma block is always encoded and decoded before the chromatic components. As a result, the decoded luma block is known while encoding or decoding the u" and v" blocks. However, since the color components are downsampled horizontally and vertically, the same downsampling must be performed to the decoded luma channel. We used a simple downsampling scheme consisting in taking the mean of the four luma pixels collocated with a given chroma pixel. In the Y'CbCr encoding scheme, the inverse PQ-OETF is applied after the luma downsampling for the computation of $\widetilde{Y}$.

Similarly, the LDR RGB components must be given in low resolution for performing the prediction. Since the LDR layer is originally encoded in the Y'CbCr 4:2:0 format, only the LDR luma needs to be downsampled. The low resolution LDR luma and chroma are then converted to RGB.

5.4 Pre-analysis

In general, we cannot assume that the parameters s and γ used in the prediction model are known in advance. A first step thus consists in determining the parameters that best fit the HDR and LDR image pair. This can be done in a preprocessing stage before encoding. Therefore, in this section all the computation can be performed using the original LDR and HDR images without compression. From the prediction equations in section 5.3, we note that only the ratio $s' = \frac{s}{\gamma}$ must be determined.

From the color model in equation (5.3), we directly obtain :

$$f(Y)^{\frac{1}{\gamma}} = R_{LDR} \cdot \left(\frac{Y}{R}\right)^{s'} = G_{LDR} \cdot \left(\frac{Y}{G}\right)^{s'} = B_{LDR} \cdot \left(\frac{Y}{B}\right)^{s'} \tag{5.19}$$

Thus, we want to find the value of s' that minimizes the mean square error (MSE) on all the pixels. For simplicity, only the red and green components are used in our minimization problem. For natural content, no difference was observed when the blue component was taken into account. Given a pixel i, let us define the function F^i as

$$F^i(s') = \left(R^i_{LDR} \cdot \left(\frac{Y^i}{R^i}\right)^{s'} - G^i_{LDR} \cdot \left(\frac{Y^i}{G^i}\right)^{s'}\right)^2 , \tag{5.20}$$

where R^i_{LDR}, G^i_{LDR}, R^i and G^i are respectively the values of R_{LDR}, G_{LDR}, R and G at pixel position i.

The problem to solve is then expressed as

$$\widehat{s'} = \underset{s'}{\operatorname{argmin}} \sum_{i=1}^{n} F^i(s') , \tag{5.21}$$

where n is the number of pixels. The problem in equation (5.21) can be solved by finding the value of s' for which $\sum_{i=1}^{n} F^{i'}(s') = 0$, where $F^{i'}$ denotes the first derivative of F^i. Newton's iterative numerical method was used for that purpose. Given an initialization value $s'_0 = 0.4$, the value s'_k at iteration k is given by :

$$s'_k = s'_{k-1} - \frac{\sum_{i=1}^{n} F^{i'}(s'_{k-1})}{\sum_{i=1}^{n} F^{i''}(s'_{k-1})} \tag{5.22}$$

where the two first derivatives $F^{i'}$ and $F^{i''}$ can be determined analytically as

$$\begin{aligned} F^{i'}(s') = {} & A^i_{11} \cdot \left(\frac{Y^i}{R^i}\right)^{2s'} + A^i_{12} \cdot \left(\frac{Y^i}{G^i}\right)^{2s'} \\ & + A^i_{13} \cdot \left(\frac{(Y^i)^2}{R^i \cdot G^i}\right)^{s'} \end{aligned} \tag{5.23}$$

$$\begin{aligned} F^{i''}(s') = {} & A^i_{21} \cdot \left(\frac{Y^i}{R^i}\right)^{2s'} + A^i_{22} \cdot \left(\frac{Y^i}{G^i}\right)^{2s'} \\ & + A^i_{23} \cdot \left(\frac{(Y^i)^2}{R^i \cdot G^i}\right)^{s'} \end{aligned} \tag{5.24}$$

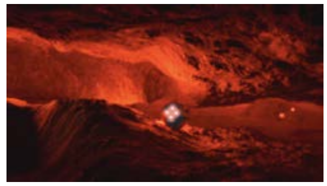
with

$$\begin{aligned} A^i_{11} &= 2 \cdot \ln\left(\frac{Y^i}{R^i}\right) \cdot \left(R^i_{LDR}\right)^2 \\ A^i_{12} &= 2 \cdot \ln\left(\frac{Y^i}{G^i}\right) \cdot \left(G^i_{LDR}\right)^2 \\ A^i_{13} &= -2 \cdot \ln\left(\frac{(Y^i)^2}{R^i \cdot G^i}\right) \cdot R^i_{LDR} \cdot G^i_{LDR} \\ A^i_{21} &= A^i_{11} \cdot 2 \cdot \ln\left(\frac{Y^i}{R^i}\right) \\ A^i_{22} &= A^i_{12} \cdot 2 \cdot \ln\left(\frac{Y^i}{G^i}\right) \\ A^i_{23} &= A^i_{13} \cdot \ln\left(\frac{(Y^i)^2}{R^i \cdot G^i}\right) \end{aligned}$$

The iterative process in equation (5.22) is stopped when the difference between the value of s' at two successive iterations is less than 10^{-4}. In our experiments, we observed a fast convergence and three iterations were usually sufficient to reach the precision of 10^{-4}.

In order to increase the robustness of the method, some pixels are removed from the sums in equation (5.22). First, the pixels for which at least one of the HDR RGB


(a) Market3


(b) Tibul2


(c) StEM WarmNight 1


(d) StEM WarmNight 2

(e) Balloon

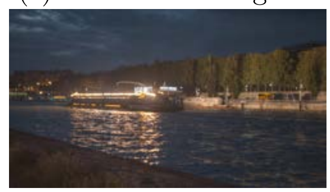

(f) Seine

Figure 5.5: First frames of each sequence used in the experiment. The LDR versions are represented.

components is less than a threshold of 0.02 are removed. Those pixels are not reliable because of the color noise that may appear in very dark regions. Moreover, too small R^i, G^i or B^i values can cause inaccurate computations of $F^{i'}$ and $F^{i''}$ in equations (5.23) and (5.24). A second type of problem may appear for too bright pixels. In practice, after tone mapping, some pixel's RGB values may exceed the maximum LDR value for one or several RGB components. A simple clipping operation is generally applied in this case in order to keep all the values in the LDR range. However, since this operation is performed on the RGB channels independently, it modifies the hue of the clipped pixels. Therefore, the assumption of preservation of the hue in the model is no longer satisfied. For that reason, we exclude from the computation all the pixels that exceed 99% of the maximum LDR value in at least one of the components R_{LDR}, G_{LDR} or B_{LDR}.

5.5 Experimental Results

For our experiment, we have used six HDR test sequences presented in table 5.1. Their spatial resolution is 1920x1080 pixels. The sequences Market3, Tibul2 and StEM WarmNight are parts of the MPEG standard sequences for HDR scalability [95]. Note that StEM WarmNight is originally one sequence containing two shots. In our experiments it was separated into two sequences. The sequences Balloon and Seine were produced by Binocle and Technicolor within the framework of the french collaborative project NEVEx.

For the base layer of the sequences Market3, Tibul2 and StEM WarmNight, LDR versions generated by a manual color grading process were already provided in the MPEG set of sequences. The LDR versions of Balloon and Seine were generated with the local TMO described in [85] and implemented in the publicly available pfstmo library [89]. This TMO explicitly uses the color correction of Tumblin and Turk with the saturation parameter s. The default value of 0.8 was kept for this parameter and

Sequence	Frames	Frame Rate	Intra period
Market3	0-48	50	48
Tibul2	0-32	30	32
StEM WarmNight 1	3527-3551	24	24
StEM WarmNight 2	3729-3753	24	24
Balloon	0-24	25	24
Seine	0-24	25	24

Table 5.1: HDR Sequences used for the tests.

the tone mapped image was further gamma corrected with a gamma of $\frac{1}{2.2}$. It should be noted that in this case, the theoretical value of the ratio $s' = \frac{s}{\gamma}$, which is required by the encoder, is known in advance. In order to test the accuracy of the pre-analysis step defined in section 5.4, we applied our algorithm to each HDR frame with its corresponding LDR frame. For both the sequences Balloon and Seine, the theoretical value of s' was recovered with the required precision of 10^{-4} at each frame.

Figure 5.5 shows the LDR versions of the first frame of each sequence used as a base layer. For the sake of simplicity, the RGB colorspace of both the LDR and HDR versions are defined with the standard BT.709 color primaries.

