Learning based coding and post-processing methods for 8K video reconstruction

Charles Bonnineau

DOCTORAT BRETAGNE LOIRE / MATHSTIC

INSA RENNES

THÈSE DE DOCTORAT DE

L'INSTITUT NATIONAL DES SCIENCES APPLIQUÉES RENNES

ÉCOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Traitement du signal*

Par
Charles BONNINEAU

Learning based coding and post-processing methods for 8K video reconstruction

Thèse présentée et soutenue à Rennes, le 30 juin 2022
Unité de recherche : VAADER, IETR/INSA
Thèse N° : 22ISAR 18 / D22 - 18

Rapporteurs avant soutenance :

Anissa Mokraoui	Professeur, Université de Paris XVIII, France
Frédéric Dufaux	Directeur de Recherche CNRS, Centrale Supelec, France

Composition du Jury :

Aline Roumy	Directrice de Recherche INRIA, INRIA Rennes, France (Présidente)
Anissa Mokraoui	Professeur, Université de Paris XVIII, France
Frédéric Dufaux	Directeur de Recherche CNRS, Centrale Supelec, France
Marco Cagnazzo	Maître Assistant, Université de Padoue, Italie
Thibaud Biatek	Ingénieur Recherche, Ateme, France
Olivier Déforges	Professeur, IETR/INSA, France (Directeur de thèse)
Jean-François Travers	Ingénieur Recherche, TDF, France (Co-directeur de thèse)
Wassim Hamidouche	Maître de Conférence, IETR/INSA, France (Encadrant de thèse)

ACKNOWLEDGEMENT

First of all, I want to express my gratitude to Wassim Hamidouche, one of my PhD supervisors at the IETR. He was continuously involved in this project and actively contributed to every paper written during these three years. His advice taught me to better valorize my work and be a more ambitious scientist. I would also like to thank Jean-François Travers, my PhD supervisor at TDF, for his confidence, support, and continuous interest in my work. He always answered my questions and shared his knowledge with a lot of passion, which greatly inspired me. I'm also grateful to Olivier Déforges, my PhD Director. He helped me find the right and positive way to interpret my results. I'm thankful to him for his availability, especially during the writing of this manuscript.

I would like to express my gratitude to Naty Sidaty, who encouraged me to present my research in standardization bodies and was of immense support during this thesis. Also, I'm grateful to Thibaud Biatek, who was involved at the beginning of this project and gave me precious advice. I would also like to thank Jean-Yves Aubié, my lab manager at IRT b<>com. He was always available and helped me to demonstrate the industrial relevancy of this work.

I'm greateful to Anissa Mokraoui and Frédéric Dufaux for accepted to review this manuscript. I also thank Aline Roumy, Marci Cagnazzo, and Thibaud Biatek for being part of my jury.

Many thanks to my colleagues from b<>com and TDF for their precious support throughout these three years. I'm particularly grateful to Glenn, Antonin, Nicolas. R, Nicolas. E, François, and Franck. They gave me precious advice inside and outside the lab that contributed to who I am today as a scientist and a person.

Last but not least, my final thanks go to my partner Aude, my parents, Brigitte and Alex, my sister Claire and my friends, without whom I could not have achieved this work. I cannot thank them enough for their invaluable support and encouragement throughout this journey.

RÉSUMÉ EN FRANÇAIS

Introduction

La télévision ultra-haute définition (UHDTV) [1] est un format numérique de vidéo basé sur de nouvelles caractéristiques immersives telles qu'une plage de dynamique étendue, un gamut de couleur plus large et une résolution temporelle et spatiale plus importante. L'introduction de ce nouveau format a pour but d'améliorer la qualité d'expérience (QoE) de l'utilisateur final par rapport au précédent format standard : la télévision haute définition (HDTV) [2]. Parmi ces nouvelles caractéristiques se trouve la résolution vidéo 8K, correspondant à une résolution spatiale de 7680×4320 pixels, soit 4 fois plus de pixels que la résolution 4K (3840×2160) et 16 fois plus que la résolution HD (1920×1080), comme illustré dans la Figure 2.3. Une telle résolution permet d'accroître la sensation de réalisme perçue par l'utilisateur en augmentant la quantité de détails restitués depuis la scène capturée. Cette résolution d'image a récemment connu beaucoup d'engouement dans la communauté de la vidéo avec l'apparition de nouveaux capteurs et écran TV 8K, mais aussi au travers d'expérimentations réalisées par des diffuseurs, e.g. au Japon avec la NHK. Néanmoins, cet afflux d'information spatiale nécessite d'importantes ressources en débit pour garantir une bonne qualité de vidéo, imposant de nouveaux challenges aux diffuseurs de contenus.

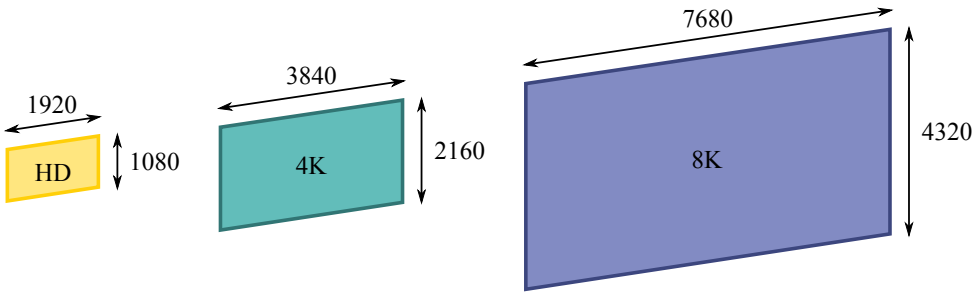


Figure 1 – Illustration des résolutions spatiale high-definition (HD), 4K et 8K.

Ainsi, sous la pression des différents acteurs industriels, de nouveaux standards de compression ont émergé, comme *versatile video coding* [3], ou VVC/H.266, finalisé en juillet 2020. Cette norme est née d'un travail collaboratif mené par le *joint video exploration team* (JVET) constitué

d'experts de l'ITU et de l'ISO/IEC respectivement représentés par les groupes VCEG et MPEG. L'objectif du standard VVC est de réduire le débit nécessaire pour une même qualité perçue d'environ 40% par rapport à son prédécesseur *high efficiency video coding* (HEVC) [4]. Comme ce dernier l'a permis pour l'introduction de la 4K, le déploiement de VVC devrait permettre l'introduction de nouveaux services plus lourds tels que la 8K ou la vidéo 360°.

En France, le conseil supérieur de l'audiovisuel (CSA), nouvellemment autorité de régulation de la communication audiovisuelle et numérique (ARCOM), a lancé des travaux techniques préparant une bascule technologique sur la télévision numérique terrestre (TNT) à l'horizon 2024, tablant sur un format de diffusion 4K délivré en DVB-T2/HEVC accompagnés de services TV enrichis (replay, VOD, push). Dans ce contexte, l'introduction de nouveaux formats multimédia n'est envisageable qu'en assurant la compatibilité avec le parc d'équipement existant. Ainsi, des tests ont été réalisés pour une transmission de 2 à 3 programmes avec des débits allant de 10 à 17Mbps pou la 4K et 3 à 17Mbps pour la HD avec HEVC [5]. Cependant, dû aux fortes contraintes de débit imposées par le format de résolution vidéo 8K, il est aujourd'hui impossible d'envisager une diffusion 8K sur la TNT française avec rétro compatibilité 4K via simulcast, i.e., les signaux 8K et 4K encodés indépendamment et multiplexés.

D'une part, différentes alternatives au simulcast existent aujourd'hui, tel que la compression scalable, proposée par example par les extensions de codecs SVC et SHVC. Néanmoins, dû à un gain en débit trop faible par rapport aux contraintes matérielles imposées, ces architectures n'ont pas été adoptées dans l'industrie. D'autre part, les technologies de compression et de post-traitement basées IA ont récemment démontré des performances très compétitives par rapport aux algorithmes conventionnels. Le potentiel de ces techniques pour la compression vidéo est déjà bien connu dans la littérature, avec par exemple l'application de super-résolution après décodage ou encore la compression d'image bout-en-bout [6, 7]. Ces algorithmes pourraient ainsi permettre de retrouver un signal 8K via un suréchantillonnage spatial du signal 4K guidé ou non par des métadonnées, e.g. LCEVC [8].

L'objectif de cette thèse a été d'explorer et de développer des algorithmes innovants et performants permettant la reconstruction d'un flux 8K à partir d'un flux 4K. Dans un premier volet, Une étude subjective à permis d'évaluer la qualité offerte par la résolution 8K et les performances de différentes méthodes de compression, tel que VVC, HEVC et des méthodes de codage offrant la rétrocompatibilité avec un signal 4K. Deuxièmement, de nouvelles méthodes de super-résolution basées sur un apprentissage multitâche adaptées à des images compressées ont été proposées. Dernièrement, une nouvelle méthode de compression de résidu de suréchantillonnage, appelée CAESR, a été développée et protégée par un brevet.

Etat de l'art

Nouvelles méthodes de codage vidéo

Le Chapitre 2 de la thèse présente les nouveaux formats vidéo et les dernières avancées dans les standards de compression. L'arrivée de ces nouveaux formats et l'évolution des supports de visionnage ont considérablement bouleversé notre manière de consommer les données vidéo. Pour répondre à cela, la communauté scientifique travaille sans relâche depuis des décennies sur le développement de standards de compression toujours plus efficaces. Une frise chronologique du développement des différents standards de compression dans le temps est donnée dans la Figure 2.7. Chaque standard de compression a contribué à améliorer les performances de son prédécesseur. Aujourd'hui, VVC offre environ 40% de gains pour la même QoE par rapport à HEVC, grâce à l'amélioration d'outils existant, tels qu'une taille de bloc supérieure et un algorithme CABAC amélioré, mais aussi à l'intégration de nouveaux outils de codage.

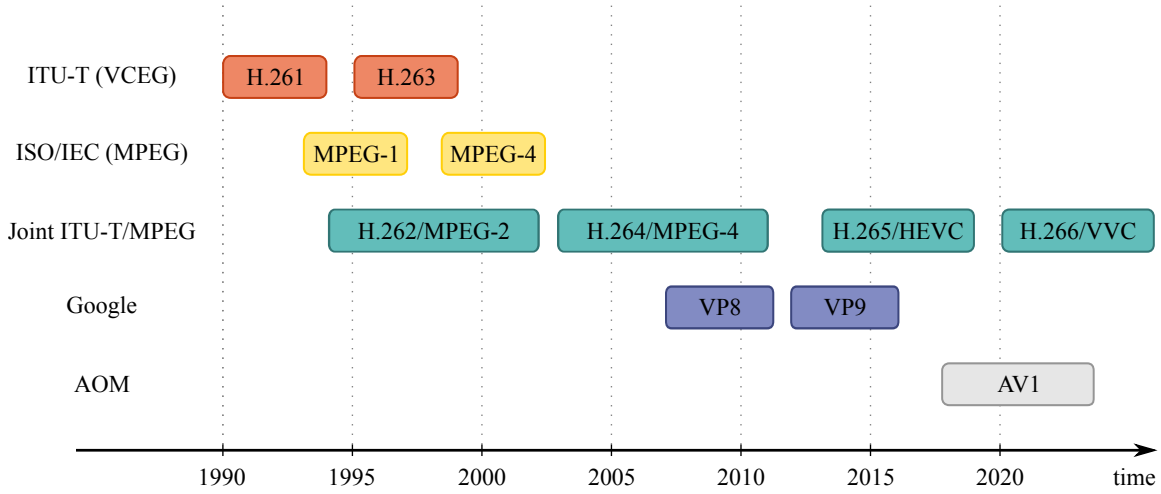


Figure 2 – Frise chronologique des standards de compression vidéo.

Comme mentionné dans l'introduction, une façon de réduire le débit requis par le simulcast est d'utiliser un codec scalable, tel que *scalable high efficiency video coding* (SHVC) [9], l'extension scalable de HEVC. Ainsi, dans un contexte de transmission rétro compatible 4K/8K, le signal 4K est encodé sous la forme d'une couche de base exploitée par une couche d'amélioration et d'un module de traitement inter-couche, permettant de réduire le débit du signal 8K. Cependant, la complexité apportée par la scalabilité et une introduction tardive dans le standard n'ont pas permis à SHVC d'atteindre les objectifs d'adoption visés. Récemment, un nouveau standard appelé *low complexity enhancement video coding* (LCEVC) a été publié à MPEG [8]. L'objectif

de ce standard est d'encoder une version sous-échantillonnée du signal d'entrée avec n'importe quel codec de base et de transmettre des métadonnées visant à améliorer le flux suréchantillonné côté récepteur. Ainsi, cette approche offre également la rétrocompatibilité spatiale de manière agnostique au codec de base avec une complexité supplémentaire moindre.

Algorithmes basés IA pour la compression d'image et de vidéo

Le Chapitre 3 présente les différentes applications de l'IA dans le domaine de compression d'image et de vidéo. Le principe de l'apprentissage automatique est d'estimer une fonction f permettant de réaliser la tâche souhaitée, i.e., prédire la ou les sorties $\boldsymbol{y}$ par rapport à des données d'entrées $\boldsymbol{x}$. Les paramètres θ de cette fonction f sont appris par rapport à un jeu de données représentatif de la tâche à effectuer, appelé la vérité de terrain. Récemment, les réseaux de neurones artificiels, directement inspirés des neurones biologiques, se sont démarqués dans ce domaine. Cet outil mathématique permet d'estimer n'importe quelle fonction f sous la forme d'une suite d'opérations linéaire et non-linéaire représentées par des couches de neurones successives. Dans un premier temps, le réseau prédit une sortie $\hat{\boldsymbol{y}}$ par rapport à une entrée $\boldsymbol{x}$. Une fonction de coût $\mathcal{L}$ est ensuite calculée entre l'échantillon prédit $\hat{\boldsymbol{y}}$ et sa valeur correspondante $\boldsymbol{y}$ dans la vérité de terrain. Les paramètres θ du réseau f sont finalement ajustés grâce à un algorithme de rétro propagation du gradient dans le but de minimiser la fonction de coût $\mathcal{L}$.

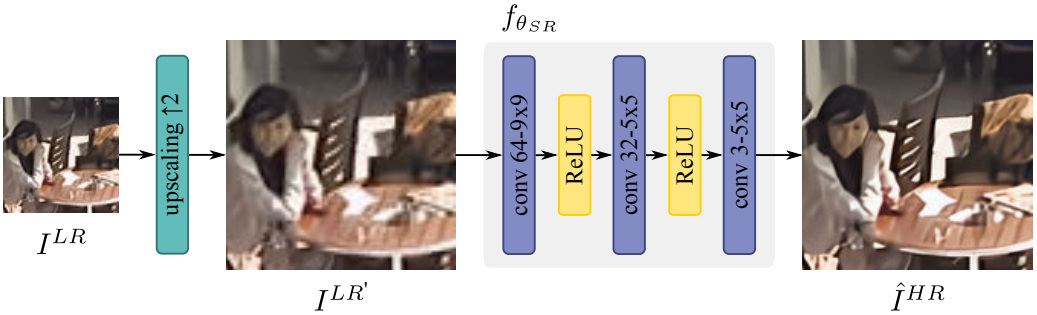


Figure 3 – Architecture de SRCNN [10].

Ces dernières années, les réseaux de neurones profonds, et plus particulièrement les réseaux de neurones convolutifs, ont conduit à de grandes avancées dans le domaine de la compression d'image et de vidéo. En effet, de nombreux problèmes inverses complexes sont rencontrés dans ce domaine, comme la restauration de dégradations spatiales ou de quantification contenues dans l'image décodée. Ainsi, les réseaux de neurones profonds peuvent être utilisés pour estimer une image de haute résolution par rapport à une ou plusieurs images de basse résolution ou alors pour améliorer l'image reconstruite après décodage pour corriger certains artefacts de compression.

Une représentation de l'architecture de SRCNN [10], le premier réseau de neurones convolutif dédié à la super-résolution, est illustrée dans la Figure 4.12.

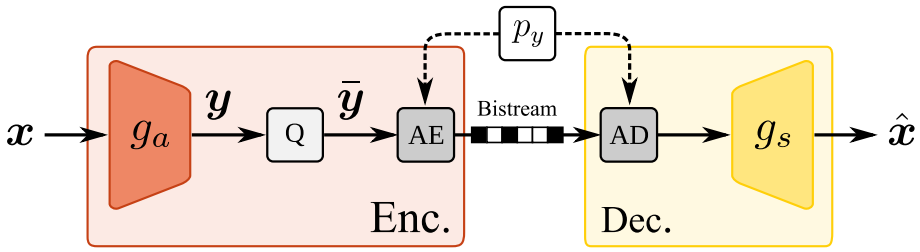


Figure 4 – Description de la compression d'image bout-en-bout [7].

Par ailleurs, les réseaux de neurones ont permis de directement rivaliser avec les codecs conventionnels. L'objectif de la compression d'image et de vidéo avec perte est de réduire le nombre de bits requis pour représenter le signal tout en préservant l'information visuelle la plus importante. Cet objectif peut être interprété comme un problème d'optimisation débit distorsion, où le modèle cherche à minimiser la distorsion dans l'image reconstruite sous une certaine contrainte de débit. Les systèmes de compression conventionnels sont composés de différents modules, i.e., modes de codages, ne pouvant pas être optimisés de façon conjointe. En d'autres mots, l'amélioration d'un mode de codage n'améliore pas forcément les performances globales du système. De plus, ils sont généralement basés sur un schéma de codage en bloc, générant des artéfacts le long des frontières de bloc. En 2017, Ballé *et. al* ont proposé un réseau de neurones auto-encodeur pour la compression d'image bout-en-bout, atteignant les performances de JPEG [7]. Contrairement aux méthodes de codage d'images et de vidéos classiques, la compression bout-en-bout remplace le codec par un auto-encodeur. L'auto-encodeur est composé de deux fonctions, une fonction d'analyse et une fonction de synthèse. Une illustration de ce système de codage est donnée dans la Figure 3.10. L'avantage de ce système est qu'il est complètement dérivable, ce qui permet d'optimiser tous les paramètres du système de codage par rapport à une fonction de coût débit distorsion, définie par:

$$\mathcal{L}(\lambda) = D + \lambda R, \tag{1}$$

avec D la distortion, R le débit et λ un multiplicteur Lagrangien. De plus, n'importe quel système dérivable peut être entraîné conjointement à ce schéma de compression, maximisant les performances du système global. Les auto-encodeurs sont aujourd'hui au niveau de VVC pour la compression d'image et au niveau de HEVC pour la compression de vidéo.

Contributions

Evaluation d'algorithmes et de standards pour la compression de vidéo 8K

Le Chapitre 4 présente différentes évaluations objectives et subjectives d'algorithmes de codage appliqués à des contenus vidéo 8K. La première partie de ce chapitre présente une étude subjective évaluant les performances des deux derniers standards de compression MPEG, i.e., VVC et HEVC, par le biais de leurs logiciels de référence respectifs : HM-16.20 et VTM-11. Cette étude mesure également l'apport subjectif de la 8K par rapport à la 4K non compressée. La qualité subjective est mesurée grâce à la méthodologie de test *double stimulus quality scale* (DSCQS) standardisée par l'ITU dans la recommandation BT.500 [11]. Cette étude a démontré que VVC offre environ 40% de gain en débit pour la même qualité perçue par rapport à HEVC. Le débit de transparence, correspondant au débit pour lequel aucune différence n'est perçue par rapport à la source, a été déterminé et varie de 11Mbps à 180Mbps selon la séquence utilisée. Nous avons par ailleurs confirmé que l'observateur moyen distingue une différence entre la 8K et la 4K pour certaines séquences et avons mesuré cette différence. Plusieurs métriques objectives ont été comparées par rapport aux scores subjectifs collectés. Les résultats ont montré que les métriques MS-SSIM [12] et VMAF [13] proposent les meilleures performances selon l'indicateur de corrélation utilisé.

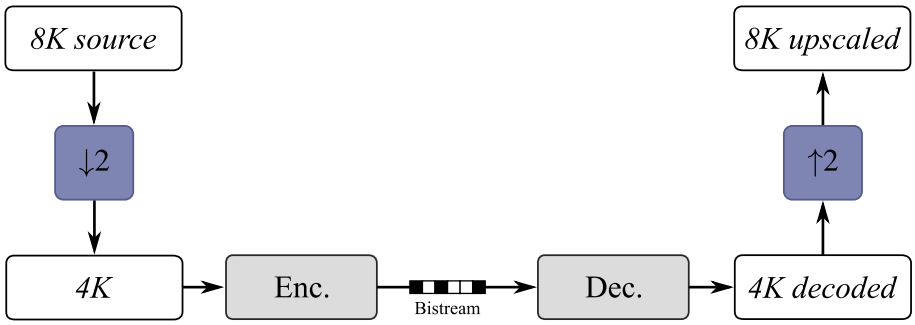


Figure 5 – Illustration de la configuration .

La seconde partie de ce chapitre présente une étude objective comparant différents algorithmes de compression permettant la transmission d'un flux 8K avec rétrocompatibilité 4K. Les méthodes testées sont le simulcast, la scalabilité spatiale avec SHVC et le suréchantillonnage spatial appliqué après décodage, tel qu'illustré dans la Figure 4.11. Deux méthodes de suréchantillonnage ont été comparées : un filtre Lanczos [14] et EDSR [15], un algorithme état-de-l'art de la super-résolution basé IA. Cette étude a montré que SHVC permet d'obtenir environ 19% de gain moyen par rapport à un simulcast HEVC. Cependant, le coût de la scalabilité pour

SHVC est d'approximativement 14% par rapport à un codage HEVC à pleine résolution. Nous avons également démontré que l'approche de suréchantillonnage spatiale offre de meilleures performances en débit par rapport à un simulcast sur toute la plage de débit testée. De plus, à bas débit, de meilleures performances sont obtenues par la configuration de suréchantillonnage spatial par rapport au codage pleine résolution pour certaines séquences. Les résultats ont aussi montré que les performances de EDSR sont supérieures à celle d'un filtre d'interpolation Lanczos. En revanche, les performances de EDSR se rapprochent de celle d'un filtre d'interpolation Lanczos à mesure que le débit baisse. La première hypothèse explorée dans cette thèse est qu'un algorithme de super-résolution adapté aux images compressé doit être considéré pour améliorer les performances sur ce type de contenus.

Apprentissage multitâche pour la super-résolution de vidéos compressées

Dans le Chapitre 5, l'apprentissage multitâche [16] a été exploré pour l'amélioration de la super-résolution appliquée à des images compressées. Dans un premier axe, un modèle appelé MTL-EDSR a été développé. Ce réseau génère à la fois une image haute résolution et une image basse résolution à partir d'une image basse résolution compressée avec VVC. Ainsi, deux tâches sont réalisées avec un seul et même réseau : la super-résolution de l'image d'entrée et la réduction des artéfacts de codage. Le coeur de l'architecture de ce modèle est basée sur EDSR et la partie multitâche est réalisée par un partage dur des paramètres. Une illustration du modèle proposé est représentée dans la Figure 5.2.

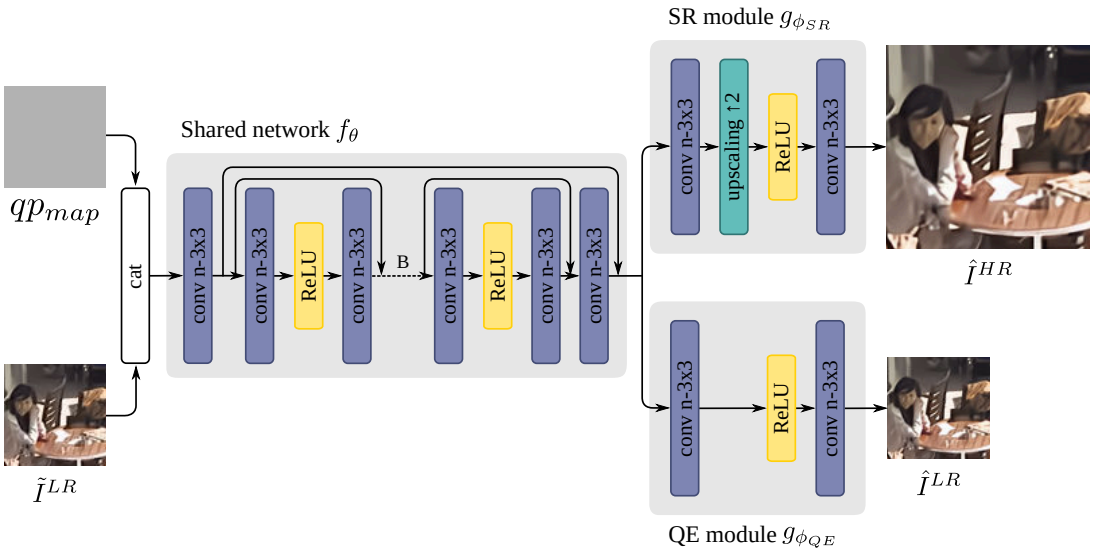


Figure 6 – Architecture de MTL-EDSR.

Cette architecture multitâche permet d'obtenir des performances similaires à des réseaux spécialisés. En effet, les tâches de super-résolution et d'amélioration de la qualité apprennent des caractéristiques très semblables qui sont partagées par le modèle multitâche. Cependant, cette mutualisation de paramètres permet de réduire le nombre de paramètres d'un facteur deux lorsque les deux tâches doivent être réalisées. Plusieurs stratégies d'apprentissage destinées à améliorer les performances d'algorithmes de restauration d'images compressées ont également été appliquées afin d'améliorer les performances du modèle, comme l'utilisation du QP en entrée du réseau et une technique de pré-entraînement.

Dans le deuxième axe, un autre modèle multitâche, appelé MTL-Unet, a été développé afin de réaliser des tâches de vision par ordinateur haut niveau en plus de la tâche principale de super-résolution, i.e., estimation de la qualité sans référence et segmentation sémantique. Contrairement à MTL-EDSR, ce modèle est basé sur l'architecture de Unet [17], un réseau état-de-l'art de la segmentation sémantique. Les résultats montrent que MTL-Unet réalise les différentes tâches de traitement d'images haut niveau et de super-résolution avec succès. Ainsi, en plus de la tâche de super-résolution, le réseau peut réaliser des tâches d'analyse d'images supplémentaire pouvant être intégrées dans la boucle de codage pour améliorer les performances. Malgré cette observation, l'objectif initial d'améliorer les performances de la super-résolution seule n'a pas été permis par l'élaboration de ces deux modèles.

Autoencodeur conditionel et super-résolution pour un codage scalable efficace

La dernière partie de la thèse vise à résoudre la problématique de dépendance au contenu de la compression basée sous-échantillonnage observé dans le Chapitre 3. En effet, bien que permettant une rétrocompatibilité agnostique au codec de base, certaines hautes fréquences ne peuvent pas être retrouvées par un simple suréchantillonnage sans information supplémentaire. LCEVC, récemment publié comme nouveau standard MPEG, propose une solution à ce problème. Ce codec transmet le résidu entre l'image suréchantillonnée et la source sous la forme de métadonnées, permettant de retrouver les détails perdus lors du sous-échantillonnage côté récepteur. Cependant, l'architecture de LCEVC est basée sur des modules de compression conventionnels, i.e., transformée d'Hadamard, décomposition en bloc, limitant les performances de codage. Dans ce chapitre, nous proposons un nouveau modèle de compression entièrement différentiable appelé CAESR, permettant de transmettre les informations spatiales manquantes du côté du récepteur après suréchantillonnage.

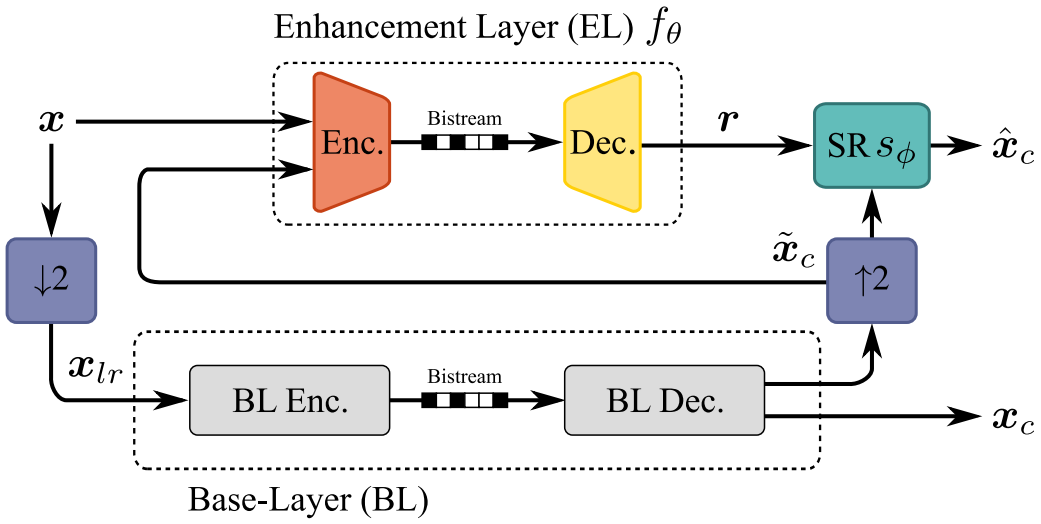


Figure 7 – Architecture de CAESR.

L'architecture de CAESR est illustrée dans la Figure 7. Cette approche est basée sur le principe de codage conditionnel permettant d'apprendre une combinaison non-linéaire de la source et du signal de la couche de base suréchantillonné, offrant de meilleures performances que le codage résidual classique. Premièrement, l'image source $\boldsymbol{x}$ et la reconstruction de la couche de base suréchantillonnée $\tilde{\boldsymbol{x}}_c$ sont concaténées. Le tenseur résultant de cette combinaison $(\tilde{\boldsymbol{x}}_c, \boldsymbol{x}) \in \mathbb{R}^{W \times H \times 6}$ est alors encodé par la partie d'analyse de f_θ en un vecteur latent $\boldsymbol{y}$. Ce vecteur latent est ensuite quantifié et encodé comme décrit dans le Chapitre 2. Côté décodeur, le signal résiduel r est reconstruit par la partie de synthèse de f_θ et concaténé avec l'image suréchantillonnée de la couche de base pour former l'entrée du réseau de super-résolution s_ϕ. Enfin, l'image de sortie $\hat{\boldsymbol{x}}_c$ est reconstruite par le réseau s_ϕ.

Les résultats montrent que l'ajout de ces métadonnées permet de stabiliser les performances du suréchantillonnage spatial pour une configuration *All Intra*. Les performances de l'algorithme ont également été comparées à LCEVC et SHVC pour la configuration *Random Access*. L'algorithme proposé offre de meilleures performances que LCEVC pour toutes les séquences. En revanche, pour certaines séquences, il est impossible d'améliorer les performances débit distorsion de la base suréchantillonnée sans métadonnées. Cela peut s'expliquer par la présence de corrélation temporelle dans les résidus, correctement exploitée par le codage hiérarchique de la couche de base. Un test à été réalisé montrant qu'une couche temporelle simple peut améliorer les performances pour ces séquences. Cependant, la propagation de l'erreur limitant le traitement temporel à des GOP de taille très réduites pose de nouvelles limitations.

Dans le contexte du cas d'usage de la thèse, l'ajout de ces métadonnées peut être intéressant pour apporter de la flexibilité de débit entre la couche de base et la couche d'amélioration. Ainsi, il est possible d'imaginer transmettre le signal 4K via cannal terrestre, et fournir les données d'amélioration 8K via un autre canal de transmission. Plusieurs débits ont donc été sélectionnées pour encoder la couche de base relativement aux débits utilisés sur la TNT française : 2.5Mpbs, 5Mpbs, 7.5Mpbs et 10Mpbs. Les résultats ont montré que pour une allocation plus importante de métadonnées que ce qui est permis pour l'amélioration du codec, les performances de CAESR sont supérieures à LCEVC. À la suite de ces résultats, une démonstration a été réalisée sur un dispositif d'affichage 8K afin de mettre en avant les performances de l'algorithme.

Conclusion

Dans cette thèse, plusieurs méthodes innovantes ont été proposées pour permettre de reconstruire un signal 8K à partir d'un signal 4K. L'étude proposée dans le Chapitre 3 à tout d'abord permis de poser les bases de la thèse en évaluant différents algorithmes de l'état-de-l'art pour répondre au cas d'usage posé. Cette étude a permis de définir deux axes de recherches. Dans un premier axe, nous avons développé deux algorithmes de super-résolution dédiés aux images et vidéos compressés. Ce premier axe a permis de démontrer que même avec un modèle adapté, certaines hautes fréquences ne peuvent pas être restituées sans information complémentaire côté récepteur, rendant les performances d'un schéma de codage basé sur un suréchantillonnage spatial dépendant du contenu. Ainsi, une méthode basée sur l'apprentissage de métadonnées spatiales a été développée dans le Chapitre 6. Ces approches originales ont permis d'apporter des gains par rapport à l'état-de-l'art et de dresser des perceptives d'amélioration par rapport au cas d'usage initialement défini. De plus, ces travaux de recherches ont été valorisés par des publications dans des conférences internationales, une revue scientifique internationale, un dépôt de brevet et des contributions en normalisation.

Le Chapitre 5 aborde des pistes d'amélioration pour la tâche de super-résolution, notamment l'aspect temporal pour retrouver des détails manquants dans les sous-pixels des images voisines. Les performances pourraient aussi être améliorées en utilisant l'information contenue dans le flux compressé de manière plus approfondie. En effet, certains travaux considèrent l'utilisation d'information de partitionnement ou de prédiction dans le calcul de l'image haute résolution courante. Pour finir, de nouvelles architectures multitâches plus avancées pourraient être développées afin de consolider les conclusions. Ces pistes n'ont pas été abordées dans le cadre de cette thèse.

Le point d'amélioration majeur du Chapitre 6 est l'extension de CAESR vers une méthode à débit variable. En effet, chaque point de codage obtenu par l'approche proposée est issu d'un entraînement propre. Cependant, il est nécessaire en pratique d'avoir la possibilité d'allouer dynamiquement le débit en fonction du contenu et de la bande passante disponible. De plus, toutes ces opérations étant réalisées côté décodeur, une attention particulière doit être apportée à la complexité de ces algorithmes afin de limiter la consommation d'énergie dans un récepteur. Enfin, des tests subjectifs devront être réalisés pour valider les performances de l'algorithme proposé.

Pour conclure, cette thèse nous a permis de constater que les récentes méthodes de compression basées IA permettent d'améliorer la reconstruction d'un signal vidéo 8K par rapport à un signal 4K. Ainsi, des algorithmes innovants exploitant les corrélations entre les deux résolutions de signal ont été développés. Là où une diffusion 8K sur la TNT française n'est absolument pas envisageable due aux contraintes de débit et de rétrocompatibilité, il est possible d'imaginer de transmettre un flux complémentaire via un autre canal de transmission, e.g. via IP, afin de retrouver les détails 8K dans un récepteur hybride.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

ADAM adaptive moment estimation. 120

AE-HP autoencoder with hyperprior. 109

AI artificial intelligence. 5, 25, 30, 31, 51, 55, 74, 88, 105

AI all intra. 112, 117

ALF adaptative loop filtering. 25, 39

AMVP advanced motion vector prediction. 21

ANN artificial neural network. 32

AOM alliance for open media. 19

ATSC advanced television systems committee. 73

AV1 AOMedia Video 1. 19

AVC advanced video coding. 13, 18–22, 24, 26, 29, 41, 50

BD Bjontegaard-Delta. xxi, xxv, xxvi, 15, 59, 64, 69, 71, 75–80, 95, 98, 114, 117

BL base-layer. xxii, xxvi, 4, 5, 25, 27–29, 50, 51, 55, 72–76, 106–108, 111, 113–118, 120–122, 124, 125, 130, 132

BTC basic test cell. xxii, 61, 63

CABAC context-adaptive binary arithmetic coding. 23, 25

CAE context-aware encoding. 12

CAESR conditional autoencoder and super-resolution. xxiii, xxvi, 5, 107, 111, 114–117, 119, 120, 122, 124, 125, 131

CNN convolutional neural network. xxii, 33–35, 37, 39, 41–44, 47, 74, 75, 80, 97, 98, 124, 125, 132

cpd cycles per degree. 11

CR compact-resolution. 42, 43, 132

CTC common test condition. xxiii, 56, 60, 74, 91, 113

CTU coding tree unit. xxi, 20, 23, 40, 41, 78

CU coding unit. 20, 23, 29

DCT discrete cosine transform. xxi, 13, 18, 22, 25, 28

DLM detail loss metric. 17

DMOS differential mean opinion score. xxii, xxv, 62, 63, 65, 66, 68–71

DNN deep neural network. 31, 33, 86, 97, 99, 106

DSCQS double stimulus continuous quality scale. x, xxii, 59, 61

DST discrete sine transform. 22, 25

DTT digital terrestrial television. 4, 5, 11, 13, 55, 107, 118, 119, 122, 131

DVB digital video broadcasting. 11, 29, 73, 131

DVC deep video coding. 49, 50

DWA dynamic weight average. 87, 92, 96, 100, 102–105

EDSR enhanced deep super-resolution. xxii, xxiii, xxvi, 5, 36, 37, 72, 74–80, 85, 88, 89, 91, 92, 94–98, 102–104, 106, 110

EL enhancement-layer. xxii, 5, 25–29, 50, 51, 72–74, 76, 80, 107, 108, 111–116, 118–121, 125, 131, 132

EVC essential video coding. 19

FC fully connected. 33

FOV field of view. 11

fps frame per second. 8

FR-IQA full-reference image quality assessment. 99

GAN generative adversarial network. 37, 42, 85, 98

GDN generalized divisive normalization. 45, 110

GMM gaussian mixture model. 48, 108

GOP group of pictures. xxiv, 19, 23, 29, 38, 49, 50, 111, 116–118, 124, 132

GPU graphic processing unit. 125

HD high-definition. v, xxi, 3, 11, 19, 50, 91

HDR high dynamic range. 3, 10, 24

HDTV télévision haute définition. v

HDTV high-definition television. xxv, 3, 10

HEVC high-efficiency video coding. vi, vii, xxi, xxiii, xxv, 3–5, 8, 12, 13, 17, 19–27, 34, 39–41, 50, 55, 59–61, 64, 69, 71–80, 82–84, 108, 111, 113, 114, 118, 124, 130, 131

HFR high frame rate. 3, 10

HHI Fraunhofer Heinrich-Hertz-Institut. 12, 57

HLS high-level syntax. 23, 26, 73

HM-16.20 HEVC test model. xxiii, xxv, xxvi, 5, 23, 59, 60, 64, 66, 70, 74, 76–80, 107, 111–114, 117

HR high-resolution. 28, 34, 35, 37, 38, 41, 42, 51, 74–76

HVS human visual system. 9, 16, 17, 37, 99

Hz Hertz. 8, 12

IBC international broadcast convention. 12

IDR instantaneous decoding refresh. 23

IEC international electrotechnical commission. 18, 27

ILF in-loop filter. 19, 22, 25, 38, 39

ILP inter-layer processing. xxi, 26, 27, 130

ILR inter-layer reference. 26, 27

IP internet protocol. 118, 131

IQA image quality assessment. 99

ISO international organization for standardization. 18, 27

ITE Institute of Image Information and Television Engineers. 57

ITU international telecommunication union. 3, 10, 11, 18, 24

ITU-T international telecommunication union. 16

JCT-VC joint collaborative team on video coding. 23, 26

JPEG joint photographic expert group. 41, 45

JVET joint video exploration team. v, xxiii, 3, 19, 24, 56, 91, 113

JVT joint video team. 18

KROCC Kendall's rank-order correlation coefficient. xxv, 70, 72

LCEVC low complexity enhancement video coding. vii, xxi, xxvi, 4, 5, 17, 27–29, 106, 107, 111, 114, 115, 117, 119, 120, 122, 124, 131

LD low delay. 49

LDB low-delay B. 23

LDP low-delay P. 23

LR low-resolution. 34–36, 40, 41, 74–76, 85, 96, 105, 106

LSTM long short-term memory. 40

LTM-4.0 LCEVC test model. 114

MFM motion field mapping. 27

MFSR multi-frame super-resolution. 34, 37, 38, 40

ML machine learning. 17, 31, 64

MOS mean opinion score. 17, 64

MPEG moving picture expert group. 3, 18, 19, 24, 26–28, 41, 49, 59, 98, 106, 130, 131

MS-SSIM multi-scale structural similarity. xxii, xxv, 17, 37, 64, 65, 70–72, 75–79

MSE mean squared error. 16, 36, 37, 45, 109

MTL multitask learning. 86

MV motion vector. 21

NN neural network. 32

NR-IQA no-reference image quality assessment. xxiii, 5, 85, 97, 99–101, 103, 104, 106

OTA over the air. 29

OTT over the top. 29

PLCC Pearson's linear correlation coefficient. xxv, 70, 72

PQF peak-quality frame. 40

PSNR peak signal-to-noise ratio. xxii, xxv, xxvi, 16, 17, 24, 29, 36, 37, 64, 65, 70–72, 75–80, 91, 93, 94, 96, 98, 102–105, 112, 114, 115, 120, 131

PU prediction unit. 20, 21, 23

PVS processed video sequence. 59, 61, 62

QE quality enhancement. 34, 39, 40, 50, 88

QoE qualité d'expérience. v, vii

QoE quality of experience. 3, 10, 55

QP quantization parameter. xxv, 14, 22, 40, 60, 75, 79, 89, 93, 102, 103, 111, 114

RA random access. xxi, 23–25, 49, 59, 60, 75, 114, 117

RB residual block. xxvi, 91–93, 96, 110

RD rate-distortion. xxiii, xxiv, 15, 23, 41–44, 49, 59, 64, 65, 76–78, 113, 114, 117, 124, 132

RDO rate-distortion optimization. 23, 39, 40, 49

ReLU rectified linear unit. 32

ResNet residual network. 33, 36, 40

RL reference layer. 27

RMSE root mean-squared error. xxv, 70, 72

RNN recurrent neural network. 38, 40, 50

RPR reference picture resampling. 25, 41

RQO resolution quantization optimization. 42

RR-IQA reduced-reference image quality assessment. 99

SAO sample adaptative offset. 22, 23, 38

SBTVD Brazilian television system. 29

SGD stochastic gradient descent. 32

SHM-9.0 SHVC test model. 74, 76, 114

SHVC scalable high-efficiency video coding. vii, xxi, xxiii, xxv, xxvi, 4, 5, 17, 26, 27, 55, 72–74, 76, 77, 80, 106, 107, 111, 114, 117, 129

SI-TI spatial and temporal information. xxii, 58, 64

SISR single image super-resolution. 34, 35, 37, 38

SNR signal-to-noise ratio. 18

SR super-resolution. 33–36, 38, 39, 42, 43, 51, 88, 110, 113

SRCNN super-resolution convolutional neural network. xxii, 35, 36

SROCC Spearman's rank ordered correlation. xxv, 70, 72

SSIM structural similarity. xxii, xxv, xxvi, 16, 17, 70–72, 91, 94, 96, 98, 100–105

SVC scalable video coding. 4, 26, 27

SVM support vector machine. 17, 40

TI temporal information. 17

TMVP temporal motion vector predictor. 27

TNT télévision numérique terrestre. vi

TU transform unit. 20, 22, 23

UHD ultra high definition. 10, 11, 19, 50, 72, 85

UHDTV télévision ultra-haute définition. v

UHDTV ultra-high-definition television. xxv, 3, 10

VAE variational autoencoder. xxii, 43–45, 47, 50

VCEG video coding expert group. 3, 18

VIF visual information fidelity. 17, 79

VLC variable-length coding. 18, 23

VMAF video multimethod assessment fusion. xxii, xxv, 17, 29, 64, 65, 70–72, 75–79, 114

VOD video on demand. 4

VTM VVC test model. 25

VTM-11 VVC test model. xxv, 59, 60, 64, 66, 70, 74–80, 91, 100

VVC versatile video coding. v, xxi, xxiii, xxv, 3–5, 8, 12, 17, 19, 24, 25, 28, 34, 39–41, 47, 50, 55, 59–61, 64, 69–72, 74, 75, 77–80, 82–84, 88, 89, 91, 96, 104–106, 108, 129–131

WCG wide color gamut. 3, 10

PART I

Introduction

CHAPTER 1

INTRODUCTION

1.1 Preamble

With the latest UHDTV system deployment [24], the quality of experience (QoE) of users is expected to improve by introducing new features to the existing HDTV system [25], including high dynamic range (HDR), wide color gamut (WCG), high frame rate (HFR), and higher spatial resolutions. Among those new features is the 8K video resolution, a spatial resolution of 7680×4320 pixels, corresponding to four times more pixels than 4K (3840×2160) and 16 times more pixels than HD (1920×1080). Such a spatial resolution allows expending the sensation of realness perceived by the observer by increasing the pixel count and thus the amount of detail reproduced from the captured scene. 8K video resolution has attracted a lot of interest from the industry with new 8K sensors, TV screens, and experimental tests performed by broadcasters, e.g. in Japan with the NHK. Nevertheless, this large amount of spatial information requires high resources in terms of bandwidth to ensure good video quality, bringing new challenges to broadcasters.

Thus, new video coding standards emerged under the pressure of the different industrial actors, like VVC/H.266 [3], finalized in July 2020. This standard is born from a collaborative work carried out by the joint video exploration team (JVET) composed of experts from the international telecommunication union (ITU) and the ISO/IEC represented by the video coding expert group (VCEG) and the moving picture expert group (MPEG), respectively. The objective of VVC was to reduce the bitrate required by high-efficiency video coding HEVC [4] by 40% for the same visual quality. Like its predecessor did for introducing 4K services, VVC should allow the deployment of new immersive services like 360° videos and 8K.

1.2 Use case and motivations

In France, technical experiments have been launched to prepare a technological switch on the French digital terrestrial television (DTT) by 2024. This work aims to deploy 4K services in DVB-T2/HEVC coupled with new TV services, e.g. replay, video on demand (VOD), push. For instance, experimental tests have been performed using bitrates from 10 to 17Mbps (4K) and 3 to 17Mbps (HD) for 2 to 3 program delivery with HEVC. In this context, the introduction of new video formats has to consider backward compatibility with legacy receivers to keep the audience's reach. However, the high bitrate requirements of 8K prevent the deployment of this technology on terrestrial networks with 4K simulcast, i.e., both 8K and 4K are encoded independently and muxed. On the one hand, several alternatives to simulcast exist, like scalable video coding, proposed by scalable extension of codecs like scalable video coding (SVC) and SHVC. However, due to a bitrate gain being too low compared to the practical constraints imposed by scalability, this type of architecture is not adopted in the industry. On the other hand, AI-based video coding and post-processing technologies have recently shown outstanding performance compared to traditional algorithms. The potential of these technics for video compression is already well-known in the literature, with, for instance, super-resolution applied as post-processing [6] or end-to-end video coding [7]. These algorithms could recover 8K resolution from a 4K signal upscaling with or without metadata, e.g. LCEVC [8].

This work aims to explore and develop efficient and innovative algorithms allowing the reconstruction of an 8K signal from a 4K one. In a first step, a subjective study evaluates the performance of different compression methods and post-processing algorithms for 8K video coding. Then, new super-resolution methods based on multitask learning has been proposed to improve performance on compressed videos. Finally, a new layered compression scheme, called CAESR, has been proposed. This coding scheme is based on a conditional autoencoder that encodes the residual between the upscaled BL signal and the source video.

1.3 Outlines

Chapter 2 defines the state-of-the-art next-generation video formats and coding algorithms. This chapter first describes the video signal's characteristics and introduces the standardized UHDTV video formats, focusing on 8K video resolution. Then, the hybrid compression model and the single-layer HEVC and VVC standards are then briefly detailed. Finally, layered-coding architectures, including SHVC of codecs and LCEVC, are presented.

Chapter 3 provides the state-of-the-art AI-based algorithms for video compression, including deep learning-based models replacing conventioanl pre and post-processing systems to the coding algorithm itself. This section first gives a brief overview of neural networks. Then, we discuss how artificial intelligence (AI)-based restoration algorithms can be integrated as pre and post-processing into a coding pipeline. Finally, this chapter reviews the recent promising approaches proposed to replace traditional image and video codecs.

Chapter 4 provides objective and subjective evaluations of standards and algorithms using a dedicated 8K resolution video dataset. First, this chapter describes our 8K video test dataset. Then, the objective and subjective quality of HEVC and VVC for 8K video coding are assessed. This section also evaluates the perceptual gain offered by 8K over 4K for each tested scene. Finally, this chapter also considers algorithms enabling 8K video delivery with 4K backward compatibility, including SHVC and spatial upscaling using super-resolution and a Lanc zos filter.

Chapter 5 explores multitask-based architectures for super-resolution on compressed contents. This process allows performing multiple tasks with a single shared network, reducing the total number of parameters. Advanced training strategies, such as prior information using qp_{map} and network pre-training, are also investigated to improve the network's performance on compressed low-resolution inputs. First, this chapter presents the principle of multitask learning and different possible architectures. Then, we introduce MTL-EDSR, a multitask network that performs super-resolution and quality enhancement using a single shared network. Finally, we presents MTL-Unet, an extension of MTL-EDSR dedicated to super-resolution and high-level vision tasks, namely, no-reference image quality assessment (NR-IQA) and semantic segmentation.

Chapter 6 presents CAESR. This learning-based layered approach uses a conditional autoencoder as an EL model and a conventional single-layer codec as a BL model. The presented method is trained to encode the residual between the upscaled reconstructed image and the source. First, this chapter presents the overall pipeline of the proposed solution. Then, we evaluate our algorithm as a codec enhancer based on the HM-16.20. We first provide an ablation study that validates the efficiency of our method. Then, we compare it against state-of-the-art layered approaches, including LCEVC and SHVC, for single-layer codec enhancement. Finally, our algorithm is assessed to deploy of new services by considering 4K video delivery on top of an HD signal regarding typical bitrates used for DTT broadcast.

PART II

State of the Art

CHAPTER 2

NEXT-GENERATION VIDEO CODING

2.1 Preamble

With the emergence of new digital video formats, the consumer's demand for more immersive video services, such as higher spatial and temporal resolutions, increases. However, high bitrate requirements are needed for this kind of service, challenging their deployment on broadcast infrastructures. To tackle this, video coding technologies have been standardized over the years to provide efficient compression algorithms and enable the deployment of new immersive video services. For instance, contributions to video coding standards like HEVC [4, 26] or its successor VVC, finalized in July 2020 as ITU-T H.266 | MPEG-I - Part 3 (ISO/IEC 23090-3) standard [3, 27], enable video signal compression to be continuously improved through the standardization bodies.

This chapter includes background on video coding technologies, from video signal characteristics and formats to the recently developed coding standards, including single-layer and layered technologies.

2.2 Background on video signal

This section first defines the characteristics of a video signal. Then, the standardized UHDTV video formats are presented, focusing on 8K video resolution.

2.2.1 Video signal characteristics

A video signal corresponds to a sequence of pictures, also called frames, characterized by a spatial resolution $W \times H$ representing the number of pixels in each row and column, respectively. The video frames are presented sequentially at a given temporal frequency, called framerate or temporal resolution, expressed in Hertz (Hz) or frame per second (fps). An example of a video sequence is given in Figure 2.1, with f denoting the framerate.

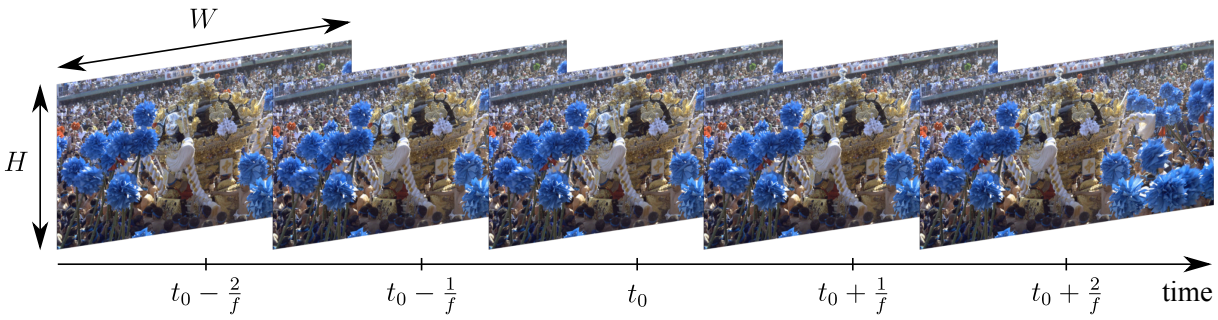
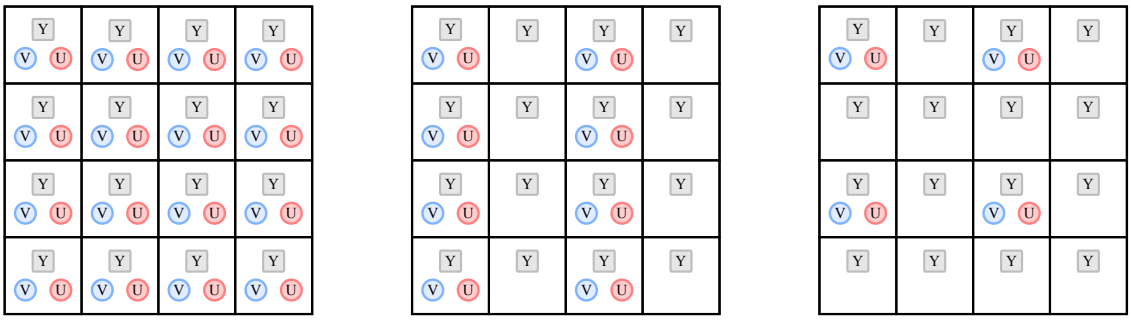


Figure 2.1 – Illustration of the temporal and spatial resolution of a video sequence.

