

## CHAPITRE 1

### REVUE DE LA LITTÉRATURE

Dans ce chapitre, nous faisons état de la littérature sur les approches de détection de la détérioration visuelle. Il ne s'agit pas ici d'une simple énumération, mais bien, d'une suite logique d'ouvrages permettant de comprendre, l'introduction et l'évolution de plusieurs concepts fondamentaux, sur lesquels reposent notre contribution.

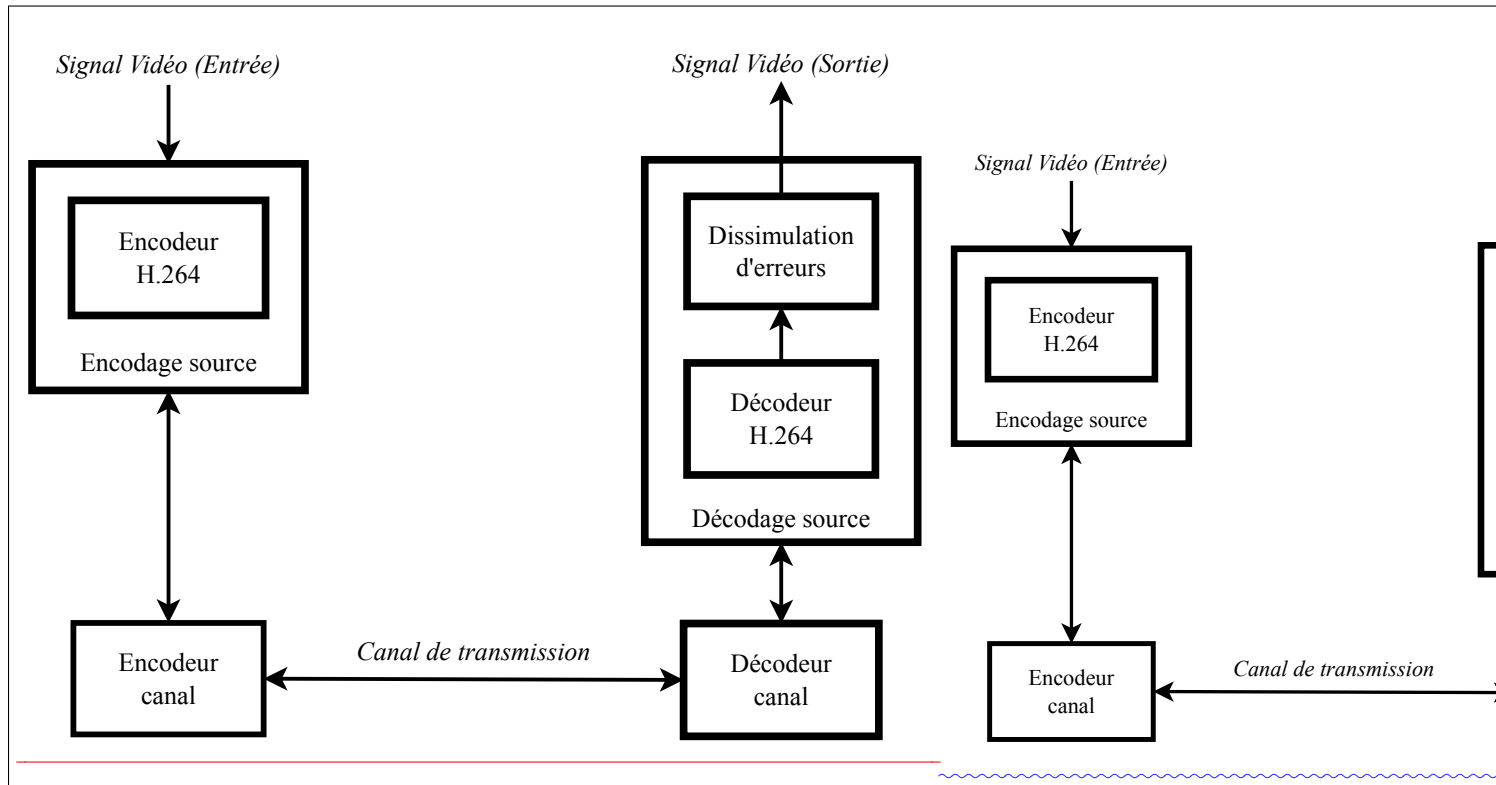
La détection de la détérioration visuelle ~~sert à identifier la dégradation visuelle engendrée suite au décodage~~ est possible uniquement si le contenu de paquets corrompus ~~. Pour décoder ces paquets~~ peut être décodé. Pour ce faire, le décodeur du canal de transmissions doit interagir avec le décodeur vidéo (source), ~~afin d'identifier et de rendre disponible les paquets corrompus~~. Ce type d'interaction porte le nom de *Joint Source Channel Decoding*. Notre prologue, section 1.1, expose les origines du *Joint Source Channel Decoding*. La section 1.2 offre un compte rendu des notions du *Joint Source Channel Decoding* qui s'appliquent spécifiquement au transport de séquences vidéo. Les améliorations apportées à une de ces notions, l'analyse syntaxique, sont présentées à la section 1.3. Finalement, à la section 1.4, les approches combinant l'analyse syntaxique à la détection de la détérioration visuelle dans le domaine des pixels sont décrites.