In our experiments, we have compared the Y'CbCr and the u"v" schemes. A first remark can be made concerning the downsampling of the chromatic components prior to the HEVC encoding. An example of downsampling in each colorspace is shown in figure 5.6. It can be seen in figure 5.6(b) that the chroma downsampling in Y'CbCr may cause disturbing artifacts in areas containing saturated colors. This is due to the highly non-linear OETF applied independently to the RGB components before the conversion to Y'CbCr. Because of this non-linearity, a part of the luminance information is contained in the Cb and Cr component, and conversely, the luma channel Y' is influenced by the chromaticity. As a result, the chroma downsampling causes errors in the luminance of the reconstructed image which are visually more significant than errors in colors. This problem does not occur in the u"v" based colorspace since the luminance is encoded separately.

In addition to the two algorithms presented in this chapter, we have made simulations with several other compression methods. Simulcast encoding (i.e. independent encoding of the LDR and HDR layers) was performed for both Y'CbCr and u"v" color representations in order to evaluate the rate distortion gains obtained by adding our ILP mode. We have also compared our color inter-layer prediction to the template based local ILP method presented in chapter 4, which is also used here for the luma channel. However, in chapter 4, the chroma was also predicted with the local ILP. Finally, the prediction of the u'v' components used in Mantiuk et al's method in [62] was also tested. For a fair comparison with our method, the luma channel is predicted with the same local ILP in our implementation of their method, in place of the global prediction scheme described in [62]. In practice, our implementation of this method is very close to our u"v" scheme. The main difference is that the value of s in equation (5.9) was

(a) Original (b) $Y'CbCr$ (c) $Y_{PQ}u''v''$

Figure 5.6: Detail of a frame in the sequence Market3. (a) The original image. (b) and (c) Reconstructed image after an encoding in respectively $Y'CbCr$ and $Y_{PQ}u''v''$ colorspaces with 420 chroma downsampling in both cases.

fixed to 1 in order to obtain $u'_{pred} = u'_{LDR}$ and $v'_{pred} = v'_{LDR}$. The value of γ was set to 2.2 which corresponds to typical gamma correction. Furthermore, [62] directly uses the CIE u'v' as color components. We have thus disabled our modification of the u'v' coordinates by setting to 0 the threshold value Y_{th} defined in equation (5.1).

For the simulations, the encoding with our modified version of HEVC was performed with random access configuration using groups of pictures (GOPs) of 8 pictures. The period of intra frames for each sequence is given in table 1. It was chosen depending on the frame rate to correspond to approximately 1 second for each sequence.

5.5.1 Quality assessment

For assessing the quality of the decoded HDR images, we have chosen to use separate indices for the quality of the luminance signal which is achromatic and that of the chromaticity signal. The reason of this choice is that most of the existing quality metrics do not accurately account for color vision.

For instance, a common method for assessing the quality of compressed images consists in computing the peak signal-to-noise ratio (PSNR) of each of the Y'CbCr components, and combining the results by a weighted sum. Alternatively, the PSNR can be computed from the perceptually quantized $R'G'B'$ components. However, the colorspaces formed by the $R'G'B'$ or by the Y'CbCr components only give a rough approximation of perceptual uniformity. It is particularly inaccurate in highly saturated colors, especially in the case of HDR images. Although a PSNR could be computed based on the CIE ΔE_{2000} color difference formula [96] which estimates well the perceived difference between two colors, this formula is only accurate with LDR data for which it was designed. Furthermore, new metrics have been developed specifically for HDR quality assessment, the most well-known being the High Dynamic Range Visual Difference Predictor (HDR-VDP) [91]. However, they only predict luminance differences and do not consider color.

In our experiment, the quality of the luminance was assessed using the peak signal-to-noise ratio (PSNR) computed on the perceptually encoded luminance with the PQ-EOTF curve. This quality index is referred to as PSNR Y_{PQ} in the rest of the chapter. The quality of the chromatic signal was assessed based on the CIE 1976 L*a*b* colorspace. A PSNR value is computed with equation (5.25) using only the chromatic components a* and b*. Note that the L* component is a non-linear function of the luminance. This non-linearity was determined with the aim of perceptual uniformity by experiments based on stimuli of relatively low luminance. It is therefore only perceptually uniform for LDR data but can be very inaccurate for modelling human perception with HDR images. For this reason, the L* component was excluded from our index in equation (5.25), and only the chromatic information contained in the a* and b* components was taken into account.

$$PSNR_{a^*b^*} = 10 \cdot log_{10}\left(\frac{1000^2}{(MSE_{a^*} + MSE_{b^*})/2}\right) \tag{5.25}$$

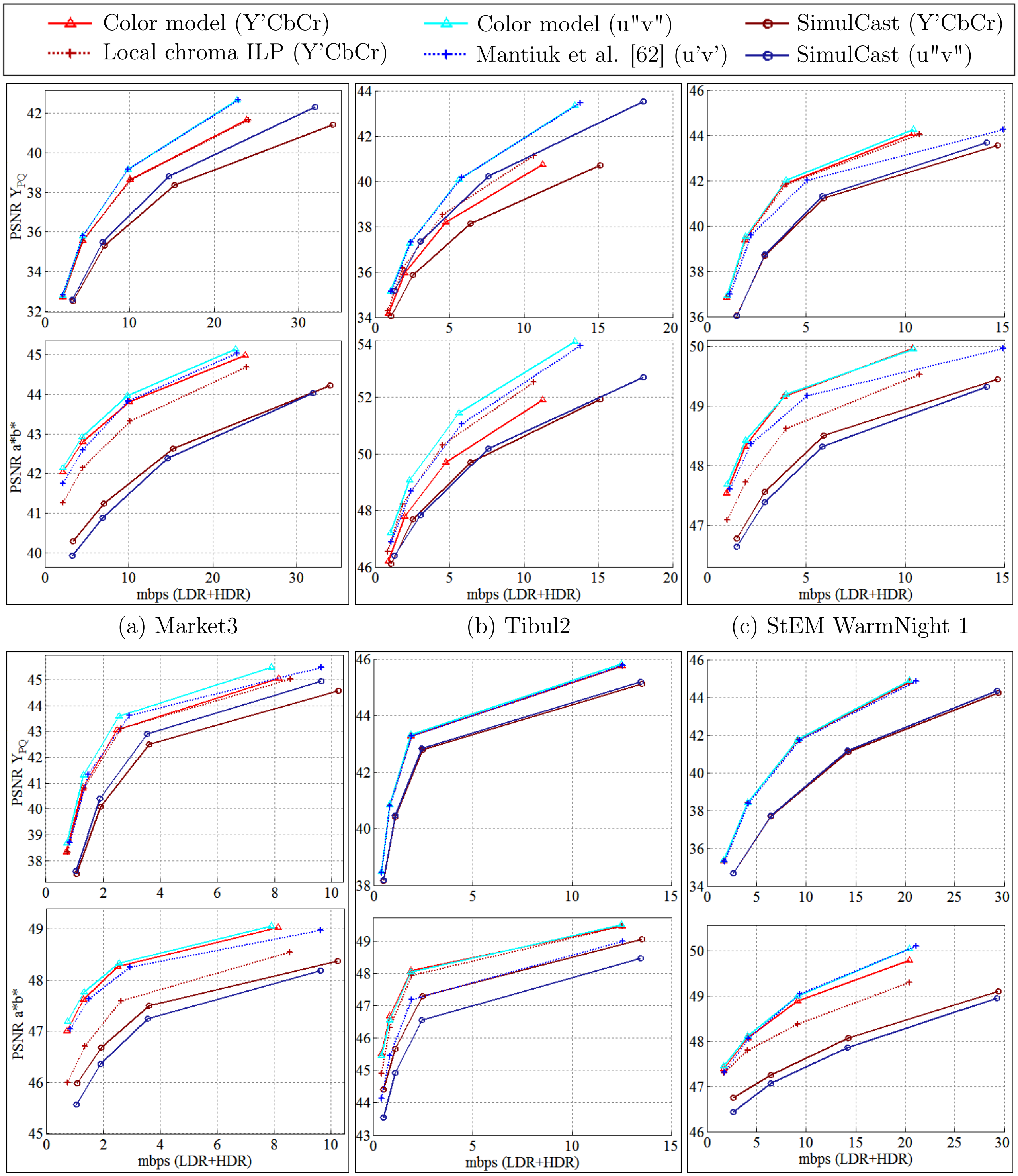
where MSE_{a^*} and MSE_{b^*} are the mean square errors for the a* and b* components respectively.

5.5.2 Rate Distortion results

For each sequence and tested method, two rate distortion curves are shown in figure 5.7, using either the distortion in luminance (i.e. achromatic), or the chromatic distortion index defined in equation (5.25). The curves were generated by encoding each sequence at varying qualities obtained with QP parameter values of 22, 27, 32, 37. Both the LDR and HDR layers were encoded with the same QP value so that both layers are of comparable quality. We can first note from the curves that the best compression performance, considering both the chromatic and the achromatic quality indices, is obtained with our u"v" compression scheme for all the sequences.