The pixels are composed of different channels (usually 3), which contain the value of the given component. For instance, when the pixels are represented in the RGB space, each component defines a color sample (red, green, and blue). However, since the human visual system (HVS) is more sensitive to the luminance component of an image, pixels are more commonly represented in the YUV (YCbCr) color space, where Y corresponds to the luma component and UV the chroma components. Moreover, it decorrelates the signal as the more significant visual information is concentrated into one channel. Thus, chroma subsampling can be performed to reduce the amount of information in the raw representation of the sequence with a limited impact on the visual quality. The three YUV subsampling types are illustrated in Figure 2.2. In 4:4:4 sampling, YUV components are fully and equally sampled. In 4:2:2 sampling, U and V components are downscaled horizontally by a factor of 2. In 4:2:0 sampling, both U and V channels are downscaled by a factor of 2.



(a) 4:4:4 (b) 4:2:2 (c) 4:2:0

Figure 2.2 – YUV subsampling.

The maximum value of pixels depends on the number of bits used to represent the signal components. Typically, raw video frames are encoded from 8-bits to 16-bits per pixel per channel. Thus, the maximum possible values per channel of a pixel for a frame sampled in

Table 2.1 – HDTV and UHDTV standards parameters.

Parameter	HDTV	UHDTV
Spatial resolution	1920×1080	3840×2160, 7680×4320
Aspect ratio	16:9	16:9
Frame frequency (Hz)	60, 60/1.001, 50, 30, 30/1.001, 25, 24, 24/1.001	120, 60, 60/1.001, 50, 30, 30/1.001, 25, 24, 24/1.001
Standard viewing angle	30°	100°
Scanning	Interlaced, Progressive	Progressive
Sampling lattice	Orthogonal	Orthogonal
Color gamut	BT.709 [2]	BT.2020 [1]
Dynamic range	SDR	SDR, HDR
Pixel aspect ratio	1:1 (square pixels)	1:1 (square pixels)

8-bits is $2^8 = 256$ and $2^{10} = 1028$ in 10-bits. Therefore, given a 60fps 7680×4320 raw video sequence sampled in YUV4:2:0 10 bits, the total amount of bits is computed as:

$$\frac{60 \text{ images}}{1 \text{ second}} \times \frac{7680 \times 4320 \text{ pixels}}{1 \text{ image}} \times \frac{1.5 \text{ channels} \times 10 \text{ bits}}{1 \text{ pixel}} \approx 29.86 \text{ Gbits/s} \tag{2.1}$$

2.2.2 UHDTV video formats

The type of delivered video formats highly influence the end-user QoE. For instance, ultra high definition (UHD) videos increase the degree of immersion by providing more spatial information. In contrast, higher framerates reduce the motion blur in the scene, which makes the video looks more realistic. Other features contribute to the QoE. For instance, expanding the color gamut, defined as limiting the range of colors represented in a particular framework [28], allows rendering richer colors, providing more realness to the user. Furthermore, increasing the Dynamic Range, which defines the ratio of the maximum light intensity to the minimum light intensity [29], allows reproducing more natural luminance variations on the screen.

In order to ensure uniformity between industrial actors and consumers, digital video formats have been standardized by the ITU. The latest UHDTV system [24] introduced new features to the existing HDTV system [25]. In addition to HFR and higher spatial resolutions, including 4K (3840x2160) and 8K (7680x4320), UHDTV consider the integration of other immersive video formats, including HDR and WCG, i.e., BT.2020 [24]. The parameters of UHDTV and HDTV systems are given in Table 2.1.

The digital video broadcasting (DVB) consortium [30] defined three phases for introducing these formats on broadcast infrastructures, including UHD-1 Phase 1, UHD-1 Phase 2, and UHD-2 [31]. The delivery deployment of such video services broadcast infrastructures is a real challenge and requires efficient compression methods to reach the available throughput while ensuring high video quality. This manuscript addresses the problem of 8K video signal delivery over terrestrial broadcast (DTT). More details about this video format are given in the following.

2.2.3 8K resolution video

8K resolution videos (7680×4320) contain four times more pixels than 4K (3840×2160) and 16 times more than HD (1920×1080). An illustration of these three spatial resolutions is given in Figure 2.3. As mentioned, increasing the pixel count in videos significantly contributes to the user's sensation of realness.

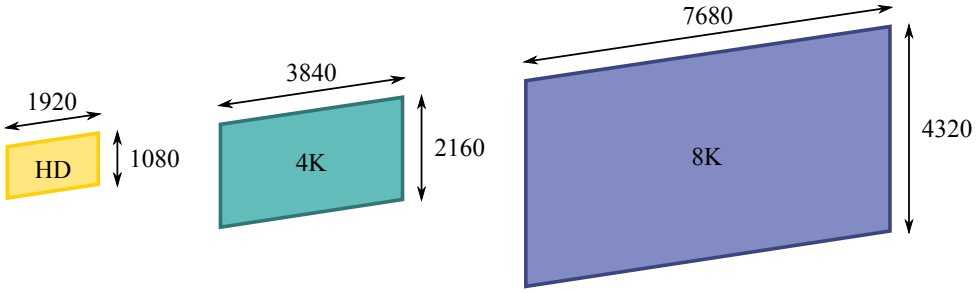


Figure 2.3 – Illustration of HD, 4K and 8K spatial resolution.

Appropriate viewing distance has to be defined to take maximum advantage of this resolution. Thus, the ITU-R provides guidelines on the viewing distance based on the screen's height [24]. For instance, the optimal viewing distance is 0.8H for 8K video and 1.5H for 4K. These values are selected based on the relationship between the angular resolution and the horizontal field of view (FOV), also called viewing angle. An illustration of the viewing distance and the viewing angle is given in Figure 2.4. Indeed, the standard viewing distance is set so that the horizontal FOV of one pixel corresponds to one arc per minute, which leads to a critical pattern frequency of 30 cycles per degree (cpd). In other words, it corresponds to the distance a viewer can distinguish between two pixels. As mentioned in [32], 8K resolution allows increasing the FOV up to 100 FOV while maintaining the angular resolution limit. Thus, as shown in Table 2.1, in addition to the introduction of UHD video formats, the FOV is increased to 100°. Moreover, increasing angular resolution improves visual fidelity while increasing the FOV and impacts the realness and the sensation of "being there" [32].

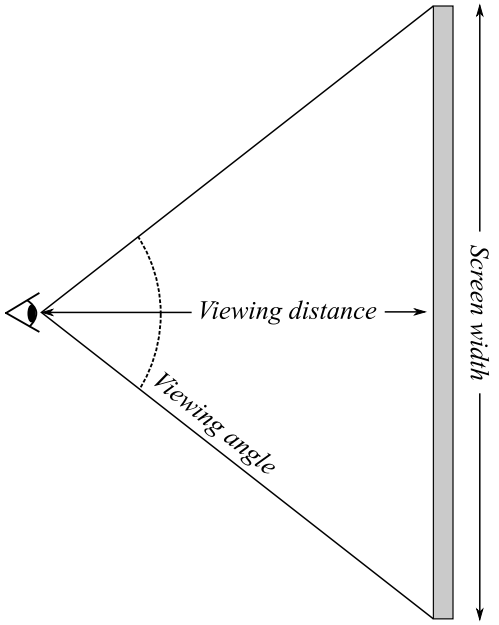


Figure 2.4 – Illustration of viewing distance and viewing angle.

Several perceptual studies have demonstrated the perceptual gain of 8K resolution. For instance, the QoE of 8K contents have been evaluated regarding different use-cases by using specific contents [33], e.g. food, people. It demonstrated that viewers experience high-order psychological effects when watching 8K videos, such as the impression of freshness and deliciousness in the content. In 2013, a subjective experiment was conducted to assess the image quality of the world's first 8K 60-Hz HEVC real-time encoder. Several studies have shown that the bitrate required for 8K applications is approximately 80Mbps using HEVC [34, 35, 36]. In 2019, a demonstration at the international broadcast convention (IBC) presented 8K video encoding using HEVC with context-aware encoding (CAE) and demonstrated that the bitrate could be lowered to around 14Mbps [37]. More recently, a demonstration proposed by the Fraunhofer Fraunhofer Heinrich-Hertz-Institut (HHI) compared their open-source VVC video codec, called Vvenc [1] [38], with x265 [2] (HEVC) for 8K video coding [39]. The results demonstrated that 8K video encoded at 25Mbps using VVC could reach the same quality that 8K encoded at 50 Mbps using HEVC.

In practice, an 8K 120Hz HEVC codec [40, 41] has been used for Japan's satellite broadcasting by using DVBS2X [42]. In that case, the use of a complete transponder or multiple bonded transponders can reach bandwidth in the range 70-80Mbps. Also, experiments in several

1. `https://github.com/fraunhoferhhi/vvenc`
2. `https://trac.ffmpeg.org/wiki/Encode/H.265`

Table 2.2 – Summary of 8K (7680×4320) trials on terrestrial television networks [5].

Country	DTT system	Channel bandwidth	Multiplex capabilities	Signal bitrate	Compression standard	framerate
Japan	ISDB-T	6 MHz	91.8 Mbps	91 Mbps	AVC	59.94Hz
Spain	DVB-T2	8 MHz	36.72 Mbps	32 Mbps	HEVC	50Hz
Brazil	ISDB-T	6 MHz	91.8 Mbps	85 Mbps	HEVC	59.94Hz
China	DTMB-A	8 MHz $\times$ 4	200 Mbps	120 Mbps	AVS3	50Hz

countries have been conducted regarding 8K service deployment on DTT networks [5]. An overview of these experiments settings is given in Table 2.2.

2.3 Overview of video coding

This section presents the different coding blocks of the well-known hybrid compression model. Then, details on the performance evaluation of compression algorithms are given.

2.3.1 Hybrid compression model

Since H.261 [43], the hybrid compression model has been widely integrated into codecs. The principle of this coding framework is to integrate a decoder inside the encoder, which performs spatial and temporal predictions regarding the already decoded samples. Thus, the residual error can be sent along with the prediction information, such as the motion vectors and the intra prediction mode, to reduce the temporal and spatial redundancy in the transmitted signal. The basic steps of hybrid compression systems are partitioning, prediction, transform, quantization, and entropy coding. An overview of the hybrid compression model is given in Figure 2.5.

The first step of the hybrid compression model is to represent the frames into blocks by recursively splitting them into partitions. The partition granularity offers flexibility with flat and complex spatial areas in the input signal. Each block is then processed individually for intra and inter prediction. The former mode exploits the spatial redundancy between blocks by using the spatially adjacent blocks to predict the current samples. The second type of prediction relies on the temporal correlation among frames by predicting samples from the previously decoded frames. The residual error, i.e., the difference between the predicted and original samples, is then computed and transformed using 2D separable linear transforms, such as the DCT. This step is crucial in hybrid compression systems as it performs energy compaction and signal separation

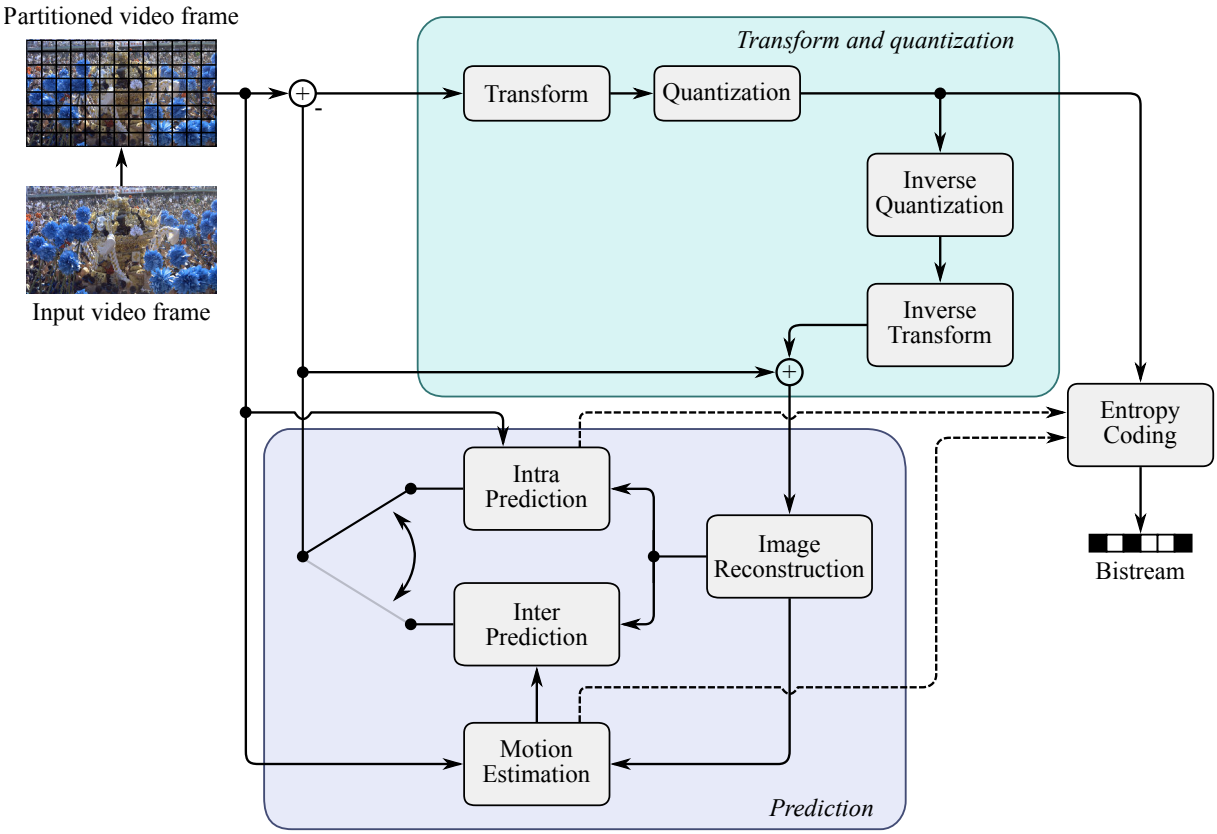


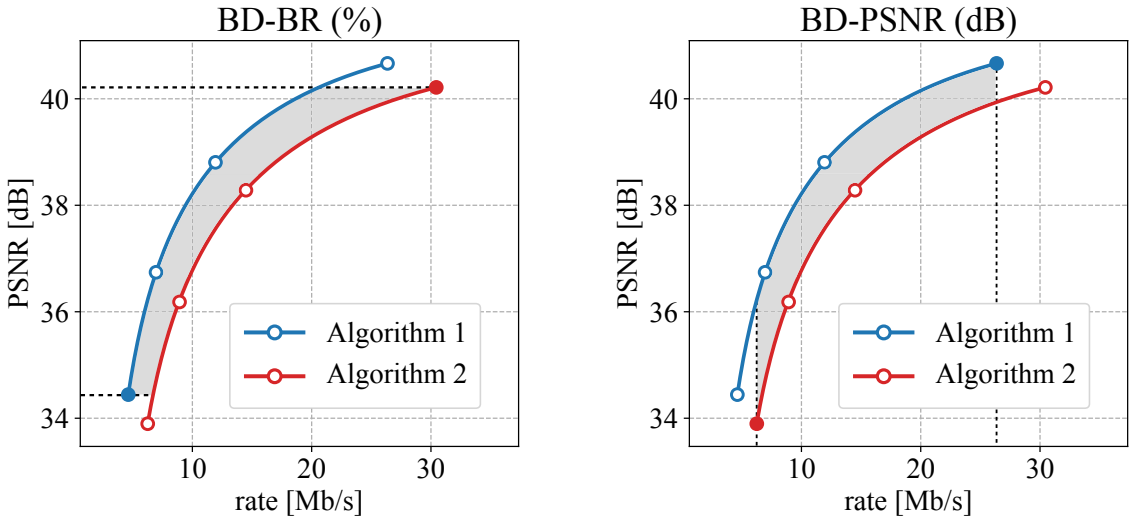
Figure 2.5 – Hybrid compresion model architecture (encoder).

into frequency components. The transformed residual signal is next quantized to a discrete set of values. This process corresponds to the lossy step of the hybrid compression framework. Uniform quantization is usually applied with a quantization step parametrized by a quantization parameter (QP) to adjust the quantization level. As the signal is transformed, quantization can be performed on the higher frequency components using adapted scanning approaches, resulting in a lower perceptual loss. The last step of the hybrid coding system is to represent the quantized and transform signal into a sequence of bits. It is generally performed by entropy coding, which allocates bitrate regarding the probability of the signal, i.e., more bits are assigned to the less common sources. In addition, the side information related to prediction modes, including motion vectors and intra prediction mode, is also transmitted to allow performing the prediction at the decoder side.

2.3.2 Evaluation of coding performance

Performance evaluation

The performance of compression algorithms can be evaluated based on multiple factors, including their RD performance and their complexity. The RD performance assesses the degradation introduced on the signal at a given compression ratio due to the lossy nature of compression systems. Thus, RD curves can be plotted to illustrate this relationship for different operating points, as shown in Figure 2.6. The more the curve related to "Algorithm 1" is above the curve associated with "Algorithm 2", the better the performances of "Algorithm 1" are compared to "Algorithm 2".



(a) BD-BR area

(b) BD-PSNR area

Figure 2.6 – Illustration of the BD assessment method.

The BD [44] method allows quantifying the RD performance of the "Algorithm 1" compared to the "Algorithm 2", called the anchor. A minimum of four operating points for both algorithms are required for this method. This metric calculates the area between the two curves, indicating how superior "Algorithm 1" is over "Algorithm 2". It can be computed to quantify either the average proportion of bitrate saved regarding the same quality or the average quality gain regarding the same bitrate. First, the curves are interpolated to be defined as a log-based third-order polynomial as:

$$f(Q) = a + b \cdot Q + c \cdot Q^2 + d \cdot Q^3, \tag{2.2}$$

where Q denotes the quality, $f(Q)$ the bitrate as the function of the quality, and a, b, c, and d are the fitting parameters. Then, the difference between the intervals of the interpolated curves is computed to obtain the quantities in the grey areas represented in 2.6.

As mentioned, the complexity of compression algorithms can also be used as a performance indicator. This measure is typically assessed by the encoding/decoding times and the memory consumption required by the evaluated compression system.

Video quality assessment

Evaluating the distortion between a source signal and its representation reconstructed by lossy compression algorithms is a very active field of research. As mentioned, it allows assessing the performance of compression algorithms, which is critical for their benchmark and to contribute to their improvement. However, evaluating perceptual quality is challenging as the degradations introduced during compression are not equally processed by the HVS.

Subjective evaluation with human observers is the most accurate way to assess the quality of videos. Thus, the international telecommunication union (ITU-T) continuously standardizes subjective test methodologies to facilitate the analysis and reproducibility of perceptual studies [45, 11]. Although optimal to assess the performance of compression algorithms, these methods are time-consuming and not adapted to video signal evaluation inside the coding pipeline. Hence, objective quality metrics are developed to simplify the assessment of the reconstructed signal's quality. For instance, the PSNR, which is a prevalent objective metric emanated from the mean squared error (MSE), is computed as:

$$PSNR = 10 \times \log_{10} \frac{d^2}{MSE}, \tag{2.3}$$

where d is the maximal pixel value and the MSE is the difference between the square of two frames pixel by pixel. The MSE is a straightforward way to assess the quality of a compressed frame $\boldsymbol{y}$ compared to uncompressed frames $\boldsymbol{x}$:

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} || \, \boldsymbol{x}(i,j) - \boldsymbol{y}(i,j) \, ||^2 \, . \tag{2.4}$$

The PSNR is ubiquitous in the compression research field and the industry due to its low complexity. However, this objective metric is known to be a poor indicator of the actual quality perceived by the HVS, i.e., the end-user.

SSIM [46] is another popular objective quality metric. It assesses the structure similarity

between images rather than a pixel by pixel difference. As the HVS is sensitive to structural variations inside images, the SSIM offers closer performance to subjective observations than the PSNR [46]. This metric is commonly computed on the luma component only by applying a window (generally 8x8) on the degraded and original images. The key idea is to combine different components of the images to assess the degradation: the luminance l, the contrast c , and the structure s. The SSIM score is computed as follows:

$$SSIM(\boldsymbol{x}, \boldsymbol{y}) = l(\boldsymbol{x}, \boldsymbol{y}) \cdot c(\boldsymbol{x}, \boldsymbol{y}) \cdot s(\boldsymbol{x}, \boldsymbol{y}) = \frac{(2\mu_{\boldsymbol{x}}\mu_{\boldsymbol{y}} + c_1)(cov_{\boldsymbol{xy}} + c_2)}{(\mu_{\boldsymbol{x}}^2 + \mu_{\boldsymbol{y}}^2 + c_1)(\mu_{\boldsymbol{x}}^2 + \mu_{\boldsymbol{y}}^2 + c_2)}, \tag{2.5}$$

with $\boldsymbol{x}$ and $\boldsymbol{y}$ denoting the degraded image and the source image, respectively, and μ, σ and cov their mean, variance and covariance, respectively. The MS-SSIM [47] can be computed by applying the SSIM on different scale representations of the images to further improve the performance. Thus, the final score is computed as follows:

$$MS\text{-}SSIM(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{M} \sum_{i=1}^{M} SSIM(\boldsymbol{x_i}, \boldsymbol{y_i}) \tag{2.6}$$

with $\boldsymbol{X}$ and $\boldsymbol{Y}$, the source and reconstructed images, respectively, and $\boldsymbol{x_i}$ and $\boldsymbol{y_i}$ their representations at different scales with i the scale index. This objective quality metric offers high correlation scores with subjective test ratings [47].

Recently, Netflix developed an objective quality metric based on machine learning (ML). This metric, called VMAF [13], is trained to produce a score computed from different other metrics, including spatial quality indicators with visual information fidelity (VIF) [48], detail loss metric (DLM) [49] and temporal quality indicators with temporal information (TI). These simple objective metrics are given as input to a support vector machine (SVM) trained to match with mean opinion score (MOS) scores.

2.4 Standardization

This section depicted the process over the last decades. The single-layer HEVC and VVC standards are then briefly detailed. Finally, layered-coding architectures, including SHVC of codecs and LCEVC, are presented.

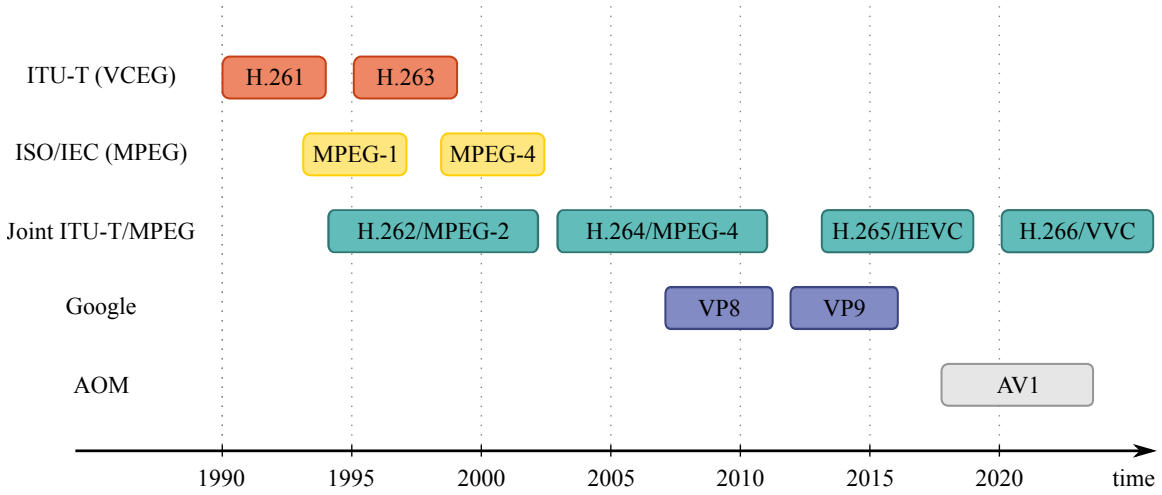


Figure 2.7 – Timeline of video coding technology and standards.

2.4.1 History of video compression standards

Video coding standards specify the bitstream syntax to unify the coding tools and ensure interoperability between industrial actors. It also allows promoting the mass adoption of codecs in the ecosystem. An overview of the video coding standardization timeline is given in Figure 2.7. In 1990, the VCEG developed H.261 [43], the first video compression standard adopting the hybrid coding scheme presented in Section 2.3.1. Standardized coding tools fulfill the roles of the different steps, such as the DCT for transform, motion compensation for inter prediction, and variable-length coding (VLC) for entropy coding. In 1993, MPEG, which started in 1988 as the result of collaborative work between the international organization for standardization (ISO) and the international electrotechnical commission (IEC), developed MPEG-1 [50]. This standard is mainly based on H.261 while including new tools like bi-directional motion prediction and slice structure coding. Later, MPEG-2 [51, 52] was standardized as another joint contribution between MPEG and ISO/IEC by incorporating new coding technologies to the existing MPEG-1 standard, e.g. interlaced frames and spatial and signal-to-noise ratio (SNR) scalability. This standard has been widely adopted and is still used in the broadcast ecosystem. Then, both groups continued developing their standards. The ITU developed H.263 [53] (1995), introducing P and B frames and arithmetic coding, and MPEG-4 Visual [54]–or ISO/IEC 14496-2– has been developed in parallel by MPEG allowing coding 10-bits and 12-bits raw videos. The subsequent success in video coding standards history came from H.264 [55], also known as AVC [56]. This standard has been developed from a collaborative work between MPEG and VCEG through the joint video team (JVT). It introduced new features to the existing MPEG-2 standard, such as

the deblocking in-loop filter (ILF). This standard encountered a high success for HD definition services deployment on broadcast infrastructures. In 2013, HEVC was born as Recommendation H.265 [4] or ISO/IEC 23008-2 [57] and offered 50% of video quality gain over AVC [58]. This standard has been mainly developed for UHD services broadcast. More recently, VVC has been developed by the JVET to meet the high bandwidth requirements of new immersive video formats, such as 8K. It has been demonstrated to offer 30-50% of subjective gains depending on the content [59]. More details about the two latest MPEG standards are given in the following sections.

To compete with those royalty-based standards, some organizations proposed developing their own. For instance, in 2010, Google developed VP8 [60] and later VP9 [61] (2012) to serve as the main encoder for Youtube. Based on this principle, the alliance for open media (AOM) was subsequently formed by Amazon, Cisco, Google, Intel, and Netflix to collaborate on a royalty-free codec, AOMedia Video 1 (AV1) [62]. Subjective quality assessment of AV1 has been performed for 4K video resolution and shown that, at the same video bitrate level, AV1 and HEVC are not significantly different in terms of perceived quality [63]. However, a patent pool has been finally established for AV1 [64], lowering the interest of the standard.

In parallel, MPEG specified another standard, called essential video coding (EVC) [65], based on two profiles, a baseline profile which contains public H.264 patents (royalty-free) and a main profile.

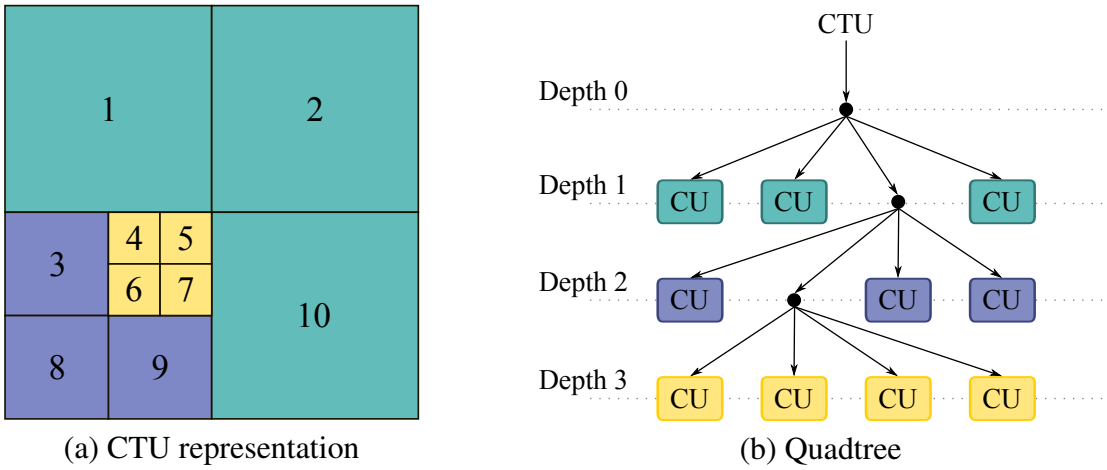
2.4.2 HEVC standard

Coding structure

As in previous MPEG standards, HEVC allows different types of frames, namely, intra (I), predictive (P), and bi-predictive (B). The I frames only consider the intra prediction for the processed samples. In addition to intra prediction, P and B frames allow unidirectional and bidirectional temporal prediction, respectively. The video frames are organized in a fixed and repetitive pattern called a group of pictures (GOP). The GOP size fits the maximum temporal distance of inter-prediction. In other words, the last frame of a GOP is either an I or a P frame. The Intra period defines the number of the frame before introducing an I frame, which provides random access points and stops error propagation.

Partitioning

In AVC, the maximum block size is set to 16×16 and can be further split until 4×4 blocks. To improve the coding performance for higher resolutions, HEVC defines the maximum block size as 64×64. Moreover, a CTU partitioning strategy is adopted to provide a recursive split of blocks. This partitioning allows a flexible and efficient way to match spatial variations in the input image. Larger blocks are efficient in compression flat areas, whereas smaller blocks are efficient to match with more complex textures. The partitions are proper to each YUV channel of the input image. An example of the HEVC CTU split is given in Figure 2.8.



(a) CTU representation (b) Quadtree

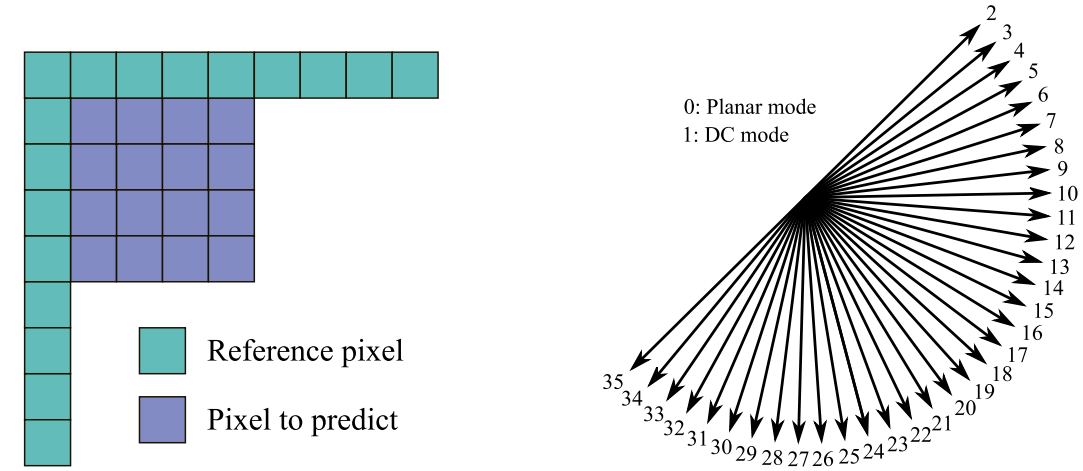
Figure 2.8 – Illustration of HEVC CTU quadtree partitioning.

Each coding unit (CU) is used for the prediction and the transform/quantization steps, denoted as prediction unit (PU) and transform unit (TU), respectively. The CU can be split following several granularity levels depending on the block type (skipped, intra, or inter). The partition decisions are then transmitted to the decoder in the bitstream using the syntax specified by the standard.

Intra prediction

Intra prediction is designed to predict block samples to encode by exploiting the spatial redundancy with previously decoded samples in the left and upper edges of the current block. As described in Figure 2.9, the available reference samples for HEVC are the $2N$ vertical and horizontal rows from the upper left corner samples. Three intra prediction modes are available in HEVC: the planar mode, the DC mode, and the angular mode [66]. The samples predicted using the DC mode use the constant average value of available collocated reference samples.

The planar intra prediction mode takes the mean of the horizontal and vertical variations of the reference samples to predict the current samples. It allows fitting with the boundary edges from the reference samples and preserves the continuity across the picture's spatial boundaries. Finally, the angular mode interpolates the reference samples from an angular prediction projection in a specific direction. This mode has been extended to 33 angles in HEVC compared to only 8 in AVC to increase flexibility and further improve prediction accuracy.



(a) Predicted and reference samples (b) Intra coding modes

Figure 2.9 – Illsutration of intra prediction in HEVC.

Inter prediction

Inter prediction exploits the temporal redundancy between frames to predict the current samples. This mode uses samples areas localized in the neighboring frames as a current samples reference. The block's temporal motion is computed by block matching algorithms applied on the reference frames from the previously coded reference pictures in both past and future. HEVC introduces several tools to reduce the signaling of motion information in the bitstream, namely advanced motion vector prediction (AMVP) and merge mode [67]. AMVP proposes to identify several prediction candidates and compete with them. Then, only the candidate's index and the residual motion vector (MV) between the current samples and the candidate. In addition, the merge mode was introduced where the current PU share and duplicate the spatial or temporal neighboring of the current block of the selected candidate.

Transform and quatization

After the appropriate prediction has been performed, the residual between the predicted and original samples is computed and transformed. As mentioned in Section 2.3.1, the transform step allows decorrelating the signal by applying 2D separable linear transforms. This principle enables separating the signal in the frequency region and compacting the signal's energy. In HEVC, the TU size can vary from 4x4 to 32x32. HEVC transform is an approximation of the DCT for inter and intra coded blocks. It also includes the possibility to perform discrete sine transform (DST) on 4x4 luma intra blocks. An illustration of the frequential bases of the DCT-II is given in Figure 2.10. Alternatively, the residual samples can be directly encoded without transform. Once transformed, the coefficients are quantized using different scanning approaches to mainly affect high-frequency components, reducing the perceptual loss. The quantization step is similar to AVC, with a QP ranging from 0 to 51 in HEVC. QPs values were selected so that an increase of 6 corresponds to a quantization step multiplied by a factor of two. Compared to AVC, HEVC include refined and new tools like additional scanning order, significance maps, sign coding, and coefficient level [68].

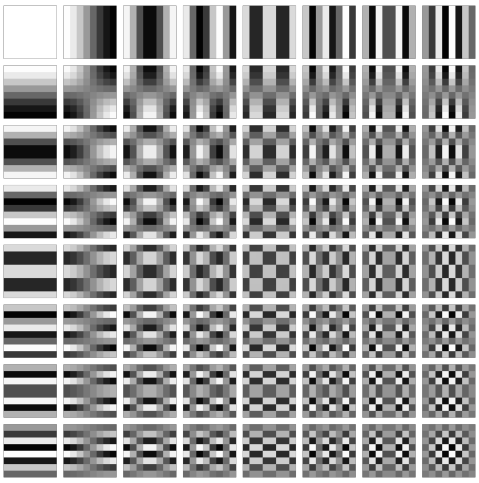


Figure 2.10 – Illustration of the frequential bases of the DCT-II for a 8×8 signal.

In loop filters

Since H.264, ILFs are introduced in the coding pipeline to improve the quality of the reconstructed frame that can be used as a reference for other frames. In addition a the deblocking filter [69], also used in AVC, HEVC introduced the sample adaptative offset (SAO) [70]. This filter is dedicated to reduce the ringing artifact. First, samples are classified, and an offset

is computed and applied to the reconstructed sample depending on the category. The SAO information is encoded in the bitstream at the CTU level.

Entropy coding

The final block of the coding chain corresponds to entropy coding. In HEVC, high-level syntax (HLS) elements are encoded using VLC and CU, PU, TU, and SAO information using context-adaptive binary arithmetic coding (CABAC) [71]. The CABAC algorithm has been improved in HEVC and follows three steps. First, the signal is binarized to produce binary symbols from the quantized representations. Then, context modeling is applied to better model the probability of the binary symbols. Finally, arithmetic coding is performed to losslessly compress the signal either on estimated probability or using equal probability.

Reference softwares

The HM-16.20 [72] is a reference implementation of the HEVC standard designed by joint collaborative team on video coding (JCT-VC) experts to evaluate its upper-bound performance. This software allows four possible types of configurations. The AI configuration considers intra-prediction only, where each frame is an instantaneous decoding refresh (IDR). The Low Delay configurations, i.e., low-delay P (LDP) and low-delay B (LDB), enable both intra and inter predictions. The frames are encoded in the same order as the display order. Thus, these configurations are adapted to applications requiring low latency, like live applications or video conferencing. Finally, the RA mode considers a hierarchical prediction scheme of frames. The frames are then reordered to the appropriate display order. This configuration provides the best performance while introducing delay corresponding to the size of the GOP. An illustration of a GOP in RA configuration is given in Figure 2.11.

Unlike industrial and open-source implementations of the standard, like x265, the HM extensively evaluates all coding modes through a rate-distortion optimization (RDO) [73]. The frame is first partitioned in CTU. Then each CTU is processed independently, and all coding modes, including split decisions and intra/inter predictions, are compared regarding an RD cost based on a Lagrangian function:

$$J = D + \lambda R \tag{2.7}$$

Where D represents the distortion and R the rate. Finally, the mode combination which minimizes that loss is selected for the given CTU.

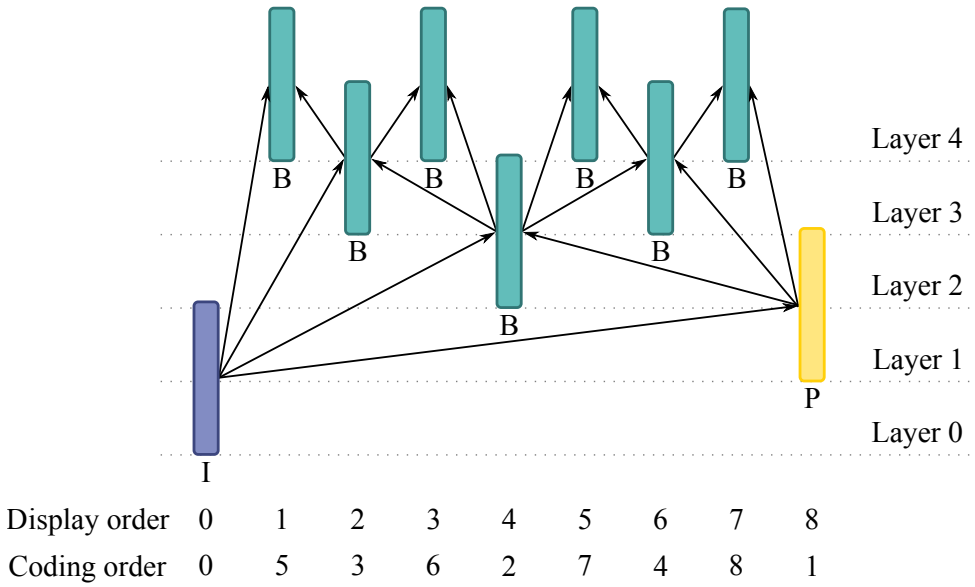


Figure 2.11 – Illustration of a RA coding configuration with GOP size of 8.

2.4.3 VVC standard

In answer to the growing interest in new immersive video services, such as 8K, HDR, and 360° videos, MPEG and ITU-T decided to develop a new video coding standard to succeed HEVC. This standard, called VVC, was published by JVET in July 2020 as VVC/H.266. The objective was to reach significant coding gains (around 40% in PSNR) compared to HEVC.

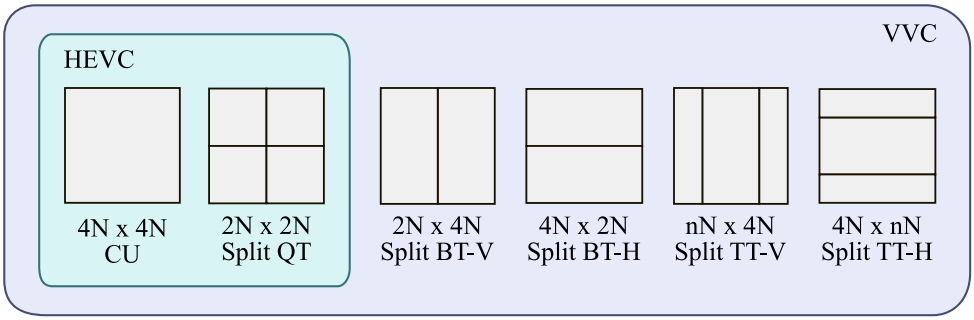


Figure 2.12 – Comparison of the available partitioning options in HEVC and VVC.

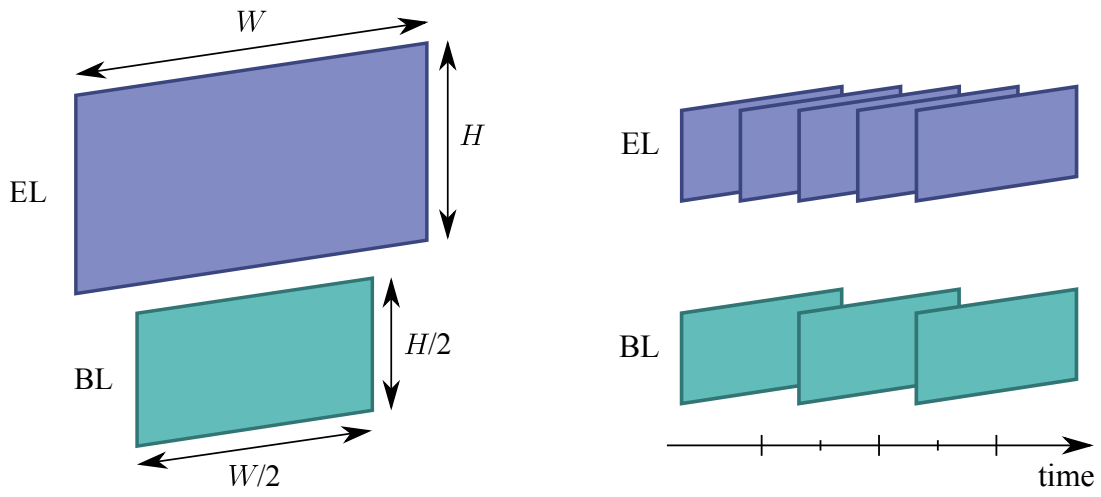
VVC follows the same hybrid coding scheme as previous AVC and HEVC standards. This standard includes several novel and refined coding tools at different levels of the coding chain. For instance, new split possibilities have been added to increase the partitioning flexibility. In addition, the maximum size of a CTU has been set to 128x128 to reduce the signal cost for higher resolution videos. An illustration of the different splits available in HEVC and VVC

is given in Figure 2.12. VVC provides a multiple transform selection MTS algorithm for the transform step, selecting the best transform among three different transforms: DCT-II, DCT-VIII, and DST-VII. In terms of intra prediction, VVC can select among 65 different modes, whereas HEVC proposed only 33 different modes as described in Section 2.4.2. Regarding the inter prediction, improvement of existing HEVC tools has been provided with SbTMVP and Bi-Directional Optical Flow. Also, some changes have been provided in entropy coding with a new advanced CABAC system. Moreover, a new ILF has been integrated with adaptative loop filtering (ALF) [74]. Finally, reference picture resampling (RPR) has been added as a new feature in VVC, allowing adaptive resolution coding to be performed independently from reference pictures.

A recent subjective test has been conducted using the VVC test model (VTM) and validated that VVC offers around 40% of bitrate reduction over HEVC for the same perceived quality targeting 4K and HD contents [59]. However, all those new coding decision possibilities introduce complexity in both the encoding and decoding processes. A comparative evaluation demonstrated that VVC is $34\times$ more complex than HEVC for AI coding configuration and around $9\times$ more complex for RA coding configuration [75]. On the decoder side, the complexity overhead of VVC is $1.8\times$ compared to HEVC.

2.4.4 Layered video coding

Delivering the same program in different formats is a current practice to target different types of receivers. The most straightforward way is to perform simulcast, where all the signal representations are encoded independently by single-layer codecs. In that case, the correlation between the different signal versions is not exploited, resulting in a bitrate overhead. The purpose of layered video coding is to deliver multiple layers consisting of a BL containing the most meaningful information and one or more ELs. It allows flexibility regarding the network's capabilities and simplifies the deployment of new services on top of legacy receivers. For instance, configurations based on different spatial resolution, framerate, bit-depth, color gamut, dynamic range, quality, or codec can be transmitted as separate layers. An example of spatial and temporal layered configurations is given in Figure 2.13. The following focuses on the former configuration.



(a) Spatial (b) Temporal

Figure 2.13 – Examples of layered configurations.

Scalable video coding

Scalability has been introduced in several video standards, namely, MPEG-3, H.263, MPEG-4, AVC. For instance, SVC has been developed as the scalable extension of MPEG-4 Part 10. However, due to late integration and a significant implementation shift between the scalable extension and the single-layer version, it failed to be adopted in the industry.

To avoid these issues, SHVC [9], published in October 2014 by the JCT-VC, focused on simple implementation and integration. The main objective of this standard is to minimize these compatibility issues by offering scalability with any HEVC single-layer core with an ILP module and minor syntax changes. Thus, all layers are based on HEVC using only additional HLS syntax, whereas SVC requires block-level syntax modifications. The HLS syntax is dedicated to high-level layer parameters, like the layer ID, the layer dependence, and the enabled interlayer coding tools. The ILP generates inter-layer reference (ILR) pictures that higher enhancement layers can use as a reference. Unlike single-layer codecs, which can only predict temporal and spatial, ILP exploits the redundancy in the reference layers to predict the EL. Thus, ILR pictures indices are included in the decoded picture buffer of the EL with other temporal reference pictures during EL prediction. An overview of an SHVC encoder for spatial scalability of 4K and 8K is given in Figure 2.14.

Two interlayer processing tools are included for spatial scalability in SHVC, namely inter-layer motion vector scaling and inter-layer texture resampling. The former tool allows adjusting the scale of the lower-resolution layers to higher-resolution layers. Thus, using dedicated

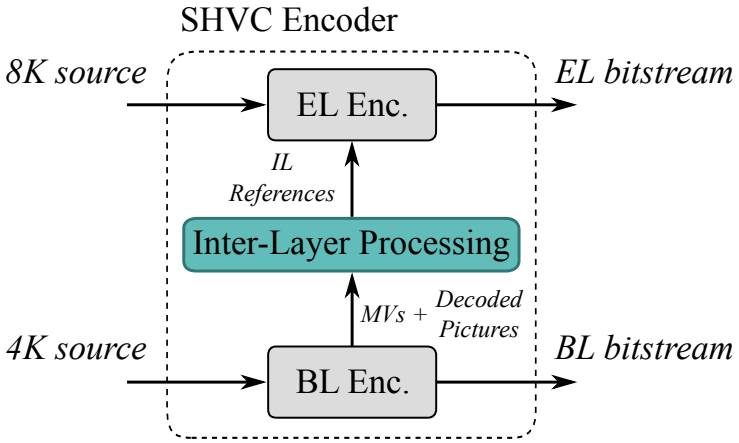


Figure 2.14 – ILP for spatial scalability in SHVC.

upscaling filters, ILR pictures are produced by rescaling reference layer (RL) pictures. The most common scaling ratios in SHVC are 1.5 and 2. In addition to the spatial redundancies between layers, SHVC allows motion vector mapping between layers. Thus, ILR motion parameters can be used in the temporal motion vector predictor (TMVP). In the case of spatial scalability, as the layers have different spatial resolutions, motion field mapping (MFM) is applied to perform the ILR inter-layer motion vector scaling. SHVC offers gains from 50% to 60% compared to SVC, depending on the type of scalability

MPEG-5 Part 2 LCEVC

Generally, scalability stands for codecs where each EL uses the same architecture (or similar) as the one performed by the single-layer BL. Thus, the encoder of each layer takes video signal as input, which allows the standard coding tools to be used for each layer. Although simplicity efforts have been made for SHVC, the complexity of inter-layer processing and the relatively low gain offered compared to simulcast prevented scalable technologies from reaching their expected success.

In 2021, a new codec called LCEVC was published in ISO/IEC 23094-2 [8] as MPEG-5 Part 2 [76], motivated by the drawbacks of traditional scalable approaches, e.g. SHVC or SVC. Instead of processing video signals for the enhancement layers, LCEVC encodes the residual information between the source and the subsequent layers. In addition to bringing more flexibility and reducing the complexity, it allows the enhancement layers to be compliant with any BL codec. However, the sparse characteristic of this residual makes it not suited to tools included in common video coding standards, such as the HEVC block-based scheme. Thus, a set of

dedicated coding tools are provided by LCEVC, e.g. small block sizes (4×4 and 2×2) and specific transform, to efficiently encode the residual signal.

An overview of the LCEVC standard architecture is given in Figure 2.15. The BL codec processes a downscaled representation of the input video, either by a factor of 2 or 4. Thus, it reduces the bit per pixel ratio and complexity, which tackles the increasing coding and decoding run times of emergent standards, e.g. VVC. In addition, post-processing tools like deblocking filters and dithering are provided to further enhance the quality of the reconstructed high-resolution (HR) signal.

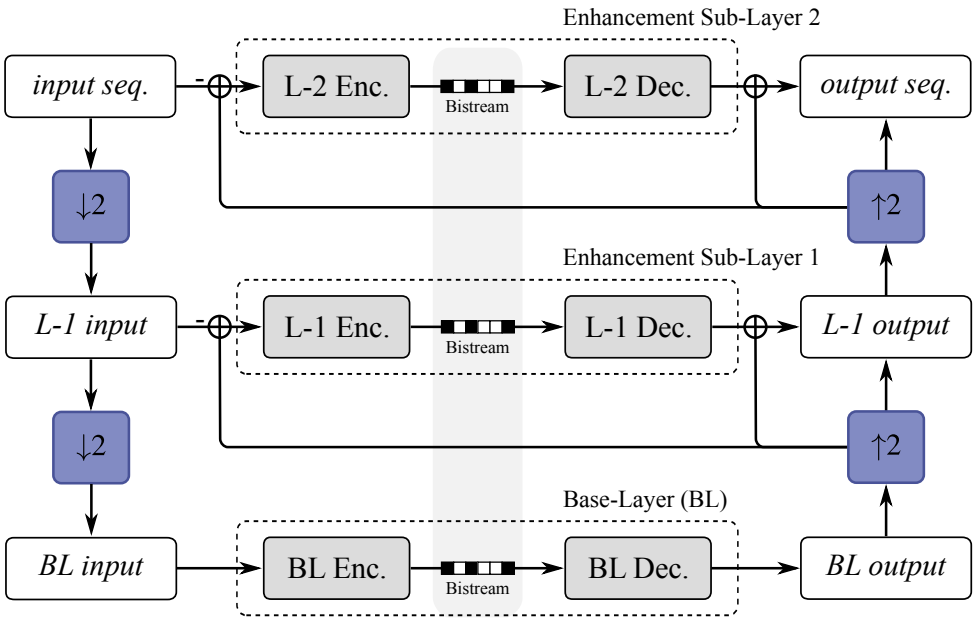


Figure 2.15 – Architecture of LCEVC.

First, the input video sequence is downscaled to the desired resolution using non-normative downscaling filters. Both downscaling steps can be performed in either both vertical and horizontal directions, horizontal direction only, or bypassed. Then, the BL codec encodes the downscaled video to produce the base-layer bitstream. The first enhancement sub-layer encodes the differential residual between the reconstructed BL and the first-order downscaled input sequence. The output of the enhancement sub-layer 1 is then reconstructed to feed the input of the enhancement sub-layer 2. Similarly, this EL encodes the residual between the reconstructed EL-1 and the input video sequence.

As mentioned, the sparsity of residual makes it not suited to traditional coding tools implemented in most MPEG standards. Indeed, as the residual contains a high proportion of close-to-zero values, processing large blocks followed by a DCT would result in an important

loss of information during quantization. Thus, the enhancement layers process small blocks of size 2x2 or 4x4, i.e., CU. It allows efficiently transform edges and textures in the residual while proving a high flexibility. Once the residual is computed, residual mode selection is applied to determine which residual blocks should be encoded and transmitted. First, each block is classified over ten classes regarding its spatial and temporal features. Then, a weight in the range of 0 to 1 is associated with each CU. CUs (or group of CUs) assigned with a 0 weight associate are not transmitted. Depending on their class, the non-zero CUs are processed by either the EL-1 or the EL-2. In addition to the residual mode selection, a vector-free temporal prediction can be performed on the reconstructed signal. Indeed, as residual signals contain a lower temporal correlation than video signals, motion compensation is not adapted. This simple temporal layer allows fixed and sharps elements, like logos, to be propagated along with the GOP. This processing is applied at the CU level, enabling parallel processing.

Each selected CU is then transformed using a Hadamard filter. This transform has the advantage of containing orthogonal rows and being self-inverse, meaning that the inverse transform is the same as the forward transform. Quantization and entropy coding are then applied to both ELs. A linear quantizer with a dedicated quantification parameter for both ELs is used for quantization. In LCEVC, entropy coding is based on Run-Length Encoder and Prefix Coding encoder as implemented in AVC.