#### 1.1 Prologue

Dans son ouvrage *A Mathematical Theory of Communication*, Shannon (1948) explique sous l'allégorie ingénieuse du télégraphe, la notion même du *Joint Source Channel Decoding*. L'usage d'un langage redondant, tel l'anglais, comme encodage pour le télégraphe a l'avantage d'être résistant aux erreurs du canal de transmissions. Le contexte du message permet la reconstruction des mots, même si plusieurs lettres de ceux-ci sont bruitées (Shannon, 1948, p. 24).

En 1979, Modestino et Daut sont les premières personnes à proposer d'utiliser le *Joint Source Channel Decoding* pour améliorer le transport d'images numériques. Déjà à cette époque,

Modestino et Daut déplorent le gaspillage de bande passante introduit par les stratégies de résilience aux erreurs. Un gaspillage qui, 30 ans plus tard, est toujours présent sur les réseaux sans fil (Duhamel et Kieffer, 2010, p.1).



**Figure 1.1** Diagramme à blocs des composantes liées au transport de séquences vidéo.  
Adaptée de Wang et Zhu (1998, p. 976)

~~Avant de poursuivre, commençons par décrire les composantes liées au Joint Source Channel Decoding.~~

~~La résume les étapes et interactions du Joint Source Channel Decoding. Tout d'abord, l'encodeur H.264 retire la redondance d'un signal vidéo. Par la suite, l'encodeur du canal va empaqueter le contenu et insérer les mécanismes de résilience à l'erreur nécessaires selon le canal de transmission. Notons l'ajout de donnée redondante comme un des mécanismes de résilience à l'erreur. Dès la réception de paquets chez le destinataire, le décodeur du canal dépaquette ceux-ci et valide que leur contenu soit intact. À l'aide des données dépaquetées, le décodeur H.264 reconstruit le signal vidéo. Cependant, si des données sont manquantes, le décodeur~~

~~H.264 fait appel à un algorithme de dissimulation d'erreurs pour permettre d'estimer les données manquantes.~~

## 1.2 Établir le lien entre le décodeur vidéo et celui du canal de transmission

Les techniques de résilience aux erreurs sur des réseaux non fiables ont suscité beaucoup d'intérêt dans les années 80 et 90. En 1998, dans leur revue sur les techniques de contrôle et de dissimulation d'erreurs lors du transport de séquences vidéo (Wang et Zhu, 1998), Wang et Zhu récapitulent, entre autres, l'état de l'art, de l'époque, des stratégies de résilience aux erreurs. Wang et Zhu présentent une multitude d'approches. Parmi celles-ci, ils identifient un type d'approches qu'ils qualifient d'approches de dissimulation par post-traitement effectué par le décodeur (Wang et Zhu, 1998, Chap. 5). Ce type résume, de façon exacte, les approches que nous présentons dans cet ouvrage. Leurs composantes sont illustrées à la figure 1.1. Parmi les approches de dissimulation présentées en 1998, on remarque : *Motion-Compensated Temporal Prediction* (Ghanbari, 1993), *Maximally Smooth Recovery* (Wang et al., 1993), *Projection onto Convex Sets (POCS)* (Sun et Kwok, 1995), *Spatial- and Frequency-Domain Interpolation* (Hemami et Meng, 1995) et (Sun et al., 1992).

Toutes ces approches de post-traitement, effectuées par le décodeur, présentées par Wang et Zhu (1998), supposent que les blocs endommagés sont connus. Pratiquement, cette identification est accomplie en ignorant les paquets corrompus, ce qui force le décodeur à dissimuler tous les blocs que contiennent ces derniers même si un seul bloc est endommagé. Pour reprendre l'exemple du télégraphe, ceci revient à rejeter l'ensemble des mots d'une phrase qui possède une lettre en erreur. Cela complique grandement la compréhension du message. Ce nivellement par le bas, quoique plus simple, est un exemple du gaspillage identifié par Modestino et Daut, où le décodeur de la source ne communique pas avec le décodeur du canal de transport.