In order to quantify our gains in comparison to other methods, we have computed the Bjontegaard Delta Rate metric [76] from the luminance distortion index and the total bitrate of all the components of both the HDR and LDR layers. Since our study focuses on the coding of the chromatic components, evaluating the rate gains only from a luminance based quality index is not enough. Therefore, we also computed the Bjontegaard Delta PSNR using the PSNR a*b* and the total bitrate.

The gains of our u"v" scheme were computed with respect to Mantiuk et al's u'v' predictions and are reported in table 5.2. Note that fairly low rate gains are observed for most sequences since we considered only the quality of the luminance for the computation of the Delta Rate metric. This is explained by the fact that the luminance was



(a) Market3 (b) Tibul2 (c) StEM WarmNight 1

(d) StEM WarmNight 2 (e) Balloon (f) Seine

Figure 5.7: Rate-Distortion Curves. For each sequence, the luminance distortion is represented in the upper graph while the chromatic distortion is shown in the lower graph. The x-axis represents the total bitrate for all the components of both the HDR and LDR layers.

Method Tested :	Ours (u"v")	
Metric :	ΔRate	ΔPSNR a*b*
StEM WarmNight 1	-17.2 %	0.23 dB
StEM WarmNight 2	-12.0 %	0.21 dB
Market3	-0.5 %	0.23 dB
Tibul2	-1.1 %	0.40 dB
Balloon	-2.7 %	0.85 dB
Seine	-3.2 %	0.04 dB
Average	-6.1 %	0.33 dB

Table 5.2: Bjontegaard gains of our u"v" scheme with respect to Mantiuk et al. color prediction [62]. The rate gains are computed from the PSNR Y_{PQ} quality index. The total bitrate of all the components of both layers is considered.

encoded the same way in our implementation of both methods in order to compare only the inter-layer prediction of the chromatic components. In the case of the sequences StEM WarmNight 1 and 2, however, our method shows significant gains in luminance of respectively 17.2% and 12% in comparison to Mantiuk et al's version. This is due to the color noise contained in the dark regions of those sequences which is better quantized by using the modified u"v" than with the original CIE u'v' color components, resulting in lower overall bitrate. Additionally, our method is more reliable for the coding of colors because of the parameter s' optimized for the image's content. In particular, for the sequence Balloon, a gain of 0.85 dB in $PSNR_{a*b*}$ is observed between our color ILP and that of Mantiuk et al.

Similarly, we have compared our Y'CbCr scheme with the local prediction of chapter 4 which also uses a Y'CbCr encoding, but where the Cb and Cr components are predicted with the same local ILP method than the luma component Y'. The Bjontegaard

Method Tested :	Ours (Y'CbCr)	
Metric :	ΔRate	ΔPSNR a*b*
StEM WarmNight 1	-2.0 %	0.55 dB
StEM WarmNight 2	-3.8 %	0.74 dB
Market3	-1.1 %	0.57 dB
Tibul2	17.1 %	-0.64 dB
Balloon	-0.8 %	0.19 dB
Seine	-0.2 %	0.36 dB
Average	1.5 %	0.30 dB

Table 5.3: Bjontegaard gains of our Y'CbCr scheme with respect to the template based local ILP. The rate gains are computed from the PSNR Y_{PQ} quality index. The total bitrate of all the components of both layers is considered.

Method Tested :	Ours (Y'CbCr)		Ours (u"v")	
Metric :	ΔRate	ΔPSNR a*b*	ΔRate	ΔPSNR a*b*
StEM WarmNight 1	-46.0 %	1.15 dB	-47.5 %	1.17 dB
StEM WarmNight 2	-43.6 %	1.14 dB	-50.6 %	1.21 dB
Market3	-39.6 %	1.97 dB	-45.9 %	2.22 dB
Tibul2	-26.7 %	0.66 dB	-50.6 %	1.84 dB
Balloon	-37.9 %	1.09 dB	-38.9 %	1.09 dB
Seine	-45.6 %	1.22 dB	-46.0 %	1.33 dB
Average	-39.9 %	1.21 dB	-46.6 %	1.48 dB

Table 5.4: Bjontegaard gains with respect to Y'CbCr Simulcast. The rate gains are computed from the PSNR Y_{PQ} quality index. The total bitrate of all the components of both layers is considered.

gains are presented in table 5.3. It can be seen that in most cases, only small rate gains of up to 3.8% are obtained by considering only the PSNR of the perceptually encoded luminance. However, substantial gains are observed for our method when evaluating the chromatic distortions. In particular, for the sequences StEM WarmNight 1, StEM WarmNight 2 and Market3, whose LDR versions were produced with a manual color grading process, gains of 0.55 dB, 0.74 dB and 0.57 dB respectively are observed in PSNR a*b*. It shows that the color model used in our scheme estimates well the relationship between the colors of the LDR and the HDR images even though the LDR versions were not explicitly generated with the color correction of Tumblin and Turk. The exception, however, is the sequence Tibul2 which has an overall very saturated red color. Since the prediction equations of our Y'CbCr scheme are based on the approximation in equation (5.15), they may produce artifacts in those extreme colors as in the example of figure 5.4. Despite the parameter p, introduced in order to reduce those artifacts, our method remains less efficient than the local ILP when the whole sequence contains very saturated colors.

Finally, table 5.4 shows the Bjontegaard gains with respect to the Simulcast method with Y'CbCr encoding. On average, almost 40% of the bitrate is saved at equal luminance quality by using our Y'CbCr inter-layer prediction scheme. Better compression performance is obtained with our u"v" scheme which reaches an average 46.6% gain. Higher gains are also observed for the u"v" version by considering the chromatic quality index, PSNR a*b*. Note that the highest gains of the u"v" ILP are obtained with the sequence Tibul2, which shows that this color representation better handles the difficult case of highly saturated colors. This is in accordance with the fact that, unlike the Y'CbCr scheme, no approximation was necessary for the derivation of the u"v" prediction equations.

5.6 Conclusion

In the context of the scalable compression of HDR content with a LDR base layer, we have developed a new inter-layer prediction method specifically for the chromatic components. Our method is based on a model linking the colors in the HDR layer to those in the LDR layer. In particular, it follows the general assumption that the hues of the colors in an HDR image were preserved in the LDR version. In addition, the model uses a single parameter to adjust the saturations of the HDR colors in the prediction. A pre-analysis step was developed in order to determine the optimal value for this parameter given an HDR image and its associated LDR version. From the model, we derived prediction equations for two encoding schemes using different color representations of the images. In the first scheme, the classical Y'CbCr encoding is addressed while the second version uses a colorspace built from the luminance and the CIE u'v' color coordinates.

Our results are advocating for the use of the CIE u'v' based colorspace, which completely decorrelates the luminance and chrominance signals. This property enables a better downsampling of the chromatic components than the usual chroma downsampling in a Y'CbCr colorspace. Moreover, the u'v' components can be predicted more accurately from the color model than the CbCr components. We have also demonstrated that, thanks to the saturation parameter in the model, our u'v' inter-layer prediction generalizes the existing color ILP methods in the literature that uses the same u'v' representation. The experiments have confirmed that the use of the optimized saturation parameter improved the coding performance.

Regarding the coding in Y'CbCr colorspace, our ILP sheme based on the color model also shows better coding performances in most cases in comparison to other methods which directly predict the HDR layer's chroma components from those of the LDR layer.

Conclusion

In this thesis, we explored the backward compatible approaches for the compression of High Dynamic Range content. While conventional digital image formats define pixel's colors by 8-bit integer values per RGB component, this is not sufficient to represent with precision the range of luminance that the human eye can perceive. Starting from floating point numbers corresponding to physical measures of luminance and color, we have seen, along this thesis, several ways of defining an integer format for HDR data by taking into account the principles of perceptual uniformity. In chapter 3, an approximate logarithmic mapping was used for that purpose. This encoding enables a fast and lossless conversion from the half float format to 15-bit integers, but has rather inaccurate perceptual uniformity. In chapter 4, a better perceptual curve from the recent literature was used to reduce the bitdepth of the RGB components to 12 bits without visible artifact. In both chapters, the resulting R'G'B' integer components were converted to a Y'CbCr colorspace that separates the luma Y', approximating the brightness level, from the chroma components CbCr. This Y'CbCr conversion step generally followed by a downsampling of the chroma, for which the human eye is less sensitive, is the most common way of encoding image data in the field of compression. In the case of HDR images, however, some artifacts caused by the chroma downsampling were identified in chapter 5. An alternative colorspace based on the CIE standard u'v' chromaticity scale was used successfully in this chapter for solving this issue.

