



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

H.262

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

(07/95)

**TRANSMISSION DE SIGNAUX
NON TÉLÉPHONIQUES**

**TECHNOLOGIES DE L'INFORMATION –
CODAGE GÉNÉRIQUE DES IMAGES ANIMÉES
ET DU SON ASSOCIÉ: DONNÉES VIDÉO**

Recommandation UIT-T H.262

(Antérieurement «Recommandation du CCITT»)

AVANT-PROPOS

L'UIT (Union internationale des télécommunications) est une institution spécialisée des Nations Unies dans le domaine des télécommunications. L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT. Au sein de l'UIT-T, qui est l'entité qui établit les normes mondiales (Recommandations) sur les télécommunications, participent quelque 179 pays membres, 84 exploitations de télécommunications reconnues, 145 organisations scientifiques et industrielles et 38 organisations internationales.

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution n° 1 de la Conférence mondiale de normalisation des télécommunications (CMNT), (Helsinki, 1993). De plus, la CMNT, qui se réunit tous les quatre ans, approuve les Recommandations qui lui sont soumises et établit le programme d'études pour la période suivante.

Dans certains secteurs de la technologie de l'information qui correspondent à la sphère de compétence de l'UIT-T, les normes nécessaires se préparent en collaboration avec l'ISO et la CEI. Le texte de la Recommandation H.262 de l'UIT-T a été approuvé le 10 juillet 1995. Son texte est publié, sous forme identique, comme Norme internationale ISO/CEI 13818-2.

NOTE

Dans la présente Recommandation, l'expression «Administration» est utilisée pour désigner de façon abrégée aussi bien une administration de télécommunications qu'une exploitation reconnue.

© UIT 1996

Droits de reproduction réservés. Aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'UIT.

TABLE DES MATIÈRES

| | <i>Page</i> |
|---|-------------|
| Résumé..... | iii |
| Introduction..... | iv |
| Intro. 1 Objet..... | iv |
| Intro. 2 Application..... | iv |
| Intro. 3 Profils et niveaux..... | iv |
| Intro. 4 Syntaxe échelonnable et syntaxe non échelonnable | v |
| 1 Domaine d'application..... | 1 |
| 2 Références normatives | 1 |
| 3 Définitions..... | 2 |
| 4 Abréviations et symboles | 8 |
| 4.1 Opérateurs arithmétiques | 8 |
| 4.2 Opérateurs logiques | 9 |
| 4.3 Opérateurs relationnels | 9 |
| 4.4 Opérateurs binaires | 9 |
| 4.5 Affectation | 9 |
| 4.6 Mnémoniques..... | 10 |
| 4.7 Constantes..... | 10 |
| 5 Conventions..... | 10 |
| 5.1 Méthode de description de la syntaxe du flux binaire | 10 |
| 5.2 Définition des fonctions | 11 |
| 5.3 Valeur réservée, valeur interdite et bit marqueur | 12 |
| 5.4 Précision arithmétique | 12 |
| 6 Syntaxe et sémantique du flux binaire de données vidéo | 12 |
| 6.1 Structure des données vidéo codées..... | 12 |
| 6.2 Syntaxe du flux binaire de données vidéo codées..... | 24 |
| 6.3 Sémantique du flux binaire de données vidéo codées..... | 40 |
| 7 Processus de décodage des données vidéo | 66 |
| 7.1 Structures syntaxiques supérieures..... | 66 |
| 7.2 Décodage à longueur variable..... | 67 |
| 7.3 Balayage inverse des coefficients | 70 |
| 7.4 Quantification inverse..... | 71 |
| 7.5 Transformation DCT inverse | 76 |
| 7.6 Compensation de mouvement..... | 76 |
| 7.7 Echelonnabilité spatiale | 91 |
| 7.8 Echelonnabilité SNR..... | 104 |
| 7.9 Echelonnabilité temporelle | 109 |
| 7.10 Subdivision des données | 113 |
| 7.11 Echelonnabilité hybride | 114 |
| 7.12 Sortie du processus de décodage..... | 115 |
| 8 Profils et niveaux..... | 119 |
| 8.1 Compatibilité avec ISO/CEI 11172-2 | 120 |
| 8.2 Relation entre profils définis..... | 120 |
| 8.3 Relation entre niveaux définis..... | 122 |
| 8.4 Couches échelonnables | 123 |
| 8.5 Valeurs paramétriques pour profils, niveaux et couches définis | 126 |

| | <i>Page</i> |
|--|-------------|
| Annexe A – Transformation en cosinus discrète | 130 |
| Annexe B – Tables des codes à longueur variable | 131 |
| B.1 Adressage des macroblocs | 131 |
| B.2 Type de macrobloc..... | 132 |
| B.3 Structure des macroblocs | 137 |
| B.4 Vecteurs de mouvement..... | 138 |
| B.5 Coefficients DCT | 139 |
| Annexe C – Vérificateur de mémoire vidéo | 147 |
| Annexe D – Caractéristiques supportées par l'algorithme | 152 |
| D.1 Vue d'ensemble..... | 152 |
| D.2 Formats vidéo | 152 |
| D.3 Qualité d'image | 153 |
| D.4 Contrôle du débit binaire | 153 |
| D.5 Mode à faible délai | 154 |
| D.6 Accès aléatoire/interconnexion des canaux..... | 154 |
| D.7 Echelonnabilité | 154 |
| D.8 Compatibilité | 162 |
| D.9 Différences entre la présente Spécification et ISO/CEI 11172-2 | 163 |
| D.10 Complexité..... | 165 |
| D.11 Edition des flux binaires codés | 165 |
| D.12 Modes d'enrichissement | 166 |
| D.13 Robustesse aux erreurs..... | 167 |
| D.14 Séquences concaténées | 176 |
| Annexe E – Restrictions de profil et de niveau..... | 177 |
| E.1 Restrictions applicables aux éléments syntaxiques dans les profils | 177 |
| E.2 Combinaisons de couches autorisées | 189 |
| Annexe F – Bibliographie | 211 |

Résumé

La présente Recommandation | Norme internationale spécifie la représentation codée des données vidéo et du processus de décodage nécessaire pour reconstruire des images. Elle décrit un système de codage générique des informations vidéo, servant à une large gamme d'applications, de débits, de résolutions et de qualités d'image. Son algorithme de codage hybride est basé sur la compensation de mouvement et sur la transformée en cosinus discrète (DCT). Les images à coder peuvent être issues d'un balayage entrelacé ou progressif. Les éléments algorithmiques nécessaires sont intégrés dans une syntaxe unique et un nombre limité de sous-ensembles est défini en termes de profils (caractéristiques fonctionnelles) et de niveaux (paramètres), afin de faciliter l'application pratique de la présente norme relative au codage générique des données vidéo.

Introduction

Intro. 1 Objet

La présente Partie de cette Spécification a été élaborée en réponse au besoin croissant d'une méthode de codage générique des images animées et du son associé, pour diverses applications telles que les supports numériques d'enregistrement, la télédiffusion et la communication. L'utilisation de la présente Spécification implique que les données vidéo d'images animées peuvent être manipulées sous forme de données informatiques, stockées sur divers supports d'enregistrement, être émises et reçues au moyen des réseaux existants et futurs, puis distribuées sur les canaux de télédiffusion existants et futurs.

Intro. 2 Application

Les applications de la présente Spécification couvrent différents domaines tels que:

| | |
|------|--|
| CaTV | Télévision par câble, distribuée par câbles à fibres optiques, à conducteurs métalliques, etc. (<i>cable TV</i>) |
| CDAD | Distribution audionumérique par câble (<i>Cable Digital Audio Distribution</i>) |
| DSB | Diffusion audionumérique (de Terre et par satellite) (<i>Digital Sound Broadcasting</i>) |
| DTTB | Télédiffusion numérique par voie hertzienne de Terre (<i>Digital Terrestrial Television Broadcasting</i>) |
| EC | Cinéma électronique (<i>Electronic Cinema</i>) |
| HTT | Spectacle télédiffusé à domicile (<i>Home Television Theatre</i>) |
| IPC | Communications interpersonnelles (<i>Interpersonal Communications</i>) (visioconférence, visio- phone, etc.) |
| ISM | Supports d'enregistrements interactifs (<i>Interactive Storage Media</i>) (disques optiques, etc.) |
| JET | Journalisme Electronique Télévisuel, y compris le Reportage d'Actualités par Satellite (RAS) |
| MMM | Messagerie multimédia (<i>Multimedia Mailing</i>) |
| NCA | Nouvelles et actualités (<i>News and Current Affairs</i>) |
| NDB | Base de données sur réseau (via ATM, etc.) (<i>Networked Database</i>) |
| RVS | Télévidéosurveillance (<i>Remote Video Surveillance</i>) |
| SFS | Service Fixe par Satellite (par exemple, vers des têtes de réseau) |
| SRS | Service de Radiodiffusion par Satellite (à domicile) |
| SSM | Support d'enregistrement séquentiel (magnétoscopes numériques, etc.) (<i>Serial Storage Media</i>) |

Intro. 3 Profils et niveaux

La présente Spécification a une vocation générique, en ce sens qu'elle vise une large gamme d'applications, de débits, de résolutions, de qualités et de services. Les applications devraient couvrir entre autres les supports d'enregistrement numérique, la télédiffusion et les communications. Au cours de l'élaboration de la présente Spécification, diverses exigences ont été prises en considération sur la base d'applications typiques. Les éléments algorithmiques nécessaires ont été mis au point et ont été intégrés dans une syntaxe unique. C'est pourquoi la présente Spécification facilitera l'échange de flux binaires entre applications différentes.

Compte tenu cependant des impératifs d'application pratique de la syntaxe générale décrite dans la présente Spécification, un nombre limité de modules de cette syntaxe sont également stipulés en tant que «profils» et «niveaux». Ces termes, et leurs analogues, sont définis formellement à l'article 3.

Un «profil» est un sous-ensemble défini de la syntaxe générale du flux binaire, elle-même définie par la présente Spécification. Dans le cadre des limites imposées par la syntaxe d'un profil donné, il est toujours possible de prescrire une très grande étendue de variation de performance pour les codeurs et les décodeurs, en fonction des valeurs attribuées aux paramètres du flux binaire. Il est par exemple possible de spécifier des images de dimensions allant jusqu'à environ 2^{14} pels en largeur par 2^{14} lignes en hauteur. Pour le moment, il n'est ni pratique ni économique de mettre en œuvre un décodeur capable de traiter toutes les tailles d'image possibles.

Pour résoudre ce problème, des «niveaux» sont définis dans chaque profil. Un niveau est défini comme étant un ensemble de contraintes imposées sur des paramètres dans le flux binaire. Ces contraintes peuvent être de simples limites numériques. Elles peuvent également prendre la forme de limites imposées à des combinaisons arithmétiques des paramètres (par exemple, largeur d'image fois la hauteur d'image fois la fréquence image).

Les flux binaires conformes à la présente Spécification font appel à une syntaxe commune. Afin d'obtenir un sous-ensemble de la syntaxe complète, on inclut dans le flux binaire des fanions et des paramètres qui signalent la présence ou l'absence d'éléments syntaxiques devant apparaître ultérieurement dans le flux binaire. Pour spécifier des contraintes syntaxiques (et donc définir un profil), il suffit donc de délimiter les valeurs de ces fanions et paramètres, spécifiant la présence d'éléments syntaxiques ultérieurs.

Intro. 4 Syntaxe échelonnable et syntaxe non échelonnable

On peut subdiviser la syntaxe complète en deux catégories principales: l'une est la syntaxe non échelonnable, qui est structurée comme un surensemble de la syntaxe définie dans ISO/CEI 11172-2. La principale caractéristique de la syntaxe non échelonnable est la présence d'outils supplémentaires de compression pour les signaux vidéo entrelacés. La deuxième catégorie est la syntaxe échelonnable, dont la principale caractéristique est de permettre la reconstruction de données vidéo utiles à partir de segments d'un flux binaire total. Le processus consiste à structurer le flux total en deux couches différentes ou plus, en commençant par une couche de base autonome et en ajoutant un certain nombre de couches d'amélioration. La couche de base peut utiliser la syntaxe non échelonnable ou, en certaines occurrences, utiliser une syntaxe conforme à ISO/CEI 11172-2.

Intro. 4.1 Vue d'ensemble de la syntaxe non échelonnable

La représentation codée qui est définie dans la syntaxe non échelonnable réalise un taux de compression élevé tout en préservant une bonne qualité d'image. L'algorithme n'est pas sans perte car les valeurs exactes des pels ne sont pas conservées au cours du codage. L'obtention d'une bonne qualité iconographique aux débits binaires en cause exige un taux de compression très élevé, ce qui n'est pas réalisable avec le seul codage intra-image. La nécessité d'un accès aléatoire est toutefois mieux prise en compte avec un pur codage intra-image. Le choix entre les techniques est fondé sur le besoin de trouver un compromis entre une qualité d'image élevée, un taux de compression élevé et la nécessité de permettre un accès aléatoire au flux binaire codé.

Un certain nombre de techniques sont utilisées pour obtenir un taux de compression élevé. L'algorithme utilise d'abord une compensation de mouvement par blocs, ce qui permet de diminuer la redondance temporelle. Cette compensation de mouvement est utilisée aussi bien pour la prédiction causale de l'image actuelle à partir d'une image précédente que pour la prédiction non causale (interpolative) à partir d'images précédentes et futures. Les vecteurs de mouvement sont définis pour chaque région de 16 pels \times 16 lignes de l'image. Le signal différentiel, c'est-à-dire l'erreur de prédiction, est encore comprimé au moyen de la transformée en cosinus discrète (DCT) afin d'en éliminer les corrélations spatiales avant sa quantification au cours d'un processus irréversible qui rejette les informations moins importantes. Finalement, les vecteurs de mouvement sont combinés avec les informations résiduelles de transformation DCT puis codés au moyen de mots à longueur variable.

Intro. 4.1.1 Traitement temporel

En raison des exigences contradictoires de l'accès aléatoire et d'une compression très efficace, trois principaux types d'image seront définis. Les images à codage intratrame ou images intra (images I) sont codées sans référence à d'autres images. Elles fournissent des points d'accès à la séquence codée, à partir desquels le décodage peut commencer; mais elles ne sont codées qu'à un taux de compression modeste. Les images à codage prédictif ou images prédites (images P) sont codées plus efficacement, avec une prédiction basée sur la compensation de mouvement à partir d'une précédente image intra ou prédite. Les images P sont généralement utilisées comme références pour la prédiction des images suivantes. Les images codées par prédiction bidirectionnelle ou images bidirectionnelles (images B) offrent le taux de compression le plus élevé mais nécessitent la présence d'images de référence aussi bien antérieures que postérieures pour effectuer la compensation de mouvement. Ces images bidirectionnelles ne sont jamais utilisées comme références pour la prédiction (sauf dans le cas où l'image résultante est utilisée comme référence dans une couche d'amélioration spatialement échelonnable). L'organisation de ces trois types d'image dans une séquence est très souple. Le choix en est laissé au codeur et dépendra des exigences de l'application. La Figure Intro. 1 montre la relation entre les trois différents types d'image.

Intro. 4.1.2 Codage d'images vidéo entrelacées

Chaque image d'une vidéo entrelacée est formée de deux trames, séparées par un signal de synchronisation de trame. Selon la présente Spécification, les deux trames d'une image complète peuvent être codées soit comme une seule image soit comme deux images. Le codage image (ou «bi-trame») ou le codage trame (ou monotrane) peut être sélectionné dynamiquement, image par image. Le codage image est normalement préféré lorsque la scène vidéo contient un nombre important de détails avec peu de mouvements. Le codage trame, dans lequel la deuxième trame peut être prédite d'après la première, donne de meilleurs résultats lorsque les mouvements sont rapides.

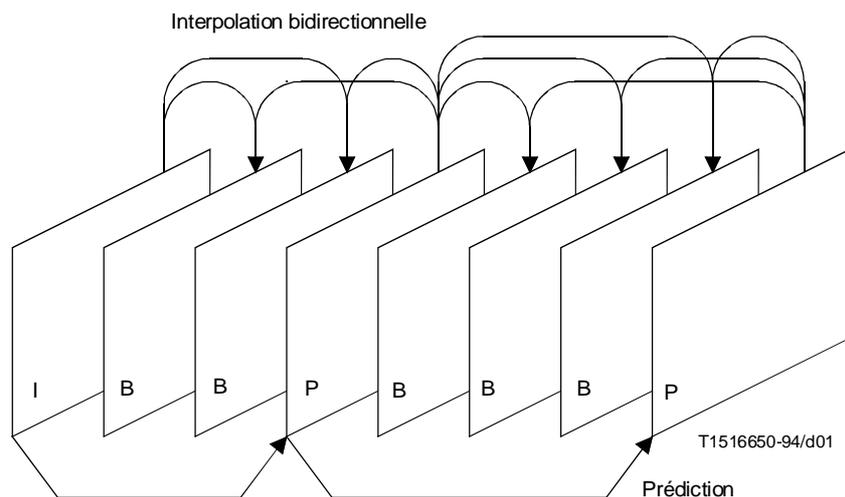


Figure Intro. 1 – Exemple de structure temporelle d'images

Intro. 4.1.3 Représentation du mouvement – Macroblocs

Comme dans ISO/CEI 11172-2, le choix de 16×16 macroblocs pour l'unité de compensation de mouvement est le résultat d'un compromis entre le gain de codage obtenu en utilisant les informations de mouvement et la servitude binaire due à leur représentation. Chaque macrobloc peut être temporellement prédit, selon une parmi plusieurs méthodes différentes. Par exemple, en codage image, la prédiction de la précédente image peut elle-même se fonder sur les deux trames de cette image ou sur une seule. Selon le type de macrobloc, l'information du vecteur de mouvement, et toutes autres informations associées, est codée avec un signal d'erreur de prédiction dans chaque macrobloc. Les vecteurs de mouvement sont codés différemment tout en respectant les derniers vecteurs de mouvement codés au moyen de mots à longueur variable. On peut programmer image par image la longueur maximale des vecteurs de mouvement, de manière que les applications les plus exigeantes puissent être mises en œuvre sans compromettre la performance du système dans des situations plus courantes.

Il appartient au codeur de calculer les vecteurs de mouvement appropriés. La présente Spécification ne spécifie pas la façon dont il convient d'effectuer ce calcul.

Intro. 4.1.4 Réduction de la redondance spatiale

Les signaux des images originales et les signaux d'erreur de prédiction possèdent, les uns comme les autres, un degré élevé de redondance spatiale. La présente Spécification fait appel à une méthode de filtrage par transformée DCT sur des blocs, avec quantification pondérée en termes de perception visuelle et codage des longueurs de séquence. Après une prédiction compensée en mouvement ou une interpolation, l'image résiduelle est subdivisée en 8×8 blocs. Ces blocs sont convertis dans le domaine DCT, où ils sont ensuite pondérés avant d'être quantifiés. A l'issue du processus de quantification, de nombreux coefficients ont une valeur nulle. On fait alors appel à un codage à longueur variable sur un tableau à 2 dimensions afin de coder efficacement tous les coefficients.

Intro. 4.1.5 Formats de chrominance

En plus du format 4:2:0 qui est retenu dans ISO/CEI 11172-2, la présente Spécification supporte les formats de chrominance 4:2:2 et 4:4:4.

Intro. 4.2 Extensions échelonnables

Les outils d'échelonnabilité figurant dans la présente Spécification sont conçus pour gérer des applications pouvant supporter plusieurs couches vidéo. Entre autres domaines d'application notables, on citera les télécommunications vidéo, les communications vidéo sur réseaux à mode de transfert asynchrone (ATM), l'interfonctionnement des standards vidéo, la hiérarchisation des services vidéo selon diverses résolutions spatiales, temporelles et qualitatives, la TVHD à TV

imbriquée, les systèmes autorisant la migration vers une TVHD à résolution temporelle plus élevée, etc. Une solution simple, en termes de vidéo échelonnable, est la technique de diffusion simultanée qui est fondée sur la transmission ou l'enregistrement, après lecture du support, de multiples sources vidéo codées indépendamment les unes des autres; mais une autre solution, plus efficace, consiste à effectuer un codage échelonnable des données vidéo, dans lequel la largeur de bande attribuée à une reproduction donnée du support vidéo peut être partiellement réutilisée pour le codage de la reproduction vidéo suivante. Dans un codage vidéo échelonnable, on part du principe que, pour chaque type de flux binaire codé, des décodeurs présentant divers degrés de complexité peuvent décoder et afficher des données vidéo codées, lues de manière appropriée. Un codeur échelonnable est susceptible d'être plus complexe qu'un codeur à une seule couche. La présente Recommandation | Norme internationale distingue toutefois plusieurs modes d'échelonnabilité, s'adressant à des applications ne se recoupant pas, de complexité correspondante. Les principaux outils d'échelonnabilité offerts sont les suivants:

- subdivision des données;
- échelonnabilité SNR;
- échelonnabilité spatiale; et
- échelonnabilité temporelle.

Par ailleurs, des combinaisons de ces outils d'échelonnabilité de base sont également possibles: de telles combinaisons seront appelées *échelonnabilité hybride*. Dans le cas de l'échelonnabilité de base, on peut avoir deux couches de flux vidéo: la *couche inférieure* et la *couche d'amélioration*; alors que, dans le cas de l'échelonnabilité hybride, on peut avoir jusqu'à trois couches de données. Les Tableaux Intro. 1 à Intro. 3 présentent quelques exemples d'application selon divers degrés d'échelonnabilité.

Tableau Intro. 1 – Applications de l'échelonnabilité SNR

| Couche inférieure | Couche d'amélioration | Application |
|-----------------------------|---|--|
| Recommandation UIT-R BT.601 | Même résolution et même format que la couche inférieure | Service à deux niveaux de qualité pour la télévision conventionnelle |
| Haute définition | Même résolution et même format que la couche inférieure | Service à deux niveaux de qualité pour la TVHD |
| Haute définition en 4:2:0 | Diffusion simultanée en format chromatique 4:2:2 | Production/distribution vidéo |

Tableau Intro. 2 – Applications de l'échelonnabilité spatiale

| Couche de base | Couche d'amélioration | Application |
|-----------------------------|-----------------------------|--|
| Balayage progressif (30 Hz) | Balayage progressif (30 Hz) | Compatibilité ou échelonnabilité CIF/SCIF |
| Balayage entrelacé (30 Hz) | Balayage entrelacé (30 Hz) | Echelonnabilité TVHD/SDTV |
| Balayage progressif (30 Hz) | Balayage entrelacé (30 Hz) | Compatibilité avec l'ISO/CEI 11172-2 ou avec la présente Spécification |
| Balayage entrelacé (30 Hz) | Balayage progressif (60 Hz) | Migration vers TVHD à haute résolution temporelle et balayage progressif |

Tableau Intro. 3 – Applications de l'échelonnabilité temporelle

| Couche de base | Couche d'amélioration | Couches supérieures | Application |
|-----------------------------|-----------------------------|-----------------------------|--|
| Balayage progressif (30 Hz) | Balayage progressif (30 Hz) | Balayage progressif (60 Hz) | Migration vers TVHD à haute résolution temporelle et balayage progressif |
| Balayage entrelacé (30 Hz) | Balayage entrelacé (30 Hz) | Balayage progressif (60 Hz) | Migration vers TVHD à haute résolution temporelle et balayage progressif |

Intro. 4.2.1 Extension à échelonnabilité spatiale

L'échelonnabilité spatiale est un outil destiné à être utilisé dans des applications vidéo telles que les télécommunications, l'interfonctionnement de standards vidéo, la consultation de bases de données vidéo, l'interfonctionnement TV-TVHD, etc., c'est-à-dire dans des systèmes vidéo dont la principale caractéristique commune est l'exigence d'au moins deux couches de résolution spatiale. L'échelonnabilité spatiale implique la production, à partir d'une seule source vidéo, de deux couches de résolution spatiale telles que la couche inférieure soit codée indépendamment pour fournir la couche de base de la résolution spatiale et que la couche d'amélioration utilise cette couche inférieure comme base d'interpolation spatiale pour apporter la pleine résolution spatiale de la source d'entrée vidéo. La couche de base inférieure et la couche d'amélioration peuvent soit utiliser toutes les deux les outils de codage décrits dans la présente Spécification, ou bien utiliser les outils de l'ISO/CEI 11172-2 pour la couche inférieure et ceux de la présente Spécification pour la couche supérieure d'amélioration. Cette dernière variante offre l'avantage supplémentaire de faciliter l'interfonctionnement entre normes de codage vidéo. En outre, l'échelonnabilité spatiale offre la possibilité de choisir le format vidéo à employer dans chaque couche. Elle permet aussi d'assurer une meilleure robustesse aux erreurs de transmission en acheminant les données plus importantes de la couche inférieure par un canal présentant de meilleures caractéristiques en termes de protection contre les erreurs, tandis que les données moins critiques de la couche d'amélioration peuvent être acheminées par un canal présentant une protection d'erreur de qualité moindre.

Intro. 4.2.2 Extension à échelonnabilité SNR

L'échelonnabilité SNR (*Signal-Noise Ratio*) (rapport signal sur bruit) est un outil destiné à être utilisé dans des applications vidéo telles que les télécommunications, les services vidéo de diverses qualités, la TV conventionnelle et la TVHD, c'est-à-dire dans des systèmes vidéo dont la principale caractéristique commune est l'exigence d'au moins deux couches de qualité vidéo. L'échelonnabilité SNR implique la production, à partir d'une seule source vidéo, de deux couches vidéo de même résolution spatiale, telles que la couche inférieure soit codée indépendamment pour fournir la couche de base de la qualité vidéo et que la couche d'amélioration soit codée de manière à renforcer cette couche inférieure. Une fois ajoutée à la couche inférieure, la couche d'amélioration apporte une meilleure qualité de reproduction de la source vidéo d'entrée. La couche de base inférieure et la couche d'amélioration peuvent soit utiliser toutes les deux les outils de codage décrits dans la présente Spécification, ou bien utiliser les outils de l'ISO/CEI 11172-2 pour la couche inférieure et ceux de la présente Spécification pour la couche supérieure. L'échelonnabilité SNR permet aussi d'assurer une meilleure robustesse aux erreurs de transmission en acheminant les données plus importantes de la couche inférieure par un canal présentant de meilleures caractéristiques en termes de protection contre les erreurs, tandis que les données moins critiques de la couche d'amélioration peuvent être acheminées par un canal présentant une protection d'erreur de qualité moindre.

Intro. 4.2.3 Extension à échelonnabilité temporelle

L'échelonnabilité temporelle est un outil destiné à être utilisé dans une gamme d'applications vidéo diverses, allant des télécommunications à la TVHD, pour lesquelles il peut être nécessaire d'assurer une migration vers des systèmes offrant une résolution temporelle supérieure à celle d'autres systèmes analogues. Souvent, les systèmes vidéo à faible résolution temporelle pourront être les systèmes existants actuellement ou les systèmes moins coûteux des premières générations, l'objectif étant d'introduire progressivement des systèmes plus évolués. L'échelonnabilité temporelle implique la stratification des images vidéo: alors que la couche inférieure est codée indépendamment pour fournir la base de temps principale, la couche d'amélioration subit un codage temporel prédictif par rapport à la couche inférieure. Une fois

décodées et démultiplexées dans le temps, ces deux couches fournissent la pleine résolution temporelle de la source vidéo. Les systèmes vidéo à faible résolution temporelle ne peuvent décoder que la couche inférieure pour fournir la résolution temporelle de base, tandis que les futurs systèmes plus évolués pourront décoder les deux couches et fournir des signaux de haute résolution temporelle tout en conservant l'interfonctionnement avec les systèmes vidéo des générations antérieures. Un avantage complémentaire de l'échelonnabilité temporelle est d'offrir une certaine élasticité aux erreurs de transmission étant donné que les données de la couche inférieure, qui sont les plus importantes, peuvent être acheminées par un canal présentant de meilleures caractéristiques en termes de protection contre les erreurs, tandis que les données moins critiques de la couche d'amélioration peuvent être acheminées par un canal présentant une moindre qualité en termes de protection contre les erreurs.

Intro. 4.2.4 Extension vers la subdivision des données

La subdivision des données est un outil destiné à être utilisé lorsque deux canaux sont disponibles pour la transmission et/ou pour l'enregistrement d'un flux binaire vidéo, comme ce peut être le cas dans les réseaux ATM, en radiodiffusion de Terre, avec des supports magnétiques, etc. Le flux binaire est subdivisé entre ces canaux de manière que ses parties les plus critiques (comme les en-têtes, les vecteurs de mouvement, les coefficients DCT) soient transmises dans le canal offrant les meilleures caractéristiques en termes de protection contre les erreurs et que les données moins critiques (telles que les coefficients DCT d'ordre supérieur) soient transmises dans le canal présentant une moindre qualité en termes de protection contre les erreurs. Ce procédé permet de minimiser les erreurs de type dégradation introduite dans le canal car les parties critiques du flux binaire sont dans un canal mieux protégé. Aucune donnée de ces deux canaux ne peut être traitée par un décodeur non destiné au décodage de flux binaires à subdivision des données.

NORME INTERNATIONALE

RECOMMANDATION UIT-T

TECHNOLOGIES DE L'INFORMATION – CODAGE GÉNÉRIQUE DES IMAGES ANIMÉES ET DU SON ASSOCIÉ: DONNÉES VIDÉO

1 Domaine d'application

La présente Recommandation | Norme internationale spécifie la représentation codée des informations d'image pour supports d'enregistrement numérique et vidéocommunications, ainsi que le processus de décodage correspondant. Cette représentation est compatible avec la transmission à débit binaire constant, la transmission à débit binaire variable, l'accès aléatoire, l'interconnexion des canaux, le décodage échelonné, l'édition du flux binaire ainsi que des fonctions spéciales telles que la lecture rapide en avant, la lecture rapide en arrière, le ralenti, la pause et les arrêts sur image. La présente Recommandation | Norme internationale est compatible en aval avec ISO/CEI 11172-2 et compatible aussi bien en amont qu'en aval avec les formats de télévision à définition améliorée (EDTV), de télévision à haute définition (TVHD) et de télévision conventionnelle (SDTV).

La présente Recommandation | Norme internationale est principalement applicable aux supports d'enregistrement numérique, à la vidéodiffusion et aux vidéocommunications. Les supports d'enregistrement peuvent être reliés au décodeur directement ou par l'intermédiaire de moyens de communication tels que des bus d'interconnexion, des réseaux locaux ou des liaisons de télécommunication.

2 Références normatives

Les Recommandations UIT-T et les Normes internationales suivantes contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente Recommandation | Norme internationale. Au moment de la publication, les éditions indiquées étaient en vigueur. Toute Recommandation ou Norme internationale est sujette à révision et les parties prenantes aux accords fondés sur la présente Recommandation | Norme internationale sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des Recommandations et Normes indiquées ci-après. Les membres de la CEI et de l'ISO possèdent le registre des Normes internationales en vigueur. Le Bureau de la normalisation des télécommunications (TSB) de l'UIT tient à jour une liste des Recommandations UIT-T en vigueur.

- Recommandations et Rapports du CCIR, 1990, XVII^e Assemblée plénière, Düsseldorf, 1990 – Volume XI – Partie 1 – Service de radiodiffusion (télévision) – Recommandation UIT-R B.T.601-3, *Paramètres de codage de télévision numérique pour studios*.
- Volumes X et XI (Partie 3) du CCIR – Recommandation UIT-R BR.648, *Enregistrement numérique des signaux audio*.
- Volumes X et XI (Partie 3) du CCIR – Rapport UIT-R 955-2, *Radiodiffusion sonore par satellite pour récepteurs portatifs et récepteurs dans des véhicules automobiles*.
- ISO/CEI 11172-1:1993, *Technologies de l'information – Codage de l'image animée et du son associé pour les supports de stockage numérique jusqu'à environ 1,5 Mbit/s – Partie 1: Systèmes*.
- ISO/CEI 11172-2:1993, *Technologies de l'information – Codage de l'image animée et du son associé pour les supports de stockage numérique jusqu'à environ 1,5 Mbit/s – Partie 2: Vidéo*.
- ISO/CEI 11172-3:1993, *Technologies de l'information – Codage de l'image animée et du son associé pour les supports de stockage numérique jusqu'à environ 1,5 Mbit/s – Partie 3: Audio*.
- IEEE Standard Specifications for the Implementations of 8 by 8 Inverse Discrete Cosine Transform (Spécifications de norme IEEE pour les applications de la DCT inverse à des blocs de 8 × 8 éléments), IEEE Std. 1180-1990, 6 décembre 1990.

- Publication CEI 908:1987, *Système audionumérique à disque compact*.
- Publication CEI 461:1986, *Code temporel de commande pour les magnétoscopes*.
- Recommandation UIT-T H.261 (1993), *Codec vidéo pour services audiovisuels à $p \times 64$ kbit/s*.
- Recommandation T.81 du CCITT (1992) (JPEG) | ISO/CEI 10918-1:1994, *Technologie de l'information – Compression numérique et codage des images fixes de nature photographique – Prescriptions et lignes directrices*.

3 Définitions

Pour les besoins de la présente Recommandation | Norme internationale, les définitions suivantes s'appliquent.

- 3.1 coefficient AC:** tout coefficient DCT pour lequel la fréquence n'est pas nulle dans une dimension ou dans les deux dimensions.
- 3.2 image non décodée:** image codée qui provoquerait le sous-remplissage de la mémoire tampon VBV selon la définition donnée en C.7. De telles images ne peuvent apparaître qu'à l'intérieur de séquences dont le champ `low_delay` est mis à la valeur '1'. L'expression «image sautée» désigne parfois le même concept.
- 3.3 image B monotrame:** image B structurée en trame unique.
- 3.4 image B bi-trame:** image B structurée en deux trames.
- 3.5 image B; image à codage prédictif bidirectionnel:** image qui est codée au moyen de la prédiction utilisant la compensation de mouvement à partir de précédentes et/ou futures trames de référence uniques ou doubles.
- 3.6 compatibilité amont:** une nouvelle norme de codage est dite à compatibilité amont avec une ancienne norme de codage si les décodeurs étudiés pour fonctionner avec l'ancienne norme de codage sont en mesure de continuer à fonctionner en décodant tout ou partie du flux binaire produit conformément à la nouvelle norme de codage.
- 3.7 vecteur de mouvement différé:** vecteur de mouvement qui est utilisé pour la compensation de mouvement à partir d'une trame de référence antérieure à ce vecteur dans l'ordre d'affichage.
- 3.8 prédiction différée:** prévision à partir de trames de référence futures.
- 3.9 couche de base:** première couche indépendamment décodable d'une hiérarchie échelonnée.
- 3.10 flux binaire; flux:** série ordonnée d'éléments binaires qui constitue la représentation codée des données.
- 3.11 débit binaire:** débit auquel le flux binaire codé est présenté à l'entrée d'un décodeur à partir d'un support d'enregistrement.
- 3.12 bloc:** matrice de 8 lignes sur 8 colonnes de pels, soit 64 coefficients DCT (originaux, quantifiés ou déquantifiés).
- 3.13 trame inférieure:** l'une des deux trames qui composent une image bi-trame. Chaque ligne d'une trame inférieure est spatialement localisée immédiatement au-dessous de la ligne homologue de la trame supérieure.
- 3.14 limite d'octet:** un bit de flux binaire codé est dit sur une limite d'octet si sa position correspond à un multiple de 8 comptée à partir du premier bit du flux.
- 3.15 octet:** mot de 8 bits.
- 3.16 canal:** support numérique qui conserve ou transporte un flux binaire constitué conformément à la Rec. UIT-T H.262 | ISO/CEI 13818-2.
- 3.17 format de chrominance:** nombre de blocs de chrominance d'un macrobloc.
- 3.18 diffusion jumelée de données de chrominance:** type d'échelonnabilité (qui est un sous-ensemble de l'échelonnabilité SNR), où la (les) couche(s) d'amélioration ne contient(en)t que des données codées de raffinement pour les coefficients DC et toutes les données pour les coefficients AC, des composantes de chrominance.

- 3.19 chrominance (composante de):** matrice, bloc ou échantillon isolé qui représente un des deux signaux de différence de couleur et qui s'applique aux couleurs primaires selon le mode défini dans le flux binaire. Les symboles utilisés pour les signaux de chrominance sont Cr et Cb.
- 3.20 image bi-trame à codage B:** image B à deux trames ou paire d'images B à trame unique.
- 3.21 image bi-trame codée:** image bi-trame à codage I (intra), P (prédicatif) ou B (bidirectionnel).
- 3.22 image bi-trame à codage I:** image I à deux trames ou paire d'images à trame unique, la première image étant de type I et la deuxième une image de type I ou P.
- 3.23 image bi-trame à codage P:** image P à deux trames ou paire d'images P à trame unique.
- 3.24 image codée:** image composée d'un en-tête d'image, des extensions facultatives qui le suivent immédiatement et des données d'image ultérieures. Une image codée peut être bi-trame ou monotrème.
- 3.25 flux binaire de données vidéo codées:** représentation codée d'une série composée d'une ou de plusieurs images telles que définies dans la Rec. UIT-T H.262 | ISO/CEI 13818-2.
- 3.26 ordre codé:** ordre dans lequel les images sont transmises et décodées. Cet ordre n'est pas nécessairement le même que l'ordre d'affichage.
- 3.27 représentation codée:** élément de données représenté sous forme codée.
- 3.28 paramètres de codage:** ensemble de paramètres définissables par l'utilisateur, qui caractérise un flux binaire de données vidéo. Les flux binaires sont caractérisés par des paramètres de codage. Les décodeurs sont caractérisés par les flux binaires qu'ils sont capables de décoder.
- 3.29 composante:** matrice, bloc ou échantillon isolé dans une des trois matrices (une de luminance et deux de chrominance) constituant une image.
- 3.30 compression:** réduction du nombre d'éléments binaires utilisés pour représenter un élément de données.
- 3.31 données vidéo codées à débit binaire constant:** flux de données vidéo comprimées dont le débit binaire moyen est constant.
- 3.32 débit binaire constant:** exploitation telle que le débit binaire est constant du début à la fin du flux binaire codé.
- 3.33 élément de données:** représentation d'une donnée avant codage et après décodage.
- 3.34 subdivision des données:** méthode consistant à doubler un flux binaire afin d'augmenter l'élasticité aux erreurs. Les deux flux binaires doivent être recombinaés avant décodage.
- 3.35 image D:** type d'image qui ne doit pas être utilisé, sauf dans l'ISO/CEI 11172-2.
- 3.36 coefficient DC:** coefficient DCT pour lequel la fréquence est nulle dans les deux dimensions.
- 3.37 coefficient DCT:** amplitude d'une fonction spécifique à base cosinus.
- 3.38 tampon d'entrée de décodeur:** tampon de type premier entré-premier sorti (FIFO), spécifié dans le vérificateur de mémoire vidéo.
- 3.39 décodeur:** matérialisation d'un processus de décodage.
- 3.40 décodage (processus):** processus défini dans la Rec. UIT-T H.262 | ISO/CEI 13818-2, consistant à lire un flux binaire codé entrant et à produire après décodage des images ou des échantillons audio.
- 3.41 quantification inverse:** processus de remise à l'échelle des coefficients DCT quantifiés, après le décodage de leur représentation dans le flux binaire et avant leur présentation à l'opération de transformation DCT inverse.
- 3.42 support d'enregistrement numérique (DSM) (Digital Storage Media):** dispositif ou système d'enregistrement ou de transmission numérique.

- 3.43 transformée en cosinus discrète (DCT)** (*Discrete Cosine Transform*): résultat d'une transformation en cosinus, discrète, directe ou inverse. L'opération DCT est une transformation orthogonale discrète, qui peut être inversée. La DCT inverse est définie dans l'Annexe A de la Rec. UIT-T H.262 | ISO/CEI 13818-2.
- 3.44 format d'affichage:** rapport hauteur/largeur (exprimées en unités SI) de l'affichage prévu.
- 3.45 ordre d'affichage:** ordre dans lequel les images décodées sont affichées. Il s'agit normalement du même ordre que celui dans lequel ces images ont été présentées à l'entrée du décodeur.
- 3.46 processus d'affichage:** processus (non normatif) permettant l'affichage des images reconstruites.
- 3.47 prédiction anticipée à double polarité:** mode prédictif consistant à mettre en moyenne deux prédictions monotrames anticipées. La taille de bloc prédite est de 16×16 échantillons de luminance. Les prédictions anticipées à double polarité ne sont utilisées que pour les images à balayage entrelacé.
- 3.48 édition:** opération consistant à manipuler un ou plusieurs flux binaires de données codées afin d'obtenir un nouveau flux binaire de données codées. Les flux binaires convenablement édités doivent toujours répondre aux prescriptions définies dans la Rec. UIT-T H.262 | ISO/CEI 13818-2.
- 3.49 codeur:** matérialisation d'un processus de codage.
- 3.50 codage (processus de):** processus, non spécifié dans la Rec. UIT-T H.262 | ISO/CEI 13818-2, qui effectue la lecture d'un train d'images d'entrée et qui produit un flux binaire valide, tel que défini dans la Rec. UIT-T H.262 | ISO/CEI 13818-2.
- 3.51 couche d'amélioration:** référence relative à une couche (au-dessus de la couche de base) à l'intérieur d'une hiérarchie échelonnée. Indépendamment de la forme d'échelonnabilité, il est possible de définir son processus de décodage par rapport au processus de décodage de la couche inférieure et par rapport au processus de décodage supplémentaire approprié concernant la couche d'amélioration proprement dite.
- 3.52 lecture rapide en avant:** processus d'affichage rapide d'une séquence d'images, ou de parties de séquence d'images, dans leur ordre d'affichage.
- 3.53 lecture rapide en arrière:** processus d'affichage rapide d'une séquence d'images dans l'ordre inverse de leur affichage.
- 3.54 trame:** pour un signal vidéo entrelacé, ensemble de lignes alternées d'une image bi-trame. Une image bi-trame entrelacée se compose donc de deux trames: la trame supérieure et la trame inférieure.
- 3.55 prédiction monotrame:** mode prédictif utilisant seulement une trame de l'image de référence. La taille de bloc prédite correspond à 16×16 échantillons de luminance.
- 3.56 période de trame:** inverse du double de la fréquence d'image.
- 3.57 image monotrame; image à structure monotrame:** image codée dont l'élément structurel `picture_structure` a la valeur «Top field» ou «Bottom field».
- 3.58 indicateur:** variable entière de 1 bit qui peut prendre une valeur parmi deux (zéro ou un).
- 3.59 interdite:** qualificatif d'une valeur qui ne doit jamais être utilisée dans les articles définissant le flux binaire codé. Cela permet généralement d'éviter l'émulation de codes de déclenchement.
- 3.60 mise à jour forcée:** processus par lequel des macroblocs subissent de temps en temps un codage intra pour veiller à ce qu'il n'y ait pas d'accumulation excessive, dans les codeurs et dans les décodeurs, d'erreurs de correspondance entre les processus de transformation DCT inverse.
- 3.61 compatibilité aval:** une nouvelle norme de codage est dite à compatibilité aval avec une ancienne norme de codage si les décodeurs étudiés pour fonctionner avec la nouvelle norme de codage sont en mesure de décoder les flux binaires produits conformément à l'ancienne norme de codage.
- 3.62 vecteur de mouvement anticipé:** vecteur de mouvement qui est utilisé pour la compensation de mouvement à partir d'une ou de deux trames de référence postérieures à ce vecteur dans l'ordre d'affichage.
- 3.63 prédiction anticipée:** prédiction à partir de l'image (trame) de référence passée.
- 3.64 image bi-trame:** ensemble de lignes d'information spatiale acheminées par un signal vidéo. En balayage progressif, ces lignes contiennent des pels qui commencent à partir d'un certain instant et qui forment des lignes successives jusqu'en bas de l'image. En balayage entrelacé, une image bi-trame se compose de deux trames: la trame supérieure et la trame inférieure. L'une de ces trames commencera une période de trame plus tard que l'autre.

- 3.65 prédiction bi-trame:** mode prédictif utilisant les deux trames de l'image de référence.
- 3.66 période d'image:** inverse de la fréquence image.
- 3.67 image bi-trame; image à structure bi-trame:** image codée dont l'élément structurel `picture_structure` a la valeur «Frame».
- 3.68 fréquence image:** fréquence à laquelle les images bi-trames doivent sortir du processus de décodage.
- 3.69 trame de référence postérieure:** trame qui se construit postérieurement à l'image actuelle, dans l'ordre d'affichage.
- 3.70 réordonnement de trame:** processus de réordonnement des images bi-trames reconstruites lorsque l'ordre de codage diffère de l'ordre d'affichage. Il y a réordonnement des images bi-trames en présence d'images à codage bidirectionnel dans un flux binaire. Il n'y a pas de réordonnement des images bi-trames lors du décodage de flux binaires à faible délai.
- 3.71 groupe d'images:** notion définie uniquement dans ISO/CEI 11172-2 (MPEG-1 vidéo). Dans la Rec. UIT-T H.262 | ISO/CEI 13818-2, l'insertion d'en-têtes de groupes d'images permet d'obtenir une fonctionnalité analogue.
- 3.72 en-tête:** bloc de données dans le flux binaire codé, contenant la représentation codée d'un certain nombre d'éléments de données se rapportant aux données codées faisant suite à l'en-tête dans le flux binaire.
- 3.73 échelonnabilité hybride:** combinaison de deux (ou plus de deux) types d'échelonnabilité.
- 3.74 entrelacement:** caractéristique de balayage en télévision conventionnelle où une alternance de lignes de trame représente différents instants sur l'échelle du temps. Dans une image bi-trame entrelacée, l'une des trames est censée être balayée en premier: c'est la première trame, qui peut être la trame supérieure ou la trame inférieure de l'image bi-trame.
- 3.75 image I monotrane:** image I structurée en trame unique.
- 3.76 image I bi-trame:** image I structurée en deux trames.
- 3.77 image I; image à codage intra:** image codée au moyen d'informations ne provenant que de cette image.
- 3.78 codage intra:** codage d'un macrobloc ou d'une image au moyen d'informations ne provenant que de ce macrobloc ou de cette image.
- 3.79 niveau:** ensemble défini de contraintes sur les valeurs que peuvent prendre les paramètres de la Rec. UIT-T H.262 | ISO/CEI 13818-2 dans un profil particulier. Chaque profil peut contenir un ou plusieurs niveaux. Dans un contexte différent, le terme niveau désigne la valeur absolue d'un coefficient non nul (voir «exécution»).
- 3.80 couche:** dans une hiérarchie échelonnée, désigne l'un des ensembles ordonnés de flux binaires et (le résultat de) son processus de décodage (comportant implicitement le décodage de **toutes** les couches situées au-dessous).
- 3.81 flux binaire de couche:** flux binaire simple associé à une couche spécifique (toujours associé à des qualificatifs de couche, par exemple «flux binaire de couche d'amélioration»).
- 3.82 couche inférieure:** référence relative à la couche située immédiatement au-dessous d'une couche d'amélioration donnée (comportant implicitement le décodage de **toutes** les couches situées au-dessous de cette couche d'amélioration).
- 3.83 composante de luminance:** matrice, bloc ou pel isolé qui représente les informations monochromes du signal et qui s'applique aux couleurs primaires selon le mode défini dans le flux binaire. Le symbole utilisé pour la luminance (ou clarté) est *Y*.
- 3.84 Mbit:** symbole du mégabit, soit 1 000 000 bits.
- 3.85 macrobloc:** ensemble des quatre blocs de 8×8 éléments de données de luminance et des deux (en format de chrominance 4:2:0), quatre (en format de chrominance 4:2:2) ou huit (en format de chrominance 4:4:4) blocs de 8×8 éléments de données de chrominance correspondants, issus d'une section de 16×16 éléments de la composante de luminance de l'image. Le terme de *macrobloc* est utilisé pour faire référence parfois aux données échantillonnées et parfois à la représentation codée des valeurs échantillonnées et à d'autres éléments de données définis dans l'en-tête du macrobloc selon la syntaxe indiquée dans la Rec. UIT-T H.262 | ISO/CEI 13818-2. C'est le contexte qui détermine l'usage de ce terme.

- 3.86 compensation de mouvement:** utilisation de vecteurs de mouvement pour améliorer l'efficacité de prédiction de valeurs d'échantillon. La prédiction utilise les vecteurs de mouvement pour fournir des segments représentatifs de trames de référence antérieures et/ou postérieures, contenant des valeurs d'échantillon déjà décodées qui servent à former le signal d'erreur de prédiction.
- 3.87 estimation de mouvement:** processus d'estimation des vecteurs de mouvement au cours du processus de codage.
- 3.88 vecteur de mouvement:** vecteur à deux dimensions utilisé pour la compensation de mouvement en tant que segment allant d'une position définie par des coordonnées dans l'image actuelle jusqu'à une autre position définie par des coordonnées dans une image ou trame de référence.
- 3.89 codage non intra:** codage d'un macrobloc ou d'une image utilisant des informations issues à la fois de ce macrobloc ou de cette image et de macroblocs ou images construits à d'autres instants.
- 3.90 parité opposée:** la parité opposée de supérieure est inférieure et vice versa.
- 3.91 image P monotrame:** image prédite P structurée en trame unique.
- 3.92 image P bi-trame:** image prédite P structurée en deux trames.
- 3.93 image P; image à codage prédictif:** image codée par prédiction utilisant la compensation de mouvement à partir de trames ou images de référence antérieures.
- 3.94 paramètre:** variable syntaxique de la Rec. UIT-T H.262 | ISO/CEI 13818-2 qui peut prendre une valeur parmi une série de valeurs possibles. Une variable qui ne peut prendre que deux valeurs est un fanion et non un paramètre.
- 3.95 parité (d'une trame):** la parité d'une trame peut être paire ou impaire.
- 3.96 image bi-trame (monotrame) de référence antérieure:** image bi-trame (monotrame) qui se construit antérieurement à l'image actuelle, dans l'ordre d'affichage.
- 3.97 image:** données d'image originales, codées ou reconstruites. Une image originale ou reconstruite se compose de trois matrices rectangulaires de nombres de 8 bits, représentant le signal de luminance et les deux signaux de chrominance. La définition d'une «image codée» figure en 3.21 de la Rec. UIT-T H.262 | ISO/CEI 13818-2. En vidéo à balayage progressif, une image correspond à une bi-trame, alors qu'en balayage entrelacé, une image peut être une bi-trame ou la trame supérieure ou la trame inférieure d'une bi-trame, selon le contexte.
- 3.98 données d'image:** dans le contexte de la vérification des données vidéo, on entend par données d'image tous les bits de l'image codée, ainsi que toutes les données d'en-tête et les données d'utilisateur qui, le cas échéant, la précèdent immédiatement (y compris les bits de bourrage intercalés), et les éléments de bourrage qui la suivent, jusqu'au code de déclenchement suivant (mais sans inclure celui-ci), sauf lorsque le code de déclenchement suivant est un code de fin de séquence, auquel cas ledit code est inclus dans les données image.
- 3.99 prédiction:** utilisation d'un prédicteur pour fournir une estimation de la valeur d'échantillon ou de l'élément de données en cours de décodage.
- 3.100 erreur de prédiction:** différence entre la valeur réelle d'un échantillon ou d'un élément de données et la valeur indiquée par son prédicteur.
- 3.101 prédicteur:** combinaison linéaire de valeurs déjà décodées d'un échantillon ou d'un élément de données.
- 3.102 profil:** sous-ensemble syntaxique défini dans la Rec. UIT-T H.262 | ISO/CEI 13818-2.
- NOTE – Dans la Rec. UIT-T H.262 | ISO/CEI 13818-2, le terme «profil» a le sens indiqué ci-dessus. Il convient de ne pas le confondre avec d'autres acceptions; en particulier, il n'a pas le sens défini par le Groupe spécial pour la normalisation fonctionnelle (SGFS) de ISO/CEI JTC 1.
- 3.103 progressif (balayage):** caractéristique d'images cinématographiques où chaque échantillon représente toujours le même segment temporel.
- 3.104 matrice de quantification:** ensemble de 64 valeurs de 8 bits (octets), utilisé par le quantificateur inverse.
- 3.105 coefficients DCT quantifiés:** coefficients DCT avant quantification inverse. Une représentation en codage de longueur variable des coefficients DCT quantifiés est acheminée dans le flux binaire vidéo comprimé.
- 3.106 échelle de quantification:** facteur d'échelle codé dans le flux binaire et utilisé par le processus de décodage pour échelonner la quantification inverse.
- 3.107 accès aléatoire:** processus consistant à commencer à un point arbitraire la lecture et le décodage du flux binaire codé.

- 3.108 image bi-trame reconstruite:** ensemble de trois matrices rectangulaires d'octets représentant le signal de luminance et les deux signaux de chrominance. On obtient une image bi-trame reconstruite en décodant une image bi-trame codée.
- 3.109 image reconstruite:** image obtenue par décodage d'une image codée. Une image reconstruite est soit une image bi-trame reconstruite (après décodage d'une image bi-trame) ou une seule trame d'une image bi-trame (après décodage d'une image monotrème). Si l'image codée est une image monotrème, l'image reconstruite est la trame supérieure ou la trame inférieure de l'image bi-trame reconstruite.
- 3.110 trame de référence:** une des deux trames d'une image bi-trame reconstruite. Les trames de référence sont utilisées pour la prédiction anticipée et pour la prédiction différée lors du décodage d'images P et d'images B. A noter que lors du décodage d'images P monotrèmes, la prédiction de la deuxième image P monotrème d'une image bi-trame codée utilise la première trame reconstruite de cette même image bi-trame codée, en tant que trame de référence.
- 3.111 image bi-trame de référence:** image bi-trame reconstruite après avoir été codée sous la forme d'une image I ou P. Les images bi-trames de référence sont utilisées pour la prédiction anticipée et pour la prédiction différée lors du décodage d'images P et d'images B.
- 3.112 délai de réordonnement:** délai au cours du processus de décodage dû au réordonnement des images bi-trames.
- 3.113 (valeur) réservée:** dans les articles définissant le flux binaire codé, une valeur est dite réservée si elle peut être utilisée dans les futures extensions définies par l'UIT-T | ISO/CEI.
- 3.114 format d'échantillon (SAR) (*Sample Aspect Ratio*):** distance entre échantillons, définie (dans la Rec. UIT-T H.262 | ISO/CEI 13818-2) comme étant le déplacement vertical des lignes d'échantillons de luminance dans une image bi-trame, divisé par le déplacement horizontal des échantillons de luminance. Ce format est donc exprimé en (mètres par ligne) ÷ (mètres par échantillon).
- 3.115 hiérarchie échelonnable:** données vidéo codées constituées d'un ensemble ordonné de plusieurs flux binaires vidéo.
- 3.116 échelonnabilité:** capacité d'un décodeur d'analyser un ensemble ordonné de flux binaires afin de produire une séquence reconstruite. Des données vidéo utiles sont produites lorsque les sous-ensembles sont également décodés. Le sous-ensemble élémentaire qui peut ainsi être décodé est le premier flux binaire de l'ensemble, appelé couche de base. Chacun des autres flux binaires de cet ensemble est appelé couche d'amélioration. Dans le cas d'une couche d'amélioration spécifique, le terme «couche inférieure» désigne le flux binaire qui précède la couche d'amélioration.
- 3.117 information collatérale:** information, contenue dans le flux binaire, qui est nécessaire pour commander le décodeur.
- 3.118 prédiction 16 × 8:** mode prédictif analogue à la prédiction monotrème, mais selon lequel la taille de bloc prédite correspond à 16 × 8 échantillons de luminance.
- 3.119 exécution:** nombre de zéros qui précèdent un coefficient non nul dans l'ordre de balayage. La valeur absolue du coefficient non nul est appelée «niveau».
- 3.120 saturation:** consiste à limiter une valeur qui sort d'un domaine de variation défini en lui assignant la valeur maximale ou minimale du domaine de variation considéré, selon le cas.
- 3.121 macrobloc sauté:** macrobloc pour lequel aucune donnée n'est codée.
- 3.122 tranche:** série continue de macroblocs faisant tous partie de la même rangée horizontale de macroblocs.
- 3.123 échelonnabilité SNR:** type d'échelonnabilité dans laquelle la (les) couche(s) d'amélioration ne contient (contiennent) que des données codées de raffinement pour les coefficients DCT de la couche inférieure.
- 3.124 données d'origine; données d'entrée:** expression désignant les données vidéo ou certains de leurs attributs avant l'opération de codage.
- 3.125 prédiction spatiale:** prédiction fondée sur une image bi-trame décodée du décodeur de la couche inférieure utilisée dans le cadre de l'échelonnabilité spatiale.
- 3.126 échelonnabilité spatiale:** type d'échelonnabilité selon lequel une couche d'amélioration utilise également des prédictions fondées sur des données d'échantillon appartenant à une couche inférieure, sans faire appel à des vecteurs de mouvement. Les couches peuvent avoir divers types de dimensions d'image, de fréquence d'image ou de format de chrominance.

- 3.127 codes de départ ou déclenchement (système et vidéo):** mots codés uniques à 32 bits insérés dans un flux binaire codé, servant entre autres à l'identification de certaines structures dans la syntaxe de codage.
- 3.128 remplissage (bits ou octets de):** mots de code qui peuvent être insérés dans le flux binaire codé, et qui sont ignorés par le processus de décodage. Leur fonction est d'accroître le débit du flux binaire, jusqu'à la valeur souhaitée.
- 3.129 prédiction temporelle:** prédiction établie à partir d'une image bi-trame décodée du décodeur de la couche inférieure utilisée dans le contexte de l'échelonnabilité spatiale.
- 3.130 échelonnabilité temporelle:** type d'échelonnabilité tel qu'une couche d'amélioration utilise également des prédictions fondées sur des données d'échantillon issues d'une couche inférieure utilisant des vecteurs de mouvement. Les couches ont des types identiques de dimensions d'image et de format de chrominance mais peuvent avoir des fréquences d'image différentes.
- 3.131 trame supérieure:** une des deux trames qui composent une image bi-trame. Chaque ligne d'une trame supérieure est spatialement située immédiatement au-dessus de la ligne correspondante de la trame inférieure.
- 3.132 couche supérieure:** couche la plus élevée (dotée de l'identificateur `layer_id` dont la valeur est la plus élevée) d'une hiérarchie échelonnée.
- 3.133 débit binaire variable:** exploitation telle que le débit binaire varie dans le temps au cours du décodage d'un flux binaire codé.
- 3.134 codage à longueur variable (VLC) (*Variable Length Coding*):** procédure de codage réversible qui assigne des mots de code courts à des événements fréquents et des mots de code longs aux événements moins fréquents.
- 3.135 vérificateur de mémoire vidéo (VBV) (*Video Buffering Verifier*):** décodeur hypothétique qui est théoriquement raccordé à la sortie du codeur. Sa fonction consiste à exercer une contrainte sur la variabilité du débit de données qu'un codeur ou un processus d'édition peut assurer.
- 3.136 séquence vidéo:** structure syntaxique du niveau le plus élevé pour un flux binaire de données vidéo codées. Cette structure contient une série composée d'une ou de plusieurs images bi-frames codées.
- 3.137 décodeur de profil xxx:** décodeur permettant de décoder un flux binaire ou une hiérarchie échelonnée de flux binaires dont la couche supérieure est conforme aux spécifications concernant le profil xxx (xxx désignant l'un des noms de profils définis).
- 3.138 hiérarchie échelonnée de profil xxx:** ensemble de flux binaires dont la couche supérieure est conforme aux spécifications concernant le profil xxx.
- 3.139 flux binaire de profil xxx:** flux binaire d'une hiérarchie échelonnée comportant une indication de profil correspondant à xxx. On notera que ce flux binaire ne peut être décodé qu'avec tous les flux binaires de sa couche inférieure (sauf s'il s'agit d'un flux binaire de la couche de base).
- 3.140 ordre de balayage en zigzag:** ordonnancement séquentiel spécifique des coefficients DCT en allant (généralement) de la plus petite à la plus grande fréquence spatiale.

4 Abréviations et symboles

Les opérateurs mathématiques utilisés pour décrire la présente Spécification sont similaires aux opérateurs employés dans le langage de programmation C. Les divisions entières avec troncature et les arrondis seront toutefois définis spécifiquement. Les boucles de numérotation et de comptage commencent généralement à zéro.

4.1 Opérateurs arithmétiques

| | |
|-------------|---|
| + | Addition |
| - | Soustraction (dans le cas d'un opérateur binaire) ou négativation (dans le cas d'un opérateur unaire) |
| ++ | Incrémententation, c'est-à-dire $x++$ est équivalent à $x = x + 1$ |
| -- | Décrémententation, c'est-à-dire $x--$ est équivalent à $x = x - 1$ |
| $\times, *$ | Multiplication |

| | |
|----------------------|---|
| ^ | Puissance |
| / | Division entière avec troncature du résultat vers zéro. Par exemple, les fractions 7/4 et -7/-4 sont tronquées à 1 et les fractions -7/4 et 7/-4 sont tronquées à -1. |
| // | Division entière avec arrondi à l'entier le plus proche. Les valeurs milieu sont arrondies par excès, sauf indication contraire. Par exemple, 3//2 sera arrondi à 2 et -3//2 sera arrondi à -2. |
| DIV | Division entière avec arrondi à l'entier inférieur. Par exemple, 3 DIV 2 sera arrondi à 1 et -3 DIV 2 à -2. |
| ÷ | Symbole utilisé pour désigner une division dans des équations mathématiques non destinées à faire l'objet d'une troncature ou d'un arrondi. |
| % | Opérateur de base arithmétique (modulo), défini seulement pour des nombres positifs. |
| Sign() | $\text{Sign}(x) = \begin{cases} 1 & x > 0, \\ 0 & x == 0, \\ -1 & x < 0 \end{cases}$ |
| Abs() | $\text{Abs}(x) = \begin{cases} x & x \geq 0, \\ -x & x < 0 \end{cases}$ |
| $\sum_{i=a}^{i < b}$ | Sommation des $f(i)$, i prenant toutes les valeurs entières de a à b , b non compris. |

4.2 Opérateurs logiques

| | |
|----|-------------|
| | OU logique |
| && | ET logique |
| ! | NON logique |

4.3 Opérateurs relationnels

| | |
|---------------|---|
| > | Supérieur à |
| >= | Supérieur ou égal à |
| < | Inférieur à |
| <= | Inférieur ou égal à |
| == | Egal à |
| != | Différent de |
| max [, ... ,] | Valeur maximale dans la liste d'arguments |
| min [, ... ,] | Valeur minimale dans la liste d'arguments |

4.4 Opérateurs binaires

| | |
|----|--|
| & | ET |
| | OU |
| >> | Décalage à droite avec remplissage à gauche par bit de signe |
| << | Décalage à gauche avec remplissage de zéros à droite |

4.5 Affectation

| | |
|---|-------------------------|
| = | Opérateur d'affectation |
|---|-------------------------|

4.6 Mnémoniques

Les symboles mnémoniques suivants sont définis pour décrire les différents types de données utilisés dans le flux binaire codé.

- bslbf** (bit string, left bit first) Chaîne binaire dont le premier élément est à gauche, le terme à *gauche* correspondant à l'ordre dans lequel les chaînes binaires sont écrites dans la présente Spécification. Les chaînes binaires sont généralement écrites sous la forme d'une suite de 1 et de 0 logiques, encadrée par des apostrophes (par exemple '1000 0001'). Les blancs à l'intérieur d'une chaîne binaire ont pour objet de faciliter la lecture et n'ont pas de signification particulière. Par souci de commodité, les longues chaînes sont parfois écrites en notation hexadécimale: dans ce cas, la conversion habituelle en binaire donnera la valeur de la chaîne binaire. Le chiffre hexadécimal le plus à gauche vient en premier et, dans chaque chiffre hexadécimal, le premier des quatre éléments binaires est celui de poids fort.
- uimbsf** (unsigned integer, most significant bit first) Entier non signé, bit le plus significatif en premier.
- simsbf** (signed integer, most significant bit first) Entier signé, en format de complément à deux, bit le plus significatif (signe) en premier.
- vlclbf** (variable length code, left bit first) Code de longueur variable, bit de gauche en premier, où «gauche» désigne l'ordre dans lequel les codes de longueur variable sont écrits. L'ordre des octets pour les mots à plusieurs octets consiste à mettre l'octet le plus significatif en premier.

4.7 Constantes

π 3,141 592 653 58...

e 2,718 281 828 45...

5 Conventions

5.1 Méthode de description de la syntaxe du flux binaire

Le flux binaire récupéré par le décodeur est décrit au paragraphe 6.2. Chaque donnée élémentaire du flux binaire est écrite en caractères gras. Elle est décrite par son nom, par sa longueur en bits, et par un symbole mnémonique indiquant son type et son ordre de transmission.

L'action provoquée par un élément de données décodé dans un flux binaire dépend de la valeur de cet élément et des éléments de données déjà décodés. Le décodage des éléments de données et la définition des variables descriptrices d'état utilisées pour ce décodage figurent au paragraphe 6.3. Les structures suivantes, en caractères normaux, sont créées pour exprimer les conditions affectant les éléments de données présents.

| | |
|--|--|
| <pre>while (condition) { data_element ... }</pre> | <p>Si la condition est vraie, le groupe d'éléments de données apparaîtra le premier dans le flux de données, et ce jusqu'à ce que la condition ne soit plus vraie.</p> |
| <pre>do { data_element ... } while (condition)</pre> | <p>L'élément de données apparaît toujours au moins une fois.</p> <p>L'élément de données est répété jusqu'à ce que la condition ne soit plus vraie.</p> |
| <pre>if (condition) { data_element ... } else { data_element ... }</pre> | <p>Si la condition est vraie, le premier groupe d'éléments de données apparaîtra en premier dans le flux de données.</p> <p>Si la condition n'est pas vraie, le deuxième groupe d'éléments de données apparaîtra en premier dans le flux de données.</p> |
| <pre>for (i = m; i < n; i++) { data_element ... }</pre> | <p>Le groupe d'éléments de données apparaît (m – n) fois. L'insertion de structures conditionnelles dans ce groupe d'éléments de données peut dépendre de la valeur de la variable i de commande d'itérations, qui est mise à zéro pour la première apparition, incrémentée à un pour la deuxième apparition, et ainsi de suite.</p> |
| <pre>/* commentaire ... */</pre> | <p>Commentaire explicatif qui peut être supprimé entièrement sans que cela exerce une quelconque influence sur la syntaxe.</p> |

Cette syntaxe utilise la convention du «Code C» selon laquelle une variable ou expression évaluée comme différente de zéro équivaut à une condition qui est vraie tandis qu'une variable ou expression évaluée comme égale à zéro équivaut à une condition qui est fausse. Une chaîne littérale est souvent utilisée dans une condition. Par exemple:

```
if ( scalable_mode == «échelonnabilité spatiale» ) ...
```

Dans de tels cas, la chaîne littérale est celle qui est utilisée au paragraphe 6.3 pour décrire la valeur d'un élément du flux binaire. Dans cet exemple, on constate que le terme «échelonnabilité spatiale» est défini dans le Tableau 6-10 comme étant représenté par le nombre binaire '01' (deux bits).

Comme indiqué plus haut, le groupe d'éléments de données peut contenir des structures conditionnelles imbriquées. Par souci de concision, les accolades { } seront omises s'il n'y a qu'un seul élément de données en cause.

data_element [n] l'élément de données [n] est le n + 1^{ème} élément d'un tableau de données.

data_element [m][n] l'élément de données [m][n] est le m + 1, n + 1^{ème} élément d'un tableau de données à deux dimensions.

data_element [1][m][n] l'élément de données [1][m][n] est le 1 + 1, m + 1, n + 1^{ème} élément d'un tableau de données à trois dimensions.

Bien que la syntaxe soit exprimée en termes de procédures, il convient de ne pas en déduire que le paragraphe 6.2 met en œuvre une procédure de décodage satisfaisante. Ce paragraphe définit en particulier un flux binaire d'entrée correct et sans erreur. Les décodeurs réels devront toujours comporter la possibilité de détecter les codes de déclenchement de façon à commencer correctement le décodage et à identifier les erreurs, les effacements ou les insertions au cours du décodage. Les méthodes de détection de tels états et les mesures à prendre à leur sujet ne sont pas normalisées.

5.2 Définition des fonctions

On définira ci-dessous plusieurs fonctions utilitaires pour l'algorithme de codage des images.

5.2.1 Fonction `bytealigned()`

La fonction `bytealigned()` renvoie la valeur 1 si la position actuelle est sur une limite d'octet, c'est-à-dire que le prochain bit du flux binaire est le premier bit d'un octet. Dans le cas contraire, elle renvoie la valeur 0.

5.2.2 Fonction `nextbits()`

La fonction `nextbits()` permet la comparaison d'une chaîne binaire avec les prochains bits à décoder dans le flux binaire.

5.2.3 Fonction `next_start_code()`

La fonction `next_start_code()` élimine tout remplissage par bits ou octets nuls puis localise le prochain code de déclenchement.

| <code>next_start_code() {</code> | Nombre de bits | Mnémonique |
|--|----------------|-------------|
| <code>while (!bytealigned())</code> | | |
| zero_bit | 1 | '0' |
| <code>while (nextbits() != '0000 0000 0000 0000 0000 0001')</code> | | |
| zero_byte | 8 | '0000 0000' |
| <code>}</code> | | |

Cette fonction permet de vérifier si la position actuelle est sur une limite d'octet. Si ce n'est pas le cas, il y a remplissage par des bits à zéro. Après cette vérification, un nombre quelconque d'octets remplis de zéros peut être présent avant le code de déclenchement. Celui-ci est donc toujours sur une limite d'octet et peut être précédé d'un nombre quelconque de bits de remplissage à zéro.

5.3 Valeur réservée, valeur interdite et bit marqueur

Les termes «valeur réservée» (*reserved*) et «valeur interdite» (*forbidden*) sont utilisés pour décrire certaines valeurs de plusieurs champs du flux binaire codé.

Le qualificatif «réservée» indique que la valeur correspondante pourra être utilisée dans les futures extensions définies par l'UIT-T | ISO/CEI.

Le qualificatif «interdite» indique que la valeur correspondante ne doit jamais être utilisée (généralement afin d'éviter d'émuler des codes de déclenchement).

Le terme «marker_bit» (bit marqueur) indique un entier codé sur un bit dans lequel la valeur zéro est interdite (ce fanion doit donc avoir la valeur '1'). Ces bits marqueurs sont introduits à plusieurs endroits de la syntaxe pour éviter une émulation de code de déclenchement.

5.4 Précision arithmétique

De façon à réduire les divergences entre applications de la présente Spécification, les règles suivantes sont spécifiées concernant les opérations arithmétiques.

- a) si la précision arithmétique n'est pas spécifiée, comme dans le calcul de la transformée IDCT, la précision doit être suffisante pour que des erreurs significatives ne se produisent pas dans les valeurs entières finales;
- b) si des plages de valeurs sont indiquées avec deux points, le point final est inclus s'il y a une parenthèse et est exclu si le caractère < (inférieur à) ou > (supérieur à) est utilisé. Par exemple, l'expression [a .. b> signifie de *a* à *b*, y compris *a* mais non compris *b*.

6 Syntaxe et sémantique du flux binaire de données vidéo

6.1 Structure des données vidéo codées

Les données vidéo codées forment un ensemble ordonné de flux binaires vidéo, appelés couches. S'il n'y a qu'une seule couche, les données vidéo codées sont appelées flux binaire de données vidéo non échelonnables. S'il y a au moins deux couches, les données vidéo codées sont appelées hiérarchie échelonnée.

La première couche (de l'ensemble ordonné) est appelée couche de base et peut toujours être décodée indépendamment. On trouvera aux paragraphes 7.1 à 7.6 et au paragraphe 7.12 une description du processus de décodage pour la couche de base, sauf dans le cas d'une subdivision des données, décrite au paragraphe 7.10.

D'autres couches sont dites couches d'amélioration et ne peuvent être décodées que de concert avec toutes les couches inférieures (qui les précèdent dans l'ensemble ordonné), en commençant par la couche de base. On trouvera aux paragraphes 7.7 à 7.11 une description du processus de codage pour la hiérarchie échelonnée.

La Rec. UIT-T 222.0 | ISO/CEI 13818-1 décrit la manière dont on peut multiplexer plusieurs couches.

La couche de base d'une hiérarchie échelonnée peut être conforme à la présente Spécification ou à d'autres normes, telles que ISO/CEI 11172-2. Pour plus de détail, voir 7.7 à 7.11. Les couches d'amélioration doivent être conformes à la présente Spécification.

Dans tous les cas sauf la subdivision de données, la couche de base ne contient pas la structure d'extension à l'échelonnabilité des séquences [*sequence_scalable_extension()*]. Les couches d'amélioration doivent toujours contenir cette structure.

En général, le flux binaire de données vidéo peut être considéré comme une hiérarchie syntaxique dont les structures possèdent une ou plusieurs structures subordonnées. Par exemple, la structure «*picture_data()*» contient une ou plusieurs structures de type «*slice()*» qui, à leur tour, contiennent une ou plusieurs structures de type «*macroblock()*».

Cette hiérarchie est très proche de celle qui a été utilisée dans ISO/CEI 11172-2.

6.1.1 Séquence vidéo

La séquence vidéo est la structure syntaxique de niveau le plus élevé du flux binaire de données vidéo.

Une séquence vidéo commence par un en-tête de séquence qui peut, facultativement, être suivi d'un en-tête de groupe d'images puis d'une ou plusieurs images bi-trames codées. L'ordre de celles-ci dans le flux binaire de données vidéo codées est celui dans lequel le décodeur les traitera mais n'est pas nécessairement l'ordre correct pour l'affichage. Une

séquence vidéo se termine par un code de fin de séquence (*sequence_end_code*). A divers points de la séquence vidéo, une image codée particulière peut être précédée soit par un en-tête de répétition de séquence soit par un en-tête de groupe d'images ou par l'un et l'autre. (Si à la fois un en-tête de répétition de séquence et un en-tête de groupe d'images précèdent immédiatement une image particulière, l'en-tête de groupe d'images doit suivre l'en-tête de répétition de séquence.)

6.1.1.1 Séquences à balayage progressif et à balayage entrelacé

La présente Spécification traite du codage de séquences à balayage progressif aussi bien qu'entrelacé.

A la sortie du processus de décodage, les séquences à balayage entrelacé se présentent sous la forme d'une série de monotrames reconstruites, séparées dans le temps par une synchronisation trame. Les deux trames d'une image bi-trame peuvent être codées séparément (images à codage monotrame). En variante, ces deux trames peuvent être codées pour former ensemble une bi-trame (images à codage bi-trame). Une même séquence vidéo peut utiliser des images à codage monotrame et à codage bi-trame.

Dans une séquence à balayage progressif, chaque image de la séquence doit être de type bi-trame. A la sortie du processus de décodage, la séquence se présente sous la forme d'une série de bi-frames reconstruites, séparées dans le temps par une synchronisation image.

6.1.1.2 Bi-trame

Une image bi-trame se compose de trois matrices rectangulaires d'entiers: une matrice de luminance (Y) et deux matrices de chrominance (Cb et Cr).

On peut spécifier dans le flux binaire (ou autrement) la relation de ces composantes Y, Cb et Cr avec les signaux (analogiques) des couleurs primaires rouge, vert et bleu (E'_R , E'_G et E'_B), la chromaticité de ces primaires et les caractéristiques de transfert de l'image originale. Ces informations n'ont pas d'incidence sur le processus de décodage.

6.1.1.3 Trame

Une trame se compose d'une ligne sur deux des échantillons contenus dans les trois matrices rectangulaires des entiers représentant une image bi-trame.

Une bi-trame est l'assemblage d'une trame supérieure et d'une trame inférieure. La trame supérieure est celle qui contient la ligne située le plus haut dans chacune des trois matrices. L'autre trame est la trame inférieure.

6.1.1.4 Image

Une image reconstruite s'obtient par décodage d'une image codée, c'est-à-dire d'un en-tête d'image, des extensions facultatives qui viennent immédiatement après l'en-tête et des données d'image. Une image codée peut être de type bi-trame ou monotrame. Une image reconstruite est soit une bi-trame reconstruite (après décodage d'une image bi-trame) soit une seule trame d'une bi-trame reconstruite (après décodage d'une image monotrame).

6.1.1.4.1 Images à codage monotrame

Si des images à codage monotrame sont utilisées, elles doivent apparaître en paires (une trame supérieure suivie d'une trame inférieure) pour constituer ensemble une bi-trame codée. Les deux monotrames qui forment une bi-trame codée doivent être codées dans le flux binaire selon leur ordre d'apparition prévu à l'issue du processus de décodage.

Lorsque la première image de la bi-trame codée est de type image P monotrame, la deuxième image de la bi-trame codée doit aussi être une image P monotrame. De même, lorsque la première image de la bi-trame codée est de type image B monotrame, la deuxième image de la bi-trame codée doit aussi être une image B monotrame.

Lorsque la première image de la bi-trame codée est une image I monotrame, la deuxième image de la bi-trame doit être soit une image I monotrame ou une image P monotrame. Si la seconde image est de type P monotrame, alors certaines restrictions s'appliquent (voir 7.6.3.5).

6.1.1.4.2 Images à codage bi-trame

Lorsque l'on code des séquences à balayage entrelacé au moyen d'images à codage bi-trame, les deux trames de ces images doivent être entrelacées les unes avec les autres avant de coder la bi-trame complète comme une seule image à codage bi-trame.

6.1.1.5 Types d'images

Il existe trois types d'images, utilisant des méthodes de codage différentes:

- une **image (I) à codage intra** est codée au moyen d'informations ne provenant que d'elle-même;
- une **image (P) à codage prédictif** est codée par prédiction à compensation de mouvement à partir d'une trame de référence antérieure (d'image bi-trame ou monotrane);
- une **image (B) à codage prédictif bidirectionnel** est codée par prédiction à compensation de mouvement à partir d'une image I ou P antérieure et/ou postérieure.

6.1.1.6 En-tête de séquence

Un en-tête de séquence vidéo commence par un code d'en-tête de séquence (*sequence_header_code*) suivi d'une série d'éléments de données. Dans la présente Spécification, la structure *sequence_header()* doit être suivie de la structure *sequence_extension()* qui achemine d'autres paramètres que ceux qui sont utilisés par ISO/CEI 11172-2. Lorsque la structure *sequence_extension()* est présente, la syntaxe et les sémantèmes définis dans ISO/CEI 11172-2 ne sont pas applicables et c'est la présente Spécification qui s'applique.

Dans les en-têtes de séquence répétés, tous les éléments de données doivent avoir, à l'exception autorisée de ceux qui définissent des matrices de quantification (structures *load_intra_quantiser_matrix*, *load_non_intra_quantiser_matrix* et, facultativement, *intra_quantiser_matrix* et *non_intra_quantiser_matrix*) les mêmes valeurs que dans le premier en-tête de séquence. Les matrices de quantification peuvent être redéfinies chaque fois qu'un en-tête de séquence apparaît dans le flux binaire. [A noter que les matrices de quantification peuvent aussi être mises à jour au moyen de la structure *quant_matrix_extension()*.]

Tous les éléments de données contenus dans l'extension de séquence [*sequence_extension()*] qui suit un en-tête de séquence répété [*sequence_header()*] doivent avoir les mêmes valeurs que dans la première extension de séquence [*sequence_extension()*].