Le *Joint Source Channel Decoding* (Duhamel et Kieffer, 2010) cherche, à l'aide de la logique de la source, à identifier précisément, où se trouve l'erreur dans un ensemble de bits corrompus. Par exemple, pour l'encodage MPEG-4 Visual, Talluri propose des règles permettant de vérifier

la validité de : la portée des vecteurs de mouvement, des codes entropiques décodés, la portée des valeurs de coefficient DCT et du nombre de coefficients DCT afin qu'il ne dépasse pas le nombre permis par la norme (Talluri, 1998). Une fois l'erreur identifiée, il existe deux types de solutions possibles : la reconstruction et la dissimulation.

D'une part, la reconstruction vise à retrouver les valeurs originales des bits corrompus à l'aide de modèles mathématiques complexes, comme présentée par Duhamel et Kieffer (2010).

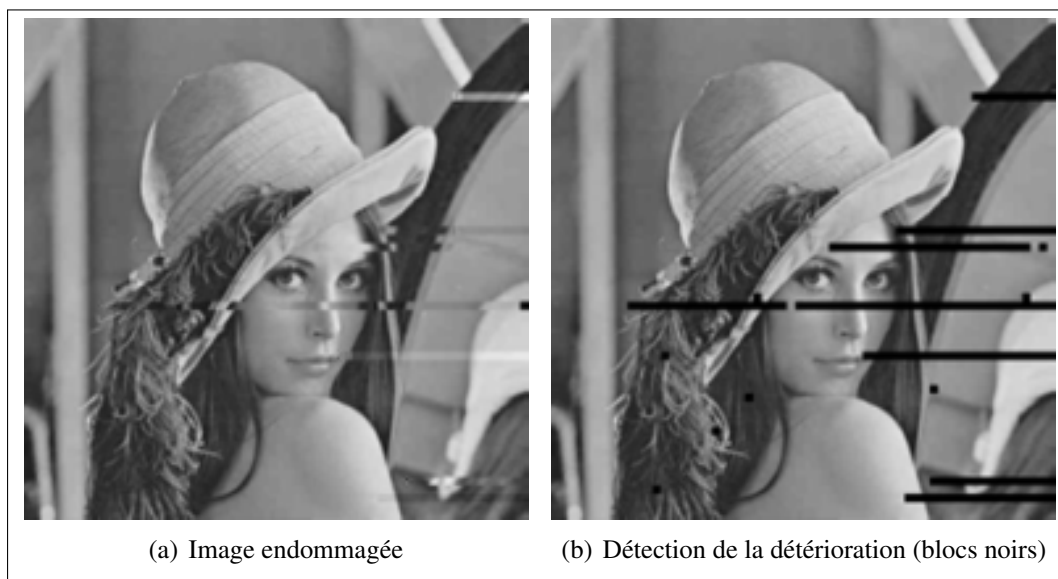
D'autre part, la dissimulation tente de décoder les paquets corrompus, et par la suite, identifier et dissimuler la détérioration visuelle engendrée par cette opération. L'article de Ye et al., parue en mai 2003, présente un algorithme destiné au décodage d'images JPEG corrompues. Ce dernier est capable d'identifier et classifie des blocs erronés, selon leurs caractéristiques dans le domaine des pixels (Ye et al., 2003).

Ye et al. définissent quatre mesures permettant d'identifier les blocs erronés. La première est la validité de la valeur assignée à un pixel. Il s'agit ici de déterminer si un pixel est dans un intervalle de données plausibles. Par exemple : pour les valeurs d'un pixel non signées de niveaux de gris assignées sur huit bits, les valeurs possibles sont de 0 à 255. La deuxième mesure est l'hétérogénéité des bordures de blocs horizontales mesurée à l'aide d'un filtre Sobel. La troisième est l'hétérogénéité des bordures de blocs verticales, encore une fois mesurée à l'aide d'un filtre Sobel. Dans d'autres oeuvres, l'hétérogénéité en bordure de blocs est aussi connue sous le nom d'effet de bloc. Le filtre Sobel est défini par les opérateurs suivants (horizontal et vertical respectivement) :

$$S_h = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad S_v = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \quad (1.1)$$

La dernière mesure, présentée par Ye et al., évalue la continuité des bordures propres au contenu de l'image qui traversent un bloc. Par la suite, ces mesures sont comparées à des seuils, afin de déterminer si le bloc évalué est erroné. Le résultat de la détection d'erreurs à l'aide de

ces mesures est présenté à la figure 1.2, où les blocs noirs de l'image (b) représentent les erreurs détectées. On y constate de faux positifs, surtout dans le chapeau et les cheveux de Lena Söderberg.