All the HDR color encodings have in common that they require more than the traditional 8 bit integer data and are therefore not compatible with existing decoders and displays. Backward compatible compression schemes must then be used to enable legacy equipment to read a low dynamic range version of the HDR content. In a first approach studied in chapter 3, we reduced the bitdepth at the input of the encoder and applied the inverse operation to the decoded image. Two methods were presented to adapt this quantization step to the content of the image in order to optimize the rate distortion performance. First considering a uniform quantization scheme based on minimum and maximum values of the input data, we have shown how a regional adaptation could be used to address different applications. While a block-wise quantization is optimal for high or very high bitrate encoding, frame-wise or even GOP-wise methods, which preserve respectively spatial and temporal coherence, are better suited to applications requiring lower bitrate encodings. Then, we derived the optimal non-uniform quantization scheme based on the probability distribution of the HDR content and the QP setting of an HEVC encoder. Those methods are fast and easy to implement since they require only existing low bitdepth codecs. However, backward compatibility is not fully addressed since the low bitdepth image is generated with an automatic process that only takes compression performance into account. Displaying that image directly on a LDR screen might not give satisfactory results, especially in the case of a block-wise quantization which may result in large gaps in brightness between blocks. Further processing is then required on the LDR image for a better rendering.

Our next contributions were then directed towards scalable schemes that encode the LDR and the HDR versions in separate layers without requiring prior knowledge on the tone mapping operator used. The choice of the TMO then belongs to the producer. The scalable encoder decorrelates the LDR and HDR signals by a prediction of the HDR layer from the already encoded and decoded LDR layer. Our inter-layer prediction scheme proposed in chapter 4 successfully handles the case of local TMOs by automatically learning a non-linear inverse tone curve for each block of the image. The curve is learnt only from already decoded HDR and LDR data in a template (i.e. neighborhood) of the block, so that the decoder can determine the same curve without the transmission of parameters. This template based technique enables significant coding gains on the HDR layer compared to the existing local inter-layer prediction methods which does not use the template and only perform linear predictions. Further coding gains were obtained by sending a multiplication factor to the decoder for re-scaling the predicted block in the cases where the inverse tone curve derived from the template is not accurate enough.

Finally, we have shown in chapter 5 that a specific inter-layer prediction strategy could be defined for the chromatic components. A model describing the relationship between the LDR and HDR luminance and RGB components was determined from color correction equations used in many tone mapping operators. This correction was designed to keep consistent color hues and saturations between the HDR and the tone mapped versions. Chromatic inter-layer prediction could then be derived either for the CbCr components in the case of a Y'CbCr encoding of the HDR layer, or for the u'v' components in an alternative color encoding scheme. Even in cases where the LDR version was generated by a manual process which does not explicitly include the color correction on which our model is based, this inter-layer prediction method was found to give more accurate results than the template based local prediction of the chromatic components.

Future work and perspectives

Several approaches can be considered to extend the work of this thesis.

First, in the single-layer approach with the optimal tone curve, compression performance was optimized for the encoding of an image with only a luma channel. Further research is necessary to determine the best quantization technique for the chroma. Computing a curve with the same technique and same lagrangian parameter is unlikely to give optimal results since the luma and chroma signals have very different distributions. It is expected that the chroma components will be quantized with too much precision compared to the luma. A different lagrangian parameter should then be used for each component.

In terms of quality assessment, the problem of color also arises. The ΔE_{2000} color difference formula which is perceptually uniform for LDR data has no equivalent in HDR. Even advanced metrics for high dynamic range images such as HDR-VDP do not take this aspect into consideration. In the last chapter of this thesis color quality assessment was required since the proposed inter-layer prediction was designed for the chromatic components. Our solution was to use two separate indices to assess the quality of luminance and chrominance. However, we are aware that the relevance of our chrominance index and the relative importance of both indices remains an open question. That makes the interpretation of rate distortion results sometimes difficult. Performing subjective evaluations would be necessary to quantify more accurately the gains of our method.

Besides the problem of quality assessment, the flexibility of our inter-layer color prediction might be extended in order to address a larger range of cases more accurately. Although our prediction is based on a widely used color correction, other color correction methods for the tone mapping exist, as presented in [97] or [98] for example. Without changing our model, an adjustment of the saturation parameter for different regions of the image would be a point to consider. Such improvements may also be beneficial in the case of a manually tone mapped LDR version.

With regard to complexity, improvements could be performed on our scalable encoding method. The decoding times of our implementation in chapter 4 was shown to be twice as long as intra decoding. The most time consuming step in this prediction algorithm is the linear least square minimization which is based on floating point calculations. The implementation of an integer based least square solving process could be considered to approximate the results obtained in our method with reduced computation times. Further studies would then be needed to find a reasonable tradeoff between the accuracy of the minimization and the computation times.

Finally, the tools developed in our local inter layer-prediction scheme are rather generic and might also serve in the more general context of inter predictions (i.e. temporal predictions). Since inter predictions are based on previously encoded frames of the sequence, our method could be used to predict local illumination changes between frames. In these conditions, the prediction adjustment factor method could also be beneficial as we have seen that it may decrease the decoding complexity and improve the quality of the predictions in the favorable cases without affecting the overall performance in the worst case. New applications might also benefit from these tools. For instance, in cloud-based compression [99], an image is encoded from related images previously stored in a large database. In this case, the reference image used for predictions has similar content, but the pictures may have been taken in different illumination conditions and at different angles. Our approach could then be extended to solve the problem of illumination differences.

Author's publications

Articles

Conference papers

M. Le Pendu, C. Guillemot and D. Thoreau, "Adaptive Re-Quantization For High Dynamic Range Video Compression", *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pp. 7367-7371, May 2014.

M. Le Pendu, C. Guillemot and D. Thoreau, "Rate Distortion Optimized Tone Curve for High Dynamic Range Compression", *22nd European Signal Processing Conference (EUSIPCO)*, pp. 1612-1616, Sep. 2014.

M. Le Pendu, C. Guillemot and D. Thoreau, "Template based Inter-Layer Prediction for High Dynamic Range Scalable compression", *IEEE International Conference on Image Processing (ICIP)*, Sep. 2015.

Journal paper

M. Le Pendu, C. Guillemot and D. Thoreau, "Local Inverse Tone Curve Learning for High Dynamic Range Image Scalable Compression", *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5753-5763, Dec. 2015.

Under review

M. Le Pendu, C. Guillemot and D. Thoreau, "Inter-Layer Prediction of Color in High Dynamic Range Image Scalable Compression", *submitted to IEEE Transaction on Image Processing in October 2015.*

Patents

M. Le Pendu, D. Thoreau, C. Guillemot and M. Alain, "Method for encoding and decoding an image block based on dynamic range extension, encoder and decoder", *EP2958329 / US2015365701*, 2015.

M. Le Pendu, R. Boitard, D. Thoreau and C. Guillemot, "Method and apparatus for encoding and decoding HDR images", *EP2927865 / WO2015128295*, 2014.

M. Le Pendu, D. Thoreau, C. Guillemot, R. Boitard and F. Le Léannec, "Method and device for predictive coding/decoding of a group of pictures of a sequence of images with conversion of the dynamic of the values of the pixel", *WO2015062942*, 2013.

D. Thoreau, M. Alain, M. Le Pendu and F. Le Léannec, "Procédé et dispositif de codage d'images vidéo, procédé et dispositif de décodage d'un flux de données, programme d'ordinateur et support de stockage correspondants", *FR3012935*, 2014.

D. Thoreau, M. Le Pendu, Y. Olivier and C. Guillemot, "Method for coding and decoding floating data of an image block and associated devices", *WO2015052064*, 2013.

(5 other patents were filed but not yet disclosed and 5 other patent applications are pending)

Appendix A

Mathematical derivations for Rate-Distortion optimized tone curve

A.1 Positive roots of the cubic equation

In this section, we determine the positive roots of the cubic equation

$$X^3 + aX^2 - b = 0, \tag{A.1}$$

in the two cases where a and b are either both positive or both negative.