An overview of the performance of LCEVC is given in [77]. This study demonstrates that LCEVC brings gains compared to full-resolution coding, especially at a low bitrate. However, the performances in terms of PSNR are slightly better than the upscaling filter without metadata. Regarding VMAF, the gain is lower than the upscaling filter used without metadata. It can be explained by the weights used for VMAF that are not trained for the tested type of degradation. In contrast, high gains have been observed for logos and fixed elements sequences. The temporal layer which successfully propagates these residuals along the GOP can explain this observation. Also, the more recent the codec, the less the gains are compared to full-resolution coding, the less the gains are compared to full-resolution coding. In the industry, LCEVC has been selected as a layered approach in the Brazilian television system (SBTVD) [78] to enable 8K over the top (OTT) signal delivery using over a 4K over the air (OTA) BL. However, it is not planned to be introduced in the DVB toolbox yet.

2.5 Conclusion

Given the amount of information contained in new immersive video signals, highly efficient compression technologies are needed to compress and transmit this type of video. This section presented different video formats, focusing on spatial resolution, especially 8K. Although modern coding technologies offer coding gains, it is still challenging to consider delivering these new formats on broadcast infrastructures.

The following chapters will present different AI-based algorithms for video coding. We demonstrate how they can be integrated into a compression pipeline and facilitate deploying these new video services on broadcast infrastructures.

CHAPTER 3

AI-BASED ALGORITHMS FOR IMAGE AND VIDEO COMPRESSION

3.1 Preamble

Recently, deep neural networks (DNNs) have been widely explored in most scientific research fields, including image and video processing. The idea behind neural networks is to optimize a set of trainable parameters with respect to a loss function. Thus, a non-linear mapping between the input and the desired output, i.e., the ground truth, is learned. The ability of DNNs to model complex non-linear functions makes them suited to solve ill-posed problems like image restoration tasks. This chapter demonstrates that every block of a video compression pipeline, ranging from pre and post-processing models to the coding algorithm itself, can be replaced by DNNs to further improve the performance.

This chapter is organized as follows. Section 3.2 first gives a brief overview of neural networks. Then, Section 3.3 discusses how AI-based restoration algorithms can be integrated as pre and post-processing into a coding pipeline. Finally, Section 3.4 reviews the recent promising approaches proposed to replace traditional image and video codecs.

3.2 Neural network overview

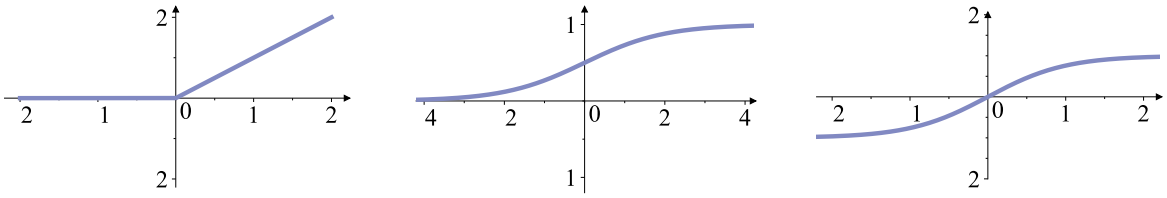
Any ML algorithm can be represented as a function f_θ with parameters θ, which learns to map inputs $\boldsymbol{x}$ to outputs $\boldsymbol{y}$. First, a ground truth containing representative inputs $\boldsymbol{x}$ and associated labels $\boldsymbol{y}$ is designed. Then, the ML model tunes its parameters θ during the training phase to solve $\boldsymbol{y} = f_\theta(\boldsymbol{x})$. Generally, the dataset is divided into training and testing sets to ensure that the model generalizes well on unknown data. This section gives an overview of the building and training phases of a neural network.

3.2.1 Basics of neural networks

Artificial neural networks (ANNs), or neural networks (NNs), are machine learning systems deriving from the biological neural network that constitutes the human brain. These networks are composed of multiple neurons connected by synapses that transmit a signal, i.e., a real number, through the network. A neural network f_θ of parameters θ is composed of weights W and bias B stacked into layers f_{θ_i}. These linear layer are generally followed by a non-linear activation function. For instance, the output of a layer f_{θ_i} followed by a rectified linear unit (ReLU) activation is computed as:

$$f_{\theta_i}(\boldsymbol{y_i}) = max(0, W_i * \boldsymbol{y_i} + B_i), \tag{3.1}$$

with $i \in 1, ..., L$ the index of the layer and L the number of layers. Other examples of non-linear activation functions are given in Figure 3.1.



(a) ReLU (b) Sigmoid (c) Tanh

Figure 3.1 – Non-linear activation functions.

The training process is divided into two steps: the forward-pass and the backward-pass. During the forward-pass, the network's output $\hat{\boldsymbol{y}}$ is predicted from the input $\boldsymbol{x}$ by successively passing through each layer of the network f_θ as:

$$\hat{\boldsymbol{y}} = f_{\theta_L} \circ f_{\theta_L - 1} \circ ... \circ f_{\theta_0}(\boldsymbol{x}). \tag{3.2}$$

Then the predicted output $\hat{\boldsymbol{y}}$ is compared with the associated ground truth label $\boldsymbol{y}$ using a loss function $\mathcal{L}$.

During the backward-pass, the network parameters θ are learned using optimization techniques, such as stochastic gradient descent (SGD) [79] or ADAM [80]. These methods minimize the loss function $\mathcal{L}$ by backpropagating the gradients from the deeper to the shallower layers of the network f_θ. At each training iteration, a batch of sample is extracted from the dataset and is used to optimize the network's parameters θ as follows:

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}(\boldsymbol{y}, \hat{\boldsymbol{y}}), \tag{3.3}$$

with $\hat{\theta}$ denoting the updated parameters, N the batch-size and $n = 1, ..., N$ the sample index.

DNNs have been investigated by stacking more layers to increase the number of learned parameters. Outstanding performance has been recently observed and demonstrated that DNNs can learn rich representations of the input $\boldsymbol{y}$. These models have been widely used for various tasks, from classification to complex restoration tasks such as image denoising and super-resolution (SR).

3.2.2 Convolutional neural networks

CNNs are a specific type of neural network where the weights W are represented by convolutional kernels. This architecture allows large-scale images to be processed with fewer operations than fully connected (FC) neural networks. Generally, the training images are cropped into patches to form fixed-size batches and accelerate the training process. The features learned by a CNN are called features maps and correspond to sub-images with increasing abstraction. CNNs are characterized by a receptive field, representing the extent of the scope of input data to which a neuron or unit within a layer can be exposed. This principle is illustrated in Figure 3.2. Several techniques allow increasing the receptive field, including using a deeper network, larger convolutional kernels or downscaling and upscaling layers.

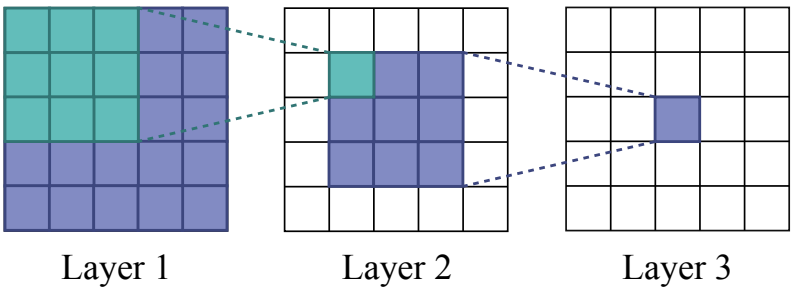


Figure 3.2 – Illustration of the receptive field of a three-layers (3×3 kernels) CNN.

Advanced architectures have been investigated to improve the performance of CNNs. Intuitively, increasing the number of layers would result in better performance as more parameters are provided to learn the objective function. However, the non-linear activations can lead to gradient vanishing during backpropagation, which degrades the performance. To tackle this, residual network (ResNet) [81] proposes adding skip connection represented by element-wise sum between layers. Thus, the network can learn the identity function to bypass some layers

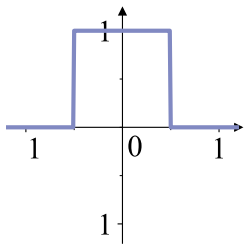
during the backward step and it also makes the feature maps more sparse and leads to better performance. More recently, attention layers have been investigated for CNNs [82]. Inspired by the human brain's attention, these layers learn masks that indicate areas of interest to the network regarding the task.

3.3 Pre and post-processing methods

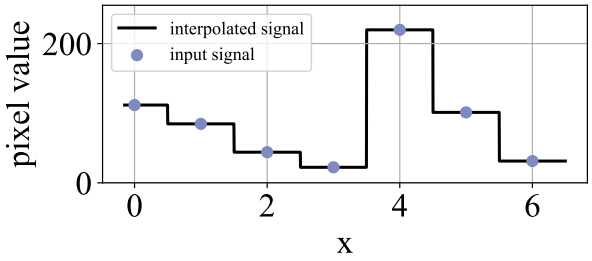
The latest video coding standard, called VVC, includes several novel and refined coding tools at different levels of the coding chain. These tools bring significant coding gains with respect to the previous standard, HEVC. However, the encoder may still introduce visible coding artifacts, mainly caused by coding decisions applied to adjust the bitrate to the available bandwidth. Thus, CNN-based quality enhancement (QE) models have been investigated as post-post processing to improve the decoded signal quality. Furthermore, downscaling-based compression methods have been explored to increase the image's bit-per-pixel ratio, which mitigates coding artifacts at the cost of a high-frequency loss. It also appears as an efficient solution for backward compatibility with legacy receivers by allowing higher resolutions to be reconstructed without additional bitstream. This approach requires efficient pre and post- processing modules to recover details from low-resolution (LR) compressed images. This section reviews different state-of-the-art SR and QE approaches and presents how they are used for downscaling-based video compression.

3.3.1 Super-resolution

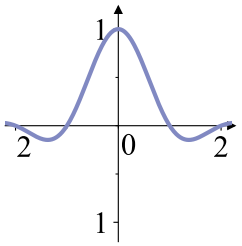
A simple way to upsample (or upscale) an image is by using linear interpolation filters. These filters consider neighboring pixels and weigh them linearly regarding the location of the pixel to generate. The most straightforward approach is the nearest neighbor kernel, which fills the missing pixels with their closest values in the original image. However, this filter generates undesirable artifacts in the reconstructed HR image, which can be solved by more sophisticated interpolation methods, e.g. bilinear, bicubic [83], or Lanczos [14]. The interpolation of a 1-D signal is illustrated in Figure 3.3 for different filters. Although low complex, these filters fail to reconstruct high-frequencies and model the signal's discontinuities, e.g. edges, resulting in blurry images. In addition to increasing images' spatial dimension, SR algorithms address all unpleasant effects of linear interpolators, including edge-smoothing, blur, and noise. These algorithms can be divided into two distinct groups: single image super-resolution (SISR) and multi-frame super-resolution (MFSR).



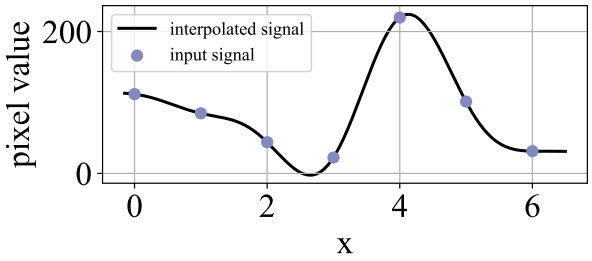
(a) 1-D nearest neighbor filter



(b) 1-D nearest neighbor signal interpolation



(c) 1-D Lanczos-3 filter



(d) 1-D Lanczos-3 signal interpolation

Figure 3.3 – Non-linear activation functions.

SISR methods aim at reconstructing an HR image from a single LR observation. This problem is considered ill-posed, as an infinity of HR images can solve a given LR image I^{LR}. Learning-based SISR methods have recently shown outstanding performance to reconstruct HR images. Dictionary-based approaches relying on neighbor embedding [84, 85] and sparse coding [86] techniques have been first explored. In 2014, the first CNN-based SR model, called SRCNN, was proposed [87]. This simple CNN, composed of three convolutional layers, allows learning the non-linear mapping between LR and HR feature maps. An overview of this architecture is given in Figure 3.4. First, the network $f_{\theta_{SR}}$ estimates the HR output image $\hat{I}^{HR}$ from the LR input image I^{LR} as:

$$\hat{I}^{HR} = f_{\theta_{SR}}(I^{LR}), \tag{3.4}$$

Then, the learned parameters $\hat{\theta}_{SR}$ are obtained by solving the following optimization problem:

$$\hat{\theta}_{SR} = \arg\min_{\theta_{SR}} \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}(I^{HR}, \hat{I}^{HR}), \tag{3.5}$$

with N the number of training samples, $n = 1, ..., N$ the sample index, and $\mathcal{L}$ the loss

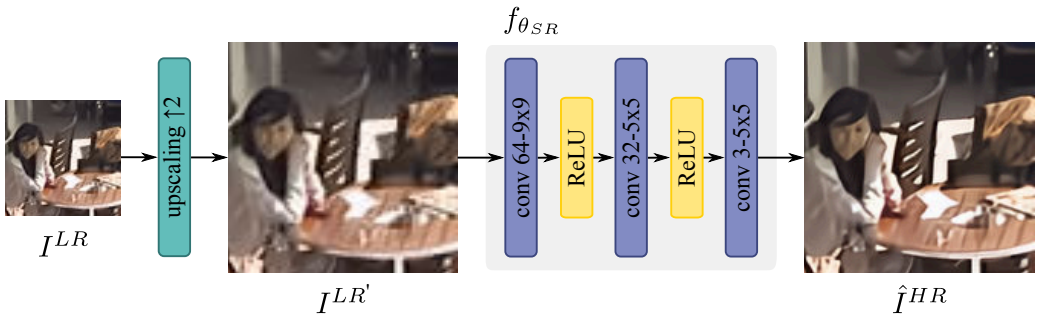


Figure 3.4 – SRCNN Architecture [10].

function computed as:

$$\mathcal{L}(I^{HR}, \hat{I}^{HR}) = || \, I^{HR} - \hat{I}^{HR} \, ||^2 \, . \tag{3.6}$$

This approach performs a bicubic upscale on the LR input image I^{LR} before processing to match with the output resolution. However, achieving the upscaling phase at the end of the network reduces the complexity as convolutions are performed in the LR space. A sub-pixel upscaling layer has been developed in [88] to prevent checkerboard artifacts generated by convolution layers [89]. ResNets have later been investigated for SR with long [90], short [91], and dense connections [92] to further improve the performance. EDSR proposes removing the batch normalization layers that impose less flexible ranges for feature maps [18]. The authors also suggest replacing the MSE with the L_1-loss, which improves the PSNR performance. An overview of this architecture is given in Figure 3.5. Examples of reconstructed images are given in Figure 3.6.

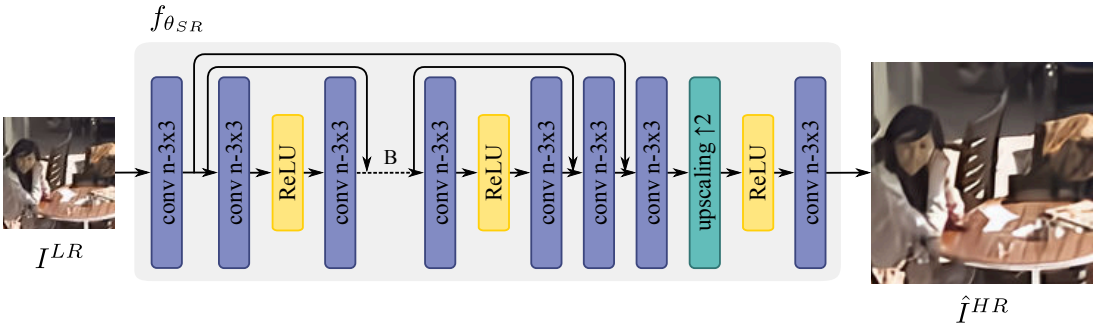


Figure 3.5 – EDSR Architecture [18].

Deep-back projection layers have been introduced into an SR network [93, 94]. This principle performs successive downscaling and upscaling on feature maps to increase the receptive field. The attention mechanism described in Section 3.2 has been investigated for SR based on first-

Figure 3.6 – Visualization of images from $kodak$ reconstructed using a Lanczos filter and EDSR.

order feature statistics with RCAN [95] and second-order feature statistics with SAN [96]. Attention masks are learned in both pixel and channel spaces in these approaches.

These presented learning-based methods use pixel-wise loss functions, e.g. MSE and L_1-loss, during the model optimization. Although these metrics are simple to compute and directly related to the PSNR, their fidelity with the HVS is limited. The MS-SSIM loss has been investigated in [97] and coupled with the L_1-loss to improve reconstruction quality. Moreover, perceptual metrics [98] and generative adversarial networks (GANs) [99] have been explored as loss functions for SISR [91]. These approaches allow more realistic HR images to be produced by hallucinating spatial textures. Some improvements have been proposed in [100], including residual-in-residual dense blocks without batch normalization. However, GAN-based approaches can cause deviation from the source content, which is undesirable in a broadcast context.

MFSR methods leverage temporal information to recover missing subpixels in the adjacent frames. Kappeler *et al.* proposed a CNN-based MFSR solution called VSRnet [101]. Here, up to two previous frames are aligned with the current one as input to the network f_θ using adaptive motion compensation with optical flow [102] as:

$$\hat{I}^{HR} = f_{\theta_{SR}}(I^{LR}_{t-k}, ..., I^{LR}_{t}, ..., I^{LR}_{t+k}), \tag{3.7}$$

Table 3.1 – State-of-the-art super-resolution models overview.

SISR model	#Parameters	PSNR (dB)	SSIM
bicubic	-	33.66	0.9299
SRCNN [10]	57K	36.66	0.9542
VDSR [90]	665K	37.53	0.9590
EDSR [18]	43M	38.20	0.9606
RCAN [95]	16M	38.33	0.9617

with $f_{\theta_{SR}}$ the MFSR model, t the temporal index and k the number of neighboring frames considered is the reconstruction process. The sub-pixel upscaling has been introduced in vESPCN [103], the spatiotemporal extension of ESPCN [88]. A recurrent neural network (RNN) dedicated to MFSR has been developed where previously inferred HR frames are used to reconstruct the current frame [104]. This solution mitigates temporal inconsistency and reduces complexity. Wang *et al.* proposed an additional neural network to rescale the optical flow maps in HR space for motion compensation [105], increasing the accuracy of final HR reconstruction and run-time. These methods rely on external architectures to compute the optical flow maps, which is suboptimal. Jointly training the optical flow model with the SR network has been first proposed in [106], where two terms are optimized in the loss function: one for the HR reconstruction error and one for the motion estimation error. Unlike in multitask learning appraoches, both models have distinct parameters. In [107], the authors proposed jointly training the overall system by optimizing the SR task only. Thus, the reconstructed motion maps look degraded for the generic motion estimation task but provide more powerful representations for the SR task. Instead of implicitly learning an optical flow for each neighboring frame, authors in [108] proposed RBPN, a recurrent approach that performs motion compensation in the feature space. Thus, features are extracted from the neighboring frames and introduced at different stages into the network by concatenation using a recurrent back-projection mechanism.

Although MFSR approaches improve temporal consistency and global performance, the complexity is highly increased due to complex additional modules performing motion estimation and compensation steps.

3.3.2 Quality enhancement

ILFs are widely used in hybrid coding architectures to improve the quality of reference images propagated into the GOP. For instance, a deblocking filter [109] and a SAO [110] can

be activated into the coding loop of HEVC to reduce blocking and ringing artifacts in decoded frames. More recently, ALF [74] has been introduced in VVC and provides significant gains in the global performance of this standard [111]. Although CNNs have been investigated to enhance the quality of codecs directly in the coding loop [112, 113, 113, 114, 115], the main issue resides in training them during the RDO process because of their high complexity. Moreover, their integration requires tuning of the host codec.

Unlike ILF, post-processing QE methods aim at reducing compression artifacts outside the coding loop without any modification in the host encoder/decoder. CNN-based models have been investigated to match degraded and uncompressed images similarly to SR approaches described in Section 2. Authors of SRCNN [87] proposed AR-CNN, a CNN-based architecture without upscaling for coding artifact removal in JPEG images [19]. An overview of this architecture is given in Figure 3.7. With $f_{\theta_{QE}}$ denoting the QE neural network, the output image $\hat{I}$ is first estimated from the compressed degraded input image $\tilde{I}$ as:

$$\hat{I} = f_{\theta_{QE}}(\tilde{I}). \tag{3.8}$$

Then, the parameters $\hat{\theta}_{QE}$ are obtained by solving the following optimization problem:

$$\hat{\theta}_{QE} = \arg\min_{\theta_{QE}} \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}(I, \hat{I}), \tag{3.9}$$

with N the number of training samples, $n = 1, ..., N$ the sample index, and $\mathcal{L}$ the loss function computed as:

$$\mathcal{L}(I, \hat{I}) = || \ I - \hat{I} \ ||^2 \ . \tag{3.10}$$

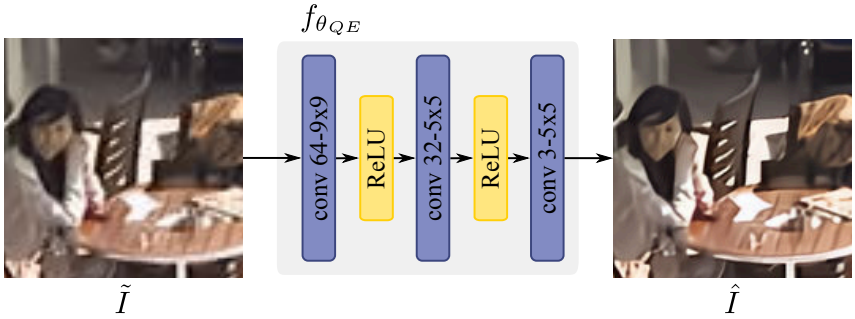


Figure 3.7 – AR-CNN Architecture [19].

This approach has been extended in real-time using downscaling and upscaling layers to

reduce the computational complexity [19]. Based on this principle, Li *et al.* proposed a ResNet architecture [116] to enhance the quality of HEVC intra-coded frames. Authors in [117] proposed an early exit architecture where the reconstruction is performed in multiple stages, allowing to stop enhancing if the processing resources are insufficient. Some approaches recently proposed enhancing VVC intra coded frames using a very deep residual network [118] or a multi-scale grouped dense network [12]. Inter-coded frame post-processing has been performed by authors in [119] and [120] by jointly training one model for I frames and one for P and B frames. A recurrent architecture has been proposed in [121] to process successive frames recursively.

Image and video analysis are performed during encoding to perform the RDO. Thus, the bitstream contains rich information about the signal that can guide the QE process. For instance, Lam *et al.* use the QP as a qp_{map} in addition to the degraded image as input of the network [122]. This map is expressed as follows:

$$qp_{map}(i,j) = \frac{QP}{QP_{max}}, \quad i = 1, ..., W; \quad j = 1, ..., H, \tag{3.11}$$

with (i, j) are the vertical and horizontal pixel coordinates and $QP_{max} = 63$ in VVC. In [123], the authors proposed combining the coded prediction residual with the prediction frame before passing through the network. The CTU partitioning has been used as prior information for HEVC post-processing in [124], providing knowledge about the spatial structure of the image. In extension to this idea, the mode selected for each CTU [125] and the predicted signal [126] further increased the reconstruction quality.

Similarly to MFSR, adjacent frames can help recover missing information in the current degraded image. Inspired by VSRnet [101], a multi-frame QE as a post-processing approach has been proposed for HEVC [127] and VVC [128]. Architectures based on RNN [129, 121] and long short-term memory (LSTM) [130] have been proposed to restore decoded frames recursively. Yang *et al.* [131, 132] observe that adjacent frames are sometimes more degraded than the current frame due to the hierarchical structure of video codecs. Thus, they proposed to use the peak-quality frames (PQFs) only in the current frame QE process. These approaches use a SVM [131] or a neural network [132] to detect the PQFs, which introduces additional latency.

The above approaches efficiently improve performance by reducing compression artifacts in decoded pictures without any modifications in the host encoder. Moreover, these works can be integrated into downscaling-based compression systems, where the LR images are degraded.

3.3.3 Downscaling-based compression

Downscaling-based compression systems downscale the signal before encoding and rescale it to the original resolution after decoding. It allows increasing the bit-per-pixel ratio at the cost of high-frequency loss, which is valuable at low bitrate, as illustrated in Figure 3.8. These approaches are divided into two distinct groups: downscaling as mode selection and downscaling as pre-processing.

The former has been first investigated by providing additional tools for processed 16×16 macroblocks of MPEG-2 [133] and AVC [134] standards. Several approaches have explored new downscaling modes for HEVC using CNNs for CTUs upscaling [135] or frame upscaling [136, 137]. Although improving the RD performance, these approaches require tuning the decoder, thus renouncing to syntax confirmation with video coding standards. RPR has been recently integrated as a new feature in the latest MPEG video coding standard, VVC [111]. This feature improves the flexibility of rate control algorithms and allows adaptive resolution coding by enabling the ability to adaptively change the coded picture resolution. A recent work improves VVC's RPR performance with some modifications, such as increasing the number of references and using HR references instead of LR ones [138].

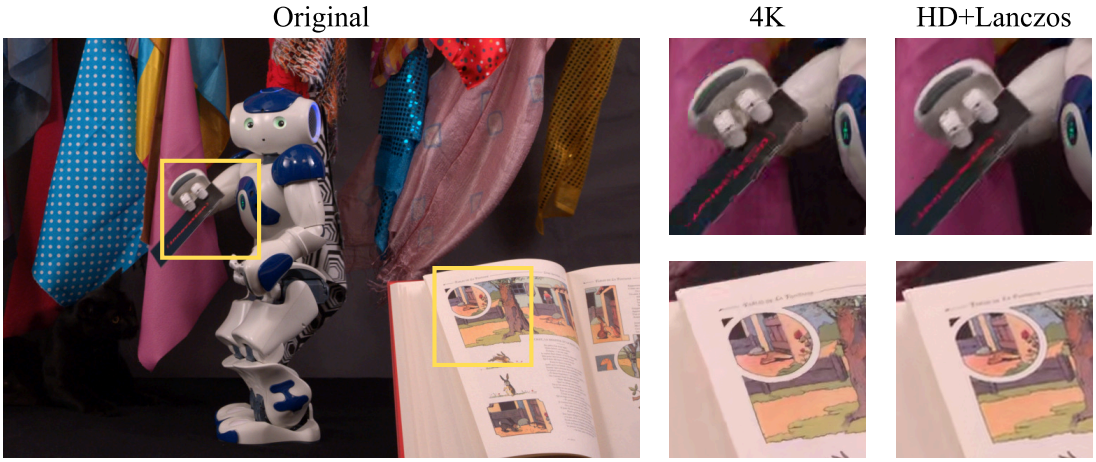


Figure 3.8 – Visual comparison of downscaling-based coding when downscale is activated and not for the sequence *CatRobot1* (5Mbps).

Downscaling as pre-processing method considers the entire downscaling of frames before encoding. An overview of this compression method is given in Figure 3.9. This principle has been first introduced for still image coding with joint photographic expert group (JPEG) [139]. The authors demonstrate that, given a bit-budget, full-resolution coding can be outperformed

by applying the optimal scale factor to the input image before encoding. Zhang *et al.* [140] proposed to apply it to JPEG2000 using custom downscaling and upscaling filters. This principle has been later explored for video coding with the well-known bitrate ladder submitted by Netflix [141], where the optimal resolution is selected regarding the available bandwidth. Some approaches suggested separating coding and scaling degradation to find the optimal scale ratio regarding an RD-cost [142, 143]. This method assumes that both degradations are independent. However, these methods require precoding, resulting in high complexity and latency, which is not adapted for live applications. In [144], the authors proposed downscaling inter-frame only and performed dictionary-based SR to reconstruct the HR signal. Afonso et al. determine the relationship between the downsampling quality and the QP used for compression with a resolution quantization optimization (RQO) module [145]. A solution for dynamic adaptation of both spatial resolution and bit-depth has been investigated in [146] and [6]. In this approach, a simple neural network is employed as an RQO module to determine if the current frame has to be downscaled or not based on spatial features. Some approaches proposed using GANs and perceptual losses to hallucinate the details lost during scaling and quantization [147, 148]. These approaches produce better-looking images but are challenging to train and produce unpleasant artifacts.

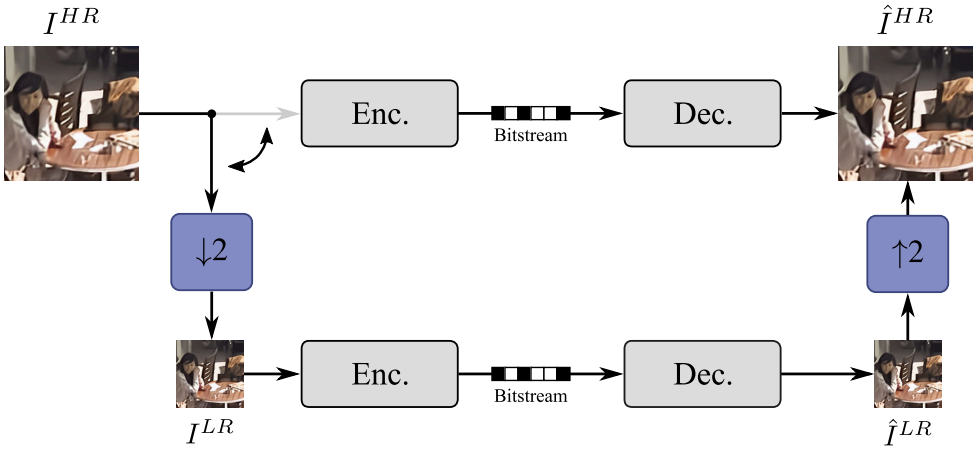


Figure 3.9 – Description of downscaling as pre-processing pipeline.

Finally, learned downscaling methods, also called compact-resolution (CR), have been investigated to improve overall performance. In [149], two CNNs: one for CR and one for SR, are jointly optimized regarding the loss function. The authors demonstrated that the proposed solution improves the performance compared to classical downscaling-based approaches, even using a Lanczos filter as an upscaling method. Li *et al.* developed a dynamic framework that

selects the appropriate downscaling method, i.e., traditional or CNN-based, depending on the content [150]. In [151], authors trained a CNN-based CR model to maintain the source's fidelity in the downscaled signal without additional terms in the loss function. In [152], a CR network is trained jointly with a semantic segmentation network to help recover textures at the receiver side. An end-to-end image codec has been jointly trained with CR and SR models [153]. Thus, the whole downscaling framework is optimized regarding the objective loss function. More details about learned compression models are given in Section 3.4.

Downscaling as post-processing also appears as an efficient solution for backward compatibility with legacy receivers. However, these approaches appear content-dependent, which must be carefully considered when deploying next-generation services, such as 8K.

3.4 End-to-end image and video coding

Lossy image and video coding aim at reducing the number of bits required to represent the signal while preserving the most important visual information. This objective can be interpreted as a RD optimization problem [154], where the model tries to minimize the distortion of the reconstructed image under a rate constraint. Traditional compression systems are composed of highly dependent modules that cannot be jointly optimized. In other words, improvement in a module does not necessarily lead to a better RD performance of the overall system. Moreover, they are generally based on a block partition scheme, resulting in blocking artifacts on the block boundaries of the reconstructed image. Recently, learned image and video compression techniques have shown outstanding performance by jointly optimizing the whole compression framework regarding a RD loss function. First, we develop the general idea behind learned compression and introduce the first end-to-end image compression system based on a variational autoencoder (VAE). Then, we present advanced entropy models and learned video coding systems. Finally, layered end-to-end coding approaches are presented, including hybrid approaches based on traditional codecs.

3.4.1 General principle

As mentioned in the previous chapter, most compression standards adopt the principle of transform-coding to compress the signal. This paradigm is based on Shannon's separation principle [155] and follows three distinct steps: transform, quantization, and lossless coding. Given an input image $\boldsymbol{x}$, the signal is first compacted and decorrelated by a spatial transform

into a representation $\boldsymbol{y}$. Then, the transformed signal $\boldsymbol{y}$ is reduced to a discrete set of symbols $\hat{\boldsymbol{y}}$ corresponding to discrete quantization levels. This quantized representation $\hat{\boldsymbol{y}}$ is then losslessly coded into a sequence of bits using entropy coding to reduce the remaining redundancies. Finally, the output signal $\hat{\boldsymbol{x}}$ is reconstructed from $\hat{\boldsymbol{y}}$ by the decoder. Based on this system, RD optimization techniques, such as the well-known Lagrangian optimization method [156], can be used to obtain specific operating points of the compression framework.

Autoencoders are a type of CNN based on an encoder-decoder architecture. First, the analysis part of the encoder denoted as g_a generates compact and meaningful representations $\boldsymbol{y}$ of the input image $\boldsymbol{x}$, called latent variables:

$$\boldsymbol{y} = g_a(\boldsymbol{x}). \tag{3.12}$$

Then, the synthesis part of the decoder denoted as g_s, uses this latent representation to reconstruct the output image $\hat{\boldsymbol{x}}$ as:

$$\hat{\boldsymbol{x}} = g_s(\boldsymbol{y}). \tag{3.13}$$

Unlike traditional compression algorithms, autoencoders are fully differentiable systems. Thus, the overall model's components can be optimized together regarding the objective loss function.

In 2017, Ballé *et al.* proposed the first learned compression model based on a VAE [7]. This compression framework, illustrated in Figure 3.10, follows the transform-coding principle previously discussed by applying quantization and entropy coding to non-linearly transformed representations $\boldsymbol{y}$. However, one limitation of such a system relies on the non-differentiability of the quantization function. Thus, the authors propose relaxing the quantization function by applying a uniform noise $\mathcal{U}(-\frac{1}{2}, +\frac{1}{2})$ on latents to emulate the quantization errors during training while enabling backpropagation, resulting in $\tilde{\boldsymbol{y}}$. During inference, the latent variables $\boldsymbol{y}$ are quantized using the $round$ function to produce $\hat{\boldsymbol{y}}$. To simplify, we use $\bar{\boldsymbol{y}}$ and $\bar{\boldsymbol{z}}$ to denote both actual and emulated quantized latents.

Unlike traditionnal autoencoders, this VAE aims at jointly minimizing the distortion D and the rate R regarding a given RD trade-off. Thus, the RD cost can be directly optimized as a loss function $\mathcal{L}$ based on a Lagrangian multiplier λ:

$$\mathcal{L}(\lambda) = D + \lambda R. \tag{3.14}$$

The term D represents the distortion between the reconstructed image $\hat{\boldsymbol{x}}$ and the original

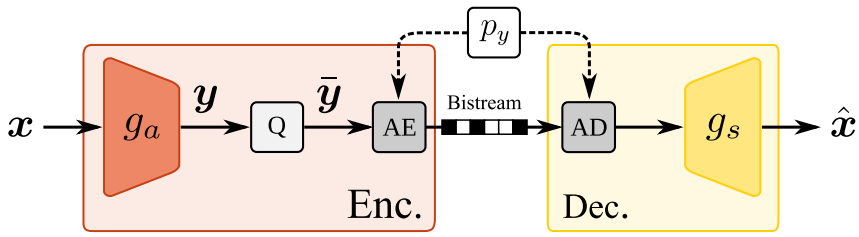


Figure 3.10 – Description of end-to-end compression using a VAE [7].

image $\boldsymbol{x}$. The value of D is computed using the MSE as:

$$D = \mathbb{E}_{\boldsymbol{x} \sim p_x}[|| \boldsymbol{x} - \hat{\boldsymbol{x}} ||^2]. \tag{3.15}$$

The term R corresponds to the rate of the quantized latents $\hat{\boldsymbol{y}}$, which is bounded by the signal's entropy. As the true probability m of latents $\hat{\boldsymbol{y}}$ is unknown, the rate is formulated as the cross-entropy of the distribution m and a selected probability model p_y as:

$$R = H(m, p_y) = \mathbb{E}_{\tilde{\boldsymbol{y}} \sim m}[-\log_2(p_y(\tilde{\boldsymbol{y}}))], \tag{3.16}$$

We can re-write this equation as follows:

$$R = H(m) + D_{KL}(m \,||\, p_y), \tag{3.17}$$

with D_{KL} denoting the Kullback-Leibler divergence assessing the mismatch between the actual distribution of latents m and the selected probability model p_y. Thus, minimizing the cross-entropy of $\hat{\boldsymbol{y}}$ regarding a given probability model p_y is equivalent to make m matches the distribution p_y while lowering its entropy $H(m)$.

In [7], the learned image compression proposed better performance than JPEG in terms of visual quality regarding the same bitrate. However, this framework used a relatively simple network based on generalized divisive normalization (GDN) activations and a fully-factorized probability model. An overview of this network's architecture is given in Figure 3.11. Recent works using more sophisticated entropy models are discussed in Section 3.4.2.

3.4.2 Advanced entropy models

Entropy estimation models play an essential role in the performance of the overall learned compression system. As discussed previously, a low practical rate relies on both a low entropy

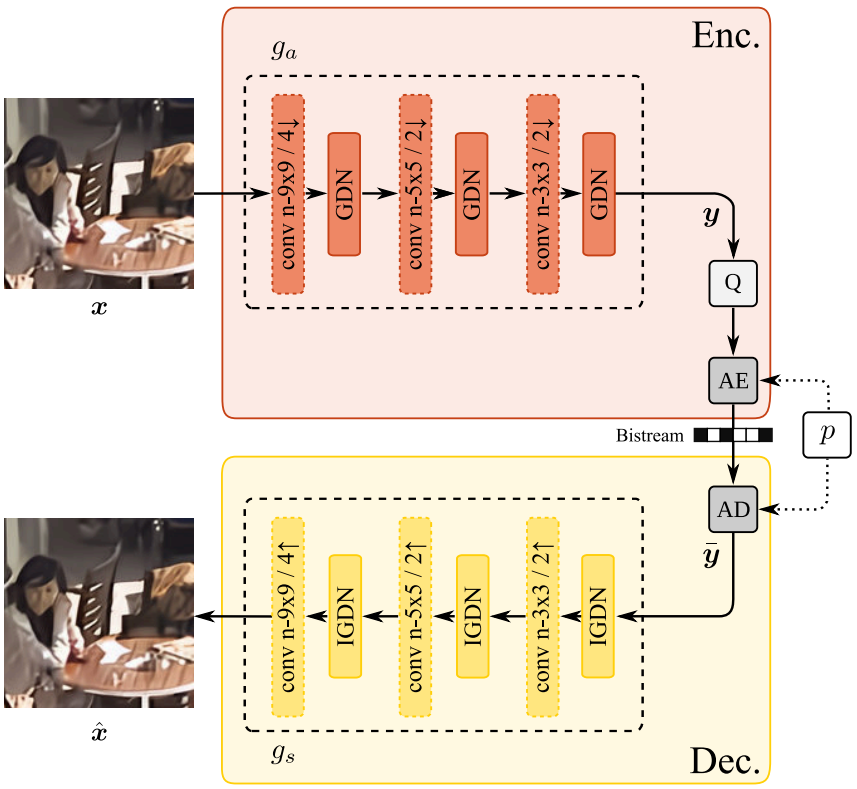


Figure 3.11 – Ballé *et al.* Architecture [7].

of latents regarding their actual distribution and a low mismatch with the entropy model p_y known at the decoder side. In the pioneering work of Ballé *et al.*, latents are assumed to be independent and a fully-factorized entropy model based on a Gaussian distribution is used in the cross-entropy rate loss function. However, this approach is suboptimal as a simple Gaussian model cannot efficiently represent all the image space. Moreover, this entropy model is fixed for all processed images, but it is clear that latents have different statistical properties regarding the spatial characteristics of the input image.

In extension to their work, the authors proposed the use of an hyperprior model to enhance the entropy estimation in the loss function [157]. This complementary neural network is trained jointly with the main encoder-decoder system to dynamically learn the appropriate parameters of the entropy model. This principle has been extended for prediction of the mean $\boldsymbol{\mu}$ and the scale $\boldsymbol{\sigma}^2$ of a parametric Gaussian entropy model [158, 20] defined by:

$$p_y(\bar{\boldsymbol{y}}|\bar{\boldsymbol{z}}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2). \tag{3.18}$$

First, the analysis part of the hyper-encoder denoted as h_a, generates additional latent variables $\boldsymbol{z}$ from the latents $\boldsymbol{y}$ as:

$$\boldsymbol{z} = h_a(\boldsymbol{y}). \tag{3.19}$$

This latent $\boldsymbol{z}$ is then quantized and entropy coded with respect to p_z using the same strategy as in Section 3.4.1 to produce $\bar{\boldsymbol{z}}$. Finally, the entropy model's parameters, e.g. the mean $\boldsymbol{\mu}$ and the scale $\boldsymbol{\sigma}^2$ are estimated from $\boldsymbol{z}$ by the synthesis part of the hyper-decoder h_s, as:

$$\{\boldsymbol{\mu}, \boldsymbol{\sigma}^2\} = h_s(\bar{\boldsymbol{z}}). \tag{3.20}$$

Thanks to the hyperprior, a more accurate entropy model is estimated, reducing the spatial redundancy in the latent $\boldsymbol{y}$. This hyperprior provide around 30% of coding gain for still image coding [159].

Autoregressive models have also been investigated [20] to further reduce the redundancy of latents. This approach proposes introducing the already decoded pixels in the hyperprior entropy model's parameter estimation. This approach is based on Pixel-CNN [160] and typically uses a 5×5 mask to process the causal context of latents. An overview of this approach is provided in Figure 3.12. However, the images are decoded sequentially, which results in a higher processing time.

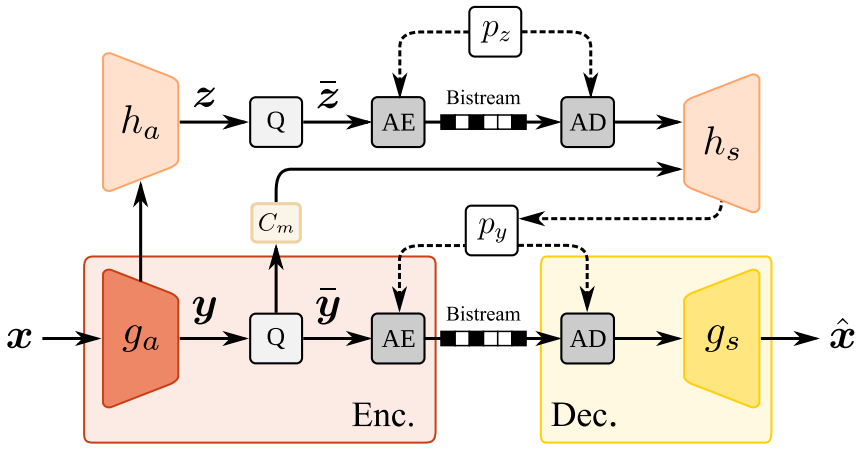


Figure 3.12 – Description of end-to-end compression using a VAE with hyperprior and autoregressive model [20].

Recently, the performance of VVC All-Intra configuration was reached by a learned image compression model [21], demonstrating all the potential of these methods. In this work, the

latent variables are entropy coded regarding a gaussian mixture model (GMM) parameterized by the output of the hyper-decoder h_s as:

$$p_y(\bar{\boldsymbol{y}}|\bar{\boldsymbol{z}}) \sim \sum_{k=1}^{K} \boldsymbol{w}^{(k)} \mathcal{N}(\boldsymbol{\mu}^{(k)}, \boldsymbol{\sigma}^{2(k)}), \tag{3.21}$$

with k the index of mixtures defined by $\boldsymbol{w}^{(k)}$, $\boldsymbol{\mu}^{(k)}$ and $\boldsymbol{\sigma}^{2(k)}$, denoting weights, means and scales, respectively. The authors proposed using residual blocks and attention modules in the architecture of the network, which further improved the performance compared to the state-of-the-art. The overall architecture is given in Figure 3.13 and 3.14.

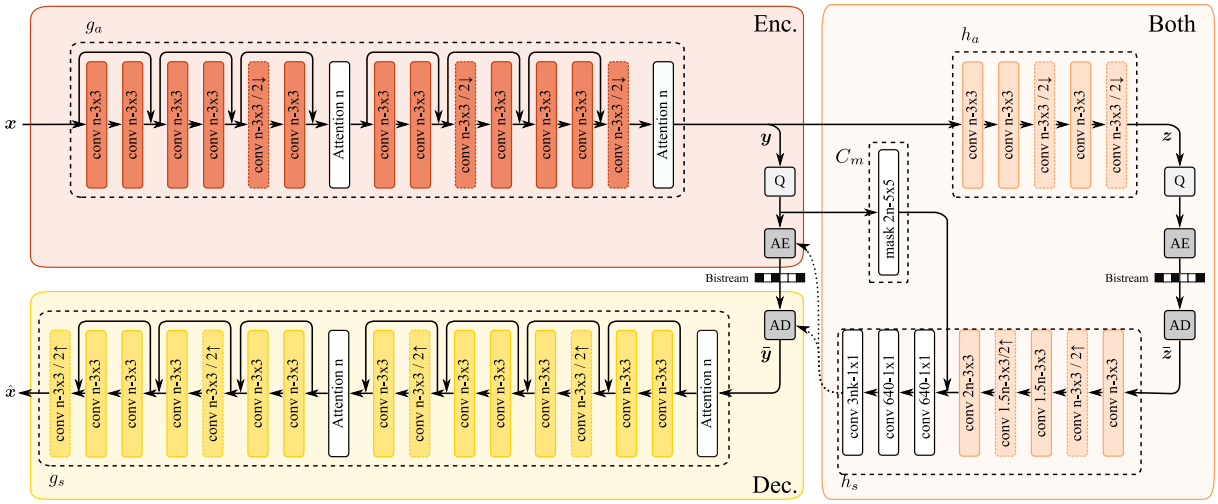


Figure 3.13 – Cheng *et al.* Architecture [21].

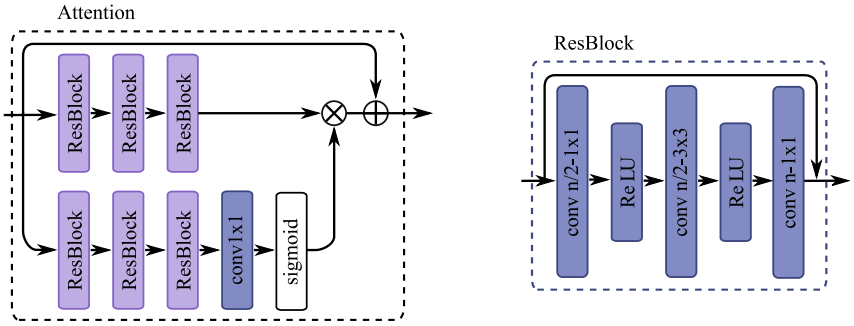


Figure 3.14 – Building blocks of Cheng *et al.* Architecture [21].

3.4.3 Learned video coding

While the first learning-based approaches focused on still image coding, a huge interest in applying this promising concept on video signals recently emerged. Indeed, video signals contain temporal redundancies that are not exploited by the aforementioned systems, making them not competitive compared to video coding modes of traditional codecs, such as low delay (LD) or RA. Although the traditional hybrid coding architecture has limitations, i.e., block partitioning and hand-crafted suboptimal system, the motion prediction scheme is highly efficient to compress videos by transmitting the temporal prediction error only to be sent through the network.

The first work that applies this principle is presented in [161]. The authors proposed using scalar quantization and Huffman coding on fixed 32×32 prediction and residual blocks. However, this approach does not benefit from the power of representation of autoencoders and falls in the same limitations of traditional systems by operating on blocks. The first fully-learned video codec, called deep video coding (DVC), was proposed in [162]. This system is inspired by the traditional MPEG hybrid coding architecture. The system operates on GOP, where a reference picture, called I-frames, is used to predict the following frames. Here, the optical flow is sent through the network to produce the predicted image to compensate at the decoder side. Thus, two networks are trained in an end-to-end fashion: one for residual compression and one for motion information compression. The two networks are trained jointly with the RD loss function. Thus, the network minimizes the RD loss function $\mathcal{L}$ by balancing the motion vectors and the residual prediction rates as:

$$\mathcal{L}(\lambda) = D + \lambda(R_y + R_v) \tag{3.22}$$

With D the distortion as defined in equation 3.15, R_y the rate of latents and R_v the rate of motion vectors as defined in equation 3.16.

Authors in [163] proposed computing the optical flow in multiple resolutions. Then, the network selects the appropriate flow resolution regarding the RDO, i.e., low-resolution for smooth areas and high-resolution for complex spatial areas. More advanced motion estimation methods have been later investigated. For instance, Djelouah *et al.* proposed computing the residual in the latent space instead of pixel space [164]. Thus, richer information is collected about the residual to reconstruct. Based on this work, authors in [165] proposed performing the motion compensation directly in the feature space. Deformable convolutions have later been investigated to further improve the performance [166]. When all the above works consider temporal information in the 2D pixel-space, authors in [167] consider each GOP as a 3D tensor

processed by a single network. While being low complex, this approach cannot handle high motion in the scene.

All the presented approach works for unidirectional prediction in a low-delay configuration. However, it is well known that a hierarchical structure provides better performance by taking advantage of the redundancy that occurs in different stages of a GOP. In extension to DVC [162], Lin *et al.* proposed using multiple reference frames to predict the current one. In their approach, authors in [168] proposed a hierarchical learned video coding scheme coupled with a QE network at the end of the coding pipeline. More recently, a deep video coding framework that holds all coding modes of a traditional video codec, i.e., all-intra, low-delay, and random access, has been developed [169]. This approach is based on conditional coding and allows disabling B or P predictions by setting connections to zero during inference.

The best learned video coding appraoches are now on par with HEVC for HD video sequences. However, it is still unclear how they behave on very high-resolution videos. Indeed, the motion estimation is generally performed on 256×256 images during training. However, the motion is usually represented in larger spatial areas for UHD scenes.

3.4.4 Layered approaches

Layered systems exploit the redundancies between different versions of a signal, to deliver multiple spatial resolution or level of quality in the same bitstream. Toderici *et al.* [170, 171] proposed a binary RNN to iteratively reconstruct the image by encoding the residual between the reconstructed image and the source. A layered-scalable multiscale autoencoder has been developed in [172]. Here both the BL and the multiple EL models are jointly encoded in an end-to-end fashion.

Some works proposed using hybrid coding architecture as a BL model enhanced by an autoencoder used an EL model. An overview of this method is given in Figure 3.15. Tsai *et al.* proposed transmitting the residual between a video encoded using AVC the source video as side information using a binary autoencoder [173]. The binary codes are then entropy-coded using a Huffman coding compression method [174]. The particularity of this work is that the network is overfitted regarding specific domains of applications, e.g. specific video games or natural image datasets. Thus, the Huffman coding tree can be stored at the receiver side to save bandwidth. Lee et al. proposed a hybrid architecture based on the VVC all-intra mode, which encodes the residual between the reconstructed image and the source using a VAE, as described in Section 3.4.1. This method provides higher subjective quality than the BL signal.

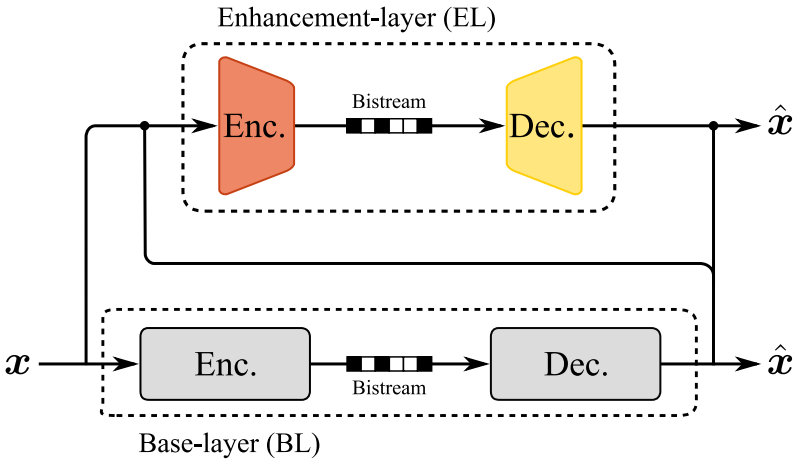


Figure 3.15 – Description of a layered system using a traditional codec as a BL and an autoencoder as an EL.

3.5 Conclusion

AI-based video compression is a recent and highly innovative field of research whose state-of-the-art, presented in this chapter, is getting richer day by day. The overview on restoration algorithms, i.e., SR and compression artifact reduction, demonstrated that the feed-forward neural network efficiently maps degraded images to non-degraded ones. Moreover, we see that modifications can be integrated into a downscaling-based compression scheme to improve the quality of HR reconstruction. In addition to these methods, end-to-end compression algorithms proposed promising performance as a codec or to enhance existing codecs with a layered architecture.

The following chapters will focus on integrating these approaches to answer the use-case of this manuscript described in the previous chapter.

