

CHAPITRE 1

UNE NOUVELLE APPROCHE D'IDENTIFICATION ET DE DISSIMULATION, DE LA DÉTÉRIORATION VISUELLE

Les chapitres antérieurs constituent les éléments d'un panorama technologique permettant l'assimilation des notions présentées dans ce chapitre. Ces éléments recouvrent un large territoire technologique s'étendant de l'encodage vidéo à son transport ainsi que l'impact, la détection et la dissimulation de la détérioration de séquences vidéos codées en respectant la norme H.264.

Nous rappelons d'abord, à la section 1.1, les limitations des approches actuelles de détection d'erreurs. Nous présentons ensuite, à la section 1.2, une amélioration fondamentale aux approches actuelles de détection de la dégradation visuelle à l'intérieur de trames corrompues. Cette approche de détection améliorée sert de base pour un nouveau type d'approche de dissimulation : les approches sélectives. À la section 1.3, deux variantes de ces dernières sont décrites.

1.1 Les limitations des approches actuelles de détection de la détérioration visuelle

Les approches de détection de la détérioration visuelle étudiées ici sont celles présentées dans notre revue de la littérature. Avant de définir les limitations de ces approches, commençons par uniformiser la nomenclature utilisée pour les décrire. Pour ce faire, nous présentons la figure 1.1, soit un survol des concepts liés à la détection de la détérioration visuelle.

Tout d'abord, nous représentons les trames comme des matrices bidimensionnelles de pixels. Cela dit, $\mathbf{F}(t)$ fait référence à la $t^{\text{ième}}$ trame d'une séquence vidéo. Cependant, pour alléger la notation, quand le contexte est clair, nous n'utilisons pas l'indice de la trame. Par exemple, $\mathbf{F}(t)$ devient \mathbf{F} . L'image de référence, celle non compressée, est définie par $\mathbf{F}_O(t)$. De plus, on note \mathbf{P} , la trame qui précède la trame \mathbf{F} , ce qui équivaut à $\mathbf{F}(t - 1)$.