Let us change the variable as $X = Z - \frac{a}{3}$. By substituting X by Z, and developing the expression, we obtain:

$$Z^3 + pZ + q = 0 \tag{A.2}$$

where $p = -\frac{a^2}{3}$ and $q = \frac{2a^3}{27} - b$. Cardano's method can be used to solve this form of cubic equation. The general solution depends on the sign of $\Delta = q^2 + \frac{4p^3}{27}$. Δ can also be expressed with a and b as $\Delta = -\frac{4a^3b}{27} + b^2$

- if $\Delta > 0$: there is only one real root to equation (A.1):

$$\begin{aligned} X_1 &= \sqrt[3]{\frac{-q + \sqrt{\Delta}}{2}} + \sqrt[3]{\frac{-q - \sqrt{\Delta}}{2}} - \frac{a}{3} && \text{(A.3)} \\ &= \sqrt[3]{\frac{b}{2} - \frac{a^3}{27} + \sqrt{\frac{b^2}{4} - \frac{a^3b}{27}}} + \sqrt[3]{\frac{b}{2} - \frac{a^3}{27} - \sqrt{\frac{b^2}{4} - \frac{a^3b}{27}}} - \frac{a}{3} && \text{(A.4)} \end{aligned}$$

- if $\Delta \leq 0$: there are three real roots to equation (A.1):

$$\begin{cases} X_1 = & 2\sqrt{-\frac{p}{3}}\cos\left(\frac{\arccos\left(\frac{3q}{2p}\sqrt{\frac{-3}{p}}\right)}{3}\right) - \frac{a}{3} \\ X_2 = & 2\sqrt{-\frac{p}{3}}\cos\left(\frac{\arccos\left(\frac{3q}{2p}\sqrt{\frac{-3}{p}}\right) + 2\pi}{3}\right) - \frac{a}{3} \\ X_3 = & 2\sqrt{-\frac{p}{3}}\cos\left(\frac{\arccos\left(\frac{3q}{2p}\sqrt{\frac{-3}{p}}\right) + 4\pi}{3}\right) - \frac{a}{3} \end{cases} \tag{A.5}$$

A.1.1 case 1 : $a > 0$ and $b > 0$

Let us now consider the first case : $a > 0$ and $b > 0$. We need to determine the sign of the roots. For $b > 0$, the condition $\Delta > 0$ is equivalent to $b > \frac{4a^3}{27}$.

- If $\Delta > 0$, the root X_1 is given by equation (A.4). It can be rearranged by noticing that $\frac{b^2}{4} - \frac{a^3 b}{27} = \left(\frac{b}{2} - \frac{a^3}{27}\right)^2 - \left(\frac{a^3}{27}\right)^2$. By defining $m = \dfrac{b}{2} - \dfrac{a^3}{27}$ and $n = \dfrac{a^3}{27}$, we obtain:

$$X_1 = \sqrt[3]{m + \sqrt{m^2 - n^2}} + \sqrt[3]{m - \sqrt{m^2 - n^2}} - \sqrt[3]{n} \tag{A.6}$$

  Knowing that $b > \frac{4a^3}{27}$, we have $m > n > 0$. Hence, $m - n < \sqrt{m^2 - n^2} < m$. As a result $X_1 > \sqrt[3]{2m - n} - \sqrt[3]{n}$. And since $2m - n > n$, we find that $X_1 > 0$.

- If $\Delta \leq 0$ the roots given by equation (A.5) can be written with respect to only a and b as

$$\begin{cases} X_1 = & \frac{a}{3}\left(2\cos\left(\frac{\arccos\left(\frac{27b}{2a^3} - 1\right)}{3}\right) - 1\right) \\ X_2 = & \frac{a}{3}\left(2\cos\left(\frac{\arccos\left(\frac{27b}{2a^3} - 1\right) + 2\pi}{3}\right) - 1\right) \\ X_3 = & \frac{a}{3}\left(2\cos\left(\frac{\arccos\left(\frac{27b}{2a^3} - 1\right) + 4\pi}{3}\right) - 1\right) \end{cases} \tag{A.7}$$

  since $0 < b \leq \frac{4a^3}{27}$, we have $\frac{27b}{2a^3} - 1 \in]-1, 1]$ and thus $\arccos\left(\frac{27b}{2a^3} - 1\right) \in [0, \pi[$. Hence, $\cos\left(\dfrac{\arccos\left(\frac{27b}{2a^3} - 1\right)}{3}\right) \in]0.5, 1]$. Since $a > 0$, we find that $X_1 > 0$. In the same way, we find that $X_2 < 0$ and $X_3 < 0$.

Therefore, when both a and b are positive, there is a unique positive root X_1 to equation (A.1):

$$X_1 = \begin{cases} \sqrt[3]{m + \sqrt{m^2 - n^2}} + \sqrt[3]{m - \sqrt{m^2 - n^2}} - \sqrt[3]{n}, & \text{if } m > n \\ \frac{a}{3}\left(2\cos\left(\frac{\arccos\left(\frac{27b}{2a^3} - 1\right)}{3}\right) - 1\right), & \text{otherwise} \end{cases} \tag{A.8}$$

where $m = \dfrac{b}{2} - \dfrac{a^3}{27}$ and $n = \dfrac{a^3}{27}$.

A.1.2 case 2 : $a < 0$ and $b < 0$

Since $b < 0$, the condition $\Delta > 0$ is now equivalent to $b < \frac{4a^3}{27}$.

- if $\Delta > 0$, the unique root is given by equation (A.6). However, for $a < 0$, and $b < \frac{4a^3}{27} < 0$, we have $m < n < 0$. Hence, $0 < \sqrt{m^2 - n^2} < n - m$. Thus, $\sqrt[3]{m + \sqrt{m^2 - n^2}} < \sqrt[3]{n}$ and $\sqrt[3]{m - \sqrt{m^2 - n^2}} < 0$.

  As a result $X_1 < \sqrt[3]{n} + 0 - \sqrt[3]{n} = 0$.

- if $\Delta \leq 0$, the roots are given by:

$$\begin{cases} X_1 = & -\dfrac{a}{3}\left(2\cos\left(\dfrac{\arccos\left(1 - \frac{27b}{2a^3}\right)}{3}\right) + 1\right) \\ X_2 = & -\dfrac{a}{3}\left(2\cos\left(\dfrac{\arccos\left(1 - \frac{27b}{2a^3}\right) + 2\pi}{3}\right) + 1\right) \\ X_3 = & -\dfrac{a}{3}\left(2\cos\left(\dfrac{\arccos\left(1 - \frac{27b}{2a^3}\right) + 4\pi}{3}\right) + 1\right) \end{cases} \tag{A.9}$$

  Knowing that $a < 0$, we can determine that $X_1 > 0$, $X_2 < 0$ and $X_3 > 0$.

In conclusion, when $a < 0$ and $b < 0$, the cubic equation (A.1) has no positive root if $b < \dfrac{4a^3}{27}$, and two positive roots otherwise.

A.2 Existence of the solution

In this section we want to determine if a solution exists for the minimization of the cost in equation (3.15). We have seen in 3.3.2.2 that the existence of a solution is equivalent to the following statement:

$$\exists c \in \mathbb{R}^{-*}, \int_{x_{min}}^{x_{max}} F'(x)\,\mathrm{d}x = 1 \tag{A.10}$$

where

$$F'(x) = \begin{cases} \sqrt[3]{m + \sqrt{m^2 - n^2}} + \sqrt[3]{m - \sqrt{m^2 - n^2}} - \sqrt[3]{n}, & \text{if } m > n \\ \dfrac{2a}{3} \cdot \cos\left(\dfrac{\arccos\left(\frac{27b}{2a^3} - 1\right)}{3}\right) - \dfrac{a}{3}, & \text{otherwise} \end{cases}$$

with $a = -\lambda' \cdot \dfrac{p(x)}{c}$, $\quad b = -\dfrac{p(x)}{c}$, $\quad m = \dfrac{b}{2} - \dfrac{a^3}{27}$, and $n = \dfrac{a^3}{27}$

In what follows, we use the notation F'_c instead of F' to note that the function depends on the parameter c. Let u be the function defined on $\mathbb{R}^{-*}$ by

$$u(c) = \int_{x_{min}}^{x_{max}} F'_c(x)\,\mathrm{d}x \tag{A.11}$$

Let us determine the limits of u as $c \to -\infty$ and $c \to 0^-$:
For a given value x : $\quad m > n \Leftrightarrow b > \frac{4a^3}{27} \Leftrightarrow -\frac{p(x)}{c} > -\frac{4\lambda'^3 \cdot p(x)^3}{27}$

$$\Leftrightarrow c < p(x)\sqrt{\frac{4\lambda'^3}{27}}$$

Hence, $\forall x$, $m > n$ as $c \to -\infty$ and $m < n$ as $c \to 0$,

- $\lim\limits_{c \to -\infty} u(c)$:

  When $c \to -\infty$, $F_c'(x) = \sqrt[3]{m + \sqrt{m^2 - n^2}} + \sqrt[3]{m - \sqrt{m^2 - n^2}} - \sqrt[3]{n}$.

  Then, we can directly find that $\lim\limits_{c \to -\infty} F_c'(x) = 0$.