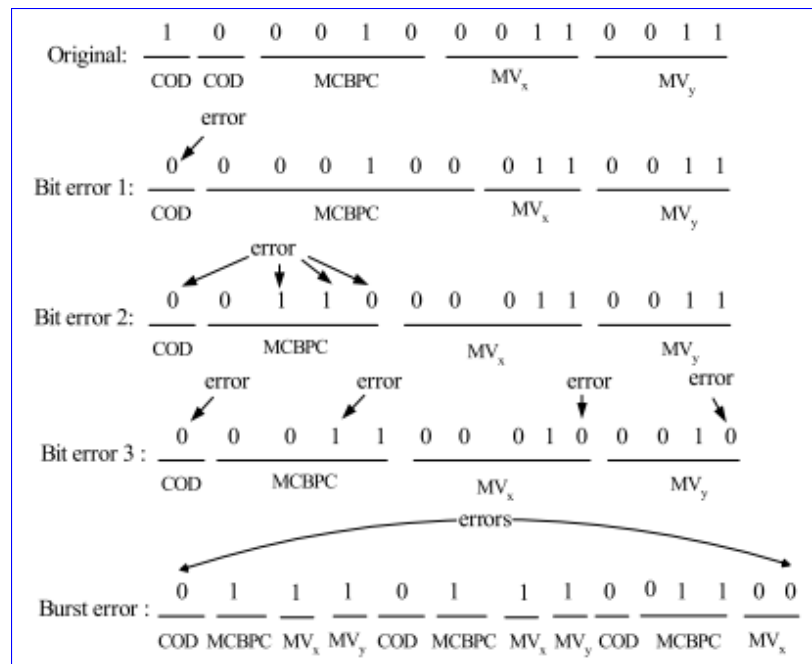


**Figure 1.2** Exemple du résultat de la détection d'erreurs avec les mesures de Ye et al.  
Tirée de Ye et al. (2003, p. 371)

### 1.3 L'amélioration des approches d'analyse syntaxique d'encodage vidéo

Dans le but d'améliorer l'efficacité des règles de validation de syntaxe présentées par Talluri (1998), Yan et Wing proposent une nouvelle règle de validation qui augmente considérablement l'efficacité de ce genre d'approche. Cette règle permet de valider, entre deux marqueurs de synchronisation, que le nombre de COD

soit égal au nombre de blocs (Yan et Wing, 2003). Expliquons le raisonnement derrière cette approche. Tout d'abord, les marqueurs de synchronisation sont des séquences binaires uniques qui servent de point de repère pour le décodeur. Le COD est un bit positionné en début de macrobloc, qui identifie si ce dernier est encodé ou pas. Le nombre de macroblocs contenus dans une tranche est disponible dans les paramètres d'encodages. S'il y a eu désynchronisation, ces deux nombres risquent d'être différents, comme illustré à la figure 1.3.



**Figure 1.3** [L'impact de la corruption de bits, lors du décodage.](#)  
[Adaptée de Yan et Wing \(2003, p. 1425\)](#)

Toujours dans le même ordre d'idée, en 2006, Superiori et Nemethova proposent un valideur de syntaxe H.264. Ce valideur identifie des séquences de codes binaires invalides engendrées par la désynchronisation des codes entropiques, suite à une erreur (Superiori et Nemethova, 2006).

**[SC]:** Ok mais quels résultats sont obtenus par leur approche ?

Ce valideur est exhaustif et définit des restrictions sur les valeurs d'un grand nombre de paramètres de la norme.

Superiori et Nemethova proposent trois catégories d'erreurs de décodage selon les caractéristiques de l'erreur (Superiori et Nemethova, 2006) :

- **Code numérique invalide** : Il s'agit ici d'un code qui n'est pas dans la table de codes numériques associés à ce paramètre.
- **Code hors de portée** : Lorsqu'une valeur décodée est à l'extérieur de l'ensemble des valeurs valides pour le paramètre en question.
- **Erreur contextuelle** : Cette erreur survient si le décodeur doit effectuer une opération invalide.