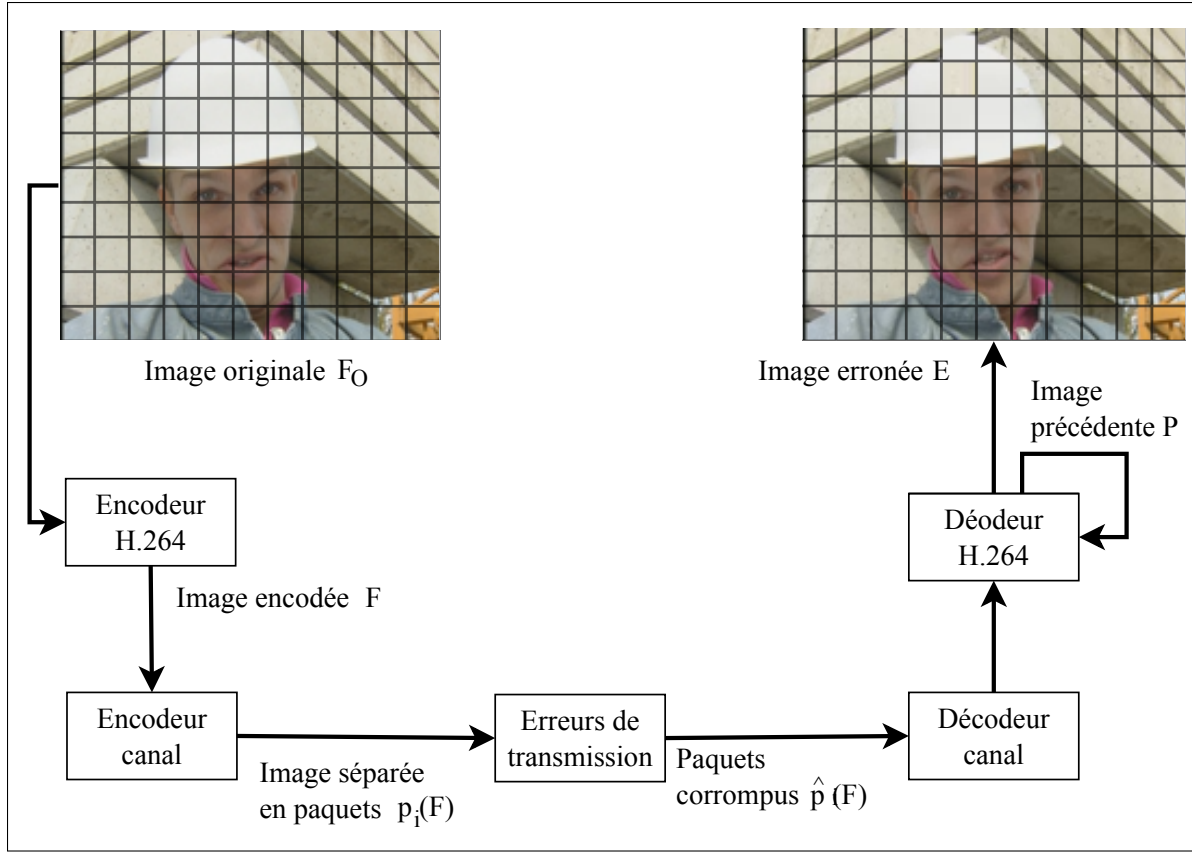


Figure 1.1 Survol des concepts liés à la détection de la détérioration visuelle.

Pour la transmission sur le canal, une trame est divisée en paquets. Cet empaquetage est représenté par $p_i(\mathbf{F})$, où i est l'indice du paquet associé à la trame \mathbf{F} . Nous désignons la corruption du paquet $p_i(\mathbf{F})$ par $\hat{p}_i(\mathbf{F})$. La trame issue du décodage des paquets corrompus $\hat{p}_i(\mathbf{F})$, lorsqu'il est possible, est représentée par $\mathbf{E}(t)$, ce qui équivaut à $\hat{\mathbf{F}}(t)$.

Les auteurs, Superiori et al. (2007) ainsi que Ikuno (2007), présentent une approche de détection où l'on mesure l'énergie des blocs ainsi que leurs effets de blocs à l'intérieur d'un différentiel \mathbf{D} . Ce différentiel, issu de la différence absolue entre la trame erronée \mathbf{E} et la trame qui la précède \mathbf{P} , est défini par la formule suivante :

$$\mathbf{D} = |\mathbf{E} - \mathbf{P}|, \quad (1.1)$$

Les données obtenues par ces mesures sont comparées à des seuils et un système de votes détermine s'il y a erreur. La méthode se base sur l'hypothèse que les pixels ne varient pas beaucoup d'une image à l'autre et que les grandes variations sont probablement dues à des erreurs de transmission. Nous présentons, à titre d'exemple de cette approche, la figure 1.2 tirée de Ikuno (2007, p. 25). Pour éviter le recours aux seuils, les auteurs Farrugia et Debono (2008) collectent une série de mesures provenant de **D** et utilisent une machine à vecteurs de support (Cortes et Vapnik, 1995) pour identifier les blocs erronés.