Si une structure de type *sequence_scalable_extension()* apparaît après le premier en-tête de séquence, tous les en-têtes de séquence ultérieurs doivent être suivis d'une structure *sequence_scalable_extension()* dans laquelle tous les éléments de données sont les mêmes que dans la première structure *sequence_scalable_extension()*. Réciproquement, si aucune structure de type *sequence_scalable_extension()* n'apparaît entre le premier en-tête de séquence [*sequence_header()*] et le premier en-tête d'image [*picture_header()*], cette structure ne doit pas se présenter dans le flux binaire de données vidéo.

Si une structure d'extension de séquence d'affichage [*sequence_display_extension()*] apparaît après le premier en-tête de séquence, tous les en-têtes de séquence ultérieurs doivent être suivis d'une structure *sequence_display_extension()* dans laquelle tous les éléments de données sont les mêmes que dans la première structure *sequence_display_extension()*. Réciproquement, si aucune structure de type *sequence_display_extension()* n'apparaît entre le premier en-tête de séquence et le premier en-tête d'image, cette structure ne doit pas se présenter dans le flux binaire de données vidéo.

La répétition de l'en-tête de séquence permet de répéter les éléments de données de l'en-tête de séquence initial de manière à permettre un accès aléatoire à la séquence vidéo.

Dans le flux binaire de données vidéo codées, la répétition d'un en-tête de séquence peut précéder soit une image I soit une image P mais pas une image B. Si une image entrelacée est codée sous forme de deux monotrames distinctes, la répétition d'en-tête de séquence ne doit pas précéder la deuxième de ces deux images monotrames.

Si un flux binaire est édité de façon que toutes les données précédant l'un quelconque des en-têtes de séquence répétés soient éliminées (ou, en variante, de façon qu'un accès aléatoire soit donné à cet en-tête de séquence), le flux binaire résultant doit être de structure conforme à la présente Spécification. Si la première image du flux binaire résultant est de type P, il se peut qu'elle contienne des macroblocs à codage non intra. Etant donné que les images de référence requises par le processus de décodage ne sont pas disponibles, l'image reconstruite ne sera peut-être pas définie totalement. La durée nécessaire pour rafraîchir totalement l'image complète dépendra des techniques de rafraîchissement employées.

6.1.1.7 Images de type I et en-tête de groupe d'images

Les images de type I (ou images intra) ont pour objet de faciliter l'accès aléatoire à la séquence. Les applications exigeant l'accès aléatoire, la lecture rapide en avant ou la lecture rapide en arrière peuvent utiliser assez souvent des images I.

Les images I peuvent aussi servir à des coupures de scène ou dans d'autres cas où la compensation de mouvement n'est pas utile.

L'en-tête de groupe d'images est un en-tête facultatif qui peut être utilisé immédiatement avant une bi-trame à codage intra afin d'indiquer au décodeur si les premières images B qui suivent consécutivement la bi-trame à codage intra peuvent être reconstruites correctement en cas d'accès aléatoire. En effet, si la bi-trame de référence précédente n'est pas disponible, ces images B, si elles existent, ne pourront pas être reconstruites correctement, sauf si elles ne font appel qu'à la prédiction différée. Ce point est exposé plus en détail dans l'article décrivant les structures `closed_gop` (groupe d'images fermé) et `broken_link` (rupture de lien). Un en-tête de groupe d'images contient également une information de code temporel qui n'est pas utilisée par le processus de décodage.

Dans le flux binaire de données vidéo codées, la première image bi-trame codée qui suit un en-tête de groupe d'images doit être à codage intra (I).

6.1.1.8 Format de chrominance 4:2:0

Dans ce format, les matrices des composantes Cb et Cr doivent avoir chacune la moitié des dimensions de la matrice de luminance Y, aussi bien verticalement qu'horizontalement. La matrice Y doit avoir un nombre pair de lignes et de colonnes.

NOTE – Lorsque des images entrelacées sont à codage monotrème, l'image reconstruite à partir de chacune de ces monotrèmes doit avoir une matrice Y comportant la moitié du nombre de lignes de la trame correspondante. Le nombre total de lignes dans la matrice Y d'une image complète doit donc être divisible par quatre.

Les échantillons de luminance et de chrominance sont positionnés comme indiqué sur la Figure 6-1.

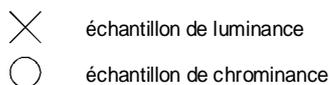
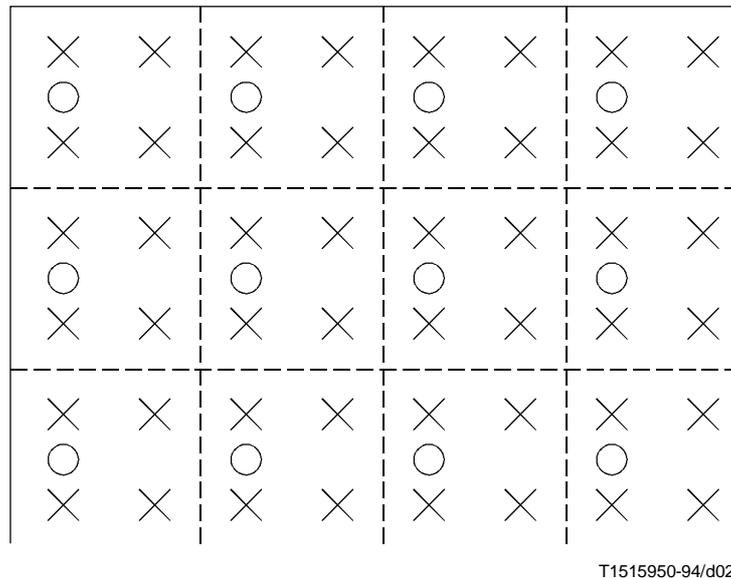


Figure 6-1 – Position des échantillons de luminance et de chrominance – Format 4:2:0

Pour spécifier plus précisément l'organisation des données, les Figures 6-2 et 6-3 montrent le positionnement (vertical) des pels (échantillons) dans une image avec un balayage entrelacé. La Figure 6-4 montre le positionnement (vertical) des échantillons dans une image avec un balayage progressif.

Dans chaque trame d'une image à balayage entrelacé, les échantillons de chrominance ne sont pas placés exactement à mi-chemin (verticalement) des échantillons de luminance de la trame, de manière que la localisation spatiale des échantillons de chrominance dans l'image bi-trame reste la même si celle-ci est à codage monotrème ou à codage bi-trème.

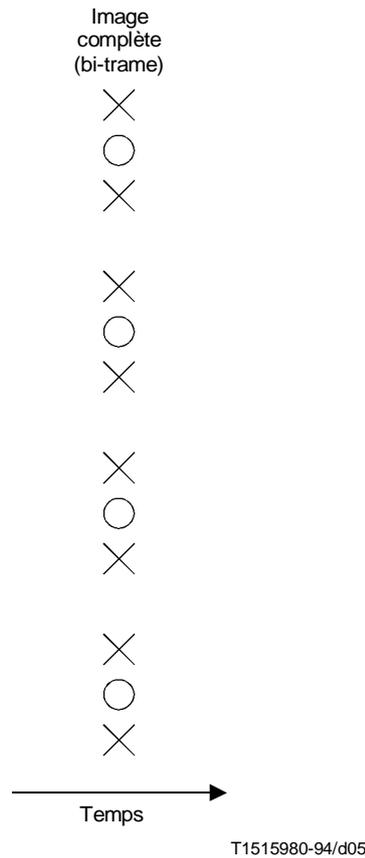


Figure 6-4 – Positions verticales et temporelles des échantillons de chrominance et de luminance d'une image à balayage progressif

6.1.1.9 Format de chrominance 4:2:2

Dans ce format, les matrices des composantes Cb et Cr doivent avoir la moitié des colonnes de la matrice de luminance Y et le même nombre de lignes. La matrice Y doit avoir un nombre pair de colonnes.

NOTE – Lorsque des images entrelacées sont à codage monotrème, l'image reconstruite à partir de chacune de ces monotrèmes doit avoir une matrice Y comportant la moitié du nombre de lignes de l'image correspondante. Le nombre total de lignes dans la matrice Y d'une bi-trème complète doit donc être divisible par deux.

Les échantillons de luminance et de chrominance sont positionnés comme indiqué sur la Figure 6-5.

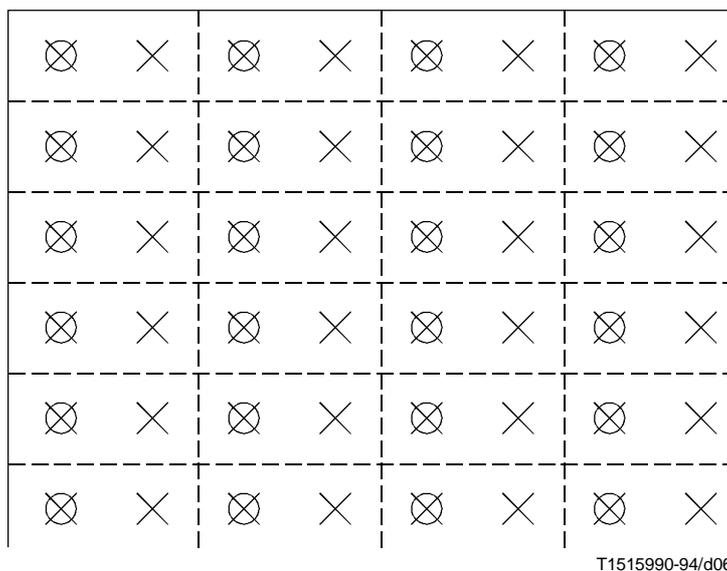
Pour clarifier l'organisation des données, la Figure 6-6 montre le positionnement (vertical) des échantillons lorsque l'image est séparée en deux trèmes.

6.1.1.10 Format de chrominance 4:4:4

Dans ce format, les matrices des composantes Cb et Cr doivent avoir le même nombre de lignes et de colonnes que la matrice de la composante Y.

NOTE – Lorsque des images entrelacées sont à codage monotrème, l'image reconstruite à partir de chacune de ces monotrèmes doit avoir une matrice Y comportant la moitié du nombre de lignes de l'image correspondante. Le nombre total de lignes dans la matrice Y d'une bi-trème complète doit donc être divisible par deux.

Les échantillons de luminance et de chrominance sont positionnés comme indiqué sur les Figures 6-6 et 6-7.



échantillon de luminance
 échantillon de chrominance

Figure 6-5 – Position des échantillons de luminance et de chrominance – Format 4:2:2

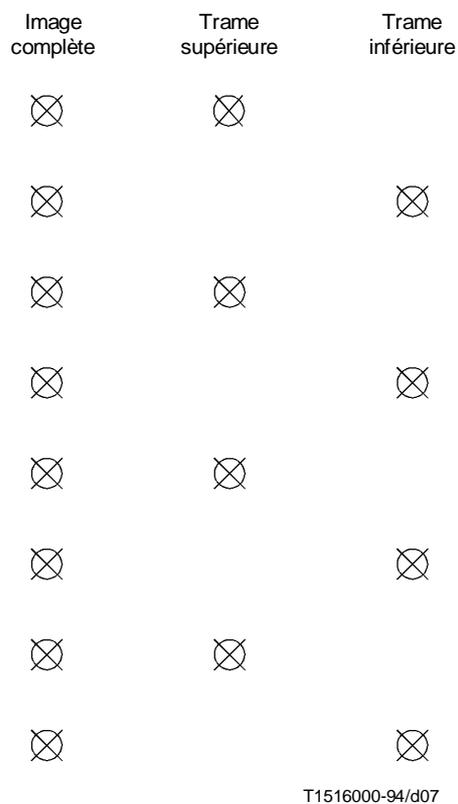
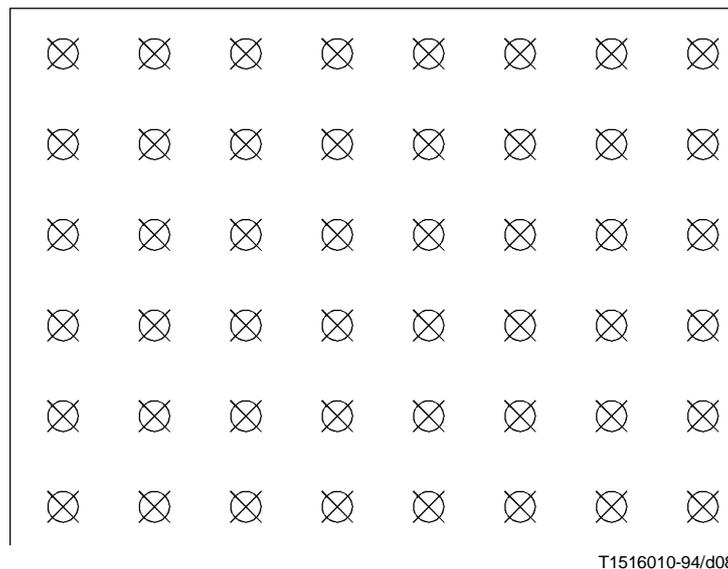


Figure 6-6 – Positions verticales des échantillons – Formats 4:2:2 et 4:4:4



X échantillon de luminance
 ○ échantillon de chrominance

Figure 6-7 – Position des échantillons de luminance et de chrominance – Format 4:4:4

6.1.1.11 Réordonnement d'une image bi-trame (ou image complète)

Lorsque la séquence contient des images codées de type B, le nombre d'images codées consécutives de type B est variable et non limité. La première image codée à la suite d'un en-tête de séquence ne doit pas être une image de type B.

Une séquence peut ne contenir aucune image bi-trame à codage P. Une séquence peut aussi ne contenir aucune bi-trame à codage I, auquel cas quelques précautions sont à prendre au début de la séquence et à l'intérieur de celle-ci, afin d'assurer aussi bien l'accès aléatoire que la reprise sur erreur.

L'ordre des bi-frames codées dans le flux binaire de données vidéo codées, également appelé ordre de codage, est l'ordre dans lequel le décodeur les traite. L'ordre des images bi-frames reconstruites à l'issue du processus de décodage, également appelé ordre d'affichage, n'est pas toujours identique à l'ordre de codage; aussi le présent paragraphe définit-il les règles applicables au réordonnement des images bi-frames qui doit se produire dans le cadre du processus de décodage.

Si la séquence ne contient aucune image bi-trame à codage B, l'ordre de codage est identique à l'ordre d'affichage. Tel est toujours le cas en particulier lorsque le fanion `low_delay` est mis à la valeur '1'.

Si la séquence contient des images bi-frames à codage B, le réordonnement s'effectue selon les règles énoncées ci-dessous:

- Si l'image bi-trame actuelle dans l'ordre de codage est une image bi-trame à codage B, l'image décodée est l'image reconstruite à partir cette image bi-trame à codage B.
- Si l'image bi-trame actuelle dans l'ordre de codage est une image à codage I ou à codage P, l'image décodée est l'image reconstruite à partir de la précédente image bi-trame à codage I ou à codage P, s'il en existe une. S'il n'y en a pas, au début de la séquence, alors aucune image n'est décodée.

La bi-trame reconstruite à partir de la bi-trame finale à codage I ou P est extraite immédiatement après la bi-trame qui a été reconstruite après suppression, du tampon du vérificateur VBV, de la dernière bi-trame codée de la séquence.

On trouvera ci-dessous un exemple d'images bi-trames examinées à partir du début d'une séquence vidéo. Dans cet exemple, il y a deux bi-trames à codage B entre des bi-trames à codage P successives; il y a également deux bi-trames à codage B entre des bi-trames à codage I et à codage P successives; et toutes les images sont à double trame. La bi-trame '1I' sert à former une prédiction pour la bi-trame '4P'. Les bi-trames '4P' et '1I' servent toutes les deux à former des prédictions pour les bi-trames '2B' et '3B'. L'ordre des bi-trames codées dans la séquence codée doit donc être le suivant: '1I', '4P', '2B', '3B'. Le décodeur doit cependant les afficher dans l'ordre '1I', '2B', '3B', '4P'.

A l'entrée du codeur:

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| I | B | B | P | B | B | P | B | B | I | B | B | P |

A la sortie du codeur, dans le flux binaire codé, et à l'entrée du décodeur:

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|----|---|---|----|----|----|
| 1 | 4 | 2 | 3 | 7 | 5 | 6 | 10 | 8 | 9 | 13 | 11 | 12 |
| I | P | B | B | P | B | B | I | B | B | P | B | B |

A la sortie du décodeur:

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
|---|---|---|---|---|---|---|---|---|----|----|----|----|

6.1.2 Tranche

Une **tranche** est une série d'un certain nombre de macroblocs consécutifs. Les premier et dernier macroblocs d'une tranche ne doivent pas être sautés. Chaque tranche doit contenir au moins un macrobloc. Les tranches ne doivent pas se chevaucher. La position des tranches peut se modifier d'une image à une autre.

Les premier et dernier macroblocs d'une tranche doivent appartenir à la même rangée horizontale de macroblocs.

Les tranches doivent apparaître dans le flux binaire selon l'ordre de leur apparition, à partir du coin supérieur gauche de l'image et en progressant par balayage de gauche à droite et de haut en bas (comme indiqué par l'ordre alphabétique sur les Figures 6-8 et 6-9.

6.1.2.1 Structure générale d'une tranche

Dans le cas le plus général, il n'est pas nécessaire que les tranches recouvrent toute l'image. La Figure 6-8 montre ce cas. Les zones qui ne sont pas incluses dans une tranche ne sont pas codées et aucune information (dans l'image en cause) n'est codée pour ces zones.

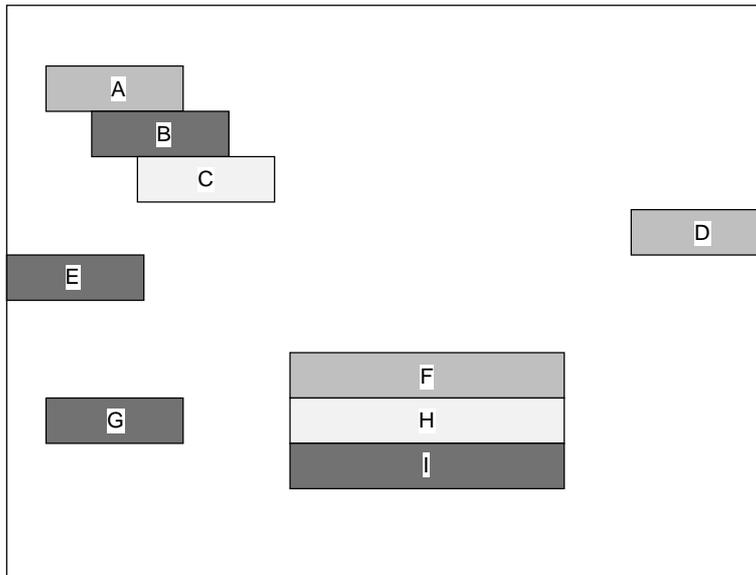
Si les tranches ne recouvrent pas toute l'image et que celle-ci soit ultérieurement utilisée pour former des prédictions, ces dernières ne doivent se fonder que sur les zones de l'image qui étaient délimitées par des tranches. C'est au codeur de veiller au respect de ces conditions.

La présente Spécification ne définit pas comment un décodeur doit agir dans les régions comprises entre les tranches.

6.1.2.2 Structure de tranche restreinte

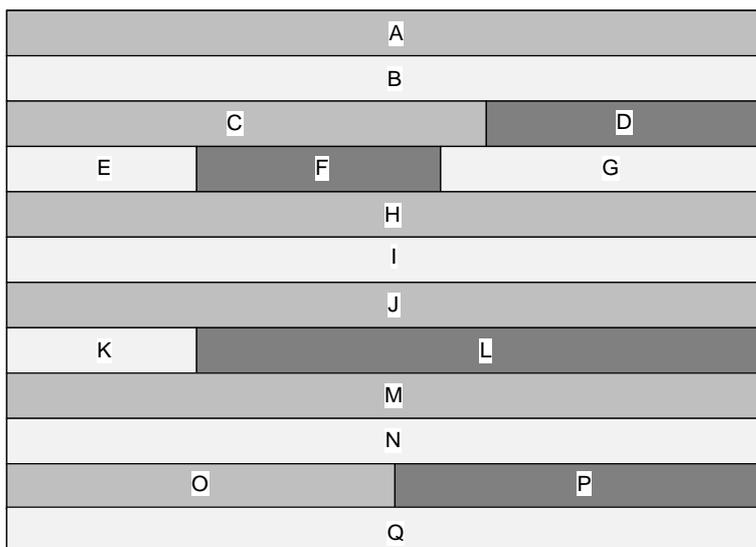
A certains niveaux de profils définis, il faut utiliser la structure de tranche restreinte qui est illustrée à la Figure 6-9. Dans ce cas, chaque macrobloc de l'image doit être inclus dans une tranche.

Lorsqu'un certain niveau de profil défini nécessite que la structure de tranche soit conforme aux restrictions précisées dans ce paragraphe, le terme «structure de tranche restreinte» peut être utilisé.



T1516020-94/d09

Figure 6-8 – Structure la plus courante d'une tranche



T1516030-94/d10

Figure 6-9 – Structure de tranche restreinte

6.1.3 Macrobloc

Un **macrobloc** contient une section de la composante de luminance et les composantes de chrominance de l'espace correspondant. Le terme «macrobloc» peut se rapporter soit aux données de source et aux données décodées, soit aux éléments de données codées correspondants. Un macrobloc est dit sauté si aucune information n'est transmise pour lui (voir 7.6.6). Un macrobloc peut avoir un des trois formats de chrominance suivants: 4:2:0, 4:2:2 ou 4:4:4. L'ordre des blocs dans les macroblocs doit être différent selon chaque format de chrominance, comme illustré ci-dessous.

Un macrobloc en format 4:2:0 se compose de 6 blocs: 4 blocs de luminance Y, 1 bloc de composante Cb et 1 bloc de composante Cr. La Figure 6-10 montre l'ordre de ces blocs.

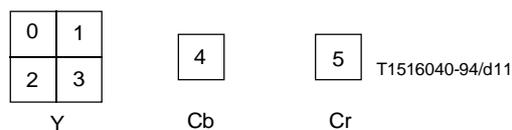


Figure 6-10 – Structure de macrobloc en format 4:2:0

Un macrobloc en format 4:2:2 se compose de 8 blocs: 4 blocs de luminance Y, 2 blocs de composante Cb et 2 blocs de composante Cr. La Figure 6-11 montre l'ordre de ces blocs.

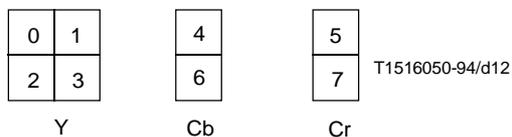


Figure 6-11 – Structure de macrobloc en format 4:2:2

Un macrobloc en format 4:4:4 se compose de 12 blocs: 4 blocs de luminance Y, 4 blocs de composante Cb et 4 blocs de composante Cr. La Figure 6-12 montre l'ordre de ces blocs.

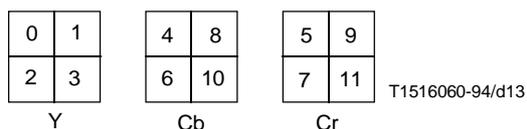


Figure 6-12 – Structure de macrobloc en format 4:4:4

Dans les images bi-trames, pour lesquelles on peut utiliser le codage DCT pour former des images aussi bien à deux trames qu'à une seule trame, l'organisation interne du macrobloc est différente selon le type d'image:

- dans le cas du codage DCT d'images à deux trames, chaque bloc doit être composé de lignes issues alternativement de l'une et de l'autre trame (voir la Figure 6-13);
- dans le cas du codage DCT d'images à une seule trame, chaque bloc doit être composé de lignes issues d'une seule des deux trames (voir la Figure 6-14).

Dans le cas des blocs de chrominance, la structure dépend du format de chrominance utilisé. Pour les formats 4:2:2 et 4:4:4 (où le macrobloc contient deux rangées de blocs de chrominance), ces derniers sont traités exactement comme les blocs de luminance. Dans le format 4:2:0 cependant, les blocs de chrominance doivent toujours être organisés en images de structure bi-trame avant le codage DCT. Il convient toutefois de noter que des prédictions fondées sur les trames peuvent être faites pour ces blocs, à condition de les appliquer (après filtrage par semi-échantillonnage) à des zones de 8 × 4 éléments.

Dans le cas d'images monotrames, chacune ne contient que les lignes issues d'une des deux trames. Dans ce cas, chaque bloc se compose de lignes extraites de la succession des lignes de l'image, comme indiqué sur la Figure 6-13.

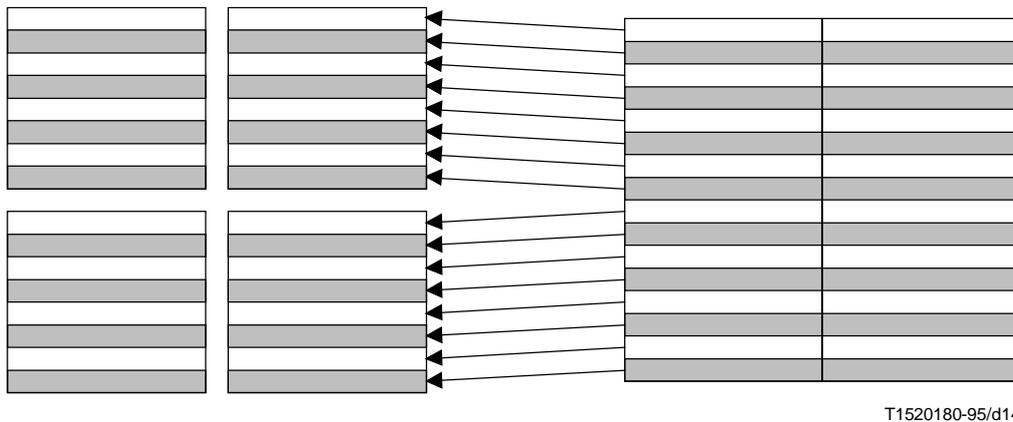


Figure 6-13 – Structure d'un macrobloc de luminance en codage DCT d'images à deux trames

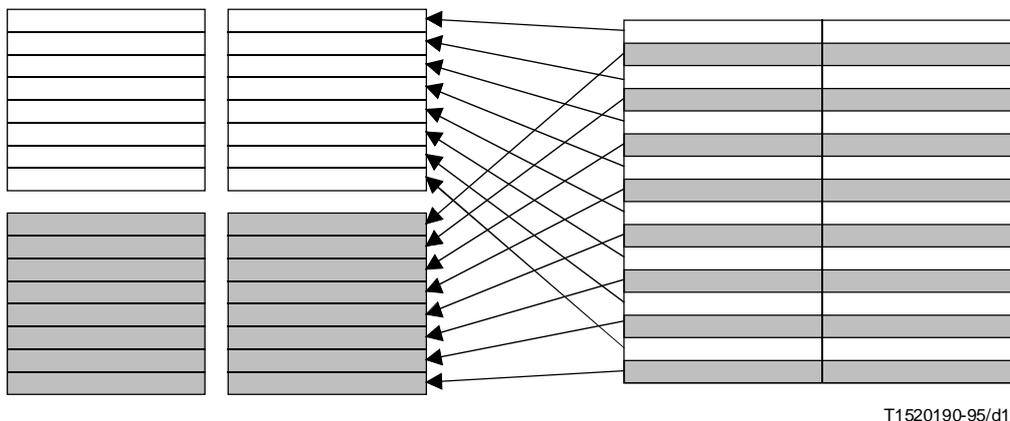


Figure 6-14 – Structure d'un macrobloc de luminance en codage DCT d'images à une trame

6.1.4 Bloc

Le terme «**bloc**» peut se rapporter soit aux données d'origine et aux données reconstruites soit aux coefficients DCT ou aux éléments de données codées correspondants.

Lorsqu'un «**bloc**» se rapporte aux données d'origine et reconstruites, il reprend une section carrée d'une composante de luminance ou de chrominance, avec le même nombre de lignes et d'échantillons. Chaque bloc comporte 8 lignes de 8 échantillons.

6.2 Syntaxe du flux binaire de données vidéo codées

6.2.1 Codes de déclenchement

Les codes de déclenchement sont des séquences binaires spécifiques qui n'apparaissent pas autrement dans le flux vidéo.

Chaque code de déclenchement se compose d'un préfixe de code de déclenchement, suivi d'une valeur de code de déclenchement. Le préfixe de code de déclenchement est une chaîne de vingt-trois éléments binaires de valeur 0, suivis d'un seul bit de valeur 1. Le préfixe de code de déclenchement se présente donc sous la forme de la chaîne binaire suivante: '0000 0000 0000 0000 0000 0001'.

La valeur du code de déclenchement est un entier codé sur huit éléments binaires, identifiant le type de code de déclenchement. La plupart des types de code de déclenchement ne possèdent qu'une seule valeur de code de déclenchement. Le code de déclenchement de tranche (structure *slice_start_code*) est toutefois représenté par de nombreuses valeurs de code de déclenchement; dans ce cas particulier, la valeur du code de déclenchement est la position verticale de la tranche (structure *slice_vertical_position*).

Tous les codes de déclenchement doivent être sur une limite d'octet. Cet alignement est réalisé par insertion de bits de valeur 0 avant le préfixe de code de déclenchement, de manière que le premier bit de ce préfixe soit l'élément le plus significatif d'un octet.

Le Tableau 6-1 définit les valeurs de code de tranche pour les codes de déclenchement utilisés dans le flux binaire de données vidéo codées.

L'utilisation des codes de déclenchement est définie dans la description syntaxique suivante, à l'exception du code d'erreur de séquence (structure *sequence_error_code*) dont l'utilisation a été réservée à une interface avec un média, pour signaler la détection d'erreurs irrécupérables.

Tableau 6-1 – Valeurs des codes de déclenchement

| Nom de la structure syntaxique | Valeur (hexadécimale) du code de déclenchement |
|--|--|
| picture_start_code | 00 |
| slice_start_code | de 01 à AF |
| reserved | B0 |
| reserved | B1 |
| user_data_start_code | B2 |
| sequence_header_code | B3 |
| sequence_error_code | B4 |
| extension_start_code | B5 |
| reserved | B6 |
| sequence_end_code | B7 |
| group_start_code | B8 |
| codes de déclenchement du système (voir Note) | de B9 à FF |
| NOTE – Les codes de déclenchement du système sont définis dans la Partie 1 de la présente Spécification. | |

6.2.2 Séquence vidéo

| video_sequence() { | Nombre de bits | Mnémonique |
|--|----------------|------------|
| next_start_code() | | |
| sequence_header() | | |
| if (nextbits() == extension_start_code) { | | |
| sequence_extension() | | |
| do { | | |
| extension_and_user_data(0) | | |
| do { | | |
| if (nextbits() == group_start_code) { | | |
| group_of_pictures_header() | | |
| extension_and_user_data(1) | | |
| } | | |
| picture_header() | | |
| picture_coding_extension() | | |
| extensions_and_user_data(2) | | |
| picture_data() | | |
| } while ((nextbits() == picture_start_code) | | |
| (nextbits() == group_start_code)) | | |
| if (nextbits() != sequence_end_code) { | | |
| sequence_header() | | |
| sequence_extension() | | |
| } | | |
| } while (nextbits() != sequence_end_code) | | |
| } else { | | |
| /* ISO/CEI 11172-2 */ | | |
| } | | |
| sequence_end_code | 32 | bslbf |
| } | | |

6.2.2.1 En-tête de séquence

| sequence_header() { | Nombre de bits | Mnémonique |
|--|----------------|------------|
| sequence_header_code | 32 | bslbf |
| horizontal_size_value | 12 | uimsbf |
| vertical_size_value | 12 | uimsbf |
| aspect_ratio_information | 4 | uimsbf |
| frame_rate_code | 4 | uimsbf |
| bit_rate_value | 18 | uimsbf |
| marker_bit | 1 | bslbf |
| vbv_buffer_size_value | 10 | uimsbf |
| constrained_parameters_flag | 1 | bslbf |
| load_intra_quantiser_matrix | 1 | uimsbf |
| if (load_intra_quantiser_matrix) | | |
| intra_quantiser_matrix[64] | 8*64 | uimsbf |
| load_non_intra_quantiser_matrix | 1 | uimsbf |
| if (load_non_intra_quantiser_matrix) | | |
| non_intra_quantiser_matrix[64] | 8*64 | uimsbf |
| next_start_code() | | |
| } | | |

6.2.2.2 Données d'extension et d'utilisateur

| extension_and_user_data(i) { | Nombre de bits | Mnémonique |
|--|----------------|------------|
| while ((nextbits()== extension_start_code) | | |
| (nextbits()== user_data_start_code)) { | | |
| if ((i != 1) && (nextbits()== extension_start_code)) | | |
| extension_data(i) | | |
| if (nextbits()== user_data_start_code) | | |
| user_data() | | |
| } | | |
| } | | |

6.2.2.2.1 Données d'extension

| | Nombre de bits | Mnémonique |
|--|----------------|------------|
| extension_data(i) { | | |
| while (nextbits()== extension_start_code) { | | |
| extension_start_code | 32 | bslbf |
| if (i == 0) { /* follows sequence_extension() */ | | |
| if (nextbits()== “Sequence Display Extension ID”) | | |
| sequence_display_extension() | | |
| else | | |
| sequence_scalable_extension() | | |
| } | | |
| /* NOTE – i never takes the value 1 because extension_data() | | |
| never follows a group_of_pictures_header() */ | | |
| if (i == 2) { /* follows picture_coding_extension() */ | | |
| if (nextbits() == “Quant Matrix Extension ID”) | | |
| quant_matrix_extension() | | |
| else if (nextbits() == “Copyright Extension ID”) | | |
| copyright_extension() | | |
| else if (nextbits() == “Picture Display Extension ID”) | | |
| picture_display_extension() | | |
| else if (nextbits() | | |
| == “Picture Spatial Scalable Extension ID”) | | |
| picture_spatial_scalable_extension() | | |
| else | | |
| picture_temporal_scalable_extension() | | |
| } | | |
| } | | |
| } | | |

6.2.2.2.2 Données utilisateur

| | Nombre de bits | Mnémonique |
|--|----------------|------------|
| user_data() { | | |
| user_data_start_code | 32 | bslbf |
| while(nextbits() != ‘0000 0000 0000 0000 0000 0001’) { | | |
| user_data | 8 | uimsbf |
| } | | |
| next_start_code() | | |
| } | | |

6.2.2.3 Extension de séquence

| sequence_extension() { | Nombre de bits | Mnémonique |
|---|----------------|------------|
| extension_start_code | 32 | bslbf |
| extension_start_code_identifieur | 4 | uimsbf |
| profile_and_level_indication | 8 | uimsbf |
| progressive_sequence | 1 | uimsbf |
| chroma_format | 2 | uimsbf |
| horizontal_size_extension | 2 | uimsbf |
| vertical_size_extension | 2 | uimsbf |
| bit_rate_extension | 12 | uimsbf |
| marker_bit | 1 | bslbf |
| vbv_buffer_size_extension | 8 | uimsbf |
| low_delay | 1 | uimsbf |
| frame_rate_extension_n | 2 | uimsbf |
| frame_rate_extension_d | 5 | uimsbf |
| next_start_code() | | |
| } | | |

6.2.2.4 Extension d'affichage de séquences

| sequence_display_extension() { | Nombre de bits | Mnémonique |
|---|----------------|------------|
| extension_start_code_identifieur | 4 | uimsbf |
| video_format | 3 | uimsbf |
| colour_description | 1 | uimsbf |
| if (colour_description) { | | |
| colour_primaries | 8 | uimsbf |
| transfer_characteristics | 8 | uimsbf |
| matrix_coefficients | 8 | uimsbf |
| } | | |
| display_horizontal_size | 14 | uimsbf |
| marker_bit | 1 | bslbf |
| display_vertical_size | 14 | uimsbf |
| next_start_code() | | |
| } | | |

6.2.2.5 Extension à l'échelonnabilité des séquences

| sequence_scalable_extension() { | Nombre de bits | Mnémonique |
|--|----------------|------------|
| extension_start_code_identifier | 4 | uimsbf |
| scalable_mode | 2 | uimsbf |
| layer_id | 4 | uimsbf |
| if (scalable_mode == "échelonnabilité spatiale") { | | |
| lower_layer_prediction_horizontal_size | 14 | uimsbf |
| marker_bit | 1 | bslbf |
| lower_layer_prediction_vertical_size | 14 | uimsbf |
| horizontal_subsampling_factor_m | 5 | uimsbf |
| horizontal_subsampling_factor_n | 5 | uimsbf |
| vertical_subsampling_factor_m | 5 | uimsbf |
| vertical_subsampling_factor_n | 5 | uimsbf |
| } | | |
| if (scalable_mode == "échelonnabilité temporelle") { | | |
| picture_mux_enable | 1 | uimsbf |
| if (picture_mux_enable) | | |
| mux_to_progressive_sequence | 1 | uimsbf |
| picture_mux_order | 3 | uimsbf |
| picture_mux_factor | 3 | uimsbf |
| } | | |
| next_start_code() | | |
| } | | |

6.2.2.6 En-tête de groupe d'images

| group_of_pictures_header() { | Nombre de bits | Mnémonique |
|------------------------------|----------------|------------|
| group_start_code | 32 | bslbf |
| time_code | 25 | bslbf |
| closed_gop | 1 | uimsbf |
| broken_link | 1 | uimsbf |
| next_start_code() | | |
| } | | |

6.2.3 En-tête d'image

| picture_header() { | Nombre de bits | Mnémonique |
|---|----------------|------------|
| picture_start_code | 32 | bslbf |
| temporal_reference | 10 | uimsbf |
| picture_coding_type | 3 | uimsbf |
| vbv_delay | 16 | uimsbf |
| if (picture_coding_type == 2 picture_coding_type == 3) { | | |
| full_pel_forward_vector | 1 | bslbf |
| forward_f_code | 3 | bslbf |
| } | | |
| if (picture_coding_type == 3) { | | |
| full_pel_backward_vector | 1 | bslbf |
| backward_f_code | 3 | bslbf |
| } | | |
| while (nextbits() == '1') { | | |
| extra_bit_picture /* avec la valeur '1' */ | 1 | uimsbf |
| extra_information_picture | 8 | uimsbf |
| } | | |
| extra_bit_picture /* avec la valeur '0' */ | 1 | uimsbf |
| next_start_code() | | |
| } | | |

6.2.3.1 Extension de codage d'image

| picture_coding_extension() { | Nombre de bits | Mnémonique |
|--|----------------|------------|
| extension_start_code | 32 | bslbf |
| extension_start_code_identifieur | 4 | uimsbf |
| f_code[0][0] /* horizontal avant */ | 4 | uimsbf |
| f_code[0][1] /* vertical avant */ | 4 | uimsbf |
| f_code[1][0] /* horizontal arrière */ | 4 | uimsbf |
| f_code[1][1] /* vertical arrière */ | 4 | uimsbf |
| intra_dc_precision | 2 | uimsbf |
| picture_structure | 2 | uimsbf |
| top_field_first | 1 | uimsbf |
| frame_pred_frame_dct | 1 | uimsbf |
| concealment_motion_vectors | 1 | uimsbf |
| q_scale_type | 1 | uimsbf |
| intra_vlc_format | 1 | uimsbf |
| alternate_scan | 1 | uimsbf |
| repeat_first_field | 1 | uimsbf |
| chroma_420_type | 1 | uimsbf |
| progressive_frame | 1 | uimsbf |
| composite_display_flag | 1 | uimsbf |
| if (composite_display_flag) { | | |
| v_axis | 1 | uimsbf |
| field_sequence | 3 | uimsbf |
| sub_carrier | 1 | uimsbf |
| burst_amplitude | 7 | uimsbf |
| sub_carrier_phase | 8 | uimsbf |
| } | | |
| next_start_code() | | |
| } | | |

6.2.3.2 Extension relative à une matrice de quantification

| quant_matrix_extension() { | Nombre de bits | Mnémonique |
|---|----------------|------------|
| extension_start_code_identifier | 4 | uimsbf |
| load_intra_quantiser_matrix | 1 | uimsbf |
| if (load_intra_quantiser_matrix) | | |
| intra_quantiser_matrix[64] | 8 * 64 | uimsbf |
| load_non_intra_quantiser_matrix | 1 | uimsbf |
| if (load_non_intra_quantiser_matrix) | | |
| non_intra_quantiser_matrix[64] | 8 * 64 | uimsbf |
| load_chroma_intra_quantiser_matrix | 1 | uimsbf |
| if (load_chroma_intra_quantiser_matrix) | | |
| chroma_intra_quantiser_matrix[64] | 8 * 64 | uimsbf |
| load_chroma_non_intra_quantiser_matrix | 1 | uimsbf |
| if (load_chroma_non_intra_quantiser_matrix) | | |
| chroma_non_intra_quantiser_matrix[64] | 8 * 64 | uimsbf |
| next_start_code() | | |
| } | | |

6.2.3.3 Extension d'affichage d'image

| picture_display_extension() { | Nombre de bits | Mnémonique |
|---|----------------|------------|
| extension_start_code_identifier | 4 | uimsbf |
| for (i = 0; i < number_of_frame_centre_offsets; i ++) { | | |
| frame_centre_horizontal_offset | 16 | simsbf |
| marker_bit | 1 | bslbf |
| frame_centre_vertical_offset | 16 | simsbf |
| marker_bit | 1 | bslbf |
| } | | |
| next_start_code() | | |
| } | | |

6.2.3.4 Extension à l'échelonnabilité temporelle d'images

| picture_temporal_scalable_extension() { | Nombre de bits | Mnémonique |
|---|----------------|------------|
| extension_start_code_identifier | 4 | uimsbf |
| reference_select_code | 2 | uimsbf |
| forward_temporal_reference | 10 | uimsbf |
| marker_bit | 1 | bslbf |
| backward_temporal_reference | 10 | uimsbf |
| next_start_code() | | |
| } | | |

6.2.3.5 Extension à l'échelonnabilité spatiale d'images

| picture_spatial_scalable_extension() { | Nombre de bits | Mnémonique |
|---|----------------|------------|
| extension_start_code_identifier | 4 | uimsbf |
| lower_layer_temporal_reference | 10 | uimsbf |
| marker_bit (valeur "1") | 1 | bslbf |
| lower_layer_horizontal_offset | 15 | simsbf |
| marker_bit (valeur "1") | 1 | bslbf |
| lower_layer_vertical_offset | 15 | simsbf |
| spatial_temporal_weight_code_table_index | 2 | uimsbf |
| lower_layer_progressive_frame | 1 | uimsbf |
| lower_layer_deinterlaced_field_select | 1 | uimsbf |
| next_start_code() | | |
| } | | |

6.2.3.6 Extension de droit d'auteur

| copyright_extension() { | Nombre de bits | Mnémonique |
|---|----------------|------------|
| extension_start_code_identifieur | 4 | uimsbf |
| copyright_flag | 1 | bslbf |
| copyright_identifieur | 8 | uimsbf |
| original_or_copy | 1 | bslbf |
| valeur réservée | 7 | uimsbf |
| marker_bit (valeur "1") | 1 | bslbf |
| copyright_number_1 | 20 | uimsbf |
| marker_bit (valeur "1") | 1 | bslbf |
| copyright_number_2 | 22 | uimsbf |
| marker_bit (valeur "1") | 1 | bslbf |
| copyright_number_3 | 22 | uimsbf |
| next_start_code() | | |
| } | | |

6.2.3.7 Données d'image

| picture_data() { | Nombre de bits | Mnémonique |
|--|----------------|------------|
| do { | | |
| slice() | | |
| } while (nextbits() == slice_start_code) | | |
| next_start_code() | | |
| } | | |

6.2.4 Tranche

| slice() { | Nombre de bits | Mnémonique |
|--|----------------|------------|
| slice_start_code | 32 | bslbf |
| if (vertical_size > 2800) | | |
| slice_vertical_position_extension | 3 | uimsbf |
| if (<sequence_scalable_extension() est présente dans le flux binaire>) { | | |
| if (scalable_mode == "subdivision des données") | | |
| priority_breakpoint | 7 | uimsbf |
| } | | |
| quantiser_scale_code | 5 | uimsbf |
| if (nextbits() == '1') { | | |
| intra_slice_flag (valeur "1") | 1 | bslbf |
| intra_slice | 1 | uimsbf |
| reserved_bits | 7 | uimsbf |
| while (nextbits() == '1') { | | |
| extra_bit_slice /* avec la valeur '1' */ | 1 | uimsbf |
| extra_information_slice | 8 | uimsbf |
| } | | |
| } | | |
| extra_bit_slice /* avec la valeur '0' */ | 1 | uimsbf |
| do { | | |
| macroblock() | | |
| } while (nextbits() != '000 0000 0000 0000 0000 0000') | | |
| next_start_code() | | |
| } | | |

6.2.5 **Macrobloc**

| | | |
|--|----------------|------------|
| macroblock() { | Nombre de bits | Mnémonique |
| while (nextbits() == '0000 0001 000') | | |
| macroblock_escape | 11 | bslbf |
| macroblock_address_increment | 1-11 | vlclbf |
| macroblock_modes() | | |
| if (macroblock_quant) | | |
| quantiser_scale_code | 5 | uimsbf |
| if (macroblock_motion_forward | | |
| (macroblock_intra && concealment_motion_vectors)) | | |
| motion_vectors(0) | | |
| if (macroblock_motion_backward) | | |
| motion_vectors(1) | | |
| if (macroblock_intra && concealment_motion_vectors) | | |
| marker_bit | 1 | bslbf |
| if (macroblock_pattern) | | |
| coded_block_pattern() | | |
| for (i = 0; i < block_count; i ++) { | | |
| block(i) | | |
| } | | |
| } | | |

6.2.5.1 Modes des macroblocs

| macroblock_modes() { | Nombre de bits | Mnémonique |
|--|----------------|------------|
| macroblock_type | 1-9 | vlclbf |
| if ((spatial_temporal_weight_code_flag == 1) && (spatial_temporal_weight_code_table_index != '00')) { | | |
| spatial_temporal_weight_code | 2 | uimsbf |
| } | | |
| if (macroblock_motion_forward macroblock_motion_backward) { | | |
| if (picture_structure == 'frame') { | | |
| if (frame_pred_frame_dct == 0) | | |
| frame_motion_type | 2 | uimsbf |
| } else { | | |
| field_motion_type | 2 | uimsbf |
| } | | |
| } | | |
| if ((picture_structure == "Frame picture") && (frame_pred_frame_dct == 0) && (macroblock_intra macroblock_pattern) { | | |
| dct_type | 1 | uimsbf |
| } | | |
| } | | |

6.2.5.2 Vecteurs de mouvement

| motion_vectors (s) { | Nombre de bits | Mnémonique |
|--|----------------|------------|
| if (motion_vector_count == 1) { | | |
| if ((mv_format == field) && (dmvc != 1)) | | |
| motion_vertical_field_select[0][s] | 1 | uimsbf |
| motion_vector(0, s) | | |
| } else { | | |
| motion_vertical_field_select[0][s] | 1 | uimsbf |
| motion_vector(0, s) | | |
| motion_vertical_field_select[1][s] | 1 | uimsbf |
| motion_vector(1, s) | | |
| } | | |
| } | | |

6.2.5.2.1 Vecteur de mouvement

| | Nombre de bits | Mnémonique |
|--|----------------|------------|
| motion_vector (r, s) { | | |
| motion_code[r][s][0] | 1-11 | vlclbf |
| if ((f_code[s][0] != 1) && (motion_code[r][s][0] != 0)) | | |
| motion_residual[r][s][0] | 1-8 | uimsbf |
| if (dmv == 1) | | |
| dmvector[0] | 1-2 | vlclbf |
| motion_code[r][s][1] | 1-11 | vlclbf |
| if ((f_code[s][1] != 1) && (motion_code[r][s][1] != 0)) | | |
| motion_residual[r][s][1] | 1-8 | uimsbf |
| if (dmv == 1) | | |
| dmvector[1] | 1-2 | vlclbf |
| } | | |

6.2.5.3 Structure de bloc codée

| | Nombre de bits | Mnémonique |
|--------------------------------|----------------|------------|
| coded_block_pattern () { | | |
| coded_block_pattern_420 | 3-9 | vlclbf |
| if (chroma_format == 4:2:2) | | |
| coded_block_pattern_1 | 2 | uimsbf |
| if (chroma_format == 4:4:4) | | |
| coded_block_pattern_2 | 6 | uimsbf |
| } | | |

6.2.6 Couche du bloc

La syntaxe détaillée pour les termes «premier coefficient DCT», «coefficient DCT suivant» et «fin de bloc» est entièrement décrite au paragraphe 7.2.

Le présent article ne développe pas parfaitement la syntaxe de la couche des blocs en cas d'utilisation du procédé de subdivision des données. Voir paragraphe 7.10.

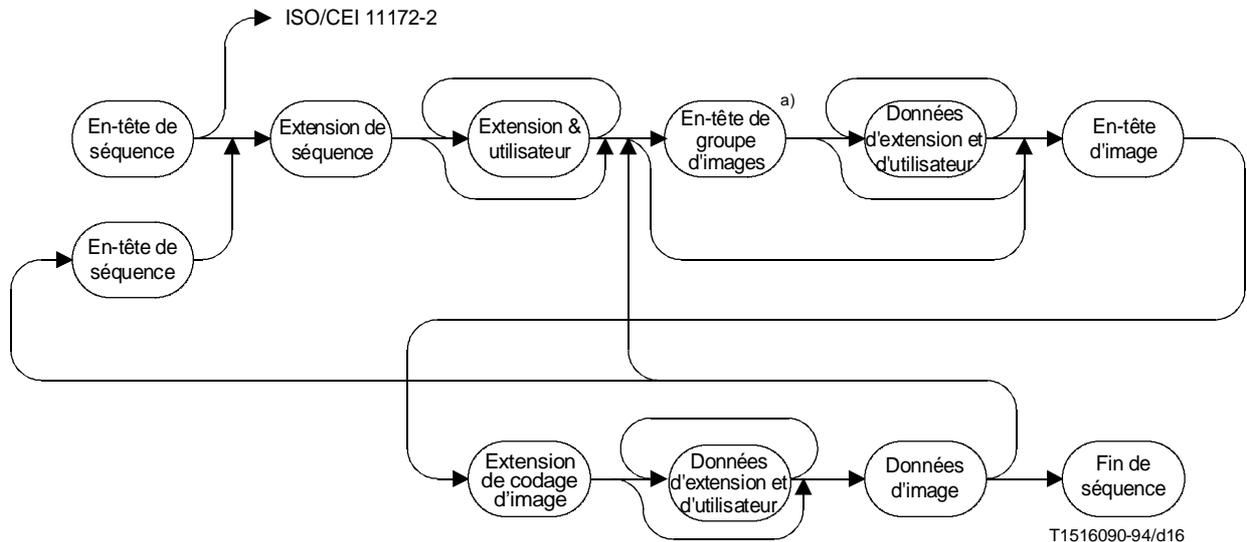
| block(i) { | Nombre de bits | Mnémonique |
|--------------------------------------|----------------|------------|
| if (pattern_code[i]) { | | |
| if (macroblock_intra) { | | |
| if (i < 4) { | | |
| dct_dc_size_luminance | 2-9 | vlclbf |
| if(dct_dc_size_luminance != 0) | | |
| dct_dc_differential | 1-11 | uimsbf |
| } else { | | |
| dct_dc_size_chrominance | 2-10 | vlclbf |
| if(dct_dc_size_chrominance != 0) | | |
| dct_dc_differential | 1-11 | uimsbf |
| } | | |
| } else { | | |
| First DCT coefficient | 2-24 | |
| } | | |
| while (nextbits() != End of block) | | |
| Subsequent DCT coefficients | 3-24 | |
| End of block | 2 or 4 | vlclbf |
| } | | |
| } | | |

6.3 Sémantique du flux binaire de données vidéo codées

6.3.1 Règles sémantiques pour les structures syntaxiques supérieures

Le présent paragraphe précise les règles qui régissent la manière dont les éléments syntaxiques de niveau supérieur peuvent être combinés pour produire un flux binaire conforme. Les autres paragraphes préciseront la portée sémantique de tous les champs du flux binaire de données vidéo.

La Figure 6-15 montre la structure de haut niveau du flux binaire de données vidéo codées.



^{a)} Après un en-tête de groupe d'images (GOP), la première image doit être de type I.

Figure 6-15 – Organisation du flux binaire à haut niveau

Les règles sémantiques suivantes sont applicables:

- Si le premier en-tête de séquence [sequence_header()] d'une séquence n'est pas suivi d'une structure d'extension de séquence [sequence_extension()], le flux doit être conforme à ISO/CEI 11172-2, sans référence à la présente Spécification.
- Si le premier en-tête de séquence [sequence_header()] d'une séquence est suivi d'une structure d'extension de séquence [sequence_extension()], toutes les apparitions ultérieures d'un en-tête de séquence doivent être immédiatement suivies d'une extension de séquence.
- Une extension de séquence [sequence_extension()] ne doit apparaître qu'à la suite d'une structure d'en-tête de séquence [sequence_header()].
- Après un en-tête de séquence [sequence_header()], il doit y avoir au moins une image codée avant un code de répétition d'en-tête de séquence ou de fin de séquence. Cela implique que l'extension de séquence ne précède pas immédiatement un code de fin de séquence.
- Si une extension de séquence [sequence_extension()] apparaît dans le flux binaire, chaque structure d'en-tête d'image [picture_header()] doit être immédiatement suivie d'une extension de codage d'image [picture_coding_extension()].
- Le code de fin de séquence (sequence_end_code) doit être placé à la fin du flux binaire, de telle sorte qu'à l'issue du décodage et après réordonnancement des images bi-frames, il ne doit y avoir aucune image bi-frame manquante.
- Une extension de codage d'image [picture_coding_extension()] ne doit apparaître qu'à la suite d'une structure d'en-tête d'image [picture_header()].
- La première image bi-trame codée qui suit un en-tête de groupe d'images [group_of_pictures_header()] doit être une image bi-trame à codage I.

Un certain nombre d'extensions diverses sont définies en plus des structures `sequence_extension()` et `picture_coding_extension()`. L'ensemble des extensions autorisées n'est pas le même en chaque point de la syntaxe, pour lequel des extensions sont autorisées. Le Tableau 6-2 définit pour chaque extension un identificateur de code de déclenchement d'extension (`extension_start_code_identifieur`) de quatre bits.

**Tableau 6-2 – Codes d'identification des codes de déclenchement d'extension
(`extension_start_code_identifieur` codes)**

| <code>extension_start_code_identifieur</code> | Nom du code |
|---|--|
| 0000 | valeur réservée |
| 0001 | identificateur d'extension de séquence |
| 0010 | identificateur d'extension d'affichage de séquence |
| 0011 | identificateur d'extension de la matrice de quantification |
| 0100 | identificateur d'extension de droit d'auteur |
| 0101 | identificateur d'extension à l'échelonnabilité de séquences |
| 0110 | valeur réservée |
| 0111 | identificateur d'extension d'affichage d'image |
| 1000 | identificateur d'extension de codage d'image |
| 1001 | identificateur d'extension à l'échelonnabilité spatiale d'images |
| 1010 | identificateur d'extension à l'échelonnabilité temporelle d'images |
| 1011 | valeur réservée |
| 1100 | valeur réservée |
| ... | ... |
| 1111 | valeur réservée |

A chaque point du flux binaire où des extensions sont autorisées, on peut insérer un nombre quelconque des extensions appartenant à l'ensemble admissible défini. Chaque type d'extension ne doit cependant pas apparaître plus d'une fois.

Si un décodeur détecte une extension dont l'identificateur est décrit comme étant de type «à valeur réservée» dans la présente Spécification, ce décodeur doit rejeter toutes les données suivantes jusqu'au prochain code de déclenchement. Cette prescription permettra de définir ultérieurement des extensions compatibles avec la présente Spécification.

6.3.2 Séquence vidéo

code de fin de séquence (`sequence_end_code`) – Le code de fin de séquence est la chaîne binaire '000001B7' en notation hexadécimale. Il termine la séquence vidéo.

6.3.3 En-tête de séquence

code d'en-tête de séquence (`sequence_header_code`) – Chaîne binaire égale à '000001B3' en notation hexadécimale. Elle identifie le début d'un en-tête de séquence.

valeur de la dimension horizontale (`horizontal_size_value`) – Ce mot forme les 12 bits les moins significatifs de la structure `horizontal_size`.

valeur de la dimension verticale (`vertical_size_value`) – Ce mot forme les 12 bits les moins significatifs de la structure `vertical_size`.

dimension horizontale (`horizontal_size`) – Cette structure se présente sous la forme d'un entier non signé de 14 éléments binaires, dont les 12 bits de poids faible sont définis dans la structure `horizontal_size_value` et dont les 2 bits de poids fort sont définis dans la structure `horizontal_size_extension`. La dimension horizontale correspond à la largeur de la partie affichable de la composante de luminance en pels. La largeur de la composante de luminance codée d'images découpées en macroblocs (structure `mb_width`) est égale au résultat de l'opération $(horizontal_size + 15)/16$. La partie affichable est alignée sur le bord gauche des images codées.

Pour éviter toute émulation d'un code de déclenchement, la valeur de dimension horizontale doit être différente de zéro, ce qui exclut les multiples de 4096.

dimension verticale (vertical_size) – Cette structure se présente sous la forme d'un entier arithmétique de 14 éléments binaires, dont les 12 bits les moins significatifs sont définis par la structure vertical_size_value et dont les 2 bits les plus significatifs sont définis par la structure vertical_size_extension. La dimension verticale indique, en nombre de lignes, la hauteur de la partie affichable de la composante de luminance de l'image complète (bi-trame) en pels.

Si la structure progressive_sequence a la valeur '1', la hauteur en macroblocs de la composante de luminance codée des images bi-trames, mb_height, est égale à (vertical_size + 15)/16.

Si la structure progressive_sequence a la valeur '0', la hauteur en macroblocs de la composante de luminance codée des images bi-trames, mb_height, est égale à 2*((vertical_size + 31)/32). La hauteur en macroblocs de la composante de luminance codée des images monotrames, mb_height, est égale à ((vertical_size + 31)/32).

La partie affichable est alignée sur le bord supérieur des images codées.

Pour éviter toute émulation parasite d'un code de déclenchement, la valeur de dimension verticale ne doit pas être nulle, ce qui exclut les multiples de 4096 comme valeurs de la structure vertical_size.

information de format d'image (aspect_ratio_information) – Cette structure se présente sous la forme d'un entier de quatre éléments binaires, défini dans le Tableau 6-3.

Tableau 6-3 – Structure aspect_ratio_information

| aspect_ratio_information | SAR | DAR |
|--------------------------|---------------------------|------------------|
| 0000 | Valeur interdite | Valeur interdite |
| 0001 | 1,0 (échantillons carrés) | – |
| 0010 | – | 3 ÷ 4 |
| 0011 | – | 9 ÷ 16 |
| 0100 | – | 1 ÷ 2,21 |
| 0101 | – | Valeur réservée |
| ... | | ... |
| 1111 | – | Valeur réservée |

L'information de format d'image spécifie qu'il s'agit soit du format d'échantillon (SAR) (*sample aspect ratio*) de l'image bi-trame reconstruite qui est 1,0 (échantillons carrés) ou, en variante, du format d'affichage (DAR) (*display aspect ratio*).

- Si la structure sequence_display_extension() n'est pas présente, il est prévu que toute l'image bi-trame reconstruite soit appliquée sur toute la région active de l'écran d'affichage. On peut calculer comme suit le format d'échantillon:

$$SAR = DAR \times \frac{\text{horizontal_size}}{\text{vertical_size}}$$

NOTE – Dans ce cas, les structures horizontal_size et vertical_size sont contraintes par le format SAR de l'image d'origine et par le format DAR choisi.

- Si la structure sequence_display_extension() est présente, on peut calculer comme suit le format d'échantillon:

$$SAR = DAR \times \frac{\text{display_horizontal_size}}{\text{display_vertical_size}}$$

code de fréquence image (frame_rate_code) – Cette structure se présente sous la forme d'un entier de quatre bits. Elle est utilisée pour définir une valeur de fréquence image comme indiqué dans le Tableau 6-4. La fréquence image (frame_rate) peut être déduite comme suit des structures ci-après: frame_rate_value, frame_rate_extension_n et frame_rate_extension_d:

$$\text{frame_rate} = \text{frame_rate_value} * (\text{frame_rate_extension_n} + 1) \div (\text{frame_rate_extension_d} + 1)$$

Lorsque le Tableau 6-4 contient une entrée correspondant exactement au débit binaire, les structures frame_rate_extension_n et frame_rate_extension_d doivent avoir une valeur nulle. Les structures frame_rate_extension_n + 1 et frame_rate_extension_d + 1 ne doivent pas avoir de diviseur commun supérieur à 1.

Tableau 6-4 – Structure frame_rate_value

| frame_rate_code | frame_rate_value |
|-----------------|---------------------------|
| 0000 | Valeur interdite |
| 0001 | 24 000 ÷ 1001 (23,976...) |
| 0010 | 24 |
| 0011 | 25 |
| 0100 | 30 000 ÷ 1001 (29,97...) |
| 0101 | 30 |
| 0110 | 50 |
| 0111 | 60 000 ÷ 1001 (59,94...) |
| 1000 | 60 |
| 1001 | Valeur réservée |
| ... | ... |
| 1111 | Valeur réservée |

Si la structure progressive_sequence a la valeur '1', la période entre deux images bi-frames successives à la sortie du processus de décodage est l'inverse de la fréquence d'image (frame_rate). Voir la Figure 7-18.

Si la structure progressive_sequence a la valeur '0', la période entre deux trames successives à la sortie du processus de décodage est la moitié de l'inverse de la fréquence d'image (frame_rate). Voir la Figure 7-20.

La fréquence d'image signalée dans la couche d'amélioration en mode d'échelonnabilité temporelle est la fréquence d'image combinée qui résulte de l'opération de remultiplexage temporel, si la structure picture_mux_enable est mise à la valeur '1' dans la structure sequence_scalable_extension().

valeur du débit binaire (bit_rate_value) – Valeur représentée par les 18 bits inférieurs de la structure bit_rate.

débit binaire (bit_rate) – Entier codé sur 30 bits. Les 18 bits inférieurs de cet entier sont représentés par la structure bit_rate_value et les 12 bits supérieurs par la structure bit_rate_extension. Cet entier de 30 bits spécifie le débit binaire du flux mesuré en unités de 400 bits/seconde, arrondi par excès. La valeur zéro est interdite.

Le débit binaire ainsi spécifié limite le débit de fonctionnement maximal du vérificateur VBV de mémoire vidéo, tel qu'il est défini en annexe au point C.3.

Le vérificateur VBV fonctionne en mode constant ou en mode variable, selon les valeurs codées dans la structure vbv_delay. Dans un cas comme dans l'autre, le débit binaire spécifié doit être la limite supérieure du débit auquel les données codées sont appliquées à l'entrée du vérificateur VBV.

NOTE – Etant donné que le fonctionnement en débit binaire constant n'est qu'un cas particulier du fonctionnement en débit binaire variable, il n'est pas prescrit que la valeur de la structure bit_rate corresponde au débit binaire auquel les données sont effectivement fournies. Il est cependant recommandé, en mode à débit constant, que la structure bit_rate représente le débit binaire réel.

bit marqueur (marker_bit) – Ce bit unique doit être mis à ‘1’ pour éviter l’émulation de codes de déclenchement.

valeur de capacité du vérificateur de mémoire vidéo (vbv_buffer_size_value) – Cette valeur correspond aux 10 éléments binaires les moins significatifs de la structure vbv_buffer_size.

capacité du vérificateur de mémoire vidéo (vbv_buffer_size) – Cette structure est représentée par un entier codé sur 18 éléments binaires. Les 10 éléments binaires les moins significatifs de cet entier correspondent à la structure vbv_buffer_size_value et les 8 éléments binaires les plus significatifs correspondent à la structure vbv_buffer_size_extension. Cet entier définit la capacité du vérificateur de mémoire vidéo (VBV, voir l'Annexe C) nécessaire pour décoder la séquence. Il est défini comme suit:

$$B = 16 * 1024 * vbv_buffer_size$$

où B est la capacité minimale du vérificateur VBV (en bits) requise pour décoder la séquence (voir l'Annexe C).

indicateur de flux paramétrique contraint (constrained_parameters_flag) – Cet indicateur (utilisé dans ISO/CEI 11172-2) n'a pas de signification dans la présente Spécification et doit avoir la valeur ‘0’.

charger une matrice de quantificateur intra (load_intra_quantiser_matrix) – Voir 6.3.11 «Extension relative à une matrice de quantification».

matrice de quantificateur intra (intra_quantiser_matrix) – Voir 6.3.11 «Extension relative à une matrice de quantification».

charger une matrice de quantificateur non intra (load_non_intra_quantiser_matrix) – Voir 6.3.11 «Extension relative à une matrice de quantification».

matrice de quantificateur non intra (non_intra_quantiser_matrix) – Voir 6.3.11 «Extension relative à une matrice de quantification».

6.3.4 Extensions et données d'utilisateur

code de déclenchement d'une extension (extension_start_code) – Cette structure correspond à la chaîne binaire ‘000001B5’ en notation hexadécimale. Elle signale le début d'extensions définies uniquement dans cette Spécification.

6.3.4.1 Données d'utilisateur

code de déclenchement de données d'utilisateur (user_data_start_code) – Cette structure correspond à la chaîne binaire ‘000001B2’ en notation hexadécimale. Elle signale le début de données utilisateur. Celles-ci sont transmises jusqu'à réception d'un autre code de déclenchement.

données utilisateur (user_data) – Entier codé sur un nombre quelconque d'octets consécutifs. Cette structure est définie par les utilisateurs pour leurs applications spécifiques. La série d'octets consécutifs de données utilisateur ne doit pas contenir de zéros binaires en chaîne de longueur égale ou supérieure à 23 éléments binaires consécutifs.

6.3.5 Extension de séquence

identificateur de code de déclenchement d'extension (extension_start_code_identifieur) – Cette structure est représentée par un entier de 4 éléments binaires qui identifie le type d'extension. Voir le Tableau 6-2.

indication du profil et du niveau (profile_and_level_indication) – Cette structure est représentée par un entier de 8 éléments binaires qui identifie le profil et le niveau de codage. La signification de ces éléments binaires est exposée au paragraphe 8.

NOTE – En hiérarchie échelonnée, les flux binaires de chaque couche peuvent donner à la structure profile_and_level_indication une valeur différente de ce qui est spécifié au paragraphe 8.

séquence à balayage progressif (progressive_sequence) – Lorsque cette structure est mise à la valeur ‘1’, la séquence vidéo codée ne contient que des images bi-frames à balayage progressif. Lorsque cette structure est mise à la valeur ‘0’, la séquence vidéo codée peut contenir aussi bien des images bi-frames que des images monoframes et les bi-frames peuvent être à balayage progressif ou entrelacé.

format de chrominance (chroma_format) – Cette structure est représentée par un entier de deux éléments binaires indiquant le format de chrominance tel que défini dans le Tableau 6-5.

Tableau 6-5 – Signification de la structure chroma_format

| chroma_format | Signification |
|---------------|-----------------|
| 00 | Valeur réservée |
| 01 | 4:2:0 |
| 10 | 4:2:2 |
| 11 | 4:4:4 |

extension de dimension horizontale (horizontal_size_extension) – Ce mot forme les 2 bits les plus significatifs de la structure horizontal_size.

extension de dimension verticale (vertical_size_extension) – Ce mot forme les 2 bits les plus significatifs de la structure vertical_size.

extension de débit binaire (bit_rate_extension) – Ce mot forme les 12 bits les plus significatifs de la structure bit_rate.

extension de capacité du vérificateur de mémoire vidéo (vbv_buffer_size_extension) – Ce mot forme les 8 bits les plus significatifs de la structure vbv_buffer_size.

faible délai (ou faible retard) (low_delay) – Ce fanion, lorsqu'il est mis à '1', indique que la séquence ne contient pas d'images B, que la description du vérificateur VBV n'indique pas le délai de réordonnement des images bi-frames et que le flux binaire peut contenir des images complètes, c'est-à-dire que C.7 relatif aux vérificateurs VBV est applicable.

Ce fanion lorsqu'il est mis à '0' indique que la séquence peut contenir des images B, que le délai de réordonnement des images bi-frames est présent dans la description du vérificateur VBV et que le flux binaire ne doit pas contenir d'images complètes, c'est-à-dire que C7 relatif aux vérificateurs VBV n'est pas applicable.

Ce fanion n'est pas utilisé au cours du processus de décodage et peut donc être ignoré par les décodeurs; il est cependant nécessaire de définir et de vérifier la conformité des flux binaires à faible délai.

numérateur d'extension de fréquence d'image (frame_rate_extension_n) – Entier codé sur 2 bits servant à déterminer la fréquence d'image. Voir code de fréquence d'image (frame_rate_code).

dénominateur d'extension de fréquence d'image (frame_rate_extension_d) – Entier codé sur 5 bits servant à déterminer la fréquence d'image. Voir code de fréquence d'image (frame_rate_code).

6.3.6 Extension d'affichage de séquences

La présente Spécification ne définit pas le processus d'affichage. Les informations de cette extension n'ont pas d'incidence sur le processus de décodage et peuvent être négligées par les décodeurs conformes à la présente Spécification.

format vidéo (video_format) – Cette structure est représentée par un entier de 3 bits qui indique le mode de présentation des images avant leur codage selon la présente Spécification. Sa signification est définie dans le Tableau 6-6. Si l'extension d'affichage de séquences est absente du flux binaire, le format vidéo peut alors être présumé «non spécifié».

Tableau 6-6 – Signification de la structure video_format

| video_format | Signification |
|--------------|---------------------------|
| 000 | En composantes Y/C |
| 001 | PAL |
| 010 | NTSC |
| 011 | SECAM |
| 100 | MAC |
| 101 | Format vidéo non spécifié |
| 110 | Valeur réservée |
| 111 | Valeur réservée |

description chromatique (colour_description) – Fanion qui, s'il est mis à '1', indique la présence, dans le flux binaire, des structures colour primaries, transfer_characteristics et matrix_coefficients.

couleurs primaires (colour primaries) – Entier codé sur un octet qui décrit les coordonnées de chromaticité des couleurs primaires de la source. Cette structure est définie dans le Tableau 6-7.

Tableau 6-7 – Couleurs primaires

| Valeur | Primaires |
|--------|---|
| 0 | (Valeur interdite) |
| 1 | Recommandation UIT-R BT.709 primaire x y vert 0,300 0,600 bleu 0,150 0,060 rouge 0,640 0,330 blanc D65 0,3127 0,3290 |
| 2 | Système vidéo non spécifié Caractéristiques d'image inconnues |
| 3 | Valeur réservée |
| 4 | Recommandation UIT-R BT.470-2 Système M primaire x y vert 0,21 0,71 bleu 0,14 0,08 rouge 0,67 0,33 blanc C 0,310 0,316 |
| 5 | Recommandation UIT-R BT.470-2 Systèmes B, G primaire x y vert 0,29 0,60 bleu 0,15 0,06 rouge 0,64 0,33 blanc D65 0,313 0,329 |
| 6 | SMPTE 170M primaire x y vert 0,310 0,595 bleu 0,155 0,070 rouge 0,630 0,340 blanc D65 0,3127 0,3290 |
| 7 | SMPTE 240M (1987) primaire x y vert 0,310 0,595 bleu 0,155 0,070 rouge 0,630 0,340 blanc D65 0,3127 0,3291 |
| 8-255 | Valeur réservée |

Si la structure d'extension d'affichage de séquence n'est pas présente dans le flux binaire ou si la structure de description de couleur a la valeur zéro, le système considère que la chromaticité est celle qui correspond à une structure colour primaries de valeur 1.

caractéristiques de transfert (transfer_characteristics) – Cette structure codée sous forme d'entier sur 8 bits décrit la caractéristique de transfert optoélectronique de l'image d'origine. Elle est définie dans le Tableau 6-8.

Tableau 6-8 – Caractéristique de transfert

| Valeur | Caractéristique de transfert |
|--------|---|
| 0 | (Valeur interdite) |
| 1 | Recommandation UIT-R BT.709 $V = 1,099 L_c^{0,45} - 0,099$ pour $1 \geq L_c \geq 0,018$ $V = 4,500 L_c$ pour $0,018 > L_c \geq 0$ |
| 2 | Système vidéo non spécifié Caractéristiques d'image inconnues |
| 3 | Valeur réservée |
| 4 | Recommandation UIT-R BT.470-2 Système M Gamma d'affichage considéré 2,2 |
| 5 | Recommandation UIT-R BT.470-2 Systèmes B, G Gamma d'affichage considéré 2,8 |
| 6 | SMPTE 170M $V = 1,099 L_c^{0,45} - 0,099$ pour $1 \geq L_c \geq 0,018$ $V = 4,500 L_c$ pour $0,018 > L_c \geq 0$ |
| 7 | SMPTE 240M (1987) $V = 1,1115 L_c^{0,45} - 0,1115$ pour $L_c \geq 0,0228$ $V = 4,0 L_c$ pour $0,0228 > L_c$ |
| 8 | Caractéristiques de transfert linéaires soit: $V = L_c$ |
| 9-255 | Valeur réservée |

Si la structure d'extension d'affichage de séquence n'est pas présente dans le flux binaire ou si la structure de description de couleur a la valeur zéro, le système considère que les caractéristiques de transfert sont celles qui correspondent à une structure transfer_characteristics de valeur 1.

coefficients de la matrice (matrix_coefficients) – Cette structure, codée sous forme d'entier sur 8 bits, décrit les coefficients de la matrice utilisés pour calculer les signaux de luminance et de chrominance d'après les primaires (vert, bleu et rouge). Elle est définie dans le Tableau 6-9.

Tableau 6-9 – Coefficients matriciels

| Valeur | Matrice |
|--------|--|
| 0 | (Valeur interdite) |
| 1 | Recommandation UIT-R BT.709 $E'_Y = 0,7154 E'_G + 0,0721 E'_B + 0,2125 E'_R$ $E'_{PB} = -0,386 E'_G + 0,500 E'_B - 0,115 E'_R$ $E'_{PR} = -0,454 E'_G - 0,046 E'_B + 0,500 E'_R$ |
| 2 | Système vidéo non spécifié Caractéristiques d'image inconnues |
| 3 | Valeur réservée |
| 4 | FCC $E'_Y = 0,59 E'_G + 0,11 E'_B + 0,30 E'_R$ $E'_{PB} = -0,331 E'_G + 0,500 E'_B - 0,169 E'_R$ $E'_{PR} = -0,421 E'_G - 0,079 E'_B + 0,500 E'_R$ |
| 5 | Recommandation UIT-R BT.470-2 Systèmes B, G $E'_Y = 0,587 E'_G + 0,114 E'_B + 0,299 E'_R$ $E'_{PB} = -0,331 E'_G + 0,500 E'_B - 0,169 E'_R$ $E'_{PR} = -0,419 E'_G - 0,081 E'_B + 0,500 E'_R$ |
| 6 | SMPTE 170M $E'_Y = 0,587 E'_G + 0,114 E'_B + 0,299 E'_R$ $E'_{PB} = -0,331 E'_G + 0,500 E'_B - 0,169 E'_R$ $E'_{PR} = -0,419 E'_G - 0,081 E'_B + 0,500 E'_R$ |
| 7 | SMPTE 240M (1987) $E'_Y = 0,701 E'_G + 0,087 E'_B + 0,212 E'_R$ $E'_{PB} = -0,384 E'_G + 0,500 E'_B - 0,116 E'_R$ $E'_{PR} = -0,445 E'_G - 0,055 E'_B + 0,500 E'_R$ |
| 8-255 | Valeur réservée |

Dans le Tableau 6-9:

- E'_Y est un coefficient analogique pour les valeurs comprises entre 0 et 1;
- E'_{PB} et E'_{PR} sont des coefficients analogiques pour les valeurs comprises entre -0,5 et 0,5;
- E'_R , E'_G et E'_B sont des coefficients analogiques pour les valeurs comprises entre 0 et 1;
- Y, Cb et Cr ont les relations suivantes avec les coefficients E'_Y , E'_{PB} et E'_{PR} :

$$Y = (219 * E'_Y) + 16$$

$$Cb = (224 * E'_{PB}) + 128$$

$$Cr = (224 * E'_{PR}) + 128$$

NOTE – Le processus de décodage décrit dans la présente Spécification limite les valeurs d'échantillon de sortie pour Y, Cr et Cb à la plage [0:255]. Des mots de code de valeur extérieure à la plage définie par les équations ci-dessus peuvent donc apparaître de temps en temps à la sortie du processus de décodage, en particulier 0 et 255.

Si la structure d'extension d'affichage de séquence n'est pas présente dans le flux binaire ou si la structure de description de couleur a la valeur zéro, le système considère que les coefficients matriciels sont ceux qui correspondent à une structure `matrix_coefficients` de valeur 1.

dimension horizontale d'affichage (`display_horizontal_size`) – Voir `display_vertical_size`.

dimension verticale d'affichage (`display_vertical_size`) – Les deux structures `display_horizontal_size` et `display_vertical_size` définissent ensemble un rectangle que l'on peut considérer comme étant la région active de «l'affichage prévu». Si ce rectangle est plus petit que l'image bi-trame codée, on peut s'attendre que le processus d'affichage n'affiche qu'une partie de cette image. Inversement, si le rectangle d'affichage est plus grand que l'image bi-trame codée, on peut s'attendre que le processus d'affichage n'affiche les images bi-frames reconstruites que sur une partie au lieu de la totalité du dispositif d'affichage.

La structure `display_horizontal_size` doit être exprimée dans les mêmes unités que la structure `horizontal_size` (échantillons des images bi-frames codées).

La structure `display_vertical_size` doit être exprimée dans les mêmes unités que la structure `vertical_size` (lignes des images bi-frames codées).

Les structures `display_horizontal_size` et `display_vertical_size` n'ont pas d'incidence sur le processus de décodage mais peuvent être utilisées par le processus d'affichage (non normalisé dans la présente Spécification).

6.3.7 Extension à l'échelonnabilité des séquences

Si une structure `sequence_scalable_extension()` est présente dans le flux binaire à la suite d'une structure `sequence_extension()` donnée, il s'agit d'une restriction sémantique et la structure `sequence_scalable_extension()` doit venir à la suite de toutes les autres structures `sequence_extension()`. Un flux binaire est donc échelonnable ou non échelonnable. Il n'est pas possible de mélanger, dans une même séquence, un codage d'échelonnabilité et un codage de non-échelonnabilité.

mode échelonnable (`scalable_mode`) – Cette structure indique le type d'échelonnabilité utilisé dans la séquence vidéo. Si aucune structure `sequence_scalable_extension()` n'est présente dans le flux binaire, aucune échelonnabilité n'est utilisée pour cette séquence. Le mode échelonnable indique aussi les tables de type de macrobloc à utiliser. Dans le cas d'une échelonnabilité spatiale, si aucune structure `picture_spatial_scalable_extension()` n'est présente pour une image donnée, cette image doit toutefois être décodée en mode non échelonnable [c'est-à-dire comme si la structure `sequence_scalable_extension()` n'était pas présente].