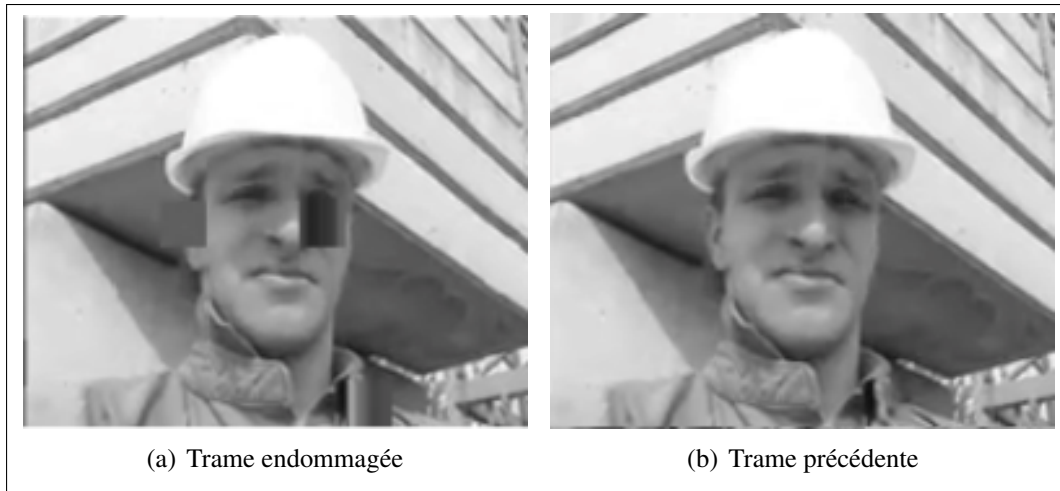
#### 1.4 La combinaison de l'analyse syntaxique et de la détection de la détérioration visuelle dans le domaine des pixels

En 2007, Superiori et al. combinent leur approche de validation de syntaxe à un algorithme d'identification de la détérioration visuelle inspiré de Ye et al. (2003), auquel un système de vote a été ajouté, afin ~~d'améliorer~~ de raffiner le résultat de la détection (Superiori et al., 2007). Dans son mémoire Ikuno, explique en détail l'algorithme proposé (Ikuno, 2007).

La détection de la détérioration visuelle est réalisée dans le domaine des pixels. Pour isoler l'erreur, Superiori et al. utilisent le différentiel de la trame à évaluer par rapport à celle qui la précède (c.-à-d. la différence entre ces deux trames). La figure 1.4 montre une trame endommagée ainsi que sa trame précédente. Le différentiel entre ces deux images est calculé en effectuant une différence entre les pixels correspondants (même position spatiale). Le résultat du différentiel des images de la figure 1.4 est présenté à la figure 1.5(a). ~~illustré aux figures 1.4 et 1.5(a).~~

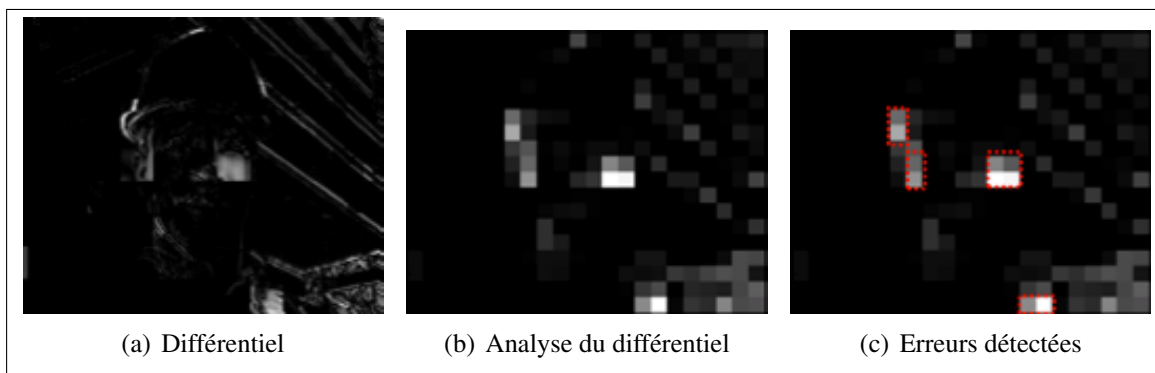
**[SC]:** Les bonnes figures 1.4 et 1.5 ? Toujours présenter l'information des figures. Le lecteur ne va pas comprendre ce que tu veux qu'il comprenne. Ici 1,4 n'est pas présenté

Par la suite, les valeurs du différentiel sont regroupées en blocs  $8 \times 8$  et la moyenne est assignée comme étant représentative de l'énergie du bloc (Ikuno, 2007, p. 27-28), comme illustré à la



**Figure 1.4** Trames utilisées par Ikuno pour démontrer l'algorithme de détection de la détérioration visuelle.  
Tirée de Ikuno (2007, p. 25)

figure 1.5(b). Les erreurs détectées à la figure 1.5(c) sont obtenues à l'aide d'un seuil. Si la valeur de l'énergie d'un bloc est supérieure à ce seuil, le bloc est considéré comme erroné.