PART III

Contributions

CHAPTER 4

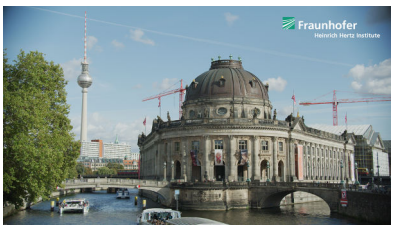
EVALUATION OF ALGORITHMS AND STANDARDS FOR 8K VIDEO DELIVERY

4.1 Preamble

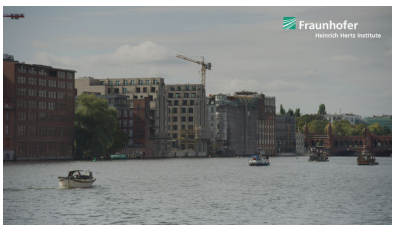
As mentioned in the state-of-the-art, 8K resolution video (7680x4320) has attracted a lot of interest from the industry. This media format dedicated to immersive video applications aims to improve the end user's QoE by increasing the amount of spatial information from the scene. On one hand, several studies have demonstrated that high bitrate requirements are needed for 8K services using HEVC [34, 35, 36], preventing their deployment over the DTT. To tackle this, contributions to compression standards like VVC [175, 27], released in June 2020, would make the delivery of 8K video contents more affordable. On the other hand, backward compatibility with 4K legacy receiver has to be considered for a successful deployment of 8K services on broadcast infrastructures. However, scalable coding approaches, such as SHVC [9], are difficult to consider because of their dependence on the BL codec. Recently, AI-based up-scalers [10, 176, 15] have shown outstanding performance over classical interpolation filters like Lanczos [14] or bicubic [177]. These algorithms may reconstruct an 8K signal through the transmission of a single 4K bitstream, enabling codec agnostic backward-compatibility with any UHD-1 receivers.

This chapter provides objective and subjective evaluations of standards and algorithms using a dedicated 8K resolution video dataset. First, Section 4.2 describes our 8K video test dataset. Section 4.3 assesses the objective and subjective quality of HEVC and VVC for 8K video coding. This section also evaluates the perceptual gain offered by 8K over 4K for each of the tested scenes. Section 4.4 evaluates algorithms enabling 8K video delivery with 4K backward compatibiltiy, including SHVC and spatial upscaling using super-resolution and a Lanczos filter. Finally, Section 6.5 concludes this chapter.

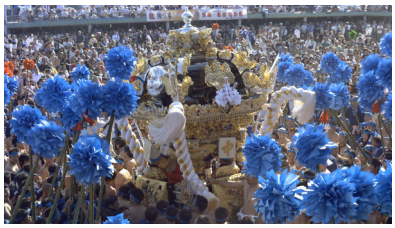
(a) LayeredKimono



(b) BodeMuseum



(c) OberbaumSpree



(d) Festival2

(e) JapaneseMaple

(f) SteelPlant

Figure 4.1 – Snapshots of the selected 8K test video sequences.

4.2 8K Video Dataset

This section presents our proposed 8K resolution video dataset. First, a description of the scenes is provided. Then the parameters of the sequences and their characteristics are given.

4.2.1 Description of the dataset

To fairly evaluate compression algorithms, CTCs video sequences have been defined by working groups like JVET. For instance, the dataset described in [23] comprises short video clips ranging from 480p to 2160p organized into six resolution classes (from E to A). The selected sequences generally last 10 seconds and have various spatial and temporal characteristics, and also show different contents (sport, video surveillance, TV news) in order to evaluate the compression algorithms regarding different scenarios. However, no 4320p sequences are represented in CTCs which prevents the evaluation of compression algorithms on 8K contents. Moreover, the limited availability of high-quality, uncompressed 8K video sequences motivates the construction of this 8K video dataset.

It is acknowledged that the higher the spatial resolution is, the more challenging the task of sampling for sensors is. Thus, high-quality hardwares must be considered to produce premium 8K contents. Although the amount of 8K video contents is increasing [1], high-quality, uncompressed

1. `https://www.youtube.com/c/8KAssociation/playlists/`

Table 4.1 – Description of the 8K test video sequences.

Video title	Description
LayeredKimono	Shows a woman wearing traditional Japanese clothes. The camera performs a very small traveling from the left to the right. There are details on the costume and the face of the woman.
BodeMuseum	Shows the Bode-Museum of Berlin. The camera is fixed. There are text and details on the architecture of the monument. There are reflections in the water in the foreground. A logo is represented in the top right of the scene.
OberbaumSpree	Shows the Spree river in Berlin. The camera makes a traveling from left to right. There are some boats moving in the front and buildings in the scene's background. A logo is represented in the top right of the scene.
Festival2	Shows people celebrating during a traditional Japanese festival. The camera is fixed, but there is high motion in the scene.
JapaneseMaple	Shows trees in fall. The camera is fixed, and the focus is done in the scene's center. The leaves are moving with the wind.
SteelPlant	Shows a foundry. The camera is fixed. There are sparkles and smoke, which represent most of the motion in the scene.

8K videos are still limited. We identified two 8K raw video sources: the Institute of Image Information and Television Engineers (ITE) [2] and the HHI [178] 8K video databases.

First, we selected multiple videos from those two sources resulting in 16 different video clips. All the sequences were evaluated by experts on an 8K TV screen to choose the contents based on video features like color, movement, texture, and homogeneous content, leading to different behaviors of the compression algorithms. We also considered the relevance of the 8K resolution in the scene selection. From this evaluation session, six video sequences were chosen. Screenshots of the selected scenes are given in Figure 4.1. Table 4.1 enumerates the selected video clips and their characteristics.

4.2.2 Sequence parameters

The details of the 8K test sequences are reported in Table 4.2. To ensure homogeneity over video sequences, we performed a color space conversion from BT.709 [2] to BT.2020 [1] for *BodeMuseum* and *OberbaumSpree* scenes. Also, as the sequences *LayeredKimono*, *Festival2*, and *JapaneseMaple* contain fewer frames than the others, we played them back in mirror mode

2. `https://www.ite.or.jp/content/test-materials/`

Table 4.2 – Parameters of the 8K test video sequences. All sequences are in 4:2:0 color subsampling format.

Sequence	Resolution (W × H)	Frame-rate	Frames	Color space	Bitdepth	Source
BodeMuseum	7680×4320	60fps	600	BT.709	10	HHI
OberbaumSpree	7680×4320	60fps	600	BT.709	10	HHI
LayeredKimono	7680×4320	60fps	300	BT.2020	10	ITE
Festival2	7680×4320	60fps	300	BT.2020	10	ITE
JapaneseMaple	7680×4320	60fps	300	BT.2020	10	ITE
SteelPlant	7680×4320	60fps	600	BT.2020	10	ITE

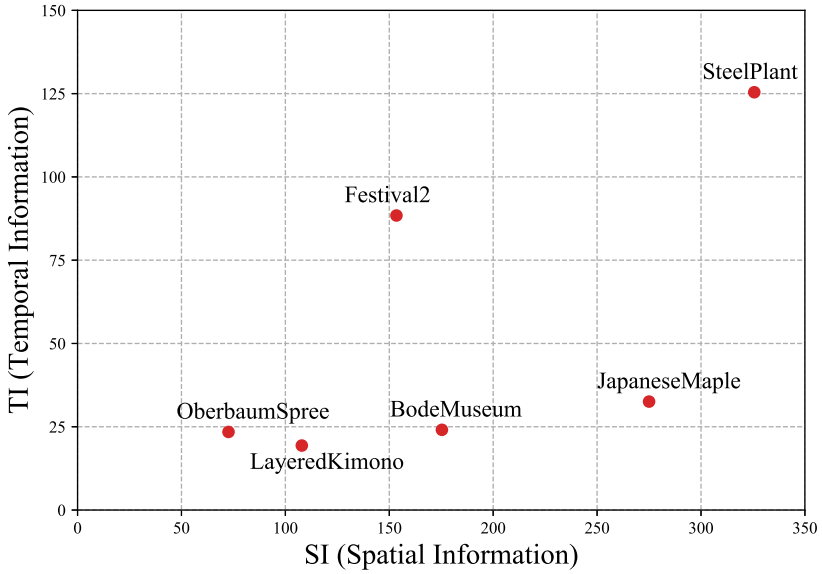


Figure 4.2 – SI-TI graph of the tested 8K video sequences.

after 5 seconds to get 10 seconds videos while preserving the motion continuity of the scene. For those sequences, the motion direction change was coherent with the initial content.

4.2.3 Statistical study

The SI-TI [11] graph of the selected sequences is plotted in Figure 4.2. This 2D plan shows that the selected contents are diverse regarding spatio-temporal features. Those criterias provide guidelines about the diversity across the dataset in terms of spatiotemporal variations. In addition to the nature itself of the contents, test video datasets must take into account that diversity in order to generalize the obtained results.

4.3 Single layer coding standards for 8K video

This section provides both objective and subjective quality assessments of the two latest MPEG video coding standards for 8K video coding. Compression points for each 8K video scene presented in 4.2 have been generated using the random access (RA) mode of the VVC and HEVC reference software models, called VTM-11 and HM-16.20, respectively. For subjective quality assessment, we used the double stimulus continuous quality scale (DSCQS) method. This study includes RD curves, BD bitrate evaluation, and a Student's t-test, offering a robust statistical analysis.

The contributions of this work are the following:

— Assess the compression gain offered by VVC over HEVC standards for 8K video contents. This gain represents approximately 41% of bitrate saving for the same visual quality,

— Determine the required bitrate for transparency, i.e., no visual difference is perceived between the source and decoded video,

— Confirm that non-expert viewers can see the difference between 4K and 8K resolutions and measure that difference,

— Evaluate several objective quality metrics based on the subjective test statistics collected on the 8K video dataset.

This section is organized as follows. Section 4.3.1 describes the subjective test materials, including the test sequences, the codecs configuration, and the subjective test methodology. The results of both the objective and the subjective experiments are given in Section 5.3.3. Finally, Section 6.3.3 concludes this section.

4.3.1 Experimental settings

Test video sequences

For this experiment, we used the 8K video dataset described in Section 4.2. Based on these six uncompressed (raw) selected 8K video sequences (scenes), ten processed video sequences (PVSs) are generated per scene:

— one 8K (7680×4320) hidden reference uncompressed video.

— one 4K (4320×2160) uncompressed video. In that case, the source signal is first down-scaled to 4K and then rescaled to 8K by using the *Lanczos3* [14] filter provided by *ffmpeg* [3] for both operations.

3. `https://www.ffmpeg.org/`

Table 4.3 – Selected QP and corresponding bitrates (Mbps), for both VTM-11 and HM-16.20 codecs, according to the test sequence.

Sequence	Codec	R_1 (QP/Mbps)	R_2 (QP/Mbps)	R_3 (QP/Mbps)	R_4 (QP/Mbps)
LayeredKimono	HEVC	38/1.9	34/3.2	29/6.3	26/11.4
	VVC	37/1.8	32/3.4	27/6.5	24/10.8
BodeMuseum	HEVC	38/4.7	33/9.8	28/22.5	25/45.4
	VVC	37/4.8	32/10.1	27/22.6	24/42.9
OberbaumSpree	HEVC	38/3.3	33/7.4	28/17.5	24/40.5
	VVC	37/3.6	32/8.1	27/18.6	23/43.9
Festival2	HEVC	39/17.5	34/32.1	29/59.5	24/130.4
	VVC	37/17.4	32/32.2	27/61.1	22/135.5
JapaneseMaple	HEVC	43/15.2	38/34.9	33/76.1	28/168
	VVC	42/15.9	37/35.7	32/79.8	27/174.9
SteelPlant	HEVC	42/19.6	38/40.5	33/86.9	28/175.5
	VVC	42/18.0	37/42.9	32/91.1	27/180.5

— 8K video encoded at four bitrates with HEVC.
— 8K video encoded at four bitrates with VVC.

In total, 60 video sequences are evaluated in this study.

The CTCs for VTM-11 [179] and HM-16.20 [72] in RA coding mode for main10 profile were used to perform a fair rate/distortion evaluation. These software models provide a reference implementation of the compression standards, representing their upper-bound coding performance with a moderate optimization level. For both codec, a GOP size of 16 and an Intra Period of 64 frames were used. For each scene, the test points are obtained using different fixed quantization parameter (QP) values. To cover a wide range of visual quality, we determined the highest bitrate value considering the transparency, i.e., the bitrate for which degradation starts to appear, as the highest bitrate point for each sequence. Also, the bitrates were carefully selected so that each bitrate R_i is approximately half of the next bitrate R_{i+1} and each VVC bitrate R_i^{VVC} is equal to the corresponding HEVC bitrate R_i^{HEVC} for $i \in \{1, 2, 3, 4\}$. The used QPs and bitrates for each sequence are given in Table 4.3. We can note that the bitrate selected for transparency varies from 11Mbps to 180Mbps, depending on the test sequence.

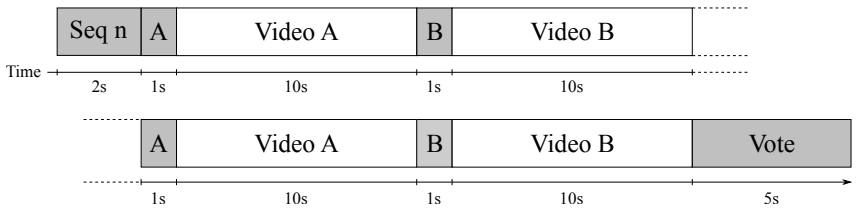


Figure 4.3 – Subjective basic test cell (BTC) structure according to the DSCQS evaluation methodology.

Experimental environment and testing procedure

In this study, we used the method described in the ITU-R Rec BT500-14 [11], called double stimulus continuous quality scale (DSCQS), to collect the video quality scores from participants. This testing method requires a prior pseudo-random sequencing of the testing videos, as the observer has no interactivity with the player. Thus, each test session of the DSCQS method consists of different random series of basic test cells (BTCs) presentations. This method presents the test videos by pairs ("video A" and "video B") separated with annotated mid-greys. For each BTC, both "video A" and "video B" are repeated twice. An example of BTC used for evaluation is illustrated in Figure 4.3. Each presented pair contains the implicit 8K uncompressed reference and one random PVS over all the ten configurations, i.e., the same scene encoded with HEVC or VVC at four bitrates or the uncompressed sequence in 4K or 8K resolution. Moreover, to prevent visual fatigue, the test is divided into three sessions of 20 minutes each. Before each experiment, participants receive clear explanations about the evaluation procedures.

After the first "video A/video B" pair presentation, the participant could report his opinion about the perceived video quality on two vertical lines with the corresponding sequence index for both "video A" and "video B". For this testing method, the vertical rating lines are divided into five segments of the same height and scaled from the lower to the higher quality with the labels *Bad*, *Poor*, *Fair*, *Good*, and *Excellent*. After each video pair visualization, participants can vote by annotating both videos along the continuous quality scale. The scores are then collected by converting the annotations into a value between 0 and 100.

This subjective study has been conducted in a controlled laboratory environment that follows the ITU-R Rec. BT500-13 [180]. The objective is to offer visualization comfort to participants and ensure the reproducibility of the test. All the experimental setup details are reported in Table 4.4. An illustration of the vizualisation workflow is given in Figure 4.4, and a picture illustrating the test conditions in Figure 4.5. A total of 22 non-expert observers aged from 22 to 53 years have taken part in this experiment. All participants have been screened for normal

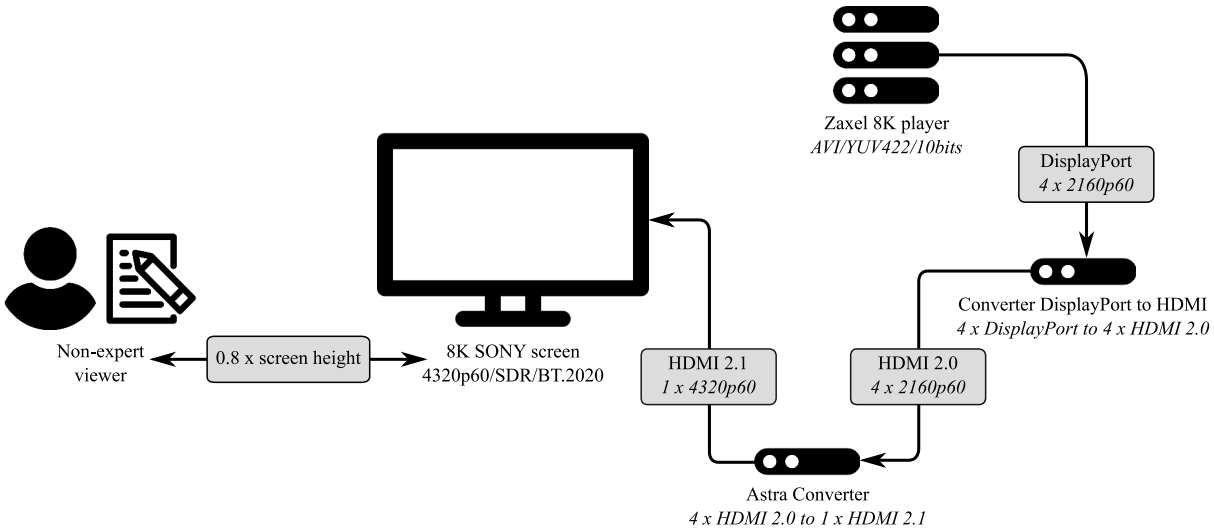


Figure 4.4 – Vizualisation workflow.

Table 4.4 – Test logistics.

Monitor	SONY 85" KD-85ZG
Player	Zaxel's Zaxtar 5 8K
Peak luminance	120 cd/m^2
Video Format	7680x4320/60p/YUV4:2:0/10bits
Viewing distance	0.75H (approximtely 0.8m)
Background color	D65 mid-grey
Background luminance	15% of the screen maximum luminance

visual acuity and color blindness using the Ishihara and Snellen vision tests, as described in the ITU-R Rec BT500-14 Recommendation [11]. To detect outliers, the rejection method based on the Kurtosis coefficient from this same recommendation has been applied and has validated the overall participant's reported votes.

4.3.2 Subjective quality assessment

At the end of the subjective test sessions, the results for each scene are assessed by the DMOS, corresponding to the average of the difference between the hidden reference and the corresponding PVS scores computed by:

$$\bar{x}_a = \frac{1}{n} \sum_{i=1}^{n} x_{i,a}, \tag{4.1}$$

Figure 4.5 – Subjective test conditions.

where n is the total number of valid participants, $\bar{x}_a$ is the DMOS value of the tested configuration a, $a \in \{R_j^m$, 4K, 8K (ref)$\}$ for $j \in \{1, 2, 3, 4\}$ and $m \in \{VVC, HEVC\}$ and $x_{i,a}$ is the differential score computed as:

$$x_{i,a} = 100 - (y_{i,ref} - y_{i,a}), \tag{4.2}$$

with the pair $(y_{i,ref}, y_{i,a})$ representing the scores attributed by the participant i, $i \in \{1, \ldots, n\}$, to respectively the hidden reference (8K) and the tested configuration a, i.e. both videos of a given BTC.

To ensure that the vote distributions are normal, the bias reduction technique described in the ITU-T P.913 Recommendation [181] has been applied. Thus, from each resulting DMOS $\bar{x}_a$, the associated confidence intervals at 95% $(\bar{x}_a - c_a, \bar{x}_a + c_a)$ can be computed as follows:

$$c_a = 1.96 \frac{s_a}{\sqrt{n}}, \tag{4.3}$$

where s_a is the standard deviation of the tested configuration a computed as:

$$s_a = \sqrt{\sum_{i=1}^{n} \frac{(x_{i,a} - \bar{x}_a)^2}{(n-1)}}, \tag{4.4}$$

with $x_{i,a}$ and $\bar{x}_a$ corresponding to the differential score of the observer i, $i \in \{1, \ldots, n\}$, and the DMOS score of the tested configuration a, respectively.

In addition, a Student's t-test with a two-tailed distribution is performed to provide a more rigorous analysis. More details are given in the following section.

4.3.3 Experimental results

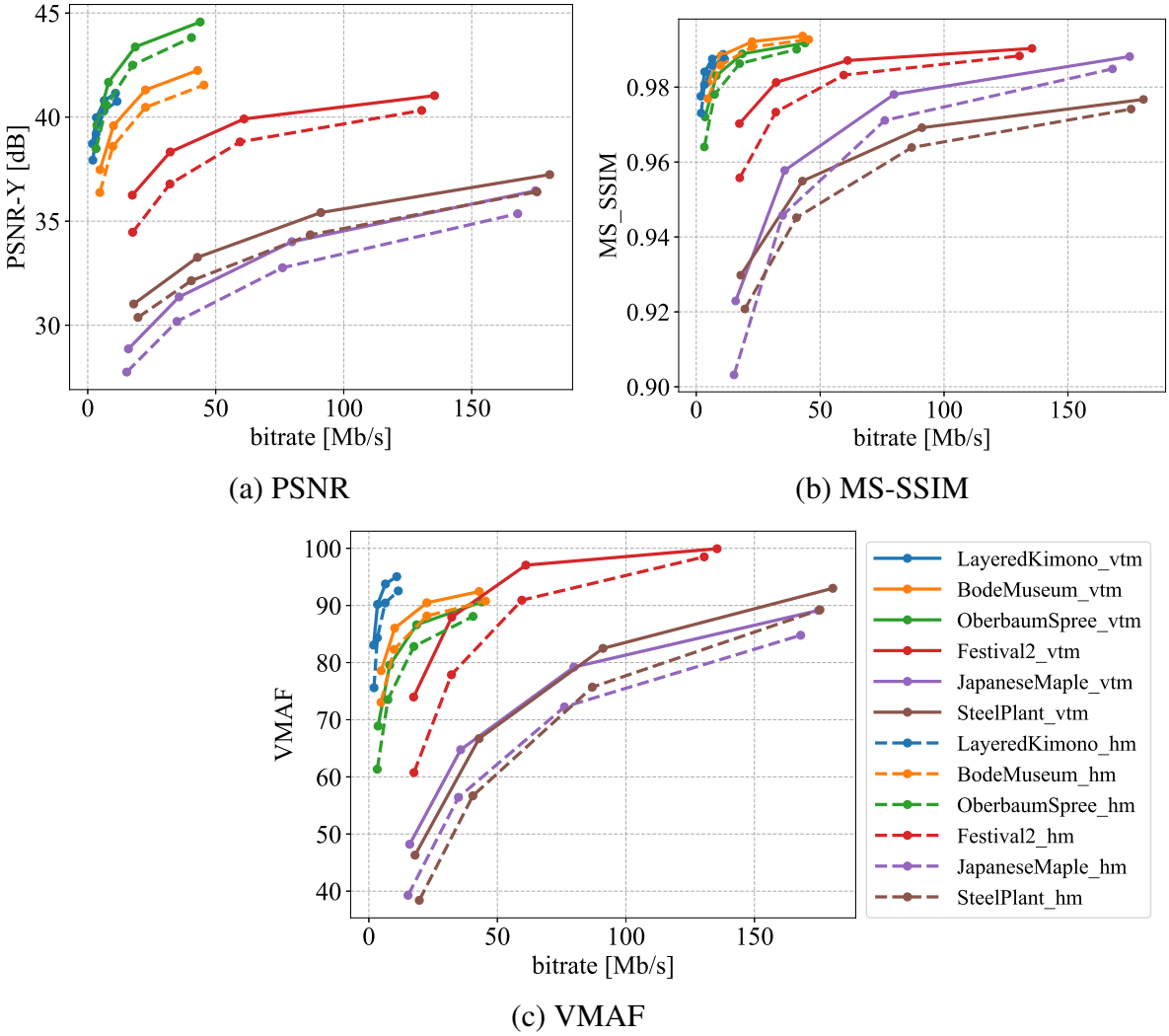
Objective quality assessment

In this experiment, objective quality metrics, incuding PSNR, MS-SSIM [47], and VMAF [13], are used to measure the distortion between the 8K reconstructed signal and the source video. VMAF is an objective metric with reference, based on ML which evaluates the quality between the source and the tested content by giving a score between 0 and 100. This metric is trained to produce a score computed from different features (motion, spatial, texture) that maximize the correlation with MOS scores. In this experiment, the VMAF scores are computed with the provided set of parameters *vmaf_v0.6.1.pkl*[4]. Although the selected VMAF model is optimized for visual quality estimation of 4K contents, we have integrated it into the study as it achieves a high correlation with subjective scores. The PSNR is assessed on the luma component only.

The RD curves are depicted in Figure 4.6. It can be noted that the bitrates selected for transparency lead to quite different PSNR values depending on the sequence. In contrast, for more perceptually correlated objective metrics like MS-SSIM or VMAF, the predicted quality converges to the maximum value for all 8K sequences. Also, those curves confirm the observation made on the scene complexity with the SI-TI graph in Figure 4.2. Three categories of sequences can be distinguished by scene complexity: Group 1 includes *LayeredKimono, OberbaumSpree, BodeMuseum* sequences, Group 2: *Festival2*, and Group 3: *JapaneseMaple, SteelPlants*.

We use the Bjontegaard-Delta (BD) computation method described in [44] to quantify the average gain in bitrate and visual quality offered by the VTM-11 over the HM-16.20 codec. The results are summarized in Table 4.5. In average, the VTM-11 codec enables around 31%, 26% and 35% of bitrate saving over the HM-16.20 codec, regarding PSNR, MS-SSIM and VMAF, respectively. However, the area between the interpolated curves covered using the BD-BR approach is limited as the selected bitrates are the same for both VVC and HEVC standards. Thus, to bring more details on the performance and consider a wider area between the curves, we compute the gain in quality of the VTM-11 over the HM-16.20 for the same bitrate using the BD method. The results are reported in Table 4.6 By considering this approach, 0.91dB, 0.005 and 5.48 of quality improvement is offered by the VTM-11 over the HM-16.20 codec for the same bitrate, regarding PSNR, MS-SSIM and VMAF quality metrics, respectively.

4. `https://github.com/Netflix/vmaf`



(a) PSNR

(b) MS-SSIM

(c) VMAF

Figure 4.6 – Objective quality comparison, using PSNR, MS-SSIM, and VMAF quality metrics for selected 8K video sequences.

Subjective quality assessment

For the subjective quality evaluation, the rectified DMOS scores and their associated 95% confidence interval are collected following the method described in Section 4.3.2. The resulting RD curves are depicted in Figure 4.7 for all 8K sequences. These curves also display the scores obtained for the 8K hidden reference videos and the 4K sequences, with their associated 95% confidence interval represented by transparent areas.

In order to confidently evaluate the statistical significance of the similarity (or not) between different tested sequences, we also performed a two-sample unequal variance Student's t-test

Table 4.5 – BD-BR scores of the VTM-11 codec compared to the anchor HM-16.20.

Sequence	BD-BR (PSNR)	BD-BR (MSSSIM)	BD-BR (VMAF)	BD-BR (DMOS *upper* and *lower* limits)
LayeredKimono	-29.77%	-21.05%	-33.30%	-44.99% [-60.92%, -20.04%]
BodeMuseum	-32.75%	-25.05%	-34.70%	-36.43% [-74.71%, +21.12%]
OberbaumSpree	-32.07%	-27.00%	-33.41%	-55.59% [-87.15%, +28.59%]
Festival2	-36.40%	-33.36%	-28.24%	-28.89% [-59.43%, +37.28%]
JapaneseMaple	-28.33%	-23.37%	-30.86%	-43.36% [-64.42%, -6.69%]
SteelPlant	-28.30%	-24.40%	-27.57%	-37.41% [-67.61%, +13.31%]
Average	**-31.27%**	**-25.7%**	**-35.30%**	**-41.11% [-69.04%, +12.26%]**

Table 4.6 – BD scores of the VTM-11 codec compared to the anchor HM-16.20.

Sequence	BD-BR (PSNR)	BD-BR (MSSSIM)	BD-BR (VMAF)	BD-BR (DMOS *upper* and *lower* limits)
LayeredKimono	+0.61dB	+0.003	+4.63	+10.76 [+19.3, +2.22]
BodeMuseum	+0.88dB	+0.002	+3.06	+5.79 [+15.21, -3.63]
OberbaumSpree	+0.81dB	+0.003	+4.09	+7.87 [+18.44, -3.35]
Festival2	+1.22dB	+0.006	+7.37	+5.13 [+12.98, -2.72]
JapaneseMaple	+1.04dB	+0.009	+6.63	+9.79 [+18.27, +1.31]
SteelPlant	+0.91dB	+0.007	+7.10	+8.83 [+20.40, -2.74]
Average	**+0.91dB**	**+0.005**	**+5.48**	**+8.03 [+17.43, -1.49]**

with a two-tailed distribution. This study allows us to determine, for each scene, if the perceived quality between each pair of tested configurations is significantly different or not.

In this experiment, regarding two different tested configurations a_1 and a_2 for a given scene, the null hypothesis, H_0, corresponds to the case that a_1 and a_2 have the same perceived quality. On the contrary, the alternate hypothesis, H_a, would be that a difference between the tested configurations a_1 and a_2 is noted.

The t-statistic can be estimated to quantify the degree of significance of the alternate hypothesis H_a. By considering the sample populations x_{a_1} and x_{a_2} from attributed scores for the tested configuration a_1 and a_2, respectively, the t-statistic can be computed as follows:

$$t_{a_1,a_2} = \frac{\bar{x}_{a_1} - \bar{x}_{a_2}}{\sqrt{\frac{s_{a_1}^2}{n_{a_1}} + \frac{s_{a_2}^2}{n_{a_2}}}}, \tag{4.5}$$

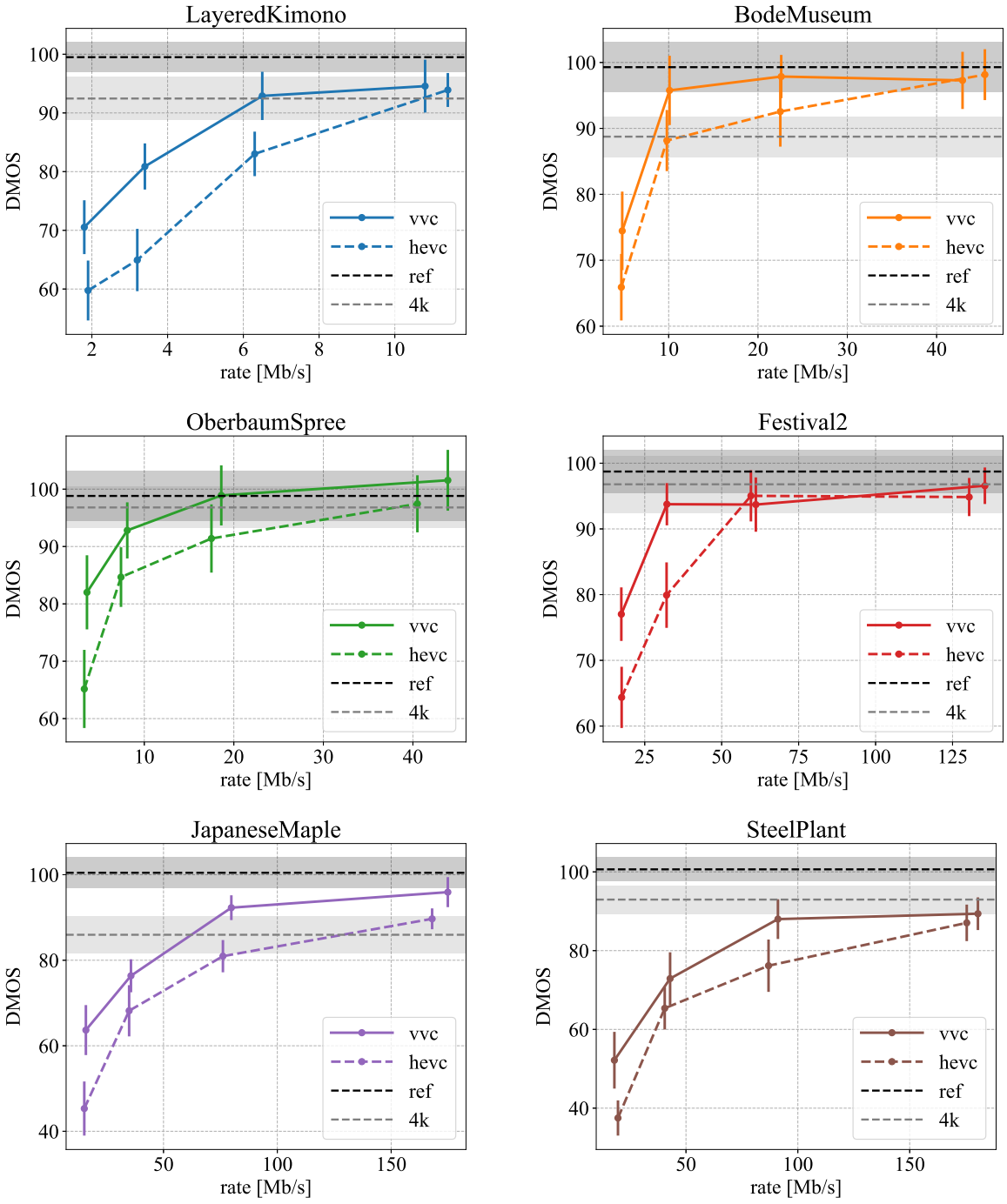


Figure 4.7 – DMOS-based comparison, with associated 95% confidence interval, for the selected 8K video sequences.

with $\bar{x}_{a_j}$, $s^2_{a_j}$ and n_{a_j} denoting the mean, the variance and the size of the sample population x_{a_j}, with $j \in \{1, 2\}$.

Table 4.7 – p-value probabilities resulting from two-sample unequal variance bilateral Student's t-test on DMOS values for each pair of tested configurations and each selected 8K video sequence.

(a) LayeredKimono

HEVC \ VVC	R1	R2	R3	R4	4K	REF
R1	**0.01**	0.00	0.00	0.00	0.00	**0.00**
R2	**0.15**	**0.00**	0.00	0.00	0.00	**0.00**
R3	0.00	**0.44**	**0.00**	0.00	0.00	**0.00**
R4	0.00	0.00	**0.79**	**0.70**	0.65	**0.01**
4K	0.00	0.00	0.88	0.47	**1.00**	**0.01**
REF	**0.00**	**0.00**	**0.02**	**0.10**	**0.01**	**1.00**

(b) BodeMuseum

HEVC \ VVC	R1	R2	R3	R4	4K	REF
R1	**0.04**	0.00	0.00	0.00	0.00	**0.00**
R2	**0.00**	**0.06**	0.00	0.01	0.90	**0.00**
R3	0.00	**0.44**	**0.13**	0.21	0.28	**0.07**
R4	0.00	0.56	**0.98**	**0.86**	0.00	**0.62**
4K	0.00	0.05	0.00	0.01	**1.00**	**0.00**
REF	**0.00**	**0.32**	**0.58**	**0.53**	**0.00**	**1.00**

(c) OberbaumSpree

HEVC \ VVC	R1	R2	R3	R4	4K	REF
R1	**0.00**	0.00	0.00	0.00	0.00	**0.00**
R2	**0.61**	**0.04**	0.00	0.00	0.04	**0.01**
R3	0.06	**0.74**	**0.09**	0.02	0.16	**0.07**
R4	0.00	0.23	**0.71**	**0.31**	0.85	**0.71**
4K	0.00	0.23	0.55	0.18	**1.00**	**0.52**
REF	**0.00**	**0.10**	**0.98**	**0.47**	**0.52**	**1.00**

(d) Festival2

HEVC \ VVC	R1	R2	R3	R4	4K	REF
R1	**0.00**	0.00	0.00	0.00	0.00	**0.00**
R2	**0.41**	**0.00**	0.00	0.00	0.00	**0.00**
R3	0.00	**0.34**	**0.55**	0.70	0.73	**0.26**
R4	0.00	0.42	**0.68**	**0.44**	0.53	**0.11**
4K	0.00	0.21	0.37	0.98	**1.00**	**0.48**
REF	**0.00**	**0.02**	**0.09**	**0.36**	**0.48**	**1.00**

(e) JapaneseMaple

HEVC \ VVC	R1	R2	R3	R4	4K	REF
R1	**0.00**	0.00	0.00	0.00	0.00	**0.00**
R2	**0.33**	**0.04**	0.00	0.00	0.00	**0.00**
R3	0.00	**0.13**	**0.00**	0.00	0.12	**0.00**
R4	0.00	0.00	**0.24**	**0.00**	0.18	**0.00**
4K	0.00	0.00	0.04	0.00	**1.00**	**0.00**
REF	**0.00**	**0.00**	**0.00**	**0.14**	**0.00**	**1.00**

(f) SteelPlant

HEVC \ VVC	R1	R2	R3	R4	4K	REF
R1	**0.00**	0.00	0.00	0.00	0.00	**0.00**
R2	**0.00**	**0.11**	0.00	0.00	0.00	**0.00**
R3	0.00	**0.55**	**0.01**	0.00	0.00	**0.0**
R4	0.00	0.00	**0.91**	**0.50**	0.07	**0.00**
4K	0.00	0.00	0.11	0.24	**1.00**	**0.00**
REF	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**1.00**

Then, by approximating the t-statistic with a Student's t-distribution, a value p, which indicates the degree of correlation between the means of the two sample populations, can be computed from the t-statistic. The higher the p-value is, the more significant the similarity between the distributions of the two populations is. A p-value lower than 0.05 indicates that there is a statistical significance that the two sample populations x_{a_1} and x_{a_2} have a different perceived quality. Indeed, there is a low probability of committing a type-I error, i.e., rejecting the null hypothesis when it is true, meaning that the null hypothesis can be confidently rejected. On the contrary, if the p-value is greater than or equal to 0.05, the null hypothesis cannot be safely rejected and both sample populations x_{a_1} and x_{a_2} can be considered to have the same perceived quality. The results for all scenes are given in Table 4.7.

The results demonstrate that the perceived quality between uncompressed 8K and 4K formats depends on the scene content. Thus, for the sequences *JapaneseMaple*, *SteelPlant*, *BodeMuseum*, and *LayeredKimono*, the visual quality between both resolutions is significantly different. As

the p-value between the configurations 4K and REF is lower than 0.05. For those sequences, the global motion in the scene is low, which facilitate the sampling of 8K details by sensors. In contrast, for the sequences with non-significant visual difference between 8K and 4K resolutions (*Festival2* and *OberbaumSpree*), the motion in the scene can explain the 8K definition loss at 60fps. Indeed, the global motion in *Festival2* video sequence prevents from perceiving the details. For the *OberbaumSpree* motion blur appears on the scene due to a continuous horizontal camera traveling. It shows that higher framerates, e.g. 100/120fps, must be considered to fully benefit from the 8K resolution.

In complement to the objective study, we observe that the bitrate required to obtain transparency with the uncompressed 8K videos is highly content-dependent. Using VVC, the bitrates needed to reach the reference's quality are between 10Mbps to 80Mbps depending on the sequence, except for *SteelPlant* and *JapaneseMaples*, which are the most complex contents as pointed in Section 4.2. For these scenes, the quality degradation with the source is always perceived on the selected bitrate range. Indeed, the p-values obtained between all R_i^{VVC} and REF configurations are lower than 0.05 for this sequence. It can be explained by the smoke in the scene, which is hard to compress and causes blocking artifacts. In comparison, the 8K source quality is obtained only for three scenes using HEVC: *BodeMuseum*, *Festival2*, *OberbaumSpree*. However, two of them are not critical (*Festival2*, *OberbaumSpree*), as no significant difference between 8K and 4K is perceived $(p > 0.05)$.

In addition, we can notice that, at the same bitrate, VVC offers perceived quality closer to the 8K reference video comparing to HEVC. For both *JapaneseMaple* and *LayeredKimono* scenes, a bitrate reduction of 50% is reached for the same level of visual quality. Indeed, we can observe in Table 4.7 that, for those two scenes, each VVC test point of bitrate R_i^{VVC} is statistically similar or better in terms of visual quality with respect to its corresponding HEVC test point at bitrate R_{i+1}^{HEVC} and significantly better at bitrate R_i^{HEVC}. Nevertheless, the results obtained with the rest of the 8K sequences with lower spatial textures do not follow this observation.

Finally, we applied the BD-BR method to the DMOS scores. Inspired by [58], we also compute the *upper* and *lower* limits for the BD-BR based on the confidence intervals. These scores are computed by comparing D_{max}^{VVC} with D_{min}^{HEVC} and D_{min}^{VVC} with D_{max}^{HEVC}, respectively, where $[D_{min}, D_{max}]$ represents the 95% confidence interval. All the results are reported in Table 4.5 and Table 4.6. These results demonstrate that VVC offers a compression gain over HEVC for the same perceived quality from 28.89% to 55.59% with an average of 41.11% over the whole 8K test dataset.

Table 4.8 – Logistic model coefficients regarding each tested objective metric.

Objective metric	β_1	β_2	β_3	β_4
MS-SSIM	316145.66	32.03	1.5	0.06
SSIM	202.32	-5828.93	-0.88	0.46
VMAF	164.66,	-34309.47	-544.33	103.13
PSNR	135.79	-41584.66	-99.69	20.57

Correlation consistency

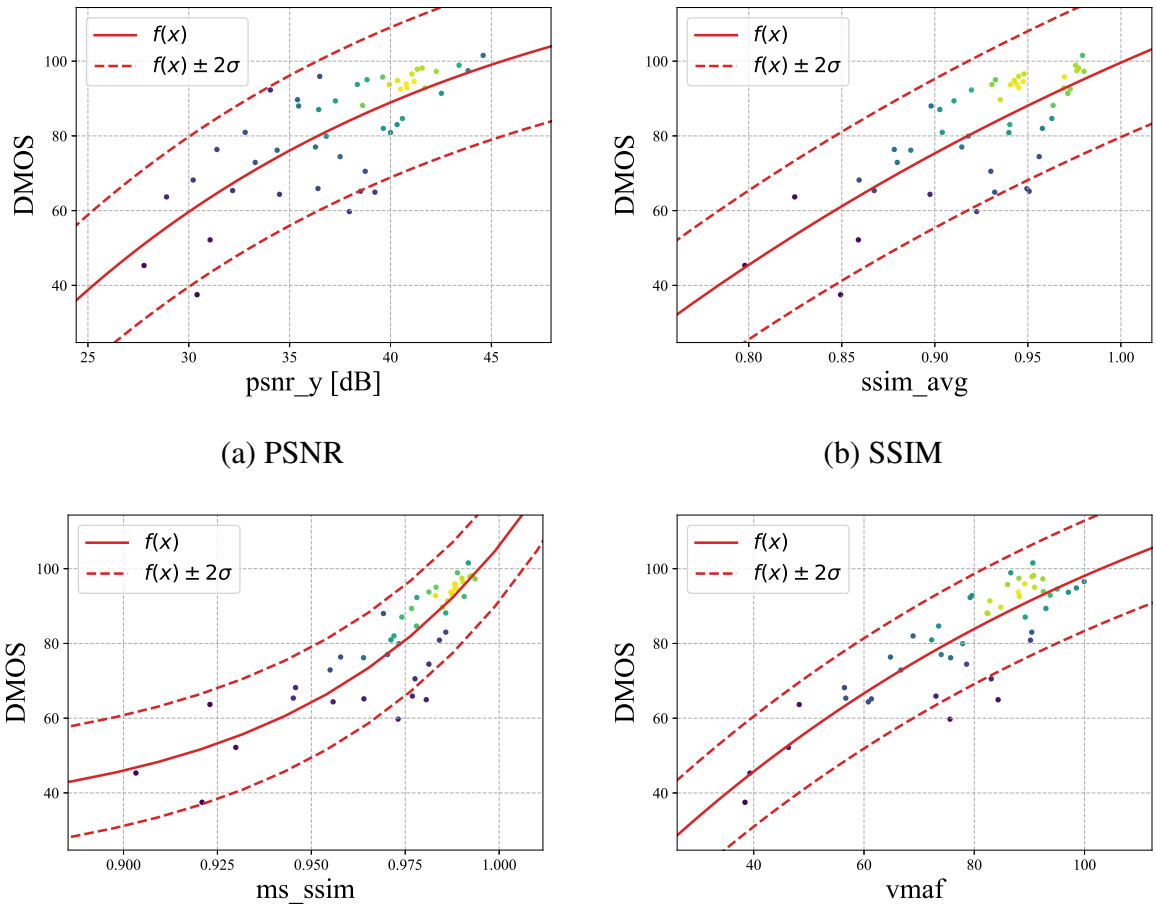
In this section, the consistency of objective quality metrics with subjective scores is evaluated. Figure 4.8 illustrates scatter plots with nonlinear logistic fitted curves $f(x)$ and corresponding standard deviations intervals $f(x) \pm 2\sigma$ for PSNR, SSIM, MS-SSIM , and VMAF quality metrics versus DMOS scores.The interpolated curves $f(x)$ are computed using the following logistic model:

$$f(x) = \beta_2 + \frac{\beta_1 - \beta_2}{1 + e^{-\frac{x-\beta_3}{|\beta_4|}}}. \tag{4.6}$$

The coefficient of this logistic model are given in Table 4.8 for each tested objective metric. The more the standard deviation intervals are close to the logistic fitted curve, the more the metric is correlated to the DMOS score. In order to quantify the correlation of the objective metrics with the subjective scores, we use the Spearman's rank ordered correlation (SROCC), Pearson's linear correlation coefficient (PLCC), Kendall's rank-order correlation coefficient (KROCC), and root mean-squared error (RMSE). The results are reported in Table 4.9. As expected, it shows that MS-SSIM and VMAF are more correlated to subjective test ratings than PSNR, which gets the lowest performance regarding all indicators. In addition to the three considered objective quality metrics, we provide correlation scores with the SSIM metric. This latter shows slightly higher correlation with DMOS compared to PSNR, while it is outperformed by both MS-SSIM and VMAF. Finally, we can notice that VMAF is a relevant quality metric for 8K resolution evaluation although being optimized for 4K resolution.

4.3.4 Analysis and discussion

In this section, objective results have demonstrated that the VTM-11 codec enables 31%, 26%, and 35% of bitrate saving over the HM-16.20 codec for PSNR, MS-SSIM, and VMAF quality metrics, respectively. On the subjective side, VVC offers around 41% of bitrate reduction



(a) PSNR (b) SSIM

(c) MS-SSIM (d) VMAF

Figure 4.8 – Scatter plots and nonlinear logistic fitted curves of PSNR, SSIM, MS-SSIM and VMAF quality metrics versus DMOS scores of the considered 8K video sequences.

over HEVC for the same visual quality regarding the BD-BR method. Regarding the Student's t-test results, a bitrate reduction of about 50% is reached for two of the overall tested scenes. We have also demonstrated that the bitrate required to obtain transparency with the 8K source is highly content-dependent. Indeed, for VVC, a bitrate from 11Mbps to 180Mbps is needed, depending on the scene's complexity. In addition, we demonstrated that the participants had noted a difference between uncompressed 4K and 8K for most of the tested sequences. However, high-motion video scenes do not benefit from the 8K definition at 60fps. Finally, a higher correlation consistency between subjective and objective results can be noticed, particularly for the VMAF and MS-SSIM quality metrics.

Table 4.9 – SROCC, PLCC, KROCC and RMSE performance of the objective quality metrics MS-SSIM, SSIM, VMAF and PSNR on the considered 8K video sequences.

Objective metric	SROCC	PLCC	KROCC	RMSE
MS-SSIM	**0.887**	0.871	**0.725**	7.409
SSIM	0.767	0.777	0.599	9.499
VMAF	0.806	**0.873**	0.603	**7.375**
PSNR	0.754	0.747	0.564	10.042

4.4 8K video delivery with 4K backward-compatibility

In the previous section, we have shown that VVC can reduce the bitrate cost of 8K of around 41% regarding the same perceived quality compared to HEVC. Although this bandwidth reduction would make the delivery of 8K services easier, backward compatibility with UHD-1 receivers is not considered. This section assesses the performance of different coding approaches that allow the delivery of an 8K video signal with 4K backward compatibility. Presented approaches include:

— simulcast of 8K and 4K single-layer signals encoded using both HEVC and VVC standards,

— spatial scalability using SHVC with a 4K BL and an 8K EL,

— spatial upscaling applied on a 4K decoded signal using both HEVC and VVC standards. We evaluate both a lightweight version of EDSR [15] and the Lanczos [14] filter.

The remainder of this section is organized as follows. Section 4.4.1 presents the different assessed backward compatible approaches. Section 4.4.2 gives the test conditions, including the coding standards and super-resolution settings. Results are then presented and analyzed in Section 6.4.2. Finally, Section 6.4.3 concludes this section.

4.4.1 Tested approaches

Simulcast

Simulcast is the process of transmitting several versions of an input signal encoded with single-layer coding approaches, e.g. HEVC or VVC, to cover different target outputs. The principle of simulcast is illustrated in Figure 4.9. In this coding scheme, each signal is encoded independently without considering any correlation between the different resolutions. Simulcast allows defining the lower bound of the performance of scalability. Although being easy to set up

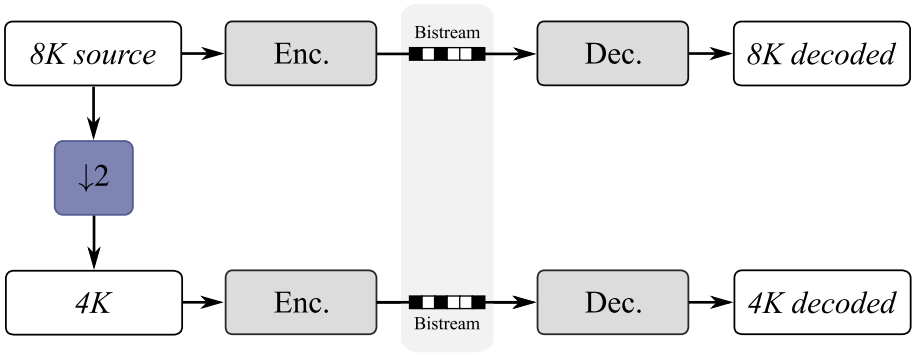


Figure 4.9 – Illustration of simulcast for 8K and 4K video delivery.

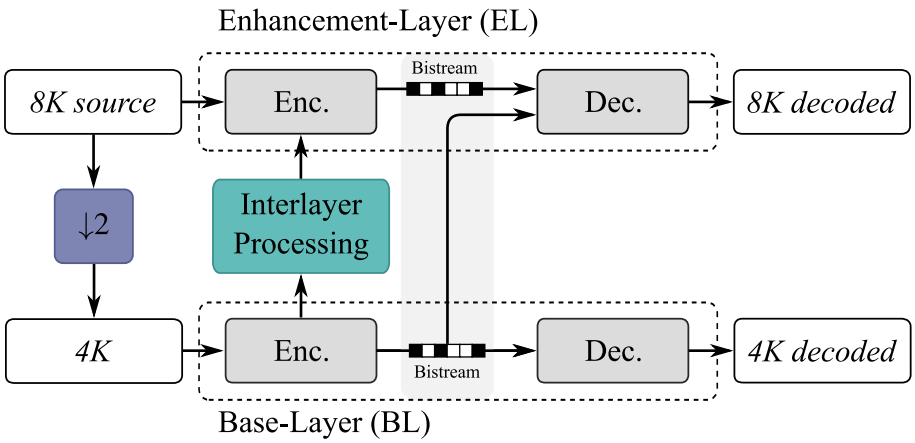


Figure 4.10 – Illustration of spatial scalability for 8K and 4K video delivery.

because of the independence of the selected codec, this approach results in a high bitrate.

Spatial scalability

Using a spatially-scalable codec might increase the efficiency by taking advantage of the existing correlations between the 4K and 8K signals. This solution is described in Figure 4.10. In the case of SHVC for spatial scalability, a BL signal (low resolution) encoded with HEVC is used as a reference by an inter-layer processing module to encode the EL signal (high resolution). The EL signal is described by using additional HLS and needs a scalable-compliant decoder to be decoded. Several standardization bodies such as the advanced television systems committee (ATSC) [182] or the DVB [30] consider SHVC as a candidate for solving compatibility issues brought by new formats introduction. However, due to a late integration in HEVC and additional complexity and latency brought by inter-layer processing, spatial scalability is not much present in the current broadcast ecosystem.

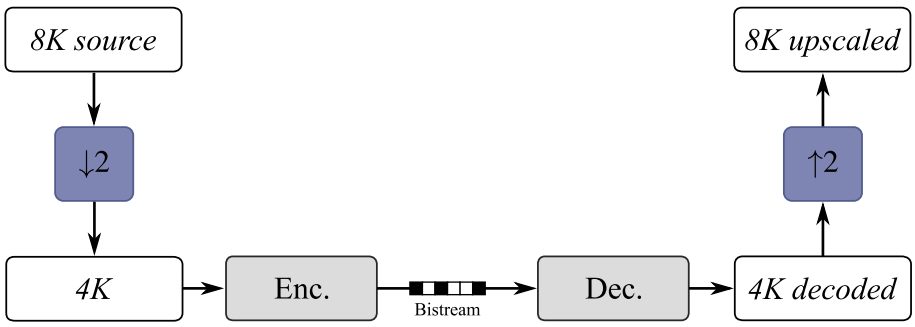


Figure 4.11 – Illustration of the downscaling/upscaling solution for 8K and 4K video delivery.