Tableau 6-10 – Définition de la structure `scalable_mode`

| Structure <code>scalable_mode</code> | Signification | Structure <code>picture_spatial_scalable_extension()</code> | Tables de structures <code>macroblock_type</code> |
|---|----------------------------|---|---|
| structure <code>sequence_scalable_extension()</code> non présente | | | B-2, B-3 et B-4 |
| 00 | Subdivision des données | | B-2, B-3 et B-4 |
| 01 | Echelonnabilité spatiale | Présente | B-5, B-6 et B-7 |
| | | Non présente | B-2, B-3 et B-4 |
| 10 | Echelonnabilité SNR | | B-8 |
| 11 | Echelonnabilité temporelle | | B-2, B-3 et B-4 |

identificateur de couche (`layer_id`) – Cette structure est codée sous forme d'un entier qui identifie les couches dans une hiérarchie échelonnable. La couche de base est toujours repérée par `layer_id` = 0. La couche de base d'une hiérarchie échelonnable n'est cependant pas assortie d'une structure `sequence_scalable_extension()` ni donc d'une structure `layer_id`, sauf en cas de subdivision des données. Chaque couche successive possède, par rapport à la couche qu'elle améliore, un identificateur de couche dont la valeur est augmentée de 1.

En cas de subdivision des données, l'identificateur de couche doit avoir la valeur zéro pour la partition zéro et la valeur un pour la partition un.

dimension horizontale de l'image de couche inférieure utilisée pour la prédiction (lower_layer_prediction_horizontal_size) – Cette structure, codée sous forme d'un entier de 14 bits, indique la dimension horizontale de l'image bi-trame de couche inférieure qui est utilisée pour la prédiction. Cette structure doit contenir la valeur contenue dans la structure horizontal_size (horizontal_size_value et horizontal_size_extension) acheminée par le flux binaire de couche inférieure.

dimension verticale de l'image de couche inférieure utilisée pour la prédiction (lower_layer_prediction_vertical_size) – Cette structure, codée sous forme d'un entier de 14 bits, indique la dimension verticale de l'image bi-trame de couche inférieure qui est utilisée pour la prédiction. Cette structure doit contenir la valeur contenue dans la structure vertical_size (vertical_size_value et vertical_size_extension) acheminée par le flux binaire de couche inférieure.

facteur m du sous-échantillonnage horizontal (horizontal_subsampling_factor_m) – Cette structure affecte le processus de suréchantillonnage en mode d'échelonnabilité spatiale, comme indiqué au 7.7.2. La valeur zéro est interdite.

facteur n du sous-échantillonnage horizontal (horizontal_subsampling_factor_n) – Cette structure affecte le processus de suréchantillonnage en mode d'échelonnabilité spatiale, comme indiqué au 7.7.2. La valeur zéro est interdite.

facteur m du sous-échantillonnage vertical (vertical_subsampling_factor_m) – Cette structure affecte le processus de suréchantillonnage en mode d'échelonnabilité spatiale, comme indiqué au 7.7.2. La valeur zéro est interdite.

facteur n du sous-échantillonnage vertical (vertical_subsampling_factor_n) – Cette structure affecte le processus de suréchantillonnage en mode d'échelonnabilité spatiale, comme indiqué au 7.7.2. La valeur zéro est interdite.

activation du multiplexage d'images (picture_mux_enable) – Si cette structure a la valeur '1', les structures picture_mux_order et picture_mux_factor sont utilisées pour remultiplexer les images avant leur affichage.

séquence multiplexée en balayage progressif (mux_to_progressive_sequence) – Ce fanion, lorsqu'il est mis à '1', indique que les images décodées qui correspondent aux deux couches doivent être temporairement multiplexées pour former à l'affichage une séquence à balayage progressif. Ce fanion doit être mis à '0' lorsque le multiplexage temporel doit produire une séquence à balayage entrelacé.

ordre de multiplexage des images (picture_mux_order) – Cette structure indique le nombre d'images de couche d'amélioration qui précèdent la première image en couche de base. Elle facilite donc le remultiplexage d'images avant leur affichage car elle contient des informations permettant d'inverser le démultiplexage effectué par le codeur.

facteur de multiplexage des images (picture_mux_factor) – Cette structure indique le nombre d'images de couche d'amélioration qui sont insérées entre des images consécutives en couche de base afin de permettre un remultiplexage correct des couches de base et d'amélioration avant affichage. Elle facilite également le remultiplexage d'images avant leur affichage car elle contient des informations permettant d'inverser le démultiplexage temporel effectué par le codeur. La valeur '000' est réservée.

6.3.8 En-tête de groupe d'images

code de déclenchement de groupe d'images (group_start_code) – Structure représentée en notation hexadécimale par la chaîne binaire '000001B8'. Elle identifie le début d'un en-tête de groupe d'images.

code temporel (time_code) – Entier codé sur 25 bits qui contient les champs suivants: drop_frame_flag, time_code_hours, time_code_minutes, marker_bit, time_code_seconds et time_code_pictures comme indiqué dans le Tableau 6-11. Ces champs correspondent à ceux qui sont définis dans la Norme CEI 461, relative aux codes temporels de commande pour les magnétoscopes (voir Bibliographie, Annexe F). Ce code se rapporte à la première image qui suit l'en-tête de groupe d'images qui a une référence temporelle (temporal_reference) de valeur zéro. Le fanion de suppression d'images (drop_frame_flag) peut être mis soit à '0' soit à '1'. Il ne peut être mis à '1' que si la fréquence d'image est de 29,97 Hz. S'il est mis à '0', on compte le nombre d'images par seconde en l'arrondissant par excès à l'entier le plus proche. Par exemple, on arrondira 29,97 Hz à 30 Hz. S'il est mis à '1', on ne compte pas les images numéro 0 et numéro 1 au début de chaque minute, sauf les minutes numéros 0, 10, 20, 30, 40, 50.

NOTE – Les informations acheminées par la structure time_code ne jouent aucun rôle dans le processus de décodage.

Tableau 6-11 – Structure *time_code*

| Structure <i>time_code</i> | Gamme de valeurs | Nombre de bits | Mnémonique |
|----------------------------|------------------|----------------|------------|
| drop_frame_flag | | 1 | uimsbf |
| time_code_hours | 0 - 23 | 5 | uimsbf |
| time_code_minutes | 0 - 59 | 6 | uimsbf |
| marker_bit | 1 | 1 | bslbf |
| time_code_seconds | 0 - 59 | 6 | uimsbf |
| time_code_pictures | 0 - 59 | 6 | uimsbf |

groupe d'images fermé (closed_gop) – Fanion codé sur un bit qui indique la nature des prédictions utilisées pour coder les éventuelles images de type B suivant immédiatement la première image à codage I suivant l'en-tête de groupe d'images.

La structure *closed_gop* est mise à '1' pour indiquer que ces images B ont été codées au seul moyen de la prédiction différée ou par codage intra.

Ce bit peut être utilisé au cours de tout processus d'édition faisant suite à un codage. Si les images précédentes ont été supprimées par édition, la structure *broken_link* (rupture de lien) peut être mise à '1' de façon qu'un décodeur puisse éviter d'afficher ces images B après la première image I qui suit l'en-tête de groupe d'images. Si toutefois le bit de la structure *closed_gop* est mis à '1', l'éditeur peut choisir de ne pas activer le bit de la structure *broken_link* car ces images B pourront être décodées correctement.

rupture de lien (broken_link) – Ce fanion codé sur un seul bit doit être mis à '0' au cours du codage. Il est mis à '1' pour indiquer que les premières images B qui suivent (le cas échéant) la première image I codée après l'en-tête de groupe d'images ne pourront pas être décodées correctement en raison de l'absence (pour édition) de la bi-trame de référence utilisée pour la prédiction.

Un décodeur peut utiliser ce fanion afin d'éviter l'affichage de bi-frames qui ne peuvent pas être décodées correctement.

6.3.9 En-tête d'image

code de déclenchement d'image (picture_start_code) – Cette structure est une chaîne de 32 éléments binaires, de valeur 00000100 en notation hexadécimale.

référence temporelle (temporal_reference) – Cette structure est un entier arithmétique de 10 bits associé à chaque image d'entrée.

La règle énoncée ci-dessous doit être appliquée lorsque le fanion *low_delay* est mis à la valeur zéro.

En cas de codage d'une image bi-trame sous forme de deux images monotrames, la référence temporelle associée à chaque image codée doit être la même. La référence temporelle de chaque image bi-trame codée doit être accrue d'une unité modulo 1024 en cas d'examen dans l'ordre d'affichage à l'issue du processus de décodage, sauf lorsque apparaît un en-tête de groupe d'images. Après un en-tête de groupe d'images, la référence temporelle de la première image bi-trame dans l'ordre d'affichage doit être mise à zéro.

La règle ci-dessous doit être appliquée si le fanion *low_delay* est mis à la valeur '1'.

Lorsque le fanion *low_delay* est mis à la valeur '1', certaines situations peuvent imposer un réexamen à plusieurs reprises de la mémoire tampon du vérificateur VBV avant d'en effacer une image codée (dite «image non décodée»).

En présence d'une image non décodée, la référence temporelle de l'image qui vient immédiatement après doit être égale à la référence temporelle de ladite image non décodée accrue de $N + 1$ modulo 1024, avec N nombre de réexamens de la mémoire tampon du vérificateur VBV ($N > 0$). Si l'image non décodée est immédiatement suivie d'un en-tête de groupe d'images, la référence temporelle de la première image codée suivant l'en-tête de groupe d'images doit être mise à la valeur N .

La référence temporelle d'une image qui ne suit pas immédiatement une image non décodée doit vérifier la règle définie en cas de fanion *low_delay* mis à la valeur zéro.

NOTE 1 – Si l'image non décodée est la première trame d'une image bi-trame codée au moyen d'images monotrames, alors les références temporelles des deux images monotrames de cette image bi-trame codée ne sont pas identiques.

type de codage d'image (picture_coding_type) – Cette structure indique si une image est à codage intra (image I), à codage prédictif (image P) ou à codage prédictif bidirectionnel (image B). La signification de la structure picture_coding_type est définie dans le Tableau 6-12.

NOTE 2 – Les images à codage intra ne comportant que des coefficients DC (images D), pouvant être utilisées selon ISO/CEI 11172-2, ne sont pas prises en compte par la présente Spécification.

Tableau 6-12 – Structure picture_coding_type

| Structure picture_coding_type | Méthode de codage |
|-------------------------------|--|
| 000 | Valeur interdite |
| 001 | Codage intra (I) |
| 010 | Codage prédictif (différé) (P) |
| 011 | Codage prédictif bidirectionnel (B) |
| 100 | (valeur à ne pas utiliser) (codage intra par coefficients DC selon ISO/CEI 11172-2) |
| 101 | Valeur réservée |
| 110 | Valeur réservée |
| 111 | Valeur réservée |

délai du vérificateur VBV (vbv_delay) – Cette structure est codée par un entier arithmétique de 16 bits. Dans tous les cas, sauf quand la structure vbv_delay est mise à la valeur hexadécimale FFFF, la valeur de vbv_delay est égale au nombre de périodes d'une horloge système à 90 kHz calculé à partir de l'horloge système à 27 MHz pendant lequel le vérificateur VBV doit attendre après avoir reçu l'octet final du code de déclenchement d'image, avant le décodage de l'image. La structure vbv_delay doit être codée de l'une ou l'autre des deux façons suivantes: soit afin de représenter le délai défini ci-dessus, soit par la valeur hexadécimale FFFF. Dès lors qu'un seul champ vbv_delay d'une séquence est codé par la valeur hexadécimale FFFF, tous les champs vbv_delay doivent être codés de la même façon. L'injection de données dans la mémoire tampon du vérificateur VBV est définie en Annexe C.3.2, si vbv_delay prend la valeur hexadécimale FFFF; sinon elle est définie en C.3.1.

Si le fanion low_delay est mis à la valeur '1' et si le flux binaire contient des images non décodées, les valeurs de vbv_delay codées dans les images non décodées risquent d'être fausses si elles ne sont pas égales à la valeur hexadécimale FFFF.

NOTE – Il existe plusieurs modes de calcul de vbv_delay dans un codeur.

Dans tous les cas il est possible de calculer cette valeur en observant que le délai de bout en bout à travers la mémoire tampon du codeur et celle du décodeur est constant pour toutes les images. Le codeur est en mesure de déterminer le délai subi par le code correspondant de déclenchement d'image dans la mémoire tampon du codeur, ainsi que le délai total de bout en bout. Aussi la valeur codée dans le champ vbv_delay (délai en mémoire tampon de décodeur du code de déclenchement d'image) est-elle calculée comme étant égale au délai total, diminué du délai subi par le code correspondant de déclenchement d'image dans la mémoire tampon du codeur, mesuré en périodes d'une horloge à 90 kHz, d'après l'horloge système à 27 MHz.

Sinon, et uniquement pour l'exploitation à débit binaire constant, il est possible de calculer comme suit ce nombre d'après l'état du vérificateur VBV:

$$vbv_delay_n = 90\,000 * B_n^* / R$$

avec:

$$n > 0$$

B_n^* = occupation de la mémoire du vérificateur VBV, mesurée en bits, immédiatement avant l'effacement de l'image n du tampon mais après élimination de tout en-tête, de toute donnée d'utilisateur et de tout bit de bourrage qui précèdent immédiatement les éléments de données de l'image n.

R = débit binaire actuel (c'est-à-dire en précision totale plutôt que selon la valeur quantifiée de la structure bit_rate dans l'en-tête de séquence).

Une méthode équivalente de calcul de `vbv_delay`, dans le cas de flux binaires à débit variable, peut consister à utiliser la relation indiquée en C.3.1. Ce calcul s'appuie sur une loi récurrente donnant la valeur `vbv_delay` en fonction de la valeur précédente, des délais de décodage de l'image actuelle et de l'image précédente, et du nombre d'octets de l'image précédente. La méthode en question est applicable à condition de connaître, au moment du codage de `vbv_delay`, le débit binaire moyen de transfert des données d'image de l'image précédente.

vecteur anticipé à codage sur tous les pixels (`full_pel_forward_vector`) – Ce fanion, qui est utilisé dans ISO/CEI 11172-2, n'est pas utilisé dans la présente Spécification. Il doit avoir la valeur zéro.

code f anticipé (`forward_f_code`) – Cette chaîne de 3 bits (qui est utilisée dans ISO/CEI 11172-2) n'est pas utilisée dans la présente Spécification. Elle doit avoir la valeur '111'.

vecteur différé à codage sur tous les pixels (`full_pel_backward_vector`) – Ce fanion, qui est utilisé dans ISO/CEI 11172-2, n'est pas utilisé dans la présente Spécification. Il doit avoir la valeur zéro.

code f différé (`backward_f_code`) – Cette chaîne de 3 bits (qui est utilisée dans ISO/CEI 11172-2) n'est pas utilisée dans la présente Spécification. Elle doit avoir la valeur '111'.

image par bit supplémentaire (`extra_bit_picture`) – Dans cette structure, un bit indique la présence ou l'absence des informations supplémentaires suivantes. Si le bit est mis à '1', une image selon informations supplémentaires (`extra_information_picture`) le suivra. S'il est mis à '0', aucune donnée supplémentaire ne le suivra. La structure `extra_bit_picture` doit être mise à '0', la valeur '1' étant réservée à d'éventuelles extensions pouvant être définies par l'UIT-T | ISO/CEI.

image selon informations supplémentaires (`extra_information_picture`) – Structure réservée. Un décodeur conforme à la présente Spécification ne doit pas tenir compte de cette structure s'il la rencontre dans un flux binaire (c'est-à-dire qu'il doit la détecter et la rejeter). Un flux conforme à la présente Spécification ne doit pas contenir cet élément syntaxique.

6.3.10 Extension relative au codage d'image

code f (`f_code[s][t]`) – Entier arithmétique de 4 bits prenant une valeur comprise entre 1 et 9 ou 15. La valeur zéro est interdite et les valeurs 10 à 14 sont réservées. Cette structure est utilisée dans le processus de décodage des vecteurs de mouvement. Voir 7.6.3.1.

Dans une image à codage I dont les vecteurs de mouvement de masquage ont la valeur zéro, la structure `f_code[s][t]` n'est pas utilisée (puisque les vecteurs de mouvement ne sont pas utilisés) et doit prendre systématiquement la valeur 15 (tous des un).

De même, dans une image à codage I ou P, la structure `f_code[1][t]` n'est pas utilisée dans le processus de décodage (puisque le code f s'applique à des vecteurs de mouvement différés) et cette structure doit prendre systématiquement la valeur 15 (tous des un).

Voir le Tableau 7-7 pour la signification des indices s et t.

précision du codage intra pour les coefficients DC (composante constante) (`intra_dc_precision`) – Entier codé sur 2 bits conformément au Tableau 6-13.

Tableau 6-13 – Précision de codage intra pour les coefficients DC de la composante constante

| <code>intra_dc_precision</code> | Précision (bits) |
|---------------------------------|------------------|
| 00 | 8 |
| 01 | 9 |
| 10 | 10 |
| 11 | 11 |

Le processus de quantification inverse pour les coefficients DC du codage intra est modifié par ce paramètre comme indiqué au 7.4.1.

structure d'image (picture_structure) – Entier codé sur 2 bits, conformément au Tableau 6-14.

Tableau 6-14 – Signification de la structure picture_structure

| picture_structure | Signification |
|-------------------|---------------------------|
| 00 | Valeur réservée |
| 01 | Trame supérieure |
| 10 | Trame inférieure |
| 11 | Image bi-trame (complète) |

Lorsqu'une image bi-trame est codée sous la forme de deux images monotrames, les deux trames doivent toujours être du même type de codage d'image, sauf lorsque la première trame codée est une image à codage I: dans ce cas, la deuxième trame peut être soit une image I ou une image P.

La première trame codée d'une bi-trame peut être une trame supérieure ou une trame inférieure. La trame suivante doit toujours être de parité opposée.

Lorsqu'une image bi-trame est codée sous la forme de deux images monotrames, les éléments syntaxiques suivants peuvent être réglés indépendamment dans chaque image monotrame:

- f_code[0][0], f_code[0][1];
- f_code[1][0], f_code[1][1];
- intra_dc_precision, concealment_motion_vectors, q_scale_type;
- intra_vlc_format, alternate_scan.

trame supérieure en premier (top_field_first) – La signification de cette structure dépend des structures picture_structure, progressive_sequence et repeat_first_field.

Si la structure progressive_sequence a la valeur '0', ce fanion indique quelle trame d'une image bi-trame reconstruite sort la première du processus de décodage.

Dans une image monotrame, la structure top_field_first doit avoir la valeur '0' et la seule trame émise par le processus de décodage est l'image monotrame décodée.

Dans une image bi-trame, le fait que la structure top_field_first soit mise à '1' indique que la trame supérieure de la bi-trame reconstruite est la première trame issue du processus de décodage. Le fait que la structure top_field_first soit mise à '0' indique que la trame inférieure de la bi-trame reconstruite est la première trame issue du processus de décodage.

Si la structure progressive_sequence a la valeur '1', ce fanion, combiné à la structure repeat_first_field, indique le nombre de fois (une, deux ou trois fois) où l'image bi-trame reconstruite est émise par le processus de décodage.

Si la structure repeat_first_field a la valeur 0, la structure top_field_first doit être mise à '0'. Dans ce cas, la sortie du processus de décodage correspondant à cette bi-trame reconstruite se compose d'une seule image bi-trame à balayage progressif.

Si la structure top_field_first est mise à 0 et la structure repeat_first_field à '1', la sortie du processus de décodage correspondant à cette bi-trame reconstruite se compose de deux images bi-frames à balayage progressif.

Si la structure top_field_first est mise à 1 et la structure repeat_first_field à '1', la sortie du processus de décodage correspondant à cette bi-trame reconstruite se compose de trois images bi-frames à balayage progressif.

codage DCT et prédiction sur bi-frames (frame_pred_frame_dct) – Si ce fanion est mis à '1', seul le codage DCT sur bi-frames et seule la prédiction sur bi-frames sont utilisés. Dans une image monotrame, cette structure doit avoir la valeur '0'. Elle doit avoir la valeur '1' si la structure progressive_frame a la valeur '1'. Ce fanion affecte la syntaxe du flux binaire.

vecteurs de mouvement de masquage (concealment_motion_vectors) – Ce fanion est mis à la valeur '1' afin d'indiquer que les vecteurs de mouvement sont codés pour des macroblocs à codage intra. La valeur '0' de ce fanion indique qu'aucun vecteur de mouvement n'est codé pour des macroblocs à codage intra.

type d'échelle de quantification (q_scale_type) – Ce fanion affecte le processus de quantification inverse comme décrit au paragraphe 7.4.2.2.

format de codage intra à longueur variable (intra_vlc_format) – Ce fanion affecte le décodage des données relatives aux coefficients de transformation, comme décrit au paragraphe 7.2.1.

variante de balayage des coefficients (alternate_scan) – Ce fanion affecte le décodage des données relatives aux coefficients de transformation, comme décrit au paragraphe 7.3.

répétition de la première trame (repeat_first_field) – Ce fanion n'est applicable qu'à une image bi-trame. Pour une image monotrème, il doit être mis à zéro et n'a pas d'incidence sur le processus de décodage.

Si la structure progressive_sequence a la valeur 0 et la structure progressive_frame la valeur 0, la structure repeat_first_field doit avoir la valeur 0 et la sortie du processus de décodage correspondant à cette image bi-trame reconstruite consiste en deux trames.

Si la structure progressive_sequence a la valeur 0 et que la structure progressive_frame soit égale à 1:

Si ce fanion est mis à 0, la sortie du processus de décodage correspondant à cette image bi-trame reconstruite se compose de deux trames. La première trame (supérieure ou inférieure selon ce qui est indiqué par top_field_first) est suivie de l'autre trame.

Si ce fanion est mis à 1, la sortie du processus de décodage correspondant à cette image bi-trame reconstruite se compose de trois trames. La première trame (supérieure ou inférieure selon ce qui est indiqué par top_field_first) est suivie de l'autre trame, puis la première trame est répétée.

Si la structure progressive_sequence a la valeur 1:

Si ce fanion est mis à 0, la sortie du processus de décodage correspondant à cette image bi-trame reconstruite se compose d'une seule trame.

Si ce fanion est mis à 1, la sortie du processus de décodage correspondant à cette image bi-trame reconstruite se compose de deux ou de trois trames, selon la valeur de top_field_first.

format de chrominance de type 4:2:0 (chroma_420_type) – Si la structure chroma_format est de type «4:2:0», la valeur de la structure chroma_420_type doit être la même que celle de la structure progressive_frame; sinon, la structure chroma_420_type n'a pas de signification et doit être mise à zéro. Ce fanion existe pour des raisons d'ordre historique.

bi-trame à balayage progressif (progressive_frame) – Si cette structure est mise à 0, elle indique que les deux trames de l'image bi-trame sont entrelacées et séparées (pour leurs échantillons spatiaux homologues) par un intervalle de temps correspondant à la période de trame. Dans ce cas, la restriction suivante est applicable:

- la structure repeat_first_field doit avoir la valeur 0 (durée pour 2 trames).

Si la structure progressive_frame est mise à 1, elle indique que les deux trames (de l'image bi-trame) correspondent en fait au même point sur l'échelle des temps. Dans ce cas, un certain nombre de restrictions s'appliquent à d'autres paramètres et fanion du flux binaire:

- la structure picture_structure doit avoir la valeur «Frame»;
- la structure frame_pred_frame_dct doit avoir la valeur 1.

La structure progressive_frame est utilisée lorsque la séquence vidéo sert de couche inférieure d'une séquence à échelonnabilité spatiale. Dans ce cas, elle affecte le processus de suréchantillonnage visant à former une prédiction dans la couche d'amélioration issue de la couche inférieure.

fanion d'affichage composite (composite_display_flag) – Ce fanion est mis à 1 pour indiquer lesquelles des trames suivantes doivent être utilisées lorsque les images d'entrée ont été codées sous forme de signal vidéo composite (analogique), avant codage de ce signal en un flux binaire conforme à la présente Spécification. Si ce fanion est mis à zéro, ces paramètres n'apparaissent pas dans le flux binaire.

Cette information se rapporte à l'image qui suit immédiatement l'extension. Si cette image est de type bi-trame, l'information se rapporte à la première trame de cette image. L'information correspondante pour la deuxième trame peut en être déduite (il n'y a pas de moyen de la représenter dans le flux binaire).

NOTES

1 Les différentes trames qui sont comprises dans le flux binaire lorsque le fanion d'affichage composite est mis à la valeur '1' ne sont pas utilisées dans le processus de décodage.

2 La structure repeat_first_field provoquera la répétition d'une trame vidéo composite extraite d'une séquence de 4 ou de 8 trames. Il est recommandé de ne pas activer simultanément les deux structures repeat_first_field et composite_display_flag.

signal V (v_axis) – Entier codé sur 1 bit qui n'est utilisé que lorsque le flux binaire représente un signal qui a été préalablement codé selon la norme des systèmes PAL. La structure v_axis est mise à 1 en cas de signe positif et à 0 dans l'autre cas.

séquence de trames (field_sequence) – Entier codé sur 3 bits qui définit le nombre de trames dans la séquence de huit trames des systèmes PAL ou dans la séquence de quatre trames des systèmes NTSC, conformément au Tableau 6-15.

Tableau 6-15 – Définition de la structure field_sequence

| Structure field_sequence | Bi-trames | Trames |
|--------------------------|-----------|--------|
| 000 | 1 | 1 |
| 001 | 1 | 2 |
| 010 | 2 | 3 |
| 011 | 2 | 4 |
| 100 | 3 | 5 |
| 101 | 3 | 6 |
| 110 | 4 | 7 |
| 111 | 4 | 8 |

sous-porteuse (sub_carrier) – Entier codé sur 1 bit. S'il est mis à 0, ce fanion indique que la relation entre sous-porteuse et fréquence ligne est correcte. S'il est mis à 1, cette relation n'est pas correcte.

amplitude de pointe (burst_amplitude) – Entier de 7 bits définissant l'amplitude de pointe (pour systèmes PAL et NTSC seulement). L'amplitude de pointe de sous-porteuse est quantifiée sous la forme du signal de luminance conforme à la Recommandation UIT-R BT.601, avec suppression du bit de poids le plus fort.

phase de sous-porteuse (sub_carrier_phase) – Entier de 8 bits définissant la phase de la sous-porteuse de référence au point de synchronisation de trame par rapport à l'instant du début de trame, conformément à la Recommandation UIT-R BT.470 (voir le Tableau 6-16).

Tableau 6-16 – Définition de la structure sub_carrier_phase

| Structure sub_carrier_phase | Phase |
|-----------------------------|--------------------------------|
| 0 | $([360^\circ \div 256] * 0)$ |
| 1 | $([360^\circ \div 256] * 1)$ |
| ... | ... |
| 255 | $([360^\circ \div 256] * 255)$ |

6.3.11 Extension relative à la matrice de quantification

Chaque matrice de quantification possède un ensemble de valeurs par défaut. Lorsqu'un code d'en-tête de séquence est décodé, toutes les matrices doivent être réduites à leurs valeurs par défaut. Des matrices définies par l'utilisateur peuvent être téléchargées et cette possibilité peut se présenter dans une structure de type `sequence_header()` ou `quant_matrix_extension()`.

En format 4:2:0, deux matrices seulement sont utilisées: l'une pour les blocs à codage intra, l'autre pour les blocs à codage non intra.

En format 4:2:2 ou 4:4:4, quatre matrices sont utilisées: deux matrices à codage intra et deux matrices à codage non intra pour les blocs de luminance et de chrominance, respectivement. A noter qu'il est toutefois possible de télécharger la même matrice définie par l'utilisateur, simultanément dans la matrice de luminance et dans la matrice de chrominance.

La matrice par défaut pour les blocs à codage intra (de luminance comme de chrominance) est la suivante:

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 8 | 16 | 19 | 22 | 26 | 27 | 29 | 34 |
| 16 | 16 | 22 | 24 | 27 | 29 | 34 | 37 |
| 19 | 22 | 26 | 27 | 29 | 34 | 34 | 38 |
| 22 | 22 | 26 | 27 | 29 | 34 | 37 | 40 |
| 22 | 26 | 27 | 29 | 32 | 35 | 40 | 48 |
| 26 | 27 | 29 | 32 | 35 | 40 | 48 | 58 |
| 26 | 27 | 29 | 34 | 38 | 46 | 56 | 69 |
| 27 | 29 | 35 | 38 | 46 | 56 | 69 | 83 |

La matrice par défaut pour les blocs à codage non intra (de luminance comme de chrominance) est la suivante:

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |

charger une matrice de quantification intra (load_intra_quantiser_matrix) – Fanion d'un seul bit qui est mis à '1' si une matrice de quantification intra doit suivre. Il est mis à zéro s'il n'y a aucune modification des valeurs à utiliser.

matrice de quantification intra (intra_quantiser_matrix) – Cette structure contient une liste de soixante-quatre entiers non signés, codés sur 8 bits. Les nouvelles valeurs, codées dans l'ordre par défaut de balayage en zigzag décrit au 7.3.1, remplacent les anciennes valeurs. La première valeur de cette matrice doit toujours être 8. Pour tous les entiers non signés, codés sur 8 bits, la valeur 0 est interdite. Avec les formats 4:2:2 et 4:4:4, les nouvelles valeurs doivent être utilisées aussi bien pour la matrice de codage intra de luminance que pour la matrice de codage intra de chrominance. Cette dernière matrice pourra cependant être remplacée par une autre matrice.

charger une matrice de quantification non intra (load_non_intra_quantiser_matrix) – Fanion d'un seul bit qui est mis à '1' si une matrice de quantification non intra doit suivre. Il est mis à zéro s'il n'y a aucune modification des valeurs à utiliser.

matrice de quantification non intra (non_intra_quantiser_matrix) – Cette structure contient une liste de soixante-quatre entiers non signés, codés sur 8 bits. Les nouvelles valeurs, codées dans l'ordre par défaut de balayage en zigzag décrit au 7.3.1, remplacent les anciennes valeurs. Pour tous les entiers non signés, codés sur 8 bits, la valeur 0 est interdite. Avec les formats 4:2:2 et 4:4:4, les nouvelles valeurs doivent être utilisées aussi bien pour la matrice de codage non intra de luminance que pour la matrice de codage non intra de chrominance. Cette dernière matrice pourra cependant être remplacée par une autre matrice.

charger une matrice de quantification intra pour la chrominance (load_chroma_intra_quantiser_matrix) – Fanion d'un seul bit qui est mis à '1' si une matrice de quantification intra pour la chrominance doit suivre. Il est mis à zéro s'il n'y a aucune modification des valeurs à utiliser. Si le format de chrominance est de type «4:2:0», ce fanion doit prendre la valeur '0'.

matrice de quantification intra pour la chrominance (chroma_intra_quantiser_matrix) – Cette structure contient une liste de soixante-quatre entiers non signés, codés sur 8 bits. Les nouvelles valeurs, codées dans l'ordre par défaut de balayage en zigzag décrit au 7.3.1, remplacent les anciennes valeurs. La première valeur de cette matrice doit toujours être 8. Pour tous les entiers non signés, codés sur 8 bits, la valeur 0 est interdite.

charger une matrice de quantification non intra pour la chrominance (load_chroma_non_intra_quantiser_matrix) – Fanion d'un seul bit qui est mis à '1' si une matrice de quantification non intra pour la chrominance doit suivre. Il est mis à zéro s'il n'y a aucune modification des valeurs à utiliser. Si le format de chrominance est de type «4:2:0», ce fanion doit prendre la valeur '0'.

matrice de quantification non intra pour la chrominance (chroma_non_intra_quantiser_matrix) – Cette structure contient une liste de soixante-quatre entiers non signés, codés sur 8 bits. Les nouvelles valeurs, codées dans l'ordre par défaut de balayage en zigzag décrit au 7.3.1, remplacent les anciennes valeurs. Pour tous les entiers non signés, codés sur 8 bits, la valeur 0 est interdite.

6.3.12 Extension d'affichage de l'image

La présente Spécification ne définit pas le processus d'affichage. Les informations contenues dans cette extension n'ont pas d'incidence sur le processus de décodage et peuvent être négligées par les décodeurs conformes à la présente Spécification.

L'extension d'affichage d'image permet de déplacer, image par image, la position du rectangle d'affichage, dont les dimensions sont spécifiées par la structure `sequence_display_extension()`. Une application particulière de cette extension est la mise en œuvre du recadrage des images (pour occuper tout l'écran).

décalage horizontal du centre d'image (frame_centre_horizontal_offset) – Entier signé de 16 bits qui indique le décalage horizontal en unités de $1/16^e$ pel. Une valeur positive indique que le centre de l'image reconstruite est à droite du centre du rectangle d'affichage.

décalage vertical du centre d'image (frame_centre_vertical_offset) – Entier algébrique de 16 bits qui indique le décalage vertical en unités de $1/16^e$ pel. Une valeur positive indique que le centre de l'image reconstruite se trouve au-dessous du centre du rectangle d'affichage.

Les dimensions de la zone rectangulaire d'affichage sont définies dans la structure `sequence_display_extension()`. Les coordonnées de la région définie à l'intérieur de l'image codée sont définies dans la structure `extension d'affichage d'image`.

Le centre de l'image bi-trame reconstruite coïncide avec le centre du rectangle défini par les dimensions horizontale et verticale.

Etant donné que (dans le cas d'une séquence à balayage entrelacé) une image codée peut correspondre à une, deux ou trois trames décodées, la structure `picture_display_extension()` peut comprendre jusqu'à trois décalages.

Le nombre de décalages du centre d'image contenu dans l'extension d'affichage d'image doit être défini comme suit:

```

if ( progressive_sequence == 1 ) {
    if ( repeat_first_field == '1' ) {
        if ( top_field_first == '1' )
            number_of_frame_centre_offsets = 3
        else
            number_of_frame_centre_offsets = 2
    } else {
        number_of_frame_centre_offsets = 1
    }
} else {
    if ( picture_structure == "field" ) {
        number_of_frame_centre_offsets = 1
    } else {
        if ( repeat_first_field == '1' )
            number_of_frame_centre_offsets = 3
        else
            number_of_frame_centre_offsets = 2
    }
}

```

Une structure `picture_display_extension()` ne doit pas apparaître si une structure `sequence_display_extension()` ne suit pas le précédent en-tête de séquence.

Si une image donnée ne possède pas d'extension d'affichage de l'image, c'est le décalage de centre d'image décodé le plus récemment qui doit être utilisé. A noter que chacun des décalages de centre d'image manquants possède la même valeur (même si deux ou trois décalages de centre d'image sont contenus dans l'extension d'affichage d'image). A la suite d'un en-tête de séquence, la valeur 0 doit être utilisée pour tous les décalages de centre d'image, jusqu'à ce qu'une extension d'affichage d'image définisse des valeurs différentes de zéro.

La Figure 6-16 montre les paramètres d'affichage de l'image. Comme indiqué, les décalages du centre d'image, contenus dans la structure `picture_display_extension()` doivent spécifier la position du centre de l'image bi-trame reconstruite, par rapport au centre du rectangle d'affichage.

NOTES

- 1 Le rectangle d'affichage peut également être plus grand que l'image bi-trame reconstruite.
- 2 Même dans une image monotrème, le décalage vertical du centre d'image continue à représenter le décalage du centre d'image bi-trème en $1/16^{\circ}$ de ligne d'image bi-trème (et non de ligne de la monotrème).
- 3 Dans l'exemple de la Figure 6-16, les décalages horizontaux et verticaux du centre d'image sont tous deux négatifs.

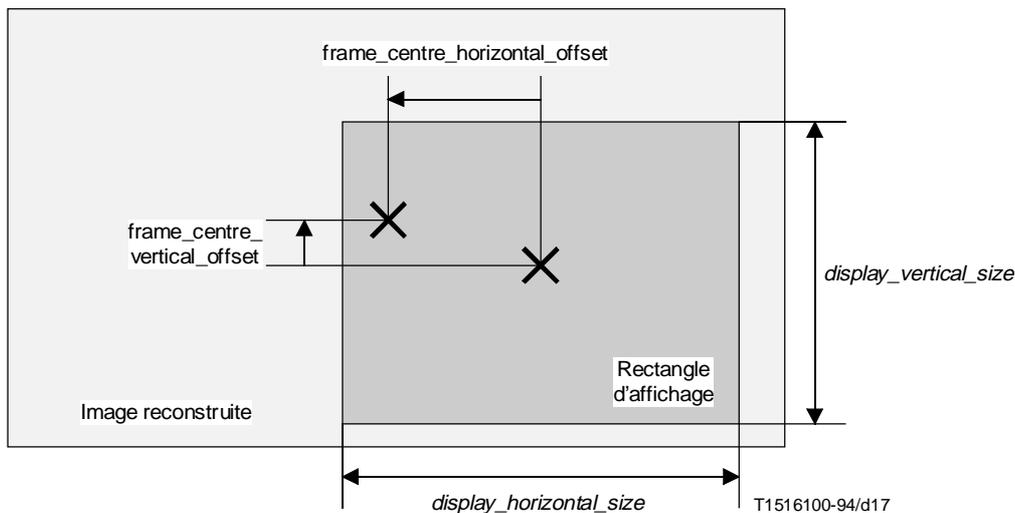


Figure 6-16 – Paramètres de décalage de centre d'image

6.3.12.1 Recadrage en plein écran

Les décalages de centre d'image peuvent servir à mettre en œuvre le procédé de recadrage en plein écran, par lequel on définit une zone rectangulaire (correspondant au plein écran) qui est déplacée panoramiquement sur toute l'image reconstruite.

A titre d'exemple uniquement, on peut citer l'emploi de ce procédé afin d'extraire une image pour écran en format 3/4 d'une image pour écran en format 9/16. Cela permet à un décodeur de fournir des images utilisables par un téléviseur conventionnel à partir d'un format codé au départ pour une plus haute définition. La zone définie par le format 3/4 est censée contenir la partie «la plus intéressante» de l'image d'origine.

La région d'image en format 3/4 est définie par les structures `display_horizontal_size` et `display_vertical_size`. La région d'image en format 9/16 est définie par les structures `horizontal_size` et `vertical_size`.

6.3.13 Extension à l'échelonnabilité temporelle des images

NOTE – Voir aussi 7.9.

code de sélection des références (`reference_select_code`) – Structure codée sur 2 bits qui identifie les bi-trèmes ou les monotrèmes de référence pour la prédiction, selon le type d'image.

référence temporelle anticipée (forward_temporal_reference) – Valeur d'entier non signée de 10 bits qui indique la référence temporelle de la bi-trame de couche inférieure à utiliser pour effectuer la prédiction anticipée. Si la couche inférieure indique une référence temporelle contenant plus de 10 bits, ce sont les bits les moins significatifs qui sont repris dans ce code. Si la couche inférieure indique une référence temporelle contenant moins de 10 bits, ce sont tous ces bits qui sont repris dans le code, les bits les plus significatifs étant forcés à zéro.

référence temporelle différée (backward_temporal_reference) – Valeur d'entier non signée de 10 bits qui indique la référence temporelle de la bi-trame de couche inférieure à utiliser pour effectuer la prédiction différée. Si la couche inférieure indique une référence temporelle contenant plus de 10 bits, ce sont les bits les moins significatifs qui sont repris dans ce code. Si la couche inférieure indique une référence temporelle contenant moins de 10 bits, ce sont tous ces bits qui sont repris dans le code, les bits les plus significatifs étant forcés à zéro.

6.3.14 Extension à l'échelonnabilité spatiale des images

référence temporelle de couche inférieure (lower_layer_temporal_reference) – Valeur d'entier non signée de 10 bits qui indique la référence temporelle de la bi-trame de couche inférieure à utiliser pour effectuer la prédiction. Si la couche inférieure indique une référence temporelle contenant plus de 10 bits, ce sont les bits les moins significatifs qui sont repris dans ce code. Si la couche inférieure indique une référence temporelle contenant moins de 10 bits, ce sont tous ces bits qui sont repris dans le code, les bits les plus significatifs étant forcés à zéro.

décalage horizontal de couche inférieure (lower_layer_horizontal_offset) – Cet entier relatif de 15 bits (format complément à deux) spécifie le décalage horizontal (du coin gauche supérieur) de l'image bi-trame de couche inférieure suréchantillonnée, par rapport à l'image de la couche d'amélioration. Ce nombre est exprimé en unités de largeur d'échantillon d'image de couche d'amélioration. Si le format de chrominance est 4:2:0 ou 4:2:2, ce paramètre doit être un nombre pair.

décalage vertical de couche inférieure (lower_layer_vertical_offset) – Cet entier relatif de 15 bits (format complément à deux) spécifie le décalage vertical (du coin gauche supérieur) de l'image bi-trame de couche inférieure suréchantillonnée, par rapport à l'image de la couche d'amélioration. Ce nombre est exprimé en unités de largeur d'échantillon d'image de couche d'amélioration. Si le format de chrominance est 4:2:0, ce paramètre doit être un nombre pair.

index de la table des codes de pondération spatio-temporelle (spatial_temporal_weight_code_table_index) – Cet entier codé sur 2 bits indique quelle table de codes de pondération spatio-temporelle doit être utilisée conformément au 7.7. Les valeurs de cette structure sont définies dans le Tableau 7-21.

bi-trame progressive de couche inférieure (lower_layer_progressive_frame) – Ce fanion doit être mis à 0 si l'image bi-trame de couche inférieure est entrelacée et être mis à '1' si elle est progressive. L'utilisation de ce fanion dans le processus de suréchantillonnage à échelonnabilité spatiale est définie au 7.7.

sélection de trame désentrelacée de couche inférieure (lower_layer_deinterlaced_field_select) – Cette structure affecte le processus de suréchantillonnage à échelonnabilité spatiale, comme défini au 7.7.

6.3.15 Extension de droit d'auteur

identificateur du code de déclenchement d'une extension (extension_start_code_identifier) – La valeur de codage de ce champ est un entier de 4 bits permettant d'identifier l'extension. Voir le Tableau 6-2.

fanion de droit d'auteur (copyright_flag) – Fanion codé sur un bit. Lorsqu'il est mis à la valeur '1', il indique la protection par droit d'auteur des données vidéo d'origine codées dans toutes les images codées qui suivent l'extension de droit d'auteur, dans l'ordre de codage, jusqu'à l'extension de droit d'auteur suivante ou jusqu'au prochain code de fin de séquence. L'identificateur de droit d'auteur et le numéro de droit d'auteur identifient l'ouvrage protégé par droit d'auteur. Lorsque ce fanion est mis à la valeur '0', cette indication ne signifie ni la présence ni l'absence d'une protection par droit d'auteur des données vidéo d'origine codées dans toutes les images codées qui suivent l'extension de droit d'auteur, dans l'ordre de codage.

identificateur de droit d'auteur (copyright_identifier) – Cet entier de 8 bits identifie un organisme d'enregistrement désigné par la Norme internationale ISO/CEI JTC 1/SC 29. La mise à zéro de ce paramètre indique que cette information n'est pas disponible. Le numéro de droit d'auteur doit prendre la valeur zéro lorsque l'identificateur de droit d'auteur est égal à zéro.

Si le fanion de droit d'auteur est mis à la valeur '0', l'identificateur de droit d'auteur n'a pas de signification et doit prendre la valeur zéro.

original ou copie (original_or_copy) – Fanion codé sur 1 bit. Mis à la valeur '1' il indique que les données constituent un document original, tandis que la valeur '0' indique qu'il s'agit d'une copie.

valeur réservée (reserved) – Entier de 7 bits réservé aux extensions futures. Il doit être mis à la valeur zéro.

numéro de droit d'auteur 1 (copyright_number_1) – Entier codé sur 20 bits représentant les bits 44 à 63 du numéro de droit d'auteur.

numéro de droit d'auteur 2 (copyright_number_2) – Entier codé sur 22 bits représentant les bits 22 à 43 du numéro de droit d'auteur.

numéro de droit d'auteur 3 (copyright_number_3) – Entier codé sur 22 bits représentant les bits 0 à 21 du numéro de droit d'auteur.

numéro de droit d'auteur (copyright_number) – Entier codé sur 64 bits, dérivé du numéro de droit d'auteur 1, du numéro de droit d'auteur 2 et du numéro de droit d'auteur 3, de la manière suivante:

$\text{copyright_number} = (\text{copyright_number_1} \ll 44) + (\text{copyright_number_2} \ll 22) + \text{copyright_number_3}$.

La signification de `copyright_number` n'est définie que si le fanion de droit d'auteur est mis à la valeur '1'. Dans ce cas, la valeur du numéro de droit d'auteur identifie exclusivement l'ouvrage bénéficiant d'une protection par droit d'auteur marqué par l'extension correspondante; elle est indiquée par l'organisme d'enregistrement désigné par l'identificateur de droit d'auteur. La mise à la valeur zéro du numéro de droit d'auteur indique qu'il n'existe pas de numéro d'identification de l'ouvrage bénéficiant d'une protection par droit d'auteur.

En cas de mise à la valeur '0' du fanion de droit d'auteur, le numéro de droit d'auteur n'a pas de signification et doit être mis à la valeur 0.

6.3.16 Tranche

code de déclenchement de tranche (slice_start_code) – Cette structure est une chaîne de 32 bits, dont les 24 premiers ont la valeur (hexadécimale) 000001 et dont les 8 derniers, qui représentent la position verticale de tranche, ont une valeur comprise entre 01 et AF inclus (en notation hexadécimale).

position verticale de tranche (slice_vertical_position) – Cette structure est codée sur les 8 derniers bits de la structure `slice_start_code`. C'est un entier non signé qui donne la position verticale en unités de macrobloc pour le premier macrobloc de la tranche.

Dans les grandes images (lorsque la dimension verticale de l'image est supérieure à 2800 lignes) la position verticale de tranche est prolongée par la structure `slice_vertical_position_extension`.

La rangée du macrobloc peut être calculée comme suit:

```
if ( vertical_size > 2800 )
    mb_row = (slice_vertical_position_extension << 7) + slice_vertical_position - 1;
else
    mb_row = slice_vertical_position - 1;
```

La valeur de la position verticale de tranche pour la première rangée des macroblocs est 1. Certaines tranches peuvent avoir la même position verticale car les tranches commencent et finissent à un endroit quelconque. La valeur maximale de la position verticale de tranche est 175, sauf si une structure d'extension de position verticale de tranche est présente, auquel cas la position verticale de tranche sera dans l'étendue [1:128].

point d'arrêt de priorité (priority_breakpoint) – Cette structure est un entier de 7 bits qui indique le point de la syntaxe où le flux binaire doit être subdivisé. Les valeurs autorisées et leur interprétation sémantique sont indiquées dans le Tableau 7-30. Cette structure doit prendre la valeur 0 dans la partition 1.

code d'échelle de quantification (quantiser_scale_code) – Entier non signé de 5 bits compris entre 1 et 31. Le décodeur doit utiliser cette valeur jusqu'à ce qu'un autre code d'échelle de quantification soit rencontré, soit dans la tranche soit dans le macrobloc. La valeur zéro est interdite.

fanion de tranche à codage intra (intra_slice_flag) – Ce fanion doit être mis à '1' pour indiquer la présence des structures `intra_slice` et `reserved_bits` dans le flux binaire.

tranche à codage intra (intra_slice) – Cette structure doit être mise à '0' si l'un quelconque des macroblocs de la tranche est un macrobloc à codage non intra. Si tous les macroblocs sont à codage intra, la structure `intra_slice` peut être mise à '1'.

Elle peut être omise du flux binaire (en mettant le fanion `intra_slice_flag` à '0'); dans ce cas, on doit supposer qu'elle a la valeur zéro.

La structure `intra_slice` n'est pas utilisée par le processus de décodage. Elle est destinée à faciliter l'exécution d'une lecture rapide avant/arrière (voir D.12) dans une application de support numérique d'enregistrement (DSM).

bits réservés (`reserved_bits`) – Cette structure est un entier de 7 bits qui doit avoir la valeur zéro, d'autres valeurs sont réservées.

bit d'une information supplémentaire de tranche (`extra_bit_slice`) – Ce fanion indique la présence de l'information supplémentaire suivante. S'il est mis à '1', une structure d'information supplémentaire de tranche (`extra_information_slice`) le suivra. S'il est mis à '0', aucune donnée ne le suit. Ce fanion doit être mis à '0', la valeur '1' étant réservée à de possibles extensions pouvant être définies par l'UIT-T | ISO/CEI.

information supplémentaire de tranche (`extra_information_slice`) – Structure réservée. Un décodeur conforme à la présente Spécification, rencontrant une information supplémentaire de tranche dans un flux binaire, doit l'ignorer (c'est-à-dire la détecter dans le flux et la rejeter). Un flux binaire conforme à la présente Spécification ne doit pas contenir cet élément syntaxique.

6.3.17 Macroblocs

NOTE – Le bourrage de macrobloc (structure «`macroblock_stuffing`»), prévu dans ISO/CEI 11172-2, ne doit pas être utilisé dans un flux binaire défini selon la présente Spécification.

échappement de macrobloc (`macroblock_escape`) – Cette structure est une chaîne de longueur fixe: '0000 0001 000' qui est utilisée lorsque la différence entre l'adresse d'un macrobloc et l'adresse de macrobloc précédente est supérieure à 33. Une telle différence fait que la valeur de la structure `macroblock_address_increment` sera plus grande de 33 que la valeur qui sera décodée par les mots de code `macroblock_escape` et `macroblock_address_increment` suivants.

Par exemple, s'il y a deux mots de code de type `macroblock_escape` avant la structure `macroblock_address_increment`, le système ajoute 66 à la valeur indiquée par la structure `macroblock_address_increment`.

incrémentation (`progression`) de l'adresse du macrobloc (`macroblock_address_increment`) – Cette structure est représentée par un entier à codage à longueur variable selon le Tableau B.1 en annexe. Elle indique la différence de valeur entre les structures `macroblock_address` et `previous_macroblock_address`. La valeur maximale de cette structure est 33. Les valeurs supérieures à 33 pourront être codées au moyen du mot de code `macroblock_escape`.

La structure `macroblock_address` est une variable définissant la position absolue du macrobloc actuel. Sa valeur est zéro pour l'adresse du macrobloc de coin supérieur gauche.

La structure `previous_macroblock_address` est une variable définissant la position absolue du dernier macrobloc non sauté (voir au 7.6.6 la définition des macroblocs sautés) sauf en début de tranche, où cette structure est réinitialisée comme suit:

$$\text{previous_macroblock_address} = (\text{mb_row} * \text{mb_width}) - 1$$

La position spatiale horizontale d'un macrobloc dans l'image (`mb_column`) peut être calculée, en unités de macrobloc, d'après l'adresse du macrobloc comme suit:

$$\text{mb_column} = \text{macroblock_address} \% \text{mb_width}$$

où `mb_width` est le nombre de macroblocs dans une rangée de l'image.

Sauf en début de tranche, si la valeur d'adresse de macrobloc déduite des codes (éventuels) `macroblock_address_increment` et `macroblock_escape` diffère de l'adresse de macrobloc précédente (`previous_macroblock_address`) de plus que 1, alors certains macroblocs ont été sautés. Il est prescrit:

- qu'il n'y ait pas de macroblocs sautés dans des images de type I, sauf si:
 - soit une structure `picture_spatial_scalable_extension()` suit l'en-tête de l'image actuelle;
 - soit une structure `sequence_scalable_extension()` est présente dans le flux binaire et que la structure `scalable_mode` ait la valeur «échelonnabilité SNR»;
- que les premier et dernier macroblocs d'une tranche ne soient pas sautés;
- que, dans une image de type B, il n'y ait pas de macroblocs sautés immédiatement après un macrobloc ayant un fanion `macroblock_intra` de valeur 1.

6.3.17.1 Modes des macroblocs

type de macrobloc (macroblock_type) – Cette structure, à code de longueur variable, indique la méthode de codage et le contenu du macrobloc conformément aux Tableaux B.2 à B.8; ce type est sélectionné par les structures `picture_coding_type` et `scalable_mode`.

macrobloc à code d'échelle de quantification (macroblock_quant) – Structure dépendant du type de macrobloc conformément aux Tableaux B.2 à B.8. Elle est mise à 1 pour indiquer que le code d'échelle de quantification est présent dans le flux binaire.

macrobloc à vecteurs de mouvement anticipés (macroblock_motion_forward) – Structure dépendant du type de macrobloc conformément aux Tableaux B.2 à B.8. Ce fanion modifie la syntaxe du flux binaire et est utilisé par le processus de décodage.

macrobloc à vecteurs de mouvement différés (macroblock_motion_backward) – Structure dépendant du type de macrobloc conformément aux Tableaux B.2 à B.8. Ce fanion modifie la syntaxe du flux binaire et est utilisé par le processus de décodage.

macrobloc à codage de structure des blocs (macroblock_pattern) – Structure dépendant du type de macrobloc conformément aux Tableaux B.2 à B.8. Ce fanion est mis à 1 pour indiquer que la structure `coded_block_pattern` est présente dans le flux binaire.

macrobloc à codage intra (macroblock_intra) – Structure dépendant du type de macrobloc conformément aux Tableaux B.2 à B.8. Ce fanion modifie la syntaxe du flux binaire et est utilisé par le processus de décodage.

fanion de code de pondération spatio-temporelle (spatial_temporal_weight_code_flag) – Structure dépendant du type de macrobloc, indiquant si le code de pondération spatio-temporelle est présent dans le flux binaire.

Si le fanion de code de pondération spatio-temporelle est mis à la valeur '0' (indiquant ainsi l'absence de code de pondération spatio-temporelle dans le flux binaire), la classe de pondération spatio-temporelle est déterminée d'après les Tableaux B.5 à B.7. Si le fanion de code de pondération spatio-temporelle est mis à la valeur '1', la classe de pondération spatio-temporelle est déterminée d'après le Tableau 7-20.

code de pondération spatio-temporelle (spatial_temporal_weight_code) – Structure codée sur 2 bits qui indique, dans le cas de l'échelonnabilité spatiale, la façon dont les prédictions spatiale et temporelle doivent se combiner pour constituer la prédiction relative au macrobloc. Le 7.7 décrit en détail la façon de former la prédiction pour l'échelonnabilité spatiale.

type de prédiction sur l'image (ou bi-trame) (frame_motion_type) – Structure codée sur 2 bits qui indique le type de prédiction, ou terme de compensation de mouvement du macrobloc selon le Tableau 6-17.

Tableau 6-17 – Signification de la structure `frame_motion_type`

| Code | <code>spatial_temporal_weight_class</code> | Type de prédiction | <code>motion_vector_count</code> | <code>mv_format</code> | <code>dmv</code> |
|------|--|-----------------------------|----------------------------------|------------------------|------------------|
| 00 | | Valeur réservée | | | |
| 01 | 0, 1 | Monotrème | 2 | Trème | 0 |
| 01 | 2, 3 | Monotrème | 1 | Trème | 0 |
| 10 | 0, 1, 2, 3 | Bi-trème | 1 | Bi-trème | 0 |
| 11 | 0, 2, 3 | Anticipée à double polarité | 1 | Trème | 1 |

Si le fanion `frame_pred_frame_dct` est égal à 1, le type de mouvement de bi-trème est omis du flux binaire. Dans ce cas, le décodage des vecteurs de mouvement et la formation de la prédiction doivent être effectués comme si le type de mouvement de bi-trème était «prédiction sur bi-trèmes».

Dans le cas de macroblocs intras, (dans une image bi-trème), avec une structure `concealment_motion_vectors` de valeur 1, la structure `frame_motion_type` n'est pas présente dans le flux binaire. Le décodage des vecteurs de mouvement et l'actualisation des prédicteurs de vecteurs de mouvement doivent alors être effectués comme si le type de mouvement de bi-trème avait indiqué une prédiction dite «bi-trèmes» (voir 7.6.3.9).

type de prédiction sur la trame (field_motion_type) – Structure codée sur deux bits qui indique le type de prédiction selon le Tableau 6-18.

Tableau 6-18 – Signification de la structure field_motion_type

| Code | spatial_temporal_weight_class | Type de prédiction | motion_vector_count | mv_format | dmv |
|------|-------------------------------|-----------------------------|---------------------|-----------|-----|
| 00 | | Valeur réservée | | | |
| 01 | 0, 1 | Monotrame | 1 | Trame | 0 |
| 10 | 0, 1 | 16 × 8 MC | 2 | Trame | 0 |
| 11 | 0 | Anticipée à double polarité | 1 | Trame | 1 |

Dans le cas de macroblocs intras (dans une image monotrame), avec une structure concealment_motion_vectors de valeur 1, la structure field_motion_type est absente du flux binaire. Le décodage des vecteurs de mouvement et l'actualisation des prédicteurs de vecteurs de mouvement doivent alors être effectués comme si le type de mouvement de trame avait indiqué une prédiction dite «monotrame» (voir 7.6.3.9).

type de codage DCT (dct_type) – Ce fanion indique si le macrobloc est à codage DCT sur bi-frames ou à codage DCT sur trames. S'il est mis à '1', le macrobloc est à codage DCT sur trames.

Si le fanion dct_type est absent du flux binaire, la valeur de dct_type (utilisée dans le reste du processus de décodage) doit être calculée comme indiqué dans le Tableau 6-19.

Tableau 6-19 – Valeur du fanion dct_type non présent dans le flux binaire

| Condition | Fanion dct_type |
|---|---|
| picture_structure == «trame» | Champ non utilisé parce qu'une image monotrame ne comporte pas la distinction entre bi-frames et trames |
| frame_pred_frame_dct == 1 | 0 («bi-trame») |
| !(macroblock_intra macroblock_pattern) | Champ non utilisé – Macrobloc non codé |
| macrobloc sauté | Champ non utilisé – Macrobloc non codé |

6.3.17.2 Vecteurs de mouvement

La structure motion_vector_count est déduite des structures field_motion_type ou frame_motion_type comme indiqué dans les Tableaux 6-17 et 6-18.

La structure mv_format est déduite des structures field_motion_type ou frame_motion_type comme indiqué dans les Tableaux 6-17 et 6-18. La structure mv_format indique si le vecteur de mouvement est de type trame ou bi-trame. Elle est utilisée dans la syntaxe des vecteurs de mouvement et dans le processus de prédiction des vecteurs de mouvement.

La structure dmv est déduite des structures field_motion_type ou frame_motion_type comme indiqué dans les Tableaux 6-17 et 6-18.

sélection de trame pour la prédiction (motion_vertical_field_select[r][s]) – Ce fanion indique la trame de référence qui doit être utilisée pour former la prédiction. Si ce fanion a la valeur 0, c'est la trame supérieure de référence qui doit être utilisée; s'il a la valeur 1, c'est la trame inférieure de référence. (Voir le Tableau 7-7 pour la signification des indices r et s.)

6.3.17.3 Vecteur de mouvement

code de mouvement (motion_code[r][s][t]) – Code de longueur variable selon la définition indiquée au Tableau B.10, utilisé lors du décodage d'un vecteur de mouvement, comme décrit au 7.6.3.1 (voir au Tableau 7-7 la signification des indices r, s et t).

résidu du mouvement (motion_residual[r][s][t]) – Entier utilisé lors du décodage d'un vecteur de mouvement, comme décrit au 7.6.3.1. (Voir le Tableau 7-7 pour la signification des indices r, s et t.) Le nombre de bits dans le flux binaire pour la structure motion_residual[r][s][t], défini par la structure r_size, est calculé comme suit à partir de la structure f_code[s][t]:

$$r_size = f_code[s][t] - 1$$

NOTE – Le nombre de bits pour les deux structures motion_residual[0][s][t] et motion_residual[1][s][t] est indiqué par la structure f_code[s][t].

vecteur de mouvement différentiel (dmvector[t]) – Code à longueur variable, selon la définition indiquée au Tableau B.11, utilisé lors du décodage d'un vecteur de mouvement, comme décrit au 7.6.3.1 (voir au Tableau 7-7 la signification de l'indice t).

6.3.17.4 Structure de bloc codé

structure de bloc codé en 4:2:0 (coded_block_pattern_420) – Code à longueur variable qui est utilisé pour calculer la variable cbp (structure de bloc codé ou coded_block_pattern) conformément au Tableau B.9.

structure de bloc codé en 1 (coded_block_pattern_1)

structure de bloc codé en 2 (coded_block_pattern_2) – Pour les données de format 4:2:2 et 4:4:4, la structure de bloc codé est étendue par l'adjonction d'un code de longueur fixe (2 ou 6 bits) dénommé coded_block_pattern_1 ou coded_block_pattern_2. La valeur du code de structure pattern_code[i] est ensuite calculée selon l'algorithme suivant:

```

for (i = 0; i < 12; i++) {
    if (macroblock_intra)
        pattern_code[i] = 1;
    else
        pattern_code[i] = 0;
}
if (macroblock_pattern) {
    for (i = 0; i < 6; i++)
        if (cbp & (1 << (5 - i))) pattern_code[i] = 1;
    if (chroma_format == "4:2:2")
        for (i = 6; i < 8; i++)
            if (coded_block_pattern_1 & (1 << (7 - i))) pattern_code[i] = 1;
    if (chroma_format == "4:4:4")
        for (i = 8; i < 12; i++)
            if (coded_block_pattern_2 & (1 << (11 - i))) pattern_code[i] = 1;
}

```

Si la structure pattern_code[i] a la valeur 1 pour i = 0 jusqu'à (block_count-1), le numéro de bloc i, défini dans les Figures 6-8, 6-9 et 6-10, est contenu dans ce macrobloc.

La valeur de la structure «block_count», qui détermine le nombre de blocs contenus dans le macrobloc, est déduite du format de chrominance comme indiqué dans le Tableau 6-20.

Tableau 6-20 – Valeur de la structure block_count en fonction du format de chrominance

| chroma_format | block_count |
|---------------|-------------|
| 4:2:0 | 6 |
| 4:2:2 | 8 |
| 4:4:4 | 12 |

6.3.18 Bloc

La sémantique de la structure block() est décrite dans le paragraphe 7 suivant.

7 Processus de décodage des données vidéo

Cet article spécifie le processus de décodage qui doit être effectué par un décodeur pour reconstituer les données d'image à partir du flux binaire codé.

A l'exception de la DCT inverse (IDCT), le processus de décodage est défini de manière que tous les décodeurs produisent des résultats numériquement identiques. Tout processus de décodage produisant des résultats identiques à ceux du processus ici décrit est, par définition, conforme à la présente Spécification.

La IDCT est définie en termes statistiques de façon à permettre différentes mises en œuvre de cette fonction. La spécification de la transformation IDCT est donnée en Annexe A.

Les paragraphes 7.1 à 7.6 spécifient le plus simple processus de décodage, qui n'utilise aucun mode d'échelonnabilité. Les paragraphes 7.7 à 7.11 spécifient le processus de décodage lorsque des extensions à l'échelonnabilité sont utilisées. Le 7.12 définit le signal de sortie du processus de décodage.

La Figure 7-1 montre sous forme de diagramme le processus de décodage des données vidéo sans aucun changement d'échelle. Ce diagramme a été simplifié pour plus de clarté.

NOTE – Dans l'ensemble de la présente Spécification, deux tables dimensionnelles sont représentées sous l'appellation $name[q][p]$ où 'q' est l'index dans la dimension verticale et 'p' l'index dans la dimension horizontale.

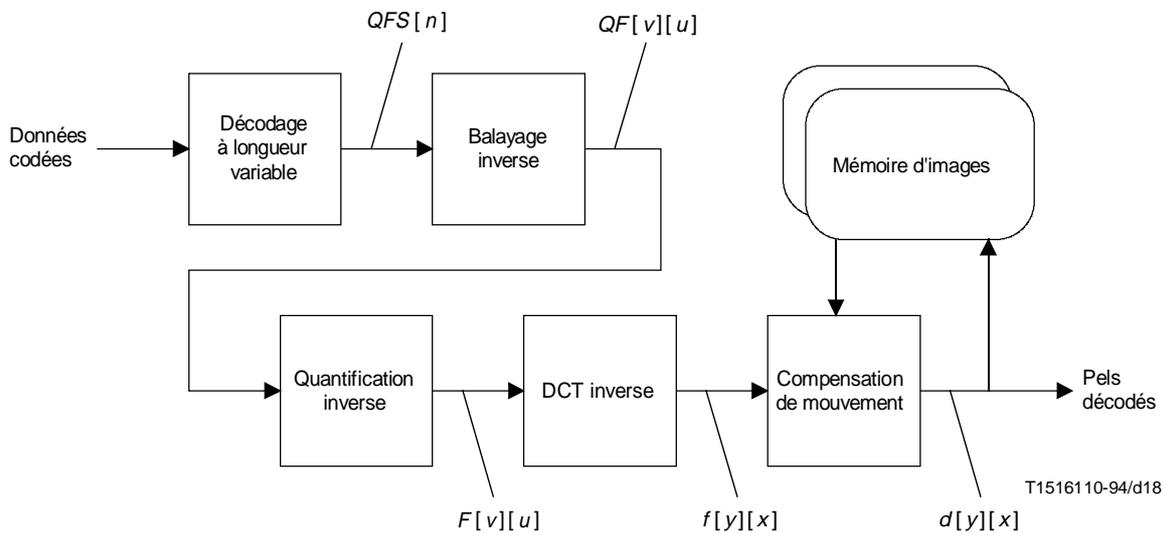


Figure 7-1 – Processus simplifié de décodage des données vidéo

7.1 Structures syntaxiques supérieures

Les divers paramètres et fanions contenus dans le flux binaire pour la structure macroblock() et pour toutes les structures supérieures à cette dernière doivent être interprétés comme indiqué au paragraphe 6. Un grand nombre de ces paramètres et fanions ont une incidence sur le processus de décodage décrit dans les paragraphes suivants. Une fois que tous les macroblocs d'une image donnée ont été traités, l'image est reconstruite entièrement.

Les images monotrames reconstruites doivent être appariées pour former des bi-trames (ou images complètes) reconstruites. (Voir la structure <picture_structure> au paragraphe 6.3.10.)

La séquence des images complètes reconstruites doit être réordonnée comme décrit au 6.1.1.11.

Si $progressive_sequence = 1$, les bi-trames reconstruites doivent être émises par le processus de décodage selon les intervalles réguliers de la période d'image, comme indiqué sur la Figure 7-19.

Si `progressive_sequence == 0`, les bi-frames reconstruites doivent être subdivisées en une séquence de trames qui doivent être émises par le processus de décodage selon les intervalles réguliers de la période de trame, comme indiqué sur la Figure 7-20. Si une image bi-trame possède une structure `repeat_first_field == 1`, la première trame de cette image doit être répétée après la seconde. (Voir la structure «`repeat_first_field`» au 6.3.10.)

7.2 Décodage à longueur variable

Le paragraphe 7.2.1 spécifie le processus de décodage utilisé pour les coefficients DC ($n = 0$) dans un bloc codé en intra. (n est l'index du coefficient dans l'ordre de balayage en zigzag approprié.) Le paragraphe 7.2.2 spécifie le processus de décodage pour tous les autres coefficients: les coefficients AC ($n > 0$) et tous les coefficients DC des blocs codés non intras.

Soit cc la composante chromatique, liée au numéro de bloc comme spécifié dans le Tableau 7-1. La composante cc est égale à 0 pour la composante de luminance Y, à 1 pour la composante Cb et à 2 pour la composante Cr.

Tableau 7-1 – Définition de l'index de composante chromatique, cc

| Numéro de bloc | cc | | |
|----------------|-------|-------|-------|
| | 4:2:0 | 4:2:2 | 4:4:4 |
| 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 |
| 4 | 1 | 1 | 1 |
| 5 | 2 | 2 | 2 |
| 6 | | 1 | 1 |
| 7 | | 2 | 2 |
| 8 | | | 1 |
| 9 | | | 2 |
| 10 | | | 1 |
| 11 | | | 2 |

7.2.1 Coefficients DC intra

Dans les blocs à codage intra, les coefficients DC sont codés en mots de longueur variable définissant la structure `dct_dc_size` conformément aux Tableaux B.12 et B.13. Si la structure `dct_dc_size` n'a pas la valeur zéro, elle doit être suivie d'un mot de code de longueur fixe représenté par la structure `dc_dct_differential`, contenant le nombre de bits indiqué par `dct_dc_size`. Une valeur différentielle est d'abord extraite des données codées puis est ajoutée à un prédicteur afin de retrouver le coefficient décodé final.

Si la variable cc est égale à 0, le Tableau B.12 doit être utilisé pour définir la structure `dct_dc_size`. Si elle n'est pas égale à 0, c'est le Tableau B.13 qui doit être utilisé pour la structure `dct_dc_size`.

Trois prédicteurs sont tenus à jour: un pour chacune des composantes chromatiques (définies par la variable cc). Chaque fois qu'un coefficient DC intra est décodé, le prédicteur correspondant est ajouté à la valeur différentielle pour retrouver le coefficient réel. Puis le prédicteur est mis à la valeur du coefficient qui vient d'être décodé. A divers moments, comme décrit ci-dessous, les prédicteurs doivent être réinitialisés. La valeur de réinitialisation est déduite du paramètre `intra_dc_precision` comme spécifié au Tableau 7-2.

Tableau 7-2 – Relation entre le paramètre `intra_dc_precision` et la valeur de réinitialisation du prédicteur

| <code>intra_dc_precision</code> | Bits de précision | Valeur de réinitialisation |
|---------------------------------|-------------------|----------------------------|
| 0 | 8 | 128 |
| 1 | 9 | 256 |
| 2 | 10 | 512 |
| 3 | 11 | 1024 |

Les prédicteurs doivent être réinitialisés aux instants suivants:

- en début de tranche;
- à chaque décodage d'un macrobloc non intra;
- chaque fois qu'un macrobloc est sauté, c'est-à-dire lorsque `macroblock_address_increment > 1`.

Les prédicteurs sont représentés par la structure `dc_dct_pred[cc]`.

La donnée décodée pour le premier coefficient, `QFS[0]`, doit être calculée à partir des structures `dc_dct_size` et `dc_dct_differential` par tout procédé équivalant à l'algorithme suivant:

```

if ( dc_dct_size == 0 ) {
    dct_diff = 0;
} else {
    half_range = 2 ^ ( dc_dct_size - 1 );
    if ( dc_dct_differential >= half_range )
        dct_diff = dc_dct_differential;
    else
        dct_diff = (dc_dct_differential + 1) - (2 * half_range);
}
QFS[0] = dc_dct_pred[cc] + dct_diff;
dc_dct_pred[cc] = QFS[0]

```

NOTE 1 – Le symbole $^$ indique une puissance (et non un OU exclusif).

NOTE 2 – Les structures `dct_diff` et `half_range` sont des variables temporaires qui ne sont pas utilisées ailleurs dans la présente Spécification.

Dans un flux binaire conforme, la valeur `QFS[0]` doit se trouver dans l'étendue suivante:

de 0 à $((2^{(8 + \text{intra_dc_precision})}) - 1)$

7.2.2 Autres coefficients

Tous les coefficients, sauf les coefficients DC intras, doivent être codés conformément aux Tableaux B.14, B.15 et B.16.

Dans tous les cas, un mot de longueur variable doit d'abord être décodé conformément au Tableau B.14 ou au Tableau B.15. La valeur décodée de ce mot indique une des trois situations suivantes:

- 1) *Fin de bloc* – Dans ce cas, il n'y a plus de coefficients dans le bloc et le reste des coefficients du bloc (pour lesquels aucune valeur n'a encore été décodée) doit être mis à zéro. Cette situation est dénommée «fin de bloc» dans la spécification syntaxique du 6.2.6.

- 2) Un coefficient «normal», dans lequel une valeur d'exécution (*run*) et de niveau (*level*) est décodée, suivi d'un bit unique, *s*, donnant le signe du coefficient *signed_level*, est calculé à partir des valeurs *level* et *s* comme indiqué ci-dessous. Les coefficients de type *run* doivent être mis à zéro et le coefficient suivant doit avoir la valeur *signed_level*:

```

if (s = 0)
    signed_level = level;
else
    signed_level = (-level);

```

- 3) Un coefficient d'échappement codé, dans lequel les valeurs *run* et *signed_level* sont des codes de longueur fixe comme décrit au 7.2.2.3.

7.2.2.1 Sélection de table

Le Tableau 7-3 indique la table à utiliser pour le décodage des coefficients DCT.

Tableau 7-3 – Sélection des tables de décodage VLC des coefficients DCT

| intra_vlc_format | 0 | 1 |
|--|------|------|
| Blocs à codage intra (macroblock_intra = 1) | B.14 | B.15 |
| Blocs à codage non intra (macroblock_intra = 0) | B.14 | B.14 |

7.2.2.2 Premier coefficient d'un bloc à codage non intra

Dans le cas du premier coefficient d'un bloc à codage non intra (dans un macrobloc de type à codage non intra), le Tableau B.14 est modifié comme indiqué par les Notes 2 et 3 situées au bas de ce tableau.

Cette modification n'a d'incidence que sur l'entrée qui représente $run = 0$ et $level = \pm 1$. Comme il n'est pas possible de coder une fin de bloc en tant que premier coefficient d'un bloc (car dans ce cas le bloc serait de type «non codé»), il n'y a pas de possibilité d'ambiguïté.

Les éléments syntaxiques qui utilisent ce Tableau B.14 modifié sont dénommés «premier coefficient DCT» dans la spécification syntaxique du 6.2.6. Les autres coefficients sont dénommés «coefficients DCT suivants».

NOTE – Si le Tableau B.4 est utilisé pour un bloc à codage intra, le premier coefficient doit être codé comme spécifié dans le 7.2.1. Le Tableau B.14 ne doit donc pas être modifié car le premier coefficient qui utilise le Tableau B.14 est le deuxième coefficient du bloc.

7.2.2.3 Codage par échappement

De nombreuses combinaisons possibles de type d'exécution et de niveau ne possèdent pas de code de longueur variable pour les représenter. Pour coder ces statistiquement rares combinaisons, une méthode de codage par échappement est utilisée.

Le Tableau B.16 définit la méthode de codage par échappement. Le mot de code VLC de l'échappement est suivi d'un code de longueur fixe (6 bits) donnant la structure «*run*». Celle-ci est suivie d'un code de longueur fixe (12 bits) donnant les valeurs de la structure «*signed_level*».

NOTE – L'attention est appelée sur le fait que la méthode de codage par échappement utilisée dans la présente Spécification est différente de celle qui est utilisée dans ISO/CEI 11172-2.

7.2.2.4 Résumé

En résumé du 7.2.2, le processus de décodage des mots à longueur variable doit être équivalent à ce qui suit. Au début de ce processus, la variable n doit prendre la valeur zéro pour les blocs non intras et la valeur 1 pour les blocs intras.

```

eob_not_read = 1;
while ( eob_not_read )
{
    <decodage VLC et, au besoin décodage de coefficient codé par échappement >
    if ( <le décodage VLC indique une fin de bloc (eob)> ) {
        eob_not_read = 0;
        while ( n < 64 ) {
            QFS[n] = 0;
            n = n + 1;
        }
    } else {
        for ( m = 0; m < run; m++ ) {
            QFS[n] = 0;
            n = n + 1;
        }
        QFS[n] = signed_level
        n = n + 1;
    }
}

```

NOTE – Les structures eob_not_read et m sont des variables temporaires qui ne doivent pas être utilisées ailleurs dans la présente Spécification.

7.3 Balayage inverse des coefficients

Soit $QFS[n]$ la donnée en sortie du décodeur de mots à longueur variable, où la variable n est comprise entre 0 et 63.

Ce paragraphe spécifie la façon dont cette donnée unidimensionnelle, $QFS[n]$, est convertie en une table à deux entrées de coefficients repérés par l'index $QFS[v][u]$ où les variables u et v sont toutes les deux comprises entre 0 et 7.

Deux modes de balayage des coefficients sont définis. Le mode de balayage à utiliser doit être déterminé par la structure $alternate_scan$ (variante de balayage), qui est codée dans l'extension d'en-tête d'image.

La Figure 7-2 définit la structure composite $scan[alternate_scan][v][u]$ pour le cas où $alternate_scan$ a la valeur zéro. La Figure 7-3 définit la structure composite $scan[alternate_scan][v][u]$ pour le cas où $alternate_scan$ a la valeur un.

Le balayage inverse peut être effectué conformément à tout algorithme équivalent à ce qui suit:

```

for ( v = 0; v < 8; v++ )
    for ( u = 0; u < 8; u++ )
        QF[v][u] = QFS[scan[alternate_scan][v][u]]

```

NOTE – Les variantes de balayage définies ici sont souvent désignées par l'expression «ordre de balayage en zigzag».

7.3.1 Balayage inverse pour téléchargement de matrice

Lorsque les matrices de quantification sont téléchargées, elles sont codées dans le flux binaire selon un ordre de balayage qui est converti en matrice à deux dimensions pour le quantificateur inverse, exactement comme pour les coefficients DCT.

Pour le téléchargement de matrices, la variante de balayage définie par la Figure 7-2 (c'est-à-dire de type $scan [0][v][u]$) doit toujours être utilisée.

Soit $W[w][v][u]$ la matrice de pondération dans le quantificateur inverse (voir 7.4.2.1) et $W'[w][n]$ la matrice telle qu'elle est codée dans le flux binaire. La matrice de téléchargement devra donc être équivalente à ce qui suit:

```

for ( v = 0; v < 8; v++ )
    for ( u = 0; u < 8; u++ )
        W[w][v][u] = W'[w][scan[0][v][u]]

```

| | | <i>u</i> | | | | | | | |
|------------|---|----------|----|----|----|----|----|----|----|
| | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 | 0 | 0 | 1 | 5 | 6 | 14 | 15 | 27 | 28 |
| 1 | 1 | 2 | 4 | 7 | 13 | 16 | 26 | 29 | 42 |
| 2 | 2 | 3 | 8 | 12 | 17 | 25 | 30 | 41 | 43 |
| 3 | 3 | 9 | 11 | 18 | 24 | 31 | 40 | 44 | 53 |
| 4 | 4 | 10 | 19 | 23 | 32 | 39 | 45 | 52 | 54 |
| 5 | 5 | 20 | 22 | 33 | 38 | 46 | 51 | 55 | 60 |
| 6 | 6 | 21 | 34 | 37 | 47 | 50 | 56 | 59 | 61 |
| <i>v</i> 7 | 7 | 35 | 36 | 48 | 49 | 57 | 58 | 62 | 63 |

Figure 7-2 – Définition de la structure $scan[0][v][u]$

| | | <i>u</i> | | | | | | | |
|------------|---|----------|----|----|----|----|----|----|----|
| | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 | 0 | 0 | 4 | 6 | 20 | 22 | 36 | 38 | 52 |
| 1 | 1 | 1 | 5 | 7 | 21 | 23 | 37 | 39 | 53 |
| 2 | 2 | 2 | 8 | 19 | 24 | 34 | 40 | 50 | 54 |
| 3 | 3 | 3 | 9 | 18 | 25 | 35 | 41 | 51 | 55 |
| 4 | 4 | 10 | 17 | 26 | 30 | 42 | 46 | 56 | 60 |
| 5 | 5 | 11 | 16 | 27 | 31 | 43 | 47 | 57 | 61 |
| 6 | 6 | 12 | 15 | 28 | 32 | 44 | 48 | 58 | 62 |
| <i>v</i> 7 | 7 | 13 | 14 | 29 | 33 | 45 | 49 | 59 | 63 |

Figure 7-3 – Définition de la structure $scan[1][v][u]$

7.4 Quantification inverse

La table à deux entrées des coefficients, $QF[v][u]$, subit une quantification inverse pour produire les coefficients DCT reconstruits. Ce processus consiste essentiellement à effectuer une multiplication des données par le pas de quantification. Ce pas est modifié par deux mécanismes: une matrice de pondération est utilisée pour modifier la valeur du pas de quantification à l'intérieur d'un bloc et un facteur d'échelle sert à modifier le pas au prix de quelques bits (au lieu de coder toute une matrice de pondération à chaque fois).

La Figure 7-4 montre le processus global de quantification inverse. A l'issue de la phase arithmétique de quantification inverse appropriée, les coefficients obtenus, $F'[v][u]$, sont saturés afin de donner les valeurs $F[v][u]$. Ensuite, une opération de contrôle d'adaptation est effectuée pour fournir les coefficients DCT finalement reconstruits, $F[v][u]$.