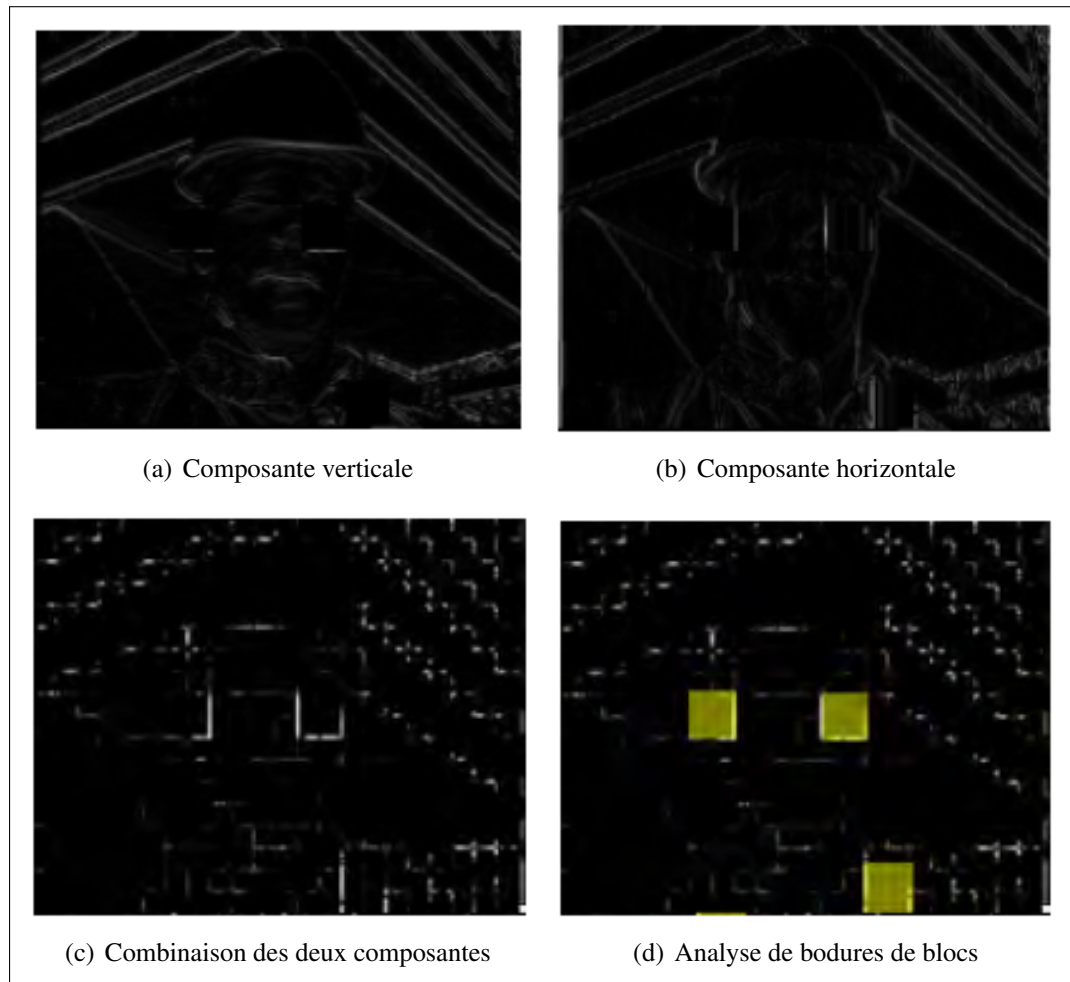


**Figure 1.5** Exemple de l'analyse et de la detection d'erreurs.  
Tirée de Ikuno (2007, p. 25)

De plus, la détection d'effets de bloc en bordure de blocs est effectuée. Pour trouver les bordures à l'intérieur des trames, les composantes horizontales 1.6(a) et verticales 1.6(b), d'un filtre de Haar (Haar, 1911), sont combinées et seulement les valeurs en bordure de bloc sont conservées 1.6(c). L'analyse des effets de bloc est accomplie pour les blocs  $8 \times 8$  et pour



les macroblocs  $16 \times 16$ . Ceux-ci sont comparés, à des seuils, afin de savoir s'il y a présence d'erreurs.