Spatial upscaling

Applying downscaling and upscaling operations to the signal outside the coding pipeline allows covering a wide range of compatible UHD-1 or UHD-2 receivers without additional bitstream. Thus, the bandwidth is limited to 4K only, while the receiver can display both resolutions. Also, this solution allows different resolutions to be provided without being dependent on the base codec. Figure 4.11 illustrates this coding configuration. In the image processing field, the process of estimating a HR version of a LR content is referred to as super-resolution. In the last past few years, learning-based super-resolution approaches have outperformed state-of-the-art methods through the last progress in the AI field. The objective of these methods is to learn the non-linearity between LR images and their HR version by analyzing local statistics of images. For our study, we have used a lightweight version of EDSR described in [15] for super-resolution. This approach is based on an end-to-end CNN optimized to recover details from LR images by minimizing the $L1$ loss between the reconstructed HR images and their corresponding HR ground truth. In this experiment, we evaluate this approach based on both HEVC and VVC. The models are trained using the 4K sequences from the BVI-DVC dataset [183]. More details on the training process are provided in Section 4.4.2

4.4.2 Experimental settings

For this study, the CTCs for VTM-11 [179], HM-16.20 [72], and SHVC test model (SHM-9.0) [184] are used to provide a fair rate-distortion assessment. For spatial scalability evaluation, we compare SHVC with HEVC simulcast encoded with the SHM-9.0 BL mode, denoted as HEVC*. All coding configurations are summarized in Table 4.10. The 4K sequences are generated by a Lanczos downscale for simulcast and spatial upscaling approaches. The SHVC EL model being sensitive to the downscaling operator used to create the BL, we used the

Table 4.10 – Standard verification models specifications.

Standard	Reference Software	Cfg	Profile	GOP Size	Intra Period
VVC	VTM-11.0	RA	main10	16	64
HEVC	HM-16.20	RA	main10	16	64
HEVC*	SHM-9.0 (BL)	RA	main10	16	64
SHVC	SHM-9.0	RA	main10	16	64

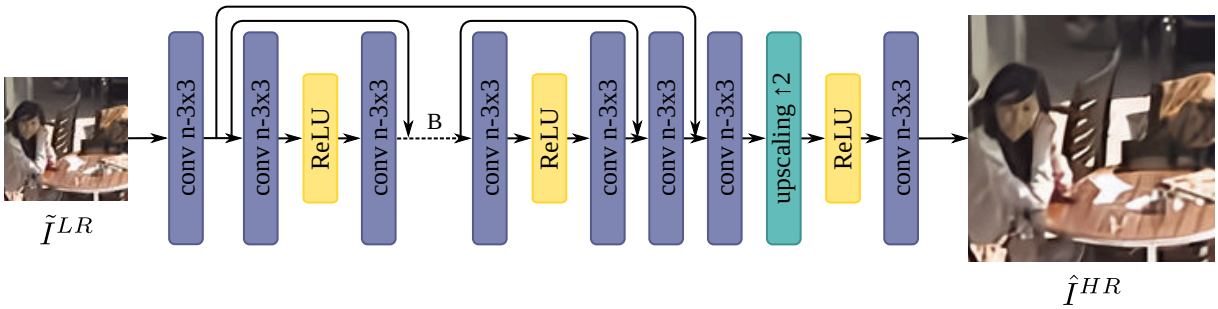


Figure 4.12 – Architecture of EDSR [15].

downscaling filter described in [9] to generate the 4K BL sequences. All 8K videos are encoded with QP values of 27, 32, 37, and 42. For spatial upscaling, the 4K sequences are encoded with QP values of 22, 27, 32, and 37 to cover a similar bitrate range. We used objective metrics, including PSNR, MS-SSIM, and VMAF, to measure the distortion between the reconstructed 8K signal and the original one. We used the BD method described in [44] to quantify the average bitrate gain between configurations. The assessed bitrate corresponds to both the 8K and 4K rates for simulcast and spatial scalability. For spatial upscaling, we assessed the bandwidth on 4K only.

The super-resolution network EDSR, provided by the authors in [15], is trained to recover the HR version of uncompressed LR data. The architecture of this CNN is illustrated in Figure 4.12. In our case, we focus on assessing this method on video compressed using both HEVC and VVC. As learning-based super-resolution is sensitive to the training data, we trained the network using a compressed version of the LR images. For this study, we selected 200 clips from the BVI-DVC dataset [183] which contains 4K 10bits videos. First, we generated pairs of LR/HR videos by applying a Lanczos downscaling filter with a scale factor of 2. Then, each LR video clip has been encoded using the RA configuration of the VTM-11 with four QP values, including 22, 27, 32, and 37, to cover a large panel of distortions. After decoding, we generated YUV4:4:4 tensors by duplicating the chroma components for both reconstructed and original images. Finally, we

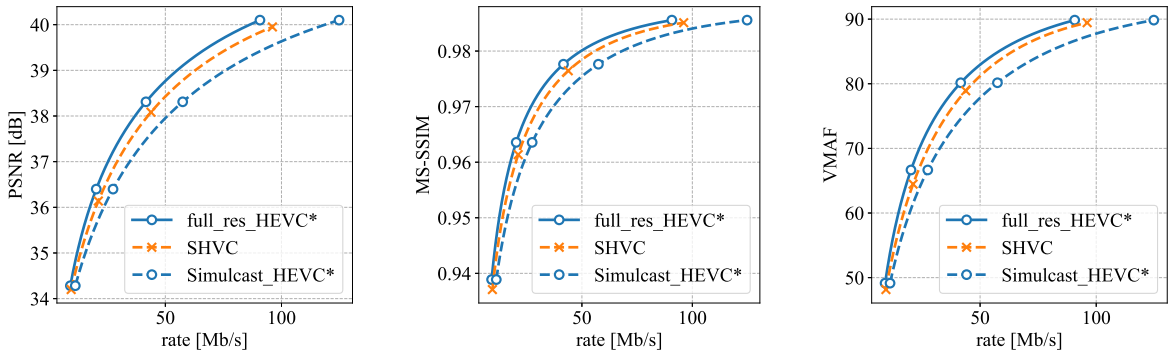


Figure 4.13 – Average RD curves over the selected 8K video sequences (SHVC).

extracted all the reconstructed and original frames and cropped them into patches of size 48×48, denoted as $\tilde{I}^{LR}$ and I^{HR} , respectively. For super-resolution, We used a lightweight version of EDSR for our study, denoted as EDSR*, by setting $n = 64$ and $B = 16$. To evaluate the performance of EDSR*, we also compared results obtained with a Lanczos filter applied on the LR sequences.

4.4.3 Experimental results

In this section, we present and analyze the results of the experiments described in Section 4.4.2. First, we compare simulcast and spatial scalability using SHM-9.0. Then, we evaluate the spatial upscaling approach over 4K/8K simulcast and 8K full-resolution coding based on both VTM-11 and HM-16.20.

Spatial scalability

In this experiment we assess spatial scalability with a 4K BL signal and an 8K EL signal over simulcast. Figure 4.13 illustrates the RD curves for each objective quality metric. The curves represented in dashed lines correspond to the configurations enabling 8K delivery with 4K backward compatibility. Table 4.11 gives the BD-rate results of SHVC with respect to the HEVC simulcast configuration for the six 8K video sequences. Inter-layer predictions enable -18.43%, -18.58%, and -18.47% of average BD-rate savings over simulcast for PSNR, MS-SSIM, and VMAF, respectively. We see that spatial scalability performs better for the sequence with the less 8K (HR) information regarding the results in Section 4.3. Indeed, for those sequences the 8K and 4K resolutions are highly correlated which is exploited by the inter-layer processing

Table 4.11 – BD-rate (%) for SHVC compared to HEVC* simulcast. The values in bracket indicate the BD-rate assessed with HEVC* 8K coding as anchor.

	PSNR	MS-SSIM	VMAF
LayeredKimono	-15,77 [+25,67]	-16,12 [+22,44]	-16,53 [+23,26]
BodeMuseum	-16,91 [+20,86]	-17,85 [+20,13]	-18,64 [+18,72]
OberbaumSpree	-21,09 [+8,84]	-21,16 [+8,43]	-21,40 [+8,31]
Festival2	-22,39 [+5,49]	-21,49 [+4.00]	-21,47 [+6,08]
JapaneseMaple	-17,10 [+11,48]	-18,21 [+9,90]	-17,94 [+10,32]
SteelPlant	-17,29 [+10,80]	-16,64 [+10,24]	-14,88 [+13,20]
Average	**-18,43 [+13,86]**	**-18,58 [+12,52]**	**-18,47 [+13,32]**

module. However, the performance of spatial scalability worst than HEVC full-resolution coding on the whole range of bitrate and for all sequences.

Spatial upscaling

This experiment evaluates the spatial upscaling approach over simulcast based on both VVC and HEVC standards. We also compare the performance of 8K full-resolution coding to assess the cost of backward compatibility. A rate-distortion evaluation has been conducted using objective visual quality metrics, including PSNR, MS-SSIM, and VMAF. The average performance based on HM-16.20 and VTM-11 overall the tested sequences is illustrated in Figure 4.14 and Figure 4.15, respectively. The per-sequence interpolated RD curves are depicted in Figures 4.16, 4.17, 4.18, 4.19, 4.20, and 4.21. BD-rate performance compared to both 8K full-resolution coding and 4K/8K simulcast are represented in Table 4.12.

First, we can notice that EDSR* proposes better average performance than the Lanczos filter for HM-16.20 and VTM-11 regarding all objective quality metrics. The results show that the reconstruction gain is higher based on HEVC than VVC. In addition, the performance gap between those two methods is lower regarding MS-SSIM than the two other metrics. The results also demonstrate that the gain offered by EDSR* is content-dependent. For the sequences *Festival2* and *OberbaumSpree*, a negligible gain is assessed for EDSR* compared to Lanczos. As mentioned in the analysis of the subjective study conducted in Section 6.3.2, those two scenes contain few high-resolution details. Generally, the more spatial information the 8K source contains, the more the EDSR* super-resolution network outperforms the Lanczos filter. Concerning the compression level, the RD-curves show that the performance gap between the

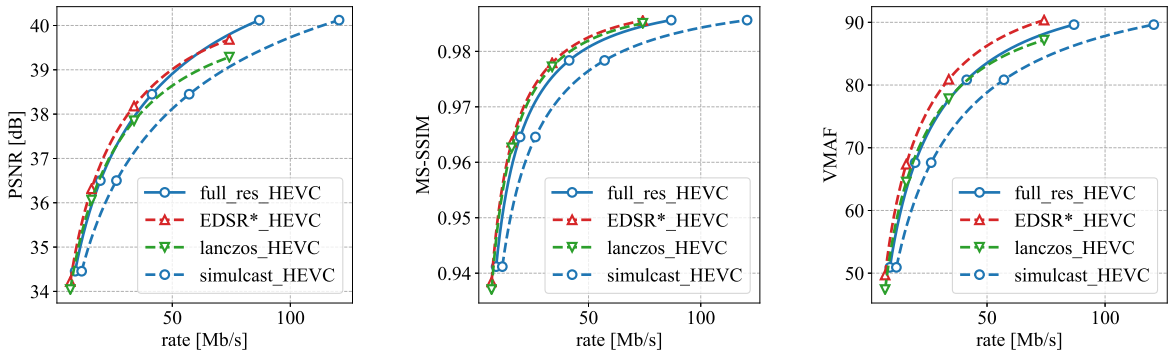


Figure 4.14 – Average RD curves over the selected 8K video sequences (HEVC).

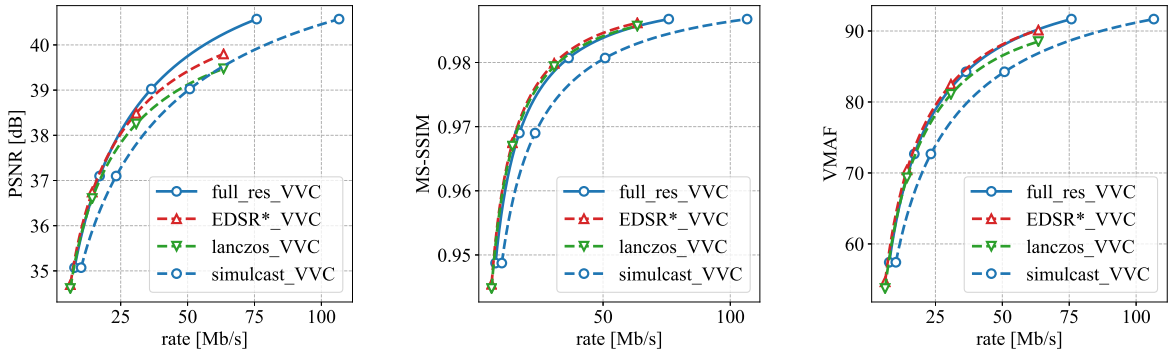


Figure 4.15 – Average RD curves over the selected 8K video sequences (VVC).

EDSR* network and Lanczos increases proportionally with the bitrate increase.

Compared to simulcast, spatial upscaling offers BD-rate gains on the overall dataset and for all objective quality metrics. Regarding EDSR* based on HM-16.20, average bitrate reduction of -34.05%, -38.06%, and -40.40% over HEVC simulcast are assessed for the same level of PSNR, MS-SSIM, and VMAF, respectively. Those gains are lower in the VVC context with -26.32%, -34.23%, and -30.38% of BD-rate gain regarding PSNR, MS-SSIM, and VMAF, respectively. Generally, we notice that spatial upscaling based on VTM-11 is less performant than on HM-16.20 compared to simulcast and full-resolution. It can be explained by the introduction of new coding modes into VVC, including new partitioning options and larger CTUs, which improve the 8K single-layer coding.

Compared to 8K full-resolution coding with HEVC, spatial upscaling provides bitrate gains of -9.33% for the same level of PSNR while offering backward compatibility with 4K receivers. Concerning VVC, a loss of 2.37% of bitrate is observed for the same PSNR. Regarding MS-SSIM and VMAF, spatial upscaling is preferred from both full-resolution coding and simulcast in the

Table 4.12 – BD-rate (%) of spatial upscaling compared to simulcast regarding HEVC and VVC. The values in bracket indicate the BD-rate assessed with full-resolution coding as anchor.

Codec	HM-16.20					
Upscaling method	Lanczos			EDSR*		
Metric	PSNR	MS-SSIM	VMAF	PSNR	MS-SSIM	VMAF
LayeredKimono	-27.43 [+2.49]	-39.02 [-14.06]	-30.25 [-1.59]	-35.57 [-8.99]	-40.66 [-16.37]	-42.30 [-18.66]
BodeMuseum	-17.87 [+13.96]	-31.25 [-4.77]	-24.24 [+5.02]	-30.42 [-3.55]	-36.42 [-11.98]	-40.24 [-17.32]
OberbaumSpree	-34.11 [-11.21]	-35.61 [-13.64]	-34.18 [-11.48]	-36.54 [-14.50]	-38.03 [-16.89]	-43.51 [-24.00]
Festival2	-35.25 [-6.68]	-40.33 [-14.25]	-36.53 [-8.56]	-39.89 [-13.35]	-43.54 [-18.86]	-42.74 [-17.50]
JapaneseMaple	-18.24 [+9.18]	-31.44 [-8.91]	-25.11 [+0.01]	-29.25 [-5.24]	-33.96 [-12.22]	-37.12 [-15.87]
SteelPlant	-22.04 [+3.12]	-32.86 [-11.63]	-23.47 [+1.09]	-32.62 [-10.36]	-35.76 [-15.39]	-36.51 [-15.73]
Average	**-25.82 [+1.81]**	**-35.09 [-11.21]**	**-28.96 [-2.59]**	**-34.05 [-9.33]**	**-38.06 [-15.29]**	**-40.40 [-18.18]**
Codec	VTM-11					
Upscaling method	Lanczos			EDSR*		
	PSNR	MS-SSIM	VMAF	PSNR	MS-SSIM	VMAF
LayeredKimono	-22.68 [+10.61]	-36.65 [-9.37]	-23.95 [+8.74]	-25.80 [+6.20]	-37.76 [-10.96]	-29.94 [+0.24]
BodeMuseum	-13.27 [+21.61]	-28.86 [-0.36]	-21.64 [+9.80]	-24.01 [+6.56]	-32.80 [-5.91]	-28.04 [+0.80]
OberbaumSpree	-31.48 [-6.22]	-32.76 [-8.42]	-32.43 [-7.71]	-29.14 [-3.10]	-34.08 [-10.20]	-34.08 [-9.96]
Festival2	-31.20 [+0.02]	-36.57 [-8.02]	-34.54 [-4.88]	-32.37 [-1.68]	-37.97 [-10.04]	-36.78 [-8.12]
JapaneseMaple	-8.21 [+23.71]	-27.28 [-2.34]	-18.68 [+9.67]	-20.56 [+7.44]	-29.82 [-5.70]	-26.38 [-0.53]
SteelPlant	-16.04 [+11.42]	-30.79 [-8.25]	-15.93 [+11.65]	-26.01 [-1.19]	-32.96 [-11.05]	-27.08 [-2.74]
Average	**-20.48 [+10.19]**	**-32.15 [-6.12]**	**-24.53 [+4.54]**	**-26.31 [+2.37]**	**-34.23 [-8.98]**	**-30.38 [-3.39]**

whole range of selected bitrates. It can be explained by the pyramidal quality computation aspect of those two metrics. Indeed, MS-SSIM and VMAF (with the contribution of the VIF quality metric [48] in the final score computation) assess the quality based on several spatial versions of the input image. As the 4K stream is encoded using a lower QP for the spatial upscaling configuration, the quality is preferred from 8K single-layer coding. We can notice that the performance of spatial upscaling compared to both simulcast and full-resolution coding are also content-dependent. For the sequences with a low amount of high-resolution details, i.e., *Festival2* and *OberbaumSpree*, 4K spatial upscaling performance is on-par with 8K full resolution coding. The EDSR* super-resolution network proposes the best performance compared to all tested configurations for the most complex video sequences (JapaneseMaple and SteelPlant).

4.4.4 Analysis and discussion

In this study, three approaches allowing the transmission of both 8K and 4K signals have been assessed, including simulcast, spatial scalability, and spatial upscaling. First, an experiment evaluating HEVC and its scalable extension SHVC was conducted. Experimental results have shown that spatial scalability achieves -18.43% BD-rate savings compared to simulcast regarding PSNR. However, the performance of spatial scalability is always worst than HEVC full-resolution coding, with an average bitrate overhead of around +14%. Moreover, scalable-compliant decoders will still be required to decode the EL bitstream.

Then, we have demonstrated that the super-resolution network proposes better performance than the Lanczos filter for all sequences encoded using HM-16.20 and VTM-11. We have also shown that spatial upscaling offers BD-rate gains over simulcast on the overall dataset regarding both codecs. Moreover, we demonstrated that 4K spatial upscaling proposes better performance than 8K full-resolution coding for some sequences, especially at low bitrate, while providing both 8K and 4K resolution at the receiver side. However, EDSR* performance is close to the Lanczos filter in this bitrate range due to degradations introduced in the LR signal. Future works will consider subjective evaluation to consolidate the objective results.

4.5 Conclusion

Several compression algorithms have been assessed for 8K resolution video in this chapter. The tested approaches include the single-layer coding standards HEVC and VVC and scalable methods, namely simulcast, spatial scalability with SHVC and spatial upscaling using learning-based super-resolution and a Lanczos filter.

In the first study, we have demonstrated that VVC offers around 41% bitrate reduction over HEVC for the same visual quality. In addition, we showed that the participants had noted a difference between uncompressed 4K and 8K for most of the tested sequences. Then, we have deducted that super-resolution is a good candidate for 8K delivery from 4K stream. It allows a codec agnostic reconstruction of 8K resolution from 4K without any side information. In addition, it outperforms simulcast and proposes performance close to 8K full resolution coding for some of the tested sequences. Altough being trained using a PSNR-related loss function, subjective assessment of AI-based super-resolution algorithms might be considered to consolidate those observations.

Nevertheless, several tracks remain for improving CNN-based super-resolution applied on

compressed video contents. First, we have seen that spatial upscaling proposes better performance at a low bitrate. However, the super-resolution and the Lanczos filter performance converge as the bitrate decreases. Two hypotheses can be formulated:

— Coding artifacts introduced during the compression process have a high impact on the super-resolution network.

— High-frequencies lost during downscaling and quantization cannot be recovered without side information.

Moreover, the performance of spatial upscaling using super-resolution compared to simulcast and full resolution coding is content-dependent. Those items are investigated in the next two chapters.

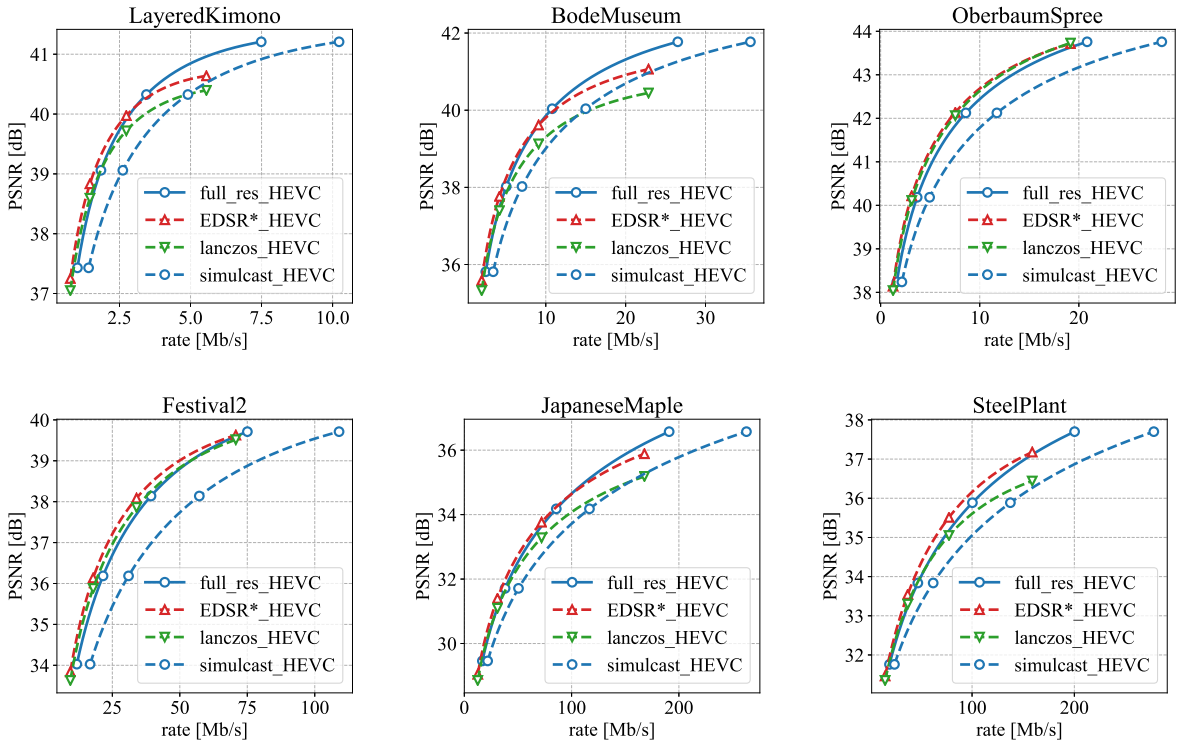


Figure 4.16 – PSNR-based comparison for the selected 8K video sequences (HEVC).

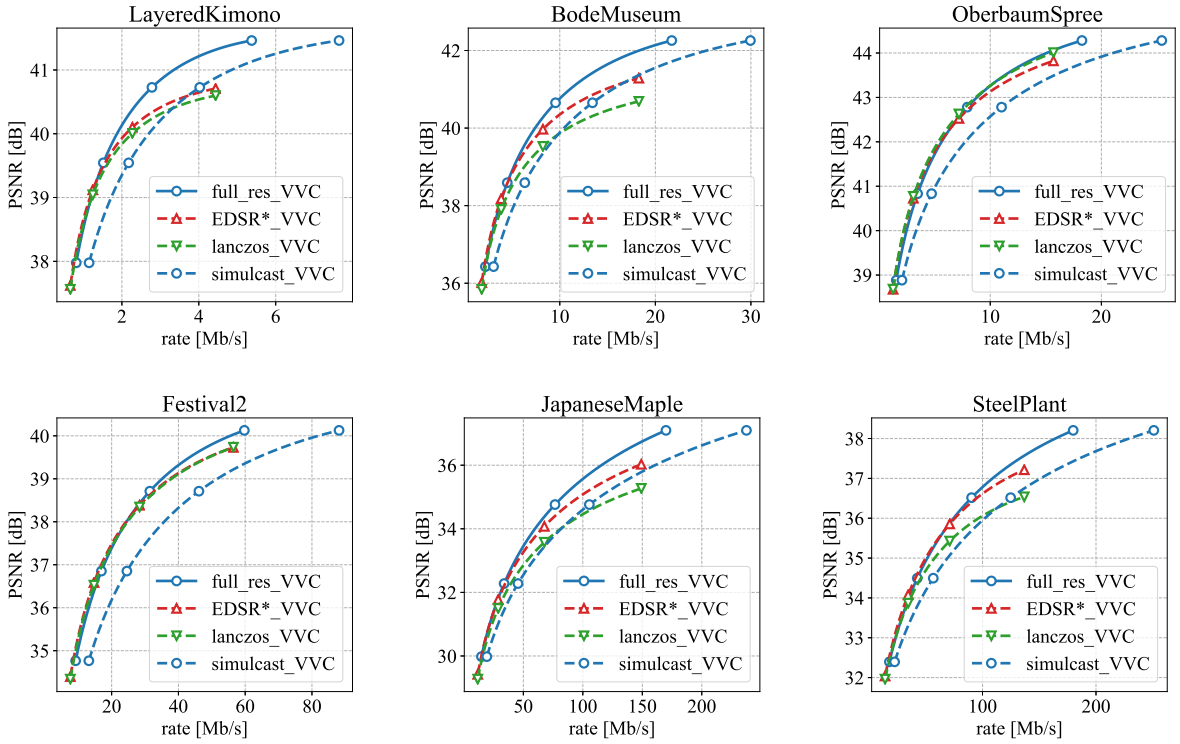


Figure 4.17 – PSNR-based comparison for the selected 8K video sequences (VVC).

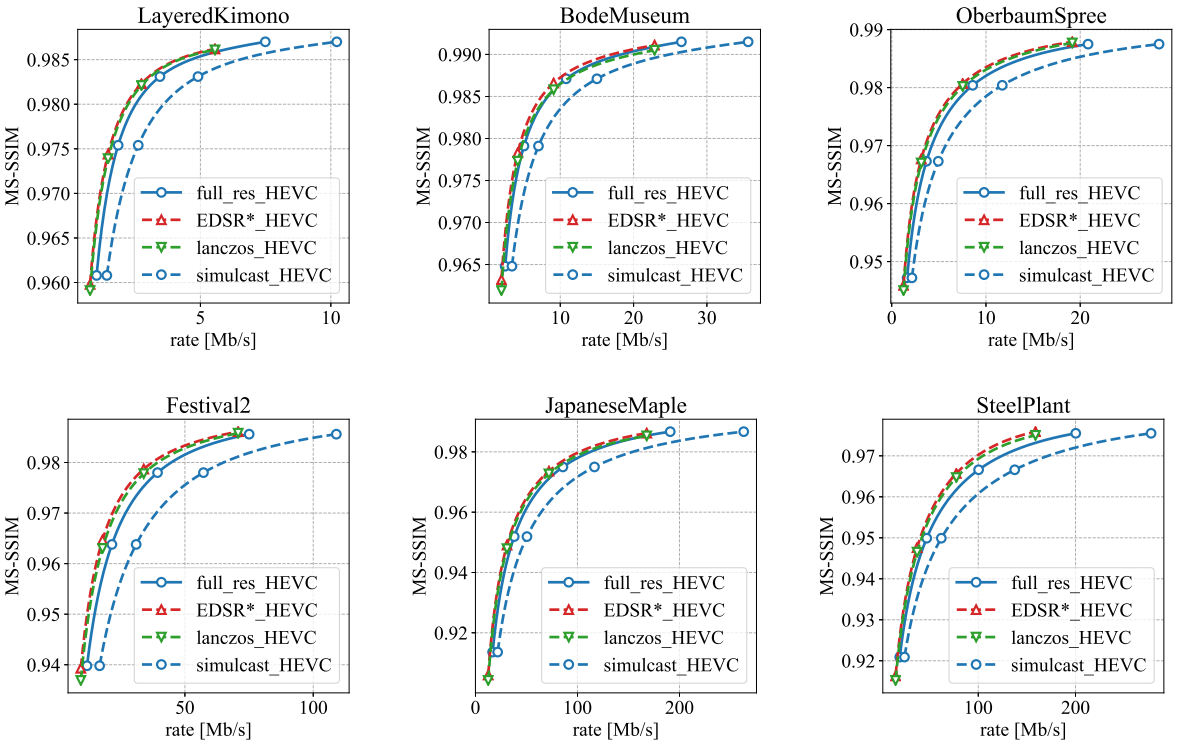


Figure 4.18 – MS-SSIM-based comparison for the selected 8K video sequences (HEVC).

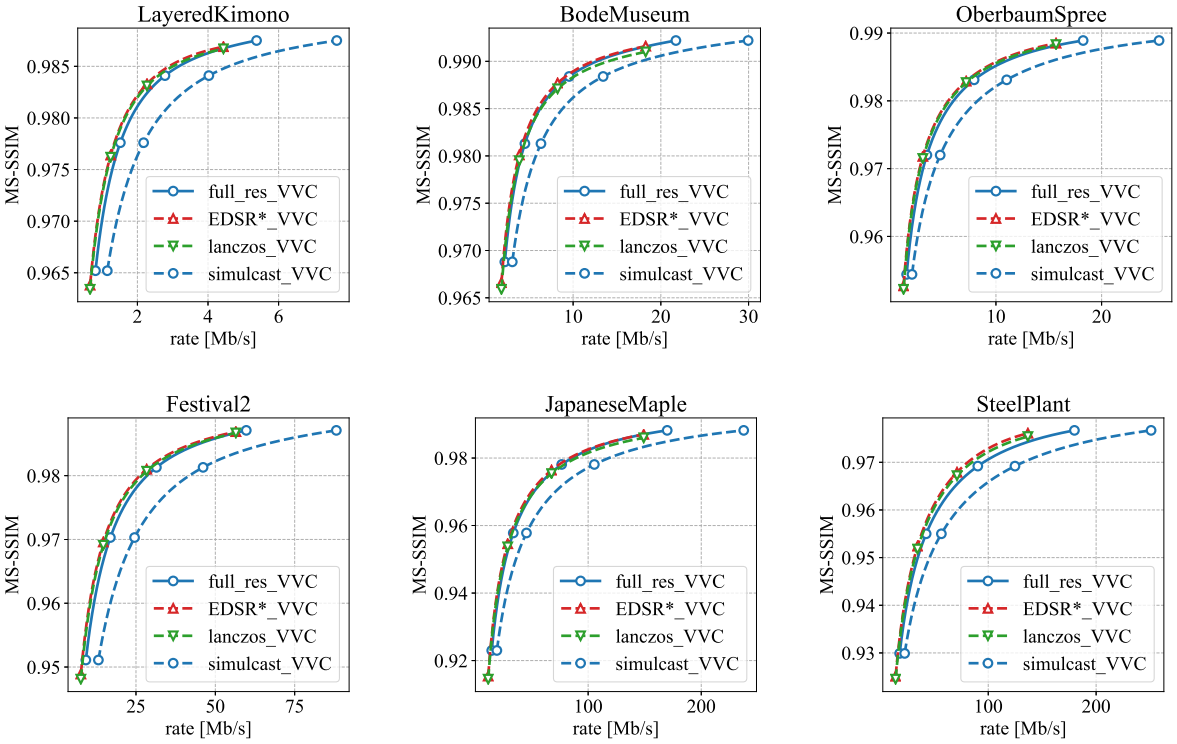


Figure 4.19 – MS-SSIM-based comparison for the selected 8K video sequences (VVC).

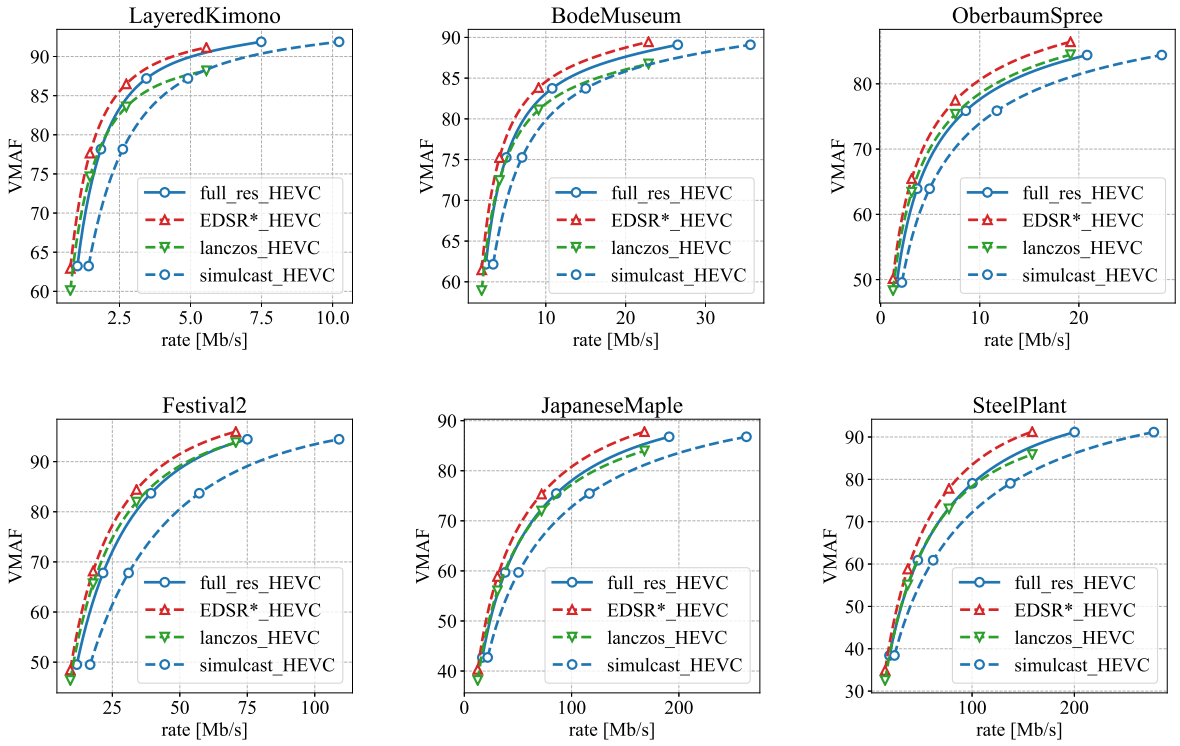


Figure 4.20 – VMAF-based comparison for the selected 8K video sequences (HEVC).

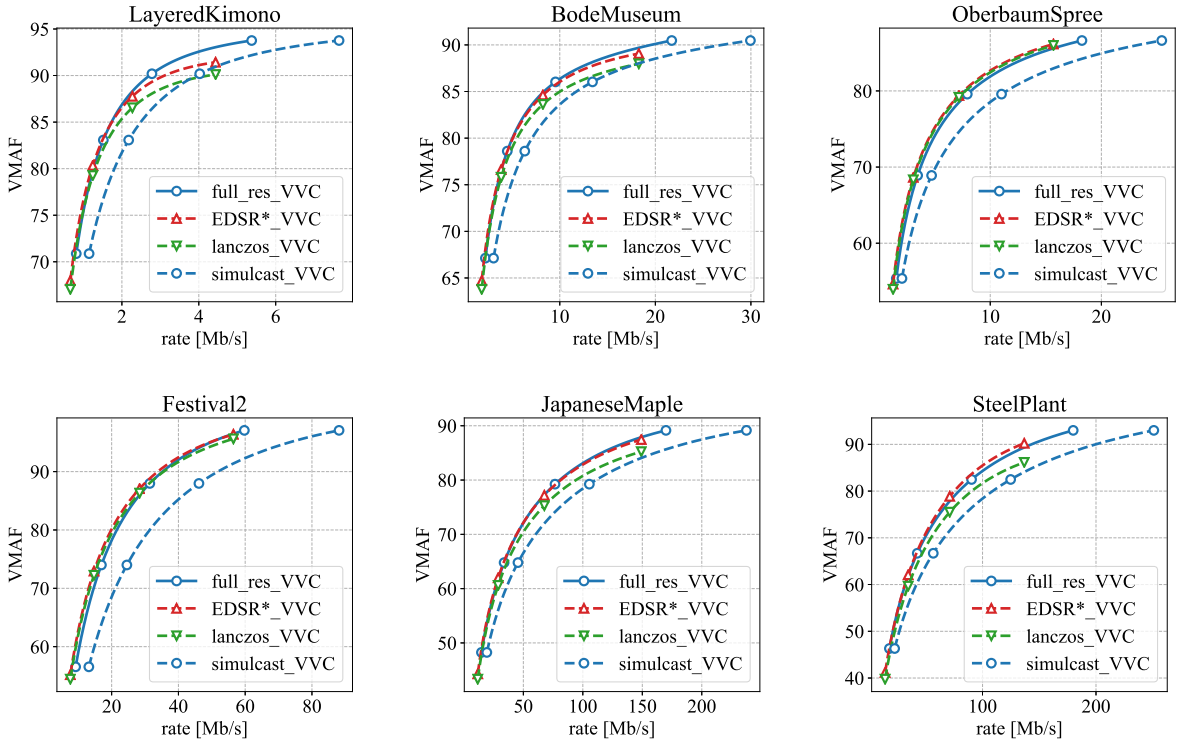


Figure 4.21 – VMAF-based comparison for the selected 8K video sequences (VVC).

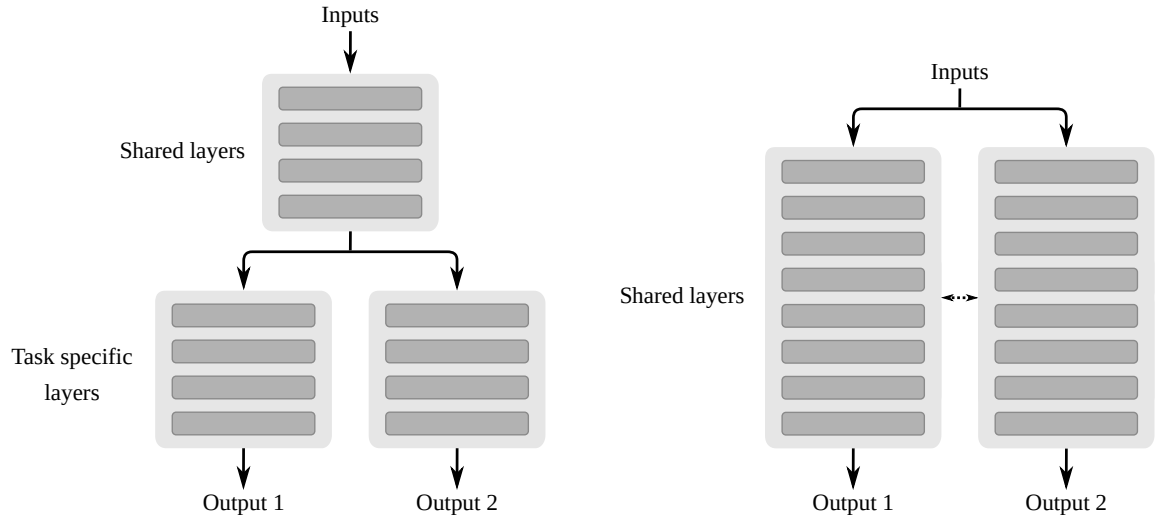
CHAPTER 5

MULTITASK LEARNING FOR SUPER-RESOLUTION OF COMPRESSED VIDEOS

5.1 Preamble

In the previous chapter, we evaluated several compression algorithms for 8K video delivery. We subjectively assessed 8K uncompressed video contents and concluded that regular viewers perceive the difference between 8K and 4K resolutions. We also demonstrated that spatial upscaling using super-resolution is suitable for ensuring backward compatibility with UHD-1 receivers. However, learning-based super-resolution offers performance similar to a Lanczos filter at low bitrate due to spatial information loss generated by quantization. Previous works proposed using perceptual loss and GANs [185, 186] to hallucinate details in the upscaled compressed image. However, those algorithms are hard to control and suffer from source signal fidelity loss, which is unpractical in a broadcast context. Thus, dedicated models based on pixel-wise loss must be considered to improve those algorithms' performance without affecting the signal's fidelity. This chapter explores multitask-based architectures for super-resolution on compressed contents. This process allows performing multiple tasks with a single shared network, reducing the total number of parameters. Advanced training strategies, such as prior information using qp_{map} and network pre-training, are also investigated to improve the performance of the network on compressed LR inputs.

This chapter is organized as follows. First, Section 5.2 presents the principle of multitask learning and different possible architectures. Then, Section 5.3 introduces MTL-EDSR, a multitask network that performs super-resolution and quality enhancement using a single shared network. Section 5.4 presents MTL-Unet, an extension of MTL-EDSR dedicated to super-resolution and high-level vision tasks, namely, no-reference image quality assessment (NR-IQA) and semantic segmentation. Finally, Section 6.5 concludes this chapter.



(a) Hard parameter sharing (b) Soft parameter sharing

Figure 5.1 – Parameter sharing.

5.2 Multitask learning

Recently, DNNs based on multitask learning (MTL) [16] have been proposed to perform multiple tasks using a single shared network. Therefore, all the tasks can access the learned representations to exploit redundant features and improve performance. Compared to single-task learning, training a model to perform multiple tasks simultaneously poses two key challenges: sharing the parameters and balancing the tasks, i.e., computing the multitask loss.

5.2.1 Multitask loss

Selecting the appropriate loss function is an essential step for training a DNN. In the context of multitask learning, the way the task-related losses are combined is crucial to design such a model. By considering a multitask neural network that performs two tasks, the most straightforward way to compute the multitask loss $\mathcal{L}_{mtl}$ is by summing the two task-related losses, denoted as $\mathcal{L}_1$ and $\mathcal{L}_2$, weighted by a parameter α:

$$\mathcal{L}_{weighted} = \alpha\,\mathcal{L}_1 + (1-\alpha)\,\mathcal{L}_2. \tag{5.1}$$

However, wrong tuning of the weight parameter would make the easier task dominant. Moreover, manually searching for the best α parameter is hard in practice and time-consuming. Thus, some solutions have been proposed to automatically compute the multitask loss $\mathcal{L}_{mtl}$ during training [187, 188].

In [187], Kendall *et al.* proposed to learn the task-related weights directly during training by considering the uncertainty of each task. Thus, both task-related losses $\mathcal{L}_1$ and $\mathcal{L}_2$ are weighted by a learned uncertainty term, denoted as σ_i with $i \in (1, 2)$. Those weigthing terms are computed by deriving a multi-task loss function based on maximizing the Gaussian likelihood with task-dependent uncertainty. The multitask loss function can be expressed as follows:

$$\mathcal{L}_{uncertainty} = \frac{1}{2\sigma_1^2}\mathcal{L}_1 + \frac{1}{2\sigma_2^2}\mathcal{L}_2 + \log\sigma_1 + \log\sigma_2. \tag{5.2}$$

The authors have successfully applied this approach to a multitask model that performs three tasks: per-pixel depth regression, semantic and instance segmentation.

Another solution called dynamic weight average (DWA) have been developed in [188]. This approach consists on weighting the tasks specific losses regarding their rate of change for each task. By considering $k \in (1, 2)$, The final multitask loss $\mathcal{L}_{DWA}$ can be expressed as follows:

$$\mathcal{L}_{DWA} = \sum_k \lambda_k \mathcal{L}_k. \tag{5.3}$$

The weighting parameters λ_k are computed regarding the loss values of the previous iterations as:

$$\lambda_k(t) = \frac{K\exp(\omega_k(t-1)/T)}{\sum_i \exp(\omega_i(i-1)/T)}, \tag{5.4}$$

with T is the temperature, t an iteration index, k the task index and ω_k the relative descending rate of the task k computed as:

$$\omega_k(t-1) = \frac{\mathcal{L}_k(t-1)}{\mathcal{L}_k(t-2)}. \tag{5.5}$$

Here, the temperature T is a parameter used to control the softness of task weighting [189]. This parameter allows to balance the distribution across the different tasks. A large enough value of T results in an equal weighting of the tasks, as $\lambda_1 \approx 1$.

5.2.2 Parameter sharing

When designing a multitask model, the second challenge is to share the learned representations across the tasks. The sharing of parameters can be performed by either hard parameter sharing or soft parameter sharing.

By considering hard parameter sharing, the representations learned by the multitask network

are either totally shared between tasks or task-specific. Multitask architectures based on this principle are composed of a "trunk", i.e., shared layers, and multiple task-specific "branches", i.e., task-specific layers. An illustration is provided in Fig 5.1 (a). In a context where the tasks are close, the learned representations can globally be mutualized without interfering. However, bypassing or weighting some unrelated feature maps might be helpful for specific tasks to learn correctly.

The process of allowing the multitask model to partially share representations across the tasks is called soft parameter sharing. This principle is illustrated in Fig 5.1 (b). In [22], authors developed a soft sharing architecture based on cross-stitch units applied between multiple task-specific architectures. In this approach, the cross-stitch units learn the optimal linear combinations between the layers of several task-specific networks that minimize the multitask loss function. Given two activation maps x_a and x_b, and a cross-stitch unit composed of parameters α, the output maps of the cross stitch unit $\tilde{x}_A$ and $\tilde{x}_B$ are computed at pixel position (i, j) as follows:

$$\begin{bmatrix} \tilde{x}_A^{ij} \\ \tilde{x}_B^{ij} \end{bmatrix} = \begin{bmatrix} \alpha_{AA} & \alpha_{AB} \\ \alpha_{BA} & \alpha_{BB} \end{bmatrix} \begin{bmatrix} x_A^{ij} \\ x_B^{ij}. \end{bmatrix} \tag{5.6}$$

Thus, the multitask network can dynamically set the layers to shared or task-specific by varying alpha to zero from one during training.

5.3 Quality enhancement

In this section we present MTL-EDSR, a learning-based post-processing solution dedicated to images and videos encoded with VVC AI mode. Our method relies on multitask learning and performs quality enhancement and super-resolution using a single shared network optimized for multiple degradation levels. The proposed solution enables a good performance in mitigating coding artifacts and super-resolution with fewer network parameters than traditional specialized architectures. We investigate advanced multitask architecture such as automatic multitask loss computation and soft parameter sharing.

5.3.1 Proposed solution

Our approach aims to exploit the similarity between two tasks: SR and QE. We propose to use hard parameter sharing using a shared network f_θ and two task-specific modules $g_{\phi_{SR}}$ and

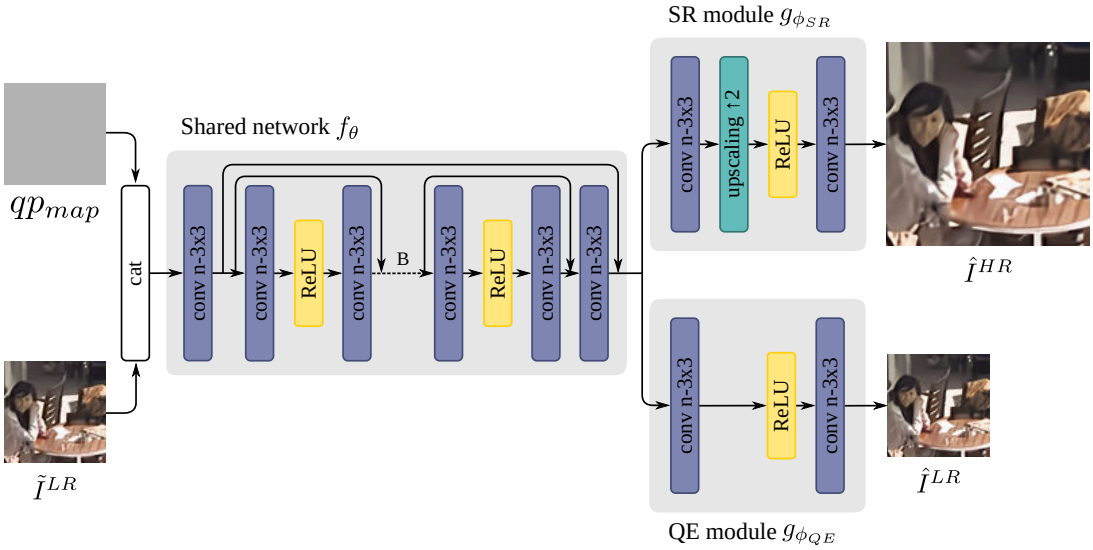


Figure 5.2 – Architecture of the proposed MTL-EDSR network.

$g_{\phi_{QE}}$. This approach allows the model to fully benefit from the feature redundancy between both tasks. Consequently, the number of parameters can be reduced while maintaining a good reconstruction quality compared to specialized architectures.

To make the model capable of generalizing across several input QP, we use qp_{map} [190] as prior information to the network. This prior input corresponds to a uniform normalized map computed as:

$$qp_{map}(i,j) = \frac{QP}{QP_{max}}, \quad i = 1, ..., W; \quad j = 1, ..., H, \tag{5.7}$$

with (i, j) are the vertical and horizontal pixel coordinates. The value of QP_{max} is equal to 63 in VVC.

In the following, let I^{LR} denotes a low-resolution image of size $W \times H$ and $\tilde{I}^{LR}$ its reconstructed version that may include coding artifacts. We first extract the shared features y from the input image $\tilde{I}^{LR}$ concatenated with its corresponding qp_{map} using the shared network f_θ as follows:

$$y = f_\theta(\tilde{I}^{LR}, qp_{map}), \tag{5.8}$$

The output images $\hat{I}^{HR}$ and $\hat{I}^{LR}$ are then estimated from the shared features y using the

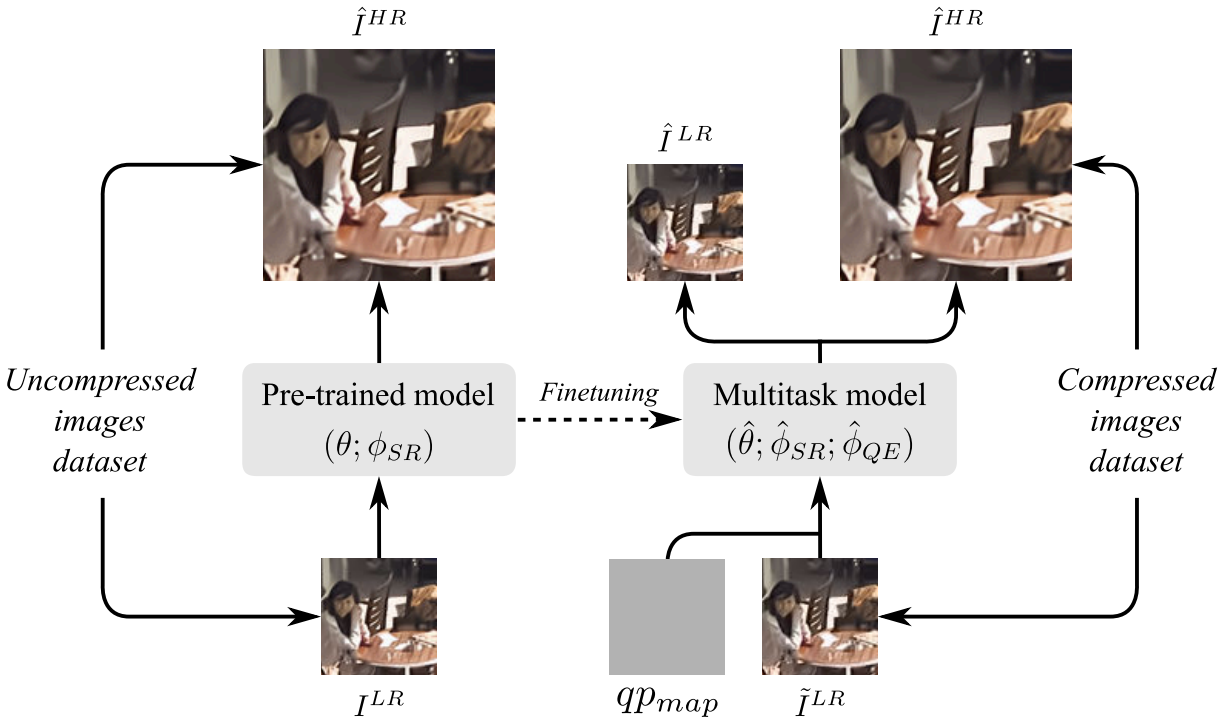


Figure 5.3 – Pre-training methodology.

task-specific modules $g_{\phi_{SR}}$ and $g_{\phi_{QE}}$ according to the following equations:

$$\hat{I}^{HR} = g_{\phi_{SR}}(y). \tag{5.9}$$

$$\hat{I}^{LR} = g_{\phi_{QE}}(y). \tag{5.10}$$

We selected L1-loss [191] to compute the task-specific losses $\mathcal{L}_{SR}$ and $\mathcal{L}_{QE}$ between the estimated images $\hat{I}^{HR}$ and $\hat{I}^{LR}$, and the original images I^{HR} and I^{LR}.

As our architecture is mainly inspired by [15], we first pre-train the network to perform super-resolution on uncompressed images. The pre-trained parameters $\hat{\theta}$ and $\hat{\phi}_{SR}$ are obtained by solving the following optimization problem:

$$(\hat{\theta}; \hat{\phi}_{SR}) = \underset{(\theta;\phi_{SR})}{\arg\min} \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}_{SR}(g_{\phi_{SR}}(f_{\theta}(I_n^{LR})), I_n^{HR}), \tag{5.11}$$

with I_n^{HR} the high-resolution training images, I_n^{LR} the corresponding low-resolution versions, N the number of training samples and $n = 1, ..., N$ the sample index. The pre-training process is illustrated in Fig 5.3.