NOTE – L'attention est attirée sur le fait que la méthode d'exécution du contrôle d'adaptation utilisée dans la présente Spécification est différente de celle qui est employée par ISO/CEI 11172-2.

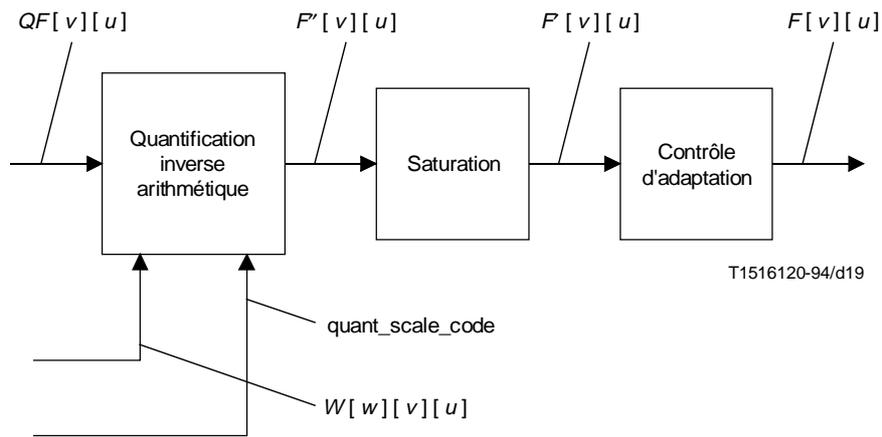


Figure 7-4 – Processus de quantification inverse

7.4.1 Coefficients DC des blocs à codage intra

Les coefficients DC des blocs à codage intra doivent subir une quantification inverse selon une méthode différente de tous les autres coefficients.

Dans les blocs à codage intra, les valeurs $F''[0][0]$ doivent être obtenues par multiplication des valeurs $QF[0][0]$ par un facteur constant: *intra_dc_mult* (constant dans le sens que ce facteur n'est modifié ni par la matrice de pondération ni par le facteur d'échelle). Ce multiplicateur est en relation avec le paramètre *intra_dc_precision* qui est codé dans l'extension de codage d'image. Le Tableau 7-4 spécifie la relation entre la structure *intra_dc_precision* et le multiplicateur *intra_dc_mult*.

On a donc: $F''[0][0] = \text{intra_dc_mult} \times QF[0][0]$

Tableau 7-4 – Relation entre la structure *intra_dc_precision* et le multiplicateur *intra_dc_mult*

| <i>intra_dc_precision</i> | Bits de précision | <i>intra_dc_mult</i> |
|---------------------------|-------------------|----------------------|
| 0 | 8 | 8 |
| 1 | 9 | 4 |
| 2 | 10 | 2 |
| 3 | 11 | 1 |

7.4.2 Autres coefficients

Tous les coefficients autres que le coefficient DC d'un bloc à codage intra doivent subir une quantification inverse comme spécifié dans ce paragraphe.

7.4.2.1 Matrices de pondération

Lorsque des données de format 4:2:0 sont utilisées, deux matrices de pondération sont appliquées. L'une doit être utilisée pour les macroblocs à codage intra et l'autre pour les macroblocs à codage non intra. Lorsque des données relatives aux formats 4:2:2 ou 4:4:4 sont utilisées, quatre matrices sont utilisées, ce qui permet d'avoir des matrices différentes pour les données de luminance et pour les données de chrominance. Chaque matrice possède un ensemble de valeurs par défaut qui peuvent être remplacées par téléchargement d'une matrice définie par l'utilisateur, comme expliqué au 6.2.3.2.

Soit $W[w][v][u]$ une matrice de pondération, où w prend une valeur de 0 à 3 pour indiquer de quelle matrice il s'agit. Le Tableau 7-5 résume les règles régissant la sélection de la variable w .

Tableau 7-5 – Sélection de la variable w

| | 4:2:0 | | 4:2:2 et 4:4:4 | |
|--|-----------------------|-------------------------|-----------------------|-------------------------|
| | Luminance (cc = 0) | Chrominance (cc ≠ 0) | Luminance (cc = 0) | Chrominance (cc ≠ 0) |
| Blocs intra (macroblock_intra = 1) | 0 | 0 | 0 | 2 |
| Blocs non intra (macroblock_intra = 0) | 1 | 1 | 1 | 3 |

7.4.2.2 Facteur d'échelle du quantificateur

Le facteur d'échelle du quantificateur est codé sous la forme d'une structure de longueur fixe à 5 bits: `quantiser_scale_code`. Cet élément indique l'échelle de quantification (*quantiser_scale*) à appliquer dans la quantification inverse arithmétique.

La structure `q_scale_type` (codée dans l'extension de codage d'image) indique le type de correspondance qui doit s'appliquer entre le code d'échelle de quantification et l'échelle de quantification. Le Tableau 7-6 montre les deux types de correspondance qui existent entre les structures `quantiser_scale_code` et *quantiser_scale*.

7.4.2.3 Formules de reconstruction

L'équation suivante spécifie l'algorithme permettant de reconstruire la matrice $F''[v][u]$ à partir des valeurs $QF[v][u]$ (pour tous les coefficients sauf les coefficients DC des blocs à codage intra).

$$F''[v][u] = ((2 \times QF[v][u] + k) \times W[w][v][u] \times \text{quantizer_scale}) / 32$$

où:

$$k = \begin{cases} 0, & \text{blocs intras,} \\ \text{Sign}(QF[v][u]), & \text{blocs non intras} \end{cases}$$

NOTE – L'équation ci-dessus utilise l'opérateur «/» défini en 4.1

7.4.3 Saturation

Les coefficients issus de la phase de quantification inverse arithmétique sont saturés jusqu'à occuper l'étendue $[-2048; +2047]$. On a alors:

$$F'[v][u] = \begin{cases} 2047, & F''[v][u] > 2047 \\ F''[v][u], & -2048 \leq F''[v][u] \leq 2047 \\ -2048, & F''[v][u] < -2048 \end{cases}$$

Tableau 7-6 – Relation entre les structures *quantiser_scale* et *quantiser_scale_code*

| quantiser_scale_code | quantiser_scale[q_scale_type] | |
|----------------------|-------------------------------|------------------|
| | q_scale_type = 0 | q_scale_type = 1 |
| 0 | (Valeur interdite) | |
| 1 | 2 | 1 |
| 2 | 4 | 2 |
| 3 | 6 | 3 |
| 4 | 8 | 4 |
| 5 | 10 | 5 |
| 6 | 12 | 6 |
| 7 | 14 | 7 |
| 8 | 16 | 8 |
| 9 | 18 | 10 |
| 10 | 20 | 12 |
| 11 | 22 | 14 |
| 12 | 24 | 16 |
| 13 | 26 | 18 |
| 14 | 28 | 20 |
| 15 | 30 | 22 |
| 16 | 32 | 24 |
| 17 | 34 | 28 |
| 18 | 36 | 32 |
| 19 | 38 | 36 |
| 20 | 40 | 40 |
| 21 | 42 | 44 |
| 22 | 44 | 48 |
| 23 | 46 | 52 |
| 24 | 48 | 56 |
| 25 | 50 | 64 |
| 26 | 52 | 72 |
| 27 | 54 | 80 |
| 28 | 56 | 88 |
| 29 | 58 | 96 |
| 30 | 60 | 104 |
| 31 | 62 | 112 |

7.4.4 Contrôle d'adaptation

Le contrôle d'adaptation doit être effectué par tout procédé équivalant à ce qui suit. Tout d'abord, on doit faire la sommation de tous les coefficients saturés qui ont été reconstitués dans le bloc, $F'[v][u]$. Cette valeur est ensuite testée pour déterminer si elle est impaire ou paire. Si la somme est paire, une correction doit être apportée à un seul coefficient: $F[7][7]$. On a donc:

$$\text{somme} = \sum_{v=0}^{v < 8} \sum_{u=0}^{u < 8} F'[v][u]$$

$$F[v][u] = F'[v][u] \text{ pour tout } u, v \text{ excepté } u = v = 7$$

$$F[7][7] = \begin{cases} F'[7][7] & \text{si la somme est impaire} \\ \left. \begin{array}{l} F'[7][7] - 1 \text{ si } F'[7][7] \text{ est impaire} \\ F'[7][7] + 1 \text{ si } F'[7][7] \text{ est paire} \end{array} \right\} & \text{si la somme est paire} \end{cases}$$

NOTES

1 Il peut être utile de remarquer que la correction ci-dessus pour le coefficient $F[7][7]$ peut être simplement effectuée par basculement du bit le moins significatif de la représentation en complément à deux du coefficient. De même, comme on ne s'intéresse qu'à «l'imparité» ou à la «parité» de la *somme*, un OU exclusif (codé sur le seul bit le moins significatif) peut être utilisé pour calculer la «*somme*».

2 Avertissement – Des données non nulles injectées à faible débit dans le transformateur IDCT peuvent provoquer des valeurs nulles à la sortie de transformateurs IDCT conformes. Si cela se produit dans un codeur, une désadaptation peut apparaître dans certaines images avec un décodeur utilisant un transformateur IDCT conforme mais différent. Tout codeur doit résoudre ce problème, par exemple en vérifiant les valeurs de sortie de son propre transformateur IDCT. Il doit veiller à ne jamais insérer de coefficients non nuls dans le flux binaire lorsque le bloc en question reconstruit jusqu'à zéro au moyen de sa propre fonction de transformation IDCT. Si le codeur ne prend pas cette mesure, des cas peuvent se produire où des désadaptations très grandes et très visibles apparaissent entre les états du codeur et du décodeur.

7.4.5 Résumé

En résumé, le processus de quantification inverse est tout processus numériquement équivalant à ce qui suit:

```
for (v = 0; v < 8; v++) {
  for (u = 0; u < 8; u++) {
    if ( (u==0) && (v==0) && (macroblock_intra) ) {
      F'[v][u] = intra_dc_mult * QF[v][u];
    } else {
      if ( macroblock_intra ) {
        F'[v][u] = ( QF[v][u] * W[w][v][u] * quantiser_scale * 2 ) / 32;
      } else {
        F'[v][u] = ( ( ( QF[v][u] * 2 ) + Sign(QF[v][u]) ) * W[w][v][u]
          * quantiser_scale ) / 32;
      }
    }
  }
}
```

```
sum = 0;
for (v = 0; v < 8; v++) {
  for (u = 0; u < 8; u++) {
    if ( F'[v][u] > 2047 ) {
      F[v][u] = 2047;
    } else {
      if ( F'[v][u] < -2048 ) {
        F[v][u] = -2048;
      }
    }
  }
}
```

```

        } else {
            F[v][u] = F'[v][u];
        }
    }
    sum = sum + F[v][u];
    F[v][u] = F'[v][u];
}

}

if ((sum & 1) == 0) {
    if ((F[7][7] & 1) != 0) {
        F[7][7] = F'[7][7] - 1;
    } else {
        F[7][7] = F'[7][7] + 1;
    }
}
}

```

7.5 Transformation DCT inverse

Une fois les coefficients DCT, $F[v][u]$, reconstruits, la transformation DCT inverse décrite dans l'Annexe A doit être appliquée pour obtenir les valeurs transformées inverses, $f[y][x]$. Ces valeurs doivent être saturées de façon que:

$$-256 \leq f[y][x] \leq 255, \text{ pour tous les } x, y$$

7.5.1 Blocs non codés et macroblocs sautés

Dans un macrobloc qui n'est pas sauté, si la structure `pattern_code[i]` a la valeur 1 pour un bloc donné du macrobloc, la donnée relative au coefficient est insérée dans le flux binaire pour ce bloc. Celui-ci sera décodé comme spécifié dans les articles qui précèdent.

Si toutefois la structure `pattern_code[i]` a la valeur 0, ou si le macrobloc est sauté, ce bloc ne contient aucune donnée de coefficient. Tous les coefficients de domaine d'échantillon $f[y][x]$ pour un tel bloc doivent alors prendre la valeur 0.

7.6 Compensation de mouvement

Le processus de compensation de mouvement forme des prédictions fondées sur des images déjà décodées, qui sont combinées avec les données relatives aux coefficients (issues du transformateur IDCT) afin de reconstituer les échantillons finalement décodés. La Figure 7-5 montre, sous forme de diagramme, les grandes lignes de ce processus.

En général, un maximum de quatre prédictions distinctes sont formées pour chaque bloc et sont combinées ensemble pour former le bloc de prédiction finale $p[y][x]$.

Dans le cas de macroblocs à codage intra, aucune prédiction n'est formée, si bien que le bloc $p[y][x]$ aura la valeur zéro. La saturation indiquée sur la Figure 7-5 reste requise de façon à éliminer les valeurs négatives de $f[y][x]$. Les macroblocs à codage intra peuvent contenir des vecteurs de mouvement du type dénommé «vecteurs de mouvement de masquage». Aucune prédiction n'est cependant formée dans le cours normal des événements. Cette information vectorielle cinétique est destinée à être utilisée dans le cas où les erreurs du flux binaire empêchent le décodage des données relatives aux coefficients. La manière dont un décodeur doit utiliser ces informations n'est pas spécifiée ici. La seule prescription concernant ces vecteurs de mouvement est qu'ils doivent être conformes à la syntaxe correcte pour ce type de vecteurs. On peut trouver au 7.6.3.9 une description de la méthode d'utilisation de ces vecteurs de mouvement.

Si un bloc n'est pas codé, soit parce que le macrobloc entier est sauté ou parce que le bloc spécifique n'est pas codé, il n'y a pas de données relatives au coefficient. Dans ce cas, $f[y][x]$ a la valeur 0 et les échantillons décodés correspondent simplement à la prédiction finale, $p[y][x]$.

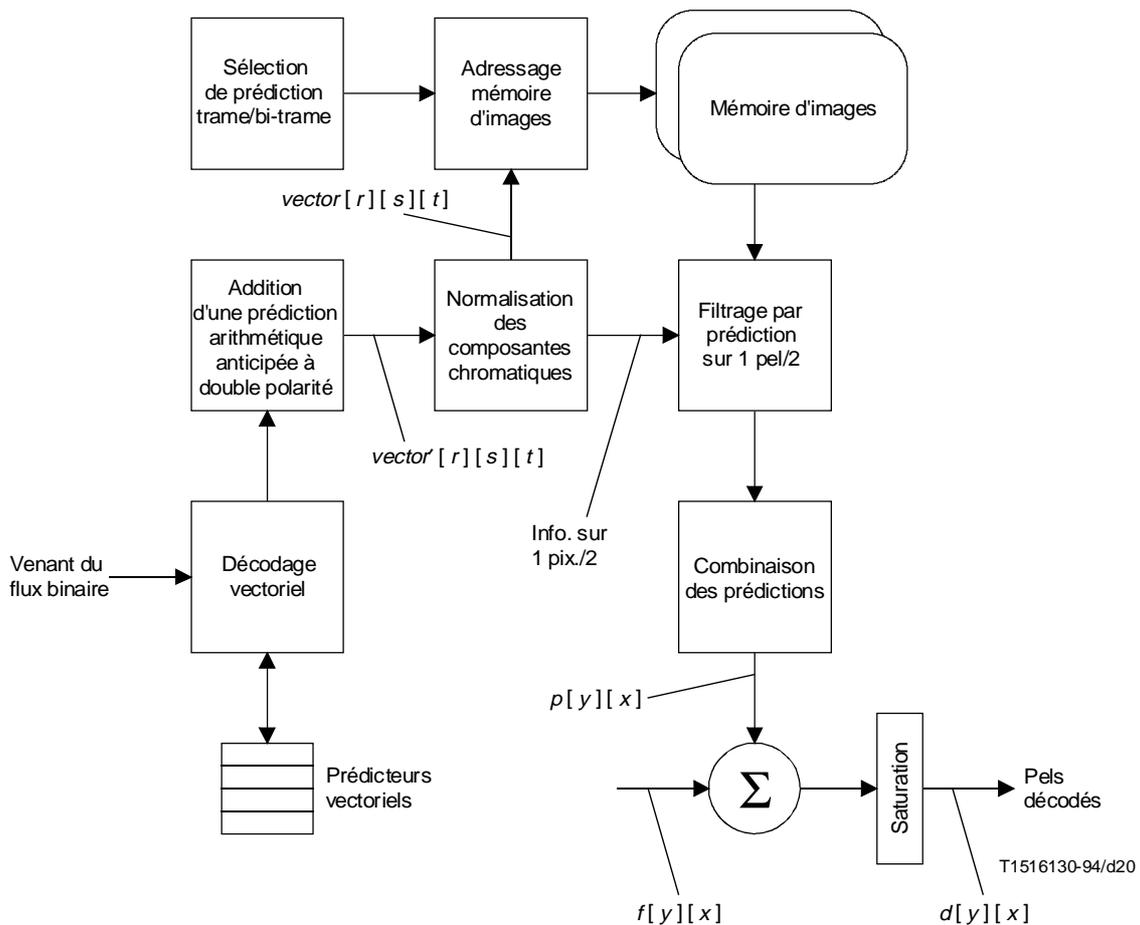


Figure 7-5 – Processus simplifié de la compensation de mouvement

7.6.1 Modes de prédiction

Il existe deux modes principaux de prédiction:

- la prédiction sur trames; et
- la prédiction sur bi-trames.

En prédiction sur trames, les prédictions sont construites indépendamment pour chaque trame au moyen de données issues d'une ou de plusieurs trames déjà décodées. En prédiction sur bi-trames, les prédictions sont construites pour chaque bi-trame au moyen d'une ou de plusieurs images bi-trames déjà décodées. Il est nécessaire de bien comprendre que les trames et bi-trames servant de références aux prédictions peuvent elles-mêmes avoir été décodées comme des images soit monotrames soit bi-trames.

A l'intérieur d'une image monotrame, toutes les prédictions se font sur trames. Dans une image bi-trame, on peut au contraire effectuer des prédictions soit sur trames soit sur bi-trames (le choix étant fait macrobloc par macrobloc).

En plus de la distinction principale entre prédictions sur trames et prédictions sur bi-trames, deux modes particuliers de prédiction sont utilisés:

- *la compensation de mouvement 16 × 8* – Compensation dans laquelle deux vecteurs de mouvement sont utilisés pour chaque macrobloc. Le premier vecteur sert pour la région 16 × 8 supérieure, le second pour la région 16 × 8 inférieure. Dans le cas d'un macrobloc à codage prédictif bidirectionnel, un total de quatre vecteurs de mouvement sera utilisé, dont deux pour la prédiction anticipée et deux pour la prédiction différée. Dans la présente Spécification, seule la compensation de mouvement de type 16 × 8 doit être utilisée pour des images monotrames;

- *la prédiction anticipée à double polarité* – Prédiction dans laquelle un seul vecteur de mouvement est codé (sous sa forme complète) dans le flux binaire, assorti d'un petit vecteur de mouvement différentiel. Dans le cas d'images monotrames, deux vecteurs de mouvement seront ensuite déduits de ces informations. Ils serviront à former des prédictions sur la base de deux trames de référence (la trame supérieure et la trame inférieure), ces deux prédictions étant intégrées pour former la prédiction finale. Dans le cas d'images bi-frames, ce processus est répété pour les deux trames de chaque image, si bien qu'un total de quatre prédictions sur trames est effectué. Ce mode ne doit être utilisé que pour des images de type P, dans lesquelles il n'y a pas d'images B entre trames ou bi-frames prédites et de référence.

7.6.2 Sélection des trames et bi-frames pour la prédiction

La sélection des trames et bi-frames à utiliser pour former des prédictions doit être faite conformément au présent paragraphe.

7.6.2.1 Prédiction sur trames

Pour des images à codage P, les deux trames de référence à partir desquelles les prédictions doivent être établies sont la trame supérieure de référence et la trame inférieure de référence les plus récemment décodées. Le cas le plus simple illustré sur la Figure 7-6 doit être suivi pour la prédiction de la première image d'une bi-trame codée ou pour l'application de la prédiction sur trame à l'intérieur d'une image bi-trame. Dans ces cas, les deux trames de référence font partie de la même image bi-trame reconstruite.

NOTES

1 Les trames de référence peuvent elles-mêmes avoir été reconstruites à partir de deux images monotrames ou d'une seule image bi-trame.

2 En cas de prédiction d'une image monotrame, la trame en cours de prédiction peut être soit une trame supérieure ou une trame inférieure.

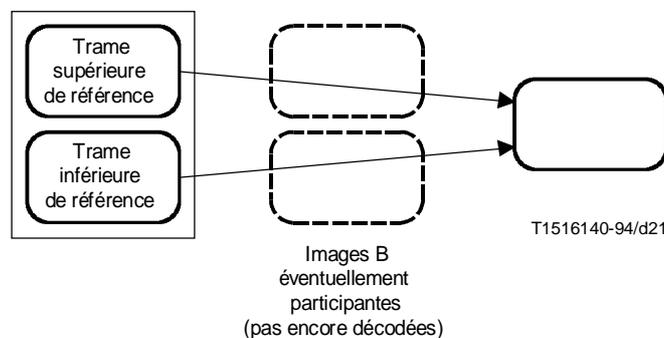


Figure 7-6 – Prédiction de la première trame ou prédiction sur une trame de bi-trame

Le cas de la prédiction effectuée sur la seconde image monotrame d'une bi-trame codée est plus compliqué parce qu'il faut utiliser les deux trames de référence décodées le plus récemment et que la plus récente trame décodée a été obtenue à partir du décodage de la première image monotrame de la bi-trame codée. La Figure 7-7 montre la situation lorsque cette deuxième image est la trame inférieure. La Figure 7-8 montre la situation lorsque cette seconde image est la trame supérieure.

NOTE – La trame de référence précédente peut elle-même avoir été reconstruite par décodage d'une image monotrame ou d'une image bi-trame.

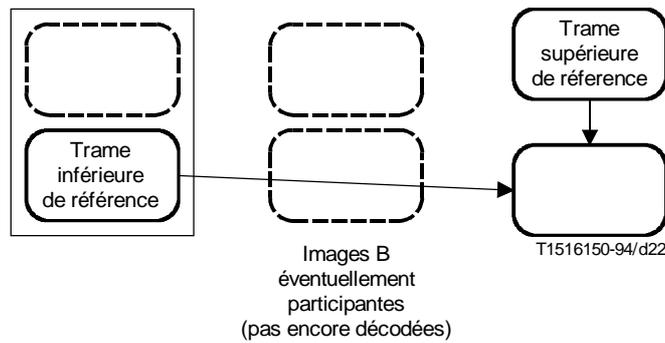


Figure 7-7 – Prédiction de la seconde image monoframe qui est une trame inférieure

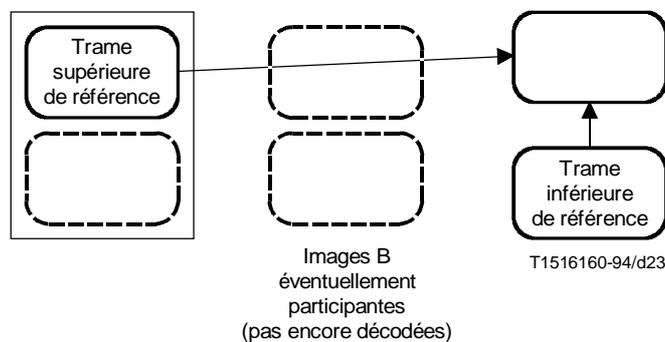


Figure 7-8 – Prédiction de la seconde image monoframe qui est une trame supérieure

La prédiction sur les trames d'images de type B doit être effectuée sur la base des deux bi-frames de référence reconstruites le plus récemment. La Figure 7-9 illustre cette situation.

NOTE – Les bi-frames de référence peuvent elles-mêmes avoir été reconstruites sur la base de deux images monoframes ou d'une seule image bi-trame.

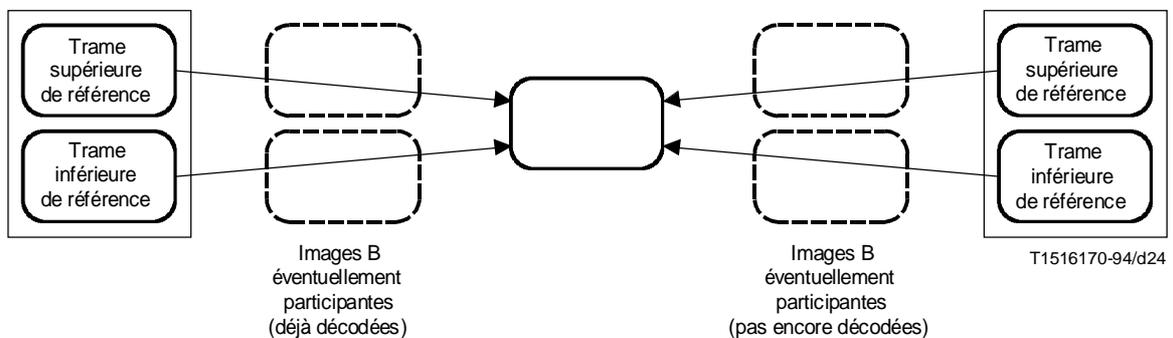


Figure 7-9 – Prédiction sur trames d'images B monoframes ou d'images B bi-frames

7.6.2.2 Prédiction sur bi-trames

Dans les images de type P, la prédiction doit être effectuée sur la base des images bi-trames de référence reconstruites le plus récemment. Cette situation est illustrée par la Figure 7-10.

NOTE 1 – L'image bi-trame de référence peut elle-même avoir été codée sous la forme de deux images monotrames ou d'une seule image bi-trame.

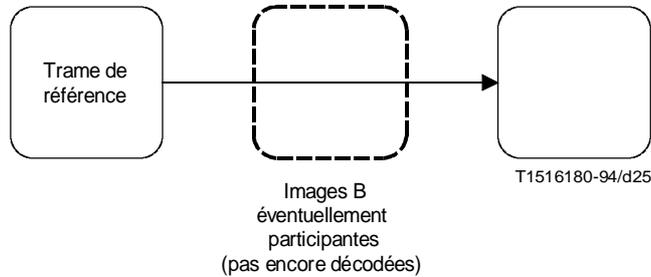


Figure 7-10 – Prédiction sur bi-trames issues d'images I et d'images P

De même, avec des images de type B, la prédiction sur bi-trames doit être effectuée sur la base des deux bi-trames de référence reconstruites le plus récemment, comme illustré par la Figure 7-11.

NOTE 2 – L'image bi-trame de référence peut elle-même avoir été codée sous la forme de deux images monotrames ou d'une seule image bi-trame.

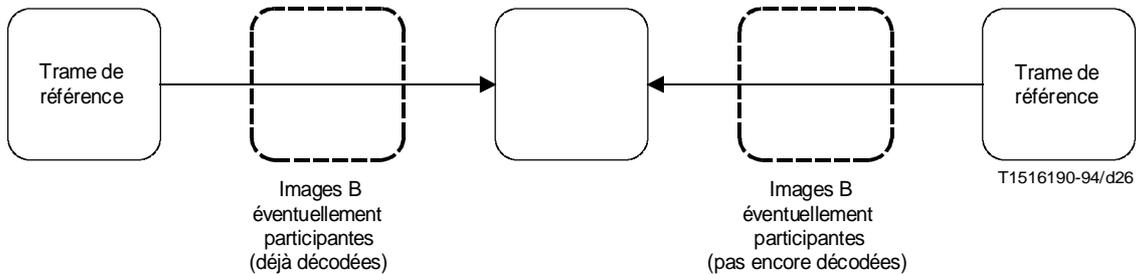


Figure 7-11 – Prédiction sur bi-trames d'images de type B

7.6.3 Vecteurs de mouvement

Les vecteurs de mouvement sont codés différemment par rapport aux vecteurs de mouvement déjà décodés, afin de réduire le nombre de bits nécessaires pour les représenter. Pour décoder les vecteurs de mouvement, le décodeur doit tenir à jour quatre prédicteurs de vecteur de mouvement (ayant chacun une composante horizontale et une composante verticale), dénommés $PMV[r][s][t]$. Pour chaque prédiction, on calcule d'abord un vecteur de mouvement, $vector'[r][s][t]$. Ce vecteur est ensuite normalisé en fonction de la structure d'échantillonnage (4:2:0, 4:2:2 ou 4:4:4) afin de donner un vecteur de mouvement de type $vector[r][s][t]$, pour chaque composante chromatique. Les significations associées aux dimensions de ces vecteurs sont définies sous la forme du Tableau 7-7.

Tableau 7-7 – Signification des indices dans les structures $PMV[r][s][t]$, $vector[r][s][t]$ et $vector'[r][s][t]$

| | 0 | 1 |
|-----|---|--|
| r | Premier vecteur de mouvement du macrobloc | Second vecteur de mouvement du macrobloc |
| s | Vecteur de mouvement anticipé | Vecteur de mouvement différé |
| t | Composante horizontale | Composante verticale |

NOTE – L'indice r prend également les valeurs 2 et 3 pour les vecteurs de mouvement dérivés qui sont employés en prédiction anticipée à double polarité. Comme ces vecteurs de mouvement sont calculés, ils ne font pas l'objet de prédicteurs des vecteurs de mouvement.

7.6.3.1 Décodage des vecteurs de mouvement

Chaque composante du vecteur de mouvement, $vector'[r][s][t]$, doit être calculée par tout procédé équivalant à ce qui suit. On notera que les prédicteurs des vecteurs de mouvement doivent aussi être mis à jour par ce processus.

```

r_size = f_code[s][t] - 1
f = 1 << r_size
high = ( 16 * f ) - 1;
low = ( (-16) * f );
range = ( 32 * f );

if ( (f == 1) || (motion_code[r][s][t] == 0) )
    delta = motion_code[r][s][t];
else {
    delta = ( ( Abs(motion_code[r][s][t]) - 1 ) * f ) + motion_residual[r][s][t] + 1;
    if (motion_code[r][s][t] < 0)
        delta = - delta;
}

prediction = PMV[r][s][t];
if ( (mv_format == "field") && (t == 1) && (picture_structure == "Frame picture") )
    prediction = PMV[r][s][t] DIV 2;

vector'[r][s][t] = prediction + delta;
if (vector'[r][s][t] < low)
    vector'[r][s][t] = vector'[r][s][t] + range;
if (vector'[r][s][t] > high)
    vector'[r][s][t] = vector'[r][s][t] - range;

if ( (mv_format == "field") && (t == 1) && (picture_structure == "Frame picture") )
    PMV[r][s][t] = vector'[r][s][t] * 2;
else
    PMV[r][s][t] = vector'[r][s][t];

```

Les paramètres contenus dans le flux binaire doivent être tels que le vecteur de mouvement différentiel reconstruit, $delta$, soit à l'intérieur de l'étendue $[low:high]$. En outre, le vecteur de mouvement reconstruit, $vector'[r][s][t]$, et la valeur actualisée du prédicteur de vecteur de mouvement, $PMV[r][s][t]$, doivent également se trouver dans l'étendue $[low:high]$.

Les structures r_size , f , $delta$, $high$, low et $range$ sont des variables temporaires qui ne sont pas utilisées dans la suite de la présente Spécification.

Les structures $motion_code[r][s][t]$ et $motion_residual[r][s][t]$ correspondent à des champs récupérés à partir du flux binaire; mv_format est récupéré à partir du flux binaire selon les indications des Tableaux 6-17 et 6-18.

r , s et t spécifient la composante particulière du vecteur de mouvement en cours de traitement, conformément au Tableau 7-7.

La structure $vector'[r][s][t]$ désigne le vecteur de mouvement final reconstruit pour la composante de luminance codée dans le macrobloc.

7.6.3.2 Contraintes relatives aux vecteurs de mouvement

Dans les images bi-frames, la composante verticale des vecteurs de mouvement de trame doit être limitée de manière qu'ils ne s'appliquent qu'à la moitié de l'étendue supportée par la structure f_code applicable à ces vecteurs de mouvement. Cette restriction fait en sorte que les prédicteurs de vecteur de mouvement aient toujours des valeurs appropriées au décodage des vecteurs de mouvement de bi-trame. Le Tableau 7-8 résume les étendues des vecteurs de mouvement qui peuvent être codés en fonction de f_code .

Tableau 7-8 – Etendue admissible des vecteurs de mouvement en fonction de $f_code[s][t]$

| $f_code[s][t]$ | Composantes verticales ($t == 1$) des vecteurs de trame dans des images bi-frames | Dans tous les autres cas |
|-----------------|---|--------------------------|
| 0 | (Valeur interdite) | |
| 1 | [-4: +3,5] | [-8: +7,5] |
| 2 | [-8: +7,5] | [-16: +15,5] |
| 3 | [-16: +15,5] | [-32: +31,5] |
| 4 | [-32: +31,5] | [-64: +63,5] |
| 5 | [-64: +63,5] | [-128: +127,5] |
| 6 | [-128: +127,5] | [-256: +255,5] |
| 7 | [-256: +255,5] | [-512: +511,5] |
| 8 | [-512: +511,5] | [-1024: +1023,5] |
| 9 | [-1024: +1023,5] | [-2048: +2047,5] |
| 10-14 | (Valeur réservée) | |
| 15 | (Utilisé lorsqu'un $f_code[s][t]$ déterminé ne sera pas utilisé) | |

7.6.3.3 Actualisation des prédicteurs de vecteur de mouvement

Une fois que tous les vecteurs de mouvement présents dans le macrobloc ont été décodés par le processus défini dans l'article précédent, il peut être nécessaire d'actualiser d'autres prédicteurs de vecteur de mouvement. La raison en est que certains modes de prédiction font appel à un nombre de vecteurs de mouvement inférieur au nombre maximal possible. Le reste des prédicteurs utilisables dans l'image doit toujours contenir les valeurs «sensibles» pour le cas où elles seraient utilisées par la suite.

Les prédicteurs des vecteurs de mouvement doivent être actualisés comme spécifié dans les Tableaux 7-9 et 7-10. Les règles d'actualisation des prédicteurs des vecteurs de mouvement en cas de macroblocs sautés sont spécifiées dans le 7.6.6.

NOTE – Il est possible, dans une mise en œuvre particulière, d'optimiser l'actualisation (et la réinitialisation) de prédicteurs des vecteurs de mouvement, selon le type d'image. Dans une image P par exemple, les prédicteurs des vecteurs de mouvement différés ne seront pas utilisés et n'ont pas besoin d'être tenus à jour.

7.6.3.4 Réinitialisation des prédicteurs de vecteurs de mouvement

Tous les prédicteurs de vecteurs de mouvement doivent être remis à zéro dans les cas suivants:

- en début de tranche;
- à chaque décodage d'un macrobloc à codage intra sans vecteurs de mouvement de masquage;
- à chaque décodage, dans une image P, d'un macrobloc à codage non intra contenant une structure $macroblock_motion_forward$ de valeur zéro;
- à chaque macrobloc sauté dans une image P.

Tableau 7-9 – Actualisation des prédicteurs de vecteurs de mouvement dans les images bi-frames

| Structure frame_motion_ type | Structures macroblock_motion_ | | Structure macroblock_ intra | Prédicteurs à actualiser |
|--|---|-----------------|---|--|
| | forward | backward | | |
| Prédiction sur bi-frames ^{a)} | – | – | 1 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ ^{b)} |
| Prédiction sur bi-frames | 1 | 1 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ $PMV[1][1][1:0] = PMV[0][1][1:0]$ |
| Prédiction sur bi-frames | 1 | 0 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ |
| Prédiction sur bi-frames | 0 | 1 | 0 | $PMV[1][1][1:0] = PMV[0][1][1:0]$ |
| Prédiction sur bi-frames ^{a)} | 0 | 0 | 0 | $PMV[r][s][t] = 0$ ^{c)} |
| Prédiction sur trames | 1 | 1 | 0 | (Aucun) |
| Prédiction sur trames | 1 | 0 | 0 | (Aucun) |
| Prédiction sur trames | 0 | 1 | 0 | (Aucun) |
| Prédiction anticipée à double polarité | 1 | 0 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ |
| <p>a) La structure frame_motion_type n'est pas présente dans le flux binaire mais on suppose qu'elle est fondée sur des images bi-frames.</p> <p>b) Si la structure concealment_motion_vectors a la valeur zéro, le prédicteur $PMV[r][s][t]$ est mis à zéro (pour tous les r, s et t).</p> <p>c) (Ce cas ne se produit que dans les images P) le prédicteur $PMV[r][s][t]$ est mis à zéro (pour tous les r, s et t). Voir 7.6.3.4.</p> <p>NOTE – $PMV[r][s][1:0] = PMV[u][v][1:0]$ signifie que: $PMV[r][s][1] = PMV[u][v][1]$ et $PMV[r][s][0] = PMV[u][v][0]$</p> | | | | |

7.6.3.5 Prédiction dans les images de type P

Dans les images de type P, si la structure *macroblock_motion_forward* (macrobloc à vecteurs de mouvement anticipés) a la valeur zéro et si la structure *macroblock_intra* (macrobloc à codage intra) a aussi la valeur zéro, aucun vecteur de mouvement n'est codé pour le macrobloc, bien qu'une prédiction doive être formée. Si cela se produit dans une image monotrame de type P, les règles suivantes doivent être appliquées:

- le type de prédiction doit être «monotrame»;
- le vecteur de mouvement (sur trame) doit avoir la valeur (0;0);
- les prédicteurs de vecteur de mouvement doivent être remis à zéro;
- les prédictions doivent être formées sur la base de même parité que la trame en cours de prédiction.

Si cela se produit dans une image bi-trame de type P, les règles suivantes doivent être appliquées:

- le type de prédiction doit être « bi-trame »;
- le vecteur de mouvement (sur bi-trame) doit avoir la valeur (0;0);
- les prédicteurs de vecteur de mouvement doivent être remis à zéro.

Si une image monotrame de type P est utilisée comme deuxième trame d'une bi-trame dont la première trame est une image monotrame de type I, une série de contraintes sémantiques s'appliquent. Elles ont pour objet de faire en sorte que la prédiction ne soit effectuée que sur la base de l'image monotrame de type I. Ces contraintes sont les suivantes:

- aucun macrobloc ne doit avoir été codé avec des structures *macroblock_motion_forward* et *macroblock_intra* de valeur zéro;
- la prédiction anticipée à double polarité ne doit pas être utilisée;
- on ne doit pas utiliser une prédiction sur trames dans laquelle la structure **motion_vertical_field_select** (sélection de trame pour composante verticale de mouvement) indique la même parité que la trame en cours de prédiction;
- aucun macrobloc ne doit avoir été sauté.

Tableau 7-10 – Actualisation des prédicteurs de vecteurs de mouvement dans les images monotrames

| Structure field_motion_-type | Structures macroblock_motion_- | | Structure macroblock_-intra | Prédicteurs à actualiser |
|---|---------------------------------------|----------|------------------------------------|--|
| | forward | backward | | |
| Prédiction sur trames ^{a)} | – | – | 1 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ ^{b)} |
| Prédiction sur trames | 1 | 1 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ $PMV[1][1][1:0] = PMV[0][1][1:0]$ |
| Prédiction sur trames | 1 | 0 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ |
| Prédiction sur trames | 0 | 1 | 0 | $PMV[1][1][1:0] = PMV[0][1][1:0]$ |
| Prédiction sur trames ^{a)} | 0 | 0 | 0 | $PMV[r][s][t] = 0$ ^{c)} |
| Prédiction cinétiquement compensée (16 × 8 MC) | 1 | 1 | 0 | (Aucun) |
| Prédiction cinétiquement compensée (16 × 8 MC) | 1 | 0 | 0 | (Aucun) |
| Prédiction cinétiquement compensée (16 × 8 MC) | 0 | 1 | 0 | (Aucun) |
| Prédiction anticipée à double polarité | 1 | 0 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ |
| <p>a) La structure field_motion_type n'est pas présente dans le flux binaire mais on suppose qu'elle est fondée sur des images monotrames.</p> <p>b) Si la structure concealment_motion_vectors a la valeur zéro, le prédicteur $PMV[r][s][t]$ est mis à zéro (pour tous les r, s et t).</p> <p>c) (Ce cas ne se produit que dans les images P) le prédicteur $PMV[r][s][t]$ est mis à zéro (pour tous les r, s et t). Voir 7.6.3.4.</p> <p>NOTE – $PMV[r][s][1:0] = PMV[u][v][1:0]$ signifie que: $PMV[r][s][1] = PMV[u][v][1]$ et $PMV[r][s][0] = PMV[u][v][0]$</p> | | | | |

7.6.3.6 Algorithme supplémentaire pour la prédiction anticipée à double polarité

En prédiction anticipée à double polarité, un seul vecteur de mouvement de trame ($vector'[0][0][1:0]$) doit avoir été décodé par le processus décrit ci-dessus. Il représente le vecteur de mouvement utilisé pour former des prédictions sur la base de la trame de référence (ou des trames de référence dans le cas d'une image bi-trame) ayant la même parité que la prédiction en cours de formation. Dans ce contexte, le terme «parité» sert à différencier les deux trames: la trame supérieure a la parité zéro et la trame inférieure a la parité un.

Pour former un vecteur de mouvement représentant la parité opposée ($vector'[r][0][1:0]$), le vecteur de mouvement existant est normalisé de façon à refléter la nouvelle distance temporelle entre les trames. Une correction est apportée à la composante verticale (pour refléter le décalage vertical entre les lignes de la trame supérieure et celles de la trame inférieure) puis on ajoute un petit vecteur de mouvement différentiel. Ce processus est décrit sur la Figure 7-12, qui montre le cas d'une image bi-trame.

La structure $dmvector[0]$ représente la composante horizontale du vecteur de mouvement différentiel et la structure $dmvector[1]$ représente la composante verticale. Ces deux composantes du vecteur de mouvement différentiel doivent être décodées directement, au moyen du Tableau B.11 et ne doivent prendre que les valeurs -1, 0 et +1.

La structure $m[parity_ref][parity_pred]$ est la distance de trames, mesurée entre la trame prédite et la trame de référence, comme indiqué dans le Tableau 7-11. La structure « $parity_ref$ » est la parité de la trame de référence pour laquelle on calcule le nouveau vecteur de mouvement. La structure « $parity_pred$ » représente la parité de la trame qui doit être prédite.

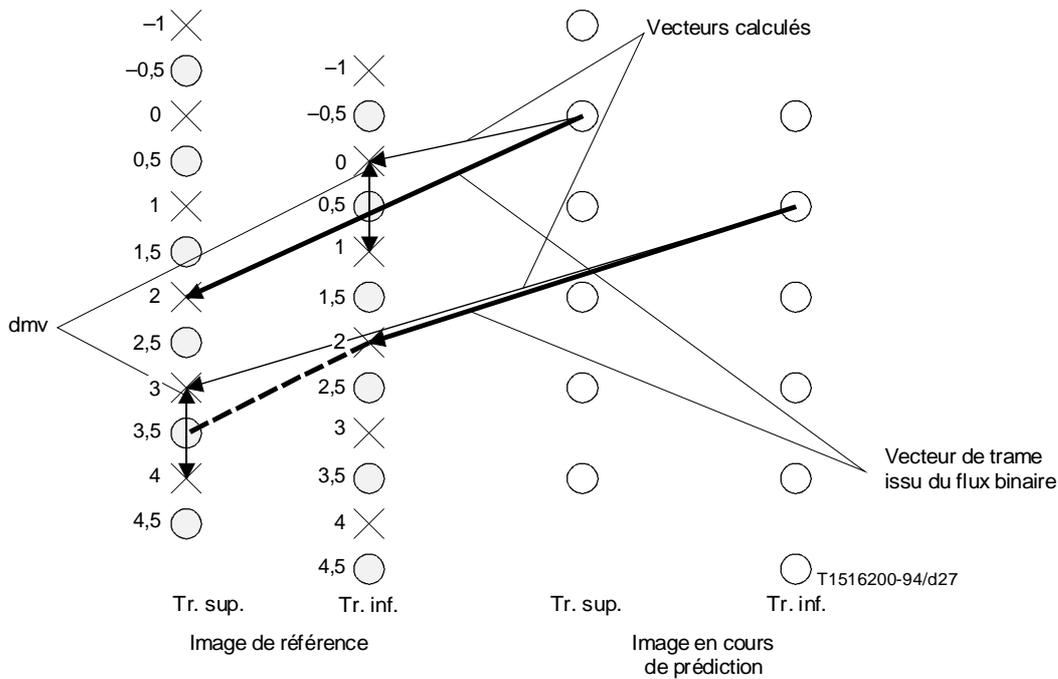


Figure 7-12 – Normalisation des vecteurs de mouvement pour la prédiction anticipée à double polarité

Tableau 7-11 – Définition de la structure $m[parity_ref][parity_pred]$

| Structure $picture_structure$ | Structure top_field_first | Structure $m[parity_ref][parity_pred]$ | |
|--------------------------------|-------------------------------|--|-----------|
| | | $m[1][0]$ | $m[0][1]$ |
| 11 (Bi-trame) | 1 | 1 | 3 |
| 11 (Bi-trame) | 0 | 3 | 1 |
| 01 (Trame supérieure) | - | 1 | - |
| 10 (Trame inférieure) | - | - | 1 |

La structure $e[parity_ref][parity_pred]$ représente l'ajustement nécessaire pour refléter le décalage vertical entre les lignes de la trame supérieure et celles de la trame inférieure, comme indiqué au Tableau 7-12.

Tableau 7-12 – Définition de la structure $e[parity_ref][parity_pred]$

| Structure $parity_ref$ | Structure $parity_pred$ | Structure $e[parity_ref][parity_pred]$ |
|-------------------------|--------------------------|--|
| 0 | 1 | +1 |
| 1 | 0 | -1 |

Le vecteur de mouvement (ou les vecteurs de mouvement) utilisé(s) pour les prédictions relatives à la parité opposée doit (doivent) être calculé(s) comme suit:

$$vector'[r][0][0] = ((vector'[0][0][0] * m[parity_ref][parity_pred])/2) + dmvector[0];$$

$$vector'[r][0][1] = ((vector'[0][0][1] * m[parity_ref][parity_pred])/2) + e[parity_ref][parity_pred] + dmvector[1];$$

Dans le cas d'images bi-trames, un seul vecteur de mouvement de ce type est nécessaire et alors la variable $r = 2$. Le vecteur de mouvement (codé) qui est utilisé pour la prédiction relative à la même parité est donc $vector'[0][0][1:0]$ et le vecteur de mouvement utilisé pour la prédiction relative à la parité opposée est donc $vector'[2][0][1:0]$.

Dans le cas d'images bi-trames, deux de ces vecteurs de mouvement sont nécessaires. Les deux trames de chaque image utilisent le vecteur de mouvement codé ($vector'[0][0][1:0]$) pour les prédictions de la même parité. La trame supérieure doit utiliser le vecteur $vector'[2][0][1:0]$ pour la prédiction de la parité opposée et la trame inférieure doit utiliser le vecteur $vector'[3][0][1:0]$ pour la prédiction de parité opposée.

7.6.3.7 Vecteurs de mouvement pour les composantes de chrominance

Les vecteurs de mouvement calculés dans les paragraphes précédents se rapportaient à la composante de luminance, dans laquelle:

$$vector[r][s][t] = vector'[r][s][t] \quad (\text{pour tous les } r, s \text{ et } t)$$

Pour chacune des deux composantes de chrominance, les vecteurs de mouvement doivent être normalisés comme suit:

- 4:2:0 Les deux composantes, horizontale et verticale, du vecteur de mouvement sont normalisées au moyen d'une division par deux:

$$vector[r][s][0] = vector'[r][s][0] / 2;$$

$$vector[r][s][1] = vector'[r][s][1] / 2;$$

- 4:2:2 La composante horizontale du vecteur de mouvement est normalisée au moyen d'une division par deux, la composante verticale n'étant pas modifiée:

$$vector[r][s][0] = vector'[r][s][0] / 2;$$

$$vector[r][s][1] = vector'[r][s][1];$$

- 4:4:4 Le vecteur de mouvement n'est pas modifié:

$$vector[r][s][0] = vector'[r][s][0];$$

$$vector[r][s][1] = vector'[r][s][1];$$

7.6.3.8 Contraintes sémantiques concernant les prédictions

Il est prescrit que la seule exigence du flux binaire concernant un décodeur soit que les prédictions doivent être effectuées sur la base de tranches déjà codées dans une image bi-trame ou monotrane de référence. Cette règle s'applique même s'il y a des macroblocs sautés et des macroblocs d'images P dans lesquels on part de l'hypothèse de l'existence d'un vecteur de mouvement de valeur zéro (comme expliqué au 7.6.3.5).

NOTE – Comme expliqué au 6.1.2, il n'est en général pas nécessaire que les tranches recouvrent la totalité de l'image. A de nombreux niveaux définis de certains profils, on fait cependant appel à la «structure de tranche restreinte»; dans ce cas, les tranches ne recouvrent pas la totalité de l'image et la règle sémantique peut être énoncée de façon plus simple: «le flux binaire est soumis à la restriction que les vecteurs de mouvement reconstruits ne doivent pas se rapporter à des échantillons extérieurs au bord de l'image codée».

7.6.3.9 Vecteurs de mouvement de masquage

Les vecteurs de mouvement de masquage sont des vecteurs de mouvement qui peuvent être acheminés par des macroblocs à codage intra, afin de masquer des erreurs lorsque des erreurs sur les données empêchent le décodage des données des coefficients. Un vecteur de mouvement de masquage doit être présent pour tous les macroblocs à codage intra si (et seulement si) la structure `concealment_motion_vectors` a la valeur 1 (dans la structure `picture_coding_extension`).

Dans le processus normal, aucune prédiction ne doit être formée pour de tels macroblocs (comme on peut s'y attendre puisque la structure `macroblock_intra` a la valeur 1). La présente Spécification ne spécifie pas la façon dont il faut réaliser la reprise sur erreur. Il est toutefois recommandé que les vecteurs de mouvement de masquage puissent être utilisés par un décodeur effectuant le masquage en formant des prédictions comme si les structures `field_motion_type` et `frame_motion_type` (qui permettent de déterminer le type de prédiction), avaient les valeurs ci-dessous:

- dans une image monotrane: `field_motion_type` = «sur trame»;
- dans une image bi-trame: `frame_motion_type` = «sur bi-trame».

NOTE – Si le masquage s'applique à une image de type I, le décodeur doit effectuer la prédiction comme pour une image de type P.

Les vecteurs de mouvement de masquage seront normalement utilisés dans le cas où une erreur de données se traduit par une perte d'information. Il n'y a donc pas grand intérêt à coder le vecteur de mouvement de masquage dans le macrobloc auquel il est censé s'appliquer étant donné que, si l'erreur de données entraîne un besoin de reprise sur erreur, il est très probable que le vecteur de mouvement de masquage lui-même sera perdu ou déformé. En conséquence, les règles sémantiques suivantes sont appropriées:

- pour tous les macroblocs à l'exception de ceux qui sont situés dans la rangée inférieure des macroblocs, il convient que les vecteurs de mouvement de masquage soient utilisables dans le macrobloc qui se trouve au-dessous (verticalement) du macrobloc dans lequel apparaît le vecteur de mouvement;
- lorsque le vecteur de mouvement est appliqué au macrobloc identifié selon la règle précédente, un décodeur doit toujours faire l'hypothèse que ce vecteur de mouvement peut se rapporter à des échantillons extérieurs aux tranches codées dans l'image bi-trame ou monotrane de référence;
- pour tous les macroblocs de la rangée inférieure des macroblocs, les vecteurs de mouvement de masquage reconstruits ne seront pas utilisés. Le vecteur de mouvement de valeur (0;0) pourra donc être utilisé pour réduire la servitude inutile.

7.6.4 Formation des prédictions

Les prédictions sont formées par lecture des pels de prédiction dans les trames ou bi-trames de référence. Un pel donné est prédit par lecture du pel correspondant dans la trame ou bi-trame de référence décalée par le vecteur de mouvement.

Une valeur positive de la composante horizontale d'un vecteur de mouvement indique que la prédiction est faite sur la base d'échantillons (ou pels) (dans la trame/bi-trame de référence) qui se trouvent à droite des échantillons en cours de prédiction.

Une valeur positive de la composante verticale d'un vecteur de mouvement indique que la prédiction est faite sur la base d'échantillons (ou pels) (dans la trame/bi-trame de référence) qui se trouvent au-dessous des pels en cours de prédiction.

Tous les vecteurs de mouvement sont spécifiés avec une précision d'un demi-pel. Si une composante du vecteur de mouvement est impaire, les pels seront donc lus à partir de la moitié de la distance entre les pels se trouvant dans la trame/bi-trame de référence. Ces demi-pels sont calculés par simple interpolation linéaire à partir des pels de référence.

Dans le cas de prédictions sur monotrane, il est nécessaire de déterminer laquelle des deux trames disponibles devra être utilisée pour former la prédiction. Dans le cas d'une prédiction anticipée à double polarité, cette information sera spécifiée par le fait qu'un vecteur de mouvement est calculé pour chacune des deux trames et qu'une prédiction est formée sur la base de chaque trame. Dans le cas d'une prédiction sur trame et d'une prédiction à compensation de mouvement (16×8), un bit additionnel (fanion *motion_vertical_field_select*) est codé pour indiquer quelle trame doit être utilisée.

Si le fanion *motion_vertical_field_select* a la valeur zéro, la prédiction est faite sur la base de la trame supérieure de référence.

Si le fanion *motion_vertical_field_select* a la valeur 1, la prédiction est faite sur la base de la trame inférieure de référence.

Pour chaque bloc de prédiction, les vecteurs de mouvement avec précision d'un pel entier (structures *int_vec[t]*) et les fanions pour précision d'un demi-pel (*half_flag[t]*) doivent être formés comme suit:

```
for (t = 0; t < 2; t++) {
    int_vec[t] = vector[r][s][t] DIV 2;
    if ((vector[r][s][t]) - (2 * int_vec[t]) != 0)
        half_flag[t] = 1;
    else
        half_flag[t] = 0;
}
```

Ensuite, chaque pel du bloc de prédiction est lu et la prédiction avec précision d'un demi-pel est appliquée comme suit:

```
if ( (! half_flag[0]) && (! half_flag[1]) )
    pel_pred[y][x] = pel_ref[y + int_vec[1]][x + int_vec[0]];
```

```

if ( (! half_flag[0]) && half_flag[1] )
    pel_pred[y][x] = ( pel_ref[y + int_vec[1]][x + int_vec[0]] +
                    pel_ref[y + int_vec[1]+1][x + int_vec[0]] ) // 2;

if ( half_flag[0] && (! half_flag[1]) )
    pel_pred[y][x] = ( pel_ref[y + int_vec[1]][x + int_vec[0]] +
                    pel_ref[y + int_vec[1]][x + int_vec[0]+1] ) // 2;

if ( half_flag[0] && half_flag[1] )
    pel_pred[y][x] = ( pel_ref[y + int_vec[1]][x + int_vec[0]] +
                    pel_ref[y + int_vec[1]][x + int_vec[0]+1] +
                    pel_ref[y + int_vec[1]+1][x + int_vec[0]] +
                    pel_ref[y + int_vec[1]+1][x + int_vec[0]+1] ) // 4;

```

où $pel_pred[y][x]$ est le pel de prédiction en cours de formation et où $pel_ref[y][x]$ représente les pels dans la trame ou bi-trame de référence.

7.6.5 Sélection du vecteur de mouvement

Le Tableau 7-13 montre les modes de prédiction utilisés dans les images monotrames et le Tableau 7-14 montre les prédictions utilisées dans les images bi-trames. Dans chaque tableau, les vecteurs de mouvement présents dans le flux binaire sont énumérés dans l'ordre de leur apparition dans ce flux.

Tableau 7-13 – Prédications et vecteurs de mouvement dans les images monotrames

| field_ motion_ type | Structures macroblock_motion_ - | | macro- block_ - intra | Vecteur de mouvement | Prédiction formée pour |
|--|------------------------------------|----------|-----------------------------|--|--|
| | forward | backward | | | |
| Prédiction sur trames ^{a)} | – | – | 1 | $vector^*[0][0][1:0]^b)$ | Aucune prédiction (le vecteur de mouvement est de type masquage) |
| Prédiction sur trames | 1 | 1 | 0 | $vector^*[0][0][1:0]$ $vector^*[0][1][1:0]$ | Trame entière, anticipée Trame entière, différée |
| Prédiction sur trames | 1 | 0 | 0 | $vector^*[0][0][1:0]$ | Trame entière, anticipée |
| Prédiction sur trames | 0 | 1 | 0 | $vector^*[0][1][1:0]$ | Trame entière, différée |
| Prédiction sur trames ^{a)} | 0 | 0 | 0 | $vector^*[0][0][1:0]^c) d)$ | Trame entière, anticipée |
| Prédiction à compensation de mouvement (16 × 8 MC) | 1 | 1 | 0 | $vector^*[0][0][1:0]$ $vector^*[1][0][1:0]$ $vector^*[0][1][1:0]$ $vector^*[1][1][1:0]$ | Trame supérieure 16 × 8, anticipée Trame inférieure 16 × 8, anticipée Trame supérieure 16 × 8, différée Trame inférieure 16 × 8, différée |
| Prédiction à compensation de mouvement (16 × 8 MC) | 1 | 0 | 0 | $vector^*[0][0][1:0]$ $vector^*[1][0][1:0]$ | Trame supérieure 16 × 8, anticipée Trame inférieure 16 × 8, anticipée |
| Prédiction à compensation de mouvement (16 × 8 MC) | 0 | 1 | 0 | $vector^*[0][1][1:0]$ $vector^*[1][1][1:0]$ | Trame supérieure 16 × 8, différée Trame inférieure 16 × 8, différée |
| Prédiction anticipée à double polarité | 1 | 0 | 0 | $vector^*[0][0][1:0]$ $vector^*[2][0][1:0]^c) e)$ | Trame entière, de même parité, anticipée Trame entière, de parité opposée, anticipée |

- a) La structure **field_motion_type** n'est pas présente dans le flux binaire mais est censée être de type codage sur trames.
- b) Ce vecteur de mouvement n'est présent que si la structure **concealment_motion_vectors** a la valeur 1.
- c) Ces vecteurs de mouvement ne sont pas présents dans le flux binaire.
- d) Ce vecteur de mouvement est censé avoir la valeur (0; 0) comme expliqué au 7.6.3.5.
- e) Ces vecteurs de mouvement sont calculés d'après $vector^*[0][0][1:0]$ comme décrit au 7.6.3.6.

NOTE – Les vecteurs de mouvement sont énumérés dans l'ordre de leur apparition dans le flux binaire.

Tableau 7-14 – Prédiction et vecteurs de mouvement dans les images bi-frames

| frame_ motion_ type | Structures macroblock_motion_ | | macro- block_ intra | Vecteur de mouvement | Prédiction formée pour |
|--|----------------------------------|----------|---------------------------|--|--|
| | forward | backward | | | |
| Prédiction sur bi-frames ^{a)} | – | – | 1 | $vector[0][0][1:0]$ ^{b)} | Aucune prédiction (le vecteur de mouvement est de type masquage) |
| Prédiction sur bi-frames | 1 | 1 | 0 | $vector[0][0][1:0]$ $vector[0][1][1:0]$ | Bi-trame, anticipée Bi-trame, différée |
| Prédiction sur bi-frames | 1 | 0 | 0 | $vector[0][0][1:0]$ | Bi-trame, anticipée |
| Prédiction sur bi-frames | 0 | 1 | 0 | $vector[0][1][1:0]$ | Bi-trame, différée |
| Prédiction sur bi-frames ^{a)} | 0 | 0 | 0 | $vector[0][0][1:0]$ ^{c) d)} | Bi-trame, anticipée |
| Prédiction sur trames | 1 | 1 | 0 | $vector[0][0][1:0]$ $vector[1][0][1:0]$ $vector[0][1][1:0]$ $vector[1][1][1:0]$ | Trame supérieure, anticipée Trame inférieure, anticipée Trame supérieure, différée Trame inférieure, différée |
| Prédiction sur trames | 1 | 0 | 0 | $vector[0][0][1:0]$ $vector[1][0][1:0]$ | Trame supérieure, anticipée Trame inférieure, anticipée |
| Prédiction sur trames | 0 | 1 | 0 | $vector[0][1][1:0]$ $vector[1][1][1:0]$ | Trame supérieure, différée Trame inférieure, différée |
| Prédiction anticipée à double polarité | 1 | 0 | 0 | $vector[0][0][1:0]$ $vector[0][0][1:0]$ $vector[2][0][1:0]$ ^{c) e)} $vector[3][0][1:0]$ ^{c) e)} | Trame supérieure, de même parité, anticipée Trame inférieure, de même parité, anticipée Trame supérieure, de parité opposée, anticipée Trame inférieure, de parité opposée, anticipée |

a) **frame_motion_type** n'est pas présente dans le flux binaire mais est censée être de type codage sur bi-frames.
b) Ce vecteur de mouvement n'est présent que si la structure **concealment_motion_vectors** a la valeur 1.
c) Ces vecteurs de mouvement ne sont pas présents dans le flux binaire.
d) Ce vecteur de mouvement est censé avoir la valeur (0; 0) comme expliqué au 7.6.3.5.
e) Ces vecteurs de mouvement sont calculés d'après $vector[0][0][1:0]$ comme décrit au 7.6.3.6.

NOTE – Les vecteurs de mouvement sont énumérés dans l'ordre de leur apparition dans le flux binaire.

7.6.6 Macroblocs sautés

Un macrobloc sauté est un macrobloc pour lequel aucune donnée n'est codée, c'est-à-dire ne fait partie d'une tranche codée. Sauf au début d'une tranche, si le nombre (adresse du macrobloc – adresse du macrobloc précédent – 1) est positif, il indique alors le nombre de macroblocs qui ont été sautés. Le décodeur doit former une prédiction concernant les macroblocs sautés et les utiliser ensuite comme valeurs finales d'échantillon décodé.

Le traitement des macroblocs sautés est différent selon qu'il s'agit d'images de type P ou d'images de type B. Le processus diffère en outre selon qu'il s'agit d'images monotrames ou bi-frames.

Il ne doit pas y avoir de macroblocs sautés dans des images de type I, sauf si:

- soit une structure `picture_spatial_scalable_extension()` suit l'en-tête de l'image actuelle;
- soit une structure `sequence_scalable_extension()` est présente dans le flux binaire et la structure `scalable_mode` a la valeur «échelonnabilité SNR».

7.6.6.1 Images monotrames de type P

- la prédiction doit être effectuée comme si la structure *field_motion_type* avait la valeur «codage sur trames»;
- la prédiction doit être effectuée sur la base de la trame de même parité que la trame en cours de prédiction;
- les prédicteurs de vecteur de mouvement doivent être remis à zéro;
- le vecteur de mouvement doit avoir la valeur zéro.

7.6.6.2 Images bi-trames de type P

- la prédiction doit être effectuée comme si la structure *frame_motion_type* avait la valeur «codage sur bi-trames»;
- les prédicteurs de vecteur de mouvement doivent être remis à zéro;
- le vecteur de mouvement doit avoir la valeur zéro.

7.6.6.3 Images monotrames de type B

- la prédiction doit être effectuée comme si la structure *field_motion_type* avait la valeur «codage sur trames»;
- la prédiction doit être effectuée sur la base de la trame de même parité que la trame en cours de prédiction;
- le sens de la prédiction (anticipée/différée/bidirectionnelle) doit être le même que pour le macrobloc précédent;
- les prédicteurs de vecteur de mouvement ne sont pas modifiés;
- les vecteurs de mouvement sont déduits des prédicteurs appropriés de vecteur de mouvement. La normalisation des vecteurs de mouvement selon les composantes de couleur doit être effectuée comme indiqué au 7.6.3.7.

7.6.6.4 Images bi-trames de type B

- la prédiction doit être effectuée comme si la structure *frame_motion_type* avait la valeur «codage sur bi-trames»;
- le sens de la prédiction (anticipée/différée/bidirectionnelle) doit être le même que pour le macrobloc précédent;
- les prédicteurs de vecteur de mouvement ne sont pas modifiés;
- les vecteurs de mouvement sont déduits des prédicteurs appropriés de vecteur de mouvement. La normalisation des vecteurs de mouvement selon les composantes de couleur doit être effectuée comme indiqué au 7.6.3.7.

7.6.7 Combinaison de prédictions

L'étape finale consiste à combiner les diverses prédictions afin de former les blocs de prédiction finale.

Il est également nécessaire d'organiser les données en blocs de type monotrame ou de type bi-trame, en vue de les ajouter directement aux coefficients décodés.

Les données de transformation sont organisées en mode trame ou en mode bi-trame, selon ce qui est spécifié par la structure *dct_type*.

7.6.7.1 Prédictions simples sur bi-trames

Dans le cas de prédictions simples sur bi-trames, le seul traitement complémentaire qui peut être requis consiste à calculer la moyenne des prédictions anticipées et différées dans les images B. Si la structure *pel_pred_forward*[y][x] est l'échantillon de pel à prédiction anticipée et si la structure *pel_pred_backward*[y][x] est l'échantillon de pel à prédiction différée correspondant, l'échantillon de prédiction finale doit être formé comme suit:

$$pel_pred[y][x] = (pel_pred_forward[y][x] + pel_pred_backward[y][x])/2$$

Les prédictions pour les formats de chrominance 4:2:0, 4:2:2 et 4:4:4 doivent avoir respectivement les dimensions suivantes: 8 échantillons sur 8 lignes, 8 échantillons sur 16 lignes et 16 échantillons sur 16 lignes.

7.6.7.2 Prédiction simple sur monotrames

Dans le cas de prédictions simples sur monotrames (c'est-à-dire ni du type à compensation de mouvement 16×8 , ni du type à double polarité), le seul traitement complémentaire qui puisse être requis consiste à calculer la moyenne des prédictions anticipées et différées dans les images de type B. Cette opération doit être effectuée tel comme indiqué dans le paragraphe précédent.

Dans le cas de prédictions simples sur monotrames dans une image bi-trame, les prédictions concernant les composantes de chrominance des formats 4:2:0, 4:2:2 et 4:4:4 doivent avoir pour chaque trame les dimensions suivantes: 8 pels sur 4 lignes, 8 pels sur 8 lignes et 16 pels sur 8 lignes.

Dans le cas de prédictions simples sur monotrames dans une image monotrame, les prédictions concernant les composantes de chrominance des formats 4:2:0, 4:2:2 et 4:4:4 doivent avoir pour chaque trame les dimensions suivantes: 8 pels sur 8 lignes, 8 pels sur 16 lignes et 16 pels sur 16 lignes.

7.6.7.3 Prédiction avec compensation de mouvement 16×8

Dans ce mode de prédiction, on forme des prédictions distinctes pour les régions supérieure et inférieure 16×8 du macrobloc.

Les prédictions pour les formats de chrominance 4:2:0, 4:2:2 et 4:4:4 doivent avoir, pour chaque région 16×8 , respectivement les dimensions suivantes: 8 pels sur 4 lignes, 8 pels sur 8 lignes et 16 pels sur 8 lignes.

7.6.7.4 Prédiction à double polarité

En mode de prédiction anticipée à double polarité, deux prédictions sont formées pour chaque trame, de façon semblable aux prédictions différées et anticipées pour les images de type B. Si la structure $pel_pred_same_parity[y][x]$ est le pel de prédiction issu de la trame de même parité et si la structure $pel_pred_opposite_parity[y][x]$ est le pel correspondant issu de la trame de parité opposée, le pel de prédiction finale doit être formé comme suit:

$$pel_pred[y][x] = (pel_pred_same_parity[y][x] + pel_pred_opposite_parity[y][x]) // 2;$$

Dans le cas d'une prédiction anticipée à double polarité dans une image bi-trame, les prédictions des composantes de chrominance de chaque trame, pour les formats 4:2:0, 4:2:2 et 4:4:4, doivent avoir respectivement les dimensions suivantes: 8 pels sur 4 lignes, 8 pels sur 8 lignes et 16 pels sur 8 lignes.

Dans le cas d'une prédiction anticipée à double polarité dans une image monotrame, les prédictions des composantes de chrominance, pour les formats 4:2:0, 4:2:2 et 4:4:4, doivent avoir respectivement les dimensions suivantes: 8 pels sur 8 lignes, 8 pels sur 16 lignes et 16 pels sur 16 lignes.

7.6.8 Adjonction de données relatives aux prédictions et aux coefficients

Les blocs de prédiction ont été formés et réorganisés en blocs d'échantillons de prédiction $p[y][x]$ qui correspondent à la structure de codage sur monotrames/bi-trames utilisée par les blocs de données de transformation.

Les données de transformation, $f[y][x]$, doivent être ajoutées aux données de prédiction puis être saturées afin de former les échantillons décodés ultimes $d[y][x]$ comme suit:

```
for (y = 0; y < 8; y++) {
  for (x = 0; x < 8; x++) {
    d[y][x] = f[y][x] + p[y][x];
    if (d[y][x] < 0) d[y][x] = 0;
    if (d[y][x] > 255) d[y][x] = 255;
  }
}
```

7.7 Echelonnabilité spatiale

Ce paragraphe spécifie le processus complémentaire de décodage qui est requis pour les extensions à l'échelonnabilité spatiale.

Aussi bien la couche inférieure que la couche d'amélioration doivent utiliser la «structure en tranche restreinte» (sans solution de continuité entre les tranches).

La Figure 7-13 représente, sous forme de diagramme, le processus de décodage des données vidéo avec l'extension d'échelonnabilité spatiale. Ce diagramme est simplifié par souci de clarté.

7.7.1 Structures syntaxiques supérieures

En général, la couche de base d'une hiérarchie à échelonnabilité spatiale peut être conforme à une norme de codage quelconque, y compris la Recommandation H.261, l'ISO/CEI 11172-2 et la présente Spécification. On notera cependant que, dans le cadre de la présente Spécification, la décodabilité d'une hiérarchie à échelonnabilité spatiale n'est prise en compte que si la couche de base est conforme à la présente Spécification ou à l'ISO/CEI 11172-2.

En raison du «couplage non déterministe» des couches, une seule contrainte syntaxique est nécessaire dans la couche d'amélioration en cas d'entrelacement des deux couches, inférieure et d'amélioration. Dans ce cas, la structure d'image doit adopter la même valeur que dans la bi-trame de référence utilisée pour la prédiction à partir de la couche inférieure. Voir au 7.7.3.1 la façon d'identifier cette image de référence.

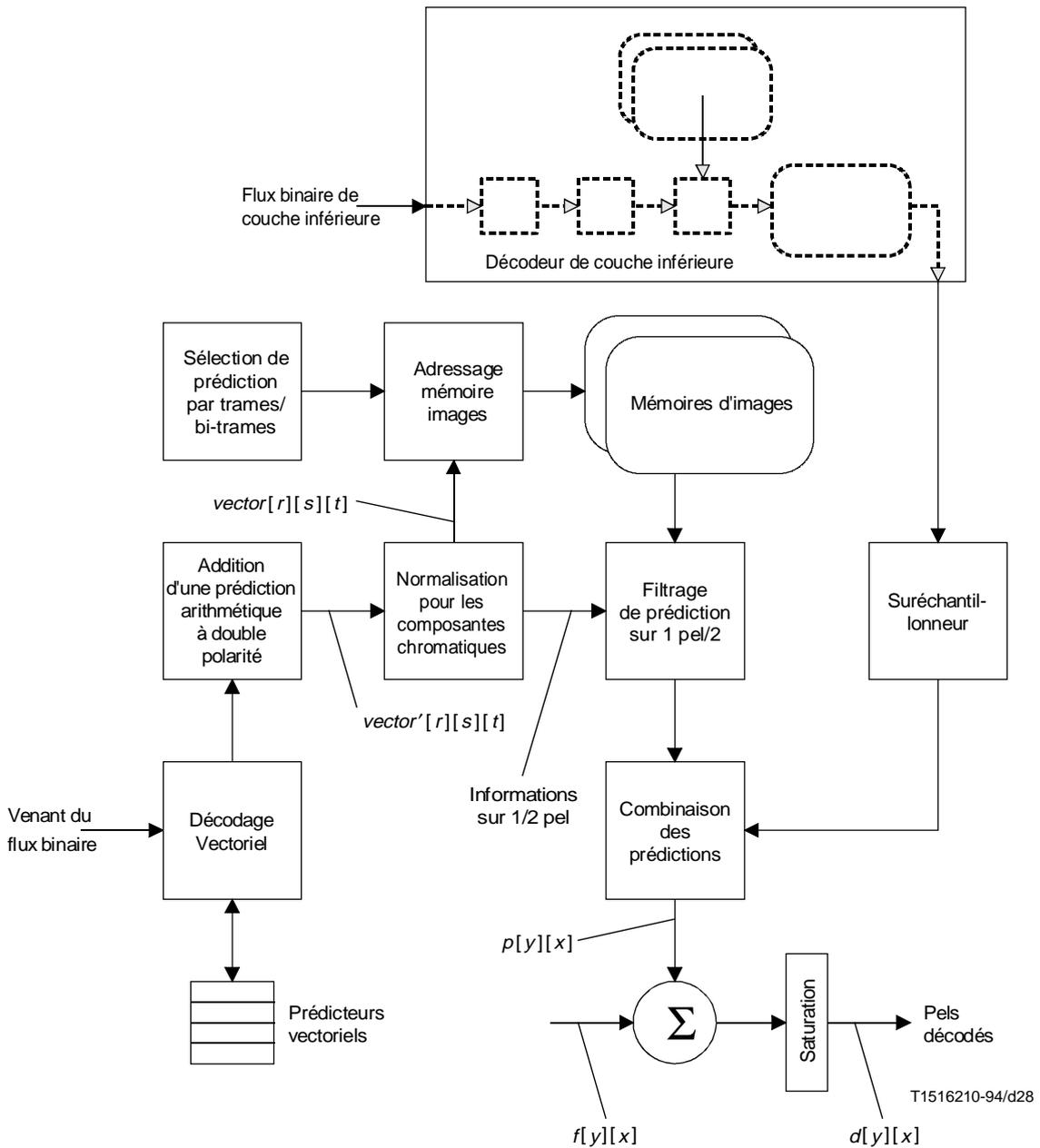


Figure 7-13 – Processus simplifié de compensation de mouvement pour l'échelonnabilité spatiale

7.7.2 Prédiction dans la couche d'amélioration

Une prédiction temporelle par compensation de mouvement est effectuée sur la base des trames de référence présentes dans la couche d'amélioration, comme indiqué en 7.6. En outre, une prédiction spatiale est établie à partir de l'image bi-trame décodée de la couche inférieure ($d_{\text{lower}}[y][x]$), comme indiqué en 7.7.3. Ces prédictions sont sélectionnées individuellement ou combinées pour former la prédiction proprement dite.

En général, on forme jusqu'à quatre prédictions distinctes pour chaque macrobloc et on les combine ensemble pour former le macrobloc de prédiction finale $p[y][x]$.

Si un macrobloc n'est pas codé, soit parce que tout le macrobloc est sauté soit parce que le macrobloc spécifique n'est pas codé, il n'y a pas de données relatives aux coefficients. Dans ce cas, la structure de données de transformation $f[y][x]$ est égale à zéro et les échantillons décodés sont simplement la prédiction finale, $p[y][x]$.

7.7.3 Formation de la prédiction spatiale

La formation de la prédiction spatiale nécessite l'identification de la bi-trame de référence correcte et la définition du processus de rééchantillonnage spatiale. Ces opérations sont décrites ci-après.

Le processus de rééchantillonnage est défini pour une bi-trame complète mais, pour le décodage d'un macrobloc, on n'a besoin que de la région 16×16 de l'image suréchantillonnée, correspondant à la position occupée par ce macrobloc.

7.7.3.1 Sélection de l'image bi-trame de référence

La prédiction spatiale est effectuée sur la base de l'image bi-trame reconstruite de la couche inférieure indiquée par la structure `lower_layer_temporal_reference`. Toutefois, si les flux binaires de la couche inférieure et de la couche d'amélioration sont encastrés dans un multiplex conforme à la Rec. UIT-T H.220.0 | ISO/CEI 13818-1 (systèmes), cette information est neutralisée par l'information de synchronisation donnée par les horodateurs de décodage (DTS) (*decoding time stamps*) contenus dans les en-têtes de flux PES.

NOTE – Si la structure `group_of_pictures_header()` apparaît souvent dans le flux binaire de couche inférieure, la référence temporelle de l'image de couche inférieure risque d'être ambiguë (parce que la structure `temporal_reference` est réinitialisée après un en-tête de groupe d'images).

L'image reconstruite sur la base de laquelle la prédiction spatiale est formée doit être l'une des images suivantes:

- l'image de couche inférieure coïncidente ou décodée le plus récemment;
- l'image de couche inférieure coïncidente ou décodée le plus récemment, à codage de type I ou P;
- l'image de couche inférieure à codage de type I ou P immédiatement précédente dans l'ordre de décodage, à condition que la couche inférieure ne comporte pas un fanion `low_delay` mis à la valeur '1'. On notera également que l'échelonnabilité spatiale ne fonctionnera efficacement que lorsque les prédictions sont formées sur la base de bi-frames de couche inférieure qui sont également coïncidentes (ou presque), en termes d'instant d'affichage, avec la bi-trame prédite dans la couche d'amélioration.

7.7.3.2 Processus de rééchantillonnage

La prédiction spatiale est réalisée par rééchantillonnage de la bi-trame de couche inférieure reconstruite selon la même maille d'échantillonnage que dans la couche d'amélioration. Cette maille est définie en termes de coordonnées d'image bi-trame, même si une image bi-trame entrelacée de couche inférieure a en fait été codée au moyen de deux images monotrames.

Ce processus de rééchantillonnage est illustré par la Figure 7-14.

Les prédictions spatiales ne doivent être faites que pour des macroblocs de couche d'amélioration qui s'inscrivent totalement dans l'image bi-trame reconstruite par suréchantillonnage dans la couche inférieure.

Le processus de suréchantillonnage diffère selon que la bi-trame reconstruite dans la couche inférieure est à balayage entrelacé ou progressif (indication fournie par la structure `lower_layer_progressive_frame`) et selon que la bi-trame de couche d'amélioration est à balayage entrelacé ou progressif (indication fournie par la structure `progressive_frame`).

Lorsque la structure `lower_layer_progressive_frame` a la valeur '1', l'image bi-trame reconstruite en couche inférieure (renommée `prog_pic`) est rééchantillonnée verticalement comme indiqué au 7.7.3.4. L'image bi-trame résultante est considérée comme étant à balayage progressif, si la structure `progressive_frame` a la valeur '1', et à balayage entrelacé si la structure `progressive_frame` a la valeur '0'. L'image bi-trame résultante est rééchantillonnée horizontalement comme indiqué au 7.7.3.6. La structure `lower_layer_deinterlaced_field_select` (sélection de trame désentrelacée de couche inférieure) doit prendre alors la valeur '1'.