**Figure 1.6** Composantes et analyse du filtre de Haar.  
Tirée de Ikuno (2007, p. 26)

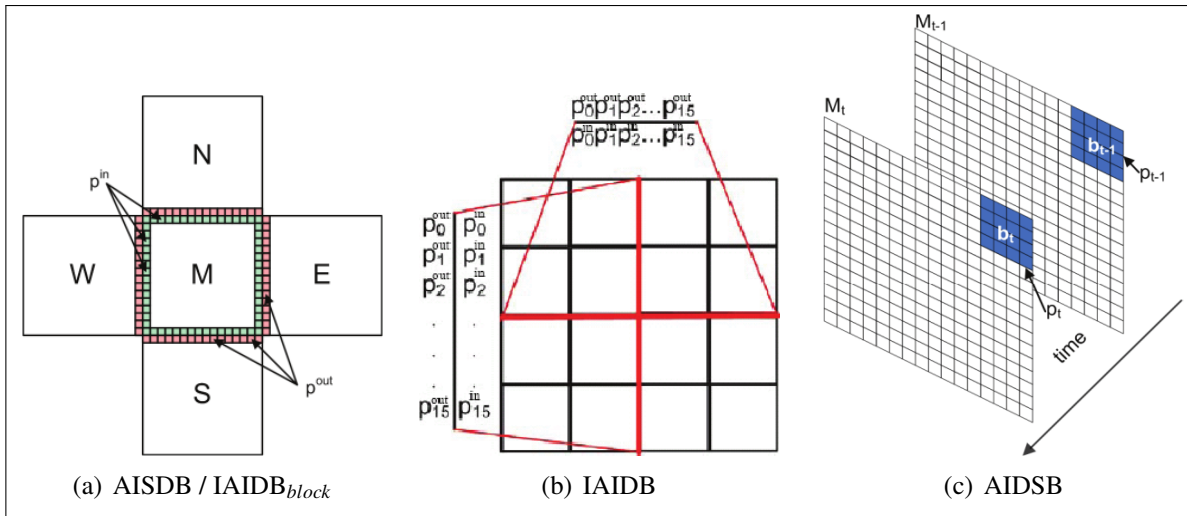
Les mesures d'énergie et d'effets de bloc sont interprétées par un système de vote qui détermine où se trouvent les blocs erronés à l'intérieur de la trame évaluée.

L'année suivante, Farrugia et Debono reprennent les travaux de Superiori et al. (2007) et retirent les seuils et le système de vote. Ceux-ci sont remplacés par une machine à vecteurs de support (Cortes et Vapnik, 1995). Cette dernière est une méthode d'apprentissage supervisée qui, une fois entraînée sur un ensemble de données, est capable de classer d'autres données. Pour ce

genre d'algorithme, le temps d'entraînement est long, mais la classification est très rapide, ce qui en fait un bon choix pour des considérations temps réel (Farrugia et Debono, 2008).

Le vecteur d'entrées (*feature vector*) défini par Farrugia et Debono possède huit composantes :

1. AIDB
2. La moyenne du  $IAIDB_{block}$
3. L'écart type du  $IAIDB_{block}$
4. IAIDB vertical
5. IAIDB horizontal
6. La moyenne du AIDSB
7. L'écart type du AIDSB
8. TC



**Figure 1.7** Visualisation des composantes du vecteur d'entrées.

Tirée de Farrugia et Debono (2010, p. 79-80)

[SC]: Expliciter tous les acronymes. Aussi la figure montre AISDB et tu parles de AIDB partout, c'est quoi la relation ?

Ce vecteur représente un macrobloc contenu à l'intérieur d'une trame d'une séquence vidéo. Le AIDB est la différence moyenne des pixels en bordure de macroblobs (M, à la figure 1.7(a)).

Le  $IAIDB_{block}$  est la même mesure que le AIDB, mais appliquée aux blocs  $4 \times 4$  à l'intérieur d'un macrobloc. Le IAIDB est la moyenne de la différence des pixels pour les bordures internes (verticales et horizontales) d'un macrobloc (lignes rouges figure 1.7(b)). Le AIDSB est l'énergie d'un macrobloc  $16 \times 16$ , tel que défini par Ikuno (2007), soit la moyenne du différentiel par rapport à la trame précédente (voir la figure 1.7(c)). La consistance de la texture (TC) est basée sur une approche d'analyse de texture présentée par Wang et He (1990) connue sous le nom de patron binaire local (*Local Binary Pattern (LBP)*).

[SC]: Résultats de cette approche ? performance ?

[SC]: IMPORTANT : Dans un chapitre sur l'état de l'art, on s'attend à avoir une analyse critique des méthodes existantes avec leurs performances, leurs points forts et leurs points faibles. Cela sert à justifier pourquoi on choisit une approche différente.

[SC]: Revoir les références, il manque des accents à décembre. Revoir Ikuno 2007 car accent mal placé et 'master thesis' à franciser. REvoir avec la tite dame les question de référence et traduction de 'in proceeding' , 'in' , ...

En somme, ce chapitre offre un bref survol des travaux de recherche en lien avec : le *Joint Source Channel Decoding*, la résilience aux erreurs lors du transport de vidéo, la validation de syntaxe pour les normes d'encodage vidéo et la détection de la détérioration visuelle dans le domaine des pixels. Ces concepts sont présentés dans un ordre quasi chronologique, où la cohésion conceptuelle a raison de la chronologie. Conséquemment, ils servent de notions fondatrices aux approches proposées dans cet ouvrage, et présentées dans le prochain chapitre.