Finally, we optimize the overall multitask network by combining both task-specific losses in

the multitask loss function $\mathcal{L}_{mtl}$ with a weighting parameter α as follows:

$$\mathcal{L}_{mtl} = \alpha\, \mathcal{L}_{SR}(\hat{I}^{HR}, I^{HR}) + (1-\alpha)\, \mathcal{L}_{QE}(\hat{I}^{LR}, I^{LR}). \tag{5.12}$$

The schematic in Fig.5.2 illustrates the structure of the different components of MTL-EDSR. The shared network f_θ mainly consists of B RB with short and long skip connections. These operations allow the network to learn the identity function, improving the gradient flow from the deep to the shallow layers during the back-propagation step. It also leads to more sparse feature maps, and thus, better performance. For each convolutional layer, we use 256 filters of size 3×3. We introduce the non-linearity with the activation function ReLU between layers at different stages of the network. This structure is directly inspired by the EDSR network [15], which proposes state-of-the-art performance for super-resolution. We split the network at a very deep stage of the architecture to maximize the parameter sharing between tasks. For the super-resolution module $g_{\phi_{SR}}$, we use the Pixel-Shuffle upscaling layer [176] at the end of the network. The same structure is used for the quality enhancement module $g_{\phi_{QE}}$, without the upscaling layer.

5.3.2 Training procedure and dataset

Dataset

For the whole experiments, we train the networks with the DIV2K image dataset [192]. This later consists of 900 HD PNG pictures with a high diversity of spatial characteristics. To prevent network overfitting, we evaluate the performance on the Set5 image dataset [85]. The low-resolution images I^{LR} are generated by a bicubic downscale applied on the high-resolution images I^{HR}. To generate the reconstructed versions $\tilde{I}^{LR}$ of the uncompressed images I^{LR}, we use the VTM-11 in all-intra configuration with $QP \in \{22, 27, 32, 37\}$ in order to simulate different levels of coding artifacts. We first convert the images from PNG to YUV4:2:0 format. Then, we collect the reconstructed images and convert them back to RGB. For training, we use 64×64 patches extracted from the training set to reduce GPU memory usage. To test the performance of our network on video sequences, we also generate data from the ClassB and ClassA of the JVET CTC [23] using VVC all-intra, as described above. We also include two 8K videos, selected from the dataset given in [178]. For the whole experiments, the quality is assessed on the luma component using PSNR and SSIM [193] image quality metrics computed between the estimated and original images. We also compute Δ-PSNR and Δ-SSIM that indicate the gain compared to the decoded images prior post-processing. For Super-Resolution, we use

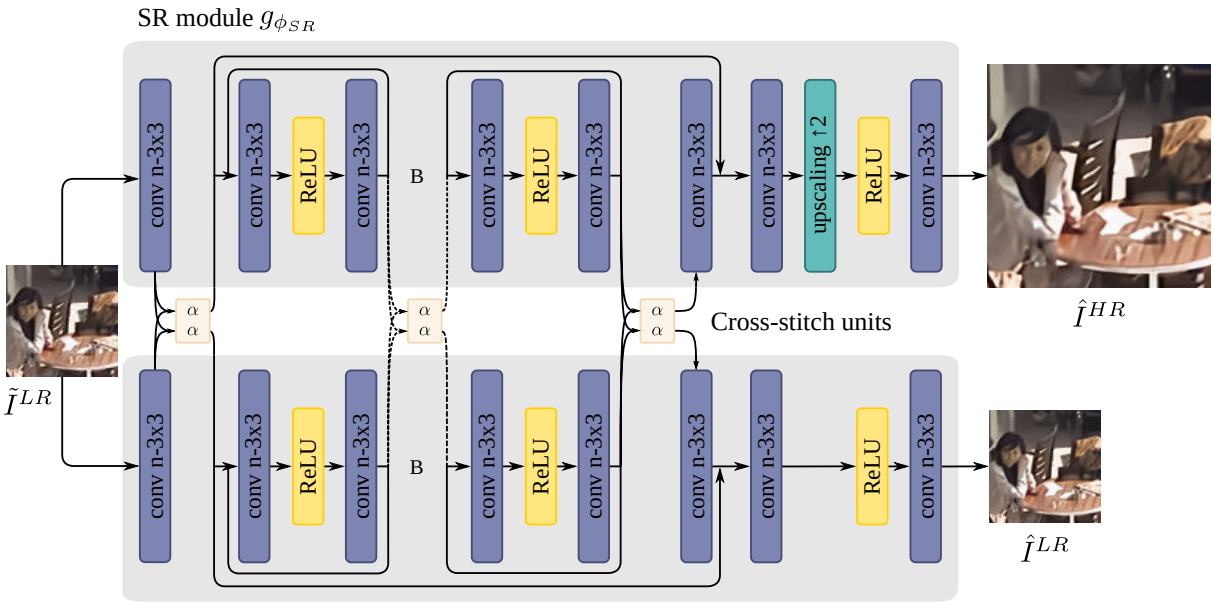


Figure 5.4 – Soft parameter sharing (cross-stitch [22]).

bicubic interpolation as anchor.

Baselines

Single-task architectures derive from the proposed MTL-EDSR by setting α to 0 and 1 in the multitask loss $\mathcal{L}_{mtl}$, defined in (5.12), for quality enhancement and super-resolution, respectively. We also include a sequential configuration based on these two single-task networks. For an input image $\tilde{I}^{LR}$, the sequential configuration can be expressed as:

$$\hat{I}^{HR} = g_{\hat{\phi}_{SR}}(f_{\hat{\theta}}(g_{\phi_{QE}}(f_{\hat{\theta}}(\tilde{I}^{LR} \oplus qp_{map})))). \tag{5.13}$$

In that case, quality enhancement is applied to the input image before passing through the super-resolution specialized network. For this experiment, we set the number of RB for each specialized network in both sequential and single-task configurations to $B = 4$, leading to approximately 7.7 million parameters per network. We also include single-task models with $B = 8$ to match the performance with the multitask configuration.

Regarding the tested multitask architectures, we compare our approach with a soft parameter sharing architecture based on cross-stitch [22]. The soft sharing architecture is illustrated in Fig 5.4. We also include the multitask loss computation described in Section 5.2.1, including uncertainty-based [187] and DWA [188].

Training

We train our model over 250 epochs, with a learning rate of 10^{-4} for from-scratch training and 10^{-5} for fine-tuning. For fine-tuning, the pre-trained weights are obtained by training the network for super-resolution on uncompressed image pairs during 1000 epochs with a learning-rate of 10^{-4}. We apply a learning rate decay with a gamma of 0.5 every 75 epochs to improve the convergence. We use a batch size of 8 and optimize the model with ADAM [80] by setting $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. The parameter α in (5.7) was tuned and fixed to 0.9 after a grid search on different values. All the experiments are performed on an NVIDIA Telsa V100 GPU using PyTorch.

5.3.3 Experimental results

Multi-QP optimization

In the first experiment, we want to evaluate the ability of our multitask model to generalize across several QPs through an ablation study. The tested configurations include multi-QPs training, fine-tuning and qp_{map}, as described in Section 5.3.1. Since less data are available for the training of the QP-specific networks than for a single multi-QPs network, we multiply the number of epochs by the number of tested QPs, i.e., 4, for these QP-specific configurations. We also adjust the learning rate decay to be applied every 4×75 epochs in this case. Thus, all the presented models are trained with the same number of parameter updates allowing a fair evaluation. The qp_{map} is computed for each tested QP by (5.7). We set the number of RB to $B = 8$ for all the tested models, leading to around 13 million parameters per network.

Table 5.1 shows the performance of our model on Set5 dataset for different input QP in terms of PSNR (dB) and Δ-PSNR (dB) for both super-resolution and quality enhancement. We perform an ablation study to evaluate the contribution of each component of our multi-QP model in the global performance of the network. We observe that a fine-tuning of the network pre-trained with uncompressed images leads to 0.08dB and 0.06dB of gain for SR and QE, respectively. We notice that even using parameters pre-trained for super-resolution, quality enhancement performs better as well. We also see that the qp_{map} contributes to the performance of our multi-QP model by increasing the quality of reconstruction by 0.06dB for SR and 0.05dB for QE. It can be noted that the models based on QP-specific training perform slighly better in terms of quality than our multi-QP model. However, training one network per QP requires four times more training time and parameters than a multi-QPs network to reach this level of performance.

Fig. 5.5 visualizes the convergence of each multi-QP configuration by assessing the PSNR on

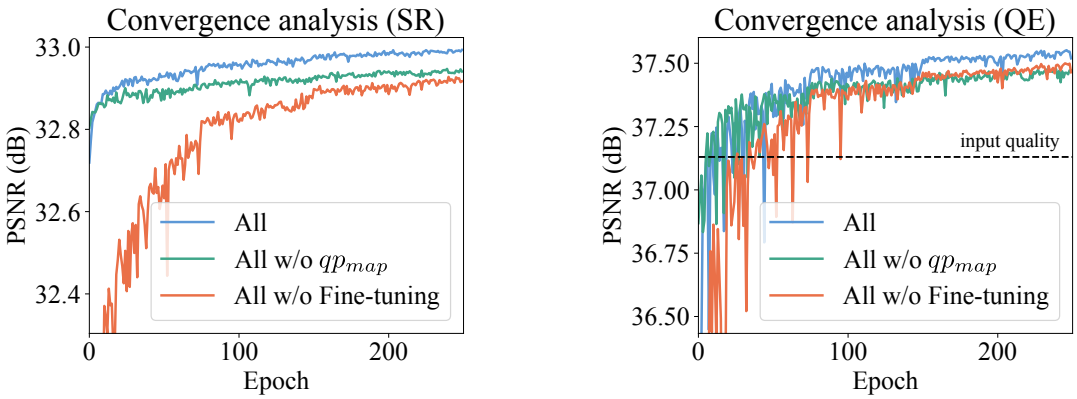
Table 5.1 – Ablation study of our model on Set5 for both tasks in terms of PSNR (dB) and Δ-PSNR (dB).

Multi-QPs	✓	✓	✗	✗	✓	✓	✓	✓
qp_{map}	✓	✓	✓	✓	✗	✗	✓	✓
Fine-tuning	✓	✓	✓	✓	✓	✓	✗	✗
Task	SR	QE	SR	QE	SR	QE	SR	QE
QP22	35.80 [+2.66]	43.07 [+0.43]	**35.85 [+2.71]**	**43.19 [+0.55]**	35.69 [+2.55]	42.92 [+0.28]	35.67 [+2.53]	42.98 [+0.34]
QP27	34.17 [+1.91]	39.18 [+0.39]	**34.18 [+1.92]**	**39.24 [+0.45]**	34.16 [+1.90]	39.21 [+0.42]	34.09 [+1.83]	39.13 [+0.34]
QP32	**32.16 [+1.22]**	35.62 [+0.40]	**32.16 [+1.22]**	**35.67 [+0.45]**	32.14 [+1.20]	35.63 [+0.41]	32.10 [+1.16]	35.58 [+0.36]
QP37	29.85 [+0.70]	32.20 [+0.35]	**29.89 [+0.74]**	**32.27 [+0.42]**	29.78 [+0.63]	32.13 [+0.28]	29.81 [+0.66]	32.16 [+0.31]
Average	33.00 [+1.63]	37.52 [+0.39]	**33.02 [+1.65]**	**37.59 [+0.46]**	32.94 [+1.57]	37.47 [+0.34]	32.92 [+1.55]	37.46 [+0.33]

the validation set at each training epoch for both tasks. We clearly notice that fine-tuning offers a more stable training with a faster convergence than from-scratch training for super-resolution. It is not surprising as the network starts to learn with weights that are already tuned for a related task. Although this configuration also leads to better results for quality enhancement, this observation is less pronounced in that case. Moreover, the training is globally less stable for this task. It can be explained by the fact that the loss related to super-resolution is more weighted in the proposed multitask loss $\mathcal{L}_{mtl}$. However, we notice that the use of qp_{map} leads to a better convergence for both tasks.

Multitask learning

Table 5.2 gives the performance in terms of PSNR, SSIM, Δ-PSNR and Δ-SSIM for both tasks regarding the different baselines described in Section 6.3.1. In this experiment, all the models are trained without fine-tuning and qp_{map}. Firstly, we notice that the sequential configuration does not perform significantly better than the single-task by considering the same total number of parameters. Moreover, super-resolution with $B = 8$ enables better performance than the sequential model regarding this task. Similarly, we notice that a hard parameter sharing with $B = 8$ proposes better results than a soft parameter sharing with $B = 4$, while the number of parameter is the same. For our multitask approach MTL-EDSR, we notice that the performance of single-task is reached with half parameters. In addition, the multitask performs better in terms of



(a) Super resolution

(b) Quality enhancement

Figure 5.5 – Convergence analysis of MTL-EDSR regarding different training configurations.

quality than single-task considering the same total number of parameters. It can be explained by the fact that a large number of features are computed twice between the specialized architectures. Thus, a deeper multitask model allows new representations to be learned, increasing the quality of reconstruction for both tasks. Regarding, the multitask loss, all tested solutions provide similar performances for both tasks.

In Fig.5.6, we display the average feature maps for different convolutional layers of the single-task and multitask architectures. As shown in this figure, the features are globally similar and become more complex and specialized in the deeper layers. For Conv_{17}, the average feature map of multitask is more similar to super-resolution than quality enhancement, mostly because $\alpha = 0.9$ in the multitask loss $\mathcal{L}_{mtl}$ of the proposed model. This demonstrates that a high correlation exists between the presented single-task models which can be exploited in the multitask architecture.

Coding performance

In the last experiment, we investigated the performance of our multitask model applied as post-processing for video delivery against single-task networks. The input signal is first downscaled and encoded. Then, both post-processing tasks are performed on the decoded signal outside the coding loop, as presented in [194] for super-resolution. The bit-rate is assessed on the low-resolution signal. For this experiment, we consider the same total number of parameters for both tested configurations, i.e., $B = 8$ for our multitask network and $B = 4$ for each single-task network. We use the BD-Rate method described in [44] to evaluate our approach. Table 5.3

Table 5.2 – Average performance (images, QPs) of the different Baselines in (Δ-)PSNR (dB) and (Δ-)SSIM computed on the Set5 dataset. The value of B corresponds to the number of RB used in the shared network f_θ for Baseline-B.

Method	Baseline-B	Multitask loss	Super-Resolution		Quality Enhancement	
			PSNR	SSIM	PSNR	SSIM
Naive	w/o Enh	-	-	-	37.12	0.9532
	Bicubic	-	31.37	0.8714	-	-
Single-task SR	SR-4	-	32.80	0.8874	-	-
	SR-8	-	32.87	0.8884	-	-
Single-task QE	QE-4	-	-	-	37.33	0.9542
	QE-8	-	-	-	37.38	0.9847
Sequential	QE-4;SR-4	-	32.81	0.8873	37.33	0.9542
Multitask	Hard sharing-8	$\alpha = 0.9$	**32.90**	**0.8885**	37.40	**0.9548**
		uncertainty [187]	32.87	0.8881	**37.41**	0.9547
		DWA [188]	32.89	0.8884	37.40	**0.9548**
	Soft sharing-4	$\alpha = 0.9$	32.80	0.8873	37.36	0.9544
		uncertainty [187]	32.80	0.8874	37.34	0.9542
		DWA [188]	32.80	0.8871	37.22	0.9544

presents the results for both super-resolution and quality enhancement.

We can notice that, in average, our multitask model allows 2.8%/2.1% and 2.3%/1.1% of bit-rate savings over specialized networks for the same objective quality, using PSNR and SSIM, regarding super-resolution and quality enhancement, respectively. We can also notice that these gains are higher for the sequences where our method performs well against naive anchors. These video sequences including *BQTerrace* and *SubwayTree* contain more spatial information and need more powerful models to be accurately reconstructed.

5.3.4 Analysis and discussion

In this section, we presented MTL-EDSR, a multitask learning-based approach that performs both super-resolution and quality enhancement of VVC intra-coded LR frames. We used a multi-QPs training strategy based on fine-tuning and prior information. We demonstrated that our method allows a significant reduction of parameters, while maintaining a good quality of reconstruction compared to specialized solutions. We also showed that our approach offers quality enhancements compared to single-task models when the same total number of parameters is considered.

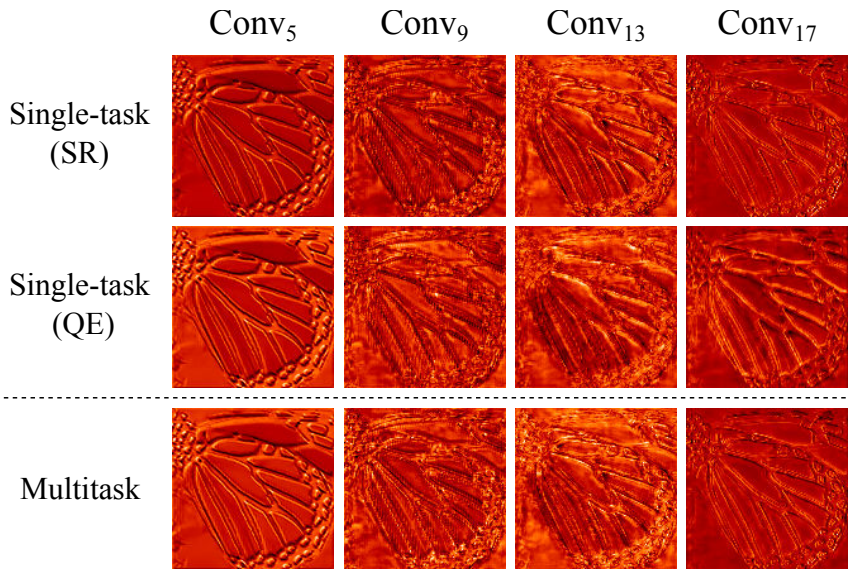


Figure 5.6 – Feature analysis.

5.4 High-level vision tasks

High-level vision tasks can be helpful for video coding algorithms by analyzing the scene content on the server or client-side. In the previous section, we introduced MTL-EDSR, a multitask CNN that performs simultaneously super-resolution and quality enhancement using hard parameter sharing. In extension to this work, we present MTL-Unet, a multitask network dedicated to super-resolution and high-level vision tasks, namely, semantic segmentation and no-reference image quality assessment (NR-IQA). Similarly to MTL-EDSR, we show that the proposed model can perform the different tasks with a low impact on the super-resolution reconstruction error, reducing the number of parameters compared to specialized architectures.

5.4.1 High-level vision tasks in video coding

In contrast to low-level image processing tasks, e.g. super-resolution and quality enhancement, high-level vision tasks aim at understanding the scenes beyond local statistics. A higher level of abstraction is required for those tasks, e.g. semantic segmentation, depth estimation, NR-IQA, widely achieved in the literature using DNN [17, 195, 196, 197].

Table 5.3 – BD-rate (%) of MTL-EDSR computed over single-task baselines regarding PSNR and SSIM for different resolution classes. The values in bracket indicate the gain compared to naive anchors, i.e., bicubic upscale and input quality.

Dataset	Sequence	Super-Resolution		Quality Enhancement	
		PSNR	SSIM	PSNR	SSIM
8K (7680x4320)	*SubwayTree*	-3.55 [-17.17]	-1.15 [-9.12]	-2.19 [-5.85]	-0.90 [-3.39]
	TiergartenParkway	-0.88 [-7.90]	-0.93 [-5.46]	-1.64 [-3.26]	-0.80 [-1.99]
ClassA1 (3840x2160)	*Campfire*	-1.19 [-9.51]	-0.78 [-6.16]	-1.30 [-2.92]	-0.58 [-1.53]
	FoodMarket4	-1.97 [-15.97]	-1.55 [-9.66]	-2.19 [-5.13]	-0.97 [-2.67]
	Tango2	-1.47 [-8.41]	-1.40 [-5.61]	-3.91 [-6.04]	-1.19 [-3.23]
ClassA2 (3840x2160)	*CatRobot1*	-2.47 [-16.75]	-2.53 [-15.87]	-2.41 [-5.62]	-1.47 [-3.87]
	DaylightRoad2	-1.37 [-10.28]	-1.31 [-7.30]	-1.89 [-4.77]	-0.94 [-2.44]
	ParkRunning3	-1.44 [-12.45]	-1.34 [-10.15]	-1.46 [-3.18]	-0.84 [-2.34]
ClassB (1920x1080)	*BasketballDrive*	-5.44 [-53.94]	-4.22 [-40.42]	-1.92 [-4.29]	-1.68 [-3.59]
	BQTerrace	-7.99 [-55.00]	-5.43 [-34.09]	-2.64 [-4.97]	-1.58 [-2.82]
	Cactus	-3.01 [-33.17]	-2.35 [-22.93]	-2.16 [-4.55]	-1.37 [-3.24]
	MarketPlace	-3.52 [-28.84]	-1.63 [-18.10]	-3.10 [-5.25]	-0.78 [-2.64]
	RitualDance	-2.54 [-17.50]	-2.55 [-12.82]	-3.61 [-7.10]	-1.66 [-5.08]
	Average	**-2.83 [-22.07]**	**-2.09 [-15.21]**	**-2.34 [-4.84]**	**-1.14 [-2.99]**

Semantic segmentation

Semantic segmentation of images is a supervised learning task with two objectives: segmenting objects in the input image and assigning a label to the segmented areas.

This field has been investigated in MPEG-4 standard [198] with object-based video coding [199, 200, 201, 202]. This approach allows controlling the degree of compression of video objects based on their semantical relevance, which is helpful for rate control algorithms. Connecting learned low-level image processing models with semantic segmentation has already been investigated in the literature. Recently, semantic segmentation has been proposed in learned video coding to enhance the quality of reconstruction compared to traditional approaches [203, 204]. Authors in [205] have extended this approach by applying semantic segmentation to the upscaled image on both encoder and decoder sides, avoiding transmitting semantic maps through the network.

Regarding image post-processing, authors in [206] proposed a model that combines denoising and semantic segmentation CNNs to improve both tasks' performance. A multitask learning-based model has been developed to perform super-resolution and semantic segmentation in [207]. However, those approaches are based on GANs and perceptual loss, which is hard to control and inappropriate in a broadcast context.

No-reference image quality assessment

Real-time communication and streaming services require quality adaptations using objective quality assessment methods to collect the delivered quality statistics. Image quality assessment (IQA) is a fundamental problem in the image processing field and requires efficient algorithms to accurately model the HVS. There are three types of IQA approaches:

— full-reference image quality assessment (FR-IQA) which require both the source and degraded images,
— reduced-reference image quality assessment (RR-IQA) which require both a description of the source and degraded images,
— NR-IQA which only require the degraded image

In a broadcast context, NR-IQA methods are the more suited to assess services at the receiver-side, as transmitting additional information about the source through the network cannot is sometimes unrealistic. However, this latter is the more challenging as no-reference are used in the quality evaluation process. Traditional NR-IQA approaches predict a score or a map from features that relate to a specific kind of distortion, such as the amount of noise or block boundaries detector. As a significant part of inverse problems, some recent works have investigated NR-IQA using DNN [208, 209, 210, 211] and have outperformed traditional approaches by learning the appropriate features that map the reconstructed degradation map from a single image and the FR-IQA ground truth.

5.4.2 Proposed solution

The proposed MTL-Unet network is based on hard parameter sharing using a shared network f_θ and two task-specific modules $g_{\phi_{SR}}$ and g_{ϕ_k}, with $k \in \{IQA, Seg\}$ for the task of NR-IQA and semantic segmentation, respectively. First we extract the shared features y from the input reconstructed image $\tilde{I}^{LR}$ as:

$$y = f_\theta(\tilde{I}^{LR}). \tag{5.14}$$

Then, the main and additional outputs are generated by the super-resolution module $g_{\phi_{SR}}$ and the additional task module g_{ϕ_k} following:

$$\hat{I}^{HR} = g_{\phi_{SR}}(y), \tag{5.15}$$

and

$$\hat{I}^k = g_{\phi_k}(y), \tag{5.16}$$

for $k \in \{IQA, Seg\}$, respectively. We selected DWA [188] loss function, detailed in Section 5.2, to optimize our multitask network:

$$\mathcal{L}_{DWA} = \sum_k \lambda_k \mathcal{L}_k. \tag{5.17}$$

The proposed architecture, described in Figure 5.7, is based on Unet [17], commonly used for semantic segmentation tasks. This encoder-decoder solution is composed of pooling and upscaling layers, increasing the network's receptive field and, thus, the level of abstraction. However, unlike other encoder-decoder networks, Unet-based architectures allow preserving spatial information of the input by transmitting the high-resolution feature maps to the decoder side of the network. The learned representations are transferred from encoder to decoder by a concatenation along the feature axis. Thus, a better fidelity with the input image can be obtained for the reconstructed map. It is also valuable in our case for the super-resolution task, which needs as much spatial information from the input as possible. The network comprises several convolution layers composed of 64 kernels of size 3×3. For simplicity, we do not display activations in the core of the shared network f_θ.

5.4.3 Training procedure and dataset

Datasets

As in Section 6.3.1, we use the DIV2K image dataset [192] for super-resolution and NR-IQA tasks. The low-resolution images I^{LR} are generated by a bicubic downscale applied on the high-resolution images I^{HR}. To generate the reconstructed versions $\tilde{I}^{LR}$ of the uncompressed images I^{LR}, we use the VTM-11 in all-intra configuration with $QP \in \{22, 27, 32, 37\}$. We first convert the images from PNG to YUV4:2:0 format. Then, we collect the reconstructed images and convert them back to RGB. The ground truth degradation maps are generated using the SSIM between the uncompressed and compressed images which outputs a spatial degradation map. The output maps are denoted I^{SSIM}. As the task of semantic segmentation requires labeled images, we used another training dataset, called PASCAL-VOC dataset [212], which contains 3600 images and 459 classes. We restrict the number of classes to 10: *Tree, Sky, Building, Ground, Wall, Grass, Floor, Person, Water*, and set *None* for the others in order to simplify the

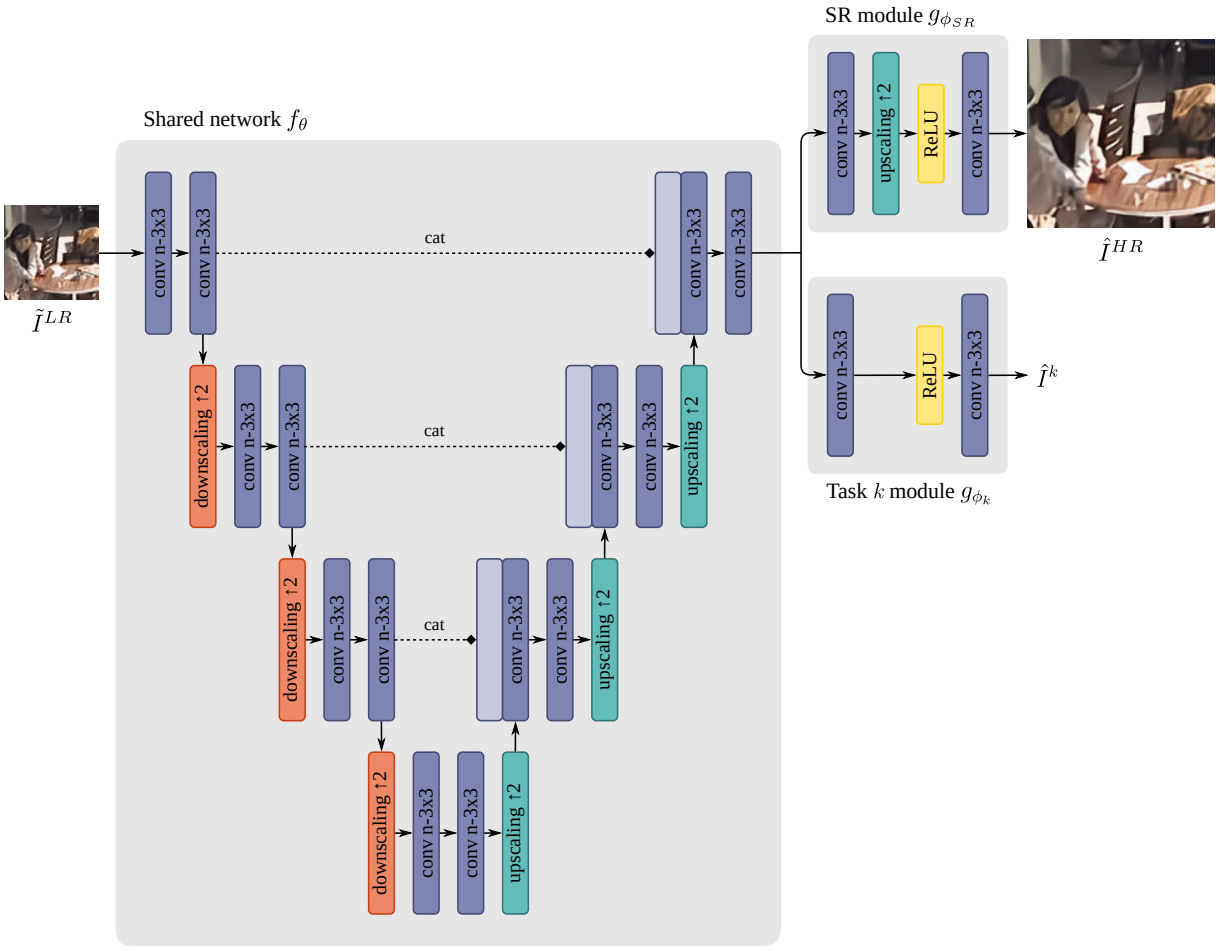


Figure 5.7 – Architecture of the proposed MTL-Unet network.

task. We use the cross-entropy as a loss function as described in Section 5.4.1.

Baselines

For the NR-IQA task, we selected the $L1$-loss between the true SSIM-map I^{SSIM} and the output of the network as a loss function $\hat{I}^{SSIM}$:

$$\mathcal{L}_{IQA} = ||\hat{I}^{IQA} - I^{IQA}||_1 \tag{5.18}$$

For the semantic segmentation task, we used the cross-entropy loss function between ground-truth I^{Seg} and predicted labels $\hat{I}^{Seg}$ as follows:

$$\mathcal{L}_{Seg} = -\sum_{c=1}^{C} I_c^{Seg}\, log(\hat{I}_c^{Seg}), \tag{5.19}$$

Table 5.4 – Comparison of MTL-EDSR and MTL-Unet for super-resolution in single-task mode.

	MTL-EDSR		MTL-Unet	
	PSNR	SSIM	PSNR	SSIM
QP22	**35.62**	**0.9359**	35.40	0.9347
QP27	**34.05**	**0.9126**	33.94	0.9118
QP32	**32.06**	**0.8802**	31.99	0.8794
QP37	**29.74**	**0.8348**	29.67	0.8337
Average	**32.87**	**0.8884**	32.75	0.8899

with C denoting the number of classes.

All the networks are trained from scratch in multi-QP training mode. DWA loss is evaluated against weighted multitask loss with $\alpha = 0.5$ and uncertainty-based multitask loss. For MTL-EDSR, we set the number of residual blocks $B = 8$, providing a fair evaluation with MTL-Unet regarding the number of parameters.

Training

For training, we use 64×64 patches extracted from the training set to reduce GPU memory usage. For the whole experiments, the quality is assessed on the luma component using PSNR and SSIM [193] image quality metrics computed between the estimated and original images. For the task of semantic segmentation, we use two different datasets to train the multitask network: one for super-resolution and one for semantic segmentation. Thus, the task-specific loss functions are computed on the dataset related to the task.

We train the models over 250 epochs, with a learning rate of 10^{-4}. We apply a learning rate decay with a gamma of 0.5 every 75 epochs to improve the convergence. We use a batch size of 8 and optimize the model with ADAM [80] by setting $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. All the experiments are performed on an NVIDIA Telsa V100 GPU using PyTorch.

5.4.4 Experimental results

Super-resolution

In this experiment, we evaluate the performance of the U-net compared to the EDSR-based architecture described in Section 5.3 for super-resolution in single-task mode. Both networks have a similar amount of trained parameters. The results are represented in Table 5.4.

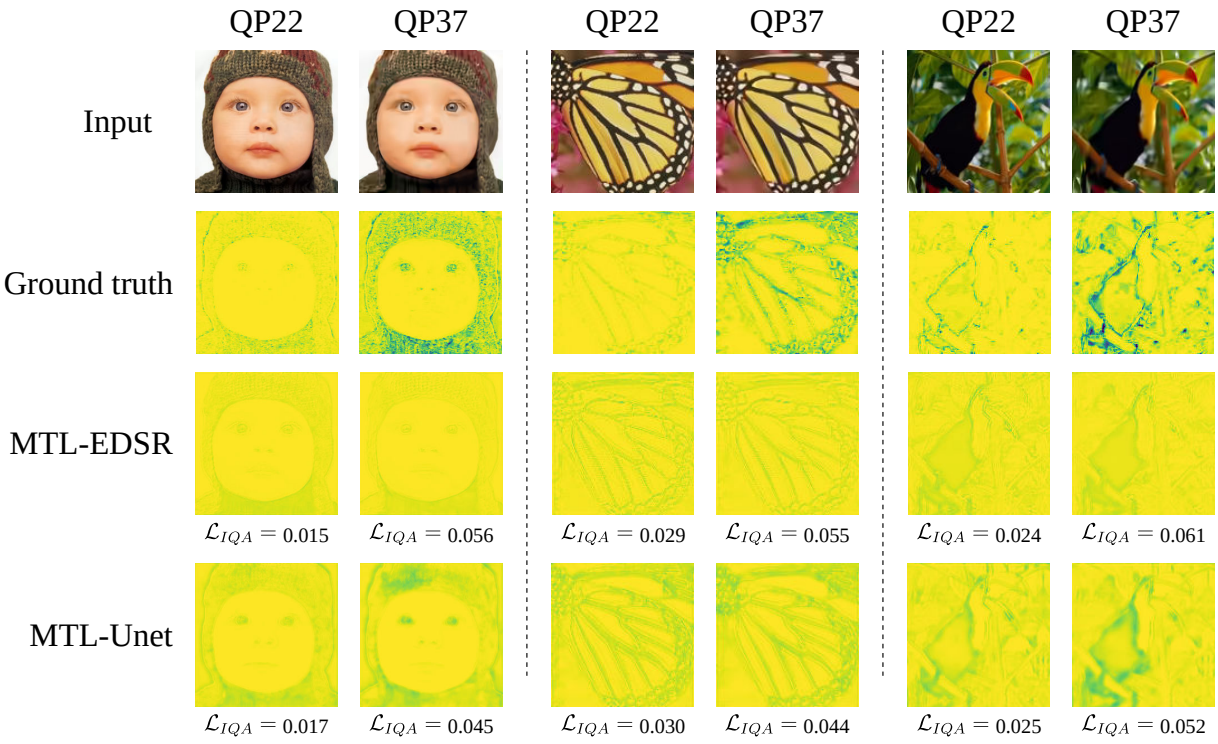


Figure 5.8 – Visual comparison of MTL-EDSR and MTL-Unet for the task of NR-IQA.

We notice that MTL-EDSR outperforms the MTL-Unet for all QPs in both PSNR and SSIM. Although the high-resolution feature maps are transferred at the decoder side by MTL-Unet, the dimensionality reduction causes a loss in performance compared to MTL-EDSR. In this latter, the dimensionality of features remains unchanged along with the network, making this architecture more suited to super-resolution. Indeed, as many details from the input image as possible are required to recover the missing high-resolution details.

High-level vision tasks

In this test, we visually compare the performance of both MTL-Unet and MTL-EDSR for the tasks of semantic segmentation and NR-IQA. Visual reconstructions using both architectures on Set5 dataset samples are represented in Figure 5.8 for the task of NR-IQA, and in Figure 5.9 for the task of semantic segmentation. The networks are optimized with the DWA-based multitask loss function, defined in Section 5.4.2. For the task of NR-IQA, we notice that MTL-Unet reconstruction is closer to the target SSIM map than MTL-EDSR for which the degradation is not detected. For semantic segmentation tasks, MTL-Unet reproduces better quality and more dense maps than MTL-EDSR.

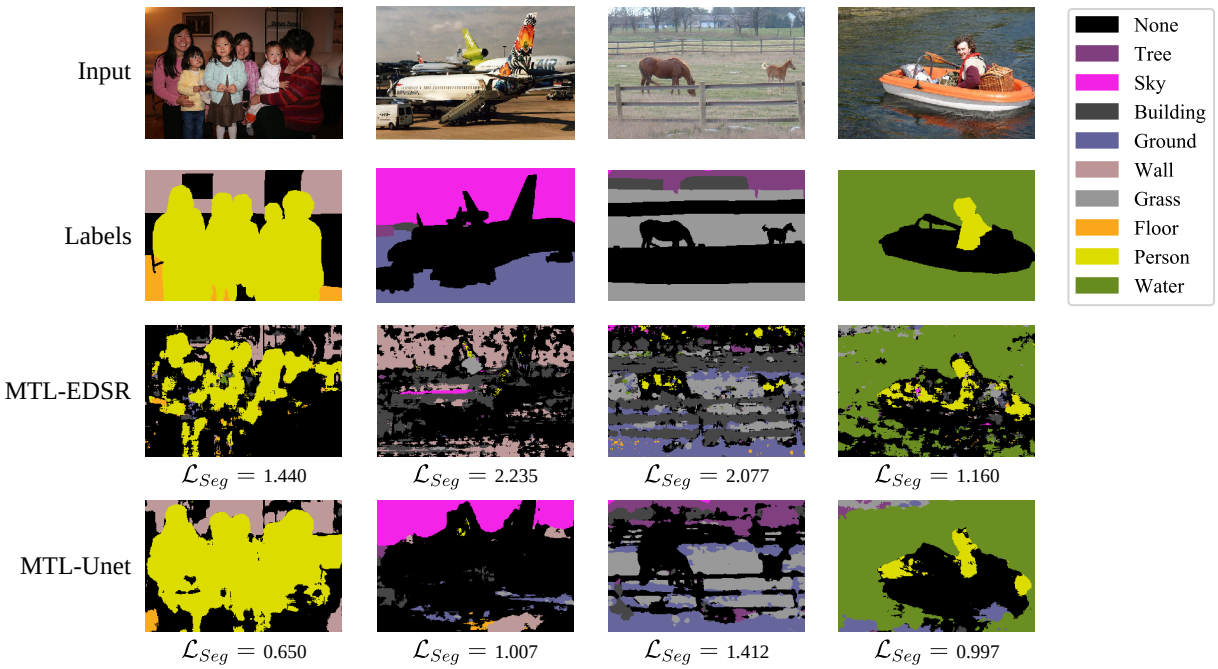


Figure 5.9 – Visual comparison of MTL-EDSR and MTL-Unet for the task of semantic segmentation ($QP = 22$).

Multitask learning

This experiment evaluates MTL-Unet for super-resolution trained with semantic segmentation or NR-IQA as an additional task. Results are reported in Table 5.5. We notice that the performance of MTL-Unet is higher by considering NR-IQA as an additional task, which is more related to the main task of super-resolution than semantic segmentation. We also show that DWA outperforms weighted loss ($\alpha = 0.5$) and uncertainty-based multitask loss for almost all QPs, regarding PSNR and SSIM. However, the single-task Unet performs better than MTL-Unet for the super-resolution task regarding all the tested multitask configurations. This observation denotes a negative transfer between super-resolution and the high-level vision additional tasks. Although the single-task Unet outperforms MTL-Unet for super-resolution, the proposed multitask architecture allows tasks useful for video coding algorithms to be performed in parallel with super-resolution using a single shared network.

5.4.5 Analysis and discussion

In this section, we presented MTL-Unet, a multitask learning-based approach that performs both super-resolution and high-level vision tasks from VVC intra-coded input frames. Similarly to MTL-EDSR described in section 5.3, we demonstrated that the proposed method allows a

Table 5.5 – Performance of MTL-Unet for the different tested additional tasks and multitask losses. The values in bracket indicate the Δ-PSNR and Δ-SSIM compared to MTL-Unet in single-task mode.

$\alpha = 0.5$				
	$k = Seg$		$k = IQA$	
	PSNR	SSIM	PSNR	SSIM
QP22	35.04 [-0.36]	0,9296 [-0.0051]	35,24 [-0.16]	0,9314 [-0.0033]
QP27	33.70 [-0.24]	0,9072 [-0.0046]	33,80 [-0.14]	0,9086 [-0.0032]
QP32	31.85 [-0.14]	0,8751 [-0.0043]	31,92 [-0.07]	0,8763 [-0.0031]
QP37	29.62 [-0.05]	0,8298 [-0.0039]	**29,70 [-0.03]**	0,8314 [-0.0023]
Average	32,55 [-0.20]	0,8854 [-0.0045]	32,67 [-0.09]	0,8869 [-0.0030]
Uncertainty [187]				
	$k = Seg$		$k = IQA$	
	PSNR	SSIM	PSNR	SSIM
QP22	34.90 [-0.50]	0.9306 [-0.0041]	35.09 [-0.31]	0.9328 [-0.0019]
QP27	33.61 [-0.33]	0.9087 [-0.0031]	33.72 [-0.22]	0.9098 [-0.0020]
QP32	31.79 [-0.20]	0.8766 [-0.0028]	31.83 [-0.16]	0.8771 [-0.0023]
QP37	29.57 [-0.10]	0.8311 [-0.0026]	29.59 [-0.08]	0.8318 [-0.0019]
Average	32.47 [-0.28]	0.8868 [-0.0032]	32.56 [-0.19]	0.8878 [-0.0020]
DWA [188]				
	$k = Seg$		$k = IQA$	
	PSNR	SSIM	PSNR	SSIM
QP22	35.28 [-0.12]	0.9337 [-0.0010]	**35.38 [-0.02]**	**0.9345 [-0.0002]**
QP27	33.85 [-0.09]	0.9109 [-0.0009]	**33.92 [-0.02]**	**0.9114 [-0.0004]**
QP32	31.92 [-0.07]	0.8784 [-0.0010]	**31.96 [-0.03]**	**0.8790 [-0.0004]**
QP37	29.63 [-0.04]	0.8325 [-0.0012]	29.66 [-0.01]	**0.8334 [-0.0003]**
Average	32.67 [-0.08]	0.8889 [-0.0010]	**32.73 [-0.02]**	**0.8896 [-0.0003]**

significant reduction of parameters, while maintaining a good quality of reconstruction compared to specialized solutions. Our approach also provides better quality of reconstruction compared to single-task models when the same total number of parameters is considered.

5.5 Conclusion

In this chapter, we investigated different multitask learning approaches for super-resolution on LR images encoded using VVC AI mode. Two multitask models have been designed:

— MTL-EDSR, which is dedicated to super-resolution and VVC quality enhancement.
— MTL-Unet, which is dedicated to super-resolution and high-level vision tasks, i.e., NR-IQA and semantic segmentation.

Although the developed architectures do not improve single-task super-resolution, we showed that additional tasks of several natures could be performed simultaneously with a low impact on the reconstruction error. On one hand, DNNs-based architectures recently achieved outstanding performance in several domains impacting video coding, e.g. super-resolution, quality enhancement, NR-IQA or semantic segmentation. On the other hand, one model for each task is trained and deployed in practice, while some of the learned reresentations can be useful for other tasks. To tackle this, the proposed multitask approaches could be considered to save storage space at the receiver side while performing multiple tasks.

However, several tracks remain in this work. Regarding super-resolution applied on compressed LR video, the temporal aspect can be considered to provide information about the missing sub-pixel located in the neighboring frames [213, 214]. Also, the use of richer prior information can be considered. For instance, a recent contribution [215] proposed using the LR input's prediction, which gives both spatial and temporal knowledge about the decoded LR image. Moreover, some works considered providing the prior information at deeper stages of the network to improve the performance [190]. Regarding multitask learning for super-resolution, performing all the tested tasks together with a single network has not been investigated in this work. Furthermore, assessing the other tasks would allow to know the impact of super-resolution on higher-level tasks. Finally, as proposed by authors in [188], more sophisticated multitask architectures can be considered to perform soft-parameter sharing without consequently increasing the number of parameters.

This chapter demonstrated that dedicated architecture and training strategies do not significantly improve super-resolution applied on compressed data, thus remaining content-dependent. Consequently, we decided to stop the explorations on this topic for this work. In the following we consider side information coupled with super-resolution to recover the lost details. As mentioned in chapter 4, the BL codec's dependence and the additional delay introduced by scalable codecs, such as SHVC [26], make them difficult to set up in practice. Thus, low-complex and codec agnostic approaches must be considered, such as metadata approaches like by LCEVC [216] recently proposed as MPEG-5 Part 2 standard.

CHAPTER 6

LEARNING-BASED VIDEO CODING FOR EFFICIENT LAYERED COMPRESSION

6.1 Preamble

The previous chapters evaluate super-resolution algorithms for downscaling-based compression. The results demonstrate that, at low bitrate, this type of framework can improve the performance of single-layer coding by increasing the bit-per-pixel ratio. However, the degradations generated by the quantization step reduce the performance of super-resolution models compared to conventional interpolation methods. Moreover, the performances of this type of framework are content-dependent due to some high-frequencies that cannot be recovered by post-processing. A solution to this problem would be to transmit the lost spatial information using an enhancement layered approach, like LCEVC. This codec encodes the residual between the reconstructed BL signal and the source as side information. However, this solution is based on handcrafted modules that are not jointly optimized, i.e., upscaling, partitioning, transform. Recently, autoencoders have shown outstanding performance in compressing image and video signals by training all modules together in an end-to-end fashion.

This chapter presents CAESR. This learning-based layered approach uses a conditional autoencoder as an EL model and a conventional single-layer codec as a BL model. The presented method is trained to encode the residual between the upscaled reconstructed image and the source. Section 6.2 presents the overall pipeline of the proposed solution. Section 6.3 evaluates our algorithm as a codec enhancer based on the HM-16.20. We first provide an ablation study that validates the efficiency of our method. Then, we compare it against state-of-the-art layered approaches, including LCEVC and SHVC, for single-layer codec enhancement. Finally, Section 6.4 evaluates our algorithm for the deployment of new services by considering 4K video delivery on top of an HD signal regarding typical bitrates used for DTT broadcast.

6.2 Conditional autoencoder and super-resolution (CAESR)

This section presents the implementation details of the proposed model. Our approach relies on conditional coding that allows a non-linear mixture of the source and the reconstructed signal to be learned, thus improving the performance compared to residual coding [217]. The overall pipeline of the proposed solution is described in Fig. 6.1.

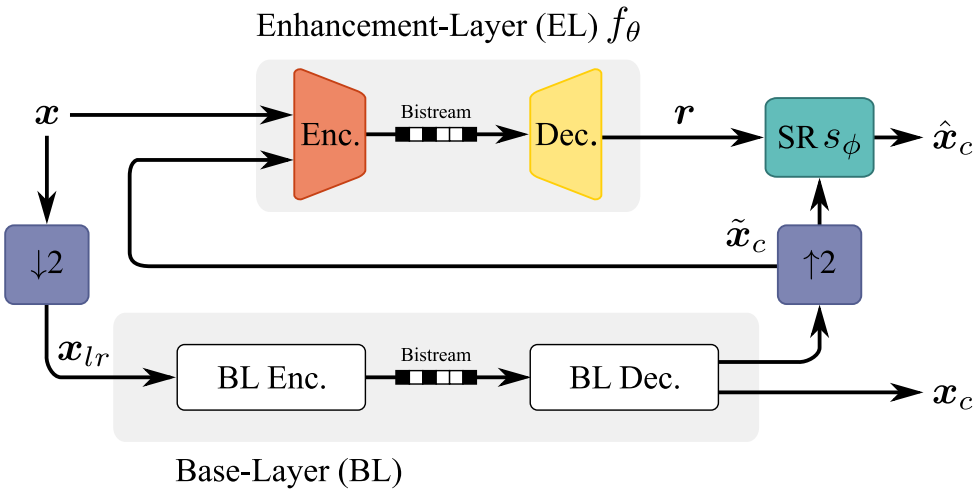


Figure 6.1 – Description of CAESR.

6.2.1 Framework and formulation

Given an input image $\boldsymbol{x} \in \mathbb{R}^{W \times H \times C}$ of width W and height H represented by C channels, we first apply a spatial downscale by a factor 2 to generate the input BL image $\boldsymbol{x}_{lr}$. This latter is encoded with a BL encoder, e.g. HEVC or VVC. The decoded image $\boldsymbol{x}_c$ is then rescaled to the original resolution $W \times H$ to form the EL model's input $\tilde{\boldsymbol{x}}_c$. To perform conditional coding, the source image $\boldsymbol{x}$ and the upscaled base reconstruction $\tilde{\boldsymbol{x}}_c$ are concatenated along the feature axis to feed the autoencoder. The resulting tensor $(\tilde{\boldsymbol{x}}_c, \boldsymbol{x}) \in \mathbb{R}^{W \times H \times 2C}$ is encoded by the encoder part of f_θ, denoted as g_a, into a latent vector $\boldsymbol{y}$. Additional latent variables $\boldsymbol{z}$ are produced by the hyper-encoder h_a to capture spatial dependencies among the element of $\boldsymbol{y}$. Both latents are quantized using the $round$ function to produce $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{z}}$. At training, we apply a uniform noise $\mathcal{U}(-\frac{1}{2}, +\frac{1}{2})$ on latents to emulate the quantization errors while enabling backpropagation, resulting in $\tilde{\boldsymbol{y}}$ and $\tilde{\boldsymbol{z}}$. To simplify, we use $\bar{\boldsymbol{y}}$ and $\bar{\boldsymbol{z}}$ to denote both actual and emulated quantized latents. The latent variables are then entropy coded regarding a GMM parameterized by the

output of the hyper-decoder h_s as:

$$p(\bar{\boldsymbol{y}}|\bar{\boldsymbol{z}}) \sim \sum_{k=1}^{K} \boldsymbol{w}^{(k)} \mathcal{N}(\boldsymbol{\mu}^{(k)}, \boldsymbol{\sigma}^{2(k)}), \tag{6.1}$$

with k the index of mixtures defined by $\boldsymbol{w}^{(k)}$, $\boldsymbol{\mu}^{(k)}$ and $\boldsymbol{\sigma}^{2(k)}$, denoting weights, means and scales, respectively.

At the decoder side, the latent residual signal r is reconstructed by the synthesis part of f_θ, denoted as g_s, and concatenated with the upscaled based-layer image $\tilde{\boldsymbol{x}}_c$ to form the input of the super-resolution network s_ϕ. Finally, the output image $\hat{\boldsymbol{x}}_c$ is reconstructed from the following equation:

$$\hat{\boldsymbol{x}}_c = s_\phi(\tilde{\boldsymbol{x}}_c, r). \tag{6.2}$$

In this work, the upscaling operation is applied before feeding the super-resolution module s_ϕ using an interpolation filter to make the network performing both high-resolution details recovering and conditional coding process inversion.

All components of the overall differentiable system are jointly trained to minimize the following rate distortion loss function $\mathcal{L}$ based on a Lagrangian multiplier λ:

$$\mathcal{L}(\lambda) = D(\hat{\boldsymbol{x}}_c, \boldsymbol{x}) + \lambda R. \tag{6.3}$$

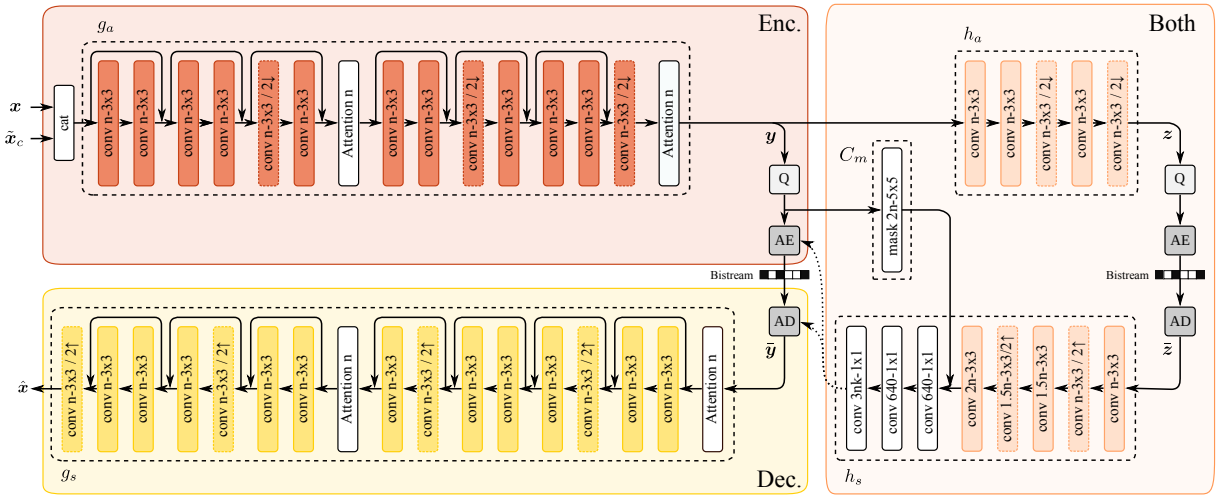
The distortion D is measured using the MSE between $\hat{\boldsymbol{x}}_c$ and $\boldsymbol{x}$. The term R corresponds to the Shannon entropy of $\tilde{\boldsymbol{y}}$, computed as:

$$R = \mathbb{E}_{\tilde{\boldsymbol{y}} \sim m}[-\log_2(p(\tilde{\boldsymbol{y}}|\tilde{\boldsymbol{z}}))], \tag{6.4}$$

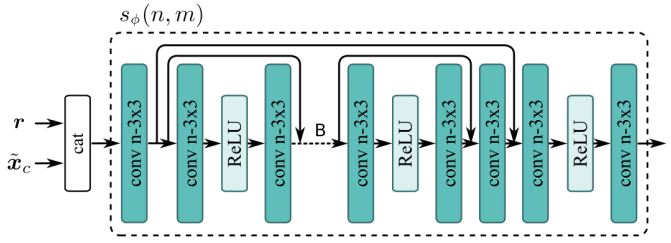
with m the true distribution of latents.