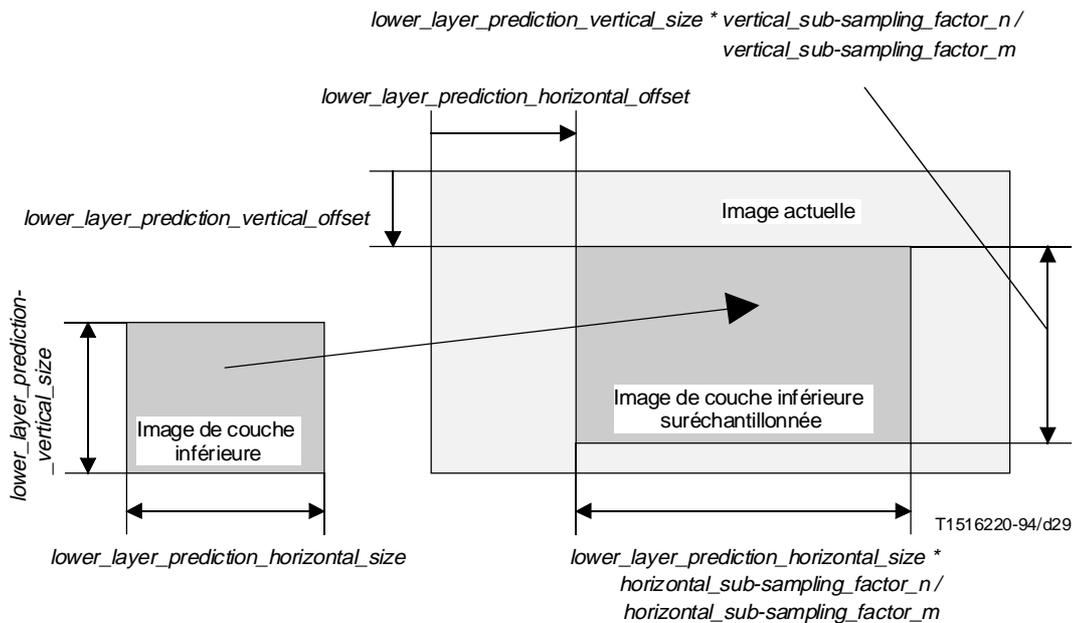


Figure 7-14 – Formation de la prédiction «spatiale» par interpolation de l'image de couche inférieure

Lorsque la structure `lower_layer_progressive_frame` a la valeur '0', et que la structure `progressive_frame` a la valeur '0', chaque image monotrame reconstruite en couche inférieure est désentrelacée comme indiqué au 7.7.3.4, afin de produire une trame à balayage progressif (renommée `prog_pic`). Cette trame est rééchantillonnée verticalement comme indiqué au paragraphe 7.7.3.5. La trame résultante est rééchantillonnée horizontalement comme indiqué au 7.7.3.6. Finalement la trame résultante est sous-échantillonnée afin d'obtenir une trame à balayage entrelacé. La structure `lower_layer_deinterlaced_field_select` (sélection de trame désentrelacée de couche inférieure) doit prendre la valeur '1'.

Lorsque la structure `lower_layer_progressive_frame` a la valeur '0', et que la structure `progressive_frame` a la valeur '1', chaque image monotrame reconstruite en couche inférieure est désentrelacée comme indiqué au 7.7.3.4, afin de produire une trame à balayage progressif (renommée `prog_pic`). Une seule trame de ce type est nécessaire. Lorsque la structure `lower_layer_deinterlaced_field_select` a la valeur '0', la trame supérieure est utilisée, sinon c'est la trame inférieure. La trame utilisée est rééchantillonnée verticalement comme indiqué au 7.7.3.5. La trame résultante est rééchantillonnée horizontalement comme indiqué au paragraphe 7.7.3.6.

Pour les images bi-trames entrelacées, si la bi-trame actuelle (et, implicitement la bi-trame de couche inférieure) est codée sous forme d'images monotrames, le processus de désentrelacement décrit au 7.7.3.5 est effectué à l'intérieur de chaque trame.

Les décalages verticaux et horizontaux d'image de couche inférieure, qui définissent la position dans l'image en cours de l'image bi-trame de couche inférieure, doivent être pris en compte dans les définitions des rééchantillonnages indiquées respectivement aux paragraphes 7.7.3.5 et 7.7.3.6. Les décalages d'images de couche inférieure sont limités à des valeurs paires lors du sous-échantillonnage des données de chrominance dans la couche d'amélioration, afin d'aligner les échantillons de chrominance entre ces deux couches.

Le processus de suréchantillonnage est résumé dans le Tableau 7-15.

Tableau 7-15 – Processus de suréchantillonnage

| lower_layer_deinterlaced_field_select | lower_layer_progressive_frame | progressive_frame | Application du processus de désentrelacement | Entité utilisée pour la prédiction |
|---------------------------------------|-------------------------------|-------------------|--|------------------------------------|
| 0 | 0 | 1 | Oui | Trame supérieure |
| 1 | 0 | 1 | Oui | Trame inférieure |
| 1 | 1 | 1 | Non | Bi-trame |
| 1 | 1 | 0 | Non | Bi-trame |
| 1 | 0 | 0 | Oui | Les deux trames |

7.7.3.3 Traitement des composantes de couleur

En raison des différences entre mailles d'échantillonnage des composantes de luminance et de chrominance, certaines variables utilisées dans 7.7.3.4 à 7.7.3.6 prennent des valeurs différentes lors du rééchantillonnage de luminance et de chrominance. Par ailleurs, il est admissible qu'il y ait une différence de format de chrominance entre la couche inférieure et la couche d'amélioration.

Le Tableau 7-16 définit les valeurs applicables aux variables utilisées dans 7.7.3.4 à 7.7.3.6.

Tableau 7-16 – Variables locales utilisées dans 7.7.3.3 à 7.7.3.5

| Variable | Valeur pour traitement de luminance | Valeur pour traitement de chrominance |
|-------------|--|---|
| ll_h_size | lower_layer_prediction_horizontal_size | lower_layer_prediction_horizontal_size / chroma_ratio_horizontal[lower] |
| ll_v_size | lower_layer_prediction_vertical_size | lower_layer_prediction_vertical_size / chroma_ratio_vertical[lower] |
| ll_h_offset | lower_layer_horizontal_offset | lower_layer_horizontal_offset / chroma_ratio_horizontal[enhance] |
| ll_v_offset | lower_layer_vertical_offset | lower_layer_vertical_offset / chroma_ratio_vertical[enhance] |
| h_subs_m | horizontal_subsampling_factor_m | horizontal_subsampling_factor_m |
| h_subs_n | horizontal_subsampling_factor_n | horizontal_subsampling_factor_n * format_ratio_horizontal |
| v_subs_m | vertical_subsampling_factor_m | vertical_subsampling_factor_m |
| v_subs_n | vertical_subsampling_factor_n | vertical_subsampling_factor_n * format_ratio_vertical |

Les Tableaux 7-17 et 7-18 donnent des définitions supplémentaires.

Tableau 7-17 – Facteurs de sous-échantillonnage pour la chrominance dans les couches = {inférieure, amélioration}

| Format de chrominance dans la couche inférieure | chroma_ratio_horizontal[layer] | chroma_ratio_vertical[layer] |
|---|--------------------------------|------------------------------|
| 4:2:0 | 2 | 2 |
| 4:2:2 | 2 | 1 |
| 4:4:4 | 1 | 1 |

Tableau 7-18 – Rapports entre les formats de chrominance

| Format de chrominance dans la couche inférieure | Format de chrominance dans la couche d'amélioration | format_ratio_horizontal | format_ratio_vertical |
|---|---|-------------------------|-----------------------|
| 4:2:0 | 4:2:0 | 1 | 1 |
| 4:2:0 | 4:2:2 | 1 | 2 |
| 4:2:0 | 4:4:4 | 2 | 2 |
| 4:2:2 | 4:2:2 | 1 | 1 |
| 4:2:2 | 4:4:4 | 2 | 1 |
| 4:4:4 | 4:4:4 | 1 | 1 |

7.7.3.4 Désentrelacement

Si l'opération de désentrelacement n'est pas nécessaire (d'après le Tableau 7-16), l'image bi-trame reconstruite en couche inférieure ($d_{lower}[y][x]$) est renommée `input_pic`.

Chaque trame de couche inférieure est d'abord bourrée de zéros pour former un canevas de balayage progressif à une fréquence d'image égale à la fréquence de trame de la couche inférieure et avec le même nombre de lignes et d'échantillons par ligne que l'image bi-trame de couche inférieure. Le Tableau 7-19 spécifie les filtres à appliquer ensuite. La composante de luminance est filtrée au moyen du filtre à ouverture bi-trame applicable si la structure `picture_structure` = «image bi-trame» ou, si ce n'est pas le cas, au moyen du filtre à ouverture monotrème. La composante de chrominance est filtrée au moyen du filtre à ouverture monotrème.

Tableau 7-19 – Filtre de désentrelacement

| Valeur temporelle | Valeur verticale | Filtre à ouverture bi-trame | | Filtre à ouverture monotrème |
|-------------------|------------------|-----------------------------|---------------------------|-------------------------------|
| | | Filtre pour première trame | Filtre pour seconde trame | Filtre (pour les deux trames) |
| -1 | -2 | 0 | -1 | 0 |
| -1 | 0 | 0 | 2 | 0 |
| -1 | 2 | 0 | -1 | 0 |
| 0 | -1 | 8 | 8 | 8 |
| 0 | 0 | 16 | 16 | 16 |
| 0 | 1 | 8 | 8 | 8 |
| 1 | -2 | -1 | 0 | 0 |
| 1 | 0 | 2 | 0 | 0 |
| 1 | +2 | -1 | 0 | 0 |

Les colonnes «valeur temporelle» et «valeur verticale» du Tableau 7-19 indiquent les coordonnées spatiales et temporelles relatives des pels auxquels s'appliquent les coefficients de filtrage définis dans les deux autres colonnes. On calcule une somme intermédiaire en additionnant les coefficients multipliés les uns par les autres.

La sortie du filtre (sum) est ensuite normalisée selon la formule suivante:

$$prog_pic[y][x] = sum // 16$$

puis saturée sur l'étendue [0:255].

L'ouverture du filtre peut aller au-delà des dimensions de l'image codée. Dans ce cas, les pels des lignes extérieures aux bords de l'image active doivent prendre la valeur du plus proche pel voisin (par excès ou par défaut) dans la même trame comme indiqué ci-dessus.

Pour tous les pels [y][x]:

```

if (y < 0 && (y&1 == 1))
    y = 1
if (y < 0 && (y&1 == 0))
    y = 0
if (y >= ll_v_size &&
    ((y-ll_v_size)&1 == 1))
    y = ll_v_size - 1
if (y >= ll_v_size &&
    ((y-ll_v_size)&1 == 0))
    y = ll_v_size - 2

```

7.7.3.5 Rééchantillonnage vertical

L'image bi-trame soumise au rééchantillonnage vertical (prog_pic) est rééchantillonnée selon la maille d'échantillonnage vertical de la couche d'amélioration avec application d'une interpolation linéaire entre les sites des échantillons, conformément à la formule suivante, dans laquelle vert_pic désigne la trame résultante:

$$\text{vert_pic}[y_h + \text{ll_v_offset}][x] = (16 - \text{phase}) * \text{prog_pic}[y_1][x] + \text{phase} * \text{prog_pic}[y_2][x]$$

où

```

y_h + ll_v_offset = coordonnée de l'échantillon de sortie dans vert_pic
y_1              = (y_h * v_subs_m) / v_subs_n
y_2              = y_1 + 1   if y_1 < ll_v_size - 1
                  y_1       et où
phase            = (16 * ((y_h * v_subs_m) % v_subs_n)) // v_subs_n

```

Les échantillons qui se trouvent à l'extérieur de l'image bi-trame reconstruite de couche inférieure, nécessaires pour le suréchantillonnage, s'obtiennent par extension du cadre de l'image bi-trame reconstruite de couche inférieure.

NOTE – Le calcul de la phase implique que la position de l'échantillon dans la couche d'amélioration, avec la coordonnée $y_h = 0$, coïncide spatialement avec la position du premier échantillon de la couche inférieure. Il est admis que cela est une approximation pour la composante de chrominance si son format est de type 4:2:0 (structure chroma_format = 4:2:0).

7.7.3.6 Rééchantillonnage horizontal

L'image bi-trame soumise au rééchantillonnage horizontal (vert_pic) est rééchantillonnée selon la maille d'échantillonnage horizontal de la couche d'amélioration avec application d'une interpolation linéaire entre les sites des échantillons, conformément à la formule suivante, dans laquelle hor_pic désigne la trame résultante:

$$\text{hor_pic}[y][x_h + \text{ll_h_offset}] = ((16 - \text{phase}) * \text{vert_pic}[y][x_1] + \text{phase} * \text{vert_pic}[y][x_2]) // 256$$

où

```

x_h + ll_h_offset = coordonnée de l'échantillon de sortie dans hor_pic
x_1              = (x_h * h_subs_m) / h_subs_n
x_2              = x_1 + 1   if x_1 < ll_h_size - 1
                  x_1       et où
phase            = (16 * ((x_h * h_subs_m) % h_subs_n)) // h_subs_n

```

Les échantillons qui se trouvent à l'extérieur de l'image bi-trame reconstruite de couche inférieure, nécessaires pour le suréchantillonnage, s'obtiennent par extension du cadre de l'image bi-trame reconstruite de couche inférieure.

7.7.3.7 Réentrelacement

Lorsqu'il n'est pas nécessaire d'effectuer un réentrelacement, le résultat du processus de rééchantillonnage (hor_pic) est renommé (spat_pred_pic).

Si hor_pic a été obtenu à partir de la trame supérieure d'une image bi-trame entrelacée de couche inférieure, les lignes paires de hor_pic sont copiées sur les lignes paires de spat_pred_pic.

Si hor_pic a été obtenu à partir de la trame inférieure d'une image bi-trame entrelacée de couche inférieure, les lignes impaires de hor_pic sont copiées sur les lignes impaires de spat_pred_pic.

Si hor_pic a été obtenu à partir d'une image à balayage progressif de la couche inférieure, hor_pic est copié sur spat_pred_pic.

7.7.4 Sélection et combinaison des prédictions spatiales et temporelles

On peut choisir ou combiner les prédictions spatiales et temporelles afin de former des prédictions finales. La structure macroblock_type (Tableaux B.5, B.6 et B.7) et la structure additionnal_spatial_temporal_weight_code (Tableau 7-21) indiquent, au moyen de la structure spatial_temporal_weight_class (classe de pondération spatio-temporelle), si la prédiction est de nature temporelle uniquement, spatiale uniquement ou combinée spatio-temporelle avec pondération. Les classes sont définies comme suit:

- classe 0: prédiction purement temporelle;
- classe 1: prédiction purement spatiale pour aucune des deux trames;
- classe 2: prédiction purement spatiale pour la trame supérieure;
- classe 3: prédiction purement spatiale pour la trame inférieure;
- classe 4: prédiction purement spatiale.

Dans les images à codage intra, si la structure spatial_temporal_weight a la valeur 0, un codage intra normal est effectué; sinon, la prédiction est purement spatiale. Dans les images prédites et interpolées, si la structure spatial_temporal_weight a la valeur 0, la prédiction est purement temporelle; si elle a la valeur 4, la prédiction est purement spatiale; sinon, on utilise un (ou deux) coefficient(s) de pondération pour combiner les prédictions spatiale et temporelle.

Les valeurs possibles des coefficients de combinaison spatio-temporelle (spatial_temporal_weight) sont indiquées dans une table de pondération qui est sélectionnée dans l'extension spatiale à l'échelonnabilité d'image. Jusqu'à quatre tables de pondération différentes peuvent être utilisées, selon que les couches actuelle et inférieure sont à balayage entrelacé ou progressif comme indiqué dans le Tableau 7-20 (les valeurs autorisées, quoique non recommandées, sont indiquées entre parenthèses).

Tableau 7-20 – Valeurs prévues (ou autorisées) de la structure spatial_temporal_weight_code_table_index

| Format dans la couche inférieure | Format de la couche d'amélioration | spatial_temporal_weight_code_table_index |
|---|------------------------------------|--|
| Balayage progressif ou entrelacé | Balayage progressif | 00 |
| Balayage progressif coïncidant avec les trames supérieures de couche d'amélioration | Balayage entrelacé | 10 (00; 01; 11) |
| Balayage progressif coïncidant avec les trames inférieures de couche d'amélioration | Balayage entrelacé | 01 (00; 10; 11) |
| Balayage entrelacé (picture_structure == Frame-Picture) | Balayage entrelacé | 00 ou 11 (01; 10) |
| Balayage entrelacé (picture_structure != Frame-Picture) | Balayage entrelacé | 00 |

Les modes macroblocs () utilisent un code de deux bits (le code de pondération spatio-temporelle) afin de décrire la prédiction pour chaque trame (ou image bi-trame), comme indiqué au Tableau 7-21. Dans ce tableau, le nombre entier de pondération spatio-temporelle identifie les codes de pondération spatio-temporelle également utilisables en cas de prédiction anticipée à double polarité (voir les Tableaux 7-22 et 7-23).

Tableau 7-21 – Coefficients spatial_temporal_weights et classes spatial_temporal_weight_classes pour l'index spatial_temporal_weight_code_table_index et pour les codes spatial_temporal_weight_codes

| spatial_temporal_weight_code_table_index | spatial_temporal_weight_code | spatial_temporal_weight (s) | spatial_temporal_weight class | spatial_temporal_integer_weight |
|--|------------------------------|-----------------------------|-------------------------------|---------------------------------|
| 00 ^{a)} | – | (0,5) | 1 | 0 |
| 01 | 00 | (0; 1) | 3 | 1 |
| | 01 | (0; 0,5) | 1 | 0 |
| | 10 | (0,5; 1) | 3 | 0 |
| | 11 | (0,5; 0,5) | 1 | 0 |
| 10 | 00 | (1; 0) | 2 | 1 |
| | 01 | (0,5; 0) | 1 | 0 |
| | 10 | (1; 0,5) | 2 | 0 |
| | 11 | (0,5; 0,5) | 1 | 0 |
| 11 | 00 | (1; 0) | 2 | 1 |
| | 01 | (1; 0,5) | 2 | 0 |
| | 10 | (0,5; 1) | 3 | 0 |
| | 11 | (0,5; 0,5) | 1 | 0 |
| ^{a)} Pour spatial_temporal_weight_code_table_index == 00, aucun code spatial_temporal_weight_code n'est transmis. | | | | |

NOTE – La prédiction purement spatiale (weight_class == 4) est signalée par différentes valeurs de la structure macroblock_type (voir les Tableaux B.5 à B.7).

Lorsque la combinaison spatio-temporelle spatial_temporal_weight est donnée sous la forme (a; b), le terme «a» indique la mesure dans laquelle la trame supérieure est déduite de la prédiction spatiale et le terme «b» indique la mesure dans laquelle la prédiction pour la trame inférieure est déduite de la prédiction spatiale pour cette trame.

Lorsque la pondération spatio-temporelle est donnée sous la forme (a), le terme «a» indique la mesure dans laquelle la prédiction de l'image est déduite de la prédiction spatiale pour cette image.

La méthode précise pour le calcul du prédicteur est la suivante:

La structure pel_pred_temp[y][x] est utilisée pour désigner la prédiction temporelle (formée à l'intérieur de la couche d'amélioration) comme défini dans 7.6 pour la structure pel_pred[y][x]. La structure pel_pred_spat[y][x] est utilisée pour désigner la prédiction formée sur la base de la couche inférieure, en extrayant les échantillons appropriés, situés à l'emplacement défini par la position de macrobloc actuelle, à partir de la structure spat_pred_pic.

Si la structure spatial_temporal_weight a la valeur 0, aucune prédiction n'est faite sur la base de la couche inférieure. Donc:

$$\text{pel_pred}[y][x] = \text{pel_pred_temp}[y][x];$$

Si la structure spatial_temporal_weight a la valeur 1, aucune prédiction n'est faite sur la base de la couche inférieure. Donc:

$$\text{pel_pred}[y][x] = \text{pel_pred_spat}[y][x];$$

Si le coefficient de pondération a la valeur «1/2», la prédiction est la moyenne des prédictions temporelle et spatiale. Donc:

$$\text{pel_pred}[y][x] = (\text{pel_pred_temp}[y][x] + \text{pel_pred_spat}[y][x]) // 2;$$

Si `progressive_frame` = 0, la composante de chrominance est traitée comme si le balayage était entrelacé, c'est-à-dire que le premier coefficient de pondération est utilisé pour les lignes de chrominance de la trame supérieure et que le deuxième coefficient de pondération est utilisé pour les lignes de chrominance de la trame inférieure.

L'adjonction de données relatives aux prédictions et aux coefficients s'effectue alors comme indiqué au 7.6.8.

7.7.5 Actualisation des prédicteurs de vecteurs de mouvement et sélection des vecteurs de mouvement

Dans les images bi-frames utilisant la prédiction sur monotrames, il est possible qu'une des trames soit prédite en mode purement spatial. Dans ce cas, aucun vecteur de mouvement n'est présent dans le flux binaire pour la trame faisant l'objet d'une prédiction purement spatiale. Si les deux trames d'une bi-frame font l'objet d'une prédiction purement spatiale, le type de macrobloc doit être tel qu'aucun vecteur de mouvement ne soit présent dans le flux binaire pour ce macrobloc.

La classe de pondération spatio-temporelle indique également le nombre de vecteurs de mouvement présents dans le flux binaire codé et la façon dont ils sont actualisés, conformément aux Tableaux 7-22 et 7-23.

Tableau 7-22 – Actualisation des prédicteurs de vecteurs de mouvement dans les images monotrames

| frame_motion_type | macroblock_motion_forward | | | | Prédicteurs à actualiser |
|--|-------------------------------|---|---|-------|--|
| | macroblock_motion_backward | | | | |
| | macroblock_intra | | | | |
| | spatial_temporal_weight_class | | | | |
| | | | | | |
| Prédiction sur trames ^{a)} | – | – | 1 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ ^{b)} |
| Prédiction sur trames | 1 | 1 | 0 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ $PMV[1][1][1:0] = PMV[0][1][1:0]$ |
| Prédiction sur trames | 1 | 0 | 0 | 0,1 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ |
| Prédiction sur trames | 0 | 1 | 0 | 0,1 | $PMV[1][1][1:0] = PMV[0][1][1:0]$ |
| Prédiction sur trames ^{a)} | 0 | 0 | 0 | 0,1,4 | $PMV[r][s][t] = 0$ ^{c)} |
| Prédiction avec compensation de mouvement (16 × 8 MC) | 1 | 1 | 0 | 0 | (Aucun) |
| Prédiction avec compensation de mouvement (16 × 8 MC) | 1 | 0 | 0 | 0,1 | (Aucun) |
| Prédiction avec compensation de mouvement (16 × 8 MC) | 0 | 1 | 0 | 0,1 | (Aucun) |
| Prédiction anticipée à double polarité | 1 | 0 | 0 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ |
| <p>a) La structure <code>field_motion_type</code> n'est pas présente dans le flux binaire mais est censée être de type codage sur trames.</p> <p>b) Si la structure <code>concealment_motion_vectors</code> a la valeur zéro, alors la structure $PMV[r][s][t]$ est mise à zéro (pour tous les r, s et t).</p> <p>c) La structure $PMV[r][s][t]$ est mise à zéro (pour tous les r, s et t). Voir 7.6.3.4.</p> <p>NOTE – $PMV[r][s][1:0] = PMV[u][v][1:0]$ signifie que:</p> $PMV[r][s][1] = PMV[u][v][1] \text{ et } PMV[r][s][0] = PMV[u][v][0]$ | | | | | |

Tableau 7-23 – Actualisation des prédicteurs de vecteurs de mouvement dans les images de type bi-trame

| frame_motion_type | macroblock_motion_forward | | | | Prédicteurs à actualiser |
|--|-------------------------------|---|---|-----------|--|
| | macroblock_motion_backward | | | | |
| | macroblock_intra | | | | |
| | spatial_temporal_weight_class | | | | |
| | | | | | |
| Prédiction sur bi-trames ^{a)} | – | – | 1 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ ^{c)} |
| Prédiction sur bi-trames | 1 | 1 | 0 | 0 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ $PMV[1][1][1:0] = PMV[0][1][1:0]$ |
| Prédiction sur bi-trames | 1 | 0 | 0 | 0,1,2,3 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ |
| Prédiction sur bi-trames | 0 | 1 | 0 | 0,1,2,3 | $PMV[1][1][1:0] = PMV[0][1][1:0]$ |
| Prédiction sur bi-trames ^{a)} | 0 | 0 | 0 | 0,1,2,3,4 | $PMV[r][s][t] = 0$ ^{d)} |
| Prédiction sur trames | 1 | 1 | 0 | 0 | (Aucun) |
| Prédiction sur trames | 1 | 0 | 0 | 0,1 | (Aucun) |
| Prédiction sur trames | 1 | 0 | 0 | 2 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ |
| Prédiction sur trames | 1 | 0 | 0 | 3 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ |
| Prédiction sur trames | 0 | 1 | 0 | 0,1 | (Aucun) |
| Prédiction sur trames | 0 | 1 | 0 | 2 | $PMV[1][1][1:0] = PMV[0][1][1:0]$ |
| Prédiction sur trames | 0 | 1 | 0 | 3 | $PMV[1][1][1:0] = PMV[0][1][1:0]$ |
| Prédiction anticipée à double polarité ^{b)} | 1 | 0 | 0 | 0,2,3 | $PMV[1][0][1:0] = PMV[0][0][1:0]$ |

a) La structure **frame_motion_type** n'est pas présente dans le flux binaire mais est censée être de type codage sur trames.

b) La prédiction anticipée à double polarité ne peut pas être utilisée lorsque la structure **spatial_temporal_integer_weight** = '0'.

c) Si la structure **concealment_motion_vectors** a la valeur zéro, alors la structure $PMV[r][s][t]$ est mise à zéro (pour tous les r , s et t).

d) La structure $PMV[r][s][t]$ est mise à zéro (pour tous les r , s et t). Voir 7.6.3.4.

NOTE – $PMV[r][s][1:0] = PMV[u][v][1:0]$ signifie que:
 $PMV[r][s][1] = PMV[u][v][1]$ et $PMV[r][s][0] = PMV[u][v][0]$

7.7.5.1 Réinitialisation des prédicteurs de vecteurs de mouvement

En plus des cas indiqués au 7.6.3.4, les prédicteurs de vecteurs de mouvement doivent être réinitialisés lorsque:

- un macrobloc est à prédiction purement spatiale dans une image P (**spatial_temporal_weight_class** = 4);
- un macrobloc est à prédiction purement spatiale dans une image B (**spatial_temporal_weight_class** = 4).

NOTE – Si la structure **spatial_temporal_weight_class** = 2 dans une image bi-trame en mode de prédiction sur monotrames, le vecteur transmis est appliqué pour la trame inférieure (voir le Tableau 7-25). Cependant, ce vecteur $vector[0][s][1:0]$ est prédit sur la base du vecteur $PMV[0][s][1:0]$. Le vecteur $PMV[1][s][1:0]$ est ensuite actualisé conformément au Tableau 7-23.

Tableau 7-24 – Prédications et vecteurs de mouvement dans les images monotrames

| field_motion_type | macroblock_motion_forward | | | | macroblock_motion_backward | |
|---|-------------------------------|---|---|-------|-------------------------------|---|
| | macroblock_motion_backward | | | | macroblock_intra | |
| | macroblock_intra | | | | spatial_temporal_weight_class | |
| | spatial_temporal_weight_class | | | | Vecteur de mouvement | Prédiction formée pour |
| | | | | | Vecteur de mouvement | Prédiction formée pour |
| Prédiction sur trames ^{a)} | – | – | 1 | 0 | $vector'[0][0][1:0]^b)$ | Aucune trame (ce vecteur de mouvement est de type masquage) |
| Prédiction sur trames | 1 | 1 | 0 | 0 | $vector'[0][0][1:0]$ | Trame entière, anticipée |
| | | | | | $vector'[0][1][1:0]$ | Trame entière, différée |
| Prédiction sur trames | 1 | 0 | 0 | 0,1 | $vector'[0][0][1:0]$ | Trame entière, anticipée |
| Prédiction sur trames | 0 | 1 | 0 | 0,1 | $vector'[0][1][1:0]$ | Trame entière, différée |
| Prédiction sur trames ^{a)} | 0 | 0 | 0 | 0,1,4 | $vector'[0][0][1:0]^c) d)$ | Trame entière, anticipée |
| Prédiction avec compensation de mouvement (16 × 8 MC) | 1 | 1 | 0 | 0 | $vector'[0][0][1:0]$ | Trame supérieure 16 × 8, anticipée |
| | | | | | $vector'[1][0][1:0]$ | Trame inférieure 16 × 8, anticipée |
| | | | | | $vector'[0][1][1:0]$ | Trame supérieure 16 × 8, différée |
| | | | | | $vector'[1][1][1:0]$ | Trame inférieure 16 × 8, différée |
| Prédiction avec compensation de mouvement (16 × 8 MC) | 1 | 0 | 0 | 0,1 | $vector'[0][0][1:0]$ | Trame supérieure 16 × 8, anticipée |
| | | | | | $vector'[1][0][1:0]$ | Trame inférieure 16 × 8, anticipée |
| Prédiction avec compensation de mouvement (16 × 8 MC) | 0 | 1 | 0 | 0,1 | $vector'[0][1][1:0]$ | Trame supérieure 16 × 8, différée |
| | | | | | $vector'[1][1][1:0]$ | Trame inférieure 16 × 8, différée |
| Prédiction anticipée à double polarité | 1 | 0 | 0 | 0 | $vector'[0][0][1:0]$ | Trame entière, même parité, anticipée |
| | | | | | $vector'[2][0][1:0]^c) e)$ | Trame entière, parité opposée, anticipée |

a) La structure **field_motion_type** n'est pas présente dans le flux binaire mais est censée être de type codage sur trames.

b) Ce vecteur de mouvement n'est présent que si la structure **concealment_motion_vectors** a la valeur 1.

c) Ces vecteurs de mouvement ne sont pas présents dans le flux binaire.

d) Ce vecteur de mouvement est censé avoir la valeur (0; 0) comme expliqué au 7.6.3.5.

e) Ces vecteurs de mouvement sont calculés d'après $vector'[0][0][1:0]$ comme décrit au 7.6.3.6.

NOTE – Les vecteurs de mouvement sont énumérés dans l'ordre de leur apparition dans le flux binaire.

Tableau 7-25 – Prédications et vecteurs de mouvement dans les images bi-frames

| frame_motion_type | macroblock_motion_forward | | | | Vecteur de mouvement | Prédiction formée pour |
|--|-------------------------------|---|---|-----------|--------------------------------------|---|
| | macroblock_motion_backward | | | | | |
| | macroblock_intra | | | | | |
| | spatial_temporal_weight_class | | | | | |
| | | | | | | |
| Prédiction sur bi-frames ^{a)} | – | – | 1 | 0 | $vector[0][0][1:0]$ ^{c)} | Aucune trame (ce vecteur de mouvement est de type masquage) |
| Prédiction sur bi-frames | 1 | 1 | 0 | 0 | $vector[0][0][1:0]$ | Bi-trame, anticipée |
| | | | | | $vector[0][1][1:0]$ | Bi-trame, différée |
| Prédiction sur bi-frames | 1 | 0 | 0 | 0,1,2,3 | $vector[0][0][1:0]$ | Bi-trame, anticipée |
| Prédiction sur bi-frames | 0 | 1 | 0 | 0,1,2,3 | $vector[0][1][1:0]$ | Bi-trame, différée |
| Prédiction sur bi-frames ^{a)} | 0 | 0 | 0 | 0,1,2,3,4 | $vector[0][0][1:0]$ ^{d) e)} | Bi-trame, anticipée |
| Prédiction sur trames | 1 | 1 | 0 | 0 | $vector[0][0][1:0]$ | Trame supérieure, anticipée |
| | | | | | $vector[1][0][1:0]$ | Trame inférieure, anticipée |
| | | | | | $vector[0][1][1:0]$ | Trame supérieure, différée |
| | | | | | $vector[1][1][1:0]$ | Trame inférieure, différée |
| Prédiction sur trames | 1 | 0 | 0 | 0,1 | $vector[0][0][1:0]$ | Trame supérieure, anticipée |
| | | | | | $vector[1][0][1:0]$ | Trame inférieure, anticipée |
| Prédiction sur trames | 1 | 0 | 0 | 2 | | Trame supérieure, spatiale |
| | | | | | $vector[0][0][1:0]$ | Trame inférieure, anticipée |
| Prédiction sur trames | 1 | 0 | 0 | 3 | $vector[0][0][1:0]$ | Trame supérieure, anticipée |
| | | | | | | Trame inférieure, spatiale |
| Prédiction sur trames | 0 | 1 | 0 | 0,1 | $vector[0][1][1:0]$ | Trame supérieure, différée |
| | | | | | $vector[1][1][1:0]$ | Trame inférieure, différée |
| Prédiction sur trames | 0 | 1 | 0 | 2 | | Trame supérieure, spatiale |
| | | | | | $vector[0][1][1:0]$ | Trame inférieure, différée |
| Prédiction sur trames | 0 | 1 | 0 | 3 | $vector[0][1][1:0]$ | Trame supérieure, différée |
| | | | | | | Trame inférieure, spatiale |
| Prédiction anticipée à double polarité ^{b)} | 1 | 0 | 0 | 0,2,3 | $vector[0][0][1:0]$ | Trame supérieure, même parité, anticipée |
| | | | | | $vector[0][0][1:0]$ ^{d)} | Trame inférieure, même parité, anticipée |
| | | | | | $vector[2][0][1:0]$ ^{d) f)} | Trame supérieure, parité opposée, anticipée |
| | | | | | $vector[3][0][1:0]$ ^{d) f)} | Trame inférieure, parité opposée, anticipée |

a) La structure **frame_motion_type** n'est pas présente dans le flux binaire mais on part du principe que le codage est sur bi-frames.

b) La prédiction anticipée à double polarité ne peut pas être utilisée lorsque la structure **spatial_temporal_integer_weight** = '0'.

c) Ce vecteur de mouvement n'est présent que si la structure **concealment_motion_vectors** a la valeur 1.

d) Ces vecteurs de mouvement ne sont pas présents dans le flux binaire.

e) Ce vecteur de mouvement est censé avoir la valeur (0; 0) comme expliqué au 7.6.3.5.

f) Ces vecteurs de mouvement sont calculés d'après $vector'[0][0][1:0]$ comme décrit au 7.6.3.6.

NOTE – Les vecteurs de mouvement sont énumérés dans l'ordre de leur apparition dans le flux binaire.

7.7.6 Macroblocs sautés

Dans tous les cas, un macrobloc sauté ne résulte que d'une prédiction et tous les coefficients DCT sont considérés comme égaux à zéro.

Si une extension d'échelonnabilité des séquences est présente et que le mode d'échelonnabilité soit «échelonnabilité spatiale», les règles suivantes s'appliquent en plus de celles du 7.6.6.

Dans les images I, les macroblocs sautés sont autorisés. Ils sont définis comme faisant l'objet d'une prédiction purement spatiale.

Dans les images P et dans les images B, le macrobloc sauté fait l'objet d'une prédiction purement temporelle.

Dans les images B, un macrobloc sauté ne doit pas suivre un macrobloc à prédiction purement spatiale.

7.7.7 Sous-remplissage de mémoire du vérificateur VBV dans la couche inférieure

En hiérarchie d'échelonnabilité spatiale, un sous-remplissage de mémoire du vérificateur VBV dans la couche inférieure peut provoquer des problèmes, en raison d'une éventuelle imprécision quant aux images bi-frames exactes qui seront répétées par un décodeur particulier.

7.8 Echelonnabilité SNR

Voir la Figure 7-15.

Ce paragraphe décrit le processus de décodage requis pour les extensions à l'échelonnabilité SNR.

L'échelonnabilité SNR définit un mécanisme de raffinement des coefficients DCT codés dans une autre couche (inférieure) d'une hiérarchie échelonnable. Comme illustré à la Figure 7-15, les données codées issues de deux flux binaires sont combinées après les processus de quantification inverse, par sommation des coefficients DCT. Avant que les données soient combinées, les processus de décodage respectifs des deux couches sont indépendants.

Le 7.8.1 définit la méthode d'identification de ces flux binaires dans une hiérarchie échelonnable, que l'on peut toutefois décrire comme suit.

La couche inférieure, issue du premier flux binaire, peut être elle-même non échelonnable ou nécessiter l'application du processus de décodage d'échelonnabilité spatiale ou temporelle (et donc de décodage de flux binaires supplémentaires).

La couche d'amélioration, issue du deuxième flux binaire, contient principalement des coefficients DCT codés et un petit résidu binaire. Le présent paragraphe décrira le processus de décodage pour cette couche et la combinaison des deux couches de données.

NOTE – Toutes les informations relatives à la prédiction ne sont contenues que dans le flux binaire de la couche inférieure. Il n'est donc pas possible de reconstruire une couche d'amélioration sans décoder, en parallèle, les données du flux binaire de la couche inférieure.

Par ailleurs la prédiction et la reconstruction des images, telles que décrites dans 7.6, 7.7 et 7.9 pour la combinaison des couches inférieure et d'amélioration, suivent les mêmes étapes de décodage que pour le seul flux binaire de la couche inférieure.

La sémantique et le processus de décodage décrits dans ce paragraphe comportent un mécanisme de «diffusion jumelée des données de chrominance», qui peut être utilisé pour (par exemple) améliorer un format 4:2:0 dans la couche inférieure afin d'obtenir un format 4:2:2 après traitement des données de la couche d'amélioration. Bien que les données de luminance soient traitées comme décrit ci-dessus, dans le cas présent les informations de chrominance extraites du flux binaire de la couche inférieure (à l'exception des valeurs du coefficient DC, voir 7.8.3.4) doivent être rejetées et remplacées par les nouvelles informations en provenance de la couche d'amélioration, dont le décodage fournit une meilleure résolution de chrominance.

L'échelonnabilité SNR se caractérise par le fait que les deux couches sont très étroitement couplées. Il est nécessaire que les images correspondantes de chaque couche soient décodées simultanément.

Si le flux binaire de la couche inférieure est conforme à ISO/CEI 11172-2 (et non à la présente Spécification), deux mécanismes différents de contrôle d'adaptation après transformation IDCT seront utilisés au cours du décodage. Il faut veiller à ce que le codeur en tienne compte.

7.8.1 Structures syntaxiques supérieures

Les deux couches de flux binaire décrites dans le présent paragraphe sont identifiées par leur structure `layer_id`, décodée sur la base de la structure `sequence_scalable_extension`.

Les deux flux binaires doivent avoir des identificateurs de couche consécutifs: la couche d'amélioration étant désignée par la structure $\text{layer_id} = \text{id}_{\text{enhance}}$ et la couche inférieure par la structure $\text{layer_id} = \text{id}_{\text{enhance}} - 1$.

La syntaxe et la sémantique de la couche d'amélioration sont respectivement conformes aux 6.2 et 6.3.

Si la couche inférieure est conforme à ISO/CEI 11172-2 (et non pas à la présente Spécification), aussi bien cette couche inférieure que la couche d'amélioration doivent utiliser la «structure de tranche restreinte» qui est définie dans la présente Spécification.

Des restrictions sémantiques s'appliquent à plusieurs valeurs contenues dans les en-têtes et dans les extensions de la couche d'amélioration. Elles sont énumérées ci-dessous.

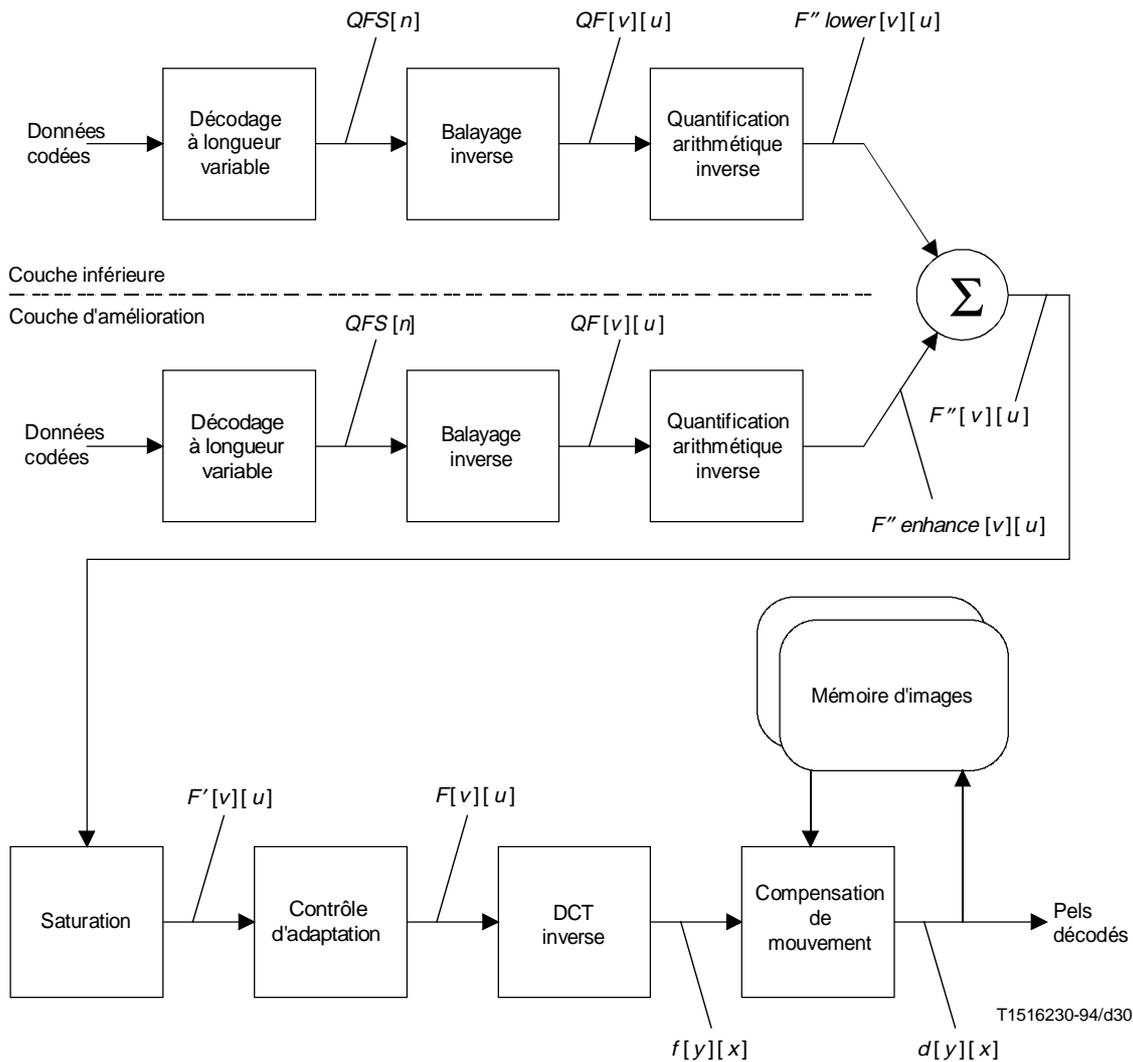


Figure 7-15 – Illustration du processus de décodage pour l'échelonnabilité SNR

En-tête de séquence

Cet en-tête doit être identique à celui de la couche inférieure, sauf pour les valeurs des structures bit_rate , vbv_buffer_size , $\text{load_intra_quantiser_matrix}$, $\text{intra_quantiser_matrix}$, $\text{load_non_intra_quantiser_matrix}$ et $\text{non_intra_quantiser_matrix}$. Ces valeurs pourront être choisies indépendamment, sauf pour $\text{load_intra_quantiser_matrix}$ qui doit avoir la valeur zéro.

Extension de séquence

Cette extension doit être identique à celle de la couche inférieure, sauf pour les valeurs des structures `profile_and_level_indication`, `chroma_format`, `bit_rate_extension` et `vbv_buffer_size_extension`, qui peuvent être choisies indépendamment.

Une valeur différente de la structure `chroma_format` dans chaque couche provoquera le basculement du fanion `chroma_simulcast` comme spécifié au Tableau 7-26.

Tableau 7-26 – Fanion `chroma_simulcast`

| chroma_format (lower layer) | chroma_format (enhancement layer) | chroma_simulcast |
|--|--|-------------------------|
| 4:2:0 | 4:2:0 | 0 |
| 4:2:0 | 4:2:2 | 1 |
| 4:2:0 | 4:4:4 | 1 |
| 4:2:2 | 4:2:2 | 0 |
| 4:2:2 | 4:4:4 | 1 |
| 4:4:4 | 4:4:4 | 0 |

Le format de chrominance de la couche d'amélioration doit être supérieur ou égal à celui de la couche inférieure.

Si la couche inférieure est conforme à ISO/CEI 11172-2 (et non pas à la présente Spécification), la structure `sequence_extension()` n'est pas présente dans la couche inférieure et les valeurs suivantes doivent être prises en compte pour le processus de décodage:

- `progressive_sequence` = 1
- `chroma_format` = «4:2:0»
- `horizontal_size_extension` = 0
- `vertical_size_extension` = 0
- `bit_rate_extension` = 0
- `vbv_buffer_size_extension` = 0
- `low_delay` = 0
- `frame_rate_extension_n` = 0
- `frame_rate_extension_d` = 0

La structure `sequence_extension()` contenue dans la couche d'amélioration doit avoir les valeurs indiquées ci-dessus.

Extension d'affichage de séquences

Cette extension ne doit pas être présente car il n'y a pas de processus d'affichage séparé pour la couche d'amélioration.

Extension à l'échelonnabilité des séquences

Cette extension doit être présente avec le mode «échelonnabilité SNR».

En-tête de groupe d'images

Cet en-tête doit être identique à celui du flux binaire de la couche inférieure.

NOTE 1 – L'en-tête de groupe d'images doit être présent dans chaque couche, de façon à pouvoir réactualiser la référence temporelle sur la même image bi-trame.

En-tête d'image

Cet en-tête doit être identique à celui qui se trouve en couche inférieure, sauf pour la valeur du retard `vbv_delay`, qui peut être sélectionnée indépendamment.

Extension du codage d'image

Cette extension doit être identique à celle qui se trouve en couche inférieure, sauf pour la valeur du type `q_scale_type` et celle de la variante `alternate_scan`, qui peuvent être sélectionnées indépendamment.

Le type `chroma_420_type` doit être mis à '0' si le fanion `chroma_simulcast` est activé. Sinon, cette structure doit avoir la même valeur qu'en couche inférieure.

Si la couche inférieure est conforme à ISO/CEI 11172-2 (et non pas à la présente Spécification), la structure `picture_coding_extension()` n'est pas présente dans la couche inférieure et les valeurs suivantes doivent être prises en compte pour le processus de décodage:

| | | |
|---|---|---|
| <code>f_code[0][0]</code> | = | <code>forward_f_code</code> dans la couche inférieure, ou 15 |
| <code>f_code[0][1]</code> | = | <code>forward_f_code</code> dans la couche inférieure, ou 15 |
| <code>f_code[1][0]</code> | = | <code>backward_f_code</code> dans la couche inférieure, ou 15 |
| <code>f_code[1][1]</code> | = | <code>backward_f_code</code> dans la couche inférieure, ou 15 |
| <code>intra_dc_precision</code> | = | 0 |
| <code>picture_structure</code> | = | «image bi-trame» |
| <code>top_field_first</code> | = | 0 |
| <code>frame_pred_frame_dct</code> | = | 1 |
| <code>concealment_motion_vectors</code> | = | 0 |
| <code>intra_vlc_format</code> | = | 0 |
| <code>repeat_first_field</code> | = | 0 |
| <code>chroma_420_type</code> | = | 1 |
| <code>progressive_frame</code> | = | 1 |
| <code>composite_display_flag</code> | = | 0 |

La structure `picture_coding_extension()` se trouvant dans la couche d'amélioration doit avoir les valeurs indiquées ci-dessus.

On doit considérer que les structures lower layer `q_scale_type` et `alternate_scan` ont la valeur zéro.

NOTE 2 – Les structures `q_scale_type` et `alternate_scan` peuvent être ajustées séparément dans la couche d'amélioration.

Extension relative à une matrice de quantification

Cette extension est facultative. Sa sémantique est décrite au 6.3.11.

Les structures `load_intra_quantiser_matrix` et `load_chroma_intra_quantiser_matrix` doivent toutes les deux avoir la valeur zéro.

NOTE 3 – Seules les matrices de codage non intra seront utilisées dans le processus de décodage ci-après.

Extension au recadrage plein écran

Cette extension ne doit pas être présente à ce niveau d'échelonnabilité.

NOTE 4 – Il n'y a pas de processus d'affichage séparé pour la couche d'amélioration. Si la fonction de recadrage plein écran est souhaitée, elle peut être immédiatement remplie au moyen des informations acheminées dans la couche inférieure par l'extension de recadrage.

En-tête de tranche

Les tranches doivent coïncider avec celles de la couche inférieure. La valeur de la structure `quantiser_scale_code` doit être choisie indépendamment de la couche inférieure.

7.8.2 Macrobloc

Le terme «macrobloc actuel» désignera, ci-après, le macrobloc qui est en cours de traitement. Le terme macrobloc actuel de la couche inférieure désignera le macrobloc qui a la même adresse que le macrobloc actuel.

Le décodage des informations relatives à l'en-tête du macrobloc sera effectué conformément aux sémantèmes du 6.3.17.

NOTE – Le Tableau B.8, qui est utilisé si le mode d'échelonnabilité est «échelonnabilité SNR», ne permettra jamais d'activer les fanions `macroblock_intra`, `macroblock_motion_forward` ou `macroblock_motion_backward` car un macrobloc situé dans la couche d'amélioration ne contient que les données d'enrichissement du macrobloc actuel de la couche inférieure.

Les éléments syntaxiques et fanions correspondants du macrobloc actuel de la couche inférieure sont toutefois applicables au processus de décodage par combinaison des couches inférieure et d'amélioration, selon la transformation DCT inverse qui est décrite au 7.8.3.5.

7.8.2.1 Type de transformation DCT

L'élément syntaxique `dct_type` peut être présent dans un des deux modes de macrobloc de couche inférieure et de couche d'amélioration, dans les deux modes ou dans aucun, selon les indications sémantiques du 6.3.17.

Si la structure `dct_type` est présente dans chacune des deux couches, insérée dans la structure `macroblock_modes()`, elle doit avoir des valeurs identiques dans chaque couche.

7.8.2.2 Macroblocs sautés

Des macroblocs peuvent être sautés dans le flux binaire de la couche d'amélioration, c'est-à-dire qu'aucune amélioration par coefficients DCT n'est effectuée ($F''_{enhance}[v][u] = 0$, pour tous les v, u). Dans ce cas, le processus de décodage décrit au 7.8.3 doit être appliqué.

Si des macroblocs sont sautés dans les deux flux binaires, dans celui de la couche inférieure comme dans celui de la couche d'amélioration, le processus de décodage est exactement tel que spécifié au 7.6.6.

Des macroblocs peuvent également être sautés dans le flux binaire de la couche inférieure, tout en étant toujours codés dans la couche d'amélioration. Le processus de décodage exposé ci-après doit alors être appliqué, mais $F''_{lower}[v][u] = 0$, pour tous les v, u .

7.8.3 Blocs

La première partie du processus de décodage des blocs de couche d'amélioration est indépendante de la couche inférieure.

La deuxième partie du processus de décodage des blocs de couche d'amélioration doit être réalisée conjointement avec le processus de décodage des blocs coïncidents de couche inférieure.

Deux ensembles de coefficients à quantification inverse, F''_{lower} et $F''_{enhance}$ sont ajoutés pour former la grandeur F'' (voir la Figure 7-15).

La grandeur F''_{lower} est calculée sur la base de la couche inférieure exactement comme défini aux 7.1 à 7.4.2.3.

La grandeur $F''_{enhance}$ est calculée comme indiqué dans les articles ci-après.

Le signal résultant, F'' , subit un traitement complémentaire, commençant par une saturation, comme indiqué aux 7.4.3 à 7.6 (7.7, 7.9).

7.8.3.1 Décodage à longueur variable

Dans un bloc de couche d'amélioration, le décodage à longueur variable (VLC) doit être effectué conformément au 7.2, comme pour un bloc à codage non intra (ce qui est signalé par la structure `macroblock_intra = 0`).

7.8.3.2 Balayage inverse

Le balayage inverse doit être effectué exactement comme indiqué au 7.3.

7.8.3.3 Quantification inverse

Dans un bloc de couche d'amélioration, la quantification inverse doit être effectuée comme indiqué au 7.4.2 pour un bloc à codage non intra.

Si la couche inférieure est conforme à ISO/CEI 11172-2 (et non à la présente Spécification), le «logarithme de quantification inverse» utilisé pour calculer la grandeur $F''_{lower}[v][u]$ (voir la Figure 7-14) doit inclure le contrôle d'adaptation (par imparité) et l'opération de saturation, comme spécifié dans ISO/CEI 11172-2.

7.8.3.4 Somme des coefficients issus des deux couches

Les coefficients homologues des blocs de chaque couche doivent être additionnés pour former le signal F'' (voir la Figure 7-15).

$$F''[v][u] = F''_{lower}[v][u] + F''_{enhance}[v][u], \text{ pour tous les } u, v$$

Si le fanion `chroma_simulcast = 1` est activé, seuls les blocs de luminance sont traités comme décrit ci-dessus.

Pour les blocs de chrominance, le coefficient DC de la couche de base est utilisé pour prédire le coefficient DC dans le bloc homologue de la couche d'amélioration, tandis que les coefficients AC de la couche de base sont rejetés et que ceux de la couche d'amélioration forment le signal F'' selon les formules suivantes (voir la Figure 7-14):

$$F''[0][0] = F''_{lower}[0][0] + F''_{enhance}[0][0]$$

$$F''[v][u] = F''_{enhance}[v][u], \text{ pour tous les } u, v \text{ sauf } u = v = 0$$

NOTE – Les blocs de diffusion jumelée des données de chrominance subissent une quantification inverse comme les blocs à codage non intra et utilisent la matrice de codage de chrominance non intra.

Le Tableau 7-27 donne les indices de bloc de chrominance dont le coefficient DC ($F''_{lower}[0][0]$) doit être utilisé pour prédire le coefficient DC dans le bloc de chrominance homologue de la couche d'amélioration ($F''_{enhance}[0][0]$).

Tableau 7-27 – Indices de bloc utilisés pour prédire le coefficient DC

| Structure chroma_format | Indice de bloc | | | | | | | |
|---------------------------------|----------------|---|---|---|---|---|----|----|
| | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| base: 4:2:0 supérieur: 4:2:2 | 4 | 5 | 4 | 5 | | | | |
| base: 4:2:0 supérieur: 4:4:4 | 4 | 5 | 4 | 5 | 4 | 5 | 4 | 5 |
| base: 4:2:2 supérieur: 4:4:4 | 4 | 5 | 6 | 7 | 4 | 5 | 6 | 7 |

7.8.3.5 Phases suivantes de décodage de macroblocs

Après sommation des coefficients issus des deux couches, le reste des phases de décodage de macroblocs est exactement conforme aux descriptions des 7.4.3 à 7.6 (7.7 et 7.9 si applicables), car il n'y a plus qu'un seul flux de données à traiter: $F''[v][u]$.

Dans ce processus, le signal spatio-temporel de prédiction combinée, $p[y][x]$, est calculé conformément aux éléments syntaxiques et aux fanions relatifs au type de macrobloc, pour le macrobloc actuel qui a été extrait de la couche inférieure.

7.9 Echelonnabilité temporelle

L'échelonnabilité temporelle met en œuvre deux couches: une couche inférieure et une couche d'amélioration. Ces deux couches acheminent la même résolution spatiale. La couche d'amélioration enrichit la résolution temporelle de la couche inférieure et, si elle est remultiplexée dans le temps avec le signal de couche inférieure, fournit la base de temps principale, qui est la fréquence d'image indiquée dans la couche d'amélioration. Le processus de décodage des images de couche d'amélioration est semblable au processus normal de décodage qui est décrit aux 7.1 à 7.6. La seule différence réside dans la «sélection des trames et bi-trames pour la prédiction», décrite au 7.6.2.

Les images de référence utilisées pour la prédiction sont choisies par le code `reference_select_code` qui est décrit dans les Tableaux 7-28 et 7-29. Dans le cas des images P, l'image de référence anticipée peut être l'un des trois types suivants: la plus récente image d'amélioration, la plus récente bi-trame de couche inférieure ou la prochaine (dans l'ordre d'affichage) bi-trame de couche inférieure. A noter que, dans ce dernier cas, la bi-trame de référence, utilisée pour la prédiction dans la couche inférieure, est différée dans le temps.

Tableau 7-28 – Sélection des références de prédiction pour des images P

| reference_select_code | Image de référence pour prédiction anticipée |
|-----------------------|--|
| 00 | Image(s) décodée(s) le plus récemment dans la couche d'amélioration |
| 01 | Bi-trame de couche inférieure la plus récente (dans l'ordre d'affichage) |
| 10 | Prochaine bi-trame de couche inférieure (dans l'ordre d'affichage) |
| 11 | Valeur interdite |

Tableau 7-29 – Sélection des références de prédiction pour des images B

| reference_select_code | Référence pour prédiction anticipée | Référence pour prédiction différée |
|-----------------------|---|---|
| 00 | Valeur interdite | Valeur interdite |
| 01 | Image(s) décodée(s) le plus récemment dans la couche d'amélioration | Image(s) décodée(s) le plus récemment dans la couche inférieure (ordre d'affichage) |
| 10 | Image(s) décodée(s) le plus récemment dans la couche d'amélioration | Prochaine image de couche inférieure (dans l'ordre d'affichage) |
| 11 | Image décodée le plus récemment dans la couche inférieure | Prochaine image de couche inférieure (dans l'ordre d'affichage) |

Dans le cas des images B, la référence anticipée peut être l'une des deux suivantes: la plus récente image d'amélioration ou la plus récente (ou coïncidente dans le temps) bi-trame de couche inférieure. En revanche, la référence différée peut être l'une des deux suivantes: la plus récente image de couche inférieure, y compris l'image coïncidente dans le temps dans l'ordre d'affichage, ou la prochaine (dans l'ordre d'affichage) bi-trame de couche inférieure. A noter que, dans ce dernier cas, la bi-trame de référence, utilisée pour la prédiction différée dans la couche inférieure, est temporellement anticipée.

La prédiction différée ne peut pas être effectuée sur la base d'une image appartenant à la couche d'amélioration. Cela permet d'éviter un réordonnement des bi-trames dans la couche d'amélioration. Le processus de compensation de mouvement forme des prédictions qui utilisent des images décodées dans la couche inférieure et/ou des images de couche d'amélioration ayant déjà fait l'objet d'une prédiction temporelle.

La couche d'amélioration peut contenir des images I, des images P ou des images B; mais les images B d'une couche d'amélioration se comportent plus comme des images P en ce sens qu'une image B, une fois décodée, peut servir à prédire les images P ou B suivantes de la couche d'amélioration.

Lorsque la plus récente bi-trame de la couche inférieure est utilisée comme référence, cela inclut la bi-trame qui coïncide dans le temps avec la bi-trame (ou la première monotrane dans le cas d'images monotrames) se trouvant dans la couche d'amélioration. Les références utilisées pour la prédiction des images P et des images B sont indiquées, respectivement, dans les Tableaux 7-28 et 7-29.

La couche inférieure et la couche d'amélioration doivent utiliser la structure de tranche restreinte.

La Figure 7-16 montre, sous forme de diagramme simplifié, le processus de compensation de mouvement pour la couche d'amélioration en échelonnabilité temporelle.

Les images I ne font pas appel à des références de prédiction autres qu'elles-mêmes; pour indiquer cela, la structure reference_select_code doit avoir la valeur '11' pour ces images.

Selon le type de codage d'image, les structures forward_temporal_reference ou backward_temporal_reference doivent prendre la valeur 0 lorsqu'elle ne désignent pas de références à utiliser pour la prédiction.

7.9.1 Structures syntaxiques supérieures

Les deux couches de flux binaire visées dans le présent paragraphe sont désignées par leur identificateur `layer_id`, qui est décodé dans la structure `sequence_scalable_extension`.

Les deux flux binaires doivent avoir des identificateurs de couche consécutifs, la couche d'amélioration étant désignée par `layer_id = idenhance` et la couche inférieure par `layer_id = idenhance-1`.

La syntaxe et la sémantique des couches d'amélioration doivent être conformes aux 6.2 et 6.3 respectivement.

Des restrictions sémantiques s'appliquent comme suit à plusieurs valeurs contenues dans les en-têtes et dans les extensions de la couche d'amélioration.

La couche inférieure doit être conforme à la présente Spécification (et non à ISO/CEI 11172-2).

En-tête de séquence

Les valeurs contenues dans cet en-tête peuvent être différentes par rapport à la couche inférieure, sauf pour les structures `horizontal_size_value`, `vertical_size_value` et `aspect_ratio_information`.

Extension de séquence

Cette extension doit être identique à celle de la couche inférieure, sauf pour les valeurs des structures `profile_and_level_indication`, `bit_rate_extension`, `vbv_buffer_size_extension`, `low_delay`, `frame_rate_extension_n` et `frame_rate_extension_d`. Ces valeurs peuvent être choisies indépendamment les unes des autres. A noter que la structure `progressive_sequence` n'indique le format de balayage que des bi-trames de la couche d'amélioration et non pas celui des bi-trames issues du multiplexage. Pour celles-ci, le format est indiqué par la structure `mux_to_progressive_sequence` (développée dans le cadre de l'extension à l'échelonnabilité des séquences).

Extension d'affichage de séquences

Cette extension ne doit pas être présente car il n'y a pas de processus d'affichage séparé pour la couche d'amélioration.

Extension à l'échelonnabilité des séquences

Cette extension doit être présente, avec le mode «échelonnabilité temporelle».

Lorsque la structure `progressive_sequence` a la valeur 0 et que `mux_to_progressive_sequence = 0`, les structures `top_field_first` et `picture_mux_factor` peuvent être sélectionnées.

Lorsque `progressive_sequence = 0` et `mux_to_progressive_sequence = 1`, la structure `top_field_first` doit contenir un complément de la valeur contenue dans la couche inférieure pour la structure `top_field_first` mais avec `picture_mux_factor = 1`.

Lorsque `progressive_sequence = 1` et `mux_to_progressive_sequence = 1`, `top_field_first` doit avoir la valeur 0 mais la structure `picture_mux_factor` peut être sélectionnée.

La combinaison des valeurs `progressive_sequence = 1` et `mux_to_progressive_sequence = 0` ne doit pas se produire.

En-tête de groupe d'images

Il n'est pas interdit que l'en-tête de groupe d'images (éventuel) soit le même dans les deux couches.

En-tête d'image

Il n'est pas interdit que l'en-tête de groupe d'images (éventuel) soit le même dans les deux couches.

Extension de codage d'image

Les valeurs contenues dans cette extension peuvent être différentes par rapport à celles de la couche inférieure, sauf pour les structures `top_field_first`, `concealment_motion_vectors` et `chroma_420_type` et `progressive_frame`. La structure `top_field_first` doit être fondée sur les valeurs de `progressive_sequence` et de `mux_to_progressive_sequence` (voir ci-dessus l'extension à l'échelonnabilité des séquences). La structure `concealment_motion_vectors` doit avoir la valeur 0. La structure `chroma_420_type` doit être identique dans les deux couches. La valeur de `progressive_frame` doit toujours être la même que dans `progressive_sequence`.

Extension à l'échelonnabilité temporelle d'images

Cette extension doit être présente pour chaque image.

Extension relative à la matrice de quantification

Cette extension peut être présente dans la couche d'amélioration.

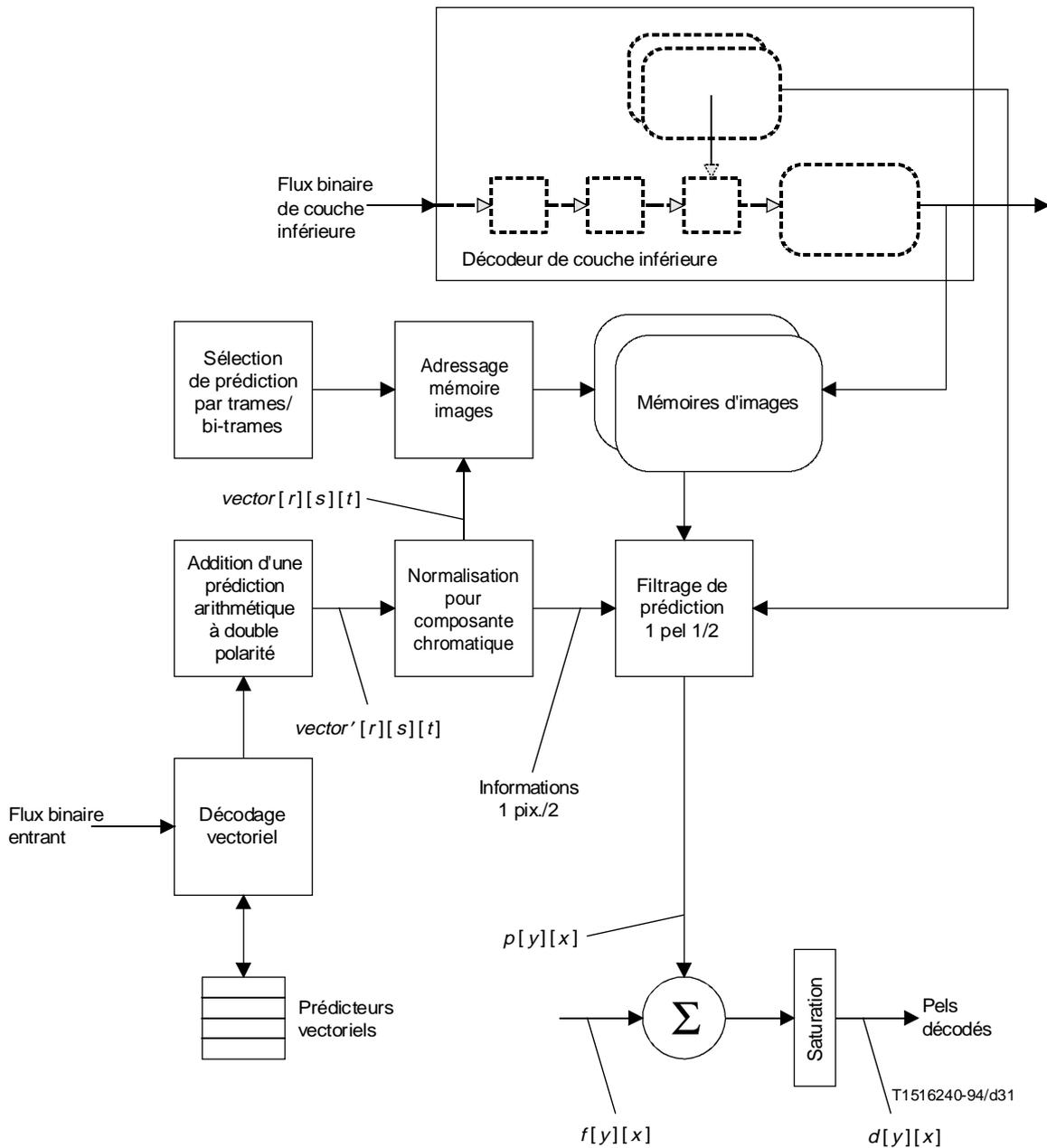


Figure 7-16 – Processus de compensation de mouvement simplifié pour la couche d'amélioration en échelonnabilité temporelle

7.9.2 Restrictions relatives à la prédiction temporelle

Bien que l'on puisse faire des prédictions temporelles sur la base d'images décodées qui sont désignées par la structure `forward_temporal_reference` ou par les deux structures `forward_temporal_reference` et `backward_temporal_reference`, l'échelonnabilité temporelle est efficace si les prédictions sont formées au moyen d'image(s) décodée(s) à partir de la couche inférieure et de la couche d'amélioration, temporellement très proches de l'image d'amélioration en cours de prédiction. Les flux binaires doivent être tels que les images P et les images B forment des prédictions sur la base des images les plus récentes ou prochaines, comme illustré par les Tableaux 7-28 et 7-29.

Si l'en-tête de groupe d'images apparaît très souvent dans la couche inférieure, une ambiguïté peut se produire à cause de la possibilité de non-unicité des références temporelles (qui sont réinitialisées à chaque en-tête de groupe d'images). Cette ambiguïté doit être résolue à l'aide d'informations issues du système pour la synchronisation des couches.

7.10 Subdivision des données

La subdivision des données est une technique qui répartit le flux binaire de données vidéo en deux couches, appelées partitions. Un point d'inflexion de priorité indique les éléments syntaxiques qui sont placés dans la partition 0, qui est la partition de base (également appelée partition de haute priorité). Le reste du flux binaire est placé en partition 1 (dite également partition de basse priorité). Les en-têtes de séquence, de groupe d'images et d'image sont copiés en redondance dans la partition. Tous les champs des en-têtes redondants doivent être identiques à ceux des originaux. Les seules extensions autorisées (et requises) dans la partition 1 sont les suivantes: `sequence_extension()`, `picture_coding_extension()` et `sequence_scalable_extension()`.

NOTE – La syntaxe de tranche indiquée au 6.2.4 est suivie dans les deux partitions jusqu'à (et y compris) l'élément syntaxique `extra_bit_slice`.

L'interprétation de l'élément `priority_breakpoint` est donnée dans le Tableau 7-30.

Tableau 7-30 – Valeurs d'inflexion de priorité et sémantèmes associés

| priority_breakpoint | Eléments syntaxiques inclus dans la partition 0 |
|---|--|
| 0 | Cette valeur est réservée pour la partition 1. Toutes les tranches de la partition 1 doivent avoir <code>priority_breakpoint = 0</code> . |
| 1 | Toutes les données indiquées par les en-têtes de séquence, de groupe d'images et par la structure <code>slice()</code> jusqu'à <code>extra_bit_slice</code> . |
| 2 | Toutes les données indiquées ci-dessus, plus les éléments syntaxiques de macrobloc jusqu'à l'élément <code>macroblock_address_increment</code> inclus. |
| 3 | Toutes les données indiquées ci-dessus, plus les éléments syntaxiques de macrobloc jusqu'à l'élément <code>coded_block_pattern()</code> (non compris). |
| 4 ... 63 | Valeur réservée. |
| 64 | Tous les éléments syntaxiques ci-dessus jusqu'à (et y compris) l'élément codé <code>coded_block_pattern()</code> ou (<code>dct_dc_differential</code>) (différence de coefficient DC), ainsi que la première paire de coefficients DCT (exécution, niveau) (ou le marqueur EOB). (Note) |
| 65 | Tous les éléments syntaxiques ci-dessus, plus au maximum 2 paires de coefficients DCT (exécution, niveau). |
| ... | |
| $63 + j$ | Tous les éléments syntaxiques ci-dessus, plus au maximum j paires de coefficients DCT (exécution, niveau). |
| ... | |
| 127 | Tous les éléments syntaxiques ci-dessus, plus au maximum 64 paires de coefficients DCT (exécution, niveau). |
| NOTE – Il n'est pas permis d'insérer un point d'inflexion de priorité immédiatement après un coefficient DC car cela pourrait provoquer l'émulation d'un code de déclenchement. | |

La sémantique du vérificateur VBV reste inchangée, c'est-à-dire que le vérificateur VBV se rapportera à la somme des deux partitions et non à une seule partition.

Les paramètres relatifs au débit du flux binaire (`bit_rate_value` et `bit_rate_extension`), à la capacité de mémoire du vérificateur VBV (`vbv_buffer_size_value` et `vbv_buffer_size_extension`) et au retard du vérificateur VBV (`vbv_delay`) doivent prendre la même valeur dans les deux partitions. Ces paramètres se rapportent aux caractéristiques du flux binaire entier, formé à partir des deux partitions.

Le processus de décodage est modifié comme suit:

- mettre à zéro l'élément `current_partition` et commencer le décodage à partir du flux binaire qui contient l'élément `sequence_scalable_extension` (partition 0);

- si *current_partition* = 0, vérifier si le point d'inflexion se trouvant dans le flux binaire est une inflexion prioritaire:
 si c'est le cas, mettre *current_partition* à 1. L'élément suivant sera décodé à partir de la partition 1;
 si ce n'est pas le cas, continuer le décodage à partir de la partition 0. Supprimer les en-têtes de séquence, de groupe d'images et d'image dans les deux partitions;
- si *current_partition* = 1, vérifier si le point d'inflexion prioritaire fait que le prochain élément à décoder doit se trouver dans la partition 0:
 si c'est le cas, mettre *current_partition* à 0. L'élément suivant sera décodé à partir de la partition 0;
 si ce n'est pas le cas, continuer le décodage à partir de la partition 1.

Un exemple est donné à la Figure 7-17, où le point d'inflexion de priorité est placé à 64 [une paire (exécution, niveau)].

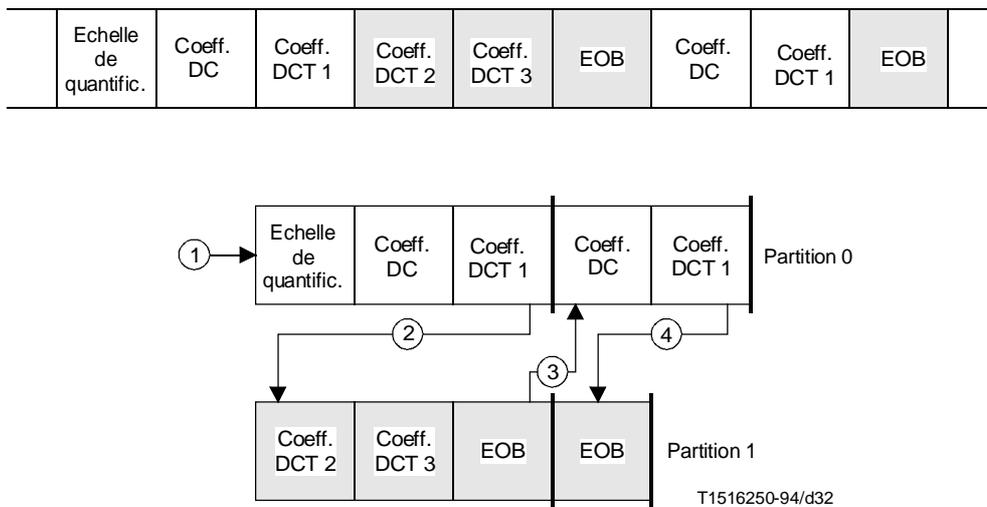


Figure 7-17 – Segment de flux binaire à deux partitions avec inflexion de priorité fixée à 64 [1 paire (exécution, niveau)] – Les 2 partitions sont assorties de flèches indiquant comment il faut commuter le décodeur entre les partitions

7.11 Echelonnabilité hybride

L'échelonnabilité hybride est l'association de deux types différents d'échelonnabilité, parmi les trois suivants: échelonnabilité SNR, échelonnabilité spatiale et échelonnabilité temporelle. Lorsque deux types d'échelonnabilité sont combinés, il y a trois flux binaires à décoder, correspondant aux trois couches désignées dans le Tableau 7-31.

Tableau 7-31 – Noms des couches

| layer_id | Nom |
|----------|-------------------------|
| 0 | Couche de base |
| 1 | Couche d'amélioration 1 |
| 2 | Couche d'amélioration 2 |
| ... | ... |

Pour l'échelonnabilité entre couches d'amélioration 1 et 2, la couche 1 est la couche inférieure et la couche 2 est sa couche d'amélioration. Aucune couche ne peut être omise de la pyramide hiérarchique. Par exemple, s'il existe un mode d'échelonnabilité SNR entre la couche d'amélioration 1 et la couche d'amélioration 2, les types de prédiction pour la couche d'amélioration 1 seront également valides pour le processus de décodage combinant les couches d'amélioration 1 et 2.

L'accouplement des couches est moins strict en échelonnabilité spatiale ou temporelle qu'en échelonnabilité SNR. Dans ces modes d'échelonnabilité, il faut donc décoder d'abord la couche de base et la suréchantillonner avant de pouvoir l'utiliser dans la couche d'amélioration. En mode d'échelonnabilité SNR, les deux couches sont décodées simultanément. L'ordre de décodage peut être résumé comme suit.

Cas 1

couche de base

<échelonnabilité spatiale ou temporelle>

couche d'amélioration 1

<échelonnabilité SNR>

couche d'amélioration 2

Décoder d'abord la couche de base, puis les deux couches d'amélioration simultanément.

Cas 2

couche de base

<échelonnabilité SNR>

couche d'amélioration 1

<échelonnabilité spatiale ou temporelle>

couche d'amélioration 2

Décoder d'abord la couche de base et la couche d'amélioration 1 simultanément, puis décoder la couche d'amélioration 2.

Cas 3

couche de base

<échelonnabilité spatiale ou temporelle>

couche d'amélioration 1

<échelonnabilité spatiale ou temporelle>

couche d'amélioration 2

Décoder d'abord la couche de base, puis la couche d'amélioration 1, puis enfin la couche d'amélioration 2.

7.12 Sortie du processus de décodage

Ce paragraphe décrit la sortie du modèle théorique représentant le processus de décodage des flux binaires conformes à la présente Spécification.

Le processus de décodage reçoit un ou plusieurs flux binaires de données vidéo codées (un pour chacune des couches). Ces couches vidéo sont généralement multiplexées au moyen d'un flux système qui contient également des informations de base de temps.

La sortie du processus de décodage se présente sous la forme d'une succession de monotrames ou de bi-trames, destinées normalement à entrer dans le processus d'affichage. L'ordre de sortie des monotrames ou des bi-trames du processus de décodage est appelé «ordre d'affichage» et peut être différent de l'ordre de codage (lorsque des images B sont utilisées). Le processus d'affichage doit présenter sur un écran les trames ou bi-trames décodées. Si le dispositif d'affichage ne peut pas suivre la fréquence d'image indiquée dans le flux binaire, le processus d'affichage peut effectuer une conversion de fréquence d'image. La présente Spécification ne décrit aucun modèle théorique représentant le processus d'affichage ou son fonctionnement.

Etant donné que certains éléments syntaxiques, tels que `progressive_frame`, peuvent être requis par le processus d'affichage, tous les éléments syntaxiques du présent modèle théorique du processus de décodage, qui sont décodés par ce processus, sont émis par celui-ci et peuvent être mis à la disposition du processus d'affichage.

Lors du décodage d'une séquence à balayage progressif (structure progressive_sequence égale à 1), les échantillons de luminance et de chrominance des bi-frames reconstruites sortent du processus de décodage sous la forme d'images bi-frames à balayage progressif, le débit de sortie étant à la fréquence d'image. La Figure 7-18 illustre cela dans le cas d'un format de chrominance égal à 4:2:0.

La même image bi-frame reconstruite est émise une seule fois si la structure repeat_first_field a la valeur 0 et deux ou trois fois de suite si repeat_first_field = 1, selon la valeur de top_field_first. La Figure 7-19 illustre cela dans le cas d'un format de chrominance de 4:2:0 et d'une structure repeat_first_field = 1.

Lors du décodage d'une séquence à balayage entrelacé (structure progressive_sequence = 0), les échantillons de luminance des images bi-frames reconstruites sont émis par le processus de décodage sous la forme de trames entrelacées, à une fréquence qui est le double de la fréquence d'image. La Figure 7-20 illustre cela.

Le flux binaire doit être tel que les trames, en sortie du processus de décodage, se présentent toujours en alternance de trame supérieure et de trame inférieure (la toute première trame d'une séquence pouvant être une trame supérieure ou inférieure).

Si l'image bi-frame reconstruite est à balayage entrelacé (structure progressive_frame égale à 0), les échantillons de luminance et de chrominance sont émis par le processus de décodage sous la forme de deux trames consécutives. La première trame issue du processus de décodage est la trame supérieure ou inférieure de l'image bi-frame reconstruite, selon la valeur de top_field_first.

Bien que tous les échantillons des images bi-frames à balayage progressif représentent le même instant, ils ne sont pas tous émis en même temps par le processus de décodage lorsque la séquence est à balayage entrelacé.