## BIBLIOGRAPHIE

- Cortes, C. et V. Vapnik, 1995. Support-vector networks. *Machine Learning*, 20 :273–297. ISSN 0885-6125. URL <http://dx.doi.org/10.1007/BF00994018>. 10.1007/BF00994018.
- Duhamel, P. et M. Kieffer, 2010. *Joint Source-Channel Decoding : A Cross-Layer Perspective with Applications in Video Broadcasting*. Academic Press, 1 edition.
- Farrugia, R. A. et C. J. Debono, Decembre 2008. A Robust Error Detection Mechanism for H.264/AVC Coded Video Sequences Based on Support Vector Machines. *IEEE Trans. on Circuits and Systems for Video Technology*, 18(12) :1766–1770. ISSN 1051-8215. doi : 10.1109/TCSVT.2008.2004919.
- Farrugia, R. A. et C. J. Debono, Février 2010. *Resilient Digital Video Transmission over Wireless Channel using Pixel-Level Artefact Detection Mechanisms*, pages 71 – 96. Intech. ISBN 978-953-7619-70-1.
- Ghanbari, M., 1993. Cell loss concealment in ATM video codes. *IEEE Trans. Circuits Syst. Video Technol.*, 3 :238 – 247.
- Haar, A., 1911. Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen*, 71 :38–53. ISSN 0025-5831. URL <http://dx.doi.org/10.1007/BF01456927>. 10.1007/BF01456927.
- Hemami, S. et T.H.-Y. Meng, Juillet 1995. Transform coded image reconstruction exploiting interblock correlation. *Image Processing, IEEE Transactions on*, 4(7) :1023 –1027. ISSN 1057-7149. doi : 10.1109/83.392344.
- Ikuno, J. C., Decembre 2007. Performance of an Error Detection Mechanism for Damaged H.264/AVC Sequences. Master's thesis, Universitat Politècnica de Catalunya.
- Modestino, J. et D. Daut, Novembre 1979. Combined Source-Channel Coding of Images. *Communications, IEEE Transactions on*, 27(11) :1644 – 1659. ISSN 0090-6778. doi : 10.1109/TCOM.1979.1094335.

- Shannon, C. E., 1948. A mathematical theory of communication. *Bell system technical journal*, 27.
- Sun, H. et W. Kwok, Avril 1995. Concealment of damaged block transform coded images using projections onto convex sets. *Image Processing, IEEE Transactions on*, 4(4) :470–477. ISSN 1057-7149. doi : 10.1109/83.370675.
- Sun, H., K. Challapali, et J. Zdepski, Aout 1992. Error concealment in digital simulcast AD-HDTV decoder. *Consumer Electronics, IEEE Transactions on*, 38(3) :108–118. ISSN 0098-3063. doi : 10.1109/30.156671.
- Superiori, L. et O. Nemethova, 2006. Performance of a H.264/AVC Error Detection Algorithm Based on Syntax Analysis. in *Proc. of the Int. Conf. on Advances in Mobile Computing and Multimedia (MoMM 2006)*, pages 49–58. Rinton Press.
- Superiori, L., O. Nemethova, et M. Rupp, Novembre 2007. Detection of visual impairments in the pixel domain of corrupted H.264/AVC packets. *Procs. of the Picture Coding Symposium (PCS 2007), Lisboa, Portugal*, pages 7–9.
- Talluri, R., Juin 1998. Error-resilient video coding in the ISO MPEG-4 standard. *Communications Magazine, IEEE*, 36(6) :112–119. ISSN 0163-6804. doi : 10.1109/35.685373.
- Wang, L. et D.-C. He, 1990. Texture classification using texture spectrum. *Pattern Recognition*, 23(8) :905–910. ISSN 0031-3203. doi : DOI : 10.1016/0031-3203(90)90135-8.
- Wang, Y. et Q.-F. Zhu, Mai 1998. Error control and concealment for video communication : a review. *Proceedings of the IEEE*, 86(5) :974–997. ISSN 0018-9219. doi : 10.1109/5.664283.
- Wang, Y., Q.-F. Zhu, et L. Shaw, Octobre 1993. Maximally smooth image recovery in transform coding. *Communications, IEEE Transactions on*, 41(10) :1544–1551. ISSN 0090-6778. doi : 10.1109/26.237889.

- Yan, B. et K. Wing, Novembre 2003. Analysis and detection of MPEG-4 visual transmission errors over error-prone channels. *Consumer Electronics, IEEE Transactions on*, 49(4) : 1424 – 1430. ISSN 0098-3063. doi : 10.1109/TCE.2003.1261250.
- Ye, S., X. Lin, et Q. Sun, Mai 2003. Content based error detection and concealment for image transmission over wireless channel. *Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on*, volume 2, pages II-368 – II-371 vol.2. doi : 10.1109/ISCAS.2003.1205984.