  Thus $\lim\limits_{c \to -\infty} u(c) = 0$

- $\lim\limits_{c \to 0^-} u(c)$:

  When, $c \to 0$, $F_c'(x) = \dfrac{a}{3} \cdot \left(2\cos\left(\dfrac{\arccos\left(\frac{27b}{2a^3} - 1\right)}{3} \right) - 1 \right)$

  Let $t = \frac{27b}{2a^3}$. We have $t \to 0$ as $c \to 0$,

  $$\frac{\mathrm{d}}{\mathrm{d}t}\left(\arccos(t-1)\right) = -\frac{1}{\sqrt{1-(t-1)^2}} = -\frac{1}{\sqrt{2t - t^2}}$$
  $$\underset{t \to 0}{\sim} -\frac{1}{\sqrt{2t}}$$

  By integration, we obtain:

  $\arccos(t-1) = \arccos(-1) + \int_0^t \frac{\mathrm{d}}{\mathrm{d}t}(\arccos(t-1))\ \mathrm{d}t \underset{t \to 0}{=} \pi - \sqrt{2t} + o(\sqrt{t})$

  Moreover, the Taylor expansion of $\cos(x + \frac{\pi}{3})$ gives:

  $\cos(x + \frac{\pi}{3}) \underset{x \to 0}{=} \frac{1}{2} - \frac{\sqrt{3}}{2}x + o(x)$

  Hence, by composition:

  $\cos(\frac{\arccos(t-1)}{3}) = \cos(\frac{\arccos(t-1) - \pi}{3} + \frac{\pi}{3}) \underset{t \to 0}{=} \frac{1}{2} + \sqrt{\frac{t}{6}} + o(\sqrt{t})$

  We deduce that: $\quad 2\cos\left(\dfrac{\arccos\left(\frac{27b}{2a^3} - 1\right)}{3}\right) \underset{c \to 0^-}{=} 1 + 3\sqrt{\dfrac{b}{a^3}} + o\left(\sqrt{\dfrac{b}{a^3}}\right)$

  $$\underset{c \to 0^-}{=} 1 + \frac{-3c}{p(x)\sqrt{\lambda'^3}} + o(c)$$

  And $\lim\limits_{c \to 0^-} F_c'(x) = \sqrt{\frac{1}{\lambda'}}$

  Finally, by integration : $\lim\limits_{c \to 0^-} u(c) = \dfrac{x_{max} - x_{min}}{\sqrt{\lambda'}}$

Furthermore, we admit that F_c' is increasing with respect to c. Although the demonstration is not given here, it has been observed experimentally and this result does not depend on p or λ'. Therefore, u is also increasing. It is thus a bijection from $\mathbb{R}^{-*}$ to $]0, \dfrac{x_{max} - x_{min}}{\sqrt{\lambda'}}[$. Thus, if $\dfrac{x_{max} - x_{min}}{\sqrt{\lambda'}} > 1$, then $\exists! c \in \mathbb{R}^{+*}, u(c) = 1$.

In conclusion, if $\lambda' \in]0, (x_{max} - x_{min})^2[$, there exists a solution to the minimization problem. There is no solution otherwise.

A.3 Existence and unicity of λ_0^*

In this section we want to prove that :

$$\forall \lambda' \in [0, (x_{max} - x_{min})^2[,\ \exists! \lambda_0^* > 0,\ \lambda' = \lambda_0^* \cdot S_{\lambda_0^*}(x_{max})^2 \tag{A.12}$$

where $S_{\lambda_0^*}(x_{max}) = \int_{x_{min}}^{x_{max}} S'_{\lambda_0^*}(x)\,\mathrm{d}t$, and $S'_{\lambda_0^*}$ is the function defined in equation (3.24). Let us define the function v by $v(\lambda_0) = \lambda_0 \cdot S_{\lambda_0}(x_{max})^2$. We have,

$$v(\lambda_0) = \left(\int_{x_{min}}^{x_{max}} \sqrt{\lambda_0} \cdot S'_{\lambda_0}(x)\,\mathrm{d}t \right)^2 \tag{A.13}$$

v is defined in 0 and $v(0) = 0$. Moreover, with a similar method as that of section A.2, we can show that $\lim\limits_{\lambda_0 \to +\infty} v(\lambda_0) = (x_{max} - x_{min})^2$, and that v is an increasing function. v is thus a bijection from $\mathbb{R}^+$ to $[0, (x_{max} - x_{min})^2[$. Therefore, the statement (A.12) is true.

Glossary

CIE :	Commission Internationale de L'Eclairage
UCS :	Uniform Chromaticity Scale
OETF :	Opto-Electrical Transfer Function
EOTF :	Electro-Optical Transfer Function
HDR :	High Dynamic Range
LDR :	Low Dynamic Range
TMO :	Tone Mapping Operator
HEVC :	High efficiency video coding
AVC :	Advanced Video Coding
QP :	Quantization parameter
CU :	Coding unit
PU :	Prediction unit
TU :	Transform unit
GOP :	Group of pictures
DCT :	Discrete Cosine Transform
CABAC :	Context Adaptive Binary Arithmetic Coding
RD(O) :	Rate-Distortion (Optimization)
BD-Rate :	Bjontegaard rate gain (in percent)
BD-PSNR :	Bjontegaard PSNR gain (in dB)
p.d.f. :	probability density function.
GMM :	Gaussian mixture model
EM :	Expectation maximization
ILP :	Inter-Layer prediction

List of Figures

List of Tables

Bibliography

[1] Cisco, "Cisco visual networking index: Forecast and methodology, 2014-2019," 2015. `http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white_paper_c11-481360.pdf`.

[2] F. Banterle, A. Artusi, K. Debattista, and A. Chalmers, *Advanced High Dynamic Range Imaging: Theory and Practice*. Natick, MA, USA: A. K. Peters, Ltd., 1st ed., 2011.

[3] T. Smith and J. Guild, "The C.I.E. colorimetric standards and their use," *Transactions of the Optical Society*, vol. 33, no. 3, p. 73, 1931.

[4] C. A. Poynton, "A technical introduction to digital video," ch. Basic Principles, New York (N.Y.): J. Wiley, 1996.

[5] G. Fechner. Leipzig : Breitskopf and Härtel, 1860.

[6] S. Stevens and J. Stevens, "Brightness function: parametric effects of adaptation and contrast," *Journal of the Optical Society of America*, pp. 1139–1146, 1960.

[7] H. Bodmann, "Visibility assessment in lighting engineering," *Journal of the Illuminating Engineering Society*, vol. 2, no. 4, pp. 437–443, 1971.

[8] O. Blackwell and H. Blackwell, "Visual performance data for 156 normal observers of various ages," *Journal of the Illuminating Engineering Society*, vol. 1, no. 1, pp. 3–13, 1971.

[9] J. A. Ferwerda, S. N. Pattanaik, P. Shirley, and D. P. Greenberg, "A model of visual adaptation for realistic image synthesis," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, (New York, NY, USA), pp. 249–258, ACM, 1996.

[10] P. Barten, "Physical model for the contrast sensitivity of the human eye," in *Proceedings of SPIE 1666, Human Vision, Visual Processing, and Digital Display III*, pp. 57–72, 1992.

[11] A. V. Meeteren and J. J. Vos, "Resolution and contrast sensitivity at low luminances," *Vision Research*, vol. 12, no. 5, pp. 825–833, 1972.

[12] T. O. Aydın, R. Mantiuk, and H.-P. Seidel, "Extending quality metrics to full dynamic range images," in *Human Vision and Electronic Imaging XIII*, Proceedings of SPIE, (San Jose, USA), pp. 6806–10, Jan. 2008.

[13] S. J. Daly, "Visible differences predictor: an algorithm for the assessment of image fidelity," vol. 1666, pp. 2–15, 1992.

[14] D. L. MacAdam, "Visual sensitivities to color differences in daylight," *Journal of the Optical Society of America*, vol. 32, pp. 247–274, May 1942.

[15] CIE S 014-5/E:2009/ISO 11664-5:2009(E), "Colorimetry - Part 5: CIE 1976 L*u*v* Colour Space and u', v' Uniform Chromaticity Scale Diagram," 1976.

[16] CIE S 014-4/E:2007/ISO 11664-4:2008(E), "CIE Colorimetry - Part 4: 1976 L*a*b* Colour Space," 1976.

[17] P. Urban, R. S. Berns, and M. R. Rosen, "Constructing euclidean color spaces based on color difference formulas," *Color and Imaging Conference (CIC)*, vol. 2007, pp. 76–82, Jan. 2007.

[18] CIE S 014-2/E:2006/ISO 11664-2:2007(E), "CIE colorimetry - part 2: Standard illuminants for colorimetry," 1931.

[19] RP 177-1993, "Derivation of basic television color equations," *SMPTE Recommended Practice*.