Si l'image bi-frame reconstruite est à balayage progressif (progressive_frame = 1), les échantillons de luminance sont émis par le processus de décodage sous la forme de deux ou de trois trames consécutives, selon la valeur de repeat_first_field.

NOTE – Les informations que ces trames acheminent dans le flux binaire sur la base de la même image bi-frame à balayage progressif sont transmises au processus d'affichage.

Tous les échantillons de chrominance de l'image bi-frame reconstruite sont émis par le processus de décodage en même temps que la première trame d'échantillons de luminance. C'est ce qui est illustré par les Figures 7-21 et 7-22.

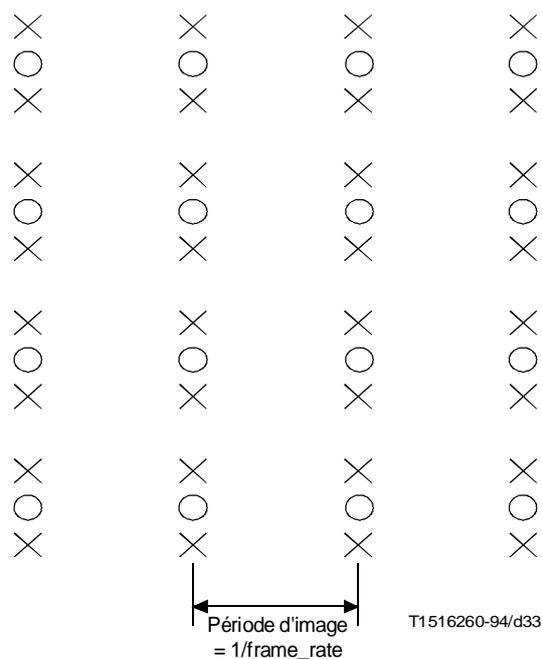


Figure 7-18 – Structure progressive_sequence == 1

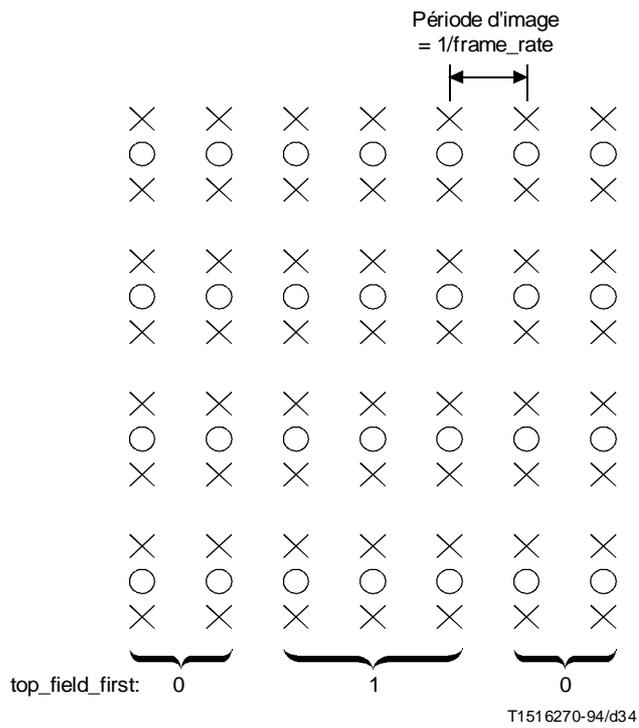


Figure 7-19 – Structures progressive_sequence == 1, repeat_first_field = 1

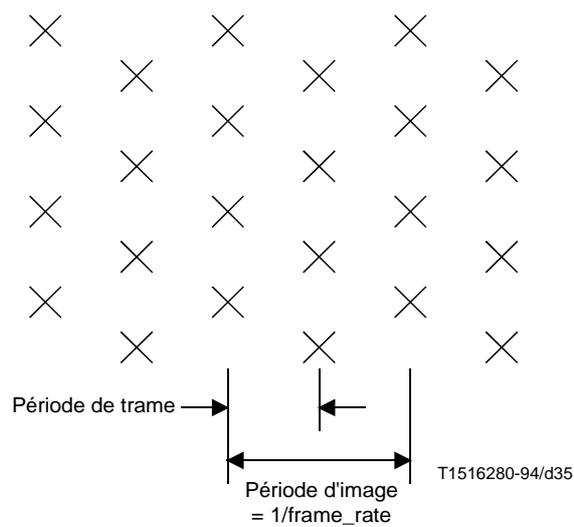


Figure 7-20 – Structure progressive_sequence == 0

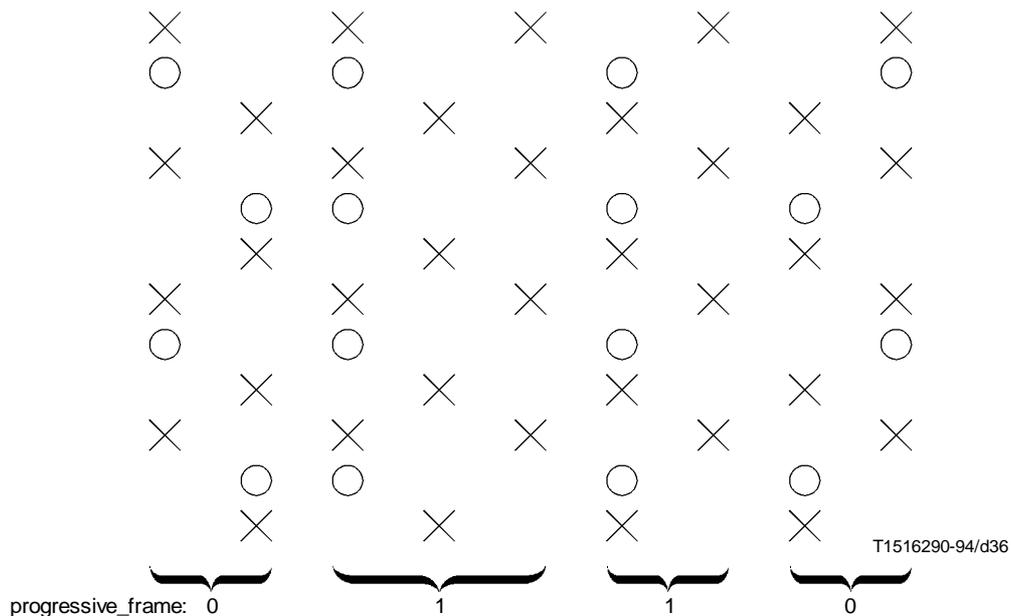


Figure 7-21 – Structure progressive_sequence == 0 avec format de chrominance 4:2:0

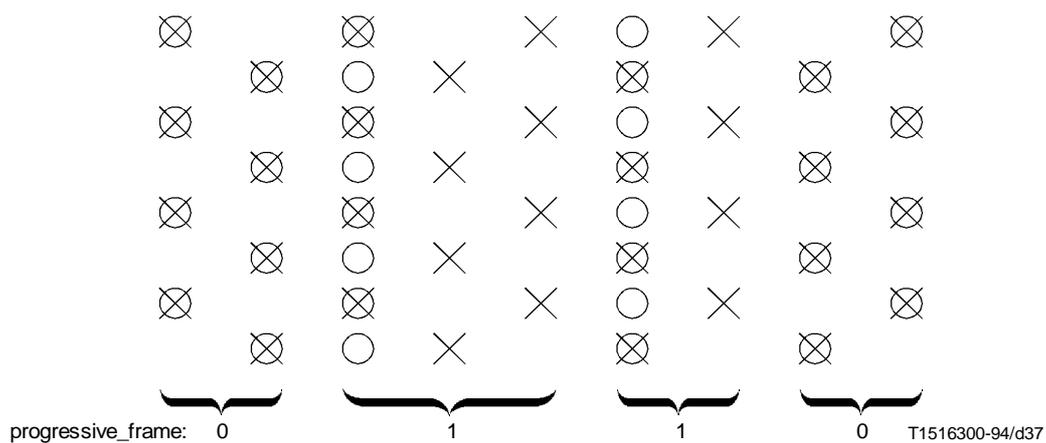


Figure 7-22 – Structure progressive_sequence == 0 avec format de chrominance 4:2:2 ou 4:4:4

8 Profils et niveaux

NOTE – Dans la présente Spécification, le terme «profil» est utilisé avec l'acception ci-après. Il convient de ne pas le confondre avec d'autres définitions du même terme et il n'a pas, en particulier, le sens qui est défini par le Groupe spécial pour la normalisation fonctionnelle (SGFS)/JTC 1.

Les profils et niveaux donnent la possibilité de définir des sous-ensembles syntaxiques et sémantiques de la présente Spécification et, ainsi, de préciser les capacités de décodage requises pour décoder un flux binaire particulier. Un profil est un sous-ensemble défini de la syntaxe totale qui est définie dans la présente Spécification pour le flux binaire. Un niveau est un ensemble défini de contraintes imposées à des paramètres contenus dans le flux binaire. Des tests de conformité seront effectués en fonction de profils définis, à des niveaux définis.

La définition de points de conformité sous la forme de profils et de niveaux a pour fonction de faciliter les échanges de flux binaires entre applications différentes. Les réalisateurs de la présente Spécification sont invités à produire les décodeurs et des flux binaires qui correspondent à ces domaines de conformité définis. Les profils et niveaux définis discrètement sont les moyens d'échanger des flux binaires entre applications conformes à la présente Spécification.

Le présent article décrit les parties contraintes des profils et niveaux définis. Tous les éléments syntaxiques et toutes les valeurs paramétriques non soumis explicitement à des contraintes peuvent prendre une valeur quelconque parmi celles qui sont autorisées par la présente Spécification. En règle générale, un décodeur doit être considéré comme conforme à un profil syntaxique donné, à un niveau de contraintes donné, s'il est en mesure de décoder correctement toutes les valeurs autorisées de tous les éléments syntaxiques spécifiés par ce profil à ce niveau. Il y a une exception à cette règle dans le cas d'un décodeur de type «profil simple, niveau principal», lequel doit être capable de décoder des flux binaires de type «profil principal, niveau inférieur». Un flux binaire doit être considéré comme conforme si les valeurs qu'il contient ne sortent pas de l'étendue autorisée et s'il ne comporte pas d'éléments syntaxiques interdits.

L'attention est appelée sur le 5.4, qui définit la convention permettant de spécifier une série de nombres. Cette convention sera utilisée chaque fois qu'il y aura lieu de spécifier une étendue de valeurs et de paramètres.

L'élément `profile_and_level_indication`, qui fait partie de l'extension de séquences (`sequence_extension`), indique le profil et le niveau auxquels le flux binaire est conforme. La signification des éléments binaires dans ce paramètre est définie dans le Tableau 8-1.

Tableau 8-1 – Signification des bits dans la structure `profile_and_level_indication`

| Bits | Longueur du champ (en bits) | Signification |
|-------|--------------------------------|--------------------------|
| [7:7] | 1 | Bit d'échappement |
| [6:4] | 3 | Identification du profil |
| [3:0] | 4 | Identification du niveau |

Le Tableau 8-2 spécifie les codes d'identification des profils et le Tableau 8-3 les codes d'identification des niveaux. Lorsque le bit d'échappement est à zéro, tout profil ayant une valeur numérique d'identification plus grande sera un sous-ensemble d'un profil ayant une valeur numérique d'identification plus petite. De même, chaque fois que le bit d'échappement est à zéro, tout niveau ayant une valeur numérique d'identification plus grande sera un sous-ensemble d'un niveau ayant une valeur numérique d'identification plus petite.

Tableau 8-2 – Identification du profil

| Identification du profil | Profil |
|--------------------------|--------------------------|
| 110 à 111 | (Valeur réservée) |
| 101 | Simple |
| 100 | Principal |
| 011 | Echelonnabilité SNR |
| 010 | Echelonnabilité spatiale |
| 001 | Supérieur |
| 000 | (Valeur réservée) |

Tableau 8-3 – Identification du niveau

| Identification du niveau | Niveau |
|--------------------------|-------------------|
| 1011 à 1111 | (Valeur réservée) |
| 1010 | Inférieur |
| 1001 | (Valeur réservée) |
| 1000 | Principal |
| 0111 | (Valeur réservée) |
| 0110 | Supérieur-1440 |
| 0101 | (Valeur réservée) |
| 0100 | Supérieur |
| 0000 à 0011 | (Valeur réservée) |

Le Tableau 8-4 décrit des profils et des niveaux correspondant à une valeur 1 du bit d'échappement. Pour ces profils et niveaux, il n'existe pas de hiérarchie implicite à partir des valeurs assignées à la structure `profile_and_level_indication`; par ailleurs, les profils et les niveaux ne sont pas nécessairement des sous-ensembles d'autres profils ou niveaux.

L'attention est appelée sur l'Annexe E, qui décrit en détail les parties de ISO/CEI 13818-2 que l'on utilise pour un profil et un niveau donnés.

Tableau 8-4 – Identification de profil et de niveau par bit d'échappement

| <code>profile_and_level_indication</code> | Nom |
|---|-------------------|
| 10000000 à 11111111 | (Valeur réservée) |

8.1 Compatibilité avec ISO/CEI 11172-2

Les «flux paramétriques contraints» selon ISO/CEI 11172-2 doivent être décodables à tous les niveaux de contrainte par des décodeurs de profil simple, principal, à échelonnabilité SNR, à échelonnabilité spatiale et de profil supérieur. Lorsqu'un flux binaire conforme au codage de flux paramétrique contraint selon ISO/CEI 11172-2 est émis, le fanion `constrained_parameters_flag` doit être activé.

De plus, les décodeurs de profil simple, principal, à échelonnabilité SNR, à échelonnabilité spatiale et de profil supérieur doivent être en mesure de décoder les flux binaires à images D seulement, selon ISO/CEI 11172-2, si ces flux sont conformes au niveau de contrainte de ces décodeurs.

8.2 Relation entre profils définis

Les décodeurs de profil simple, principal, à échelonnabilité SNR, à échelonnabilité spatiale et de profil supérieur sont en relation hiérarchique. La syntaxe gérée par un profil «supérieur» comprendra donc tous les éléments syntaxiques des profils «inférieurs» (par exemple, pour un niveau de contraintes donné, un décodeur de profil principal doit être en mesure de décoder un flux binaire conforme à des restrictions pour profil simple). Pour un profil donné, le même ensemble syntaxique est pris en charge, quel que soit le niveau de contraintes. L'ordre hiérarchique est indiqué dans le Tableau 8-2.

Les différences syntaxiques entre les contraintes des profils sont indiquées dans le Tableau 8-5. Ce tableau donne les valeurs limites applicables à un flux binaire. On notera qu'un décodeur à conformité de type profil simple doit être capable de décoder entièrement des flux binaires de type profil simple, niveau principal comme de type profil principal, niveau inférieur.

Tableau 8-5 – Contraintes syntaxiques des profils

| Elément syntaxique | Profil | | | | |
|---|---------------------------|------------------|------------------|------------------|------------------|
| | Simple | Principal | SNR | Spatial | Supérieur |
| chroma_format | 4:2:0 | 4:2:0 | 4:2:0 | 4:2:0 | 4:2:2 ou 4:2:0 |
| frame_rate_extension_n | 0 | 0 | 0 | 0 | 0 |
| frame_rate_extension_d | 0 | 0 | 0 | 0 | 0 |
| aspect_ratio_information | 0001, 0010, 0011 | 0001, 0010, 0011 | 0001, 0010, 0011 | 0001, 0010, 0011 | 0001, 0010, 0011 |
| picture_coding_type | I, P | I, P, B | I, P, B | I, P, B | I, P, B |
| repeat_first_field | Contraint | | Non contraint | | |
| sequence_scalable_extension() | Non | Non | Oui | Oui | Oui |
| scalable_mode | – | – | SNR | SNR ou spatial | SNR ou spatial |
| picture_spatial_scalable_extension() | Non | Non | Non | Oui | Oui |
| intra_dc_precision | 8, 9, 10 | 8, 9, 10 | 8, 9, 10 | 8, 9, 10 | 8, 9, 10, 11 |
| Structure de tranche | Restreinte (Voir 6.1.2.2) | | | | |

Pour tous les profils définis, une restriction sémantique s'applique au flux binaire en ce sens que toutes les données destinées à former un macrobloc doivent être représentées avec au plus le nombre d'éléments binaires indiqué dans le Tableau 8-6. Dans chaque rangée horizontale de macroblocs, deux blocs au maximum peuvent toutefois dépasser cette limite.

Tableau 8-6 – Nombre maximal de bits dans un macrobloc

| chroma_format | Nombre maximal de bits |
|----------------------|------------------------|
| 4:2:0 | 4608 |
| 4:2:2 | 6144 |
| 4:4:4 | 9216 |

Dans ce contexte, un macrobloc est censé commencer avec le premier bit de la structure `macroblock_address_increment` (ou `macroblock_escape`, le cas échéant) et continuer jusqu'au dernier bit du symbole «EOB» (fin de bloc) de la structure syntaxique `macroblock()` du dernier bloc codé (ou jusqu'au dernier bit de la structure `coded_block_pattern()` s'il n'y a pas de blocs codés). Les bits requis pour représenter une tranche quelconque précédant (ou suivant) le macrobloc ne sont pas comptés comme faisant partie du macrobloc.

Le profil syntaxique de type supérieur se distingue également par le fait qu'il véhicule diverses contraintes relatives à la fréquence des échantillons de luminance, au débit binaire maximal et à la capacité de la mémoire du vérificateur VBV. Voir les Tableaux 8-12, 8-13 et 8-14.

Les décodeurs à conformité de type profil simple @ niveau principal doivent être capables de décoder des flux binaires de type profil principal @ niveau inférieur.

8.2.1 Emploi de la structure repeat_first_field

L'utilisation de la structure repeat_first_field dans les flux binaires à profil simple et principal est soumise aux contraintes spécifiées dans le Tableau 8-7.

Tableau 8-7 – Contraintes relatives à l'emploi de la structure repeat_first_field dans les flux binaires des profils de type simple et principal

| frame_rate_code | frame_rate_value | repeat_first_field | |
|-----------------|---------------------------|---------------------------|---------------------------|
| | | progressive_sequence == 0 | progressive_sequence == 1 |
| 0000 | Valeur interdite | | |
| 0001 | 24 000 ÷ 1001 (23,976...) | 0 | 0 |
| 0010 | 24 | 0 | 0 |
| 0011 | 25 | 0 ou 1 | 0 |
| 0100 | 30 000 ÷ 1001 (29,97...) | 0 ou 1 | 0 |
| 0101 | 30 | 0 ou 1 | 0 |
| 0110 | 50 | 0 ou 1 | 0 |
| 0111 | 60 000 ÷ 1001 (59,94...) | 0 ou 1 | 0 ou 1 |
| 1000 | 60 | 0 ou 1 | 0 ou 1 |
| ... | Valeur réservée | | |
| 1111 | Valeur réservée | | |

Seuls les flux binaires de type profil principal @ niveau principal et profil simple @ niveau principal font l'objet de contraintes supplémentaires:

- si vertical_size > 480 lignes ou (frame_rate est «25 Hz») et si picture_coding_type == 011 (c'est-à-dire image de type B), il faut alors que repeat_first_field soit à la valeur 0;
- si vertical_size > 480 lignes, la fréquence d'image doit être «25 Hz».

Le profil syntaxique de type supérieur se distingue également par le fait qu'il véhicule diverses contraintes relatives à la fréquence des échantillons de luminance, au débit binaire maximal et à la capacité de la mémoire du vérificateur VBV. Voir les Tableaux 8-12, 8-13 et 8-14.

Les décodeurs à conformité de type profil simple @ niveau principal doivent être capables de décoder des flux binaires de type profil principal @ niveau inférieur.

8.3 Relation entre niveaux définis

Les niveaux inférieur, principal, supérieur-1440 et supérieur sont en relation hiérarchique. Les contraintes paramétriques d'un niveau «supérieur» sont donc égales ou supérieures aux contraintes de niveaux «inférieurs» (par exemple, pour un profil donné, un décodeur de niveau principal devra être en mesure de décoder un flux binaire conforme à des restrictions correspondant au niveau «inférieur»). L'ordre hiérarchique est indiqué dans le Tableau 8-3.

Les différentes contraintes paramétriques constituant les niveaux sont indiquées dans le Tableau 8-8.

Tableau 8-8 – Contraintes paramétriques constituant les niveaux

| Élément syntaxique | Niveau | | | |
|--|--|--------------|----------------|--------------|
| | Inférieur | Principal | Supérieur-1440 | Supérieur |
| f_code[0][0] (vecteur de mouvement horizontal anticipé) | [1:7] | [1:8] | [1:9] | [1:9] |
| f_code[1][0]^{a)} (vecteur de mouvement horizontal différé) | [1:7] | [1:8] | [1:9] | [1:9] |
| Image bitrame | | | | |
| f_code[0][1] (vecteur de mouvement vertical anticipé) | [1:4] | [1:5] | [1:5] | [1:5] |
| f_code[1][1]^{a)} (vecteur de mouvement vertical différé) | [1:4] | [1:5] | [1:5] | [1:5] |
| Etendue de vecteur de mouvement vertical ^{b)} | [-64:63,5] | [-128:127,5] | [-128:127,5] | [-128:127,5] |
| Image monotrame | | | | |
| f_code[0][1] (vecteur de mouvement vertical anticipé) | [1:3] | [1:4] | [1:4] | [1:4] |
| f_code[1][1]^{a)} (vecteur de mouvement vertical différé) | [1:3] | [1:4] | [1:4] | [1:4] |
| Etendue de vecteur de mouvement vertical ^{b)} | [-32:31,5] | [-64:63,5] | [-64:63,5] | [-64:63,5] |
| frame_rate_code | [1:5] | [1:5] | [1:8] | [1:8] |
| Densité d'échantillons | Voir Tableau 8-11 | | | |
| Fréquence des échantillons de luminance | Voir Tableau 8-12 | | | |
| Débit binaire maximal | Voir Tableau 8-13 | | | |
| Capacité de mémoire | Voir Tableau 8-14 | | | |
| a) | Pour les flux binaires de profil simple, ne comportant pas d'images B, les structures f_code[1][0] et f_code[1][1] doivent être mises à la valeur 15 (code non utilisé). | | | |
| b) | Cette restriction s'applique au vecteur de mouvement final qui a été reconstruit. Dans le cas de vecteurs de mouvements pour prédiction anticipée à double polarité, cette restriction s'applique avant l'exécution de la normalisation, après l'exécution de la normalisation et après l'adjonction du petit vecteur de mouvement différentiel. | | | |

8.4 Couches échelonnables

Les profils de type échelonnabilité SNR, échelonnabilité spatiale et supérieur peuvent utiliser plusieurs flux binaires pour coder l'image. Ces différents flux représentent des couches de codage qui, une fois combinées, créent une image de qualité supérieure à celle qui peut être obtenue à partir d'une seule couche (voir l'Annexe D). Le nombre maximal de couches pour un profil donné est spécifié au Tableau 8-9. Les couches échelonnables sont désignées conformément au Tableau 7-31. Les contraintes syntaxiques et paramétriques pour ces combinaisons de profils et de niveaux, codées avec le nombre maximal admissible de couches, sont indiquées dans les Tableaux 8-11, 8-12, 8-13 et 8-14. Lorsque le nombre de couches est inférieur au maximum admissible, il convient de consulter également les Tableaux E.21 à E.46, selon le cas.

Il y a lieu de noter que la couche de base d'un flux binaire à profil de type échelonnabilité SNR peut toujours être décodée par un décodeur à profil de type principal, de niveau équivalent. Réciproquement, un flux binaire à profil de type principal doit être décodable par un décodeur à profil SNR de niveau équivalent.

**Tableau 8-9 – Limites supérieures des couches échelonnables
en profil de type échelonnabilité SNR, échelonnabilité spatiale ou supérieur**

| Niveau | Nombre maximal de couches | Profil | | |
|----------------|--|--------|---------|-----------|
| | | SNR | Spatial | Supérieur |
| Supérieur | Toutes couches (base + amélioration) | | | 3 |
| | Couches d'amélioration à échelonnabilité spatiale | | | 1 |
| | Couches d'amélioration à échelonnabilité SNR | | | 1 |
| Supérieur-1440 | Toutes couches (base + amélioration) | | 3 | 3 |
| | Couches d'amélioration à échelonnabilité spatiale | | 1 | 1 |
| | Couches d'amélioration à échelonnabilité SNR | | 1 | 1 |
| Principal | Toutes couches (base + amélioration) | 2 | | 3 |
| | Couches d'amélioration à échelonnabilité spatiale | 0 | | 1 |
| | Couches d'amélioration à échelonnabilité SNR | 1 | | 1 |
| Inférieur | Toutes couches (base + amélioration) | 2 | | |
| | Couches d'amélioration à échelonnabilité spatiale | 0 | | |
| | Couches d'amélioration à échelonnabilité SNR | 1 | | |

8.4.1 Combinaisons admissibles des couches

Le Tableau 8-10 résume les combinaisons autorisées, avec les règles indiquées ci-après:

- profil de type échelonnabilité SNR: 2 couches au plus; profils de type échelonnabilité spatiale et supérieur: 3 couches au plus (voir le Tableau 8-9);
- un seul échelon SNR et un seul échelon spatial sont autorisés dans les combinaisons à 3 couches, l'ordre permis pouvant être SNR/spatial ou spatial/SNR (voir le Tableau 8-9);
- l'adjonction du format de chrominance 4:2:2 à une couche inférieure de format 4:2:0 est considérée comme un échelon SNR autorisant une échelonnabilité SNR ou spatiale;
- une couche de format 4:2:0 n'est pas autorisée si la couche inférieure est de format 4:2:2 (voir 7.7.3.3).

Tableau 8-10 – Combinaisons admissibles des couches

| Profil | Mode d'échelonnabilité | | | Profil/niveau du décodeur de couche de base le plus simple (couche supérieure de référence pour le niveau) ^{a)} |
|-----------|------------------------|-------------------------|-------------------------|--|
| | Couche de base | Couche d'amélioration 1 | Couche d'amélioration 2 | |
| SNR | 4:2:0 | SNR, 4:2:0 | – | MP@même niveau |
| Spatial | 4:2:0 | SNR, 4:2:0 | – | MP@même niveau |
| Spatial | 4:2:0 | Spatial, 4:2:0 | – | MP@(niveau – 1) |
| Spatial | 4:2:0 | SNR, 4:2:0 | Spatial, 4:2:0 | MP@(niveau – 1) |
| Spatial | 4:2:0 | Spatial, 4:2:0 | SNR, 4:2:0 | MP@(niveau – 1) |
| Supérieur | 4:2:0 | – | – | HP@même niveau |
| Supérieur | 4:2:2 | – | – | HP@même niveau |
| Supérieur | 4:2:0 | SNR, 4:2:0 | – | HP@même niveau |
| Supérieur | 4:2:0 | SNR, 4:2:2 | – | HP@même niveau |
| Supérieur | 4:2:2 | SNR, 4:2:2 | – | HP@même niveau |
| Supérieur | 4:2:0 | Spatial, 4:2:0 | – | HP@(niveau – 1) |
| Supérieur | 4:2:0 | Spatial, 4:2:2 | – | HP@(niveau – 1) |
| Supérieur | 4:2:2 | Spatial, 4:2:2 | – | HP@(niveau – 1) ^{b)} |
| Supérieur | 4:2:0 | SNR, 4:2:0 | Spatial, 4:2:0 | HP@(niveau – 1) |
| Supérieur | 4:2:0 | SNR, 4:2:0 | Spatial, 4:2:2 | HP@(niveau – 1) |
| Supérieur | 4:2:0 | SNR, 4:2:2 | Spatial, 4:2:2 | HP@(niveau – 1) ^{b)} |
| Supérieur | 4:2:2 | SNR, 4:2:2 | Spatial, 4:2:2 | HP@(niveau – 1) ^{b)} |
| Supérieur | 4:2:0 | Spatial, 4:2:0 | SNR, 4:2:0 | HP@(niveau – 1) |
| Supérieur | 4:2:0 | Spatial, 4:2:0 | SNR, 4:2:2 | HP@(niveau – 1) |
| Supérieur | 4:2:0 | Spatial, 4:2:2 | SNR, 4:2:2 | HP@(niveau – 1) |
| Supérieur | 4:2:2 | Spatial, 4:2:2 | SNR, 4:2:2 | HP@(niveau – 1) ^{b)} |

^{a)} On spécifie ici le plus simple décodeur conforme, en supposant que le flux binaire peut contenir toute valeur de paramètre et tout élément syntaxique autorisés pour le profil @ niveau indiqués, sauf l'échelonnabilité. On notera que pour les flux binaires à échelonnabilité spatiale de type profil supérieur @ niveau principal, la combinaison «HP @ (niveau – 1)» devient «MP @ (niveau – 1)». Lorsqu'un flux binaire de la couche de base utilise un moins grand nombre d'éléments syntaxiques ou un domaine de variation restreint des paramètres, par comparaison aux valeurs autorisées, le profil et le niveau peuvent indiquer le recours à une combinaison «profil plus simple @ niveau».

^{b)} Le format de chrominance 4:2:2 n'est pas autorisé pour une couche spatiale inférieure de la combinaison «profil supérieur @ niveau principal» (voir le Tableau 8-12).

Les Tableaux E.20 à E.45 indiquent de façon détaillée les différentes restrictions paramétriques applicables dans chaque couche d'un flux binaire, ainsi que les valeurs correspondantes appropriées de la structure `profile_and_level_indication`.

8.5 Valeurs paramétriques pour profils, niveaux et couches définis

Voir le Tableau 8-11.

Tableau 8-11 – Limites supérieures de densité d'échantillonnage

| Niveau | Couche de résolution spatiale | | Profil | | | | |
|----------------|-------------------------------|---|------------------|--------------------|------------------|--------------------|--------------------|
| | | | Simple | Principal | SNR | Spatial | Supérieur |
| Supérieur | Amélioration | Echantil./ligne Lignes/bi-trame Bi-trames/s | | 1920 1152 60 | | | 1920 1152 60 |
| | Inférieure | Echantil./ligne Lignes/bi-trame Bi-trames/s | | – | | | 960 576 30 |
| Supérieur-1440 | Amélioration | Echantil./ligne Lignes/bi-trame Bi-trames/s | | 1440 1152 60 | | 1440 1152 60 | 1440 1152 60 |
| | Inférieure | Echantil./ligne Lignes/bi-trame Bi-trames/s | | – | | 720 576 30 | 720 576 30 |
| Principal | Amélioration | Echantil./ligne Lignes/bi-trame Bi-trames/s | 720 576 30 | 720 576 30 | 720 576 30 | | 720 576 30 |
| | Inférieure | Echantil./ligne Lignes/bi-trame Bi-trames/s | – | – | – | | 352 288 30 |
| Inférieur | Amélioration | Echantil./ligne Lignes/bi-trame Bi-trames/s | | 352 288 30 | 352 288 30 | | |
| | Inférieure | Echantil./ligne Lignes/bi-trame Bi-trames/s | | – | – | | |

NOTE – Dans le cas du codage monocouche ou échelonné SNR, les limites spécifiées par le terme «couche d'amélioration» sont applicables.

Les éléments syntaxiques visés par ce tableau sont les suivants:

- échantillons/ligne: horizontal_size;
- lignes/bi-trame: vertical_size;
- bi-trames/s: frame_rate.

La limite supérieure de la fréquence d'image est la même, indépendamment de la valeur progressive_sequence = 0 et progressive_sequence = 1.

Tableau 8-12 – Limites supérieures pour la fréquence des échantillons de luminance (échantillons/s)

| Niveau | Couche de résolution spatiale | Profil | | | | |
|----------------|-------------------------------|------------|------------|------------|------------|--|
| | | Simple | Principal | SNR | Spatial | Supérieur |
| Supérieur | Amélioration | | 62 668 800 | | | 62 668 800 (4:2:2) 83 558 400 (4:2:0) |
| | Inférieure | | – | | | 14 745 600 (4:2:2) 19 660 800 (4:2:0) |
| Supérieur-1440 | Amélioration | | 47 001 600 | | 47 001 600 | 47 001 600 (4:2:2) 62 668 800 (4:2:0) |
| | Inférieure | | – | | 10 368 000 | 11 059 200 (4:2:2) 14 745 600 (4:2:0) |
| Principal | Amélioration | 10 368 000 | 10 368 000 | 10 368 000 | | 11 059 200 (4:2:2) 14 745 600 (4:2:0) |
| | Inférieure | – | – | – | | – 3 041 280 (4:2:0) |
| Inférieur | Amélioration | | 3 041 280 | 3 041 280 | | |
| | Inférieure | | – | – | | |

NOTE – Dans le cas du codage monocouche ou échelonné SNR, les limites spécifiées par le terme «couche d'amélioration» sont applicables.

La fréquence des échantillons de luminance, P, est définie comme suit:

- Si `progressive_sequence == 1`:

$$P = (16 * ((horizontal_size + 15) / 16)) \times (16 * ((vertical_size + 15) / 16)) \times frame_rate$$
- Si `progressive_sequence == 0`:

$$P = (16 * ((horizontal_size + 15) / 16)) \times (32 * ((vertical_size + 31) / 32)) \times frame_rate$$

Tableau 8-13 – Limites supérieures des débits binaires (Mbit/s)

| Niveau | Profil | | | | |
|----------------|--------|-----------|---|---|--|
| | Simple | Principal | SNR | Spatial | Supérieur |
| Supérieur | | 80 | | | 100 toutes couches 80 couche médiane + couche de base 25 couche de base |
| Supérieur-1440 | | 60 | | 60 toutes couches 40 couche médiane + couche de base 15 couche de base | 80 toutes couches 60 couche médiane + couche de base 20 couche de base |
| Principal | 15 | 15 | – 15 pour les deux couches 10 couche de base | | 20 toutes couches 15 couche médiane + couche de base 4 couche de base |
| Inférieur | | 4 | – 4 pour les deux couches 3 couche de base | | |

NOTES

- Ce tableau définit le débit maximal de fonctionnement du vérificateur VBV pour un flux binaire de données codées d'un profil et d'un niveau donnés. Ce débit est indiqué sous forme de débit binaire (voir 6.3.3).
- Ce tableau définit le débit binaire maximal admissible pour toutes les couches jusqu'à la couche indiquée (comprise). Pour les applications de codage multicouche, le débit binaire assigné à chaque couche n'est contraint que par le débit maximal autorisé pour chaque couche, comme indiqué dans ce tableau.
- 1 Mbit = 1 000 000 bits.

Tableau 8-14 – Capacité de mémoire requise pour le vérificateur VBV (en bits)

| Niveau | Couche | Profil | | | | |
|----------------|--|-----------|-----------|-----------------------------|-------------------------------------|--------------------------------------|
| | | Simple | Principal | SNR | Spatial | Supérieur |
| Supérieur | Amélioration 2 Amélioration 1 Base | | 9 781 248 | | | 12 222 464 9 781 248 3 047 424 |
| Supérieur-1440 | Amélioration 2 Amélioration 1 Base | | 7 340 032 | | 7 340 032 4 882 432 1 835 008 | 9 781 248 7 340 032 2 441 216 |
| Principal | Amélioration 2 Amélioration 1 Base | 1 835 008 | 1 835 008 | – 1 835 008 1 212 416 | | 2 441 216 1 835 008 475 136 |
| Inférieur | Amélioration 2 Amélioration 1 Base | | 475 136 | – 475 136 360 448 | | |

NOTES

- La capacité de mémoire est calculée de façon à être proportionnelle au débit binaire maximal admissible, *arrondi par défaut* au plus proche multiple de 16×1024 bits. La valeur de référence pour la normalisation est la capacité mémoire pour la combinaison profil principal, niveau principal.
- Ce tableau définit la capacité *totale* de mémoire du décodeur qui est requise pour décoder toutes les couches jusqu'à la couche indiquée (comprise). Pour les applications de codage multicouche, la répartition de la mémoire tampon entre les couches n'est contrainte que par la capacité maximale autorisée pour une couche donnée, comme indiqué dans ce tableau.
- L'élément syntaxique correspondant à ce tableau est `vbv_buffer_size` (voir 6.3.3).

Tableau 8-15 – Compatibilité descendante entre les différents profils et niveaux

| Indication de profil et de niveau dans le flux binaire | Décodeur | | | | | | | | | | |
|---|----------|-----------|---------|----------------|----------|----------|---------|-----------|---------|---------|-----------------|
| | HP @ HL | HP @ H-14 | HP @ ML | Spatial @ H-14 | SNR @ ML | SNR @ LL | MP @ HL | MP @ H-14 | MP @ ML | MP @ LL | SP @ ML |
| HP@HL | X | | | | | | | | | | |
| HP@H-14 | X | X | | | | | | | | | |
| HP@ML | X | X | X | | | | | | | | |
| Spatial@H-14 | X | X | | X | | | | | | | |
| SNR @ML | X | X | X | X | X | | | | | | |
| SNR @LL | X | X | X | X | X | X | | | | | |
| MP@HL | X | | | | | | X | | | | |
| MP@H-14 | X | X | | X | | | X | X | | | |
| MP@ML | X | X | X | X | X | | X | X | X | | |
| MP@LL | X | X | X | X | X | X | X | X | X | X | X ^{a)} |
| SP@ML | X | X | X | X | X | | X | X | X | | X |
| ISO/CEI 11172 | X | X | X | X | X | X | X | X | X | X | X |
| X Indique que le décodeur doit être capable de décoder le flux binaire, y compris toutes les couches inférieures correspondantes. | | | | | | | | | | | |
| ^{a)} Noter que les décodeurs de type SP @ ML doivent pouvoir décoder des flux binaires de type MP @ LL. | | | | | | | | | | | |

NOTE – Pour les profils et les niveaux structurés hiérarchiquement, il est recommandé que chaque couche du flux binaire contienne l'élément syntaxique `profile_and_level_indication` du décodeur «le plus simple» qui soit capable de décoder correctement cette couche du flux binaire. Si le bit d'échappement de l'élément `profile_and_level_indication` est à 0, la plus grande valeur numérique, valide et possible de cet élément sera applicable.

Annexe A

Transformation en cosinus discrète

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

La transformation en cosinus discrète (DCT) à deux dimensions ($N \times N$) est définie comme suit:

$$F(u; v) = \frac{2}{N} C(u)C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x; y) \cos \frac{(2x+1)u\pi}{2N} \cos \frac{(2y+1)v\pi}{2N}$$

avec $u, v, x, y = 0, 1, 2, \dots, N-1$

où x, y sont des coordonnées spatiales dans le domaine des échantillons

u, v sont des coordonnées dans le domaine de transformation

$$C(u); C(v) = \begin{cases} 1 \\ \sqrt{2} \end{cases} \text{ pour } u, v = 0, 1, \text{ sinon}$$

La transformation DCT inverse (IDCT) est définie comme suit:

$$f(x; y) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u; v) \cos \frac{(2x+1)u\pi}{2N} \cos \frac{(2y+1)v\pi}{2N}$$

Les données d'entrée dans l'algorithme de transformation directe et les données de sortie de la transformation inverse sont représentées sur 9 bits. Les coefficients sont représentés sur 12 éléments binaires. L'étendue dynamique des coefficients DCT est $[-2048; +2047]$.

La transformation discrète inverse de dimensions $N \times N$ doit être conforme à l'IEEE 1180-1990 du 6 décembre 1990 concernant les applications de la transformation en cosinus discrète inverse de dimensions 8×8 .

NOTES

1 Le paragraphe 2.3 de IEEE 1180-1990, concernant la «spécification des erreurs d'adaptation après transformation IDCT», prescrit la spécification d'un codage intra périodique afin de limiter l'accumulation d'erreurs d'adaptation. Chaque macrobloc doit être rafraîchi avant d'être codé 132 fois en tant que macrobloc à codage prédictif. Les macroblocs des images de type B (et les macroblocs sautés) sont exclus du décompte, parce qu'ils n'entraînent pas l'accumulation d'erreurs d'adaptation; cette exigence est identique à celle indiquée dans IEEE 1180-1990 pour la visiophonie, selon la Recommandation H.261.

2 La condition de conformité à l'IEEE IDT est une condition nécessaire pour l'application satisfaisante de la transformée IDCT, mais ne constitue pas une condition suffisante. Il convient d'attirer plus particulièrement l'attention sur la phrase suivante du 5.4: «Si la précision arithmétique n'est pas spécifiée, comme dans le calcul de la transformée IDCT, la précision doit être suffisante pour que des erreurs significatives ne se produisent pas dans les valeurs finales d'entiers.»

Annexe B

Tables des codes à longueur variable

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

B.1 Adressage des macroblocs

Voir le Tableau B.1.

Tableau B.1 – Codes à longueur variable pour la structure `macroblock_address_increment`

| Code VLC pour structure <code>macroblock_address_increment</code> | Valeur de progression | Code VLC pour structure <code>macroblock_address_increment</code> | Valeur de progression |
|---|-----------------------|---|--------------------------------|
| 1 | 1 | 0000 0101 01 | 18 |
| 011 | 2 | 0000 0101 00 | 19 |
| 010 | 3 | 0000 0100 11 | 20 |
| 0011 | 4 | 0000 0100 10 | 21 |
| 0010 | 5 | 0000 0100 011 | 22 |
| 0001 1 | 6 | 0000 0100 010 | 23 |
| 0001 0 | 7 | 0000 0100 001 | 24 |
| 0000 111 | 8 | 0000 0100 000 | 25 |
| 0000 110 | 9 | 0000 0011 111 | 26 |
| 0000 1011 | 10 | 0000 0011 110 | 27 |
| 0000 1010 | 11 | 0000 0011 101 | 28 |
| 0000 1001 | 12 | 0000 0011 100 | 29 |
| 0000 1000 | 13 | 0000 0011 011 | 30 |
| 0000 0111 | 14 | 0000 0011 010 | 31 |
| 0000 0110 | 15 | 0000 0011 001 | 32 |
| 0000 0101 11 | 16 | 0000 0011 000 | 33 |
| 0000 0101 10 | 17 | 0000 0001 000 | <code>macroblock_escape</code> |

NOTE – L'entrée intitulée «bourrage de macrobloc» dans ISO/CEI 11172-2 n'est pas disponible dans la présente Spécification.

B.2 Type de macrobloc

Les caractéristiques des macroblocs sont déterminées par la structure à codage VLC macroblock_type, conformément aux Tableaux B.2 à B.8.

Tableau B.2 – Codes à longueur variable pour macroblock_type dans les images I

| Code VLC pour macroblock_type | | | | | | | | |
|---|---|---|---|---|---|---|-------------------------|---|
| macroblock_quant | | | | | | | | |
| macroblock_motion_forward | | | | | | | | |
| macroblock_motion_backward | | | | | | | | |
| macroblock_pattern | | | | | | | | |
| macroblock_intra | | | | | | | | |
| spatial_temporal_weight_code_flag | | | | | | | | |
| permitted spatial_temporal_weight_classes (classes autorisées) | | | | | | | | |
| Description | | | | | | | | |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | Codage intra | 0 |
| 01 | 1 | 0 | 0 | 0 | 1 | 0 | Codage intra, quantifié | 0 |

Tableau B.3 – Codes à longueur variable pour structure macroblock_type dans les images P

| Code VLC pour la structure macroblock_type | | | | | | | | |
|---|---|---|---|---|---|---|---------------------------------|---|
| macroblock_quant | | | | | | | | |
| macroblock_motion_forward | | | | | | | | |
| macroblock_motion_backward | | | | | | | | |
| macroblock_pattern | | | | | | | | |
| macroblock_intra | | | | | | | | |
| spatial_temporal_weight_code_flag | | | | | | | | |
| permitted spatial_temporal_weight_classes (classes autorisées) | | | | | | | | |
| Description | | | | | | | | |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | MC (à comp. de mouvement), Codé | 0 |
| 01 | 0 | 0 | 0 | 1 | 0 | 0 | Non MC, Codé | 0 |
| 001 | 0 | 1 | 0 | 0 | 0 | 0 | MC, Non codé | 0 |
| 0001 1 | 0 | 0 | 0 | 0 | 1 | 0 | Codage intra | 0 |
| 0001 0 | 1 | 1 | 0 | 1 | 0 | 0 | MC, Codé, Quantifié | 0 |
| 0000 1 | 1 | 0 | 0 | 1 | 0 | 0 | Non MC, Codé, Quantifié | 0 |
| 0000 01 | 1 | 0 | 0 | 0 | 1 | 0 | Codage intra, Quantifié | 0 |

Tableau B.4 – Codes à longueur variable pour structure macroblock_type dans les images B

| Code VLC pour la structure macroblock_type | | | | | | | | |
|---|---|---|---|---|---|---|------------------------------------|---|
| macroblock_quant | | | | | | | | |
| macroblock_motion_forward | | | | | | | | |
| macroblock_motion_backward | | | | | | | | |
| macroblock_pattern | | | | | | | | |
| macroblock_intra | | | | | | | | |
| spatial_temporal_weight_code_flag | | | | | | | | |
| permitted spatial_temporal_weight_classes (classes autorisées) | | | | | | | | |
| Description | | | | | | | | |
| 10 | 0 | 1 | 1 | 0 | 0 | 0 | Interp, Non codé | 0 |
| 11 | 0 | 1 | 1 | 1 | 0 | 0 | Interp, Codé | 0 |
| 010 | 0 | 0 | 1 | 0 | 0 | 0 | A préd. différée, Non codé | 0 |
| 011 | 0 | 0 | 1 | 1 | 0 | 0 | A préd. différée, Codé | 0 |
| 0010 | 0 | 1 | 0 | 0 | 0 | 0 | A préd. anticipée, Non codé | 0 |
| 0011 | 0 | 1 | 0 | 1 | 0 | 0 | A préd. anticipée, Codé | 0 |
| 0001 1 | 0 | 0 | 0 | 0 | 1 | 0 | Codage intra | 0 |
| 0001 0 | 1 | 1 | 1 | 1 | 0 | 0 | Interp, Codé, Quantifié | 0 |
| 0000 11 | 1 | 1 | 0 | 1 | 0 | 0 | A préd. anticipée, Codé, Quantifié | 0 |
| 0000 10 | 1 | 0 | 1 | 1 | 0 | 0 | A préd. différée, Codé, Quantifié | 0 |
| 0000 01 | 1 | 0 | 0 | 0 | 1 | 0 | Codage intra, Quantifié | 0 |

Tableau B.5 – Codes à longueur variable pour structure macroblock_type dans les images I en échelonnabilité spatiale

| Code VLC pour la structure macroblock_type | | | | | | | | |
|---|---|---|---|---|---|---|-----------------------------|---|
| macroblock_quant | | | | | | | | |
| macroblock_motion_forward | | | | | | | | |
| macroblock_motion_backward | | | | | | | | |
| macroblock_pattern | | | | | | | | |
| macroblock_intra | | | | | | | | |
| spatial_temporal_weight_code_flag | | | | | | | | |
| permitted spatial_temporal_weight_classes (classes autorisées) | | | | | | | | |
| Description | | | | | | | | |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | Codé, Compatible | 4 |
| 01 | 1 | 0 | 0 | 1 | 0 | 0 | Codé, Compatible, Quantifié | 4 |
| 0011 | 0 | 0 | 0 | 0 | 1 | 0 | Codage intra | 0 |
| 0010 | 1 | 0 | 0 | 0 | 1 | 0 | Codage intra, Quantifié | 0 |
| 0001 | 0 | 0 | 0 | 0 | 0 | 0 | Non codé, Compatible | 4 |

**Tableau B.6 – Codes à longueur variable pour structure macroblock_type
dans les images P en échelonnabilité spatiale**

| Code VLC pour la structure macroblock_type | | | | | | | | |
|--|------------------|---------------------------|----------------------------|--------------------|------------------|---|-------------------------------------|-------|
| | macroblock_quant | | | | | | | |
| | | macroblock_motion_forward | | | | | | |
| | | | macroblock_motion_backward | | | | | |
| | | | | macroblock_pattern | | | | |
| | | | | | macroblock_intra | | | |
| | | | | | | spatial_temporal_weight_code_flag | | |
| | | | | | | permitted spatial_temporal_weight_classes (classes autorisées) | | |
| | | | | | | | Description | |
| 10 | 0 | 1 | 0 | 1 | 0 | 0 | MC, Codé | 0 |
| 011 | 0 | 1 | 0 | 1 | 0 | 1 | MC, Codé, Compatible | 1,2,3 |
| 0000 100 | 0 | 0 | 0 | 1 | 0 | 0 | Non MC, Codé | 0 |
| 0001 11 | 0 | 0 | 0 | 1 | 0 | 1 | Non MC, Codé, Compatible | 1,2,3 |
| 0010 | 0 | 1 | 0 | 0 | 0 | 0 | MC, Non codé | 0 |
| 0000 111 | 0 | 0 | 0 | 0 | 1 | 0 | Codage intra | 0 |
| 0011 | 0 | 1 | 0 | 0 | 0 | 1 | MC, Non codé, Compatible | 1,2,3 |
| 010 | 1 | 1 | 0 | 1 | 0 | 0 | MC, Codé, Quantifié | 0 |
| 0001 00 | 1 | 0 | 0 | 1 | 0 | 0 | Non MC, Codé, Quantifié | 0 |
| 0000 110 | 1 | 0 | 0 | 0 | 1 | 0 | Codage intra, Quantifié | 0 |
| 11 | 1 | 1 | 0 | 1 | 0 | 1 | MC, Codé, Compatible, Quantifié | 1,2,3 |
| 0001 01 | 1 | 0 | 0 | 1 | 0 | 1 | Non MC, Codé, Compatible, Quantifié | 1,2,3 |
| 0001 10 | 0 | 0 | 0 | 0 | 0 | 1 | Non MC, Non codé, Compatible | 1,2,3 |
| 0000 101 | 0 | 0 | 0 | 1 | 0 | 0 | Codé, Compatible | 4 |
| 0000 010 | 1 | 0 | 0 | 1 | 0 | 0 | Codé, Compatible, Quantifié | 4 |
| 0000 011 | 0 | 0 | 0 | 0 | 0 | 0 | Non codé, Compatible | 4 |

**Tableau B.7 – Codes à longueur variable pour structure macroblock_type
dans les images B en échelonnabilité spatiale**

| Code VLC pour la structure macroblock_type | | | | | | | | |
|---|---|---|---|---|---|---|--|-------|
| macroblock_quant | | | | | | | | |
| macroblock_motion_forward | | | | | | | | |
| macroblock_motion_backward | | | | | | | | |
| macroblock_pattern | | | | | | | | |
| macroblock_intra | | | | | | | | |
| spatial_temporal_weight_code_flag | | | | | | | | |
| permitted spatial_temporal_weight_classes (classes autorisées) | | | | | | | | |
| | | | | | | | Description | |
| 10 | 0 | 1 | 1 | 0 | 0 | 0 | Interp, Non codé | 0 |
| 11 | 0 | 1 | 1 | 1 | 0 | 0 | Interp, Codé | 0 |
| 010 | 0 | 0 | 1 | 0 | 0 | 0 | Préd. différée, Non codé | 0 |
| 011 | 0 | 0 | 1 | 1 | 0 | 0 | Préd. différée, Codé | 0 |
| 0010 | 0 | 1 | 0 | 0 | 0 | 0 | Préd. anticipée, Non codé | 0 |
| 0011 | 0 | 1 | 0 | 1 | 0 | 0 | Préd. anticipée, Codé | 0 |
| 0001 10 | 0 | 0 | 1 | 0 | 0 | 1 | Préd. différée, Non codé, Compatible | 1,2,3 |
| 0001 11 | 0 | 0 | 1 | 1 | 0 | 1 | Préd. différée, Codé, Compatible | 1,2,3 |
| 0001 00 | 0 | 1 | 0 | 0 | 0 | 1 | Préd. anticipée, Non codé, Compatible | 1,2,3 |
| 0001 01 | 0 | 1 | 0 | 1 | 0 | 1 | Préd. anticipée, Codé, Compatible | 1,2,3 |
| 0000 110 | 0 | 0 | 0 | 0 | 1 | 0 | Codage intra | 0 |
| 0000 111 | 1 | 1 | 1 | 1 | 0 | 0 | Interp, Codé, Quantifié | 0 |
| 0000 100 | 1 | 1 | 0 | 1 | 0 | 0 | Préd. anticipée, Codé, Quantifié | 0 |
| 0000 101 | 1 | 0 | 1 | 1 | 0 | 0 | Préd. différée, Codé, Quantifié | 0 |
| 0000 0100 | 1 | 0 | 0 | 0 | 1 | 0 | Codage intra, Quantifié | 0 |
| 0000 0101 | 1 | 1 | 0 | 1 | 0 | 1 | Préd. anticipée, Codé, Compatible, Quantifié | 1,2,3 |
| 0000 0110 0 | 1 | 0 | 1 | 1 | 0 | 1 | Préd. différée, Codé, Compatible, Quantifié | 1,2,3 |
| 0000 0111 0 | 0 | 0 | 0 | 0 | 0 | 0 | Non codé, Compatible | 4 |
| 0000 0110 1 | 1 | 0 | 0 | 1 | 0 | 0 | Codé, Compatible, Quantifié | 4 |
| 0000 0111 1 | 0 | 0 | 0 | 1 | 0 | 0 | Codé, Compatible | 4 |

Tableau B.8 – Codes à longueur variable pour structure macroblock_type dans les images I, P et B avec échelonnabilité SNR

| Code VLC pour la structure macroblock_type | | | | | | | | |
|--|---|------------------|---------------------------|---|---|---|-----------------|---|
| | | macroblock_quant | | | | | | |
| | | | macroblock_motion_forward | | | | | |
| | | | | macroblock_motion_backward | | | | |
| | | | | macroblock_pattern | | | | |
| | | | | macroblock_intra | | | | |
| | | | | spatial_temporal_weight_code_flag | | | | |
| | | | | permitted spatial_temporal_weight_classes (classes autorisées) | | | | |
| | | | | Description | | | | |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | Codé | 0 |
| 01 | 1 | 0 | 0 | 1 | 0 | 0 | Codé, Quantifié | 0 |
| 001 | 0 | 0 | 0 | 0 | 0 | 0 | Non codé | 0 |

NOTE – Il n'y a pas de différenciation entre types d'image car les macroblocs sont traités de la même manière dans les images I, P et B. Le type «non codé» est nécessaire car les blocs sautés ne sont pas autorisés en début et en fin de tranche.

B.3 Structure des macroblocs

Voir le Tableau B.9.

Tableau B.9 – Codes à longueur variable pour structure coded_block_pattern

| Code VLC pour structure coded_block_pattern | Structure de bloc codée (cbp) | Code VLC pour structure coded_block_pattern | Structure de bloc codée (cbp) |
|--|-------------------------------|---|-------------------------------|
| 111 | 60 | 0001 1100 | 35 |
| 1101 | 4 | 0001 1011 | 13 |
| 1100 | 8 | 0001 1010 | 49 |
| 1011 | 16 | 0001 1001 | 21 |
| 1010 | 32 | 0001 1000 | 41 |
| 1001 1 | 12 | 0001 0111 | 14 |
| 1001 0 | 48 | 0001 0110 | 50 |
| 1000 1 | 20 | 0001 0101 | 22 |
| 1000 0 | 40 | 0001 0100 | 42 |
| 0111 1 | 28 | 0001 0011 | 15 |
| 0111 0 | 44 | 0001 0010 | 51 |
| 0110 1 | 52 | 0001 0001 | 23 |
| 0110 0 | 56 | 0001 0000 | 43 |
| 0101 1 | 1 | 0000 1111 | 25 |
| 0101 0 | 61 | 0000 1110 | 37 |
| 0100 1 | 2 | 0000 1101 | 26 |
| 0100 0 | 62 | 0000 1100 | 38 |
| 0011 11 | 24 | 0000 1011 | 29 |
| 0011 10 | 36 | 0000 1010 | 45 |
| 0011 01 | 3 | 0000 1001 | 53 |
| 0011 00 | 63 | 0000 1000 | 57 |
| 0010 111 | 5 | 0000 0111 | 30 |
| 0010 110 | 9 | 0000 0110 | 46 |
| 0010 101 | 17 | 0000 0101 | 54 |
| 0010 100 | 33 | 0000 0100 | 58 |
| 0010 011 | 6 | 0000 0011 1 | 31 |
| 0010 010 | 10 | 0000 0011 0 | 47 |
| 0010 001 | 18 | 0000 0010 1 | 55 |
| 0010 000 | 34 | 0000 0010 0 | 59 |
| 0001 1111 | 7 | 0000 0001 1 | 27 |
| 0001 1110 | 11 | 0000 0001 0 | 39 |
| 0001 1101 | 19 | 0000 0000 1 | 0 (Note) |
| NOTE – Cette entrée ne doit pas être utilisée avec un format de chrominance 4:2:0. | | | |

B.4 Vecteurs de mouvement

Voir les Tableaux B.10 et B.11.

Tableau B.10 – Codes à longueur variable pour structure motion_code

| Code à longueur variable | motion_code[r][s][t] |
|--------------------------|----------------------|
| 0000 0011 001 | -16 |
| 0000 0011 011 | -15 |
| 0000 0011 101 | -14 |
| 0000 0011 111 | -13 |
| 0000 0100 001 | -12 |
| 0000 0100 011 | -11 |
| 0000 0100 11 | -10 |
| 0000 0101 01 | -9 |
| 0000 0101 11 | -8 |
| 0000 0111 | -7 |
| 0000 1001 | -6 |
| 0000 1011 | -5 |
| 0000 111 | -4 |
| 0001 1 | -3 |
| 0011 | -2 |
| 011 | -1 |
| 1 | 0 |
| 010 | 1 |
| 0010 | 2 |
| 0001 0 | 3 |
| 0000 110 | 4 |
| 0000 1010 | 5 |
| 0000 1000 | 6 |
| 0000 0110 | 7 |
| 0000 0101 10 | 8 |
| 0000 0101 00 | 9 |
| 0000 0100 10 | 10 |
| 0000 0100 010 | 11 |
| 0000 0100 000 | 12 |
| 0000 0011 110 | 13 |
| 0000 0011 100 | 14 |
| 0000 0011 010 | 15 |
| 0000 0011 000 | 16 |

Tableau B.11 – Codes à longueur variable pour structure dmvector[t]

| Code | Valeur |
|------|--------|
| 11 | -1 |
| 0 | 0 |
| 10 | 1 |

B.5 Coefficients DCT

Voir les Tableaux B.12 à B.16.

Tableau B.12 – Codes à longueur variable pour structure dct_dc_size_luminance

| Code à longueur variable | dct_dc_size_luminance |
|--------------------------|-----------------------|
| 100 | 0 |
| 00 | 1 |
| 01 | 2 |
| 101 | 3 |
| 110 | 4 |
| 1110 | 5 |
| 1111 0 | 6 |
| 1111 10 | 7 |
| 1111 110 | 8 |
| 1111 1110 | 9 |
| 1111 1111 0 | 10 |
| 1111 1111 1 | 11 |

Tableau B.13 – Codes à longueur variable pour structure dct_dc_size_chrominance

| Code à longueur variable | dct_dc_size_chrominance |
|--------------------------|-------------------------|
| 00 | 0 |
| 01 | 1 |
| 10 | 2 |
| 110 | 3 |
| 1110 | 4 |
| 1111 0 | 5 |
| 1111 10 | 6 |
| 1111 110 | 7 |
| 1111 1110 | 8 |
| 1111 1111 0 | 9 |
| 1111 1111 10 | 10 |
| 1111 1111 11 | 11 |

Tableau B.14 – Tableau 0 des coefficients DCT

| Code à longueur variable (Note 1) | Exécution | Niveau |
|--------------------------------------|-------------------|--------|
| 10 (Note 2) | Fin de bloc (EOB) | |
| 1 s (Note 3) | 0 | 1 |
| 11 s (Note 4) | 0 | 1 |
| 011 s | 1 | 1 |
| 0100 s | 0 | 2 |
| 0101 s | 2 | 1 |
| 0010 1 s | 0 | 3 |
| 0011 1 s | 3 | 1 |
| 0011 0 s | 4 | 1 |
| 0001 10 s | 1 | 2 |
| 0001 11 s | 5 | 1 |
| 0001 01 s | 6 | 1 |
| 0001 00 s | 7 | 1 |
| 0000 110 s | 0 | 4 |
| 0000 100 s | 2 | 2 |
| 0000 111 s | 8 | 1 |
| 0000 101 s | 9 | 1 |
| 0000 01 | Echappement | |
| 0010 0110 s | 0 | 5 |
| 0010 0001 s | 0 | 6 |
| 0010 0101 s | 1 | 3 |
| 0010 0100 s | 3 | 2 |
| 0010 0111 s | 10 | 1 |
| 0010 0011 s | 11 | 1 |
| 0010 0010 s | 12 | 1 |
| 0010 0000 s | 13 | 1 |
| 0000 0010 10 s | 0 | 7 |
| 0000 0011 00 s | 1 | 4 |
| 0000 0010 11 s | 2 | 3 |
| 0000 0011 11 s | 4 | 2 |
| 0000 0010 01 s | 5 | 2 |
| 0000 0011 10 s | 14 | 1 |
| 0000 0011 01 s | 15 | 1 |
| 0000 0010 00 s | 16 | 1 |
| 0000 0001 1101 s | 0 | 8 |
| 0000 0001 1000 s | 0 | 9 |
| 0000 0001 0011 s | 0 | 10 |
| 0000 0001 0000 s | 0 | 11 |
| 0000 0001 1011 s | 1 | 5 |

Tableau B.14 – Tableau 0 des coefficients DCT (suite)

| Code à longueur variable (Note 1) | Exécution | Niveau |
|--------------------------------------|-----------|--------|
| 0000 0001 0100 s | 2 | 4 |
| 0000 0001 1100 s | 3 | 3 |
| 0000 0001 0010 s | 4 | 3 |
| 0000 0001 1110 s | 6 | 2 |
| 0000 0001 0101 s | 7 | 2 |
| 0000 0001 0001 s | 8 | 2 |
| 0000 0001 1111 s | 17 | 1 |
| 0000 0001 1010 s | 18 | 1 |
| 0000 0001 1001 s | 19 | 1 |
| 0000 0001 0111 s | 20 | 1 |
| 0000 0001 0110 s | 21 | 1 |
| 0000 0000 1101 0 s | 0 | 12 |
| 0000 0000 1100 1 s | 0 | 13 |
| 0000 0000 1100 0 s | 0 | 14 |
| 0000 0000 1011 1 s | 0 | 15 |
| 0000 0000 1011 0 s | 1 | 6 |
| 0000 0000 1010 1 s | 1 | 7 |
| 0000 0000 1010 0 s | 2 | 5 |
| 0000 0000 1001 1 s | 3 | 4 |
| 0000 0000 1001 0 s | 5 | 3 |
| 0000 0000 1000 1 s | 9 | 2 |
| 0000 0000 1000 0 s | 10 | 2 |
| 0000 0000 1111 1 s | 22 | 1 |
| 0000 0000 1111 0 s | 23 | 1 |
| 0000 0000 1110 1 s | 24 | 1 |
| 0000 0000 1110 0 s | 25 | 1 |
| 0000 0000 1101 1 s | 26 | 1 |
| 0000 0000 0111 11 s | 0 | 16 |
| 0000 0000 0111 10 s | 0 | 17 |
| 0000 0000 0111 01 s | 0 | 18 |
| 0000 0000 0111 00 s | 0 | 19 |
| 0000 0000 0110 11 s | 0 | 20 |
| 0000 0000 0110 10 s | 0 | 21 |
| 0000 0000 0110 01 s | 0 | 22 |
| 0000 0000 0110 00 s | 0 | 23 |
| 0000 0000 0101 11 s | 0 | 24 |
| 0000 0000 0101 10 s | 0 | 25 |
| 0000 0000 0101 01 s | 0 | 26 |
| 0000 0000 0101 00 s | 0 | 27 |

Tableau B.14 – Tableau 0 des coefficients DCT (*fin*)

| Code à longueur variable (Note 1) | Exécution | Niveau |
|--|-----------|--------|
| 0000 0000 0100 11 s | 0 | 28 |
| 0000 0000 0100 10 s | 0 | 29 |
| 0000 0000 0100 01 s | 0 | 30 |
| 0000 0000 0100 00 s | 0 | 31 |
| 0000 0000 0011 000 s | 0 | 32 |
| 0000 0000 0010 111 s | 0 | 33 |
| 0000 0000 0010 110 s | 0 | 34 |
| 0000 0000 0010 101 s | 0 | 35 |
| 0000 0000 0010 100 s | 0 | 36 |
| 0000 0000 0010 011 s | 0 | 37 |
| 0000 0000 0010 010 s | 0 | 38 |
| 0000 0000 0010 001 s | 0 | 39 |
| 0000 0000 0010 000 s | 0 | 40 |
| 0000 0000 0011 111 s | 1 | 8 |
| 0000 0000 0011 110 s | 1 | 9 |
| 0000 0000 0011 101 s | 1 | 10 |
| 0000 0000 0011 100 s | 1 | 11 |
| 0000 0000 0011 011 s | 1 | 12 |
| 0000 0000 0011 010 s | 1 | 13 |
| 0000 0000 0011 001 s | 1 | 14 |
| 0000 0000 0001 0011 s | 1 | 15 |
| 0000 0000 0001 0010 s | 1 | 16 |
| 0000 0000 0001 0001 s | 1 | 17 |
| 0000 0000 0001 0000 s | 1 | 18 |
| 0000 0000 0001 0100 s | 6 | 3 |
| 0000 0000 0001 1010 s | 11 | 2 |
| 0000 0000 0001 1001 s | 12 | 2 |
| 0000 0000 0001 1000 s | 13 | 2 |
| 0000 0000 0001 0111 s | 14 | 2 |
| 0000 0000 0001 0110 s | 15 | 2 |
| 0000 0000 0001 0101 s | 16 | 2 |
| 0000 0000 0001 1111 s | 27 | 1 |
| 0000 0000 0001 1110 s | 28 | 1 |
| 0000 0000 0001 1101 s | 29 | 1 |
| 0000 0000 0001 1100 s | 30 | 1 |
| 0000 0000 0001 1011 s | 31 | 1 |
| NOTES | | |
| 1 Le dernier bit 's' indique le signe du niveau: '0' pour signe positif, '1' pour signe négatif. | | |
| 2 «EOB» (fin de bloc) ne doit pas constituer le dernier code du bloc. | | |
| 3 Ce code doit être utilisé pour le premier coefficient (DC) du bloc. | | |
| 4 Ce code doit être utilisé pour tous les autres coefficients. | | |

Tableau B.15 – Tableau 1 des coefficients DCT

| Code à longueur variable (Note 1) | Exécution | Niveau |
|--------------------------------------|-------------------|--------|
| 0110 (Note 2) | Fin de bloc (EOB) | |
| 10s | 0 | 1 |
| 010 s | 1 | 1 |
| 110 s | 0 | 2 |
| 0010 1 s | 2 | 1 |
| 0111 s | 0 | 3 |
| 0011 1 s | 3 | 1 |
| 0001 10 s | 4 | 1 |
| 0011 0 s | 1 | 2 |
| 0001 11 s | 5 | 1 |
| 0000 110 s | 6 | 1 |
| 0000 100 s | 7 | 1 |
| 1110 0 s | 0 | 4 |
| 0000 111 s | 2 | 2 |
| 0000 101 s | 8 | 1 |
| 1111 000 s | 9 | 1 |
| 0000 01 | Echappement | |
| 1110 1 s | 0 | 5 |
| 0001 01 s | 0 | 6 |
| 1111 001 s | 1 | 3 |
| 0010 0110 s | 3 | 2 |
| 1111 010 s | 10 | 1 |
| 0010 0001 s | 11 | 1 |
| 0010 0101 s | 12 | 1 |
| 0010 0100 s | 13 | 1 |
| 0001 00 s | 0 | 7 |
| 0010 0111 s | 1 | 4 |
| 1111 1100 s | 2 | 3 |
| 1111 1101 s | 4 | 2 |
| 0000 0010 0 s | 5 | 2 |
| 0000 0010 1 s | 14 | 1 |
| 0000 0011 1 s | 15 | 1 |
| 0000 0011 01 s | 16 | 1 |
| 1111 011 s | 0 | 8 |
| 1111 100 s | 0 | 9 |
| 0010 0011 s | 0 | 10 |
| 0010 0010 s | 0 | 11 |
| 0010 0000 s | 1 | 5 |
| 0000 0011 00 s | 2 | 4 |

Tableau B.15 – Tableau 1 des coefficients DCT (suite)

| Code à longueur variable (Note 1) | Exécution | Niveau |
|--------------------------------------|-----------|--------|
| 0000 0001 1100 s | 3 | 3 |
| 0000 0001 0010 s | 4 | 3 |
| 0000 0001 1110 s | 6 | 2 |
| 0000 0001 0101 s | 7 | 2 |
| 0000 0001 0001 s | 8 | 2 |
| 0000 0001 1111 s | 17 | 1 |
| 0000 0001 1010 s | 18 | 1 |
| 0000 0001 1001 s | 19 | 1 |
| 0000 0001 0111 s | 20 | 1 |
| 0000 0001 0110 s | 21 | 1 |
| 1111 1010 s | 0 | 12 |
| 1111 1011 s | 0 | 13 |
| 1111 1110 s | 0 | 14 |
| 1111 1111 s | 0 | 15 |
| 0000 0000 1011 0 s | 1 | 6 |
| 0000 0000 1010 1 s | 1 | 7 |
| 0000 0000 1010 0 s | 2 | 5 |
| 0000 0000 1001 1 s | 3 | 4 |
| 0000 0000 1001 0 s | 5 | 3 |
| 0000 0000 1000 1 s | 9 | 2 |
| 0000 0000 1000 0 s | 10 | 2 |
| 0000 0000 1111 1 s | 22 | 1 |
| 0000 0000 1111 0 s | 23 | 1 |
| 0000 0000 1110 1 s | 24 | 1 |
| 0000 0000 1110 0 s | 25 | 1 |
| 0000 0000 1101 1 s | 26 | 1 |
| 0000 0000 0111 11 s | 0 | 16 |
| 0000 0000 0111 10 s | 0 | 17 |
| 0000 0000 0111 01 s | 0 | 18 |
| 0000 0000 0111 00 s | 0 | 19 |
| 0000 0000 0110 11 s | 0 | 20 |
| 0000 0000 0110 10 s | 0 | 21 |
| 0000 0000 0110 01 s | 0 | 22 |
| 0000 0000 0110 00 s | 0 | 23 |
| 0000 0000 0101 11 s | 0 | 24 |
| 0000 0000 0101 10 s | 0 | 25 |
| 0000 0000 0101 01 s | 0 | 26 |
| 0000 0000 0101 00 s | 0 | 27 |
| 0000 0000 0100 11 s | 0 | 28 |

Tableau B.15 – Tableau 1 des coefficients DCT (*fin*)

| Code à longueur variable (Note 1) | Exécution | Niveau |
|--|-----------|--------|
| 0000 0000 0100 10 s | 0 | 29 |
| 0000 0000 0100 01 s | 0 | 30 |
| 0000 0000 0100 00 s | 0 | 31 |
| 0000 0000 0011 000 s | 0 | 32 |
| 0000 0000 0010 111 s | 0 | 33 |
| 0000 0000 0010 110 s | 0 | 34 |
| 0000 0000 0010 101 s | 0 | 35 |
| 0000 0000 0010 100 s | 0 | 36 |
| 0000 0000 0010 011 s | 0 | 37 |
| 0000 0000 0010 010 s | 0 | 38 |
| 0000 0000 0010 001 s | 0 | 39 |
| 0000 0000 0010 000 s | 0 | 40 |
| 0000 0000 0011 111 s | 1 | 8 |
| 0000 0000 0011 110 s | 1 | 9 |
| 0000 0000 0011 101 s | 1 | 10 |
| 0000 0000 0011 100 s | 1 | 11 |
| 0000 0000 0011 011 s | 1 | 12 |
| 0000 0000 0011 010 s | 1 | 13 |
| 0000 0000 0011 001 s | 1 | 14 |
| 0000 0000 0001 0011 s | 1 | 15 |
| 0000 0000 0001 0010 s | 1 | 16 |
| 0000 0000 0001 0001 s | 1 | 17 |
| 0000 0000 0001 0000 s | 1 | 18 |
| 0000 0000 0001 0100 s | 6 | 3 |
| 0000 0000 0001 1010 s | 11 | 2 |
| 0000 0000 0001 1001 s | 12 | 2 |
| 0000 0000 0001 1000 s | 13 | 2 |
| 0000 0000 0001 0111 s | 14 | 2 |
| 0000 0000 0001 0110 s | 15 | 2 |
| 0000 0000 0001 0101 s | 16 | 2 |
| 0000 0000 0001 1111 s | 27 | 1 |
| 0000 0000 0001 1110 s | 28 | 1 |
| 0000 0000 0001 1101 s | 29 | 1 |
| 0000 0000 0001 1100 s | 30 | 1 |
| 0000 0000 0001 1011 s | 31 | 1 |
| NOTES | | |
| 1 Le dernier bit 's' indique le signe du niveau: '0' pour signe positif, '1' pour signe négatif. | | |
| 2 «EOB» (fin de bloc) ne doit pas constituer le dernier code du bloc. | | |

Tableau B.16 – Codage d'exécution et de niveau après un code d'échappement

| Code à longueur fixe | Exécution |
|----------------------|-----------|
| 0000 00 | 0 |
| 0000 01 | 1 |
| 0000 10 | 2 |
| ... | ... |
| ... | ... |
| ... | ... |
| ... | ... |
| 1111 11 | 63 |

| Code à longueur fixe | signed_level |
|----------------------|------------------|
| 1000 0000 0001 | -2047 |
| 1000 0000 0010 | -2046 |
| ... | ... |
| 1111 1111 1111 | -1 |
| 0000 0000 0000 | Valeur interdite |
| 0000 0000 0001 | +1 |
| ... | ... |
| 0111 1111 1111 | +2047 |

Annexe C

Vérificateur de mémoire vidéo

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

Les flux binaires de données vidéo codées doivent correspondre aux contraintes imposées par l'intermédiaire du vérificateur de mémoire vidéo (VBV) défini dans la présente annexe. Aucun flux binaire d'une hiérarchie échelonnée ne doit enfreindre les contraintes en question et dont la définition figure ci-après.

Le vérificateur VBV est un décodeur théorique qui est censé se raccorder à la sortie d'un codeur. Il possède une mémoire tampon d'entrée qui est appelée tampon du vérificateur VBV. Les données codées sont placées dans le tampon comme indiqué en C.3 ci-dessous et en sont extraites comme indiqué en C.5, C.6 et C.7. Il est prescrit qu'un flux binaire conforme à la présente Spécification ne provoque pas de surremplissage du tampon du vérificateur VBV. Si la structure syntaxique `low_delay` a la valeur 0, le flux binaire ne doit pas provoquer de sous-remplissage du tampon du vérificateur VBV. Si elle a la valeur 1, le décodage d'une image au moment normalement prévu peut provoquer un sous-remplissage du tampon du vérificateur VBV. Si tel est le cas, l'image n'est pas décodée et le tampon du vérificateur VBV est réexaminé à des moments ultérieurs, dont la séquence est spécifiée en C.7 et C.8, jusqu'à ce que l'image soit entièrement présente dans le tampon du vérificateur VBV.

Toutes les opérations arithmétiques de la présente annexe sont effectuées en nombres réels, de façon qu'aucune erreur d'arrondi ne puisse se propager. Par exemple, le nombre de bits contenus dans la mémoire du vérificateur VBV n'est pas nécessairement un nombre entier.

C.1 Le vérificateur VBV et le codeur vidéo ont la même fréquence d'horloge ainsi que la même fréquence d'image; ils opèrent de manière synchrone.

C.2 La mémoire d'entrée du vérificateur VBV possède une capacité B, la valeur de B étant indiquée par la structure `vbv_buffer_size` codée dans l'en-tête de séquence et dans l'éventuelle extension de séquence

C.3 Le présent paragraphe définit les modalités d'introduction des données dans la mémoire tampon du vérificateur VBV. Les paragraphes C.3.1 et C.3.2 définissent deux cas qui s'excluent mutuellement. Dans l'un et l'autre cas, la mémoire d'entrée du vérificateur VBV est initialement vide. Soit R_{\max} la valeur du débit binaire spécifié dans le champ `bit_rate`.

C.3.1 Si la valeur de codage de `vbv_delay` est différente de la valeur hexadécimale FFFF, les données d'image de la nième image codée sont introduites dans la mémoire tampon selon un débit binaire $R(n)$ tel que:

$$R(n) = d_n^* / (\tau(n) - \tau(n + 1) + t(n + 1) - t(n))$$

avec:

- $R(n)$ débit (en bit/s) d'introduction des données d'image relatives à la nième image codée;
- d_n^* nombre de bits situés après le dernier bit du nième code de déclenchement d'image et avant et y compris le dernier du $(n + 1)$ ème code de déclenchement d'image;
- $\tau(n)$ délai de décodage (mesuré en secondes) codé dans le champ `vbv_delay` relatif à la nième image codée;
- $t(n)$ instant (mesuré en secondes) auquel la nième image codée est supprimée de la mémoire du vérificateur VBV; les paragraphes C.9, C.10, C.11 et C.12 donnent la définition de $t(n)$.

En ce qui concerne les bits qui précèdent le premier code de déclenchement d'image et qui suivent le dernier code de déclenchement d'image $R(n) = R_{\max}$.

Après remplissage de la mémoire tampon du vérificateur VBV avec toutes les données qui précèdent le premier code de déclenchement d'image de la séquence et avec le code de déclenchement d'image proprement dit, la mémoire tampon du vérificateur VBV est remplie par le flux binaire pendant le temps spécifié dans le champ `vbv_delay` de l'en-tête d'image. C'est à cet instant que le décodage commence. L'introduction de données se poursuit selon les débits spécifiés dans le présent paragraphe.

Tous les débits binaires se caractérisent par $R(n) \leq R_{\max}$ quelles que soient les données d'image considérées.

NOTE – Dans le cas des flux de données vidéo à débit constant, les valeurs successives du débit $R(n)$ restent constantes pendant toute la séquence, à la précision autorisée par la quantification de `vbv_delay`.

C.3.2 Si la valeur de codage de `vbv_delay` est égale à la valeur hexadécimale FFFF, les données sont introduites dans la mémoire tampon du vérificateur VBV dans les conditions spécifiées par le présent paragraphe.

Si la mémoire tampon du vérificateur VBV n'est pas pleine, les données sont introduites selon le débit R_{max} .

Si le tampon du vérificateur VBV se remplit après un certain temps de remplissage au débit R_{max} , plus aucune donnée n'entre dans le tampon avant que d'autres données en soient supprimées.

Après remplissage de la mémoire tampon du vérificateur VBV avec toutes les données qui précèdent le premier code de déclenchement d'image de la séquence et avec le code de déclenchement d'image proprement dit, la mémoire tampon du vérificateur VBV est remplie par le flux binaire jusqu'à ce qu'elle soit pleine; c'est à cet instant que le décodage commence. L'introduction de données se poursuit alors selon les débits spécifiés dans le présent paragraphe.

C.4 A partir de l'instant défini en C.3, la mémoire du vérificateur VBV est examinée périodiquement, à des moments définis dans C.9 à C.12. Les paragraphes C.5 à C.8 définissent les mesures à prendre à chaque examen de la mémoire tampon du vérificateur VBV.

C.5 Ce paragraphe définit une prescription applicable à tous les flux binaires de données vidéo codées.