6.2.2 Network architecture

The architecture of the proposed system, illustrated in Fig. 6.2, is described in this section. Skip connections represent element-wise additions between features. Q, AE and AD stand for quantization, arithmetic encoding, and arithmetic decoding steps, respectively. We fix $n = 128$. The structure of f_θ is based on the layered autoencoder with hyperprior (AE-HP) architecture described in [218], that estimates the group of parameters $\{\boldsymbol{w}^{(k)}, \boldsymbol{\mu}^{(k)}, \boldsymbol{\sigma}^{2(k)}\}$, with $k = 3$, for the entropy model described in (6.1). We also use an autoregressive context model over latents



(a) Architecture details of the conditional autoencoder f_θ.



(b) Architecture details of the super-resolution network s_ϕ.

Figure 6.2 – Architecture and details of CAESR.

[219], denoted as C_m, to improve the entropy model accuracy without increasing the rate. The main analysis and synthesis transforms, g_a and g_s, respectively, are composed of successive self-attention and residual blocks [218]. The non-linearity is integrated using the GDN activation function [220] and LeakyReLU as described in [218]. For the hyper-encoder h_a and hyper-decoder h_s, LeakyReLU activation function is used. Regarding dimensionality reduction and expansion strided convolutional layers and sub-pixel upscaling layers [176] are implemented, respectively.

Our super-resolution module is inspired by the EDSR architecture [15] which enables state-of-the-art performance. This SR architecture mainly consists of B RBs with short and long skip connections. In this work, we fix $B = 8$ and use 64 filters of size 3×3 for each convolutional layer. We introduce the non-linearity with the ReLU activation, as described in Fig. 6.2b. We removed the upscaling layer typically located at the end of the network and perform image upscaling before passing the input picture through this module.

6.3 Enhancing video codecs with CAESR

As mentioned, downscaling-based compression can enhance the performance of single-layer codecs at low bitrate while ensuring backward compatibility with legacy receivers. This section presents how CAESR can be applied as a codec enhancer by transmitting the lost spatial information to stabilize the performance. First, details on the training process and dataset are given. Second, an objective evaluation of our method is provided. We first perform an ablation study using different configurations derived from the proposed conditional system. Then, we compare our approach with layered coding methods from the state-of-the-art, including LCEVC and SHVC. Finally, we investigate a simple temporal extension providing rate-distortion performance improvement for small GOP sizes.

6.3.1 Training procedure and dataset

Dataset

We train our model using 200 4K resolution video clips collected from the BVI-DVC dataset [183]. Each model is evaluated on several Class A videos (3840x2160) of the JVET CTC tests dataset, selected regarding their high-resolution relevancy. First, the base-layer input images $\boldsymbol{x}_{lr}$ are generated by a spatial downscale of factor 2 using a Lanczos-3 filter. The reconstructed versions $\tilde{\boldsymbol{x}}_c$ of the low-resolution images $\boldsymbol{x}_{lr}$ are then obtained using the HEVC test model HM-16.20 for different QPs. All models only consider the luma component in the EL processing to concentrate the bitrate on the most relevant visual information, i.e., $\boldsymbol{x}_c$ and $\boldsymbol{x} \in \mathbb{R}^{W \times H \times 1}$. The chroma components of the output image $\hat{x}_c$ are obtained by a bicubic upscale applied on the BL image $\tilde{x}_c$. For training, we crop 256×256 high-resolution and corresponding 128×128 low-resolution patches from the training set, resulting in around 150K training pairs.

Training

Both the super-resolution and autoencoder networks are jointly trained to minimize the rate-distortion loss defined in (6.3). We train one model per base-layer $QP \in \{37, 32, 27, 22\}$ and select specific λ values in (6.3). As the base quality is starting to saturate at higher bitrate, we empirically decided to allocate more bitrate for the lower BL QPs. The models are trained over 20 epochs with a learning rate of 10^{-4}. We apply a learning rate decay with a gamma of 0.5 for the last 5 epochs to improve the convergence. We use a batch size of 4 and optimize the model with ADAM [80] by setting $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. For the whole experiments, the

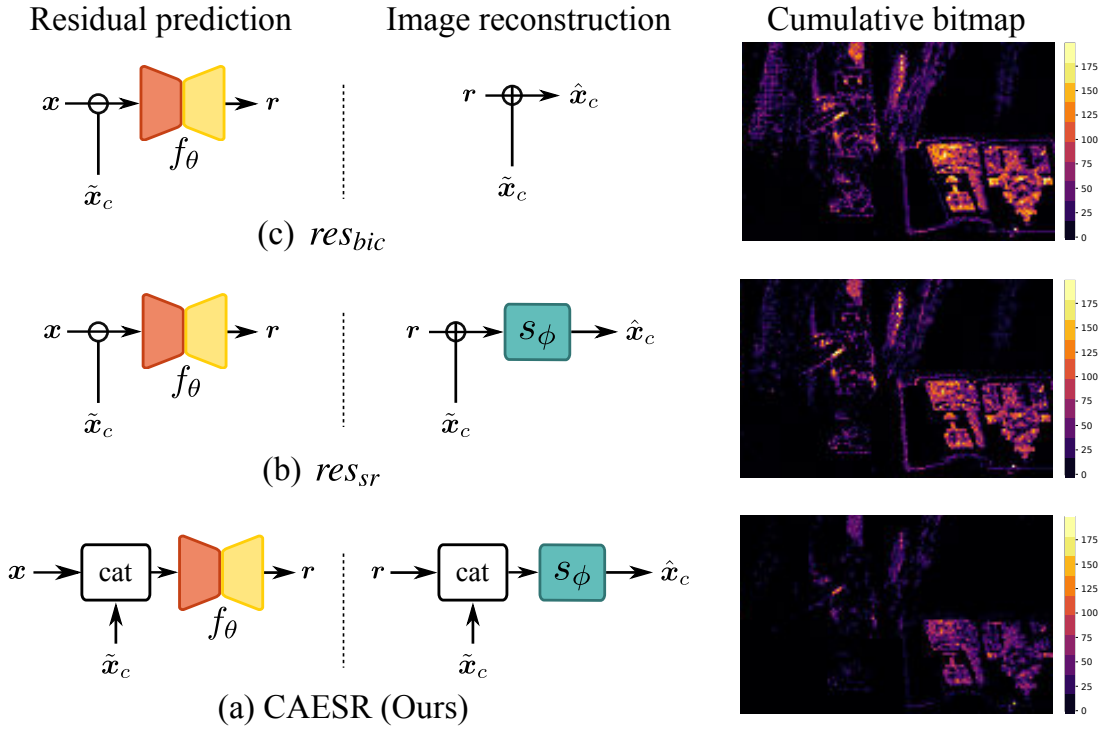


Figure 6.3 – Visualization of the configurations tested during ablation using *CatRobot1* encoded with HM-16.20 AI (qp22).

quality is assessed on the luma component using PSNR full-reference objective image quality metrics computed between the reconstructed images $\hat{\boldsymbol{x}}_c$ and original images $\boldsymbol{x}$.

6.3.2 Experimental results

Ablation study

In this experiment, we demonstrate the effectiveness of the proposed system through an ablation study. This test is performed using the all intra (AI) configuration of the HM-16.20. The models that use our EL module f_θ, including the proposed conditional coding system CAESR and the residual-based configurations with and without super-resolution, represented by res_{sr} and res_{bic}, respectively, are illustrated in Fig. 6.3. This experiment also considers configurations based on our super-resolution module s_ϕ and a bicubic interpolation filter used as post-processing modules, represented by *sr* and *bic*, respectively. The whole learned models are optimized using the training strategy described in Section 6.3.1.

The left part of Figure 6.3 represents residual prediction and image reconstruction steps for the different configurations used for the ablation study. We display cumulative bitmaps obtained

with the different tested models in the right part of Figure 6.3. We observe that the configurations that include super-resolution, i.e., (a) and (b), produce more sparse latent variables that require fewer bits for enhancement layer encoding. The joint training of the super-resolution module s_ϕ and the autoencoder f_θ allows an optimal interaction between the two models. Therefore, the autoencoder f_θ omits high frequencies that the super-resolution module s_ϕ can recover, allowing the autoencoder f_θ to focus on the most complex areas.

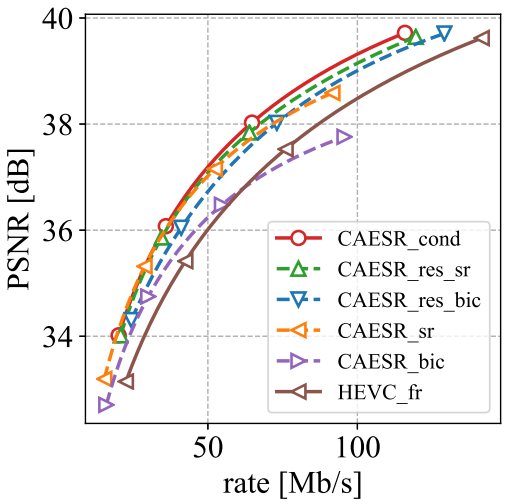


Figure 6.4 – Average performance of the tested configurations on the Class A videos from the JVET CTCs dataset [23].

The RD curves are represented in Fig. 6.4. We display incomplete systems for ablation study in dashed lines. We also add a full-resolution single layer HEVC configuration, corresponding to the high-resolution images encoded with HEVC HM-16.20 all-intra mode. We compute the global rate for configurations that consider an EL signal by the bitrates of the BL and EL signals. First, we notice that the configurations including both the autoencoder f_θ and the SR module s_ϕ in the EL, are more efficient than the others, particularly at higher bitrates. Indeed, in this bitrate range, the reconstructed residual information contains high-resolution details that cannot be recovered using a single post-processing module. Although the residual bicubic configuration, i.e., (c) in Fig. 6.3, offers lower performance, this experiment demonstrates that at a high bitrate, transmitting the residual computed from a bicubic upscale with our system offers gains in PSNR over super-resolution used as a post-processing module.

Table 6.1 – Performance SHVC, LCEVC and CAESR regarding different sequences using HM-16.20 full-resolution coding as anchor. The values in bracket indicates the BD-rate using their respective upscaled BL as anchor.

Method	SHVC		LCEVC		Ours	
BD-rate (%)	(PSNR)	(VMAF)	(PSNR)	(VMAF)	(PSNR)	(VMAF)
Campfire	+16.31 -	+18.75 -	-14.48 [-1.12]	-24.05 [+0.82]	**-23.87** **[-2.63]**	**-27.85** **[-1.05]**
CatRobot1	+30.10 -	+28.63 -	+1.36 [+1.48]	-12.89 [+3.45]	**-10.65** **[+0.46]**	**-17.88** **[+2.48]**
ParkRunning3	+11.16 -	+11.40 -	-23.70 [+0.91]	-32.81 [+2.78]	**-28.08** **[-0.71]**	**-33.81** **[+0.91]**
DaylightRoad2	+30.53 -	+29.31 -	+6.02 [+1.91]	-7.80 [+3.39]	**-5.66** **[+0.64]**	**-14.67** **[+2.06]**
Average	+22.03 -	+22,02 -	-7.70 [+0.51]	-19.25 [+2.61]	**-18.38** **[-0.70]**	**-23.54** **[+1.10]**

Model performance

This experiment evaluates CAESR against state-of-the-art layered coding approaches, including LCEVC and SHVC. All the tested methods are assessed using HEVC as a BL codec. For both CAESR and LCEVC, the BL is encoded using the HM-16.20 in RA mode with $QP \in \{22, 27, 32, 37\}$. For LCEVC, the EL is generated with the LCEVC test model (LTM-4.0) using specific quantization parameters regarding the BL QPs given in [221]. As for CAESR, we enable the luma restriction in the configuration of the LTM-4.0 [1] to provide a fair comparison. For SHVC, we use the SHM-9.0 in RA mode with $QP \in \{27, 32, 37, 42\}$ to match the bitrate with CAESR and LCEVC. All approaches are compared against HEVC full-resolution coding using HM-16.20 as an anchor with $QP \in \{27, 32, 37, 42\}$.

The RD curves for both CAESR and LCEVC are depicted in Figure 6.5. For both algorithms, we represent the performance of the upscaled BL without enhancement to highlight the EL contribution to the global performance. We also compare both approaches with HEVC full-resolution coding in terms of BD-rate performance regarding PSNR and VMAF. The results are summarized in Table 6.1 for all the tested sequences. We notice that both LCEVC and CAESR outperform SHVC on the assessed bitrate range. Indeed, as mentioned in the state-of-the-art, scalability always generates bitrate overhead. Compared to full-resolution coding, the results demontrates that CAESR and LCEVC allow a bitrate reduction of 18.38% and 7.7% regarding

1. $num\_processed\_planes = 1$

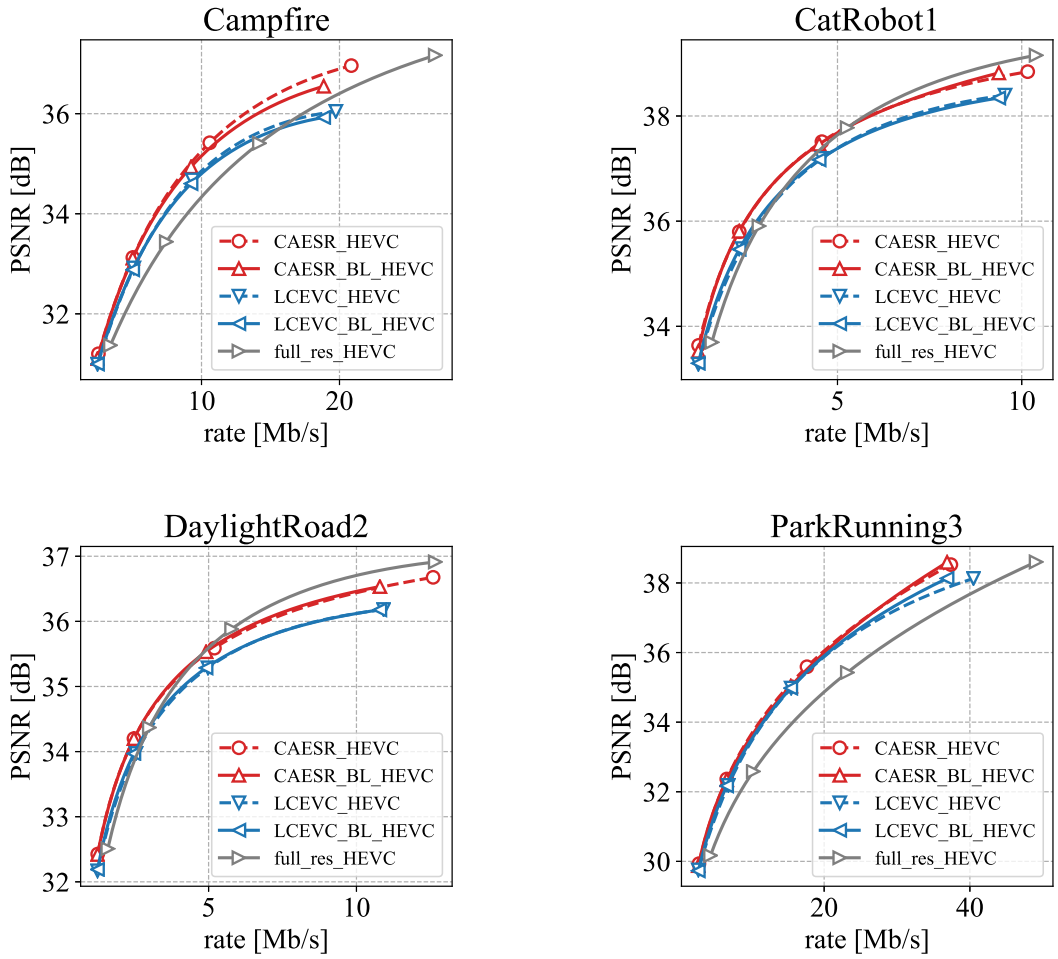


Figure 6.5 – RD-curves for objective comparison with state-of-the-art.

the same PSNR, respectively, while offering backward compatibility.

However, we notice that the performance of LCEVC and CAESR compared to their respective upscaled BL depends on the sequence. For instance, the EL of CAESR offers 2.63% of bitrate gains regarding the sequence *Campfire*, against a loss of 0.64% regarding the sequence *DaylightRoad2*. To better analyze these results, we display the bitrate allocation for both the BL and the EL of CAESR in Figure 6.6. On the one hand, we see that P and B frames of the BL have a low contribution to the global rate for the sequences *CatRobot1* and *DaylightRoad2*, while the EL bitrate remains constant for all frames. Indeed, these sequences contain temporal redundancy that is exploited by the hierarchical structure of the BL encoder. Thus, by comparing the performance of the tested approaches with their respective upscaled BL, we notice that no gains are provided for the sequences *CatRobot1* and *DaylightRoad2*. Although LCEVC provides

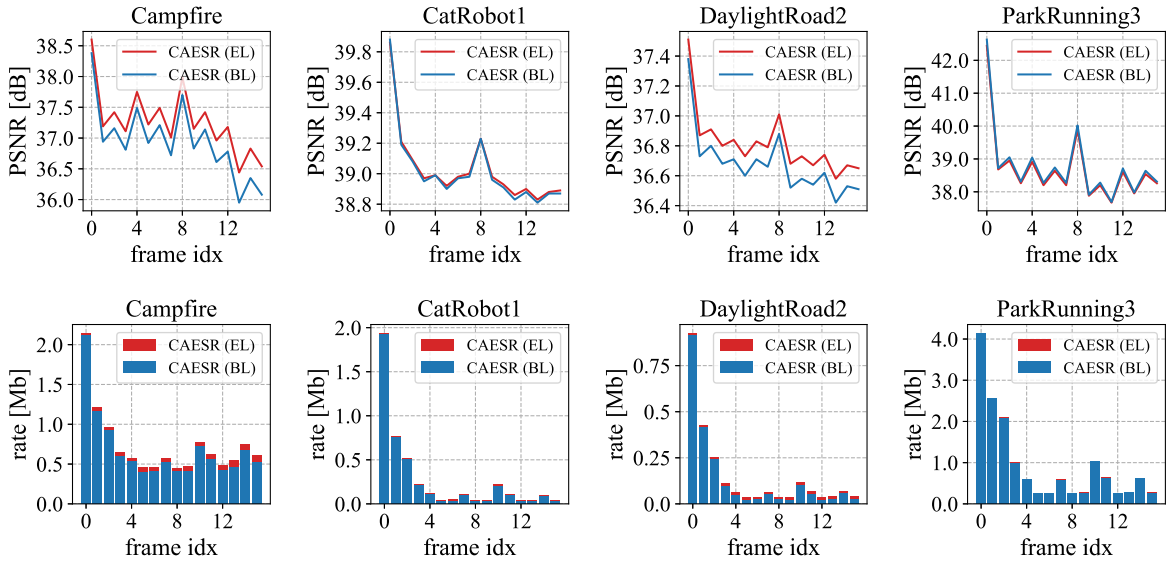


Figure 6.6 – Per-frame rate allocation analysis.

temporal processing of the residual, it is not sufficient to efficiently exploit this redundancy in the residual space. On the other hand, gains are offered for the sequence *Campfire*, where P and B frames have a higher contribution to the global rate. Indeed, this sequence contains random motion, which is hard to predict by the BL codec. Thus, a large part of the processed samples of the BL is intra predicted. Although CAESR provides the best performance for all the tested sequences, this observation limits the rate allocation for the EL and, thus, the performance compared to upscaling applied as post-processing.

Temporal extension

In this experiment, we explore a simple temporal extension for CAESR to evaluate the potential of temporal redundancy reduction in the residual space. As conditional coding can provide more than one frame to the autoencoder, we feed both the autoencoder f_θ and the super-resolution model s_ϕ with the previously decoded frame x^{t-1} without motion compensation, as shown in Figure 6.7. Thus, we train two models to perform the inter configuration. The I frame is encoded using a first model trained as described in Section 6.2.1 and the P frames using another model trained as described in Figure 6.7. Each training iteration consists of encoding and decoding one P frame based on the previously reconstructed I frame. During inference, we form GOP containing one I frame and several P frames. An illustration of the bitmap for different GOP sizes for the EL is given in Figure 6.9. It shows that fixed zones that are costly in bitrate (because

the camera better captures no-motion zones) are well captured by the temporal extension. We illustrate the evolution of the RD loss regarding different size of GOP in Figure 6.8. We notice that the temporal extension lowers the RD loss for all sequences regarding small GOP sizes. However, a GOP size superior to 4 generates error propagation as the RD loss increases.

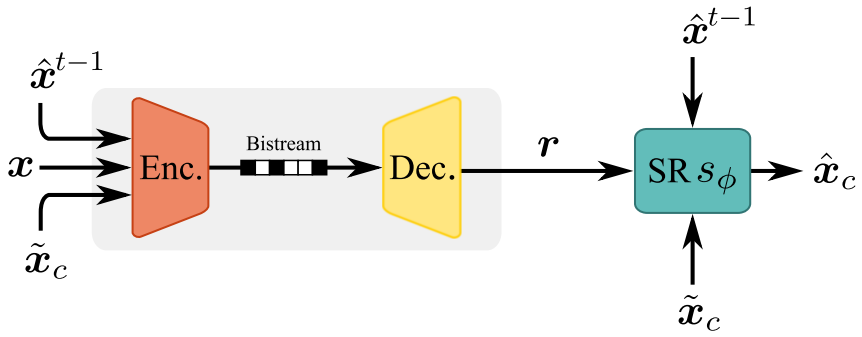


Figure 6.7 – Illustration of the temporal extension of CAESR.

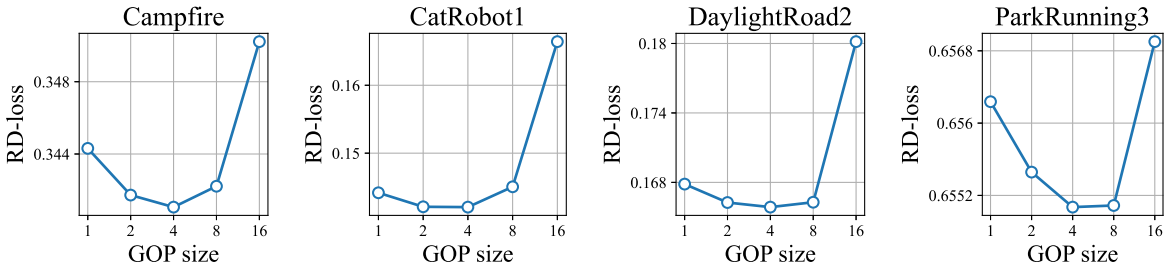


Figure 6.8 – RD loss regarding different GOP sizes.

6.3.3 Analysis and discussion

This section evaluates CAESR as a codec enhancement method using HM-16.20 as a BL codec. We demonstrated that the conditional coding process offers better average performance than residual coding in AI mode. In RA mode, the performance of our approach depends on the type of sequences. We show that the metadata can help stabilize the downscaling-based compression method's performance for sequences with low temporal redundancy. We show that our approach outperforms both LCEVC and SHVC in terms of BD-rate performance for all sequences. However, our approach cannot enhance the upscaled BL signal for low-motion sequences. Finally, we show that a simple temporal layer can improve the performance of CAESR for this type of scene. However, this temporal extension is limited to a small GOP size due to error propagation.

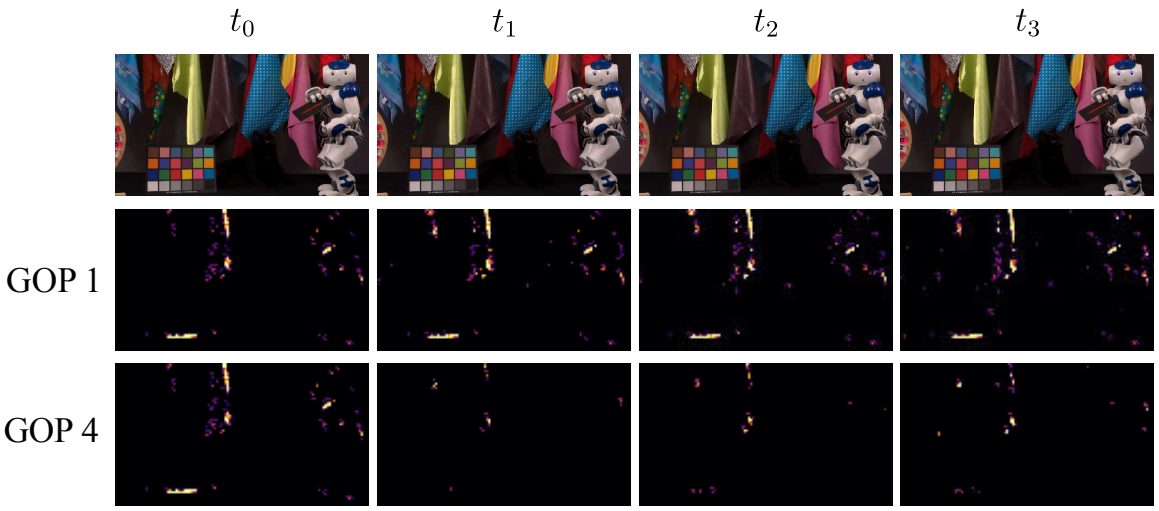


Figure 6.9 – Visualization of the bitmaps regarding different GOP sizes.

6.4 Deploying new video services with CAESR

The initial target of this manuscript was to deploy higher resolution services, e.g. 8K, from already deployed services, e.g. 4K, while avoiding using simulcast. In this context, the layered structure of CAESR can bring flexibility by allowing the delivery of the BL and the EL in separate channels. For instance, the BL can be transmitted over broadcast DTT network, while the EL can be distributed over broadband internet protocol (IP) network. This section evaluates our approach compared to an HEVC simulcast (HD/4K) using typical DTT bitrates for HD with HEVC: 2.5Mbps, 5Mbps, 7.5Mbps, and 10Mbps. We also compare the performance of our method with LCEVC assessed in similar conditions.

6.4.1 Training procedure and dataset

Baseline

For this experiment, we use another training strategy to optimize our model. As mentioned in the introduction, lowering the resolution leads to fewer compression artifacts in the reconstructed low-resolution image due to an increased bit-per-pixel ratio. However, the downscaling step generates a loss in high-resolution details, which degrades the quality. Thus, we separate the two types of degradations by training our model to recover scaling residuals only. An illustration of the training pipeline for this experiment is given in Figure 6.10. The reconstructed residuals are then applied to the decoded BL during inference, as described in Section 6.3.1. It also allows simpler control of the bitrate for the EL, as it does not depends on the BL quality. In

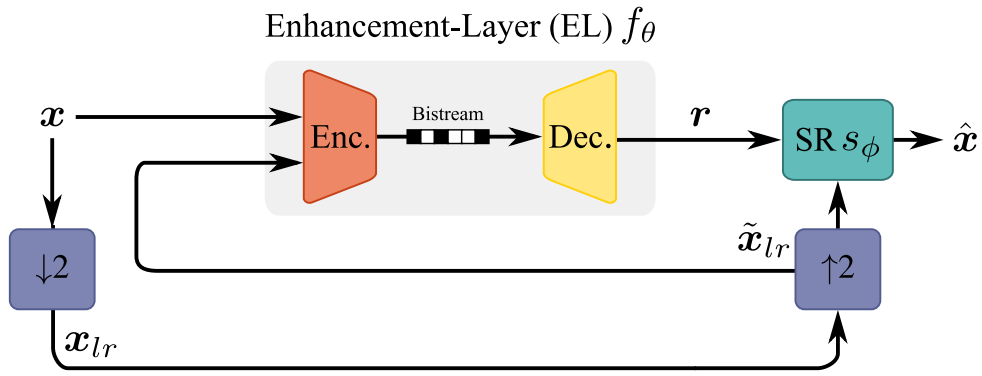


Figure 6.10 – Description of the training phase of CAESR for scaling residuals.

this section, we train two versions of the proposed system, i.e., (a) and (c) in Figure 6.3. In addition to CAESR based on conditional coding and super-resolution, it allows comparing our approach with LCEVC using the same upscaling filter for the EL residual input. Each baseline is trained for two different values of lambda, selected to cover bitrates in the range of 0 to 10Mbps approximately. Regarding LCEVC, we tune the quantization parameter to cover a similar bitrate range.

Dataset

As in Section 6.3, we use the 4K sequences from the BVI-DVC dataset to train our models. Similarly, all systems are trained to recover the luma component only, and the chroma components are upscaled using a bicubic filter. The base-layer input images x_{lr} are generated by a spatial downscale of factor 2 using a Lanczos-3 filter to match with the downscaling filter of LCEVC. For training, we crop 256×256 high-resolution and corresponding 128×128 low-resolution patches from the training set, resulting in around 150K training pairs. In this section, we used a double-pass rate control algorithm of x265 to encode the HD base layer signal using typical bitrates of DTT broadcast: 2.5Mbps, 5Mbps, 7.5Mbps, and 10Mbps.

Training

All models are trained over a total of 20 epochs with a learning rate of 10^-4 . We apply a learning rate decay with a gamma of 0.5 for the last 5 epochs to improve the convergence. For the conditional coding model, we freeze the gradients of the autoencoder f_θ and finetune the super-resolution model s_ϕ for 3 additional epochs by replacing the low-resolution images $\tilde{x}_l r$ in Figure 6.10 with compressed low-resolution images $\tilde{x}_c$. It allows training the super-resolution

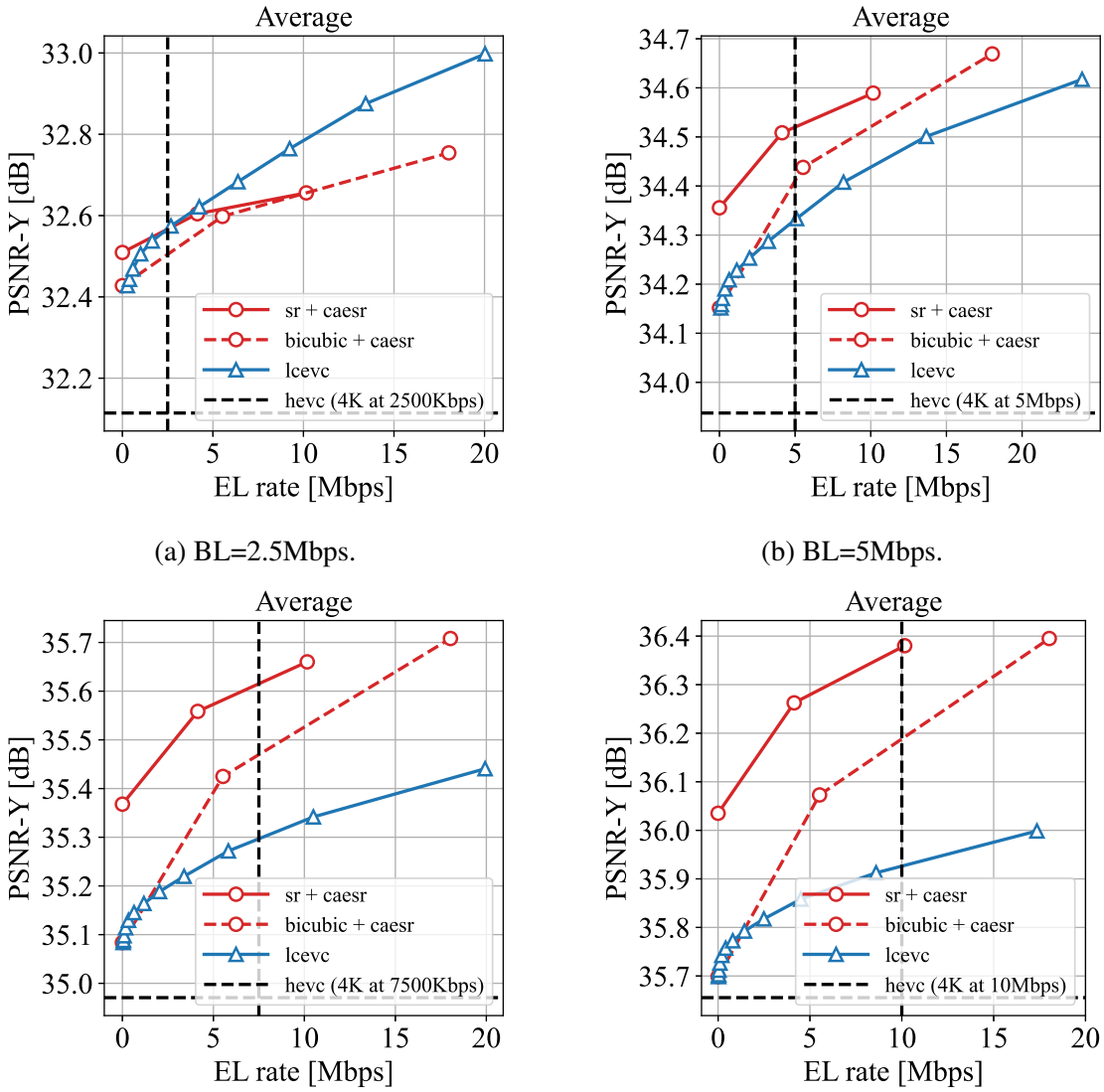
model s_ϕ on compressed images to fit with the compressed low-resolution images $\tilde{x}_c$ during inference. We use a batch size of 4 and optimize the model with adaptive moment estimation (ADAM) [80] by setting $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\gamma = 10^{-}8$. For the whole experiment, the quality is assessed on the luma component using the PSNR.

6.4.2 Experimental results

RD-comparison

The average performances of the tested configurations are represented in Figure 6.11. These curves show the quality in PSNR of each method regarding different rate allocations for the EL. Each graph represents the results obtained for different BL's bitrate. The results validate that the super-resolution and bicubic configurations provide similar performance at low-bitrate. As the EL's bitrate increases, both residual and conditional configurations converge. Compared with LCEVC, our approach performs better for higher bitrates, where fewer degradations related to compression are represented in the BL. Indeed, LCEVC includes the codec's degradations in the residual computed to enhance the upscaled BL. On the contrary, we only consider scaling artifacts in this experiment for assessing our approach. Thus, for low rates, LCEVC provides better PSNR.

In the following, a per-sequence analysis is provided. The performances are represented in Figure 6.12 for BL bitrates of 5Mbps and 7.5Mbps. These graphs depict the performances of the tested configurations for different EL rate allocations and different BL rates. We see that our approach outperforms LCEVC for all sequences regarding the higher BL rate. Regarding the CatRobot1 scene, we see that at low bitrate, LCEVC provides a slightly higher PSNR than our approach. As the camera is fixed in *CatRobot1*, LCEVC can handle the no-motion details present in the scene thanks to its simple temporal layer. This behavior is not observed for the *DaylightRoad2* and *Campfire* scenes, containing moving elements. Thus, the global motion is high and cannot be handled by the no-motion vector temporal layer of LCEVC. At low bitrate, e.g. BL=2.5Mbps, the observation LCEVC outperforms CAESR for the sequence *Campfire*. Indeed, this sequence is harder to encode than the two other, and a large amount of coding degradation is generated at 2.5Mbps, which are handled by LCEVC. Compared to simulcast, the super-resolution filter provides better PSNR for all sequences and all bitrates.



(a) BL=2.5Mbps.

(b) BL=5Mbps.

(c) BL=7.5Mbps.

(d) BL=10Mbps.

Figure 6.11 – Average RD curves over the selected 4K sequences.

Vizualisation

In this experiment, we visually compare the different tested methods. All approaches consider the same bitrate for both the BL and the EL. For instance, in Figure 6.13 (a), all layered methods are based on a BL (HD) encoded at 5Mbps, and an EL (4K) encoded at 5Mbps. For the super-resolution configuration, i.e., SR in Figure 6.13, the bitrate is assessed on the BL (HD) only. The objective is to demonstrate that coding artifacts occur, i;e., blocking, and ringing, when deploying 4K using simulcast at certain bitrates. For instance, block artifacts are not present

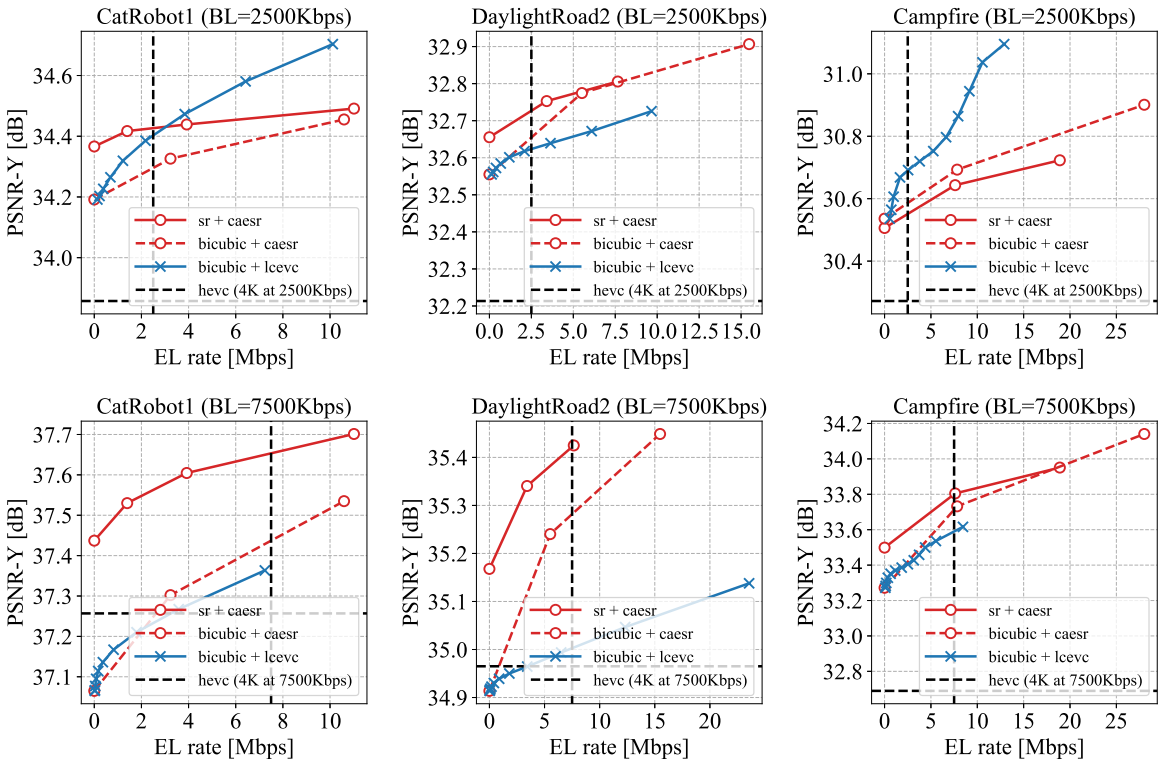
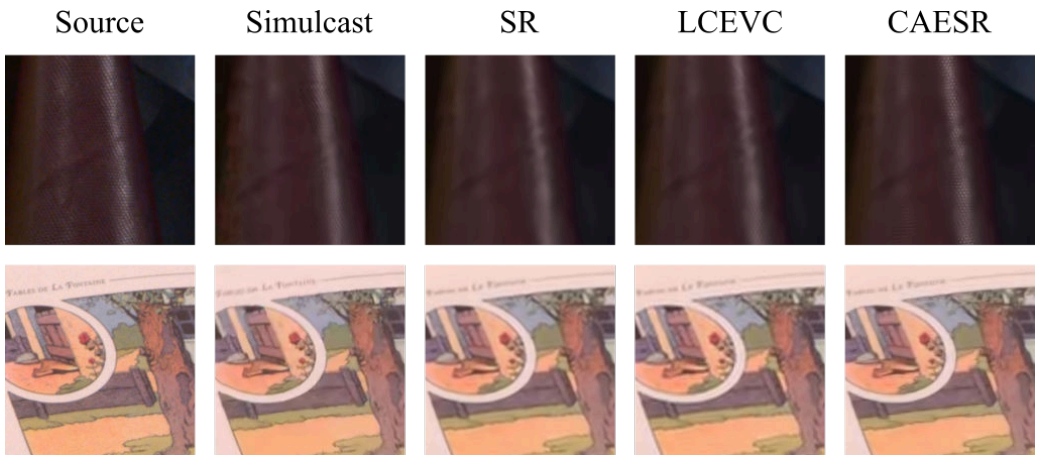


Figure 6.12 – RD curves over the selected 4K sequences.

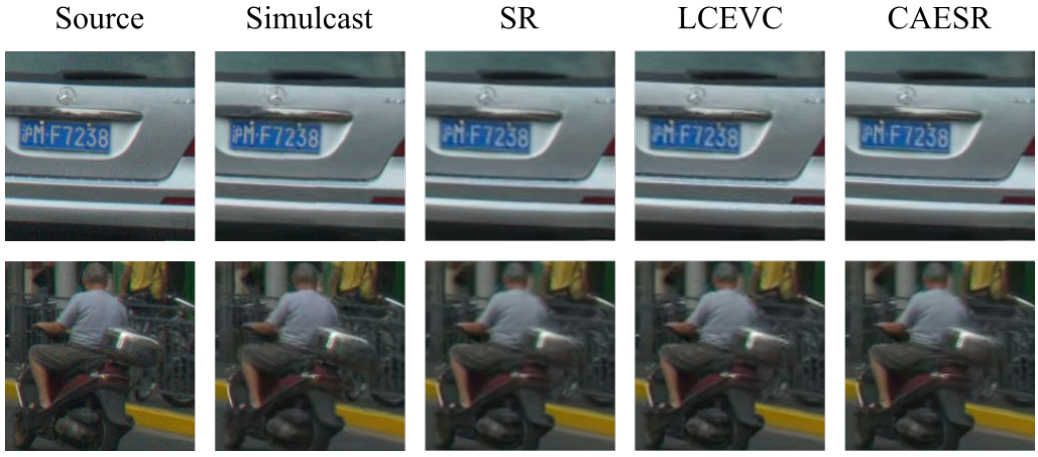
at this bitrate in the sequence *Campfire* by lowering the resolution before encoding. However, high frequencies are lost regarding the super-resolution used as a post-processing configuration. The results show that high frequencies are recovered thanks to our approach, e.g. textures and textures. We observe that LCEVC fails to reconstruct complex textures like the scarf in the sequence *CatRobot1* and the license plate of *DaylightRoad*, whereas CAESR succeeds.

6.4.3 Analysis and discussion

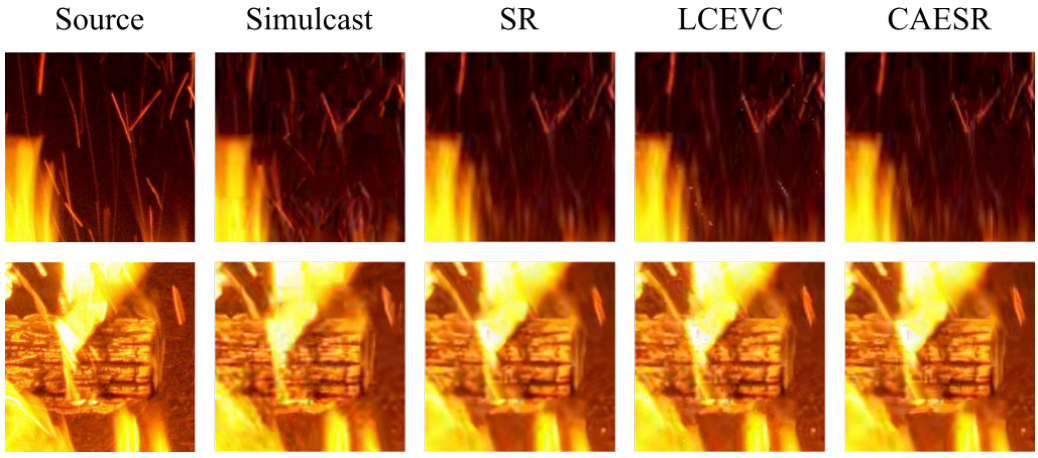
In this section, CAESR has been evaluated in the context of new video services deployment. For this use case, we modified the initial training process of CAESR by restricting the training for scaling residual recovering. We used a double-pass rate control algorithm for generating the BL regarding different bitrates used for HD broadcast on DTT. We assessed two versions of our approaches based on different upscaling methods, i.e., a bicubic filter and the super-resolution model as defined in Section 6.3. Our approach outperforms LCEVC for most BL bitrate regarding most sequences, even using a bicubic filter as BL upscaling method. Moreover, we show that the bicubic-based configuration performs better than the super-resolution after a certain bitrate



(a) *CatRobot1* 5Mbps .



(b) *DaylightRoad2* 7.5Mbps.



(c) *Campfire* 7.5Mbps.

Figure 6.13 – Visualisation of the reconstructed frame using different layered methods.

reach. Finally, our approach outperforms simulcast by considering less bitrate regarding each BL bitrates and for each tested sequence.

6.5 Conclusion

This chapter presents CAESR, a codec agnostic learning-based approach for layered video coding. The proposed solution is based on the joint training of a conditional autoencoder f_θ and a CNN super-resolution module s_ϕ. The deep autoencoder with hyperprior, learns to represent the residual information that cannot be recovered by the super-resolution module used as a post-processing step. This residual information is combined with the upscaled base-layer reconstruction at the decoder side to form the high-resolution output signal. Our approach relies on conditional coding that learns the optimal mixture of the source and the upscaled image, enabling better performance than residual coding.

Two contexts of evaluation for the proposed solution have been considered in this chapter. On the one hand, we assessed CAESR as a codec enhancement method. The results demonstrate that, for some sequences, CAESR improves downscaling-based compression performance with HEVC by transmitting additional metadata, allowing to recover high-resolution details on the receiver side. We demonstrate that our approach offers better performance than the state-of-the-art. However, we observe that our solution is restricted to a low proportion of the total bitrate allocation, which limits the performance. On the other hand, CAESR has been evaluated to deploy new video services. By transmitting additional information, the proposed solution offers the potential to deploy new video services, e.g. 8K, from already deployed ones, e.g. 4K, by transmitting an additional stream on the same or different transmission channel. We demonstrated that CAESR can be used recovers the scaling residuals at the receiver side and offers better performance than LCEVC and HEVC simulcast.

However, several tracks remain in this work. First, each CAESR operating points are the product of separate models trained with different λ in Equation 6.3. As the value of λ defines the slope on the RD curve, bitrate is allocated depending on the sequences, which is not practical. Thus, it is necessary to be able to perform variable bitrate with the selected models. Some architectures propose to apply a gain value before quantization to expend or reduce the latent vector values, varying the effect of the round quantization on latents [222]. However, the model is not optimized for a given RD tradeoff, which reduce the interest of end-to-end compression. Furthermore, a rate allocation strategy has to be defined to properly allocate the rate regarding a given granularity, e.g. frame-level or GOP-level. Regarding the BL codec, we selected HEVC

in this work due to the defined use case. However, our approach can be applied to any BL codec, including learned systems. Using a learned system as a BL codec would enable the joint training of both the BL and the EL regarding the loss function. Moreover, visualization of video reconstructed with CAESR demonstrated that our approach suffers from temporal consistency artifacts that can be annoying for the end-user. Advanced architectures or loss functions could be considered to tackle this problem. Finally, although graphic processing unit (GPU) acceleration allows an efficient parallelization of CNNs, our approach is highly complex. This aspect has to be considered as the decoding, and super-resolution processes are performed on the decoder side.

PART IV

Conclusion

CHAPTER 7

CONCLUSION

This thesis explores multiple post-processing and compression methods for 8K video services deployment on terrestrial networks. First, we evaluated several compression scenarios for 8K delivery with 4K backward compatibility and performed the world's first subjective test of VVC for 8K video coding. We demonstrated that spatial upscaling using super-resolution enables 8K video reconstruction with 4K backward compatibility in a codec agnostic way and without additional bitstream. Second, we designed super-resolution models based on multitask learning and investigated adapted training strategies to improve the performance of super-resolution applied to compressed videos. Finally, we developed a layered architecture where the enhancement layer is jointly trained with a super-resolution model to recover missing spatial information on the receiver side. The remainder of this section is as follows. First, Section 7.1 reminds the objectives of this thesis. Then, Section 7.2 highlights our work's contributions and contextualizes them regarding the initial targets. Finally, Section 7.3 draws the perspectives for future works related to this manuscript.

7.1 Thesis objectives

8K video format has recently encountered a huge interest from the industry with advances in hardwares (TV screens, captors) and experimental tests like in Japan with the NHK. From the perspective of terrestrial broadcast, the delivery of 8K must consider backward compatibility with legacy receivers to keep the audiance reach. However, simulcasting 8K with lower resolutions is inconceivable due to the high bandwidth requirements of this media format. The initial target of this thesis was to prepare the arrival of 8K resolution on the Hertzian transmission platforms by investigating new compression and post-processing algorithms to recover an 8K video signal from already deployed video services. Several tracks were possible such as contributions in scalable standards such as SHVC or scalability in VVC. However, due to a shifted standardization timeline and a late integration, spatial scalability is not much present in the current broadcast ecosystem. As another option, spatial upscaling allows recovering higher-resolution from lower

ones without additional bitstreams, which could offer 8K reconstruction on top of a 4K video stream. Recently, super-resolution algorithms based on deep learning architectures have shown outstanding performance compared to conventional state-of-the-art approach. Thus, we focused on investigating and designing new super-resolution algorithms for compressed video upscaling.

7.2 Achived work

The first part of this thesis, described in Chapter 4, investigates several state-of-the-art algorithms for 8K video delivery. First, we subjectively evaluated the two latest single-layer MPEG standards for 8K video compression. The results demonstrated that VVC offers around 40% of bitrate reduction for the same perceived quality compared to HEVC. We have shown that the bitrate required to reach transparency varies from 11 to 180Mbps, depending on the content. Moreover, we observed that 8K provides QoE improvement over 4K for most tested sequences. This test is the world's first subjective assessment of VVC for 8K resolution videos and has been published in a scientific journal [223] and presented in standardization bodies, including DVB [224] and MPEG [225]. The second axis of this chapter was to evaluate different compression scenarios for 8K video delivery with 4K backward compatibility. We observed that delivering a downscaled representation of the 8K video stream and rescaling it after decoding using super-resolution provides better performances than full resolution coding at low bitrate while providing backward-compatibility. However, the performances of this kind of system are content dependent as some high frequencies are lost during the downscaling operation performed before encoding. Moreover, the performance of super-resolution is on par with a Lanczos filter at low bitrate, which limits the interest in using such a complex approach. These results have been published and presented at an international conference [194] and the IBC show 2020 [226].

Chapter 5 investigated new architecture and dedicated training strategies to improve super-resolution models for compressed video upscaling. In the first part, we proposed MTL-EDSR, a multitask network that simultaneously performs the tasks of super-resolution and quality-enhancement of the compressed input image. This network also uses prior information from the compressed bitstream and a pretraining strategy to further improve the performance of super-resolution on compression input images. The results demonstrate that both tasks learn similar features that can be mutualized to reduce the total number of learned parameters in a context where both tasks are performed. For instance, such an architecutre could be integrated inside a scalable codec to perform both upscaling for ILP and enhancement of the BL signal using a single shared network. This work has been presented at an international conference [227]. In extension

to this work, we couple super-resolution with high-level vision tasks, i.e., non-reference quality assessment, and semantic segmentation, to force the network to learn non-explicit features. The developed method, called MTL-Unet, is based Unet to enable the high abstraction required for the additional high-level vision tasks. The results show that the different tasks can share their representation with a low impact on the super-resolution performance. Although being helpful in a compression framework, the features learned by the tested additional tasks do not improve the main task of super-resolution.

The last part of this thesis, described in Chapter 6, investigates the transmission of additional information to recover lost high-resolution details on the receiver side. The proposed solution, called CAESR, is a layered approach based on a conditional autoencoder used as an enhancement layer jointly trained with a super-resolution model. The deep autoencoder learns to represent the residual information that cannot be recovered by the super-resolution module used as a post-processing step. This residual information is combined with the upscaled base-layer reconstruction at the decoder side to form the high-resolution output signal. In the first axis, we have evaluated the proposed system for HEVC enhancement in a downscaling-based compression system boosted by metadata. The results demonstrate that our approach outperforms LCEVC for HEVC enhancement of around 10% of bitrate reduction regarding the same PSNR. However, the performance of this approach is content-dependent, which limits the rate allocation for the EL. This work has been published at an international conference [228] and is protected by a patent. In addition, the layered architecture of the proposed system brings flexibility by splitting the video information into different streams. Thus, both the enhancement and base layers signals can be transmitted in separate channels. For instance, the base layer can be transmitted through DTT and the enhancement layer using a broadband network such as IP. In a second axis, we evaluate CAESR for 8K services deployment. The results demonstrated that for bitrates typically used in french DTT broadcast, the proposed approach offers better performance than HEVC simulcast and LCEVC evaluated in similar test conditions.