[20] M. Stokes, M. Anderson, S. Chandrasekar, and R. Motta, "A standard default color space for the internet - sRGB," *W3C CSS3 Color specification*, Nov. 1996.

[21] S. Miller, M. Nezamabadi, and S. Daly, "Perceptual signal coding for more efficient usage of bit codes," *SMPTE Motion Imaging Journal*, Oct. 2012.

[22] SMPTE ST 2084:2014, "High dynamic range electro-optical transfer function of mastering reference displays," Aug. 2014.

[23] A. M. Tourapis, Y. Su, D. Singer, C. Foog, R. V. der Vleuten, and E. Francois, "Report on the XYZ/HDR exploratory experiment 1 (EE1): Electro-Optical Transfer Functions for XYZ/HDR delivery," *ISO/IEC JTC1/SC29/WG11 MPEG2014/M34165*, 2014.

[24] ITU-R SG6/W6-C group, Working Party 6C (WP 6C) - Programme production and quality assessment

[25] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, (New York, NY, USA), pp. 369–378, ACM Press/Addison-Wesley Publishing Co., 1997.

[26] T. Mitsunaga and S. Nayar, "Radiometric Self Calibration," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 374–380, Jun. 1999.

[27] E. A. Khan, A. O. Akyiiz, and E. Reinhard, "Ghost removal in high dynamic range images," *13th IEEE International Conference on Image Processing (ICIP)*, pp. 2005–2008, Oct. 2006.

[28] S. K. Nayar and T. Mitsunaga, "High dynamic range imaging: Spatially varying pixel exposures," *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 472–479, Jun. 2000.

[29] M. Schoberl, A. Belz, J. Seiler, S. Foessel, and A. Kaup, "High dynamic range video by spatially non-regular optical filtering," *19th IEEE International Conference on Image Processing (ICIP)*, pp. 2757–2760, Sep. 2012.

[30] M. D. Tocci, C. Kiser, N. Tocci, and P. Sen, "A versatile hdr video production system," *ACM Trans. Graph.*, vol. 30, pp. 41:1–41:10, Jul. 2011.

[31] G. Ward, *Graphics Gems II*, ch. Real pixels, pp. 80–83. Academic Press, 1991.

[32] G. J. Ward, "The radiance lighting simulation and rendering system," in *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '94, (New York, NY, USA), pp. 459–472, ACM, 1994.

[33] R. Bogart, F. Kainz, and D. Hess, "OpenEXR image file format," *ACM Siggraph, Sketches & Applications*, 2003.

[34] D. Hough, "Applications of the proposed IEEE 754 standard for floating-point arithmetic," *Computer*, vol. 14, no. 3, pp. 70–74, 1981.

[35] D. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the IRE*, vol. 40, pp. 1098–1101, Sep. 1952.

[36] "Adobe. tiff 6.0 specification," 1992. `http://partners.adobe.com/public/developer/tiff/index.html`.

[37] G. W. Larson, "Overcoming gamut and dynamic range limitations in digital images," in *Proceedings of IS&T 6th Color Imaging Conference*, 1998.

[38] G. W. Larson, "Logluv encoding for full-gamut, high-dynamic range images," *J. Graph. Tools*, vol. 3, pp. 15–31, Mar. 1998.

[39] H. Seetzen, W. Heidrich, W. Stuerzlinger, G. Ward, L. Whitehead, M. Trentacoste, A. Ghosh, and A. Vorozcovs, "High dynamic range display systems," *ACM Trans. Graph.*, vol. 23, Aug. 2004.

[40] R. Mantiuk, S. Daly, and L. Kerofsky, "Display adaptive tone mapping," *ACM Trans. Graph.*, vol. 27, Aug. 2008.

[41] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, "Photographic tone reproduction for digital images," *ACM Trans. Graph.*, vol. 21, pp. 267–276, Jul. 2002.

[42] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," *ACM Trans. Graph.*, vol. 21, pp. 257–266, Jul. 2002.

[43] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the Sixth International Conference on Computer Vision*, ICCV '98, (Washington, DC, USA), pp. 839–846, IEEE Computer Society, 1998.

[44] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," *ACM Trans. Graph.*, vol. 21, pp. 249–256, Jul. 2002.

[45] J. Tumblin and G. Turk, "Lcis: A boundary hierarchy for detail-preserving contrast reduction," pp. 83–90, 1999.

[46] G. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *Signal Processing Magazine, IEEE*, vol. 15, pp. 74–90, Nov. 1998.

[47] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *Signal Processing Magazine, IEEE*, vol. 15, pp. 23–50, Nov. 1998.

[48] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, pp. 1649–1668, Dec. 2012.

[49] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 560–576, Jul. 2003.

[50] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transform," *Computers, IEEE Transactions on*, vol. C-23, pp. 90–93, Jan. 1974.

[51] J. J. Rissanen, "Generalized kraft inequality and arithmetic coding," *IBM J. Res. Dev.*, vol. 20, pp. 198–203, May. 1976.

[52] A. Norkin, G. Bjontegaard, A. Fuldseth, M. Narroschke, M. Ikeda, K. Andersson, M. Zhou, and G. V. D. Auwera, "Hevc deblocking filter," vol. 22, pp. 1746–1754, Dec. 2012.

[53] C.-M. Fu, E. Alshina, E. Alshin, Y.-W. Huang, C.-Y. Chen, C.-Y. Tsai, C.-W. Hsu, S.-M. Lei, J.-H. Park, and W.-J. Han, "Sample adaptive offset in the HEVC standard," vol. 22, pp. 1755–1764, Dec. 2012.

[54] Z. Mai, H. Mansour, R. Mantiuk, P. Nasiopoulos, R. Ward, and W. Heidrich, "On-the-fly tone mapping for backward-compatible high dynamic range image/video compression," *ISCAS*, pp. 1831–1834, Jun. 2010.

[55] Z. Mai, H. Mansour, R. Mantiuk, P. Nasiopoulos, R. K. Ward, and W. Heidrich, "Optimizing a tone curve for backward-compatible high dynamic range image and video compression," vol. 20, pp. 1558–1571, Jun. 2011.

[56] A. Koz and F. Dufaux, "Methods for improving the tone mapping for backward compatible high dynamic range image and video coding," *Image Commun.*, vol. 29, pp. 274–292, Feb. 2014.

[57] F. Banterle, K. Debattista, P. Ledda, and A. Chalmers, "A gpu-friendly method for high dynamic range texture compression using inverse tone mapping," *Proceedings of graphics interface*, pp. 41–48, 2008.

[58] Y. Li, L. Sharan, and E. H. Adelson, "Compressing and companding high dynamic range images with subband architectures," *Transactions on Graphics, Proceedings of SIGGRAPH*, pp. 836–844, Jul. 2005.

[59] D. Touzé, Y. Olivier, S. Lasserre, F. L. Léannec, R. Boitard, and E. François, "HDR video coding based on local LDR quantization," *Second International Conference and SME Workshop on HDR imaging*, Mar. 2014.

[60] T. Jinno, M. Okuda, and N.Adami, "New local tone mapping and two-layer coding for HDR images," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 765–768, Mar. 2012.

[61] G. Ward and M. Simmons, "JPEG-HDR: A backwards-compatible, high dynamic range extension to jpeg," *Proceedings of the Thirteenth Color Imaging Conference*, 2005.

[62] R. Mantiuk, A. Efremov, K. Myszkowski, and H.-P. Seidel, "Backward compatible high dynamic range MPEG video compression," *ACM SIGGRAPH*, Jul. 2006.

[63] T. Richter, "On the standardization of the JPEG XT image compression," *Picture Coding Symposium (PCS)*, pp. 37–40, Dec. 2013.

[64] C. Lee and C.-S. Kim, "Rate-distortion optimized layered coding of high dynamic range videos," *Journal of Visual Communication and Image Representation*, vol. 23, pp. 908–923, Aug. 2012.

[65] M. Okuda and N. Adami, "Two-layer coding algorithm for high dynamic range images based on luminance compensation," *Journal of Visual Communication and Image Representation*, vol. 18, pp. 377–386, Oct. 2007.

[66] S. Liu, W.-S. Kim, and A. Vetro, "Bit-depth scalable coding for high dynamic range video," *SPIE Conference on Visual Communications and Image Processing*, Jan. 2008.

[67] A. Segall, "Scalable coding of high dynamic range video," *14th IEEE International Conference on Image Processing (ICIP)*, Oct. 2007.

[68] J.-U. Garbas and H. Thoma, "Inter-layer prediction for backwards compatible high dynamic range video coding with SVC," *Picture Coding Symposium (PCS)*, pp. 285–288, May 2012.