Si la mémoire du vérificateur VBV est examinée *avant* la suppression d'une quelconque donnée d'image, le nombre de bits contenus dans la mémoire doit être compris entre zéro et B bits, où B est la capacité de la mémoire tampon du vérificateur VBV, indiquée par la structure `vbv_buffer_size`.

Dans le cadre de la présente annexe, on entend par données d'image tous les bits de l'image codée, ainsi que le ou les en-têtes et les données d'utilisateur qui, le cas échéant, la précèdent immédiatement (y compris les bits de bourrage éventuellement intercalés) et tous les bits de bourrage qui la suivent, jusqu'au code de déclenchement suivant (mais celui-ci non compris), sauf lorsque le code de déclenchement suivant est un code de fin de séquence, auquel cas il est inclus dans les données d'image.

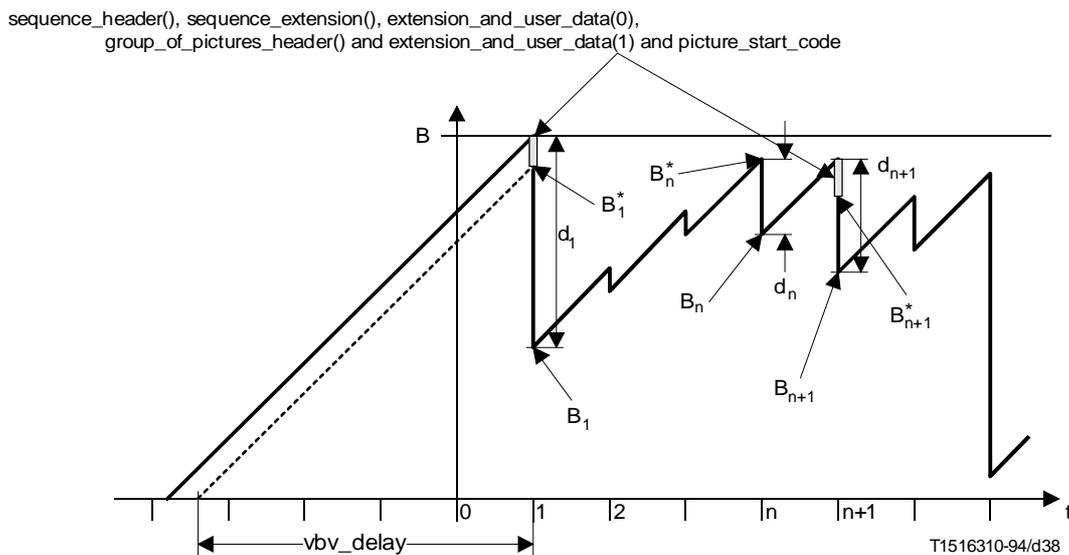


Figure C.1 – Occupation de la mémoire du vérificateur VBV – Exploitation à débit binaire constant

C.6 Ce paragraphe définit une prescription relative aux flux binaires de données vidéo codées lorsque le paramètre `low_delay` est égal à zéro.

Chaque fois que la mémoire du vérificateur VBV est examinée et avant toute suppression de bits, toutes les données de l'image qui (à ce moment) est demeurée le plus longtemps dans la mémoire doivent être présentes dans celle-ci. Les données d'image doivent alors être supprimées instantanément.

Le sous-remplissage de la mémoire du vérificateur VBV ne doit pas se produire lorsque le paramètre `low_delay` est égal à 0. Cette exigence impose que toutes les données d'image concernant la nième image soient présentes dans la mémoire tampon du vérificateur VBV à l'instant de décodage t_n .

C.7 Ce paragraphe ne s'applique que lorsque le fanion `low_delay` a la valeur un.

Lorsque le fanion `low_delay` est mis à la valeur un, certaines situations peuvent imposer un réexamen à plusieurs reprises de la mémoire tampon du vérificateur VBV avant d'en effacer une image codée. L'examen de la référence temporelle de l'image suivante (celle qui suit l'image actuellement à décoder) permet de déterminer s'il faut réexaminer la mémoire tampon du vérificateur VBV et combien de fois il faut procéder à ce réexamen (voir 6.3.10). S'il faut réexaminer la mémoire tampon du vérificateur VBV, l'image à décoder est qualifiée d'image non décodée.

Si l'image à décoder est une image non décodée, la mémoire tampon est réexaminée à intervalles de 2 périodes de trame, jusqu'à ce que l'image en question soit enlevée de la mémoire; aucune donnée d'image n'est retirée de la mémoire tampon, tant que le dernier examen de la mémoire n'a pas été réalisé.

A cet instant, le nombre de bits présents dans la mémoire tampon du vérificateur VBV immédiatement avant que l'image non décodée n'en soit sortie doit être inférieur à la capacité B, toutes les données d'image relatives à l'image qui est restée le plus longtemps dans la mémoire tampon (ladite image non décodée) doivent se trouver dans la mémoire tampon et doivent en être sorties instantanément. Le fonctionnement normal du vérificateur VBV reprend alors et les dispositions énoncées en C.5 sont applicables.

La dernière image codée d'une séquence ne doit pas être une image non décodée.

C.8 Ce paragraphe a valeur d'information seulement.

Il peut se produire que la mémoire du vérificateur VBV (voir C.7) soit sous-remplie lorsque des applications à faible délai transmettent des images de dimensions parfois importantes, par exemple en cas de découpage de scène.

Le décodage de tels flux binaires fera que le processus d'affichage associé à un décodeur répétera une monotrame ou une bi-trame déjà décodée, jusqu'à ce que le vérificateur VBV puisse revenir à son fonctionnement normal. Ce procédé est parfois désigné sous l'expression d'«images sautées». On notera que, dans les conditions normales, cette situation ne devrait pas se produire, à quelques rares exceptions près et seulement de manière occasionnelle. Elle ne doit pas se produire lorsque le fanion `low_delay` = 0.

C.9 Ce paragraphe définit les intervalles de temps qui séparent des examens successifs de la mémoire du vérificateur VBV, dans le cas où `progressive_sequence` = 1 et `low_delay` = 0. Dans ce cas, le délai de réordonnement des bi-frames existe toujours et des images à codage B peuvent apparaître.

L'intervalle de temps $t_{n+1} - t_n$ entre deux examens successifs de la mémoire tampon du vérificateur VBV est un multiple de T, où T est l'inverse de la fréquence d'image.

Si la nième image est à codage B avec une structure `repeat_first_field` égale à 0, on a un intervalle $t_{n+1} - t_n$ égal à T.

Si la nième image est à codage B avec une structure `repeat_first_field` égale à 1 et une structure `top_field_first` égale à 0, on a un intervalle $t_{n+1} - t_n$ égal à $2 * T$.

Si la nième image est à codage B avec une structure `repeat_first_field` égale à 1 et une structure `top_field_first` égale à 1, on a un intervalle $t_{n+1} - t_n$ égal à $3 * T$.

Si la nième image est à codage P ou I et si la précédente image P ou I comporte une structure `repeat_first_field` égale à 0, on a un intervalle $t_{n+1} - t_n$ égal à T.

Si la nième image est à codage P ou I et si la précédente image P ou I comporte une structure `repeat_first_field` égale à 1 et la structure `top_field_first` égale à 0, on a un intervalle $t_{n+1} - t_n$ égal à $2 * T$.

Si la nième image est à codage P ou I et si la précédente image P ou I comporte une structure `repeat_first_field` égale à 1 et une structure `top_field_first` égale à 1, on a un intervalle $t_{n+1} - t_n$ égal à $3 * T$.

Si $t_{n+1} - t_n$ ne peut pas être déterminé selon l'un des alinéas précédents parce qu'il n'y a pas d'image P ou I précédente (ce qui peut se produire en début de séquence), l'intervalle de temps est choisi arbitrairement avec les restrictions suivantes:

L'intervalle de temps qui s'écoule entre la suppression d'une bi-trame (ou de sa première trame) et la suppression de la bi-trame suivante peut être arbitrairement défini comme étant égal à T, $2 * T$ ou $3 * T$. Dans ce cas, la vitesse d'accumulation des données pour la première bi-trame est ambiguë. Le remplissage du tampon du vérificateur VBV peut donc avoir plusieurs valeurs avant que ces données en aient été supprimées. Au moins un des choix valides pour le temps de décodage doit correspondre à un ensemble de remplissages du tampon du vérificateur VBV répondant aux prescriptions de la présente annexe concernant le surremplissage et le sous-remplissage. Si le flux binaire est multiplexé pour s'intégrer à un flux binaire de données systèmes conformément à la Rec. UIT-T H.222.0 | ISO/CEI 13818-1, les informations contenues dans le flux binaire de données systèmes peuvent être utilisées pour déterminer sans ambiguïté le remplissage du tampon du vérificateur VBV après la suppression de la première image.

C.10 Ce paragraphe définit les intervalles de temps qui séparent des examens successifs de la mémoire du vérificateur VBV, dans le cas où *progressive_sequence* = 1 et *low_delay* = 1. Dans ce cas, la séquence ne contient pas d'images B et il n'y a pas de délai de réordonnancement des bi-frames.

L'intervalle de temps $t_{n+1} - t_n$ entre deux examens successifs de la mémoire tampon du vérificateur VBV est un multiple de T, où T est l'inverse de la fréquence d'image.

Si la nième image est à codage P ou I avec une structure *repeat_first_field* égale à 0, on a un intervalle $t_{n+1} - t_n$ égal à T.

Si la nième image est à codage P ou I avec une structure *repeat_first_field* égale à 1 et une structure *top_field_first* égale à 0, on a un intervalle $t_{n+1} - t_n$ égal à 2*T.

Si la nième image est à codage P ou I avec une structure *repeat_first_field* égale à 1 et une structure *top_field_first* égale à 1, on a un intervalle $t_{n+1} - t_n$ égal à 3*T.

C.11 Ce paragraphe définit les intervalles de temps qui séparent des examens successifs de la mémoire du vérificateur VBV, dans le cas où *progressive_sequence* = 1 et *low_delay* = 0. Dans ce cas, le délai de réordonnancement des bi-frames existe toujours et des images B peuvent être acheminées.

L'intervalle de temps $t_{n+1} - t_n$ entre deux examens successifs de la mémoire d'entrée du vérificateur VBV est un multiple de T, où T est l'inverse de deux fois la fréquence d'image.

Si la nième image est de *structure bi-frame* à codage B avec un élément *repeat_first_field* égal à 0, on a un intervalle $t_{n+1} - t_n$ égal à 2*T.

Si la nième image est de *structure bi-frame* à codage B avec un élément *repeat_first_field* égal à 1, on a un intervalle $t_{n+1} - t_n$ égal à 3*T.

Si la nième image est de *structure monotrame* à codage B (image B monotrame), on a un intervalle $t_{n+1} - t_n$ égal à T.

Si la nième image est de *structure bi-frame* à codage P ou à codage I et que la précédente bi-trame à codage P ou à codage I ait un élément *repeat_first_field* égal à 0, on a un intervalle $t_{n+1} - t_n$ égal à 2*T.

Si la nième image est de *structure bi-frame* à codage P ou à codage I et que la précédente bi-trame à codage P ou à codage I ait un élément *repeat_first_field* égal à 1, on a un intervalle $t_{n+1} - t_n$ égal à 3*T.

Si la nième image est la *première* trame d'une *image bi-frame* à codage P ou I de structure monotrame, on a un intervalle $t_{n+1} - t_n$ égal à T.

Si la nième image est la *seconde* trame d'une *image bi-frame* à codage P ou I de structure monotrame et que la précédente bi-trame à codage P ou à codage I utilise une structure monotrame ou possède un élément *repeat_first_field* égal à 0, on a un intervalle $t_{n+1} - t_n$ égal à (2*T - T).

Si la nième image est la *seconde* trame d'une *image bi-frame* à codage P ou I de structure monotrame et que la précédente bi-trame à codage P ou à codage I utilise une structure bi-trame et possède un élément *repeat_first_field* égal à 1, on a un intervalle $t_{n+1} - t_n$ égal à (3*T - T).

Si l'intervalle $t_{n+1} - t_n$ ne peut pas être déterminé par l'un des alinéas précédents parce qu'il n'y a pas de précédente bi-trame à codage P ou I (ce qui peut se produire en début de séquence), l'intervalle de temps est choisi arbitrairement avec les restrictions suivantes:

L'intervalle de temps qui s'écoule entre la suppression d'une bi-trame (ou de sa première trame) et la suppression de la bi-trame suivante (ou de sa première trame) peut être arbitrairement défini comme étant égal à 2*T ou 3*T. Le remplissage du tampon du vérificateur VBV peut donc avoir plusieurs valeurs avant que ces données en aient été supprimées. Au moins un des choix valides pour le temps de décodage doit correspondre à un ensemble de remplissages du tampon du vérificateur VBV répondant aux prescriptions de la présente annexe concernant le surremplissage et le sous-remplissage. Si le flux binaire est multiplexé pour s'intégrer à un flux binaire de données systèmes conformément à la Rec. UIT-T H.222.0 | ISO/CEI 13818-1, les informations contenues dans le flux binaire de données systèmes peuvent être utilisées pour déterminer sans ambiguïté le remplissage du tampon du vérificateur VBV.

La Figure C.2 montre le vérificateur VBV dans un cas simple, ne mettant en jeu que des images bi-frames. Les bi-frames P_0 , B_2 et B_4 ont une durée d'affichage de 3 trames.

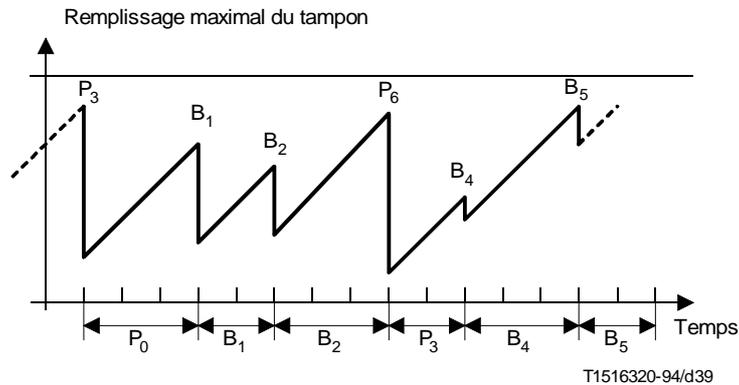


Figure C.2

C.12 Ce paragraphe définit les intervalles de temps qui séparent des examens successifs de la mémoire du vérificateur VBV, dans le cas où *progressive_sequence* = 0 et *low_delay* = 1. Dans ce cas, la séquence ne contient pas d'images à codage B et il n'y a pas de délai de réordonnancement des bi-frames.

L'intervalle de temps $t_{n+1} - t_n$ entre deux examens successifs de la mémoire d'entrée du vérificateur VBV est un multiple de T , où T est l'inverse de deux fois la fréquence d'image.

Si la n ème image est de *structure bi-trame* à codage P ou à codage I avec un élément *repeat_first_field* égal à 0, on a un intervalle $t_{n+1} - t_n$ égal à $2 * T$.

Si la n ème image est de *structure bi-trame* à codage P ou à codage I avec un élément *repeat_first_field* égal à 1, on a un intervalle $t_{n+1} - t_n$ égal à $3 * T$.

Si la n ème image est de *structure monotrème* à codage P ou à codage I, on a un intervalle $t_{n+1} - t_n$ égal à T .

La Figure C.3 montre le vérificateur VBV dans un cas simple, ne mettant en jeu que des images bi-frames. Les bi-frames I_0 , P_2 et P_4 ont un élément *repeat_first_field* égal à 1.

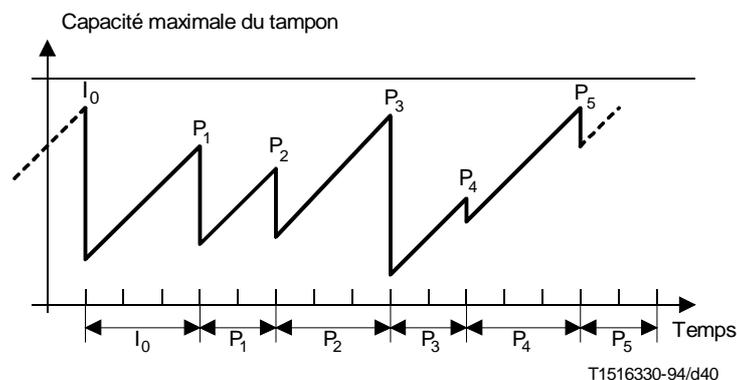


Figure C.3

Annexe D

Caractéristiques supportées par l'algorithme

(Cette annexe ne fait pas partie intégrante de la présente Recommandation | Norme internationale)

D.1 Vue d'ensemble

La présente Spécification inclut la liste non exhaustive de caractéristiques ci-après:

- 1) Il est possible de représenter différents formats d'échantillonnage des éléments de chrominance (c'est-à-dire 4:2:0, 4:2:2 et 4:4:4).
- 2) Il est possible de coder les données vidéo en balayage entrelacé ou progressif.
- 3) Le décodeur peut utiliser une décimation à 3:2 pour convertir un film à ~24 images par seconde en format vidéo à ~30 images par seconde.
- 4) Les données vidéo affichées peuvent être sélectionnées à l'aide d'une fenêtre mobile pour effectuer un recadrage plein écran en plus grand format d'affichage.
- 5) Il est possible d'utiliser une large gamme de qualités d'image.
- 6) Les canaux à débit binaire constant et variable sont supportés.
- 7) Un mode à faible délai est disponible pour les applications interactives.
- 8) L'accès aléatoire est disponible (pour les supports numériques d'enregistrement, l'acquisition et l'interconnexion des canaux).
- 9) Les flux binaires de paramètres contraints définis dans ISO/CEI 11172-2 sont décodables.
- 10) Il est possible de créer des flux binaires pour décodeurs à complexité matérielle élevée et faible.
- 11) L'édition des données vidéo codées est supportée.
- 12) Il est possible d'assurer la lecture rapide en avant/en arrière des flux binaires enregistrés.
- 13) Le flux binaire codé présente une certaine élasticité aux erreurs.

D.2 Formats vidéo

D.2.1 Formats d'échantillonnage et couleur

La présente Spécification du codage vidéo supporte le balayage entrelacé et le balayage progressif. L'indication correspondante est fournie à l'aide d'un fanion `progressive_sequence` dans le code d'extension relative aux séquences.

Le format autorisé pour les matrices d'affichage est compris entre 1 et $(2^{14} - 1)$ échantillons de luminance horizontalement et verticalement. Les données vidéo sont représentées dans un espace chromatique de luminance/chrominance permettant de choisir les couleurs primaires. Il est possible d'échantillonner la chrominance en format 4:2:0 (échantillonnage d'un élément sur deux horizontalement et verticalement) ou en format 4:2:2 (échantillonnage d'un élément sur deux horizontalement seulement). En outre, une certaine souplesse est prévue pour supporter les formats d'échantillon et d'image propres à différentes applications. Le code d'extension relative aux séquences contient un paramètre `chroma_format`.

L'information de format d'échantillon est fournie à l'aide des structures suivantes: information de format (structure `aspect_ratio_information`) et (facultativement) format d'affichage horizontal (structure `display_horizontal_size`) et format d'affichage vertical (structure `display_vertical_size`) dans l'extension d'affichage de séquences() [structure `sequence_display_extension()`]. Le Tableau D.1 donne des exemples de valeurs appropriées pour les signaux échantillonnés conformément à la Recommandation UIT-R BT.601.

La présente Spécification définit les outils requis pour supporter la chrominance 4:4:4, en vue d'une utilisation future éventuelle. Néanmoins, aucun profil ne supporte actuellement ce format.

Tableau D.1 – Exemples de formats d'affichage

| Format des signaux | <code>display_horizontal_size</code> | <code>display_vertical_size</code> |
|--------------------|--------------------------------------|------------------------------------|
| 525-lignes | 711 | 483 |
| 625-lignes | 702 | 575 |

D.2.2 Rythme des images cinématographiques

Un décodeur peut assurer la décimation à 3:2 lorsqu'une séquence d'images à balayage progressif est codée. Chaque image cinématographique codée peut spécifier de manière autonome si elle est affichée pendant deux ou trois périodes de trame vidéo, de sorte que l'on peut transmettre en vidéo à balayage progressif les données d'origine «irrégulière» ayant subi une décimation à 3:2. On transmet deux fanions, `top_field_first` et `repeat_first_field`, avec les extensions relatives au codage des images afin de donner les indications adéquates sur le temps d'affichage nécessaire.

D.2.3 Contrôle du format d'affichage

Le processus d'affichage convertit une séquence d'images bi-frames numériques (pour le balayage progressif) ou de monotrames (pour le balayage entrelacé) en données vidéo de sortie. Aux fins de la présente Spécification, cette partie n'est pas normative. La syntaxe vidéo de la présente Spécification transmet certains paramètres d'affichage destinés à être utilisés dans la reconstruction des images vidéo. Des informations facultatives (dans l'extension d'affichage de séquences) spécifient les coordonnées chromatiques, les couleurs primaires, les caractéristiques de transfert optoélectronique (par exemple, valeur du gamma) et la matrice de conversion des composantes RGB en composantes luminance/chrominance (Y/C).

De plus, il est possible de définir une fenêtre d'affichage à l'intérieur de la matrice d'affichage codée, comme dans le cas du recadrage plein écran. On peut aussi définir la matrice codée comme étant une fenêtre dans un dispositif d'affichage à grand écran. S'agissant du recadrage plein écran, la position de la fenêtre représentant la région affichée d'une image de format plus important peut être spécifiée trame par trame (voir au 6.3.12 la Spécification fournie dans l'extension relative à l'affichage des images). Une utilisation classique de la fenêtre de recadrage plein écran consiste à décrire le rectangle «important» de format 4:3 dans une séquence vidéo de format 16:9. De même, pour les petites images codées sur un grand dispositif d'affichage, il est possible de spécifier le format de l'affichage et la position de la fenêtre dans cette zone d'affichage.

D.2.4 Codage transparent de signaux vidéo composites

Le décodage à partir du système PAL/NTSC avant la transmission et le recodage en PAL/NTSC après la transmission des signaux composites d'origine, dans des applications de qualité non médiocre telles que la contribution et la distribution, nécessitent une reconstruction précise des signaux d'amplitude et de référence de phase de la porteuse (et un commutateur de signal V pour le système PAL).

Il est possible d'indiquer le format des signaux d'entrée dans l'en-tête de séquence en utilisant les bits du format vidéo (structure *video_format*). Les possibilités de format des signaux d'origine sont les suivantes: PAL, NTSC, SECAM et MAC. La reconstruction du signal de la porteuse peut être accomplie à l'aide des paramètres de porteuse suivants: `V_axis`, `field_sequence`, `sub_carrier`, `burst_amplitude` et `sub_carrier_phase`. L'activation de ces paramètres est déclenchée par le fanion `composite_display_flag` dans l'en-tête de l'image.

D.3 Qualité d'image

La qualité d'image dépend du débit binaire utilisé. On obtient une qualité très élevée en utilisant des limites de débit binaire suffisamment larges par rapport à un niveau donné de profil particulier. On peut obtenir une qualité de chrominance élevée en utilisant la chrominance 4:2:2.

Il est possible de télécharger les matrices de quantification et de les utiliser avec un petit code d'échelle de quantification (structure *quantiser_scale_code*) pour obtenir un codage pratiquement dépourvu de pertes.

De plus, le codage échelonnable avec débit binaire flexible permet d'obtenir une hiérarchisation du service ou de la qualité et une dégradation progressive. Par exemple, le décodage d'un sous-ensemble du flux binaire acheminant une image de résolution moins élevée permet d'effectuer le décodage des signaux dans un récepteur à faible coût et d'obtenir une qualité correspondante; le décodage du flux binaire complet permet d'obtenir une qualité globale élevée.

Par ailleurs, on peut assurer le fonctionnement aux débits réduits en utilisant des fréquences d'image peu élevées (soit en effectuant un prétraitement avant le codage soit en sautant les images bi-frames indiquées par le paramètre `temporal_reference` fourni dans l'en-tête de l'image) et une résolution spatiale peu importante.

D.4 Contrôle du débit

Le nombre de bits transmis par unité de temps, susceptible d'être choisi dans une large gamme, peut être contrôlé selon deux méthodes, que supporte la présente Spécification. Une description du débit (*bit_rate*) est transmise dans le code d'en-tête de séquence.

Pour le codage à débit constant (CBR) (*constant bitrate*), le nombre de bits transmis par unité de temps est constant sur le canal. Etant donné que le débit de sortie du codeur varie généralement en fonction du contenu de l'image, la régulation du débit constant repose sur des techniques telles que la mise en mémoire, tampon, etc. En débit constant, la qualité d'image peut varier en fonction du contenu des images.

L'autre mode est le codage à débit variable (VBR) (*variable bitrate*), qui permet de contraindre plus ou moins, dans le canal, le nombre de bits transmis par unité de temps. Le débit variable a pour but d'assurer un codage de qualité constante. Un modèle d'application de ce mode est le codage de qualité quasi constante sur les canaux du RNIS à large bande faisant l'objet d'une gestion (UPC) des paramètres côté utilisateur.

D.5 Mode à faible délai

Un mode à faible délai de codage et de décodage est prévu pour des communications vidéo en temps réel telles que la vidéophonie, la vidéoconférence ou la télésurveillance. Il est possible d'obtenir un délai total de codage et de décodage inférieur à 150 ms grâce au mode à faible délai défini dans la présente Spécification. L'activation du fanion `low_delay` dans le code d'en-tête de séquence spécifie un flux binaire à faible délai.

Le retard total de codage et de décodage peut rester faible si l'on crée un flux binaire ne contenant pas d'image B, ce qui évite les délais imputables au réordonnement des images bi-frames. L'utilisation d'une prédiction anticipée à double polarité pour les images P bi-frames permet de conserver une qualité d'image élevée.

La faible occupation de la mémoire, aussi bien pour le codeur que pour le décodeur, est une nécessité dans le mode à faible délai. Au niveau du codeur, il convient d'éviter les images codées de grand format. Cela est possible si l'on fait appel à une actualisation du codage intra sur la base d'une ou de plusieurs tranches par image bi-trame (tranches à codage intra) et non pas sur la base d'images bi-frames à codage intra.

En cas de dépassement du nombre souhaité de bits par bi-trame pour le fonctionnement en mode à faible délai, le codeur peut sauter une ou plusieurs images bi-frames. Cette action est indiquée par une discontinuité de la valeur de référence temporelle pour l'image suivante (voir au 6.3.9 la définition sémantique de cette structure) et peut déclencher l'application de C.7 concernant les vérificateurs VBV; en d'autres termes, le tampon du décodeur passerait en sous-remplissage si certaines bi-frames n'étaient pas répétées par le décodeur.

D.6 Accès aléatoire/interconnexion des canaux

La syntaxe de la présente Spécification supporte l'accès aléatoire et l'interconnexion des canaux. Il est possible d'assurer des caractéristiques fonctionnelles suffisantes pour l'accès aléatoire/interconnexion des canaux en codant des points d'accès aléatoire appropriés dans le flux binaire sans pertes importantes de qualité des images.

L'accès aléatoire est une caractéristique essentielle pour un support d'enregistrement vidéo. Cette caractéristique suppose que toute image soit accessible et décodable en un temps limité et implique l'existence de points d'accès au flux binaire – c'est-à-dire de segments d'information identifiables et décodables sans référence à d'autres segments de données. Dans la présente Spécification, les points d'accès sont fournis par l'en-tête de séquence [structure `sequence_header()`], suivi de l'information de codage intra (données vidéo susceptibles d'être décodées sans accès aux images précédemment décodées). Il est possible d'obtenir un espacement de deux points d'accès aléatoire par seconde sans pertes importantes de qualité des images.

L'interconnexion des canaux est un scénario analogue dans des applications telles que la radiodiffusion. Dès qu'un nouveau canal a été choisi et que le flux binaire de ce canal est disponible au niveau du décodeur, il faut localiser l'entrée de données suivantes, c'est-à-dire le point d'accès aléatoire suivant, pour commencer le décodage du nouveau programme dans les conditions décrites à l'alinéa précédent.

D.7 Echelonnabilité

La syntaxe de la présente Spécification supporte l'échelonnabilité des flux binaires. Un certain nombre d'outils d'échelonnabilité des flux binaires sont prévus pour tenir compte des différences de caractéristiques fonctionnelles entre les applications envisagées dans la présente Spécification, à savoir:

- **Echelonnabilité SNR**, principalement destinée aux applications qui nécessitent une dégradation progressive.
- **Diffusion jumelée de données de chrominance**, destinée aux applications très exigeantes en qualité de chrominance.
- **Subdivision des données**, servant essentiellement à assurer une certaine élasticité aux pertes de cellules dans les réseaux à mode de transfert asynchrone (ATM).

- **Echelonnabilité temporelle**, qui est un type d'échelonnabilité approprié à l'interfonctionnement des services utilisant des formats vidéo à balayage progressif et à résolution temporelle élevée. Cette méthode est également appropriée pour la dégradation progressive de qualité élevée en présence d'erreurs sur le canal.
- **Echelonnabilité spatiale**, qui permet d'utiliser une technique de codage à résolution multiple pour les cas d'interfonctionnement de services vidéo. En outre, ce type d'échelonnabilité fournit des modes de codage qui permettent d'assurer la compatibilité avec les normes de codage existantes, en l'occurrence ISO/CEI 11172-2, au niveau de la couche inférieure.

D.7.1 Utilisation de l'échelonnabilité SNR avec une résolution spatiale unique

L'échelonnabilité SNR sert essentiellement à fournir un mécanisme de transmission pour un service à deux couches, celles-ci assurant la même résolution d'image mais à un niveau de qualité différent. Par exemple, l'exploitation du service à deux niveaux de qualité deviendra vraisemblablement utile dans l'avenir pour certaines applications de radiodiffusion télévisuelle, en particulier lorsqu'il sera nécessaire d'avoir une excellente qualité d'image sur des récepteurs à grand format d'affichage. La séquence est codée en deux flux binaires appelés «flux binaire de couche inférieure» et «flux binaire de couche d'amélioration». Le premier peut être décodé indépendamment du second. La couche inférieure, à un débit compris entre 3 et 4 Mbit/s, fournirait alors une qualité d'image équivalant à la qualité actuelle des systèmes NTSC/PAL/SECAM. Ensuite, l'utilisation conjointe des deux flux binaires (couche inférieure et couche d'amélioration) permet à un décodeur amélioré d'offrir une qualité d'image subjectivement proche de la qualité de studio, avec un débit binaire total allant de 7 à 12 Mbit/s.

D.7.1.1 Caractéristiques supplémentaires

D.7.1.1.1 Élasticité aux erreurs

Comme indiqué en D.13, l'échelonnabilité SNR peut être un facteur d'élasticité aux erreurs. Si les flux binaires des deux couches sont reçus avec un taux d'erreur différent, la couche inférieure, mieux protégée, est un bon substitut de repli, au cas où la couche d'amélioration serait détériorée.

D.7.1.1.2 Diffusion jumelée de données de chrominance

La syntaxe échelonnable SNR peut être utilisée dans un système de diffusion jumelée de données de chrominance. L'objectif d'un tel système serait la mise en place d'un mécanisme de distribution simultanée des services avec la même résolution de luminance, mais avec un format d'échantillonnage de chrominance différent (par exemple 4:2:0 dans la couche inférieure et 4:2:2 si l'on ajoute la couche d'amélioration et les composantes de chrominance en diffusion jumelée), pour les applications nécessitant une telle caractéristique. La couche d'amélioration à échelonnabilité SNR contient certaines données de raffinement pour la luminance. Les données de chrominance au format 4:2:2 sont transmises en diffusion jumelée. Seuls les coefficients DC de chrominance font l'objet d'une prédiction depuis la couche inférieure. La combinaison de la luminance des deux couches et des données de chrominance au format 4:2:2 détermine le niveau élevé de qualité.

D.7.1.2 Processus de codage à échelonnabilité SNR

D.7.1.2.1 Description

Dans la couche inférieure, le mode de codage est analogue au scénario sans échelonnabilité en termes de décisions, de quantification adaptative et de régulation de la mémoire. Les macroblocs de prédiction d'erreur ou de codage intra font l'objet d'une transformée en cosinus discrète (DCT). Ensuite, les coefficients sont quantifiés à l'aide d'un premier quantificateur assez grossier. Les coefficients quantifiés subissent alors un codage à longueur variable avant d'être transmis avec l'information collatérale requise, à savoir le type de macrobloc (structure *macroblock_type*), les vecteurs de mouvement et la structure de bloc codée() [structure *coded_block_pattern()*].

Parallèlement, les coefficients DCT quantifiés provenant de la couche inférieure sont déquantifiés. L'erreur résiduelle entre les coefficients et les coefficients déquantifiés est ensuite requantifiée à l'aide d'un deuxième quantificateur plus fin. Les coefficients de raffinement résultants subissent un codage à longueur variable et constituent la couche d'amélioration supplémentaire, assortie d'une quantité marginale d'informations collatérales, à savoir: code d'échelle de quantification (structure *quantiser_scale_code*), structure *coded_block_pattern()* ... On utilise la table de codage à longueur variable non intra pour tous les coefficients dans la couche d'amélioration, compte tenu de la nature différentielle du signal transmis.

D.7.1.2.2 Quelques remarques importantes

Puisque la prédiction est la même pour les deux couches, il est recommandé d'employer les images affinées dans la boucle d'estimation de mouvement (par exemple, images obtenues par combinaison des couches inférieure et d'amélioration). Ainsi, il existe une dérive entre la prédiction utilisée du côté du codeur et le résultat de prédiction que le décodeur de niveau inférieur peut obtenir. Cette dérive augmente d'image P en image P et est remise à zéro à chaque image I. Néanmoins, il est apparu que la dérive avait un effet visuel peu important sur les images à codage intra apparaissant toutes les 15 images environ.

Etant donné que la couche d'amélioration contient seulement les coefficients de raffinement, la servitude binaire requise est assez limitée: la plupart de l'information relative aux macroblocs (types de macroblocs, vecteurs de mouvement, etc.) se trouve dans la couche inférieure. En conséquence, la syntaxe de flux est considérablement simplifiée:

- le tableau des types de macroblocs indique si la structure `quantiser_scale_code` de la couche d'amélioration a été modifiée et si le macrobloc est NON CODÉ (pour le premier et le dernier macrobloc des tranches), ce qui correspond à trois mots de code à longueur variable;
- la structure `quantiser_scale_code` de la couche d'amélioration est transmise si la valeur a été modifiée;
- la structure `coded_block_pattern()` est transmise pour l'ensemble des macroblocs codés.

Tous les macroblocs NON CODÉS qui ne se trouvent pas au début ou à la fin d'une tranche sont sautés étant donné que l'information de servitude binaire peut être obtenue par déduction depuis la couche inférieure.

Il est recommandé d'employer des matrices de pondération différentes pour la couche inférieure et pour la couche d'amélioration. On obtient de meilleurs résultats lorsque la première quantification est plus fine que la deuxième quantification. Néanmoins, il est recommandé de ne pas quantifier trop grossièrement le coefficient DCT qui correspond à l'entrelacement des vecteurs de mouvement, afin d'éviter les effets de broutage.

D.7.2 Flux binaire échelonnable à résolution multiple faisant appel à l'échelonnabilité SNR

L'objectif de l'échelonnabilité des résolutions est de décoder les données vidéo de la couche de base dans les conditions appropriées à un affichage avec résolution spatiale réduite. En outre, pour une telle opération, il est souhaitable de configurer un décodeur de faible complexité. Ce genre de caractéristique fonctionnelle est utile pour les applications dans lesquelles le récepteur ne peut pas ou ne «veut» pas afficher la résolution spatiale complète qui est apportée par les deux couches et pour les applications où l'on souhaite procéder à un décodage logiciel. La méthode décrite dans la suite du texte fait appel à la syntaxe d'échelonnabilité SNR brièvement exposée à l'article 7 pour la transmission des données vidéo dans les deux couches. Il est à noter qu'aucune des options proposées ne modifie la structure du décodeur, les conditions étant identiques à celles que présente la Figure 7-14. Le flux binaire créé sur les deux couches est compatible avec le profil syntaxique de type SUPÉRIEUR. Néanmoins, le décodeur de couche de base pourrait être utilisé avec une configuration différente, à complexité réduite, appropriée au décodage logiciel.

D.7.2.1 Configuration du décodeur

Dans le décodage destiné à une résolution spatiale inférieure, on pourrait utiliser une transformation DCT inverse de format réduit pour le décodage de la couche de base, ce qui permettrait de réduire en conséquence les besoins relatifs à la mémoire d'images bi-frames dans la boucle de compensation cinétique du décodeur.

Si le flux binaire des deux couches à échelonnabilité SNR est créé avec une seule boucle de compensation cinétique au niveau du codeur, les données vidéo de base subissent une dérive. Celle-ci peut ou non être acceptable, selon l'application. Dans une large mesure, la qualité d'image dépendra de la précision de sous-échantillonnage utilisée pour la compensation de mouvement dans le décodeur. Il est possible d'employer le vecteur de mouvement à haute précision tel que transmis dans la couche de base pour la compensation de mouvement, avec une précision de sous-échantillonnage comparable à celle de la couche supérieure. On peut minimiser la dérive en utilisant des filtres perfectionnés à interpolation des sous-échantillons (voir les références bibliographiques [12], [13] et [16] de l'Annexe F).

D.7.2.2 Configuration du codeur

Il est possible d'adapter le flux binaire à échelonnabilité SNR de la couche de base aux besoins spécifiques du décodeur à résolution échelonnable. Une transformation DCT en un format réduit peut être intégrée plus facilement si l'on transmet seulement les coefficients DCT appropriés au sous-ensemble correspondant, dans le flux binaire de la couche de base.

Enfin, il est possible d'assurer un décodage sans dérive à un échelon de résolution inférieur en incorporant plusieurs boucles de compensation de mouvement dans la configuration du codeur. Le processus de reconstruction est le même pour le codeur et le décodeur.

D.7.3 Allocation du débit binaire dans la subdivision des données

La subdivision des données permet de subdiviser un flux binaire pour améliorer l'élasticité aux erreurs lorsque deux canaux n'ayant pas la même protection contre les erreurs sont disponibles. Il est souvent nécessaire de contraindre le débit binaire de chaque subdivision, ce que l'on obtient au niveau du codeur en modifiant de manière adaptative le point de rupture prioritaire pour chaque tranche.

Le codeur peut faire appel à deux mémoires virtuelles pour les deux flux binaires et procéder à un contrôle de débit de rétroaction en choisissant un point d'inflexion de priorité qui correspond approximativement au débit recherché pour chaque canal. La différence entre l'objectif de débit et le débit effectif permet de corriger la valeur d'objectif pour l'image bi-trame suivante, grâce à une boucle de rétroaction.

Il est souhaitable de faire varier la subdivision du débit binaire d'une image bi-trame à l'autre afin d'améliorer l'élasticité aux erreurs. Généralement, les images I s'améliorent si elles ont une plus grande quantité de données dans la subdivision 0 par rapport aux images P, tandis que les images B pourraient s'insérer entièrement dans la subdivision 1.

D.7.4 Echelonnabilité temporelle

Une structure de codage à échelonnabilité temporelle comprenant deux couches (couche de base et couche d'amélioration) est décrite à la Figure D.1. Prenons l'hypothèse d'une entrée des données vidéo en base de temps principale dans le démultiplexeur temporel; en l'occurrence, le démultiplexage temporel vise à constituer deux séquences vidéo, une à l'entrée du codeur de la couche de base et l'autre à l'entrée du codeur de la couche d'amélioration. Le codeur de la couche de base est un codeur non hiérarchique fonctionnant en demi-base de temps, tandis que le codeur de la couche d'amélioration est assimilable à un codeur à profil syntaxique de type PRINCIPAL qui fonctionne également en demi-base de temps si ce n'est qu'il utilise des images codées de la couche de base pour la prédiction utilisant la compensation de mouvement. Les flux binaires codés des couches de base et d'amélioration sont multiplexés en un flux unique dans le multiplexeur du système. Celui-ci extrait deux flux binaires et présente les flux binaires correspondants à l'entrée des décodeurs de la couche de base et de la couche d'amélioration. La sortie du décodeur de la couche de base peut être représentée de manière indépendante en demi-base de temps ou après le multiplexage avec les images bi-frames décodées de la couche d'amélioration; elle peut aussi être représentée en base de temps principale.

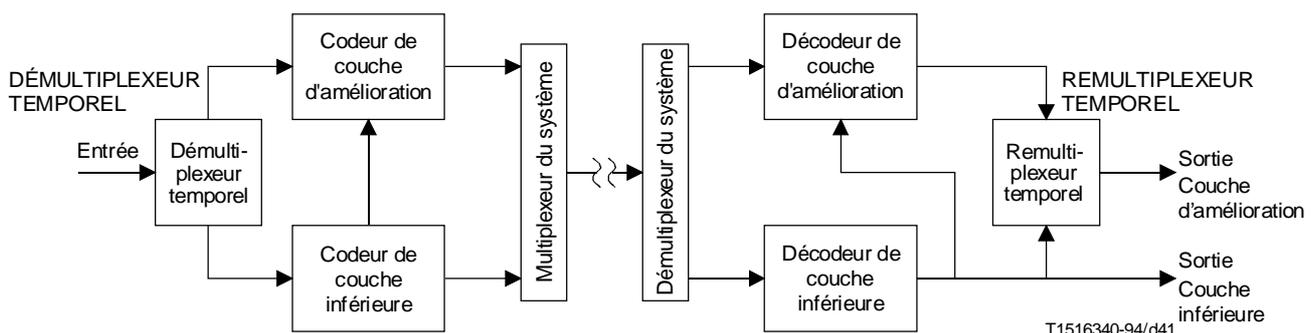


Figure D.1 – Structure de codec à deux couches pour l'échelonnabilité temporelle

Les formes ci-après d'échelonnabilité temporelle sont supportées et exprimées selon le format d'image à échelonnabilité temporelle avec couches supérieures: couche de base et couche d'amélioration.

- 1) Echelonnabilité temporelle avec couches supérieures (d'entrée) à balayage progressif: couche de base à balayage progressif et couche d'amélioration à balayage progressif.

- 2) Echelonnabilité temporelle avec couches supérieures (d'entrée) à balayage progressif: couche de base à balayage entrelacé et couche d'amélioration à balayage entrelacé.
- 3) Echelonnabilité temporelle avec couches supérieures (d'entrée) à balayage entrelacé: couche de base à balayage entrelacé et couche d'amélioration à balayage entrelacé.

D.7.4.1 Echelonnabilité temporelle avec couches supérieures (d'entrée) à balayage progressif – Couche de base à balayage progressif et couche d'amélioration à balayage progressif

En prenant l'hypothèse d'une entrée de données vidéo à balayage progressif, il est nécessaire d'effectuer un codage à balayage progressif dans les couches de base et d'amélioration, l'opération de *démultiplexage temporel* pouvant être relativement simple et consistant en un démultiplexage temporel des images bi-trames d'entrée pour former deux séquences à balayage progressif; l'opération de *remultiplexage temporel* est l'inverse de l'opération précédente, c'est-à-dire qu'elle consiste à accomplir le remultiplexage de deux séquences à balayage progressif pour obtenir à la sortie des données à balayage progressif en base de temps principale (voir la Figure D.2).

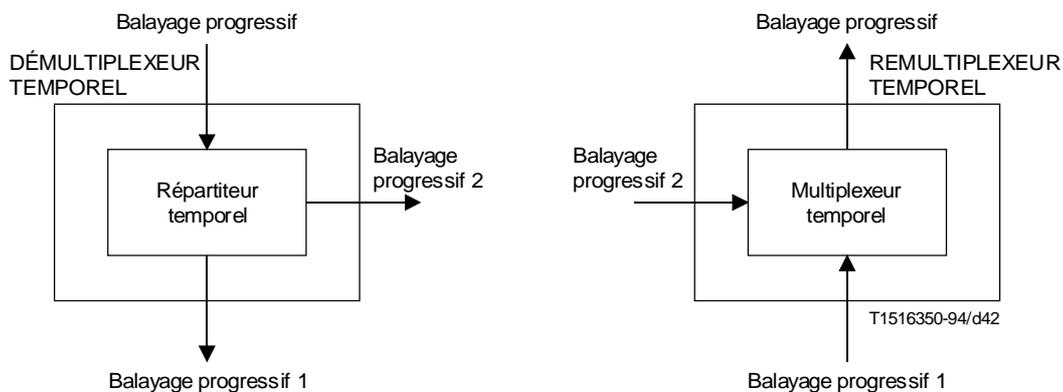


Figure D.2 – Démultiplexage et remultiplexage temporels pour l'échelonnabilité temporelle avec couches supérieures d'entrée à balayage progressif – Couche de base à balayage progressif et couche d'amélioration à balayage progressif

D.7.4.2 Echelonnabilité temporelle avec couches supérieures (d'entrée) à balayage progressif – Couche de base à balayage entrelacé et couche d'amélioration à balayage entrelacé

Là encore, en prenant l'hypothèse d'une entrée de données vidéo à balayage progressif en base de temps principale, il est nécessaire d'effectuer un codage à balayage entrelacé dans la couche de base, l'opération du *démultiplexage temporel* pouvant faire intervenir la conversion d'une image bi-trame à balayage progressif en deux images monotrames à balayage entrelacé; ce processus consiste à extraire, en balayage entrelacé, une séquence normale et une séquence complémentaire à partir des données vidéo d'entrée à balayage progressif. L'opération du *remultiplexage temporel* est l'inverse de l'opération précédente, c'est-à-dire qu'elle consiste à accomplir une conversion de deux images monotrames à balayage entrelacé en une image bi-trame à balayage progressif pour obtenir à la sortie des données à balayage progressif en base de temps principale. Les Figures D.3 et D.4 décrivent les opérations qu'il est nécessaire d'effectuer pour les deux types de conversion (balayage progressif à balayage entrelacé et inversement).

D.7.4.3 Echelonnabilité temporelle avec couches supérieures (d'entrée) à balayage entrelacé – Couche de base à balayage entrelacé et couche d'amélioration à balayage entrelacé

En prenant l'hypothèse d'une entrée de données vidéo à balayage entrelacé, il est nécessaire d'effectuer un codage à balayage entrelacé dans les couches de base et d'amélioration, l'opération de *démultiplexage temporel* pouvant être relativement simple et consistant en un démultiplexage temporel des images bi-trames d'entrée pour former deux séquences à balayage entrelacé; l'opération de *remultiplexage temporel* est l'inverse de l'opération précédente, c'est-à-dire qu'elle consiste à accomplir le remultiplexage de deux séquences à balayage entrelacé pour obtenir à la sortie des données à balayage entrelacé en base de temps principale. Les opérations de démultiplexage et de remultiplexage sont analogues à celles décrites à la Figure D.2.

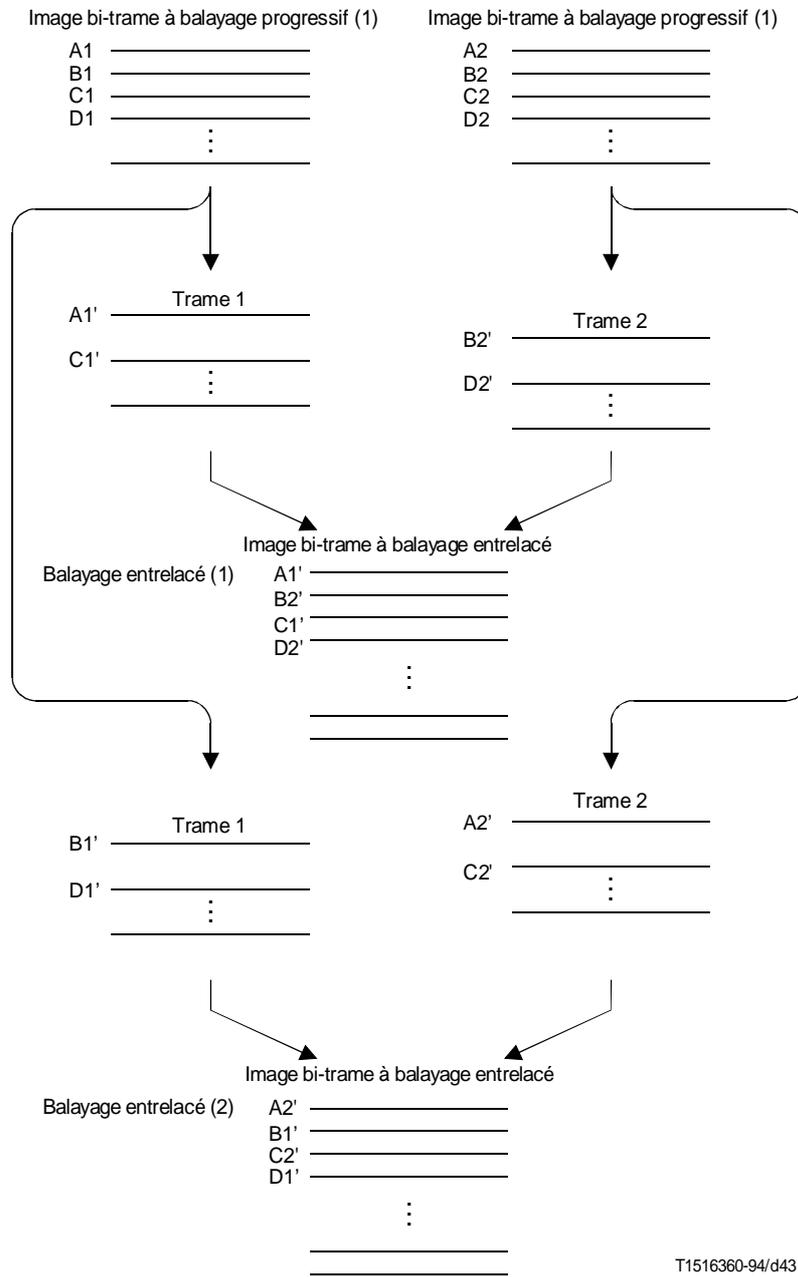


Figure D.3 – Conversion d'une image bi-trame à balayage progressif en deux images monotrames à balayage entrelacé

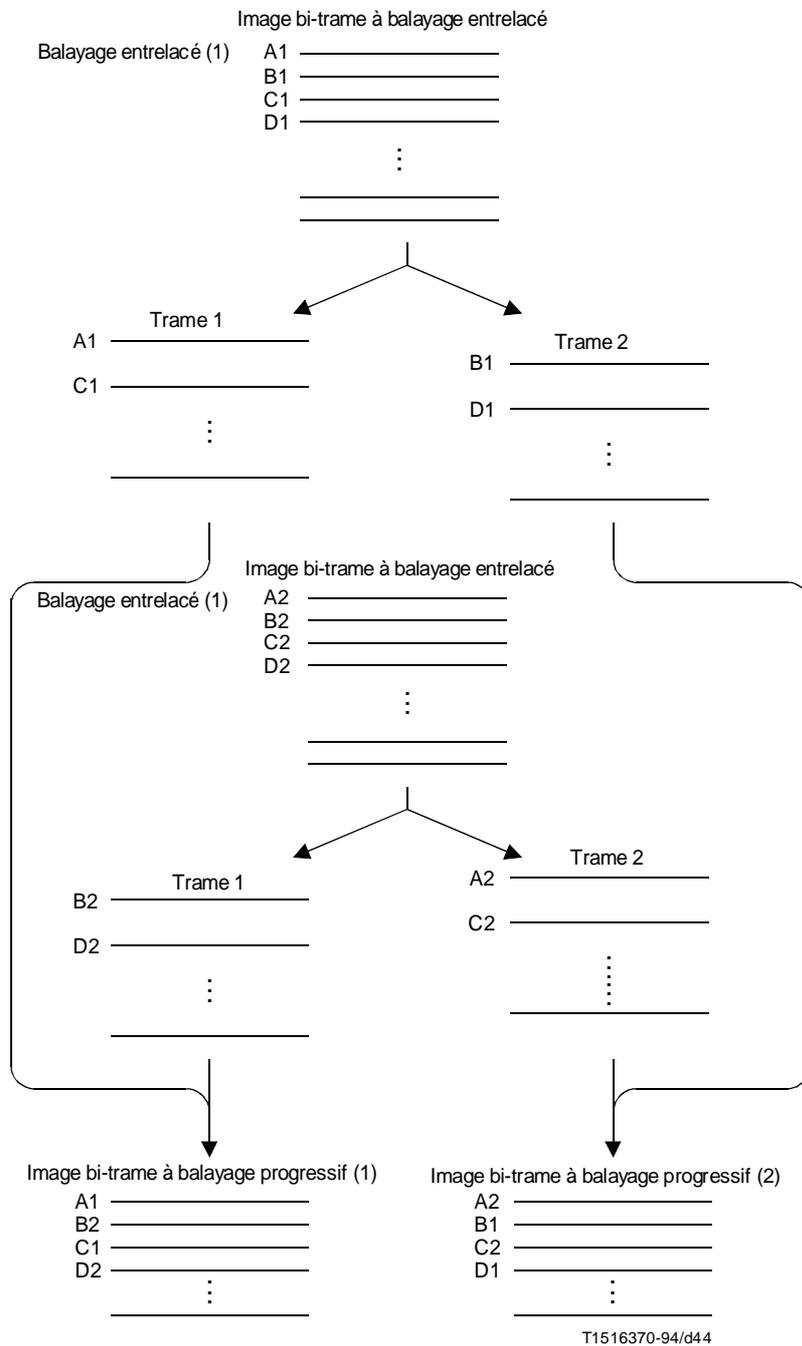


Figure D.4 – Conversion de deux images monotrames à balayage entrelacé en une image bi-trame à balayage progressif

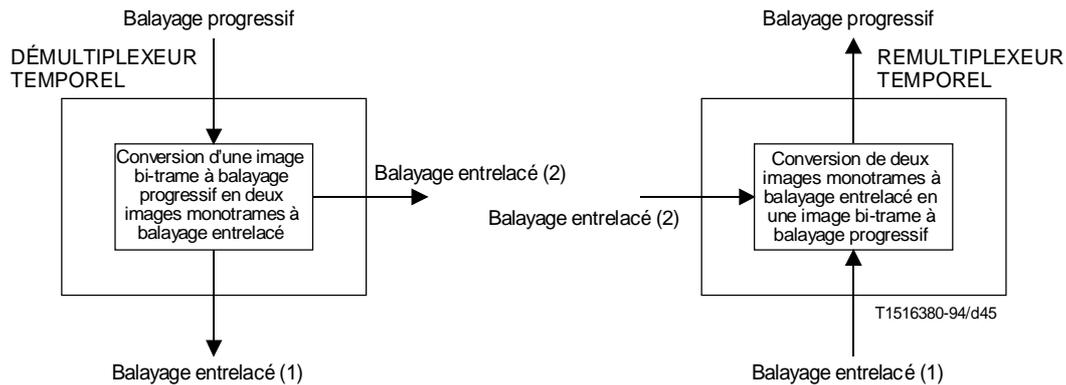


Figure D.5 – Démultiplexage et remultiplexage temporels pour l'échelonnabilité temporelle avec couches supérieures (d'entrée) à balayage progressif – Couche de base à balayage entrelacé et couche d'amélioration à balayage entrelacé

D.7.5 Hybrides des extensions à l'échelonnabilité spatiale, à l'échelonnabilité SNR et à l'échelonnabilité temporelle

La présente norme permet également les combinaisons d'outils d'échelonnabilité visant à établir plus de deux couches vidéo si cette option se révèle utile et pratique pour supporter des applications plus exigeantes. Le fait d'associer à chaque fois deux types d'échelonnabilité produit trois combinaisons explicites. De plus, pour chaque combinaison, il est possible d'obtenir des applications distinctes si l'on intervertit l'ordre dans lequel chaque échelonnabilité est appliquée. S'agissant des échelonnabilités hybrides à trois couches, on distingue la couche de base, la couche d'amélioration 1 et la couche d'amélioration 2.

D.7.5.1 Applications avec échelonnabilité hybride de type spatial et SNR

- a) *TVHD avec télévision conventionnelle à deux niveaux de qualité:*

La couche de base fournit la résolution de télévision conventionnelle en qualité de base, la couche d'amélioration 1 contribue à créer la résolution de télévision conventionnelle – mais en qualité supérieure – par échelonnabilité SNR, et la couche d'amélioration 2 utilise la résolution et le format de la TVHD en effectuant un codage avec échelonnabilité spatiale par rapport à la résolution de la télévision conventionnelle de qualité élevée obtenue à l'aide de la couche d'amélioration 1.

- b) *Télévision conventionnelle à deux niveaux de qualité et vidéophonie/télévision à basse définition:*

La couche de base fournit la qualité de vidéophonie/télévision à basse définition, la couche d'amélioration 1 crée la résolution de télévision conventionnelle en qualité de base par échelonnabilité spatiale, et la couche d'amélioration 2 fait appel à l'échelonnabilité SNR pour contribuer à créer la résolution de télévision conventionnelle de qualité élevée.

- c) *TVHD à deux niveaux de qualité et télévision conventionnelle:*

La couche de base fournit la résolution de télévision conventionnelle, la couche d'amélioration 1 fournit la résolution de TVHD en qualité de base par échelonnabilité spatiale, et la couche d'amélioration 2 fait appel à l'échelonnabilité SNR pour contribuer à créer la résolution de TVHD de qualité élevée.

D.7.5.2 Applications avec échelonnabilité hybride de type spatial et temporel

- a) *TVHD à balayage progressif et à résolution temporelle élevée avec TVHD de base à balayage entrelacé et télévision conventionnelle:*

La couche de base fournit la résolution de télévision conventionnelle, la couche d'amélioration 1 fournit la résolution de TVHD de base à balayage entrelacé à l'aide de l'échelonnabilité spatiale, et la couche d'amélioration 2 fait appel à l'échelonnabilité temporelle pour contribuer à créer le format de TVHD à balayage progressif et à résolution en base de temps principale.

- b) *TVHD à balayage progressif et à résolution élevée avec TVHD améliorée à balayage progressif et TVHD de base à balayage progressif:*

La couche de base fournit le format de TVHD de base à balayage progressif en résolution temporelle, la couche d'amélioration 1 utilise l'échelonnabilité temporelle pour contribuer à fournir le format de TVHD à balayage progressif en résolution en base de temps principale, et la couche d'amélioration 2 utilise l'échelonnabilité spatiale pour fournir le format de TVHD à balayage progressif en résolution spatiale élevée (résolution en base de temps principale).

- c) *TVHD à balayage progressif à résolution élevée avec TVHD améliorée à balayage progressif et TVHD de base à balayage entrelacé:*

La couche de base fournit le format de TVHD de base à balayage entrelacé, la couche d'amélioration 1 utilise l'échelonnabilité temporelle pour contribuer à fournir le format de TVHD à balayage progressif en résolution en base de temps principale, et la couche d'amélioration 2 utilise l'échelonnabilité spatiale pour fournir le format de TVHD à balayage progressif et à résolution spatiale élevée (résolution en base de temps principale).

D.7.5.3 Applications avec échelonnabilité hybride de type temporel et SNR

- a) *TVHD améliorée à balayage progressif avec TVHD de base à balayage progressif à deux niveaux de qualité:*

La couche de base fournit la TVHD de base à balayage progressif en base de temps secondaire, la couche d'amélioration 1 utilise l'échelonnabilité temporelle pour contribuer à fournir le format de TVHD à balayage progressif en base de temps principale, mais avec qualité de base, et la couche d'amélioration 2 utilise l'échelonnabilité SNR pour contribuer à fournir le format de TVHD à balayage progressif de qualité élevée (résolution en base de temps principale).

- b) *TVHD améliorée à balayage progressif avec TVHD de base à balayage entrelacé à deux niveaux de qualité:*

La couche de base fournit le format de TVHD à balayage entrelacé en qualité de base, la couche d'amélioration 1 utilise l'échelonnabilité SNR pour contribuer à fournir le format de TVHD à balayage entrelacé de qualité élevée, et la couche d'amélioration 2 utilise l'échelonnabilité temporelle pour contribuer à fournir le format de TVHD à balayage progressif avec résolution en base de temps principale (qualité élevée).

D.8 Compatibilité

La présente norme assure la compatibilité avec différents formats de résolution et est compatible avec ISO/CEI 11172-2 (et la Recommandation H.261).

D.8.1 Compatibilité avec les formats de résolution plus élevée et moins élevée

La présente Spécification supporte la compatibilité avec différents formats de résolution du support vidéo. La compatibilité est assurée pour les résolutions spatiale et temporelle grâce aux outils d'échelonnabilité spatiale et temporelle. Le support vidéo est codé en deux couches de résolution. Un décodeur uniquement capable ou susceptible d'afficher des données vidéo de résolution moins élevée accepte et décode le flux binaire de couche inférieure. Il est possible de reconstruire les données vidéo à résolution principale en acceptant et en décodant les deux couches de résolution fournies.

D.8.2 Compatibilité avec ISO/CEI 11172-2 (et la Recommandation H.261)

La syntaxe de la présente Spécification supporte la compatibilité ascendante et descendante avec ISO/CEI 11172-2. La compatibilité descendante avec ISO/CEI 11172-2 est assurée étant donné que la syntaxe de la présente Spécification est un surensemble de la syntaxe de ISO/CEI 11172-2. L'outil d'échelonnabilité spatiale fourni par la présente Spécification permet d'utiliser le codage de type ISO/CEI 11172-2 lorsque la résolution est moins élevée (couche de base), ce qui permet d'obtenir la compatibilité ascendante.

La syntaxe vidéo comporte les outils requis pour assurer la compatibilité avec la Recommandation H.261 en vue d'une utilisation future éventuelle, mais ce scénario n'est actuellement supporté par aucun profil.

La diffusion jumelée de données de chrominance est une simple variante qui permet d'assurer la compatibilité ascendante avec la Recommandation H.261 et avec ISO/CEI 11172-2.

D.9 Différences entre la présente Spécification et ISO/CEI 11172-2

Le présent paragraphe fait le point des différences entre le codage vidéo de type MPEG-1 et le codage vidéo de type MPEG-2.

Tous les décodeurs vidéo MPEG-2 compatibles avec les profils syntaxiques et avec les niveaux de contraintes actuellement définis sont indispensables pour le décodage des flux binaires contraints de type MPEG-1.

Dans la plupart des cas, le système MPEG-2 représente un surensemble du système MPEG-1. Par exemple, le balayage en zigzag des coefficients MPEG-1 est l'un des deux modes de balayage des coefficients MPEG-2. Néanmoins, il arrive parfois que des éléments syntaxiques (ou sémantèmes) du système MPEG-1 n'aient pas d'équivalent direct dans le système MPEG-2. La présente Spécification récapitule l'ensemble de ces éléments.

En outre, la présente Spécification peut aider les responsables de la mise en œuvre à identifier les éléments syntaxiques (ou sémantèmes) vidéo MPEG-1 qui n'ont pas d'équivalent direct en MPEG-2 et qui nécessitent donc une attention particulière par rapport aux garanties de compatibilité MPEG-1.

Dans ce paragraphe, MPEG-1 renvoie à ISO/CEI 11172-2 et MPEG-2 renvoie à la présente Spécification.

D.9.1 Désadaptation de transformée en cosinus discrète inverse (IDCT)

MPEG-1 – Le contrôle de désadaptation de transformée DCT inverse consiste à ajouter (ou à retrancher) une unité à chaque coefficient non nul qui aurait été pair après une quantification inverse. Voir la description faite dans le cadre du processus de quantification inverse aux 2.4.4.1, 2.4.4.2 et 2.4.4.3 de MPEG-1.

MPEG-2 – Le contrôle de désadaptation de DCT inverse consiste à ajouter (ou à retrancher) une unité au coefficient [7] [7] si la somme de tous les coefficients est paire après une quantification inverse. Voir la description faite au 7.4.4 de MPEG-2.

D.9.2 Bourrage de macrobloc

MPEG-1 – Il est possible d'insérer le code à longueur variable '0000 0001 111' [remplissage de macrobloc (structure *macroblock_stuffing*)] un nombre quelconque de fois avant chaque progression d'adresse de macrobloc (structure *macroblock_address_increment*). Ce code doit être rejeté par le décodeur. Voir la description faite au 2.4.2.7 de MPEG-1.

MPEG-2 – Ce code à longueur variable est réservé et n'est pas utilisé dans MPEG-2 qui prévoit uniquement la possibilité d'un bourrage par insertion d'octets de bourrage avant un code de déclenchement. Voir la description faite au 5.2.3 de MPEG-2.

D.9.3 Syntaxe d'échappement vers un autre niveau d'exécution

MPEG-1 – Les valeurs d'exécution qu'il n'est pas possible de coder avec un code à longueur variable sont codées avec le code d'échappement '0000 01' suivi d'un code à longueur fixe de 14 bits ($-127 \leq \text{niveau} \leq 127$) ou d'un code de longueur fixe à 22 bits ($-255 \leq \text{niveau} \leq 255$). Voir la description faite en 2-B5, à l'Annexe B de MPEG-1.

MPEG-2 – Les valeurs d'exécution qu'il n'est pas possible de coder avec un code à longueur variable sont codées avec le code d'échappement '0000 01' suivi d'un code à longueur fixe de 18 bits ($-2047 \leq \text{niveau} \leq 2047$). Voir la description faite au 7.2.2.3 de MPEG-2.

D.9.4 Position horizontale des échantillons de chrominance

MPEG-1 – La position horizontale des échantillons de chrominance est à mi-chemin entre les échantillons de luminance. Voir la description faite au 2.4.1 de MPEG-1.

MPEG-2 – La position horizontale des échantillons de chrominance est la même que celle des échantillons de luminance. Voir la description faite au 6.1.1.8 de MPEG-2.

D.9.5 Tranches

MPEG-1 – Les tranches ne doivent pas nécessairement commencer et prendre fin sur le même rang de macroblocs. En conséquence, il est possible que tous les macroblocs d'une image se trouvent dans une seule tranche. Voir la description faite au 2.4.1 de MPEG-1.

MPEG-2 – Les tranches débutent et s'achèvent toujours sur la même ligne horizontale de macroblocs. Voir la description faite au 6.1.2 de MPEG-2.

D.9.6 Images D

MPEG-1 – Une syntaxe spéciale est définie pour les images D (type de codage d'image avec structure *picture_coding_type* = 4). Comme les images I, les images D comportent seulement les coefficients DC des blocs à codage intra; elles sont dépourvues de fin de bloc et comportent un code spécial '1' de fin de macrobloc (structure *end_of_macroblock*).

MPEG-2 – Les images de type D (*picture_coding_type* = 4) ne sont pas autorisées. Voir la description faite au 6.3.9 de MPEG-2.

D.9.7 Vecteurs de mouvement à codage sur tous les pels

MPEG-1 – Les éléments syntaxiques de vecteur de mouvement anticipé à codage sur tous les pels (structure *full_pel_forward_vector*) et de vecteur de mouvement différé à codage sur tous les pels (structure *full_pel_backward_vector*) peuvent être mis à '1'. En pareil cas, les vecteurs de mouvement codés sont constitués d'unités à codage sur tous les pels et non pas à codage sur un pel sur deux. Il faut multiplier par deux les coordonnées de vecteur de mouvement avant de les utiliser pour la prédiction. Voir la description faite aux 2.4.4.2 et 2.4.4.3 de MPEG-1.

MPEG-2 – Les éléments syntaxiques *full_pel_forward_vector* et *full_pel_backward_vector* doivent avoir la valeur '0'. Ces vecteurs de mouvement sont toujours codés sur un pel sur deux.

D.9.8 Information de format

MPEG-1 – La valeur format de pel (structure *pel_aspect_ratio*) codée sur 4 bits dans l'en-tête de séquence spécifie le format de pel. Voir la description faite au 2.4.3.2 de MPEG-1.

MPEG-2 – La valeur information de format (structure *aspect_ratio_information*) codée sur 4 bits dans l'en-tête de séquence spécifie le format d'affichage. On obtient le format de pel à partir de cette valeur et à partir du format des images bi-frames et du format d'affichage. Voir la description faite au 6.3.3 de MPEG-2.

D.9.9 forward_f_code and backward_f_code

MPEG-1 – Les valeurs du code f (structure *f_code*) utilisées pour le décodage des vecteurs de mouvement sont celles du code f anticipé (structure *forward_f_code*) et du code f différé (structure *backward_f_code*), situées dans l'en-tête de l'image [structure *picture_header()*].

MPEG-2 – Les valeurs du code f utilisées pour le décodage des vecteurs de mouvement sont *f_code* [s][t], situées dans l'extension relative au codage des images () [structure *picture_coding_extension()*]. Il faut donner la valeur '111' à *forward_f_code* et à *backward_f_code* et ne pas en tenir compte. Voir la description faite au 6.3.9 de MPEG-2.

D.9.10 constrained_parameter_flag et horizontal_size maximum

MPEG-1 – Lorsque le fanion de flux paramétrique contraint (structure *constrained_parameter_flag*) est mis à '1', un certain nombre de contraintes sont vérifiées. L'une de ces contraintes est que la dimension horizontale (structure *horizontal_size*) soit inférieure ou égale à 768. Il convient de noter qu'un flux binaire contraint de données vidéo MPEG-1 peut comporter des images dont la dimension horizontale va jusqu'à 768 pels. Voir la description faite au 2.4.3.2 de MPEG-1.

MPEG-2 – Le mécanisme *constrained_parameter_flag* a été remplacé par le mécanisme des profils à deux niveaux. Néanmoins, il convient de noter que les flux binaires MP @ ML (MP (*main profil*) = profil principal, ML (*main level*) = niveau principal) ne peuvent pas avoir une dimension horizontale supérieure à 720 pels. Voir la description faite au 8.2.3.1 de MPEG-2.

D.9.11 Structures bit_rate et vbv_delay

MPEG-1 – Les structures *bit_rate* et *vbv_delay* sont mises respectivement aux valeurs 3FFF et FFFF (hex) afin d'indiquer un débit variable. Les autres valeurs concernent le débit binaire constant.

MPEG-2 – La sémantique de la structure *bit_rate* a changé. En mode de débit variable, le délai du vérificateur *vbv_delay* peut être mis à la valeur FFFF (hex), mais une valeur différente n'implique pas nécessairement que le débit est constant. Le mode de débit constant n'est qu'un cas particulier du mode de débit variable. Rien ne permet d'affirmer qu'un flux binaire est à débit constant sans examiner toutes les valeurs de la structure *vbv_delay* et sans effectuer de complexes calculs.

Même si le débit binaire est constant, la valeur de la structure *bit_rate* peut ne pas représenter le débit binaire réel car il suffit que la structure *bit_rate* soit une limite supérieure du débit binaire réel.

D.9.12 Vérificateur VBV

MPEG-1 – Le vérificateur VBV n'est défini que pour le mode de fonctionnement à débit constant. Le décodeur STD remplace le modèle VBV pour le fonctionnement en débit variable.

Norme MPEG-2 – Le vérificateur VBV n'est défini que pour le mode de fonctionnement à débit variable. Le fonctionnement en débit constant est considéré comme un cas particulier du débit variable.

D.9.13 Structure temporal_reference

MPEG-1 – La valeur de la structure de référence temporelle temporal_reference est augmentée de 1 modulo 1024 pour chaque image codée; elle est remise à zéro à chaque en-tête de groupe d'images.

Norme MPEG-2 – S'il n'y a pas d'images complètes, la référence temporelle est augmentée de 1 modulo 1024 pour chaque image codée et remise à zéro à chaque en-tête de groupe d'images (comme en Norme MPEG-1). S'il y a des images complètes (dans des flux binaires à faible délai), la référence temporelle suit des règles différentes.

D.9.14 Rapprochement des syntaxes MPEG-1 et MPEG-2

Il est possible de constituer des flux binaires MPEG-2 ayant une syntaxe très proche de la syntaxe MPEG-1 en utilisant des valeurs particulières pour les différents éléments syntaxiques MPEG-2 qui n'existent pas dans la syntaxe MPEG-1.

En d'autres termes, le processus de décodage MPEG-1 est identique (sauf pour les points précis mentionnés précédemment) au processus de décodage MPEG-2 lorsque les structures de la liste ci-après prennent les valeurs indiquées:

```

progressive_sequence = '1' (séquence à balayage progressif)
chroma_format = '01' (4:2:0)
frame_rate_extension_n = 0 et frame_rate_extension_d = 0 (fréquence d'image MPEG-1)
intra_dc_precision = '00' (codage intra par coefficients DC à précision de 8 bits)
picture_structure = '11' (image bi-trame, parce que progressive_sequence = '1')
frame_pred_frame_dct = 1 (seulement la prédiction fondée sur les images bi-frames et le codage DCT pour image bi-trame)
concealment_motion_vectors = '0' (pas de vecteur de mouvement de masquage)
q_scale_type = '0' (quantiser_scale de type linéaire)
intra_vlc_format = '0' (tableau de codage à longueur variable MPEG-1 pour les macroblocs à codage intra)
alternate_scan = '0' (balayage en zigzag MPEG-1)
repeat_first_field = '0' (parce que progressive_sequence = '1')
chroma_420_type = '1' (la chrominance est fondée sur les images bi-frames parce que progressive_sequence = '1')
progressive_frame = '1' (parce que progressive_sequence = '1')

```

D.10 Complexité

La Norme MPEG-2 supporte les combinaisons de décodeurs à haute et à faible performance/complexité. Cela est rendu possible par des définitions de profil et de niveau qui, pour chaque nouveau profil, présentent une nouvelle série d'outils et de caractéristiques fonctionnelles. Il est donc possible d'atténuer la lourdeur inhérente aux structures performantes de codage MPEG-2 en diminuant la complexité de réalisation.

De plus, certaines restrictions pourraient permettre de réduire les coûts de réalisation des systèmes de décodage.

D.11 Edition des flux binaires codés

La présente Spécification supporte de nombreuses opérations sur le flux binaire codé afin d'éviter le coût et les pertes de qualité du processus de recodage. L'édition et la concaténation des flux binaires codés sont possibles sans reprogrammation et sans rupture de la séquence des images décodées.

Il existe une incompatibilité entre la nécessité d'une compression élevée et la simplicité des opérations d'édition. La syntaxe et la structure de codage n'ont pas été conçues dans le dessein principal de simplifier l'édition de telle ou telle image. Néanmoins, un certain nombre de caractéristiques sont prévues pour permettre l'édition des données codées.

La présente Spécification permet l'édition des flux binaires MPEG-2 codés du fait de la hiérarchie syntaxique du flux binaire vidéo codé. Des codes de déclenchement uniques sont codés à différents niveaux dans la hiérarchie (par exemple séquence vidéo, groupe d'images, etc.). Il est possible de coder les données vidéo avec des points d'accès aux images/tranches à codage intra dans le flux binaire. Cela permet d'identifier, aux fins d'accès et d'édition, certaines parties du flux binaire sans nécessairement décoder l'intégralité du support vidéo.

D.12 Modes d'enrichissement

Certains supports numériques d'enregistrement (DSM) (*Digital Storage Media*) offrent la possibilité d'utiliser des modes d'enrichissement (par exemple, lecture rapide en avant/en arrière). La syntaxe MPEG-2 supporte tous les modes spéciaux d'accès, de recherche et d'exploration définis dans ISO/CEI 11172-2. Les caractéristiques fonctionnelles de l'enrichissement sont supportées par la hiérarchie syntaxique du flux binaire de données vidéo qui permet d'identifier les parties recherchées dans une séquence vidéo. On peut aussi s'appuyer sur les outils MPEG-2 qui offrent la possibilité d'une échelonabilité du flux binaire afin de limiter le débit binaire d'accès (subdivision des données et structure générale en tranches). Le présent paragraphe fournit des directives applicables au décodage d'un flux binaire issu d'un support DSM.

Le décodeur est informé au moyen d'un fanion de mode d'enrichissement par support DSM (structure *DSM_trick_mode_flag*) à 1 bit dans l'en-tête du PES (flux élémentaire paqueté). Ce fanion indique que le flux binaire est reconstruit par le support DSM en mode d'enrichissement, le flux binaire étant valable du point de vue syntaxique mais non valable du point de vue sémantique. Le bit de ce fanion est suivi d'une zone (*DSM_trick_modes*) à 8 bits. Les sémantèmes de *DSM_trick_modes* sont définis dans la Rec. UIT-T H.222.0 | ISO/CEI 13818-1.

D.12.1 Décodeur

Lorsque le décodeur décode le paquet de flux PES dont le fanion *DSM_trick_mode_flag* est mis à 1, il est recommandé:

- de décoder ce flux binaire et de procéder à l'affichage conformément à la structure *DSM_trick_modes*.

Prétraitement

Lorsque le décodeur rencontre le paquet de flux PES dont le fanion *DSM_trick_mode_flag* est mis à 1, il est recommandé:

- de libérer la mémoire du flux binaire non porteur de mode d'enrichissement.

Post-traitement

Lorsque le décodeur rencontre le paquet de flux PES dont le fanion *DSM_trick_mode_flag* est mis à zéro, il est recommandé:

- de libérer la mémoire du flux binaire porteur de mode d'enrichissement.

Partie vidéo

Lorsque le décodeur décode le paquet de flux PES dont le fanion *DSM_trick_mode_flag* est mis à 1, il est recommandé:

- d'ignorer les valeurs du délai de vérificateur de mémoire vidéo (structure *vbv_delay*) et de la référence temporelle (structure *temporal_reference*);
- de décoder une image et de l'afficher jusqu'au décodage de l'image suivante.

En mode d'enrichissement, le flux binaire peut présenter un intervalle entre certaines tranches. Lorsque le décodeur rencontre un intervalle entre tranches, il est recommandé:

- de décoder la tranche et de l'afficher selon sa position verticale indiquée dans l'en-tête de tranche;
- de combler l'intervalle avec la partie située au même endroit de la dernière image affichée.

D.12.2 Codeur

Il est recommandé:

- d'utiliser un codage à court format pour les tranches avec macroblocs à codage intra;
- d'utiliser un codage à brève régénération périodique par image ou tranche à codage intra.

Support numérique d'enregistrement (DSM)

Il est recommandé que le support numérique d'enregistrement fournisse les flux binaires en mode d'enrichissement avec une syntaxe parfaite.

Prétraitement

Pour le support numérique d'enregistrement, il est recommandé:

- de fournir le flux binaire «normal» conformément à l'en-tête d'image() [structure *picture_header()*] et aux structures syntaxiques supérieures.

Partie système

Pour le support numérique d'enregistrement, il est recommandé:

- de mettre le fanion *DSM_trick_mode_flag* à 1 dans un en-tête de paquet de flux PES;
- de donner au fanion *DSM_trick_modes* (8 bits) une valeur conforme au mode d'enrichissement.

Partie vidéo

Pour le support numérique d'enregistrement, il est recommandé:

- d'insérer un en-tête *sequence_header()* avec le même paramètre que pour le flux binaire normal;
- d'insérer un en-tête d'extension de séquence avec les mêmes paramètres que pour le flux binaire normal;
- d'insérer un en-tête d'image *picture_header()* avec les mêmes paramètres que pour un flux binaire normal, sauf qu'il est parfois préférable d'indiquer le mode de fonctionnement à débit variable. Une façon d'y arriver consiste à donner la valeur FFFF (hex) à la structure *vbv_delay*;
 NOTE – Etant donné que le décodeur ignore les éléments *temporal_reference* et *vbv_delay*, le support numérique d'enregistrement n'a pas besoin de donner des valeurs correctes à ces deux structures.
- de concaténer les tranches qui sont constituées de macroblocs à codage intra. Il convient de classer les tranches concaténées par ordre croissant de position verticale.