To summarize, this work successfully addresses the targets of this thesis by providing new solutions to enable the initial use case. We first exhaustively studied existing algorithms for 8K video compression by conducting the world's first 8K VVC subjective test. We provided solid scientific contributions to international conferences and standardization bodies like DVB and MPEG. We developed new approaches to improve the performance of super-resolution for video compression, such as super-resolution based on multitask learning and jointly trained with a learned enhancement layer. We developed a new innovative solution based on the joint training of an autoencoder and a super-resolution network. We compared the proposed approach

with state-of-the-art methods regarding several contexts, including the defined use case. All the contributions provided during this thesis are detailed in appendix A.

7.3 Future works

7.3.1 Multitask Learning for Super-Resolution of Compressed Videos

Regarding the architecture of multitask models for super-resolution, several tracks remain for improvement. The temporal aspect could be investigated to recover missing details in the already enhanced frames, focusing on the additional complexity of such a method [213, 214]. The multitask architecture could be considered to perform more than two tasks with a single shared network. Also, soft-sharing has been investigated using a cross-stitch algorithm [22], which increases the number of parameters proportionally to the number of performed tasks. Thus, the performances are lower when compared to specialized algorithms using the same total number of parameters. More recent approaches using attentions modules to select shared features could allow soft-sharing without consequently increasing the total number of parameters [188]. Finally, investigating richer information from the bitstream, such as the prediction information [215], could further improve the contribution of prior information to global performance.

7.3.2 Learning-Based Video Coding for Efficient Layered Compression

The most promising contribution of this thesis is the transmission of high-resolution information in an additional stream of metadata. However, several tracks remain in this work. First, variable bitrate must be considered to avoid training multiple networks to target different bitrate targets. Furthermore, a rate allocation strategy must be defined to properly allocate the rate regarding a given granularity, e.g., frame-level or GOP-level. In addition, the optimal rate allocation between the BL and the EL has to be defined to optimize the overall system's perfomance. This work investigated a simple temporal layer for the EL and demonstrated that it could improve the RD performance. However, there is room for improvement to avoid error propagation which prevents the use of larger GOP sizes. Also, the optimal bitrate for the EL has to be investigated regarding the BL rate must optimize the overall delivery. Moreover, using a learned system as a BL codec would enable the joint training of the BL and the EL regarding the loss function. Investigation on learned downscaling, i.e., CR, can also be considered to jointly train the whole processing chain. Moreover, although GPU acceleration allows an efficient parallelization of CNNs, our approach is highly complex. This aspect must be considered as

the decoding and super-resolution processes are performed on the receiver. Finally, a subjective evaluation has to be considered to validate the objective performance assessed in this work.

PART V

Appendix

APPENDIX A

PUBLICATIONS AND PATENTS

A.1 International Conferences

[C1] Bonnineau, C., Hamidouche, W., Travers, J. F., and Deforges, O. (2020, May). Versatile video coding and super-resolution for efficient delivery of 8k video with 4k backward-compatibility. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 2048-2052). IEEE.

Abstract - In this paper, we propose, through an objective study, to compare and evaluate the performance of different coding approaches allowing the delivery of an 8K video signal with 4K backward-compatibility on broadcast networks. Presented approaches include simulcast of 8K and 4K single-layer signals encoded using High-Efficiency Video Coding (HEVC) and Versatile Video Coding (VVC) standards, spatial scalability using SHVC with 4K base layer (BL) and 8K enhancement-layer (EL), and super-resolution applied on 4K VVC signal after decoding to reach 8K resolution. For up-scaling, we selected the deep-learning-based super-resolution method called Super-Resolution with Feedback Network (SRFBN) and the Lanczos interpolation filter. We show that the deep-learning-based approach achieves visual quality gain over simulcast, especially on bit-rates lower than 30Mb/s with average gain of 0.77dB, 0.015, and 7.97 for PSNR, SSIM, and VMAF, respectively and outperforms the Lanczos filter in average by 29% of BD-rate savings.

[C2] Bonnineau, C., Aubié, J., Hamidouche, W., Déforges, O., Travers, J., and Sidaty, N. An objective evaluation of codecs and post-processing tools for 8K video compression In International Broadcasting Convention IBC 2020, Amsterdam, September 2020.

Abstract - With the deployment of the latest Ultra High Definition Television (UHDTV) system, it is projected to improve the Quality of Experience (QoE) of users through the introduction of new features to the existing High Definition Television (HDTV) system, such

as High Dynamic Range (HDR), wider color gamut, High Frame-Rate (HFR) and higher spatial resolutions including 4K (3840x2160) and 8K (7680x4320). The delivery of such video formats on current broadcast infrastructures is a real challenge and r equires efficient compression methods to reach the available throughput while ensuring high video quality. On the other hand, with the progress in Deep Learning for image processing, learning-based spatial up-scalers have outperformed classical interpolation methods such as bicubic or Lanczos filters allowing a high resolution to be more accurately recovered from a lower resolution. These methods would allow the receiver to reconstruct the 8K signal while a lower resolution signal is transmitted. In this paper we propose recommendations about 8K video coding with VVC and HEVC and evaluate post-processing for this use-case through an objective study. Tested configurations include: 8K source encoded with HEVC, and 8K and 4K sources encoded with VVC and then upscaled with two methods: a Lanczos filter and a deep-learning-based Super-Resolution method called SRFBN. All configurations are tested on a set of 8K sequences using the verification model of the VVC and HEVC standard.

[C3] Bonnineau, C., Hamidouche, W., Travers, J. F., Sidaty, N., and Deforges, O. (2021, June). Multitask learning for vvc quality enhancement and super-resolution. In 2021 Picture Coding Symposium (PCS) (pp. 1-5). IEEE.

Abstract - The latest video coding standard, called versatile video coding (VVC), includes several novel and refined coding tools at different levels of the coding chain. These tools bring significant coding gains with respect to the previous standard, high efficiency video coding (HEVC). However, the encoder may still introduce visible coding artifacts, mainly caused by coding decisions applied to adjust the bitrate to the available bandwidth. Hence, pre and post-processing techniques are generally added to the coding pipeline to improve the quality of the decoded video. These methods have recently shown outstanding results compared to traditional approaches, thanks to the recent advances in deep learning. Generally, multiple neural networks are trained independently to perform different tasks, thus omitting to benefit from the redundancy that exists between the models. In this paper, we investigate a learning-based solution as a post-processing step to enhance the decoded VVC video quality. Our method relies on multitask learning to perform both quality enhancement and super-resolution using a single shared network optimized for multiple degradation levels. The proposed solution enables a good performance in both mitigating

coding artifacts and super-resolution with fewer network parameters compared to traditional specialized architectures.

[C4] Bonnineau, C., Hamidouche, W., Travers, J. F., Sidaty, N., Aubié, J. Y., and Déforges, O. (2021, December). CAESR: Conditional Autoencoder and Super-Resolution for Learned Spatial Scalability. In 2021 International Conference on Visual Communications and Image Processing (VCIP) (pp. 1-5). IEEE.

Abstract - In this paper, we present CAESR, an hybrid learning-based coding approach for spatial scalability based on the versatile video coding (VVC) standard. Our framework considers a low-resolution signal encoded with VVC intra-mode as a base-layer (BL), and a deep conditional autoencoder with hyperprior (AE-HP) as an enhancement-layer (EL) model. The EL encoder takes as inputs both the upscaled BL reconstruction and the original image. Our approach relies on conditional coding that learns the optimal mixture of the source and the upscaled BL image, enabling better performance than residual coding. On the decoder side, a super-resolution (SR) module is used to recover high-resolution details and invert the conditional coding process. Experimental results have shown that our solution is competitive with the VVC full-resolution intra coding while being scalable.

A.2 Scientific Journal

[J1] Bonnineau, C., Hamidouche, W., Fournier, J., Sidaty, N., Travers, J. F., and Déforges, O. (2022). Perceptual Quality Assessment of HEVC and VVC Standards for 8K Video. IEEE Transactions on Broadcasting.

Abstract - With the growing data consumption of emerging video applications and users' requirement for higher resolutions, up to 8K, a huge effort has been made in video compression technologies. Recently, versatile video coding (VVC) has been standardized by the moving picture expert group (MPEG), providing a significant improvement in compression performance over its predecessor high efficiency video coding (HEVC). In this paper, we provide a comparative subjective quality evaluation between VVC and HEVC standards for 8K resolution videos. In addition, we evaluate the perceived quality improvement offered by 8K over UHD 4K resolution. The compression performance of both VVC

and HEVC standards has been conducted in random access (RA) coding configuration, using their respective reference software, VVC test model (VTM-11) and HEVC test model (HM-16.20). Objective measurements, using PSNR, MS-SSIM and VMAF metrics have shown that the bitrate gains offered by VVC over HEVC for 8K video content are around 31%, 26% and 35%, respectively. Subjectively, VVC offers an average of around 41% of bitrate reduction over HEVC for the same visual quality. A compression gain of 50% has been reached for some tested video sequences regarding a Student's t-test analysis. In addition, for most tested scenes, a significant visual difference between uncompressed 4K and 8K has been noticed.

A.3 MPEG and DVB standardization contribution

[D1] Bonnineau, C., Hamidouche, W., Fournier, J., Sidaty, N., Travers, J. F., and Déforges, O. "JVET-X0186 Subjective Quality Assessment of VVC and HEVC for 8K Video Resolution", Oct, 2021.

Abstract - This contribution provides a comparative subjective quality evaluation between the VTM-11 reference software (VVC) and the HM-16.20 reference software (HEVC) for 8K resolution videos. In addition, we evaluate the perceived quality improvement offered by 8K over UHD 4K resolution. The compression performance of both VVC and HEVC standards has been conducted in random access (RA) coding configuration. This test was performed on six video scenes with various spatial and temporal characteristics collected from two different sources: the Japanese organization ITE and Fraunhofer HHI. Objective measurements using PSNR, MS-SSIM, and VMAF metrics are provided. Subjectively, VVC offers an average of around 41% of bitrate reduction over HEVC for the same visual quality. In addition, a significant visual difference between uncompressed 4K and 8K has been noticed for most tested scenes.

[D2] Bonnineau, C., Hamidouche, W., Fournier, J., Sidaty, N., Travers, J. F., and Déforges, O. "TM-AVC1256 Subjective Quality Assessment of VVC and HEVC for 8K Video Resolution", DVB, Sept, 2021.

A.4 Patents

[P1] Bonnineau, C., Hamidouche, W. and J-A. Aubié, "Procédés de décodage et de codage d'une image, dispositifs et signal associés", France, Demande FR-2106859.

BIBLIOGRAPHY

[1] ITU-R, "Recommendation BT.2020: Parameter Values for the Ultra-High Definition Television Systems for Production and International Programme Exchange," October 2015.

[2] ITU-R, "Recommendation BT.709: Parameter Values for the HDTV Standards for Production and International Programme Exchange," June 2015.

[3] B. Bross, J. Chen, J.-R. Ohm, G. J. Sullivan, and Y.-K. Wang, "Developments in international video coding standardization after avc, with an overview of versatile video coding (vvc)," *Proceedings of the IEEE*, pp. 1–31, 2021.

[4] ITU-T, "Recommendation H.265: High Efficiency Video Coding," August 2021.

[5] "Itu-r BT.2343-7 - Collection of field trials of UHDTV over DTT networks," 2021.

[6] F. Zhang, M. Afonso, and D. R. Bull, "ViSTRA2: Video Coding using Spatial Resolution and Effective Bit Depth Adaptation," *Signal Processing: Image Communication*, vol. 97, p. 116355, Sept. 2021. arXiv: 1911.02833.

[7] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end Optimized Image Compression," *arXiv:1611.01704 [cs, math]*, Mar. 2017. arXiv: 1611.01704.

[8] "Information tecnology - General video coding - Part 2: Low Complexity Enhancement Video Coding," standard, International Organization for Standardization, Geneva, CH, november 2021.

[9] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, "Overview of shvc: Scalable extensions of the high efficiency video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 20–34, 2015.

[10] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European conference on computer vision*, pp. 184–199, Springer, 2014.

[11] ITU-R, "Recommendation BT.500-14: Methodologies for the Subjective Assessment of the Quality of Television Images," October 2019.

[12] X. Li, S. Sun, Z. Zhang, and Z. Chen, "Multi-scale grouped dense network for vvc intra coding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 158–159, 2020.

[13] A. A. Zhi Li, "Toward a Practical Perceptual Video Quality Metric," in *Netflix TechBlog*, June 2016.

[14] C. E. Duchon, "Lanczos filtering in one and two dimensions," *Journal of applied meteorology*, vol. 18, no. 8, pp. 1016–1022, 1979.

[15] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 136–144, 2017.

[16] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[17] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015.

[18] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced Deep Residual Networks for Single Image Super-Resolution," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, (Honolulu, HI, USA), pp. 1132–1140, IEEE, July 2017.

[19] K. Yu, C. Dong, C. C. Loy, and X. Tang, "Deep Convolution Networks for Compression Artifacts Reduction," *arXiv:1608.02778 [cs]*, Aug. 2016. arXiv: 1608.02778.

[20] D. Minnen, J. Ballé, and G. Toderici, "Joint Autoregressive and Hierarchical Priors for Learned Image Compression," *arXiv:1809.02736 [cs]*, Sept. 2018. arXiv: 1809.02736.

[21] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned Image Compression with Discretized Gaussian Mixture Likelihoods and Attention Modules," *arXiv:2001.01568 [eess]*, Mar. 2020. arXiv: 2001.01568.

[22] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, "Cross-stitch networks for multi-task learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3994–4003, 2016.

[23] X. L. F. Bossen, J. Boyce and K. S. V. Seregin, "Document jvet-m1010: Common test conditions and software reference configurations for sdr video," 9-18 January 2019.

[24] ITU-R, "Recommendation BT.2020-2: Parameters Values of Ultra-High Definition Television Systems for Production and International Programme Exchange," October 2015.

[25] ITU-R, "Recommendation BT.709-6: Parameters Values for the HDTV Standards for Production and International Programme Exchange," June 2015.

[26] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on circuits and systems for video technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[27] W. Hamidouche, T. Biatek, M. Abdoli, E. François, F. Pescador, M. Radosavljević, D. Menard, and M. Raulet, "Versatile video coding standard: A review from coding tools to consumers deployment," 2021.

[28] E. J. Giorgianni and T. E. Madden, *Digital color management: encoding solutions*, vol. 576. Addison-Wesley Reading, MA, 1998.

[29] F. Banterle, A. Artusi, K. Debattista, and A. Chalmers, "Advanced high dynamic range imaging: Theory and practice,(2011)."

[30] DVB, "DVB Fact Sheet: Introduction to the DVB Project," May 2014.

[31] ESTI, "TS-101-154 Digital Video Broadcasting (DVB); Specification for the use of Video and Audio Coding in Broadcasting Applications based on MPEG-2 Transport Stream.."

[32] M. Sugawara, M. Emoto, K. Masaoka, Y. Nishida, and Y. Shishikui, "Super hi-vision for the next generation television determination of system parameters," *ITE Transactions on Media Technology and Applications*, vol. 1, no. 1, pp. 27–33, 2013.

[33] Y. Shishikui, "Quality-of-experience evaluation of 8k ultra-high-definition television," in *2021 IEEE International Conference on Image Processing (ICIP)*, pp. 1404–1408, IEEE, 2021.

[34] Y. Sugito, S. Iwasaki, K. Chida, K. Iguchi, K. Kanda, X. Lei, H. Miyoshi, and K. Kazui, "Video bit-rate requirements for 8k 120-hz hevc/h. 265 temporal scalable coding: experimental study based on 8k subjective evaluations," *APSIPA Transactions on Signal and Information Processing*, vol. 9, 2020.

[35] A. Ichigaya and Y. Nishida, "Required bit rates analysis for a new broadcasting service using hevc/h. 265," *IEEE Transactions on Broadcasting*, vol. 62, no. 2, pp. 417–425, 2016.

[36] S. Iwasaki, X. Lei, K. Chida, Y. Sugito, K. Iguchi, K. Kanda, H. Miyoshi, and Y. Uehara, "The required video bitrate for 8k120-hz real-time temporal scalable coding," in *2020 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–5, IEEE, 2020.

[37] "8K Live Encoding at IBC 2019," 2019. https://www.insightmedia.info/8k-live-encoding-at-ibc-2019/.

[38] A. Wieckowski, J. Brandenburg, T. Hinz, C. Bartnik, V. George, G. Hege, C. Helmrich, A. Henkel, C. Lehmann, C. Stoffers, *et al.*, "Vvenc: An open and optimized vvc encoder implementation," in *2021 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pp. 1–2, IEEE, 2021.

[39] "8K VVC Encode-Decode Demo," 2019. https://8kassociation.com/industry-info/8k-news/8k-vvc-encode-decode-demo/.

[40] Y. Sugito, K. Iguchi, A. Ichigaya, K. Chida, S. Sakaida, H. Sakate, Y. Matsuda, Y. Kawahata, and N. Motoyama, "Hevc/h. 265 codec system and transmission experiments aimed at 8k broadcasting," 2015.

[41] Y. Sugito, S. Iwasaki, K. Chida, K. Iguchi, K. Kanda, X. Lei, H. Miyoshi, and K. Kazui, "A study on the required video bit-rate for 8k 120-hz hevc/h. 265 temporal scalable coding," in *2018 Picture Coding Symposium (PCS)*, pp. 106–110, IEEE, 2018.

[42] ESTI, "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications; Part 2: DVB-S2 Extensions (DVB-S2X)."

[43] ITU-T and I. JTC, "Recommendation H.261: Video Codec for Audiovisual Services at px 64 kbits/s," 1993.

[44] G. Bjøntegaard, "Document VCEG-M33 ITU-T Q6/16: Calculation of Average PSNR Differences Between RD- Curves," April 2001.

[45] R. Sotelo, J. Joskowicz, M. Anedda, M. Murroni, and D. D. Giusto, "Subjective video quality assessments for 4k uhdtv," in *2017 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, pp. 1–6, IEEE, 2017.

[46] Z. Wang, L. Lu, and A. C. Bovik, "Video quality assessment based on structural distortion measurement," *Signal processing: Image communication*, vol. 19, no. 2, pp. 121–132, 2004.

[47] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, vol. 2, pp. 1398–1402, Ieee, 2003.

[48] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Transactions on image processing*, vol. 15, no. 2, pp. 430–444, 2006.

[49] S. Li, F. Zhang, L. Ma, and K. N. Ngan, "Image quality assessment by separately evaluating detail losses and additive impairments," *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 935–949, 2011.

[50] "International Standard 11172-2: Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5mbit/s – Part 2: Video,"

[51] ITU-T, "Recommendation H.262: Transmission of Non-Telephone Signals,"

[52] ISO/IEC, "International Standard 13818-2: Generic Coding of Moving Pictures and Associated Audio Information – Part2: Video,"

[53] ITU-T, "Recommendation H.263: Video Coding for Low Bit Rate Communication,"

[54] ISO/IEC, "International Standard 14496-2: Coding of Audio-Visual Objects – Part 2: Visual.,"

[55] ISO/IEC, "International Standard 14496-10: Advanced Video Coding.,"

[56] ITU-T, "Recommendation H.264: Advanced Video Coding.,"

[57] ISO/IEC, "International Standard 23008-2: High Efficiency Video Coding,"

[58] T. K. Tan, R. Weerakkody, M. Mrak, N. Ramzan, V. Baroncini, J.-R. Ohm, and G. J. Sullivan, "Video quality evaluation methodology and verification testing of hevc compression performance," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 76–90, 2015.

[59] N. Sidaty, W. Hamidouche, O. Déforges, P. Philippe, and J. Fournier, "Compression performance of the versatile video coding: Hd and uhd visual quality monitoring," in *2019 Picture Coding Symposium (PCS)*, pp. 1–5, IEEE, 2019.

[60] Google, "VP8 Data Format and Decoding Guide," in *Tech report*, 2011.

[61] Google, "VP9 Video Codec," in *Tech report*, 2012.

[62] Y. Chen, D. Mukherjee, J. Han, A. Grange, Y. Xu, S. Parker, C. Chen, H. Su, U. Joshi, C.-H. Chiang, *et al.*, "An overview of coding tools in av1: the first video codec from the alliance for open media," *APSIPA Transactions on Signal and Information Processing*, vol. 9, 2020.

[63] F. Zhang, A. V. Katsenou, M. Afonso, G. Dimitrov, and D. R. Bull, "Comparing vvc, hevc and av1 using objective and subjective assessments," *arXiv preprint arXiv:2003.10282*, 2020.

[64] J. Ozer, "Sisvel Launches Patent Pools for VP9 and AV1," in *Streaming Media*, 2021. https://www.streamingmedia.com/Articles/ReadArticle.aspx?ArticleID=130840.

[65] K. Choi, J. Chen, D. Rusanovskyy, K.-P. Choi, and E. S. Jang, "An overview of the mpeg-5 essential video coding standard [standards in a nutshell]," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 160–167, 2020.

[66] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, "Intra coding of the hevc standard," *IEEE transactions on circuits and systems for video technology*, vol. 22, no. 12, pp. 1792–1801, 2012.

[67] P. Helle, S. Oudin, B. Bross, D. Marpe, M. O. Bici, K. Ugur, J. Jung, G. Clare, and T. Wiegand, "Block merging for quadtree-based partitioning in hevc," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1720–1731, 2012.

[68] J. Sole, R. Joshi, N. Nguyen, T. Ji, M. Karczewicz, G. Clare, F. Henry, and A. Duenas, "Transform coefficient coding in hevc," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1765–1777, 2012.

[69] A. Norkin, G. Bjontegaard, A. Fuldseth, M. Narroschke, M. Ikeda, K. Andersson, M. Zhou, and G. Van der Auwera, "Hevc deblocking filter," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1746–1754, 2012.

[70] C.-M. Fu, E. Alshina, A. 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," *IEEE Transactions on Circuits and Systems for Video technology*, vol. 22, no. 12, pp. 1755–1764, 2012.

[71] V. Sze and M. Budagavi, "High throughput cabac entropy coding in hevc," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1778–1791, 2012.

[72] F. Bossen, "Common test conditions and software reference configurations," May 2012.

[73] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE signal processing magazine*, vol. 15, no. 6, pp. 74–90, 1998.

[74] C.-Y. Tsai, C.-Y. Chen, T. Yamakage, I. S. Chong, Y.-W. Huang, C.-M. Fu, T. Itoh, T. Watanabe, T. Chujoh, M. Karczewicz, and S.-M. Lei, "Adaptive Loop Filtering for Video Coding," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, pp. 934–945, Dec. 2013.

[75] A. Mercat, A. Mäkinen, J. Sainio, A. Lemmetti, M. Viitanen, and J. Vanne, "Comparative rate-distortion-complexity analysis of vvc and hevc video codecs," *IEEE Access*, vol. 9, pp. 67813–67828, 2021.

[76] L. Ciccarelli, F. Maurer, G. Meardi, and S. F. S. Battista, "Document JVET-N0058: White paper on low complexity enhancement video coding (lcevc)," January 2021.

[77] V. Baroncini, S. Battista, L. Ciccarelli, S. Ferrara, T. Guionnet, J. L. Tanou, G. Meardi, Y. Pan, and L. Yu, "Document jvet-n19571: Summary of lcevc tests conducted to date," July 2020.

[78] SBTVD, "TV 3.0 Project," 2021. `https://forumsbtvd.org.br/tv3_0/`.

[79] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[80] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[81] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[82] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018.

[83] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, pp. 1153–1160, Dec. 1981.

[84] Hong Chang, Dit-Yan Yeung, and Yimin Xiong, "Super-resolution through neighbor embedding," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 1, (Washington, DC, USA), pp. 275–282, IEEE, 2004.

[85] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," 2012.

[86] Jianchao Yang, J. Wright, T. S. Huang, and Yi Ma, "Image Super-Resolution Via Sparse Representation," *IEEE Transactions on Image Processing*, vol. 19, pp. 2861–2873, Nov. 2010.

[87] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a Deep Convolutional Network for Image Super-Resolution," in *Computer Vision – ECCV 2014* (D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), vol. 8692, pp. 184–199, Cham: Springer International Publishing, 2014. Series Title: Lecture Notes in Computer Science.

[88] W. Shi, J. Caballero, F. Huszar, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Las Vegas, NV, USA), pp. 1874–1883, IEEE, June 2016.

[89] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," *Distill*, vol. 1, no. 10, p. e3, 2016.

[90] J. Kim, J. K. Lee, and K. M. Lee, "Accurate Image Super-Resolution Using Very Deep Convolutional Networks," *arXiv:1511.04587 [cs]*, Nov. 2016. arXiv: 1511.04587.

[91] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Honolulu, HI), pp. 105–114, IEEE, July 2017.

[92] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual Dense Network for Image Super-Resolution," *arXiv:1802.08797 [cs]*, Mar. 2018. arXiv: 1802.08797.

[93] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep Back-Projection Networks for Super-Resolution," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (Salt Lake City, UT, USA), pp. 1664–1673, IEEE, June 2018.

[94] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, and W. Wu, "Feedback Network for Image Super-Resolution," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (Long Beach, CA, USA), pp. 3862–3871, IEEE, June 2019.

[95] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image Super-Resolution Using Very Deep Residual Channel Attention Networks," in *Computer Vision – ECCV 2018* (V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, eds.), vol. 11211, pp. 294–310, Cham: Springer International Publishing, 2018. Series Title: Lecture Notes in Computer Science.

[96] P. Li, J. Xie, Q. Wang, and W. Zuo, "Is Second-order Information Helpful for Large-scale Visual Recognition?," *arXiv:1703.08050 [cs]*, Apr. 2018. arXiv: 1703.08050.

[97] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Transactions on computational imaging*, vol. 3, no. 1, pp. 47–57, 2016.

[98] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*, pp. 694–711, Springer, 2016.

[99] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[100] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy, "ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks," in *Computer Vision – ECCV 2018 Workshops* (L. Leal-Taixé and S. Roth, eds.), vol. 11133, pp. 63–79, Cham: Springer International Publishing, 2019. Series Title: Lecture Notes in Computer Science.

[101] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video Super-Resolution With Convolutional Neural Networks," *IEEE Transactions on Computational Imaging*, vol. 2, pp. 109–122, June 2016.

[102] M. Drulea and S. Nedevschi, "Total variation regularization of local-global optical flow," in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 318–323, IEEE, 2011.

[103] J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi, "Real-Time Video Super-Resolution with Spatio-Temporal Networks and Motion Compensation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Honolulu, HI), pp. 2848–2857, IEEE, July 2017.

[104] M. S. M. Sajjadi, R. Vemulapalli, and M. Brown, "Frame-Recurrent Video Super-Resolution," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (Salt Lake City, UT), pp. 6626–6634, IEEE, June 2018.

[105] L. Wang, Y. Guo, Z. Lin, X. Deng, and W. An, "Learning for Video Super-Resolution through HR Optical Flow Estimation," *arXiv:1809.08573 [cs]*, Oct. 2018. arXiv: 1809.08573.

[106] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-Revealing Deep Video Super-Resolution," p. 9.

[107] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman, "Video Enhancement with Task-Oriented Flow," *International Journal of Computer Vision*, vol. 127, pp. 1106–1125, Aug. 2019. arXiv: 1711.09078.

[108] M. Haris, G. Shakhnarovich, and N. Ukita, "Recurrent Back-Projection Network for Video Super-Resolution," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (Long Beach, CA, USA), pp. 3892–3901, IEEE, June 2019.

[109] C.-M. Fu, E. Alshina, A. 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," *IEEE*

Transactions on Circuits and Systems for Video Technology, vol. 22, pp. 1755–1764, Dec. 2012.

[110] A. Norkin, G. Bjontegaard, A. Fuldseth, M. Narroschke, M. Ikeda, K. Andersson, M. Zhou, and G. Van der Auwera, "HEVC Deblocking Filter," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, pp. 1746–1754, Dec. 2012.

[111] W. Hamidouche, T. Biatek, M. Abdoli, E. Francois, F. Pescador, M. Radosavljevic, D. Menard, and M. Raulet, "Versatile video coding standard: A review from coding tools to consumers deployment," *IEEE Consumer Electronics Magazine*, 2022.

[112] Z. Pan, X. Yi, Y. Zhang, B. Jeon, and S. Kwong, "Efficient In-Loop Filtering Based on Enhanced Deep Convolutional Neural Networks for HEVC," *IEEE Transactions on Image Processing*, vol. 29, pp. 5352–5366, 2020.

[113] D. Ding, L. Kong, G. Chen, Z. Liu, and Y. Fang, "A Switchable Deep Learning Approach for In-loop Filtering in Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2020.

[114] C. Jia, S. Wang, X. Zhang, S. Wang, J. Liu, S. Pu, and S. Ma, "Content-Aware Convolutional Neural Network for In-Loop Filtering in High Efficiency Video Coding," *IEEE Transactions on Image Processing*, vol. 28, pp. 3343–3356, July 2019.

[115] T. Li, M. Xu, C. Zhu, R. Yang, Z. Wang, and Z. Guan, "A Deep Learning Approach for Multi-Frame In-Loop Filter of HEVC," *IEEE Transactions on Image Processing*, vol. 28, pp. 5663–5678, Nov. 2019.

[116] F. Li, W. Tan, and B. Yan, "Deep Residual Network for Enhancing Quality of the Decoded Intra Frames of Hevc," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, (Athens), pp. 3918–3922, IEEE, Oct. 2018.

[117] Q. Xing, M. Xu, T. Li, and Z. Guan, "Early Exit or Not: Resource-Efficient Blind Quality Enhancement for Compressed Images," *arXiv:2006.16581 [cs, eess]*, vol. 12361, pp. 275–292, 2020. arXiv: 2006.16581.

[118] F. Zhang, C. Feng, and D. R. Bull, "Enhancing VVC Through Cnn-Based Post-Processing," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, (London, United Kingdom), pp. 1–6, IEEE, July 2020.

[119] R. Yang, M. Xu, and Z. Wang, "Decoder-side HEVC quality enhancement with scalable convolutional neural network," in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, (Hong Kong, Hong Kong), pp. 817–822, IEEE, July 2017.

[120] R. Yang, M. Xu, T. Liu, Z. Wang, and Z. Guan, "Enhancing Quality for HEVC Compressed Videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, pp. 2039–2054, July 2019. arXiv: 1709.06734.

[121] L. Yu, L. Shen, H. Yang, L. Wang, and P. An, "Quality Enhancement Network via Multi-Reconstruction Recursive Residual Learning for Video Coding," *IEEE Signal Processing Letters*, vol. 26, pp. 557–561, Apr. 2019.

[122] Y.-H. Lam, A. Zare, F. Cricri, J. Lainema, and M. M. Hannuksela, "Efficient Adaptation of Neural Network Filter for Video Compression," in *Proceedings of the 28th ACM International Conference on Multimedia*, (Seattle WA USA), pp. 358–366, ACM, Oct. 2020.

[123] L. Ma, Y. Tian, and T. Huang, "Residual-Based Video Restoration for HEVC Intra Coding," p. 7.

[124] X. He, Q. Hu, X. Han, X. Zhang, C. Zhang, and W. Lin, "Enhancing HEVC Compressed Videos with a Partition-masked Convolutional Neural Network," *2018 25th IEEE International Conference on Image Processing (ICIP)*, pp. 216–220, Oct. 2018. arXiv: 1805.03894.

[125] H. Huang, I. Schiopu, and A. Munteanu, "Frame-Wise CNN-Based Filtering for Intra-Frame Quality Enhancement of HEVC Videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, pp. 2100–2113, June 2021.

[126] F. Nasiri, W. Hamidouche, L. Morin, N. Dhollande, and G. Cocherel, "A CNN-Based Prediction-Aware Quality Enhancement Framework for VVC," *IEEE Open Journal of Signal Processing*, vol. 2, pp. 466–483, 2021.

[127] J. Tong, X. Wu, D. Ding, Z. Zhu, and Z. Liu, "Learning-Based Multi-Frame Video Quality Enhancement," in *2019 IEEE International Conference on Image Processing (ICIP)*, (Taipei, Taiwan), pp. 929–933, IEEE, Sept. 2019.

[128] X. Meng, X. Deng, S. Zhu, and B. Zeng, "Enhancing Quality for VVC Compressed Videos by Jointly Exploiting Spatial Details and Temporal Structure," *arXiv:1901.09575 [cs]*, May 2019. arXiv: 1901.09575.

[129] G. Lu, X. Zhang, W. Ouyang, D. Xu, L. Chen, and Z. Gao, "Deep Non-Local Kalman Network for Video Compression Artifact Reduction," *IEEE Transactions on Image Processing*, vol. 29, pp. 1725–1737, 2020.

[130] X. Meng, X. Deng, S. Zhu, S. Liu, C. Wang, C. Chen, and B. Zeng, "MGANet: A Robust Model for Quality Enhancement of Compressed Video," *arXiv:1811.09150 [cs]*, Jan. 2019. arXiv: 1811.09150.

[131] R. Yang, M. Xu, Z. Wang, and T. Li, "Multi-frame Quality Enhancement for Compressed Video," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (Salt Lake City, UT), pp. 6664–6673, IEEE, June 2018.

[132] Q. Xing, Z. Guan, M. Xu, R. Yang, T. Liu, and Z. Wang, "MFQE 2.0: A New Approach for Multi-frame Quality Enhancement on Compressed Video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 949–963, Mar. 2021. arXiv: 1902.09707.

[133] Viet-Anh Nguyen, Yap-Peng Tan, and Weisi Lin, "Adaptive downsampling/upsampling for better video compression at low bit rate," in *2008 IEEE International Symposium on Circuits and Systems*, (Seattle, WA, USA), pp. 1624–1627, IEEE, May 2008.

[134] S. Uslubas, E. Maani, and A. K. Katsaggelos, "A Resolution Adaptive Video Compression System," in *Intelligent Multimedia Communication: Techniques and Applications* (J. Kacprzyk, C. W. Chen, Z. Li, and S. Lian, eds.), vol. 280, pp. 167–194, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. Series Title: Studies in Computational Intelligence.

[135] K. Liu, D. Liu, H. Li, and F. Wu, "Convolutional Neural Network-Based Residue Super-Resolution for Video Coding," in *2018 IEEE Visual Communications and Image Processing (VCIP)*, (Taichung, Taiwan), pp. 1–4, IEEE, Dec. 2018.

[136] Y. Li, D. Liu, H. Li, L. Li, F. Wu, H. Zhang, and H. Yang, "Convolutional Neural Network-Based Block Up-sampling for Intra Frame Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, pp. 2316–2330, Sept. 2018. arXiv: 1702.06728.

[137] J. Lin, D. Liu, H. Yang, H. Li, and F. Wu, "Convolutional Neural Network-Based Block Up-Sampling for HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, pp. 3701–3715, Dec. 2019.

[138] T. Fu, K. Zhang, L. Zhang, S. Wang, and S. Ma, "An Enhanced Reference Structure For Reference Picture Resampling (RPR) In VVC," in *2021 IEEE International Conference on Image Processing (ICIP)*, (Anchorage, AK, USA), pp. 2069–2073, IEEE, Sept. 2021.

[139] A. Bruckstein, M. Elad, and R. Kimmel, "Down-scaling for better transform compression," *IEEE Transactions on Image Processing*, vol. 12, pp. 1132–1144, Sept. 2003.

[140] X. Zhang and X. Wu, "Can Lower Resolution Be Better?," in *Data Compression Conference (dcc 2008)*, (Snowbird, UT, USA), pp. 302–311, IEEE, Mar. 2008. ISSN: 1068-0314.

[141] A. Aaron, Z. Li, M. Manohara, J. De Cock, and D. Ronca, "Per-title encode optimization," *The Netflix Techblog*, 2015.

[142] J. Dong and Y. Ye, "ADAPTIVE DOWNSAMPLING FOR HIGH-DEFINITION VIDEO CODING," p. 4.

[143] Ren-Jie Wang, Chih-Wei Huang, and Pao-Chi Chang, "Adaptive Downsampling Video Coding With Spatially Scalable Rate-Distortion Modeling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, pp. 1957–1968, Nov. 2014.

[144] M. Shen, P. Xue, and C. Wang, "Down-sampling based video coding with super-resolution technique," in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, (Paris, France), pp. 673–676, IEEE, May 2010.

[145] M. Afonso, F. Zhang, A. Katsenou, D. Agrafiotis, and D. Bull, "Low complexity video coding based on spatial resolution adaptation," in *2017 IEEE International Conference on Image Processing (ICIP)*, (Beijing), pp. 3011–3015, IEEE, Sept. 2017.

[146] M. Afonso, F. Zhang, and D. R. Bull, "Video Compression Based on Spatio-Temporal Resolution Adaptation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, pp. 275–280, Jan. 2019.

[147] D. Ma, M. Afonso, F. Zhang, and D. R. Bull, "Perceptually-inspired super-resolution of compressed videos," *Applications of Digital Image Processing XLII*, p. 43, Sept. 2019. arXiv: 2106.08147.

[148] D. Ma, F. Zhang, and D. R. Bull, "CVEGAN: A Perceptually-inspired GAN for Compressed Video Enhancement," *arXiv:2011.09190 [cs, eess]*, Nov. 2020. arXiv: 2011.09190.

[149] D. Ma, F. Zhang, and D. R. Bull, "Video compression with low complexity CNN-based spatial resolution adaptation," *Applications of Digital Image Processing XLIII*, p. 9, Aug. 2020. arXiv: 2007.14726.

[150] Y. Li, D. Liu, H. Li, L. Li, Z. Li, and F. Wu, "Learning a Convolutional Neural Network for Image Compact-Resolution," *IEEE Transactions on Image Processing*, vol. 28, pp. 1092–1107, Mar. 2019.

[151] W. Sun and Z. Chen, "Learned Image Downscaling for Upscaling using Content Adaptive Resampler," *IEEE Transactions on Image Processing*, vol. 29, pp. 4027–4040, 2020. arXiv: 1907.12904.

[152] M. Akbari, J. Liang, and J. Han, "DSSLIC: Deep Semantic Segmentation-based Layered Image Compression," *arXiv:1806.03348 [cs]*, Apr. 2019. arXiv: 1806.03348.

[153] W. Gao, L. Tao, L. Zhou, D. Yang, X. Zhang, and Z. Guo, "Low-rate Image Compression with Super-resolution Learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, (Seattle, WA, USA), pp. 607–610, IEEE, June 2020.

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

[155] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.

[156] H. Everett III, "Generalized lagrange multiplier method for solving problems of optimum allocation of resources," *Operations research*, vol. 11, no. 3, pp. 399–417, 1963.

[157] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," *arXiv:1802.01436 [cs, eess, math]*, May 2018. arXiv: 1802.01436.

[158] J. Lee, S. Cho, and S.-K. Beack, "Context-adaptive Entropy Model for End-to-end Optimized Image Compression," *arXiv:1809.10452 [eess]*, May 2019. arXiv: 1809.10452.

[159] Y. Hu, W. Yang, Z. Ma, and J. Liu, "Learning End-to-End Lossy Image Compression: A Benchmark," *arXiv:2002.03711 [cs, eess]*, Mar. 2021. arXiv: 2002.03711.

[160] A. Van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, *et al.*, "Conditional image generation with pixelcnn decoders," *Advances in neural information processing systems*, vol. 29, 2016.

[161] T. Chen, H. Liu, Q. Shen, T. Yue, X. Cao, and Z. Ma, "DeepCoder: A deep neural network based video compression," in *2017 IEEE Visual Communications and Image Processing (VCIP)*, (St. Petersburg, FL), pp. 1–4, IEEE, Dec. 2017.

[162] G. Lu, W. Ouyang, D. Xu, X. Zhang, C. Cai, and Z. Gao, "DVC: An End-to-end Deep Video Compression Framework," *arXiv:1812.00101 [cs, eess]*, Apr. 2019. arXiv: 1812.00101.

[163] Z. Hu, Z. Chen, D. Xu, G. Lu, W. Ouyang, and S. Gu, "Improving Deep Video Compression by Resolution-Adaptive Flow Coding," in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), vol. 12347, pp. 193–209, Cham: Springer International Publishing, 2020. Series Title: Lecture Notes in Computer Science.

[164] A. Djelouah, J. Campos, S. Schaub-Meyer, and C. Schroers, "Neural Inter-Frame Compression for Video Coding," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, (Seoul, Korea (South)), pp. 6420–6428, IEEE, Oct. 2019.

[165] H. Liu, H. Shen, L. Huang, M. Lu, T. Chen, and Z. Ma, "Learned Video Compression via Joint Spatial-Temporal Correlation Exploration," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 11580–11587, Apr. 2020.

[166] Z. Hu, G. Lu, and D. Xu, "FVC: A New Framework towards Deep Video Compression in Feature Space," *arXiv:2105.09600 [cs, eess]*, Aug. 2021. arXiv: 2105.09600.

[167] E. Agustsson, D. Minnen, N. Johnston, J. Balle, S. J. Hwang, and G. Toderici, "Scale-Space Flow for End-to-End Optimized Video Compression," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (Seattle, WA, USA), pp. 8500–8509, IEEE, June 2020.

[168] R. Yang, F. Mentzer, L. Van Gool, and R. Timofte, "Learning for Video Compression With Hierarchical Quality and Recurrent Enhancement," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (Seattle, WA, USA), pp. 6627–6636, IEEE, June 2020.

[169] T. Ladune, P. Philippe, W. Hamidouche, L. Zhang, and O. Déforges, "Conditional Coding for Flexible Learned Video Compression," *arXiv:2104.07930 [eess]*, Apr. 2021. arXiv: 2104.07930.

[170] G. Toderici, S. M. O'Malley, S. J. Hwang, D. Vincent, D. Minnen, S. Baluja, M. Covell, and R. Sukthankar, "Variable Rate Image Compression with Recurrent Neural Networks," *arXiv:1511.06085 [cs]*, Mar. 2016. arXiv: 1511.06085.

[171] G. Toderici, D. Vincent, N. Johnston, S. J. Hwang, D. Minnen, J. Shor, and M. Covell, "Full Resolution Image Compression with Recurrent Neural Networks," *arXiv:1608.05148 [cs]*, July 2017. arXiv: 1608.05148.

[172] C. Jia, Z. Liu, Y. Wang, S. Ma, and W. Gao, "Layered image compression using scalable auto-encoder," in *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 431–436, IEEE, 2019.

[173] Y.-H. Tsai, M.-Y. Liu, D. Sun, M.-H. Yang, and J. Kautz, "Learning Binary Residual Representations for Domain-specific Video Streaming," *arXiv:1712.05087 [cs]*, Dec. 2017. arXiv: 1712.05087.

[174] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1101, 1952.

[175] B. Bross, Y.-K. Wang, Y. Ye, S. Liu, J. Chen, G. J. Sullivan, and J.-R. Ohm, "Overview of the versatile video coding (vvc) standard and its applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 3736–3764, 2021.

[176] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1874–1883, 2016.

[177] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE transactions on acoustics, speech, and signal processing*, vol. 29, no. 6, pp. 1153–1160, 1981.

[178] B. Bross, H. Kirchhoffer, C. Bartnik, and M. Palkow, "Document JVET-Q0791: Multiformat berlin test sequences.," 13-17 January 2020.

[179] F. Bossen, J. Boyce, K. Suehring, X. Li, and V. Seregin, "Jvet common test conditions and software reference configurations for sdr video," March 2019.

[180] ITU-R, "Recommendation BT.500-13: Methodologies for the Subjective Assessment of the Quality of Television Images," October 2019.

[181] ITU-R, "Recommendation BT.913: Methods for the Subjective Assessment of Video for Quality, Audio and Audiovisual Quality of Internet Video and Distribution Quality Television in any Environment."

[182] W. Husak, "ISO/IEC JTC1/SC29/WG11-M37372: ATSC liaison on SHVC."

[183] D. Ma, F. Zhang, and D. Bull, "Bvi-dvc: a training database for deep video compression," *IEEE Transactions on Multimedia*, 2021.

[184] X. Li, J. Boyce, P. Onno, and Y. Ye, "Common SHM test conditions and software reference configurations," April 2013.

[185] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4681–4690, 2017.

[186] D. Ma, M. Afonso, F. Zhang, and D. R. Bull, "Perceptually-inspired super-resolution of compressed videos," in *Applications of Digital Image Processing XLII*, vol. 11137, p. 1113717, International Society for Optics and Photonics, 2019.

[187] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7482–7491, 2018.

[188] S. Liu, E. Johns, and A. J. Davison, "End-to-end multi-task learning with attention," *arXiv:1803.10704*, 2018.

[189] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[190] M.-Z. Wang, S. Wan, H. Gong, and M.-Y. Ma, "Attention-based dual-scale cnn in-loop filter for versatile video coding," *IEEE Access*, vol. 7, pp. 145214–145226, 2019.

[191] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for neural networks for image processing," *arXiv:1511.08861*, 2015.

[192] E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 126–135, 2017.

[193] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[194] C. Bonnineau, W. Hamidouche, J.-F. Travers, and O. Deforges, "Versatile video coding and super-resolution for efficient delivery of 8k video with 4k backward-compatibility," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2048–2052, IEEE, 2020.

[195] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[196] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional gans," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8798–8807, 2018.

[197] P. Voigtlaender, Y. Chai, F. Schroff, H. Adam, B. Leibe, and L.-C. Chen, "Feelvos: Fast end-to-end embedding learning for video object segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9481–9490, 2019.

[198] T. Ebrahimi and C. Horne, "Mpeg-4 natural video coding–an overview," *Signal Processing: Image Communication*, vol. 15, no. 4-5, pp. 365–385, 2000.

[199] A. Vetro and H. Sun, "An overview of mpeg-4 object-based encoding algorithms," in *Proceedings International Conference on Information Technology: Coding and Computing*, pp. 366–369, IEEE, 2001.

[200] A. Puri and A. Eleftheriadis, "Mpeg-4: An object-based multimedia coding standard supporting mobile applications," *Mobile Networks and Applications*, vol. 3, no. 1, pp. 5–32, 1998.

[201] P. Nunes and F. Pereira, "Object-based rate control for the mpeg-4 visual simple profile," in *Proc Workshop on Image Analysis for Multimedia Interactive Services WIAMIS*, 1999.

[202] A. Cavallaro, O. Steiger, and T. Ebrahimi, "Semantic video analysis for adaptive content delivery and automatic description," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 10, pp. 1200–1209, 2005.

[203] J. Liu, S. Wang, and R. Urtasun, "Dsic: Deep stereo image compression," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3136–3145, 2019.

[204] M. Akbari, J. Liang, and J. Han, "Dsslic: deep semantic segmentation-based layered image compression," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2042–2046, IEEE, 2019.

[205] T. M. Hoang, J. Zhou, and Y. Fan, "Image compression with encoder-decoder matched semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 160–161, 2020.

[206] D. Liu, B. Wen, J. Jiao, X. Liu, Z. Wang, and T. S. Huang, "Connecting image denoising and high-level vision tasks via deep learning," *IEEE Transactions on Image Processing*, vol. 29, pp. 3695–3706, 2020.

[207] M. S. Rad, B. Bozorgtabar, C. Musat, U.-V. Marti, M. Basler, H. K. Ekenel, and J.-P. Thiran, "Benefiting from multitask learning to improve single image super-resolution," *arXiv preprint arXiv:1907.12488*, 2019.

[208] L. Kang, P. Ye, Y. Li, and D. Doermann, "Convolutional neural networks for no-reference image quality assessment," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1733–1740, 2014.

[209] H. Otroshi-Shahreza, A. Amini, and H. Behroozi, "No-reference image quality assessment using transfer learning," in *2018 9th International Symposium on Telecommunications (IST)*, pp. 637–640, IEEE, 2018.

[210] Y. Li, L.-M. Po, L. Feng, and F. Yuan, "No-reference image quality assessment with deep convolutional neural networks," in *2016 IEEE International Conference on Digital Signal Processing (DSP)*, pp. 685–689, IEEE, 2016.

[211] D. Pan, P. Shi, M. Hou, Z. Ying, S. Fu, and Y. Zhang, "Blind predicting similar quality map for image quality assessment," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6373–6382, 2018.

[212] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[213] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Transactions on Computational Imaging*, vol. 2, no. 2, pp. 109–122, 2016.

[214] J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi, "Real-time video super-resolution with spatio-temporal networks and motion compensation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4778–4787, 2017.

[215] F. Nasiri, W. Hamidouche, L. Morin, N. Dhollande, and G. Cocherel, "A cnn-based prediction-aware quality enhancement framework for vvc," *arXiv preprint arXiv:2105.05658*, 2021.

[216] F. Maurer, S. Battista, L. Ciccarelli, G. Meardi, and S. Ferrara, "Overview of mpeg-5 part 2–low complexity enhancement video coding (lcevc)," *ITU Journal: ICT Discoveries*, vol. 3, no. 1, 2020.

[217] T. Ladune, P. Philippe, W. Hamidouche, L. Zhang, and O. Déforges, "Conditional coding for flexible learned video compression," *arXiv preprint arXiv:2104.07930*, 2021.

[218] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7939–7948, 2020.

[219] D. Minnen, J. Ballé, and G. Toderici, "Joint autoregressive and hierarchical priors for learned image compression," *arXiv preprint arXiv:1809.02736*, 2018.

[220] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end optimized image compression," *arXiv preprint arXiv:1611.01704*, 2016.

[221] "Verification test report on the compression performance of low complexity enhancement video coding," April 2021.

[222] Z. Cui, J. Wang, B. Bai, T. Guo, and Y. Feng, "G-VAE: A Continuously Variable Rate Deep Image Compression Framework," *arXiv:2003.02012 [cs, eess]*, Apr. 2020. arXiv: 2003.02012.

[223] C. Bonnineau, W. Hamidouche, J. Fournier, N. Sidaty, J.-F. Travers, and O. Déforges, "Perceptual quality assessment of hevc and vvc standards for 8k video," *IEEE Transactions on Broadcasting*, 2022.

[224] C. Bonnineau, W. Hamidouche, J.-F. Travers, and O. Deforges, "Dvb TM-AVC1256: Subjectve Quality Assessment of VVC and HEVC for 8k Video Resolution," October 2021.

[225] C. Bonnineau, W. Hamidouche, J.-F. Travers, and O. Deforges, "Document JVET-X0186: Subjectve Quality Assessment of VVC and HEVC for 8k Video Resolution," October 2021.

[226] C. Bonnineau, J.-Y. Aubié, W. Hamidouche, O. Déforges, J. Travers, and N. Sidaty, "An objective evaluation of codecs and post- processing tools for 8k video compression," 2020.

[227] C. Bonnineau, W. Hamidouche, J.-F. Travers, N. Sidaty, and O. Deforges, "Multitask learning for vvc quality enhancement and super-resolution," *arXiv preprint arXiv:2104.08319*, 2021.

[228] C. Bonnineau, W. Hamidouche, J.-F. Travers, N. Sidaty, J.-Y. Aubié, and O. Deforges, "Caesr: Conditional autoencoder and super-resolution for learned spatial scalability," in *2021 International Conference on Visual Communications and Image Processing (VCIP)*, pp. 1–5, IEEE, 2021.

Titre : Méthodes de codage et de post-traitement par apprentissage pour la reconstruction de signaux vidéo 8K

Mot clés : Compression vidéo, 8K, super-résolution, auto-encodeurs, apprentissage profond

Résumé :

La résolution d'image 8K a récemment connu un fort engouement dans la communauté de la vidéo. Du point de vue de la télévision numérique terrestre (TNT), l'introduction de ce nouveau format multimédia n'est envisageable qu'en assurant la compatibilité avec le parc d'équipement existant. Cependant, dû aux fortes contraintes de débit imposées par ce format, une diffusion 8K rétrocompatible sur la TNT n'est pas concevable avec les méthodes de compression actuelles. Cette thèse a pour objectif de proposer des méthodes innovantes permettant la reconstruction du signal 8K du côté des futurs récepteurs. Les dernières avancées proposées dans le domaine des technologies par apprentissage ont démontré des performances prometteuses pour la compression et le post-traitement (e.g., super-résolution) de données vidéos. Dans un premier temps, une étude évaluant les performances de différentes méthodes de compression et de super-resolution appliquées à des contenus 8K a permis de définir deux axes de recherche. Dans un premier axe, deux modèles de super-résolution multitâche dédiés au suréchantillonnage de vidéos compressées ont été développés. Le deuxième axe de la thèse a été dirigé vers le développement d'un algorithme de compression par apprentissage guidant le suréchantillonnage avec un flux de métadonnées. Les résultats ont montré que l'approche proposée offre de meilleures performances que l'état de l'art pour le cas d'usage défini.

Title: Learning based coding and post-processing methods for 8K video reconstruction

Keywords: Video Compression, 8K, Super-Resolution, Autoencoders, Deep Learning

Abstract:

8K video resolution has attracted a lot of interest from the industry as a media format for immersive video applications enhancing the end users' quality of experience. However, delivering such a format on DTT, where backward compatibility is essential, is impossible due to the high bandwidth requirements of this format. This thesis aims to propose innovative compression and post-processing methods to perform the reconstruction of this format on the receiver side. The recent advances in AI-based technologies have demonstrated outstanding performances for the compression or post-processing (e.g., super-resolution) of video data. First, a study has been performed to assess the performance of compression methods and upcsaling algorithms on 8K video. Then, two multitask super-resolution models dedicated to upscaling compressed videos have been developed. Finally, a learned compression system has been designed to guide a super-resolution model with metadata containing the high frequencies lost during downscaling. The results show that the proposed solution offers better performance than the state-of-the-art regarding the use-case of this thesis.