[69] ITU-R rec. BT.2020, *Parameter values for ultra-high definition television systems for production and international programme exchange.* 2012.

[70] ITU-R rec. BT.709, *basic parameter values for the HDTV standard for the studio and for international programme exchange.* 1990.

[71] A. Motra and H. Thoma, "An adaptive LogLuv transform for high dynamic range video compression," *IEEE Int. Conf. Image Process. (ICIP)*, pp. 2061–2064, Sept. 2010.

[72] Y. Dong, P. Nasiopoulosa, and M. T. Pourazad, "HDR video compression using high efficiency video coding (HEVC)," *Ubicomm*, pp. 205–209, Sep. 2012.

[73] J. Blinn, "Floating-point tricks," *IEEE Computer Graphics and Applications*, vol. 17, pp. 80–84, Jul. 1997.

[74] M. Iwahashi and H. Kiya, "Two layer lossless coding of HDR images," *IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, pp. 1340–1344, May 2013.

[75] "HM range extension, version 12.0 RExt 4.0 rc2," `https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-12.0+RExt-4.0rc2/`.

[76] G. Bjontegaard, "Calculation of average PSNR differences between RD curves," *document VCEG-M33, ITU-T VCEG Meeting*, 2001.

[77] L. Yang and D. Wu, "Adaptive quantization using piecewise companding and scaling for gaussian mixture," *Visual Communication and Image Representation*, pp. 959–971, Oct. 2012.

[78] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, pp. 129–137, 1982.

[79] Y.Zhang, E. Reinhard, and D. Bull, "Perception-based high dynamic range video compression with optimal bit-depth transformation," *18th IEEE International Conference on Image Processing (ICIP)*, pp. 1321–1324, Sept. 2011.

[80] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Entropy-constrained vector quantization," *IEEE Transactions on acoustics, speech, and signal processing*, 1989.

[81] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, pp. 1–38, 1977.

[82] "The JPEG-2000 still image compression standard," *ISO/IEC JTC Standard 1/SC29/WG1*, 2005.

[83] M. Winken, D. Marpe, H. Schwarz, and T. Wiegand, “Bit-depth scalable video coding,” *International Conference on Image Processing (ICIP)*, pp. 5–8, Sep. 2007.

[84] M. Takeuchi, Y. Matsuo, Y. Yamamura, J. Katto, and K. Iguchi, “A bit-depth scalable video coding approach considering spatial gradation restoration,” *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1373–1376, Mar. 2012.

[85] R. Mantiuk, K. Myszkowski, and H.-P. Seidel, “A perceptual framework for contrast processing of high dynamic range images,” *ACM Trans. Appl. Percept.*, vol. 3, pp. 286–308, Jul. 2006.

[86] S. N. Pattanaik, J. Tumblin, H. Yee, and D. P. Greenberg, “Time-dependent visual adaptation for fast realistic image display,” *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 47–54, Jul. 2000.

[87] R. Mantiuk, “pfstools hdr image gallery,” `http://pfstools.sourceforge.net/hdr_gallery.html`.

[88] G. Ward, “High dynamic range image examples,” `http://www.anyhere.com/gward/hdrenc/pages/originals.html`.

[89] G. Krawczyk and R. Mantiuk, “pfstmo tone mapping library,” `http://pfstools.sourceforge.net/pfstmo.html`.

[90] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” vol. 13-4, pp. 600–612, Apr. 2004.

[91] M. Narwaria, R. K. Mantiuk, M. P. Da Silva, and P. Le Callet, “Hdr-vdp-2.2: a calibrated method for objective quality prediction of high-dynamic range and standard images,” *Journal of Electronic Imaging*, vol. 24, no. 1, p. 010501, 2015. code available at : http://hdrvdp.sf.net/.

[92] C. Poynton, J. Stessen, and R. Nijland, “Deploying wide color gamut and high dynamic range in HD and UHD,” *SMPTE Mot. Imag. J*, vol. 124, pp. 37–49, Apr. 2015.

[93] R. Mantiuk, K. Myszkowski, and H.-P. Seidel, “Lossy compression of high dynamic range images and video,” *Human Vision and Electronic Imaging XI, SPIE*, vol. 6057, Feb. 2006.

[94] C. Schlick, “Quantization techniques for visualization of high dynamic range pictures,” *5th Eurographics Workshop on Rendering*, pp. 7–20, 1994.

[95] A. Luthra, E. François, and W. Husak, “Call for evidence (CfE) for HDR and WCG video coding,” *ISO/IEC JTC1/SC29/WG11 N15083*, Feb. 2015.

[96] G. Sharma, W. Wu, and E. N. Dalal, “The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations,” *Color research and application*, vol. 30, Feb. 2005.

[97] T. Pouli, A. Artusi, F. Banterle, A. O. Akyuz, H.-P. Seidel, and E. Reinhard, “Color correction for tone reproduction,” in *CIC21: Twenty-first Color and Imaging Conference*, pp. 215–220, Society for Imaging Science and Technology (IS&T), November 2013.

[98] R. Mantiuk, R. Mantiuk, A. Tomaszewska, and W. Heidrich, “Color correction for tone mapping,” *Computer Graphics Forum*, vol. 28, no. 2, pp. 193–202, 2009.

[99] H. Yue, X. Sun, J. Yang, and F. Wu, “Cloud-based image coding for mobile devices-toward thousands to one compression,” *Multimedia, IEEE Transactions on*, vol. 15, pp. 845–857, June 2013.

Abstract

In recent years, the display technologies have been rapidly evolving. From 3D television to Ultra High Definition, the trend is now towards High Dynamic Range (HDR) displays that can reproduce a luminance range far beyond the capabilities of conventional displays. The emergence of this technology involves new standardization effort in the field of video compression. In terms of large scale content distribution, the question of backward compatibility is critical. While the future generation of television displays will be adapted to this new format, it is necessary to enable the older equipment to decode and display a version of the same content whose dynamic range has been previously reduced by a process called "tone mapping". This thesis aims at exploring the backward compatible HDR compression schemes. In a first approach, a tone mapping operator specified by the encoder is applied to the HDR image. The resulting image, called Low Dynamic Range (LDR), can then be encoded and decoded in a conventional format. The encoder additionally transmits a small amount of information enabling a HDR capable decoder to inverse the tone mapping operator and retrieve the HDR version. The study of these schemes is directed towards the definition of tone mapping operators optimized for the compression performance. We then focus on scalable approaches, where both versions are given to the encoder without prior knowledge on the tone mapping operator used. The producer thus keeps full control on the LDR content creation process. This LDR version is compressed as a first layer. The reconstructed image is used by the scalable encoder to compress the HDR layer efficiently by performing inter-layer predictions. Thanks to a local and non-linear approach, the proposed schemes improve the coding performance compared to the existing scalable methods, especially in the case where a complex tone mapping is used for generating the LDR version.

Résumé

Les technologies d'écran ont connu récemment une évolution rapide. De la télévision 3D à l'Ultra Haute Définition, la tendance est maintenant aux écrans HDR (pour "High Dynamic Range") permettant de reproduire une gamme de luminance bien plus élevée que les écrans classiques. L'émergence de cette technologie implique de nouveaux travaux de standardisation dans le domaine de la compression vidéo. Une question essentielle pour la distribution à grande échelle de contenu HDR est celle de la rétro-compatibilité. Tandis que la future génération d'écrans de télévision sera adaptée a ce nouveau format, il est nécessaire de permettre aux équipements plus anciens de décoder et afficher une version du même contenu dont la dynamique a été préalablement réduite par un procédé appelé "tone mapping". Cette thèse vise à explorer les schémas de compression HDR rétro-compatibles. Dans une première approche, un algorithme de tone mapping spécifié par l'encodeur est appliqué à l'image HDR. L'image générée, alors appelée LDR (pour "Low Dynamic Range"), peut alors être encodée et décodée dans un format classique. L'encodeur transmet par ailleurs une quantité réduite d'information permettant à un décodeur HDR d'inverser l'opération de tone mapping et de reconstruire une version HDR. L'étude de ces schémas est axée sur la définition de méthodes de tone mapping optimisées pour les performances de compression. La suite de la thèse se concentre sur l'approche scalable dans laquelle les deux versions sont fournies à l'encodeur sans connaissance à priori sur l'opérateur de tone mapping utilisé. Le producteur garde donc le contrôle sur la création du contenu LDR. Cette version LDR est d'abord compressée comme une première couche. L'image reconstruite est utilisée par le codeur scalable pour compresser plus efficacement la couche HDR grâce à un mécanisme de prédiction inter-couches. Notre approche locale et non linéaire nous permet d'améliorer les performances de codage par rapport aux méthodes scalables existantes, en particulier dans le cas où un tone mapping complexe est utilisé pour générer la version LDR.