D.13 Robustesse aux erreurs

La plupart des supports numériques d'enregistrement et des canaux de communication ne sont pas protégés contre les erreurs. La question des systèmes appropriés de codage des canaux qu'il convient d'utiliser sort du cadre de la présente Spécification. Toutefois, la syntaxe MPEG-2 supporte des modes robustes aux erreurs pour les pertes de cellules dans les réseaux à mode de transfert asynchrone (ATM) (*Asynchronous Transfer Mode*) et aux erreurs binaires (isolées et par rafales) en cours de transmission. La structure en tranches du système de compression défini dans la présente Spécification permet à un décodeur d'opérer une reprise sur erreur d'information résiduelle et d'assurer la resynchronisation du décodage. En conséquence, les erreurs binaires qui entachent les données comprimées se traduiront par des erreurs d'étendue limitée au niveau des images décodées. Le décodeur peut faire appel à des techniques de masquage pour masquer de telles erreurs. La robustesse peut prendre la forme d'une dégradation progressive par rapport au taux d'erreur sur les bits et une reprise progressive par rapport à des paquets de données ou à des éléments binaires omis dans le support vidéo. Il convient de noter que des dispositions supplémentaires peuvent être nécessaires pour faire en sorte que tous les éléments soient supportés au niveau du système.

En tant qu'exemple de système fonctionnant en mode paquets, le RNIS à large bande et son mode ATM font l'objet d'une analyse assez détaillée dans la suite du texte. Il est possible d'appliquer un raisonnement analogue à d'autres systèmes pour lesquels certains paquets de données sont protégés individuellement au moyen d'un codage à correction d'erreurs sans voie de retour.

Le mode ATM repose sur l'utilisation de petits paquets de longueur fixe appelés «cellules», qui comprennent un en-tête de 5 octets pour l'information d'acheminement et une capacité utile de 48 octets. Les erreurs possibles correspondent à des pertes de cellules ou à des bits erronés dans le message utile. Selon les caractéristiques fonctionnelles de la couche d'adaptation ATM (AAL) (*ATM Adaptation Layer*), des indications relatives aux pertes de cellules et aux cellules entachées d'erreurs binaires peuvent être fournies.

Afin de mieux cerner l'impact des pertes de cellules en mode ATM, on se reportera au Tableau D.2 qui donne les valeurs de l'intervalle moyen entre pertes de cellules pour une gamme de taux de perte de cellules (CLR) (*Cell Loss Ratio*) et de débits binaires de service correspondant à une modélisation statistique simple (à cet effet, il faut partir de l'hypothèse selon laquelle il existe une capacité utile de cellule. Avec 1 octet par cellule pour les fonctions de couche AAL, on obtient 376 bits = 47 octets). Toutefois, il est à noter que ce tableau récapitulatif ne tient pas compte des pertes de cellules par rafales et d'autres statistiques temporelles à plus court terme.

Tableau D.2 – Intervalle moyen entre pertes de cellules pour différents taux de pertes des cellules et de débits binaires de service

| | Intervalle moyen entre les erreurs | | | |
|-----------|------------------------------------|-----------|-----------|------------|
| | 5 Mbit/s | 10 Mbit/s | 50 Mbit/s | 100 Mbit/s |
| 10^{-2} | 7,52 ms | 3,76 ms | 0,752 ms | 0,376 ms |
| 10^{-3} | 75,2 ms | 37,6 ms | 7,52 ms | 3,76 ms |
| 10^{-4} | 752 ms | 376 ms | 75,2 ms | 37,6 ms |
| 10^{-5} | 7,52 s | 3,76 s | 752 ms | 376 ms |
| 10^{-6} | 1,25 min | 37,6 s | 7,52 s | 3,76 s |
| 10^{-7} | 12,5 min | 6,27 min | 1,25 min | 37,6 s |
| 10^{-8} | 2,09 h | 1,04 h | 12,5 min | 6,27 min |

Pour des erreurs binaires isolées, il est possible de calculer facilement le taux d'erreur sur les bits (BER) (*Bit Error Ratio*) correspondant aux intervalles moyens entre erreurs indiqués dans le tableau ci-dessus. On obtient la valeur du taux d'erreur sur les bits qui aurait les mêmes conséquences en divisant le taux de perte de cellules par la capacité utile de cellule, c'est-à-dire 376.

Pour mémoire, un certain nombre de techniques sont décrites dans la suite du texte car elles permettent de minimiser l'impact des pertes de cellules et d'autres erreurs/pertes; des exemples sont fournis pour montrer comment utiliser les différents outils disponibles dans la présente Spécification afin d'obtenir une bonne protection contre les erreurs en question. Il est à noter que les techniques exposées peuvent être applicables à des paquets ayant un autre format (par exemple, réseaux locaux ou certains supports d'enregistrement) ou à des données vidéo qui comportent des erreurs non corrigées ayant des caractéristiques différentes, outre les pertes de cellules. Il peut être indiqué d'assimiler un effacement déterminé [détection d'erreur(s) sur les bits non corrigée(s) dans un bloc de données] à une perte de bloc de données, étant donné qu'une prédiction de l'impact des erreurs binaires est impossible. Néanmoins, il convient de supporter une telle caractéristique à titre d'option dans le décodeur. D'une manière générale, l'analyse qui suit fait état, le cas échéant, de «paquets de transport», l'objectif étant d'insister sur le fait que le principe est applicable à plusieurs systèmes de transport et d'enregistrement. En tout état de cause, dans certains exemples, il est fait référence au taux de perte de cellules parce que, au moment de l'élaboration de la présente Spécification, le transport des cellules était le système défini de la manière la plus complète.

Les techniques qui permettent d'obtenir une certaine robustesse aux erreurs sont réparties dans trois catégories, à savoir: méthodes de masquage de l'erreur une fois que celle-ci est apparue, limitation de l'impact d'une perte ou d'une erreur du point de vue spatial (dans une image) et limitation de cet impact dans le temps (d'une image à l'autre).

D.13.1 Méthodes de masquage

Les méthodes de masquage permettent de masquer l'effet des pertes/erreurs une fois que celles-ci se sont produites. Certaines méthodes peuvent être appliquées à l'aide de n'importe quel flux binaire codé, tandis que d'autres dépendent du codeur lorsqu'il s'agit de structurer les données ou de fournir des informations supplémentaires permettant d'améliorer les performances.

D.13.1.1 Masquage à prédiction temporelle

Un décodeur peut masquer les erreurs en estimant les pertes de données d'après les données adjacentes du point de vue spatio-temporel. A cet effet, le décodeur utilise les informations reçues correctement pour évaluer les éléments qui devraient être affichés à la place des données perdues/erronées, étant entendu que les caractéristiques iconographiques sont relativement semblables dans les blocs adjacents (en termes de dimensions spatiale et temporelle). Du point de vue temporel, l'opération revient à estimer les données erronées ou perdues sur la base des trames ou des images bi-trames situées à proximité.

D.13.1.1.1 Substitution à partir de l'image bi-trame précédente

Le moyen le plus simple consiste à remplacer un macrobloc perdu par le macrobloc situé au même endroit dans l'image précédente. Cette méthode est adaptée pour les zones d'image relativement statiques; par contre, un déplacement de bloc apparaît pour les zones en mouvement.

Il est nécessaire d'interpréter avec soin l'expression «image précédente» du fait de la prédiction bidirectionnelle et de la différence entre l'ordre de décodage et l'ordre d'affichage des images. Un macrobloc perdu dans une image P ou I peut être masqué par copie des données correspondant au même macrobloc dans l'image P ou I précédente. On s'assure ainsi que l'image est complète avant de l'utiliser pour la suite des opérations de prédiction. Les macroblocs perdus dans des images B peuvent être remplacés par ceux de la dernière image affichée, quel que soit son type, ou par ceux d'une image I ou P postérieure, stockée en mémoire mais pas encore affichée.

D.13.1.1.2 Masquage utilisant la compensation de mouvement

Il est possible d'améliorer le masquage à partir des images voisines en procédant à une estimation des vecteurs de mouvement pour le macrobloc perdu, sur la base des vecteurs de mouvement correspondant aux macroblocs voisins dans l'image affectée (à condition que les macroblocs en question ne soient pas perdus). Bien que le procédé permette d'améliorer le masquage des zones d'image en mouvement, les erreurs entachant les macroblocs entourés de macroblocs voisins à codage intra posent un problème manifeste parce qu'il n'existe généralement pas de vecteurs de mouvement en pareil cas. L'assistance que fournit le codeur en vue de résoudre ce problème est traitée au D.13.1.1.3.

Une estimation élaborée des vecteurs de mouvement pourrait consister à mémoriser les vecteurs de mouvement des macroblocs adjacents situés au-dessus et au-dessous du macrobloc perdu, en ce qui concerne les prédictions anticipée et différée (images B) dans le temps. Cela permettrait de déterminer la moyenne entre les vecteurs de mouvement correspondant aux macroblocs situés au-dessus et au-dessous (s'ils sont disponibles) du macrobloc perdu.

Avec un décodeur moins complexe, on pourrait par exemple se contenter d'une prédiction anticipée et/ou du vecteur de mouvement correspondant au macrobloc situé au-dessus du macrobloc perdu; il en résulterait des économies de mémoire et d'interpolation.

D.13.1.1.3 Utilisation de vecteurs de mouvement à codage intra

Normalement, le masquage utilisant la compensation de mouvement décrit au D.13.1.1.2 ne pourrait pas être appliqué lorsque les macroblocs situés au-dessus et au-dessous du macrobloc perdu/erroné sont à codage intra, étant donné qu'aucun vecteur de mouvement n'est associé à ce type de macrobloc. En particulier, dans les images I, un tel masquage serait impossible avec la détermination et l'utilisation classiques des vecteurs de mouvement.

On peut étendre le processus de codage pour y inclure les vecteurs de mouvement correspondant aux macroblocs à codage intra. Naturellement, il est nécessaire de transmettre séparément (par exemple, dans des paquets différents) le vecteur de mouvement et l'information codée pour un macrobloc donné afin que le vecteur de mouvement reste disponible en cas de perte des données vidéo.

Lorsque les vecteurs de mouvement de masquage (structure *concealment_motion_vectors*) sont égaux à 1, on transmet les vecteurs de mouvement avec les macroblocs à codage intra, ce qui permet d'améliorer la prestation des décodeurs en matière de masquage. Le vecteur de mouvement de masquage associé à un macrobloc avec codage intra est destiné à être utilisé uniquement pour le masquage (si nécessaire) du macrobloc situé directement au-dessous du macrobloc à codage intra.

Pour simplifier, les vecteurs de mouvement de masquage associés à des macroblocs à codage intra sont toujours anticipés et, par ailleurs, sont assimilés à des vecteurs de mouvement d'image bi-trame dans des images bi-frames et à des vecteurs de mouvement de trame dans des images monotrames.

En conséquence, il convient de faire en sorte que les codeurs programmés en vue de créer des vecteurs de mouvement de masquage transmettent, pour un macrobloc à codage intra donné, le vecteur de mouvement d'image bi-trame ou monotrane destiné à être utilisé pour le masquage (c'est-à-dire, pour la prédiction anticipée à base d'images bi-frames ou la prédiction anticipée à base de trames, respectivement) visant le macrobloc situé directement au-dessous du macrobloc à codage intra.

Les vecteurs de mouvement de masquage sont essentiellement prévus pour les images I et P, mais la syntaxe permet leur utilisation pour les images B. Dans ce dernier cas, l'opération de masquage ne joue pas un rôle crucial étant donné que les images B ne tiennent pas lieu de prédicteurs, si bien que les erreurs ne se propagent pas à d'autres images. En conséquence, il peut être superflu de transmettre des vecteurs de mouvement de masquage dans les images B.

Les vecteurs de mouvement de masquage transmis avec les macroblocs à codage intra situés dans la rangée inférieure d'une image ne peuvent pas être utilisés pour le masquage. Néanmoins, si leur valeur est égale à 1, les vecteurs «concealment_motion_vectors» doivent être transmis. Le codeur peut avoir recours au vecteur de mouvement (0, 0) pour réduire au minimum la servitude binaire du codage.

Lorsqu'on utilise les vecteurs de mouvement de masquage, il est judicieux de faire en sorte qu'une tranche contienne une rangée de macroblocs (ou moins) afin de pouvoir limiter le masquage à moins d'une rangée de macroblocs en cas de perte d'une tranche ou d'une partie de tranche. Il y a donc beaucoup moins de risques de pertes de macroblocs dans deux rangées successives, ce qui augmente les possibilités d'obtenir un résultat efficace avec les vecteurs de mouvement de masquage.

NOTE – Lorsque la valeur de la structure «concealment_motion_vectors» est égale à 1, les prédicteurs de vecteur de mouvement NE sont PAS réinitialisés en cas de transmission d'un macrobloc avec codage intra. Généralement, la présence d'un tel macrobloc a pour effet de réinitialiser les prédicteurs de vecteur de mouvement.

D.13.1.2 Masquage à prédiction spatiale

On peut aussi procéder à un masquage prédictif en créant les macroblocs correspondants par interpolation à partir des macroblocs voisins dans une même image (voir la référence bibliographique [17] dans l'Annexe F). Cette technique est la mieux adaptée pour les zones à mouvement intense, auxquelles la prédiction temporelle ne se prête pas, ou en tant que variante de masquage pour les macroblocs avec codage intra, en l'absence de vecteur de mouvement de masquage (voir D.13.1.1.3). Une telle technique pourrait aussi être particulièrement utile en cas de pertes de cellules survenant après les changements de scène.

Il existe plusieurs scénarios possibles pour l'interpolation spatiale, qui peut être exécutée dans le domaine spatial ou dans le domaine DCT, mais, en principe, c'est une méthode applicable et utile uniquement pour la prédiction des caractéristiques principales d'un macrobloc perdu (par exemple, coefficient DC et, éventuellement, coefficients AC les plus faibles). La prédiction spatiale des détails fins (fréquences élevées) risque d'échouer et, en tout état de cause, elle est d'une utilité réduite pour les images à mouvement rapide.

Le masquage des macroblocs par prédiction spatiale présente également de l'intérêt lorsqu'il est utilisé en combinaison avec des méthodes de codage en stratification (c'est-à-dire, subdivision des données ou pyramide à échelonnabilité SNR; voir D.13.1.3). Lorsque, en cas de perte de cellules, certains coefficients DCT d'un macrobloc sont récupérés depuis la couche inférieure, il est possible d'utiliser l'ensemble des informations qui sont disponibles (coefficients DCT récupérés dans le même macrobloc à partir de la couche inférieure et tous les coefficients DCT reçus dans les macroblocs adjacents) pour le masquage des erreurs. Cette technique est particulièrement utile si la couche inférieure comporte uniquement les coefficients DC en raison de contraintes inhérentes à la largeur de bande.

D.13.1.3 Utilisation du codage en stratification pour faciliter le masquage

Il est possible d'apporter une contribution supplémentaire au processus de masquage en disposant les données vidéo codées de manière à offrir les meilleures garanties pour la réception des informations essentielles. Dès lors, la perte des informations moins importantes peut être masquée avec une efficacité accrue. En l'occurrence, on peut avoir intérêt à utiliser un support de transmission ou un dispositif d'enregistrement fonctionnant à différents niveaux de priorité (par exemple, transmission sur la base de cellules avec contrôle de priorité dans le RNIS à large bande, ou bien utilisation de différents types de protection ou de correction d'erreur sur des canaux différents). Il est possible de classer les composantes issues du processus de codage par ordre d'importance selon l'effet des pertes sur l'image reconstruite. En indiquant la priorité des composantes du flux binaire et en traitant les composantes individuelles d'après leur importance propre, on peut éventuellement améliorer les résultats du masquage des erreurs.

Parmi les techniques qui permettent de produire des flux binaires, ou des couches, classés par ordre hiérarchique figurent les possibilités suivantes:

Subdivision des données – Les données de macrobloc codées sont subdivisées en plusieurs couches, la subdivision 0 contenant des informations d'adresse et de contrôle ainsi que les coefficients DCT d'ordre inférieur, la subdivision 1 contenant pour sa part les coefficients DCT à fréquence élevée.

Echelonnabilité SNR – Deux séries de coefficients sont déquantifiées et additionnées au récepteur avant le décodage. Une série de coefficients pourrait servir à raffiner l'erreur de quantification de l'autre série, mais des combinaisons différentes (y compris une émulation de la subdivision des données) sont possibles.

Echelonnabilité spatiale – Il est possible de coder la couche inférieure sans tenir compte de la couche d'amélioration, mais on peut aussi utiliser d'autres méthodes de codage classiques (Norme ISO/CEI 11172-2, etc.). La couche d'amélioration contient l'erreur de prédiction découlant d'une prédiction reposant sur la couche inférieure.

Echelonnabilité temporelle – La couche d'amélioration définit les images supplémentaires et, après remultiplexage avec la couche de base, permet d'obtenir une séquence d'images combinées ayant une fréquence d'image plus élevée.

L'utilisation de ces méthodes permet d'obtenir des couches qui, lorsqu'elles sont additionnées progressivement, produisent une qualité croissante de la séquence reconstruite. Bien que certaines de ces techniques de codage à la source puissent entraîner une augmentation du débit binaire par rapport aux systèmes non stratifiés, il est possible que la protection des systèmes stratifiés soit meilleure face aux erreurs qui se produisent sur les canaux.

Du point de vue de la robustesse aux erreurs uniquement, il convient de traiter les couches classées hiérarchiquement d'après leur qualité propre en vue d'optimiser telle ou telle fonction (par exemple, qualité d'image correspondant à un débit binaire total donné). Les composantes du flux binaire peuvent être traitées différemment à un ou à plusieurs des niveaux suivants:

- *codeur* – Possibilité d'utiliser différents codages de canal;
- *canal* – Possibilité d'offrir aux différentes composantes du flux binaire une variété de probabilités de perte de cellules/paquets ou de caractéristiques d'erreur;
- *décodeur* – Possibilité d'offrir plusieurs types de masquage d'erreur dans chaque flux binaire.

D.13.1.3.1 Utilisation de la subdivision des données

La subdivision des données permet une répartition simple des données de macrobloc en deux couches. Le pointeur PBP (*Priority Break Point*) (point de rupture prioritaire) détermine le contenu de chaque couche. Généralement, la subdivision 0 contient l'information d'adresse et de contrôle ainsi que les coefficients DCT à basse fréquence, tandis que la subdivision 1 contient les coefficients DCT à fréquence élevée.

Au niveau du codeur, la valeur du pointeur susmentionné peut être différente pour chaque tranche afin qu'il soit possible de contrôler (c'est-à-dire de maintenir constante) la répartition des bits entre les deux couches. Cette répartition peut être différente pour les images bi-trames I, P et B. Dans le cas de certains macroblocs, la gestion du débit entre les deux couches peut se traduire par la constitution d'une subdivision 0 ne contenant aucun coefficient DCT et aucun vecteur de mouvement.

Il est possible d'obtenir une bonne robustesse aux erreurs si les erreurs des canaux sont réparties de telle sorte que la subdivision 1 reçoive la plupart d'entre elles.

On suppose que les erreurs sont détectables par le décodeur, l'objectif étant de pouvoir intervenir pour empêcher l'affichage de données erronées. Pour la subdivision 1, les données erronées ne sont pas du tout affichées (ce qui revient à utiliser seulement la subdivision 0). Il convient de réduire au minimum les pertes ou les erreurs dans la subdivision 0 en utilisant un transport hautement fiable. En outre, il peut s'avérer nécessaire de procéder à un masquage dans le décodeur.

D.13.1.3.2 Utilisation du codage à échelonnabilité SNR

Le codage à échelonnabilité SNR fournit deux couches à résolution spatiale identique mais de qualité d'image différente, selon que l'on code une seule couche ou les deux à la fois. Cette technique vise essentiellement à fournir une couche de qualité inférieure utilisable même sans la présence d'une couche d'amélioration. Néanmoins, l'autre avantage est la bonne robustesse aux erreurs si l'on peut arriver à limiter essentiellement les erreurs à la couche d'amélioration.

S'agissant des erreurs de la couche d'amélioration, le signal de couche inférieure ne peut être utilisé que pour la zone d'image affectée. C'est surtout dans le cas d'erreurs fréquentes, de pertes temporaires ou d'indisponibilité permanente de la couche d'amélioration que ce type de masquage est très efficace, étant donné que le signal affiché peut être relativement dépourvu de distorsions non linéaires telles que le broutage ou le sautellement.

Si la couche d'amélioration est indisponible en permanence et que, dans ces conditions, seule la couche inférieure soit décodée, une petite dérive peut intervenir lorsqu'une seule boucle de prédiction utilisant la compensation de mouvement est prévue dans le codeur. Toutefois, il est vraisemblable que cette dérive restera invisible dans la plupart des configurations (par exemple, avec $M = 3$ et $N = 12$, on obtient normalement une correction suffisamment fréquente).

La couche inférieure d'un système à échelonnabilité SNR est bien adapté au masquage en cas de taux d'erreur très élevé ou bien de pertes temporaires ou permanentes du signal de la couche supérieure. Néanmoins, la qualité de la couche supérieure en l'absence d'erreurs n'atteint pas celle d'une structure stratifiée du type à codage en sous-bande (par exemple, subdivision des données).

D.13.1.3.3 Utilisation du codage à échelonnabilité spatiale

Le codage à échelonnabilité spatiale permet de coder la couche inférieure sans tenir compte de la couche d'amélioration, d'où la possibilité d'utiliser aussi d'autres méthodes de codage classiques (ISO/CEI 11172-2, etc.). La couche d'amélioration contient le signal de différence codé découlant d'une prédiction fondée sur la couche inférieure. En cas d'erreur dans la couche d'amélioration, il est possible d'utiliser directement le signal converti sur la base de la couche inférieure comme information de masquage pour la zone d'image affectée. C'est surtout en présence d'erreurs fréquentes ou de pertes temporaires du signal de la couche d'amélioration que ce signal de masquage se présente en étant relativement dépourvu de distorsions non linéaires telles que le broutage (qui peut survenir si les coefficients DCT à fréquence élevée sont entièrement absents de la couche inférieure) ou le sautellement (si l'information de mouvement est omise dans la couche à degré de priorité élevé).

En l'absence d'erreurs, on utilise le signal converti sur la base de la couche inférieure comme signal de prédiction supplémentaire pour un mode adaptatif en termes de macroblocs afin d'améliorer le codage de la couche supérieure. En conséquence, le flux binaire de la couche d'amélioration comporte les erreurs de prédiction résiduelles quantifiées en mode temporel ou à partir de la couche inférieure.

Le codage à échelonnabilité spatiale fournit un signal de couche inférieure très bien adapté au masquage en cas de taux d'erreur élevé ou de perte temporaire de la couche d'amélioration. Toutefois, la qualité de l'image améliorée que l'on obtient, lorsque les deux couches sont disponibles, n'est généralement pas aussi élevée que dans les autres systèmes de codage stratifié.

D.13.1.3.4 Utilisation du codage à échelonnabilité temporelle

L'échelonnabilité temporelle est une technique de codage qui permet la stratification des images bi-frames. La résolution spatiale de ces images est la même dans chaque couche, mais la base de temps des différentes couches est inférieure à celle de la source. Toutefois, la base de temps combinée des deux couches donne la base de temps principale de la source. En cas d'erreurs dans la couche d'amélioration, on peut utiliser facilement la couche de base avec la résolution spatiale principale pour le masquage. C'est notamment lorsque des erreurs fréquentes ou des pertes temporaires affectent le signal de la couche d'amélioration que la couche de base offre des caractéristiques de masquage intéressantes.

Dans certaines applications de télécommunication, on pourrait obtenir un degré élevé d'élasticité aux erreurs avec l'échelonnabilité temporelle en codant la couche de base à l'aide de la même résolution spatiale, mais seulement en demi-résolution temporelle de la source; les images bi-frames restantes qui correspondent à l'autre moitié de la résolution temporelle sont codées dans la couche d'amélioration. Généralement, il est possible d'attribuer un degré de priorité inférieur aux données de la couche d'amélioration et, en cas de pertes, les images bi-frames décodées de la couche de base peuvent être utilisées pour le masquage grâce à un processus de répétition des images bi-frames. Ce type de masquage entraîne uniquement une perte temporaire de la résolution temporelle principale tout en maintenant intégralement la qualité et la résolution spatiales.

Dans les applications de TVHD où la source est un support vidéo à balayage progressif avec résolution temporelle élevée, l'échelonnabilité temporelle permet d'obtenir un degré élevé d'élasticité aux erreurs. Dans ce genre d'application, on envisage l'utilisation d'un système reposant sur deux couches – de base et d'amélioration – qui traitent chacune les mêmes formats d'image (à balayage progressif ou à balayage entrelacé pour les deux couches), mais en demi-base de temps. Le remultiplexage temporel des signaux de couche de base et d'amélioration, qui s'effectue indépendamment du format choisi, permet toujours le maintien de la résolution temporelle principale, avec le balayage progressif existant à la source. Pour les émissions de TVHD, si le signal de la couche d'amélioration à degré de priorité moins élevé a subi une dégradation, il est possible d'utiliser le signal de la couche de base pour le masquage, soit directement comme dans le cas de la couche de base avec balayage progressif, soit après une inversion de la parité des trames pour la couche de base à balayage entrelacé.

D'une manière générale, il est possible d'attribuer un degré de priorité moins élevé aux données de la couche d'amélioration et, en cas de perte, les images bi-frames décodées de la couche de base peuvent être utilisées pour le masquage grâce à l'évaluation d'une moyenne sur la base des images bi-frames. Ce type de masquage entraîne uniquement une perte temporaire invisible de la résolution temporelle principale tout en maintenant la qualité et la résolution spatiales à leur niveau de base de temps principale.

D.13.2 Localisation spatiale

La localisation spatiale englobe les méthodes visant à minimiser le degré de propagation des erreurs à l'intérieur d'une image en assurant rapidement la resynchronisation des éléments du flux binaire qui font l'objet d'un codage différentiel entre macroblocs.

Il est possible de déceler les erreurs binaires isolées à l'aide des mots de code non valables, si bien qu'un concepteur de décodeur peut décider d'admettre le décodage d'une séquence erronée. Néanmoins, l'effet produit sur l'image est difficile à prévoir (des mots de code autorisés, mais incorrects, pourraient être créés); il peut être préférable de contrôler les erreurs en opérant un masquage de la totalité de la tranche ou des tranches affectées, même lorsqu'un seul bit est identifié comme étant erroné à l'intérieur d'un bloc de données.

En cas de longues erreurs consécutives (par exemple, perte de paquets ou de cellules), il n'y a pratiquement qu'une seule possibilité: rejeter les données jusqu'à la localisation du point de resynchronisation suivant (effectuée grâce à un code de déclenchement au niveau de la tranche suivante ou de l'en-tête d'image suivant). En augmentant le nombre de points de resynchronisation fournis, il est possible de réduire la zone de l'écran affectée par une perte ou par une erreur, ce qui, du même coup, permet d'être moins exigeant vis-à-vis du masquage et de rendre les erreurs moins visibles, aux dépens de l'efficacité du codage. En conséquence, la localisation spatiale des erreurs dépend du contrôle du format des tranches étant donné que la tranche est la plus petite unité codée pourvue de points de resynchronisation (codes de déclenchement).

D.13.2.1 Tranches à format réduit

La méthode la plus simple pour la localisation spatiale des erreurs consiste à diminuer le nombre (fixe) de macroblocs dans une tranche. L'accroissement de la fréquence des points de resynchronisation réduit la zone affectée de l'image en cas de perte. C'est une méthode efficace quel que soit le support de transport ou d'enregistrement et quel que soit le profil, étant donné que la structure des tranches est toujours présente en format vidéo avec codage MPEG.

L'utilisation d'une telle méthode entraîne une légère diminution de l'efficacité du codage, due à l'augmentation des informations de servitude binaire. La perte d'efficacité équivaut environ à 3% pour 11 macroblocs par tranche et à 12% pour 4 macroblocs par tranche sur la base du format d'image spécifié dans la Recommandation UIT-R. BT.601 à 4 Mbit/s (pourcentages calculés par rapport à un système utilisant 44 macroblocs, ou une largeur d'image, par tranche). Cette perte d'efficacité se traduit par une dégradation de la qualité de l'image pouvant aller jusqu'à environ 1 dB avec 4 macroblocs par tranche et jusqu'à 0,2 dB avec 11 macroblocs par tranche sans erreur à 4 Mbit/s. Toutefois, la méthode donne des résultats améliorés d'environ 1 à 5 dB pour un taux de perte de cellules égal à 10^{-2} , en fonction de la technique de masquage utilisée (simple substitution de macrobloc ou masquage utilisant la compensation de mouvement).

Du point de vue de la qualité de l'image perçue, l'efficacité de la méthode dépend généralement du format relatif des tranches et de l'image. Il convient donc de déterminer le format des tranches en prenant en considération celui de l'image (en macroblocs) ainsi que le compromis entre l'efficacité du codage et la dégradation visuelle due aux erreurs.

D.13.2.2 Format de tranche adaptatif

Le nombre de bits requis pour le codage d'une tranche d'image est très variable en fonction du mode de codage, de l'activité des images, etc. Si les tranches contiennent à peine quelques macroblocs, il est possible qu'un paquet de transport, même un paquet ou une cellule de petite dimension, puisse contenir plusieurs tranches. Le fait de prévoir plusieurs points de resynchronisation dans le même paquet de transport n'apporte aucune solution. Il existe un autre problème en ce qui concerne l'approche simpliste des tranches à format réduit: en effet, étant donné que l'on ne tient pas compte de la structure des paquets de transport, le premier paquet de transport valable après une perte pourrait comporter la plus grande partie de l'information correspondant à une tranche; mais ce paquet serait inutilisable en raison de la perte du code de déclenchement.

L'utilisation d'un format adaptatif est une amélioration possible par rapport au format réduit. Étant donné qu'il produit le flux binaire, le codeur suit l'évolution du contenu des données dans les paquets de transport. Le déclenchement d'une tranche intervient à la première occasion dans chaque paquet de transport (ou une fois sur deux, trois, ...). Cette méthode permet d'obtenir à peu près les mêmes résultats pour la localisation spatiale des erreurs que celle qui consiste à utiliser des tranches de petit format fixe, mais avec une efficacité accrue.

Toutefois, la méthode présente UNIQUEMENT un avantage pour la transmission par cellules ou par paquets, ou bien lorsque la détection des erreurs porte sur un grand bloc de données. Il est parfaitement superflu de prévoir des points de resynchronisation fréquents pour la localisation des petites tranches si plusieurs tranches sont perdues en cas d'erreur. De toute façon, si les erreurs binaires isolées n'affectent qu'une tranche, l'adaptation du format des tranches ne présente aucun intérêt.

Par ailleurs, cette technique adaptative nécessite une collaboration étroite entre le codeur et le dispositif de mise en paquets, l'objectif étant d'admettre une nouvelle tranche pour un nouveau paquet ou une nouvelle cellule. Intrinsèquement, le mode adaptatif n'est peut-être pas approprié pour certaines applications (par exemple, données vidéo enregistrées destinées à être distribuées au moyen de supports multiples) parce que l'on partirait du principe qu'il existe une seule structure de paquet de transport pendant le codage.

D.13.3 Localisation temporelle

La localisation temporelle englobe les méthodes qui visent à minimiser le degré de propagation des erreurs d'une image à l'autre dans la séquence temporelle en assurant rapidement la resynchronisation des images qui font l'objet d'un codage différentiel. Le codage intra s'offre alors comme une solution évidente.

D.13.3.1 Images à codage intra

L'utilisation d'images à codage intra permet d'éviter qu'une erreur unique se maintienne dans l'image décodée pendant une période supérieure à $(N + M - 1)$ images si chaque N ème image fait l'objet d'un codage intra et que $(M - 1)$ images B soient affichées avant chaque image I.

Alors que les images à codage intra, normalement utilisées comme «point d'ancrage» de la synchronisation du décodage vidéo dans une partie de séquence, sont utiles pour la localisation temporelle, il convient de veiller à ajouter des images à codage intra supplémentaires (ce qui revient à réduire N) afin d'assurer l'élasticité aux erreurs. Il faut un grand nombre de bits pour le codage intra des images qui, en outre, occupe une proportion relativement importante du flux binaire codé et risque donc plus d'être lui-même affecté par des pertes ou par des erreurs.

D.13.3.2 Tranches à codage intra

Afin d'éviter le délai additionnel associé aux images à codage intra, certaines applications nécessitant un faible délai peuvent comporter une actualisation des images, consistant seulement à effectuer un codage intra de certaines parties de l'image. Cela permet d'obtenir le même type de robustesse aux erreurs que pour les images à codage intra. Si, à titre d'exemple, on suppose qu'un nombre constant de tranches par image (de haut en bas) subit un codage intra de sorte que l'image complète est mise à jour à chaque image P, il convient de ne pas perdre de vue les trois aspects suivants pour le processus d'actualisation:

- Bien qu'une portion erronée de la scène soit normalement effacée dans les images P (avec une durée moyenne de l'ordre de $P/2$), il est possible que, du fait de la compensation de mouvement, la perturbation échappe à la régénération à codage intra et persiste pendant une durée pouvant aller jusqu'à 2 images P.
- Pour faire en sorte que les erreurs ne se propagent pas dans la zone actualisée de l'image, on pourrait appliquer des restrictions aux vecteurs de mouvement en limitant les composantes verticales du vecteur pour s'assurer qu'aucune prédiction n'est effectuée sur la base des parties «les plus anciennes» de l'image.
- L'effet visuel de la correction des erreurs est comparable à l'action d'un essuie-glace, qui chasse l'eau sur le pare-brise d'une voiture. Cet effet *d'essuie-glace* peut parfois devenir visible dans la séquence dépourvue d'erreurs à moins que le mécanisme de contrôle du débit binaire n'agisse pour maintenir la qualité de la tranche à codage intra à un niveau proche de celle des macroblocs voisins à codage non intra.

D.13.4 Résumé

Le Tableau D.3 récapitule les techniques qui permettent d'assurer la robustesse aux erreurs et donne des indications sur leur applicabilité.

Il est impossible de fournir une indication précise sur les caractéristiques de la robustesse aux erreurs parce que les évaluations sont nécessairement subjectives et dépendantes des applications, si bien que l'on doit se contenter d'utiliser cette analyse comme une orientation. Par ailleurs, il est vrai que plusieurs variantes peuvent être utilisées en combinaison pour assurer la robustesse aux erreurs. Il n'en reste pas moins que les indications qui suivent constituent une orientation sur les résultats obtenus. Ce sont les résultats d'expériences faites dans le domaine des pertes de cellules, uniquement du point de vue de la transmission des données vidéo sous forme de cellules.

Une simple substitution de macrobloc par rapport à une image bi-trame précédente, utilisée en combinaison avec la méthode des tranches de format réduit (4 macroblocs par tranche), permet d'obtenir une qualité d'image adéquate pour la plupart des séquences, en présence de taux d'erreur relativement faibles, correspondant à un taux de perte de cellules de 10^{-5} environ (dans un système de référence ayant les caractéristiques suivantes: 4 Mbit/s, profil principal, niveau principal).

Avec une technique élaborée de masquage à compensation de mouvement (interpolation spatiale et temporelle complète des vecteurs de mouvement pour les macroblocs perdus et masquage des pertes dans les images P faisant appel à une actualisation des tranches à codage intra, c'est-à-dire $N = \text{infini}$, $M = 1$), on obtient une qualité d'image appropriée pour un taux de perte de cellules égal à 10^{-3} (également dans un système de référence ayant les caractéristiques suivantes: 4 Mbit/s, profil principal, niveau principal).

En cas de pertes plus importantes, il peut s'avérer nécessaire d'utiliser l'une des méthodes de codage en stratification. Avec une protection appropriée de l'information à degré de priorité élevé, ces méthodes peuvent donner de bons résultats pour un taux de perte de cellules pouvant aller jusqu'à 10^{-2} , voire 10^{-1} . La subdivision des données, mise en œuvre à titre de post-traitement dans un système fonctionnant à 4 Mbit/s en profil principal et en niveau principal, avec attribution de 50% du débit à chaque subdivision moyennant l'absence de pertes dans la couche de base, a montré dans un cas précis que l'on pouvait obtenir les résultats suivants: pertes d'environ 0,5 dB en échelonnabilité SNR pour un taux de pertes de cellules de 10^{-3} et pertes d'environ 1,5 dB pour un taux de perte de cellules égal à 10^{-2} , pratiquement sans dégradation visible.

Tableau D.3 – Résumé des techniques de masquage d'erreur

| Catégorie | Technique | Profil/Applicabilité |
|-------------------------|--|--|
| Masquage | Prédiction temporelle – Substitution par rapport à l'image précédente | Tout profil. Technique particulièrement adaptée aux images statiques. |
| | Prédiction temporelle – Compensation de mouvement | Tout profil. Choix du degré d'élaboration dans l'estimation des vecteurs de mouvement. |
| | Prédiction temporelle – Utilisation de vecteurs de mouvement de masquage | Tout profil, mais détermination des vecteurs de mouvement à codage intra à titre d'option au niveau du codeur. |
| | Prédiction spatiale | Tout profil. Ne se prête pas aux images statiques et complexes. |
| | Subdivision des données | Technique n'existant actuellement dans aucun profil, mais susceptible d'être ajoutée sous la forme de post/prétraitement. Servitude binaire et complexité réduites au minimum. En fonction de l'allocation du débit binaire, il se peut que la couche inférieure ne soit pas en mesure de fournir à elle seule des images utilisables. |
| | Echelonnabilité SNR | Types de profils: ÉCHELONNABILITÉ SNR, ÉCHELONNABILITÉ SPATIALE et SUPÉRIEUR. Convient pour les taux d'erreur très importants ou pour l'indisponibilité temporaire de la couche d'amélioration. Mise en œuvre relativement simple. |
| | Echelonnabilité spatiale | Types de profils: ÉCHELONNABILITÉ SPATIALE et SUPÉRIEUR. Convient pour les taux d'erreur très importants ou pour l'indisponibilité temporaire de la couche d'amélioration. |
| | Echelonnabilité temporelle | Technique n'existant actuellement dans aucun profil. Convient pour les taux d'erreur très importants ou pour l'indisponibilité temporaire de la couche d'amélioration. |
| Localisation spatiale | Tranches de petit format | Tout profil. |
| | Formats de tranche adaptatifs | Tout profil, mais nécessité de connaître les caractéristiques de transmission au moment de la détermination de la taille des paquets. |
| Localisation temporelle | Images à codage intra | Tout profil, mais répercussions au niveau du délai. |
| | Tranches à codage intra | Tout profil, mais persistance éventuelle d'erreurs pendant une durée supérieure à celle de la méthode applicable aux images à codage intra. |

Etant donné qu'il existe plusieurs possibilités de codage en stratification, il peut être utile de formuler quelques observations globales. En général, on ne part pas du principe que l'utilisation des méthodes de codage en stratification les plus complexes pourrait être justifiée uniquement sur la base de l'élasticité aux erreurs. En fait, on pourrait utiliser de telles méthodes pour assurer la robustesse aux erreurs si leur fonction consistait à répondre aussi à d'autres besoins du système. La subdivision des données est très simple à mettre en œuvre et peut vraisemblablement donner des résultats de robustesse aux erreurs très proches de ceux des autres méthodes, sauf lorsque le taux d'erreur est extrêmement élevé (pertes > 10%) ou lorsque le signal de la couche d'amélioration risque d'être complètement perdu. L'échelonnabilité SNR est légèrement plus complexe et offre une efficacité relativement moindre que la subdivision des données; mais il est plus facile de produire des signaux de couche inférieure ayant une qualité utilisable lorsque la couche d'amélioration n'existe pas. Tout en étant encore plus complexe, l'échelonnabilité spatiale permet d'obtenir une bonne qualité d'images en couche inférieure, aux dépens de l'efficacité globale (dans les deux couches).

D.14 Séquences concaténées

Il y a enchaînement de séquences lorsqu'un flux élémentaire contient une fin de séquence dont le code de fin de séquence est suivi d'une autre séquence commençant par un code de déclenchement de séquence. Tout paramètre, y compris, mais non exclusivement, le profil, le niveau, la capacité de mémoire tampon de vérificateur VBV, le débit de trame, la dimension horizontale, la dimension verticale ou le débit, est susceptible de changer d'une séquence à l'autre, alors qu'aucune variation n'est autorisée à l'intérieur d'une même séquence.

Les caractéristiques des processus de décodage et d'affichage propres aux séquences concaténées sortent du champ d'application de la présente Recommandation | Norme internationale. Une application qui doit utiliser des séquences concaténées doit assurer, en faisant appel à des dispositions privées, que le décodeur sera en mesure de décoder et de restituer des séquences concaténées.

Les applications doivent assurer que les décodeurs fonctionneront de manière acceptable en présence d'un changement de valeur des paramètres. Tel devra être le cas en présence d'un changement de:

- dimension des images bi-frames;
- fréquence d'image;
- parité de trame de la première trame affichée de la nouvelle séquence par rapport à la parité de trame de la dernière trame affichée de la séquence précédente;
- état de la mémoire tampon;

entre autres paramètres dont les variations sont susceptibles de poser des problèmes.

Annexe E

Restrictions de profil et de niveau

(Cette annexe ne fait pas partie intégrante de la présente
Recommandation | Norme internationale)

E.1 Restrictions applicables aux éléments syntaxiques dans les profils

Sous forme de tableaux, la présente annexe récapitule tous les éléments syntaxiques définis dans la présente Spécification. Une classification est établie pour chaque élément afin d'indiquer s'il doit être supporté par un décodeur compatible avec un profil et un niveau donnés. Il est à noter que les spécifications normatives de conformité sont fournies dans ISO/CEI 13818-4.

NOTE – La présente annexe est incluse pour information et constitue simplement un résumé des restrictions normatives exposées à l'article 8. Si, en cas d'erreur survenue dans l'élaboration du texte, une divergence existe entre l'article 8 et la présente annexe, c'est le texte normatif de l'article 8 qui doit toujours l'emporter.

Le Tableau E.1 ci-dessous présente les différentes abréviations utilisées dans les Tableaux E.2 à E.19.

Tableau E.1 – Abréviations utilisées dans les tableaux de l'Annexe E

| Abréviation | Correspond à la catégorie | Signification |
|---|---------------------------|---|
| x | Mode | Doit être supporté par le décodeur. |
| o | Mode | Ne doit pas nécessairement être supporté par le décodeur. |
| D | Type | Elément comportant des paramètres qui dépendent du niveau (de contraintes). |
| I | Type | Elément indépendant du niveau dans le profil (syntaxique). |
| P | Type | Elément destiné au post-traitement après décodage; le décodeur doit être capable de décoder les flux binaires qui contiennent ces éléments, mais leur utilisation sort du cadre de la présente Recommandation Norme internationale. |
| NOTE – Un blanc est laissé en regard de «mode» pour les entrées qui ne sont pas des éléments syntaxiques. | | |

Tableau E.2 – En-tête de séquence

| # | Mode | | | | | | Type | |
|----------------------|--|---|---|---|---|---|--------------|---|
| | <div style="text-align: right; margin-right: 20px;"> SUPÉRIEUR SPATIAL SNR PRINCIPAL SIMPLE </div> | | | | | | Observations | |
| Eléments syntaxiques | | | | | | | | |
| 01 | horizontal_size_value | x | x | x | x | x | D | Voir le Tableau 8-7 |
| 02 | vertical_size_value | x | x | x | x | x | D | Voir le Tableau 8-7 |
| 03 | aspect_ratio_information | x | x | x | x | x | P | |
| 04 | frame_rate_code | x | x | x | x | x | D | Voir les Tableaux 8-7 et 8-6 |
| 05 | (Fréquence des pels) NOTE – Il ne s'agit pas d'un élément syntaxique | | | | | | D | Voir le Tableau 8-8; la fréquence des pels est un produit du nombre de pels par ligne, de lignes par image bi-trame et d'images bi-frames par seconde |
| 06 | bit_rate_value | x | x | x | x | x | D | Voir le Tableau 8-9 |
| 07 | vbv_buffer_size_value | x | x | x | x | x | D | Voir le Tableau 8-10 |
| 08 | constrained_parameters_flag | x | x | x | x | x | I | Prend la valeur '1' en mode MPEG-1 contraint Prend la valeur '0' en mode MPEG-2 |
| 09 | load_intra_quantiser_matrix | x | x | x | x | x | I | |
| 10 | intra_quantiser_matrix[64] | x | x | x | x | x | I | |
| 11 | load_non_intra_quantiser_matrix | x | x | x | x | x | I | |
| 12 | non_intra_quantiser_matrix[64] | x | x | x | x | x | I | |
| 13 | sequence_extension() | x | x | x | x | x | I | Toujours présent en mode MPEG-2 |
| 14 | sequence_display_extension() | x | x | x | x | x | P | |
| 15 | sequence_scalable_extension() | o | o | x | x | x | I | Voir le Tableau 8-11 pour le nombre maximal de couches échelonnables |
| 16 | user_data() | x | x | x | x | x | I | Le décodeur peut sauter ces données |

Tableau E.3 – Extension relative aux séquences

| # | Mode | | | | | | Type | |
|----|-------------------------------------|---|---|---|---|---|--------------|--|
| | Eléments syntaxiques | | | | | | Observations | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| 01 | profile_and_level_indication | x | x | x | x | x | D | Profil: choix d'une valeur dans une gamme de 8 valeurs Niveau: choix d'une valeur dans une gamme de 16 valeurs Bit d'échappement: choix d'une valeur dans une paire de valeurs |
| 02 | progressive_sequence | x | x | x | x | x | I | |
| 03 | chroma_format | x | x | x | x | x | I | Voir le Tableau 8-5 |
| 04 | horizontal_size_extension | x | x | x | x | x | D | Dépend du format des images d'entrée |
| 05 | vertical_size_extension | x | x | x | x | x | D | Dépend du format des images d'entrée |
| 06 | bit_rate_extension | x | x | x | x | x | D | Dépend du format des images d'entrée |
| 07 | vbv_buffer_size_extension | x | x | x | x | x | D | Dépend du format des images d'entrée |
| 08 | low_delay | x | x | x | x | x | I | |
| 09 | frame_rate_extension_n | x | x | x | x | x | I | Prend la valeur 0 pour tous les profils définis |
| 10 | frame_rate_extension_d | x | x | x | x | x | I | Prend la valeur 0 pour tous les profils définis |

Tableau E.4 – Eléments de l'extension d'affichage de séquences

| | Mode | | | | | | Type | Observations |
|----|---|---|---|---|---|---|------|---------------------------------------|
| | <div style="text-align: right; padding-right: 10px;"> SUPÉRIEUR SPATIAL SNR PRINCIPAL SIMPLE </div> | | | | | | | |
| # | Eléments syntaxiques | | | | | | | |
| 01 | video_format | x | x | x | x | x | P | |
| 02 | colour_description | x | x | x | x | x | P | Dépend du format des données d'entrée |
| 03 | colour primaries | x | x | x | x | x | P | |
| 04 | transfer_characteristics | x | x | x | x | x | P | |
| 05 | matrix_coefficients | x | x | x | x | x | P | |
| 06 | display_horizontal_size | x | x | x | x | x | P | Dépend du format des données d'entrée |
| 07 | display_vertical_size | x | x | x | x | x | P | Dépend du format des données d'entrée |

Tableau E.5 – Extension à l'échelonnabilité de séquences

| # | Mode | | | | | | Type | |
|----------------------|--|---|---|---|---|---|--------------|--|
| | <div style="text-align: right; margin-right: 20px;"> SUPÉRIEUR SPATIAL SNR PRINCIPAL SIMPLE </div> | | | | | | Observations | |
| Eléments syntaxiques | | | | | | | | |
| 01 | scalable_mode | o | o | x | x | x | I | Profil SNR: échelonnabilité SNR Profils de types spatial et supérieur: échelonnabilité SNR ou spatiale |
| 02 | layer_id | o | o | x | x | x | I | |
| | si (échelonnabilité spatiale) | | | | | | | |
| 03 | lower_layer_prediction_ horizontal_size | o | o | o | x | x | D | Voir le Tableau 8-8 pour la densité d'échantillonnage de luminance |
| 04 | lower_layer_prediction_ vertical_size | o | o | o | x | x | D | Voir le Tableau 8-8 pour la densité d'échantillonnage de luminance |
| 05 | horizontal_subsampling_ factor_m | o | o | o | x | x | I | |
| 06 | horizontal_subsampling_ factor_n | o | o | o | x | x | I | |
| 07 | vertical_subsampling_ factor_m | o | o | o | x | x | I | |
| 08 | vertical_subsampling_ factor_n | o | o | o | x | x | I | |
| | si (échelonnabilité temporelle) | | | | | | | |
| 09 | picture_mux_enable | o | o | o | o | o | I | |
| 10 | mux_to_progressive_sequence | o | o | o | o | o | I | |
| 11 | picture_mux_order | o | o | o | o | o | I | |
| 12 | picture_mux_factor | o | o | o | o | o | I | |

Tableau E.6 – En-tête de groupe d'images

| # | Mode | | | | | | Type | |
|----|----------------------|---|---|---|---|---|--------------|--|
| | SUPÉRIEUR | | | | | | | |
| | SPATIAL | | | | | | | |
| | SNR | | | | | | | |
| | PRINCIPAL | | | | | | | |
| | SIMPLE | | | | | | | |
| | Eléments syntaxiques | | | | | | | |
| | | | | | | | Observations | |
| 01 | time_code | x | x | x | x | x | I | |
| 02 | closed_gop | x | x | x | x | x | I | |
| 03 | broken_link | x | x | x | x | x | I | |

Tableau E.7 – En-tête d'image

| # | Mode | | | | | | Type | |
|----|---------------------------------------|---|---|---|---|---|--------------|---|
| | SUPÉRIEUR | | | | | | | |
| | SPATIAL | | | | | | | |
| | SNR | | | | | | | |
| | PRINCIPAL | | | | | | | |
| | SIMPLE | | | | | | | |
| | Eléments syntaxiques | | | | | | | |
| | | | | | | | Observations | |
| 01 | temporal_reference | x | x | x | x | x | I | |
| 02 | picture_coding_type | x | x | x | x | x | I | Profil simple: I et P à niveau principal I, P et B à bas niveau Profils principal, SNR, spatial et supérieur: I, P, B |
| 03 | vbv_delay | x | x | x | x | x | I | |
| 04 | full_pel_forward_vector | x | x | x | x | x | I | Prend la valeur '0' en mode MPEG-2 |
| 05 | forward_f_code | x | x | x | x | x | I | Prend la valeur '111' en mode MPEG-2 |
| 06 | full_pel_backward_vector | x | x | x | x | x | I | Prend la valeur '0' en mode MPEG-2 |
| 07 | backward_f_code | x | x | x | x | x | I | Prend la valeur '111' en mode MPEG-2 |
| 08 | extra_information_picture | x | x | x | x | x | I | |
| 09 | picture_coding_extension() | x | x | x | x | x | I | |
| 10 | quant_matrix_extension() | x | x | x | x | x | I | |
| 11 | picture_display_extension() | x | x | x | x | x | P | |
| 12 | picture_spatial_scalable_extension() | o | o | o | x | x | I | |
| 13 | picture_temporal_scalable_extension() | o | o | o | o | o | I | |

Tableau E.8 – Extension relative au codage des images

| # | Mode | | | | | | Type | |
|----|--|---|---|---|---|---|------|--|
| | <div style="text-align: right; margin-right: 10px;"> SUPÉRIEUR SPATIAL SNR PRINCIPAL SIMPLE Eléments syntaxiques </div> | | | | | | | Observations |
| 01 | f_code[0][0] (forward horizontal) | x | x | x | x | x | D | Niveau inférieur [1:7] Niveau principal [1:8] Niveaux supérieur-1440 et supérieur [1:9] |
| 02 | f_code[0][1] (forward vertical) | x | x | x | x | x | D | Niveau inférieur [1:4] Niveaux principal, supérieur-1440 et supérieur [1:5] |
| 03 | f_code[1][0] (backward horizontal) | x | x | x | x | x | D | Niveau inférieur [1:7] Niveau principal [1:8] Niveaux supérieur-1440 et supérieur [1:9] |
| 04 | f_code[1][1] (backward vertical) | x | x | x | x | x | D | Niveau inférieur [1:4] Niveaux principal, H-14 et supérieur [1:5] |
| 05 | intra_dc_precision | x | x | x | x | x | I | Profils simple, principal, SNR et spatial: [8:10] Profil supérieur: [8:11] |
| 06 | picture_structure | x | x | x | x | x | I | |
| 07 | top_field_first | x | x | x | x | x | I | |
| 08 | frame_pred_frame_dct | x | x | x | x | x | I | |
| 09 | concealment_motion_vectors | x | x | x | x | x | I | |
| 10 | q_scale_type | x | x | x | x | x | I | |
| 11 | intra_vlc_format | x | x | x | x | x | I | |
| 12 | alternate_scan | x | x | x | x | x | I | |
| 13 | repeat_first_field | x | x | x | x | x | I | |
| 14 | chroma_420_type | x | x | x | x | x | P | |
| 15 | progressive_frame | x | x | x | x | x | P | |
| 16 | composite_display_flag | x | x | x | x | x | P | |
| 17 | v_axis | x | x | x | x | x | P | |
| 18 | field_sequence | x | x | x | x | x | P | |
| 19 | sub_carrier | x | x | x | x | x | P | |
| 20 | burst_amplitude | x | x | x | x | x | P | |
| 21 | sub_carrier_phase | x | x | x | x | x | P | |

Tableau E.9 – Extension relative à une matrice de quantification

| Mode | | | | | | | Type | |
|------|---|-----------|---|---|---|---|--------------|--|
| | | SUPÉRIEUR | | | | | | |
| | | SPATIAL | | | | | | |
| | | SNR | | | | | | |
| | | PRINCIPAL | | | | | | |
| | | SIMPLE | | | | | | |
| # | Eléments syntaxiques | | | | | | Observations | |
| 01 | load_intra_quantiser_matrix | x | x | x | x | x | I | |
| 02 | intra_quantiser_matrix[64] | x | x | x | x | x | I | |
| 03 | load_non_intra_quantiser_matrix | x | x | x | x | x | I | |
| 04 | non_intra_quantiser_matrix[64] | x | x | x | x | x | I | |
| 05 | load_chroma_intra_quantiser_matrix | o | o | o | o | x | I | |
| 06 | chroma_intra_quantiser_matrix[64] | o | o | o | o | x | I | |
| 07 | load_chroma_non_intra_quantiser_matrix | o | o | o | o | x | I | |
| 08 | chroma_non_intra_quantiser_matrix[64] | o | o | o | o | x | I | |

Tableau E.10 – Extension relative à l'affichage des images

| Mode | | | | | | | Type | |
|------|---------------------------------------|-----------|---|---|---|---|--------------|---------------------------------------|
| | | SUPÉRIEUR | | | | | | |
| | | SPATIAL | | | | | | |
| | | SNR | | | | | | |
| | | PRINCIPAL | | | | | | |
| | | SIMPLE | | | | | | |
| # | Eléments syntaxiques | | | | | | Observations | |
| 01 | frame_centre_horizontal_offset | x | x | x | x | x | P | Dépend du format des données d'entrée |
| 02 | frame_centre_vertical_offset | x | x | x | x | x | P | Dépend du format des données d'entrée |

Tableau E.11 – Extension à l'échelonnabilité temporelle d'images

| # | Mode | | | | | | Type | |
|----|------------------------------------|---|---|---|---|---|--------------|--|
| | SUPÉRIEUR | | | | | | | |
| | SPATIAL | | | | | | | |
| | SNR | | | | | | | |
| | PRINCIPAL | | | | | | | |
| | SIMPLE | | | | | | | |
| | Eléments syntaxiques | | | | | | | |
| | | | | | | | Observations | |
| 01 | reference_select_code | o | o | o | o | o | I | |
| 02 | forward_temporal_reference | o | o | o | o | o | I | |
| 03 | backward_temporal_reference | o | o | o | o | o | I | |

Tableau E.12 – Extension à l'échelonnabilité spatiale d'images

| # | Mode | | | | | | Type | |
|----|---|---|---|---|---|---|--------------|---------------------------------------|
| | SUPÉRIEUR | | | | | | | |
| | SPATIAL | | | | | | | |
| | SNR | | | | | | | |
| | PRINCIPAL | | | | | | | |
| | SIMPLE | | | | | | | |
| | Eléments syntaxiques | | | | | | | |
| | | | | | | | Observations | |
| 01 | lower_layer_temporal_reference | o | o | o | x | x | I | |
| 02 | lower_layer_horizontal_offset | o | o | o | x | x | D | Dépend du format des données d'entrée |
| 03 | lower_layer_vertical_offset | o | o | o | x | x | D | Dépend du format des données d'entrée |
| 04 | spatial_temporal_weight_code_table_index | o | o | o | x | x | I | |
| 05 | lower_layer_progressive_frame | o | o | o | x | x | I | |
| 06 | lower_layer_deinterlaced_field_select | o | o | o | x | x | I | |

Tableau E.13 – Couche constituée de tranches

| # | Mode | | | | | | Type | |
|----------------------|--|---|---|---|---|---|--------------|---|
| | <div style="text-align: right; margin-right: 10px;"> SUPÉRIEUR SPATIAL SNR PRINCIPAL SIMPLE </div> | | | | | | Observations | |
| Eléments syntaxiques | | | | | | | | |
| 01 | slice_vertical_position_extension | x | x | x | x | x | D | Dépend du format des données d'entrée |
| 02 | priority_breakpoint | o | o | o | o | o | I | Uniquement nécessaire pour la subdivision des données |
| 03 | quantiser_scale_code | x | x | x | x | x | I | |
| 04 | intra_slice | x | x | x | x | x | I | |
| 05 | extra_information_slice | x | x | x | x | x | I | Le décodeur peut sauter ces données |
| 06 | macroblock() | x | x | x | x | x | I | |

Tableau E.14 – Couche constituée de macroblocs

| # | Mode | | | | | | Type | |
|----------------------|--|---|---|---|---|---|--------------|-------------------------------|
| | <div style="text-align: right; margin-right: 10px;"> SUPÉRIEUR SPATIAL SNR PRINCIPAL SIMPLE </div> | | | | | | Observations | |
| Eléments syntaxiques | | | | | | | | |
| 01 | macroblock_escape | x | x | x | x | x | I | |
| 02 | macroblock_address_increment | x | x | x | x | x | I | |
| 03 | macroblock_modes() | x | x | x | x | x | I | |
| 04 | quantiser_scale_code | x | x | x | x | x | I | |
| 05 | motion_vectors(0) | x | x | x | x | x | I | Vecteur de mouvement anticipé |
| 06 | motion_vectors(1) | o | x | x | x | x | I | Vecteur de mouvement différé |
| 07 | coded_block_pattern() | x | x | x | x | x | I | |
| 08 | block(i) | x | x | x | x | x | I | |

Tableau E.15 – Modes relatifs aux macroblocs

| # | Mode | | | | | | Type | |
|----|-------------------------------------|---|---|---|---|---|--------------|---|
| | Eléments syntaxiques | | | | | | Observations | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| 01 | macroblock_type | x | x | x | x | x | I | |
| 02 | spatial_temporal_weight_code | o | o | o | x | x | I | |
| 03 | frame_motion_type | x | x | x | x | x | I | 01: Prédiction à base de monotrames 10: Prédiction à base d'images bi-trames 11: Prédiction anticipée à double polarité |
| 04 | field_motion_type | x | x | x | x | x | I | 01: Prédiction à base de monotrames 10: Prédiction utilisant la compensation de mouvement 16 × 8 11: Prédiction anticipée à double polarité |
| 05 | dct_type | x | x | x | x | x | I | |

Tableau E.16 – Vecteurs de mouvement

| # | Mode | | | | | | Type | |
|----|-------------------------------------|---|---|---|---|---|--------------|--|
| | Eléments syntaxiques | | | | | | Observations | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| 01 | motion_vertical_field_select | x | x | x | x | x | I | |
| 02 | motion_vector() | x | x | x | x | x | I | |

Tableau E.17 – Vecteurs de mouvement

| # | Mode | | | | | | Type | |
|----|-------------------------------|--------|-----------|-----|---------|-----------|------|--------------|
| | Eléments syntaxiques | SIMPLE | PRINCIPAL | SNR | SPATIAL | SUPÉRIEUR | | Observations |
| 01 | motion_horizontal_code | x | x | x | x | x | I | |
| 02 | motion_horizontal_r | x | x | x | x | x | I | |
| 03 | dmv_horizontal | x | x | x | x | x | I | |
| 04 | motion_vertical_code | x | x | x | x | x | I | |
| 05 | motion_vertical_r | x | x | x | x | x | I | |
| 06 | dmv_vertical | x | x | x | x | x | I | |

Tableau E.18 – Structure de codage des blocs

| # | Mode | | | | | | Type | |
|----|--------------------------------|--------|-----------|-----|---------|-----------|------|--------------|
| | Eléments syntaxiques | SIMPLE | PRINCIPAL | SNR | SPATIAL | SUPÉRIEUR | | Observations |
| 01 | coded_block_pattern_420 | x | x | x | x | x | I | |
| 02 | coded_block_pattern_1 | o | o | o | o | x | I | 4:2:2 |
| 03 | coded_block_pattern_2 | o | o | o | o | o | I | 4:4:4 |

Tableau E.19 – Couche constituée de blocs

| # | Mode | | | | | | Type | |
|--------------|----------------------|--|---|---|---|---|------|---|
| | SUPÉRIEUR | | | | | | | |
| | SPATIAL | | | | | | | |
| | SNR | | | | | | | |
| | PRINCIPAL | | | | | | | |
| | SIMPLE | | | | | | | |
| | Eléments syntaxiques | | | | | | | |
| Observations | | | | | | | | |
| 01 | Coefficients DCT | | x | x | x | x | x | I |
| 02 | Fin de bloc | | x | x | x | x | x | I |

E.2 Combinaisons de couches autorisées

Les Tableaux E.20 à E.46 indiquent les limites paramétriques applicables à chaque couche d'un flux binaire ainsi que la combinaison profile_and_level_indication correspondante appropriée. Chaque tableau spécifie les limites d'un point de compatibilité unique dans la matrice de profil/niveau.

La notation suivante a été adoptée:

<abréviation de profil>@<abréviation de niveau>

Le Tableau E.20 donne la définition des abréviations.

Tableau E.20 – Abréviations des noms de profil et de niveau

| Profil | <abréviation de profil> | Niveau | <abréviation de niveau> |
|---------------------------------------|-------------------------|----------------|-------------------------|
| Simple | SP | Inférieur | LL |
| Principal | MP | Principal | ML |
| Echelonnable SNR | SNR | Supérieur-1440 | H-14 |
| Echelonnable spatialement | Spatial | Supérieur | HL |
| Supérieur | HP | | |
| Paramètres contraints ISO/CEI 11172-1 | | | ISO 11172 |

Tableau E.21 – Combinaison profil simple @ niveau principal

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 1 | 0 | Base | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |

Tableau E.22 – Combinaison profil principal @ niveau inférieur

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 1 | 0 | Base | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |

Tableau E.23 – Combinaison profil principal @ niveau principal

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 1 | 0 | Base | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |

Tableau E.24 – Combinaison profil principal @ niveau supérieur-1440

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 1 | 0 | Base | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | MP@H-14 |

Tableau E.25 – Combinaison profil principal @ niveau supérieur

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 1 | 0 | Base | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | MP@HL |

Tableau E.26 – Combinaison profil SNR @ niveau inférieur

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 2 | 0 | Base | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | SNR | 352/288/30 | 2 534 400 | 4 | 475 136 | SNR@LL |
| 2 | 0 | Base | 352/288/30 | 3 041 280 | 3 | 360 448 | SP@ML |
| | 1 | SNR | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| 2 | 0 | Base | 352/288/30 | 3 041 280 | 3 | 360 448 | MP@LL |
| | 1 | SNR | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |

Tableau E.27 – Combinaison profil SNR @ niveau principal

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 2 | 0 | Base | 720/576/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | SNR | 720/576/30 | 2 534 400 | 15 | 1 835 008 | SNR@ML |
| 2 | 0 | Base | 720/576/30 | 10 368 000 | 10 | 1 212 416 | SP@ML |
| | 1 | SNR | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| 2 | 0 | Base | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | SNR | 352/288/30 | 3 041 280 | 15 | 1 835 008 | SNR@ML |
| 2 | 0 | Base | 720/576/30 | 10 368 000 | 10 | 1 212 416 | MP@ML |
| | 1 | SNR | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |

Tableau E.28 – Combinaison profil spatial @ niveau supérieur-1440 (couche de base + SNR)

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 2 | 0 | Base | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | SNR | 352/288/30 | 2 534 400 | 60 | 7 340 032 | Spt@H-14 |
| 2 | 0 | Base | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | SNR | 720/576/30 | 10 368 000 | 60 | 7 340 032 | Spt@H-14 |
| 2 | 0 | Base | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | SNR | 352/288/30 | 3 041 280 | 60 | 7 340 032 | Spt@H-14 |
| 2 | 0 | Base | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | SNR | 720/576/30 | 10 368 000 | 60 | 7 340 032 | Spt@H-14 |
| 2 | 0 | Base | 1440/1152/60 | 47 001 600 | 40 | 4 882 432 | MP@H-14 |
| | 1 | SNR | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |

**Tableau E.29 – Combinaison profil spatial @ niveau supérieur-1440
(couche de base + spatial)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 2 | 0 | Base | 768/576/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 2 | 0 | Base | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 2 | 0 | Base | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 2 | 0 | Base | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 2 | 0 | Base | 1440/1152/60 | 47 001 600 | 40 | 4 882 432 | MP@H-14 |
| | 1 | Spatial | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |

**Tableau E.30 – Combinaison profil spatial @ niveau supérieur-1440
(couche de base + SNR + spatial)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | SNR | 352/288/30 | 2 534 400 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 352/288/30 | 3 041 280 | 3 | 360 448 | SP@ML |
| | 1 | SNR | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 352/288/30 | 3 041 280 | 3 | 360 448 | MP@LL |
| | 1 | SNR | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 720/576/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | SNR | 720/576/30 | 2 534 400 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 720/576/30 | 10 368 000 | 10 | 1 212 416 | SP@ML |
| | 1 | SNR | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | SNR | 352/288/30 | 3 041 280 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 720/576/30 | 10 368 000 | 10 | 1 212 416 | MP@ML |
| | 1 | SNR | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 1440/1152/60 | 10 368 000 | 15 | 1 835 008 | MP@H-14 |
| | 1 | SNR | 1440/1152/60 | 10 368 000 | 40 | 4 882 432 | Spt@H-14 |
| | 2 | Spatial | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |

**Tableau E.31 – Combinaison profil spatial @ niveau supérieur-1440
(couche de base + spatial + SNR)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 768/576/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 1440/1152/30 | 47 001 600 | 40 | 4 882 432 | Spt@H-14 |
| | 2 | SNR | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 1440/1152/30 | 47 001 600 | 40 | 4 882 432 | Spt@H-14 |
| | 2 | SNR | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 1440/1152/30 | 47 001 600 | 40 | 4 882 432 | Spt@H-14 |
| | 2 | SNR | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 1440/1152/30 | 47 001 600 | 40 | 4 882 432 | Spt@H-14 |
| | 2 | SNR | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| 3 | 0 | Base | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@H-14 |
| | 1 | Spatial | 1440/1152/60 | 47 001 600 | 40 | 4 882 432 | Spt@H-14 |
| | 2 | SNR | 1440/1152/30 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |

**Tableau E.32 – Combinaison profil supérieur @ niveau principal
(couche de base)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 1 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| 1 | 0 | Base | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |

**Tableau E.33 – Combinaison profil supérieur @ niveau principal
(couche de base + SNR)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 10 368 000 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | SNR | 4:2:2 | 352/288/30 | 3 041 280 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 10 368 000 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 15 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:2 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |

**Tableau E.34 – Combinaison profil supérieur @ niveau principal
(couche de base + spatial)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SP@ML |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SP@ML |
| | 1 | Spatial | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |

**Tableau E.35 – Combinaison profil supérieur @ niveau principal
(couche de base + SNR + spatial)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | SP@ML |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | SP@ML |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | MP@LL |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | MP@LL |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |

**Tableau E.36 – Combinaison profil supérieur @ niveau principal
(couche de base + spatial + SNR)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 15 | 1 835 008 | HP@ML |
| | 2 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 2 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:2 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 2 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SP@ML |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 15 | 1 835 008 | HP@ML |
| | 2 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SP@ML |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 2 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SP@ML |
| | 1 | Spatial | 4:2:2 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 2 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 15 | 1 835 008 | HP@ML |
| | 2 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 2 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:2 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 2 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |

**Tableau E.37 – Combinaison profil supérieur @ niveau supérieur-1440
(couche de base)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 1 | 0 | Base | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 1 | 0 | Base | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |

**Tableau E.38 – Combinaison profil supérieur @ niveau supérieur-1440
(couche de base + SNR)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 10 368 000 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | SNR | 4:2:2 | 352/288/30 | 3 041 280 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 10 368 000 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | MP@H-14 |
| | 1 | SNR | 4:2:0 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | MP@H-14 |
| | 1 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 14 745 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 1440/1152/60 | 62 668 800 | 60 | 7 340 032 | HP@H-14 |
| | 1 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | HP@H-14 |
| | 1 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:2 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | HP@H-14 |
| | 1 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |

**Tableau E.39 – Combinaison profil supérieur @ niveau supérieur-1440
(couche de base + spatial)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | MP@H-14 |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | MP@H-14 |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 2 | 0 | Base | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |

**Tableau E.40 – Combinaison profil supérieur @ niveau supérieur-1440
(couche de base + SNR + spatial)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | SP@ML |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | SP@ML |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 10 | 1 212 416 | SP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 10 | 1 212 416 | SP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | MP@LL |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | MP@LL |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 10 | 1 212 416 | MP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 10 | 1 212 416 | MP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 10 368 000 | 20 | 2 441 216 | HP@ML |
| | 2 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 15 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 2 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| | 2 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:2 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| | 2 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |

**Tableau E.41 – Combinaison profil supérieur @ niveau supérieur-1440
(couche de base + spatial + SNR)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/ 1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|--------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |

Tableau E.41 – Combinaison profil supérieur @ niveau supérieur-1440
(couche de base + spatial + SNR) (*fin*)

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| 3 | 0 | Base | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |

Tableau E.42 – Combinaison profil supérieur @ niveau supérieur
(couche de base)

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 1 | 0 | Base | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 1 | 0 | Base | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |

**Tableau E.43 – Combinaison profil supérieur @ niveau supérieur
(couche de base + SNR)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 10 368 000 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | SNR | 4:2:2 | 352/288/30 | 3 041 280 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 10 368 000 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | MP@H-14 |
| | 1 | SNR | 4:2:0 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | MP@H-14 |
| | 1 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | MP@HL |
| | 1 | SNR | 4:2:0 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | MP@HL |
| | 1 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 14 745 600 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| | 1 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| | 1 | SNR | 4:2:2 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| | 1 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 1920/1152/60 | 83 558 400 | 80 | 9 781 248 | HP@HL |
| | 1 | SNR | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 1 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:2 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 1 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |

**Tableau E.44 – Combinaison profil supérieur @ niveau supérieur
(couche de base + spatial)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | MP@H-14 |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | MP@H-14 |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | HP@H-14 |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | HP@H-14 |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 2 | 0 | Base | 4:2:2 | 960/576/30 | 14 745 600 | 25 | 3 047 424 | HP@H-14 |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |

**Tableau E.45 – Combinaison profil supérieur @ niveau supérieur
(couche de base + SNR + spatial)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | SP@ML |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | SP@ML |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 10 | 1 212 416 | SP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 10 | 1 212 416 | SP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 10 368 000 | 20 | 2 441 216 | HP@ML |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | MP@LL |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 3 | 360 448 | MP@LL |
| | 1 | SNR | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | SNR@LL |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 10 | 1 212 416 | MP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 10 | 1 212 416 | MP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SNR@ML |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 10 368 000 | 20 | 2 441 216 | HP@ML |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 20 | 2 441 216 | MP@H-14 |
| | 1 | SNR | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | Spt@H-14 |
| | 2 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 20 | 2 441 216 | MP@H-14 |
| | 1 | SNR | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | Spt@H-14 |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 960/576/30 | 14 745 600 | 20 | 2 441 216 | MP@H-14 |
| | 1 | SNR | 4:2:2 | 960/576/30 | 14 745 600 | 25 | 3 047 424 | HP@H-14 |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |

**Tableau E.45 – Combinaison profil supérieur @ niveau supérieur
(couche de base + SNR + spatial) (fin)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 15 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 2 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 14 745 600 | 15 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:2 | 720/576/30 | 11 059 200 | 15 | 1 835 008 | HP@ML |
| | 1 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 20 | 2 441 216 | HP@H-14 |
| | 1 | SNR | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | HP@H-14 |
| | 2 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 20 | 2 441 216 | HP@H-14 |
| | 1 | SNR | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | HP@H-14 |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 960/576/30 | 14 745 600 | 20 | 2 441 216 | HP@H-14 |
| | 1 | SNR | 4:2:2 | 960/576/30 | 14 745 600 | 25 | 3 047 424 | HP@H-14 |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:2 | 960/576/30 | 14 745 600 | 20 | 2 441 216 | HP@H-14 |
| | 1 | SNR | 4:2:2 | 960/576/30 | 14 745 600 | 25 | 3 047 424 | HP@H-14 |
| | 2 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |

**Tableau E.46 – Combinaison profil supérieur @ niveau supérieur
(couche de base + spatial + SNR)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/ 1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|--------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 2 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 2 | SNR | 4:2:2 | 720/576/30 | 14 745 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| | 2 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 2 534 400 | 1,856 | 327 680 | ISO 11172 |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |

**Tableau E.46 – Combinaison profil supérieur @ niveau supérieur
(couche de base + spatial + SNR) (suite)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | SP@ML |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 2 | SNR | 4:2:0 | 720/576/30 | 14 745 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 720/576/30 | 14 745 600 | 20 | 2 441 216 | HP@ML |
| | 2 | SNR | 4:2:2 | 720/576/30 | 14 745 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:2 | 720/576/30 | 11 059 200 | 20 | 2 441 216 | HP@ML |
| | 2 | SNR | 4:2:2 | 720/576/30 | 11 059 200 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |

Tableau E.46 – Combinaison profil supérieur @ niveau supérieur
(couche de base + spatial + SNR) (suite)

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 352/288/30 | 3 041 280 | 4 | 475 136 | MP@LL |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 47 001 600 | 60 | 7 340 032 | Spt@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:0 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1440/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:2 | 1440/1152/60 | 47 001 600 | 80 | 9 781 248 | HP@H-14 |
| | 2 | SNR | 4:2:2 | 1440/1152/60 | 47 001 600 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 720/576/30 | 10 368 000 | 15 | 1 835 008 | MP@ML |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | HP@H-14 |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 83 558 400 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:0 | 1920/1152/60 | 83 558 400 | 100 | 12 222 464 | HP@HL |

**Tableau E.46 – Combinaison profil supérieur @ niveau supérieur
(couche de base + spatial + SNR) (fin)**

| Nombre de couches | Fanion identificateur de couche | Mode échelonnable | Format de chrominance | Densité maximale d'échantillons (H/V/F) | Fréquence maximale des échantillons de luminance | Débit binaire total maximal/1000000 | Capacité maximale de mémoire du VBV | Combinaison profil et niveau |
|-------------------|---------------------------------|-------------------|-----------------------|---|--|-------------------------------------|-------------------------------------|------------------------------|
| 3 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | HP@H-14 |
| | 1 | Spatial | 4:2:0 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:0 | 960/576/30 | 19 660 800 | 25 | 3 047 424 | HP@H-14 |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |
| 3 | 0 | Base | 4:2:2 | 960/576/30 | 14 745 600 | 25 | 3 047 424 | HP@H-14 |
| | 1 | Spatial | 4:2:2 | 1920/1152/60 | 62 668 800 | 80 | 9 781 248 | HP@HL |
| | 2 | SNR | 4:2:2 | 1920/1152/60 | 62 668 800 | 100 | 12 222 464 | HP@HL |

Annexe F

Bibliographie

(Cette annexe ne fait pas partie intégrante de la présente Recommandation | Norme internationale)

- [1] NETRAVALI (A.N.) et HASKELL (B.G.): Digital Pictures, representation and compression, *Plenum Press*, 1988.
- [2] LEGALL (D.): MPEG: A Video Compression Standard for Multimedia Applications, *Trans. ACM*, avril 1991.
- [3] LOEFFLER (C.), LIGTENBERG (A.), MOSCHYTZ (G.S.): Practical fast 1-D DCT algorithms with 11 multiplications, *Proceedings IEEE ICASSP-89*, Vol. 2, p. 988-991, février 1989.
- [4] Voir les références normatives pour la Recommandation UIT-R BT.601.
- [5] Voir les références normatives pour la Publication 461 de la CEI.
- [6] Voir les références normatives pour la Recommandation UIT-T H.261.
- [7] Voir les références normatives pour IEEE Standard Specification, p. 1180-1990.
- [8] Rec. UIT-T T.81 (antérieurement JPEG) | ISO/CEI 10918-1.
- [9] VISCITO (E.) et GONZALES (C.): A Video Compression Algorithm with Adaptive Bit Allocation and Quantization, *Proc. SPIE Visual Communications and Image Proc. 91*, Vol. 1605 205, Boston, MA, 10-15 novembre 1991.
- [10] PURI (A.) et ARAVIND (R.): Motion Compensated Video Coding with Adaptive Perceptual Quantization, *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 1, p. 351, décembre 1991.
- [11] GONZALES (C.) et VISCITO (E.): Flexibly scalable digital video coding, *Image Communications*, Vol. 5, n° 1-2, février 1993.
- [12] JOHNSON (A.W.), SIKORA (T.) et TAN (T.K.): Filters for Drift Reduction in Frequency Scalable Video Coding Schemes, *destiné à être publié dans Electronic Letters*.
- [13] MOKRY (R.) et ANASTASSIOU (D.): Minimal Error Drift in Frequency Scalability for Motion-Compensated DCT Coding, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 4, n° 4, p. 392-407, août 1994.
- [14] NGAN (K.N.), ARNOLD (J.), SIKORA (T.), TAN (T.K.) et JOHNSON (A.W.): Frequency Scalability Experiments for MPEG-2 Standard, *Asia-Pacific Conference on Communications*, Corée, août 1993.
- [15] SIKORA (T.), TAN (T.K.) et NGAN (K.N.): A Performance Comparison of Frequency Domain Pyramid Scalable Coding Schemes Within the MPEG Framework, *Proc. PCS, Picture Coding Symposium*, p. 16.1-16.2, mars 1993, Lausanne, Suisse.
- [16] IWAHASHI (M.): Motion Compensation Technique for 2:1 Scaled-down Moving Pictures, 8-14, *Picture Coding Symposium 93*.
- [17] SIKORA (T.) et PANG, (K.): Experiments with Optimal Block-Overlapping Filters for Cell Loss Concealment in Packet Video, *Proc. IEEE Visual Signal Processing and Communications Workshop*, p. 247-250, 21-22 septembre 1993, Melbourne.
- [18] PURI (A.): Video Coding Using the MPEG-2 Compression Standard, *Proc. SPIE Visual Communications and Image Proc. 93*, Boston, MA, novembre 1993.
- [19] PURI (A.) et WONG (A.): Spatial Domain Resolution Scalable Video Coding, *Proc. SPIE Visual Communications and Image Proc. 93*, Boston, MA, novembre 1993.