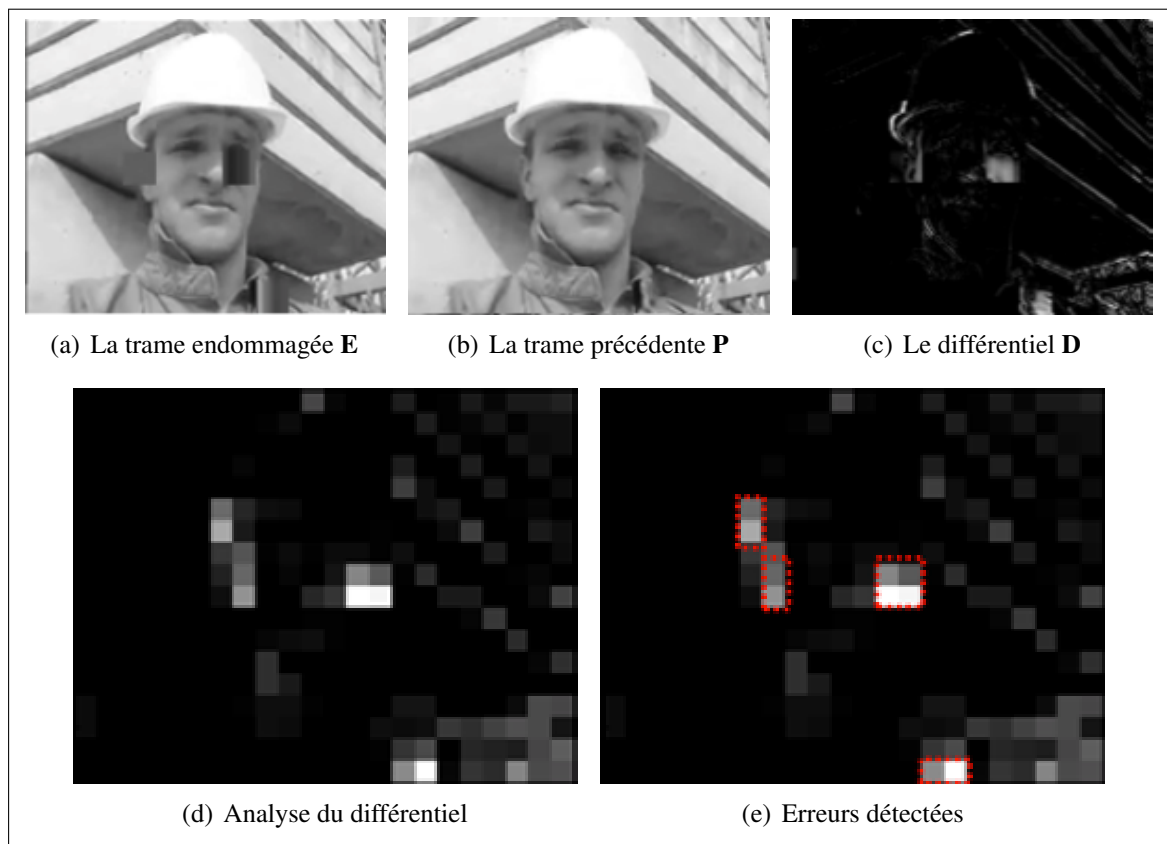


Figure 1.2 Exemple de l'analyse et de la détection d'erreurs.
Tirée de Ikuno (2007, p. 25)

Le problème commun que nous identifions, comme limitation majeure, avec ces approches est l'usage du différentiel **D**. Quoiqu'il puisse sembler alléchant, ce dernier, illustré à la figure 1.2(c), ne contient pas uniquement l'erreur due aux erreurs de transmission, mais aussi la variation des valeurs de pixels propre à la séquence. Cela signifie que le différentiel résultant

des transformations entre les images non erronées, entre \mathbf{F} et \mathbf{P} , tel le mouvement des objets, est aussi contenu dans \mathbf{D} . En effet, utilisant l'inégalité du triangle ($|a - b| \leq |a - c| + |c - b|$), on peut réécrire l'équation précédente comme :

$$\mathbf{D} = |\mathbf{E} - \mathbf{P}| \leq \underbrace{|\mathbf{E} - \mathbf{F}|}_{\mathbf{D}_e} + \underbrace{|\mathbf{F} - \mathbf{P}|}_{\mathbf{D}_v} \quad (1.2)$$

avec \mathbf{D}_e l'erreur produite par les détériorations visuelles dues aux erreurs de transmission et \mathbf{D}_v l'erreur produite par les variations entre les trames. On peut remarquer que sans \mathbf{F} , il est impossible de distinguer, à l'intérieur de \mathbf{D} , les contributions individuelles de \mathbf{D}_e et de \mathbf{D}_v . Les auteurs mentionnés précédemment (Superiori et al. (2007), Ikuno (2007) ainsi que Farrugia et Debono (2008)) misent sur les caractéristiques spatiales de la dégradation visuelle pour tenter d'effectuer cette distinction. Cependant, il est probable que la variation entre les trames \mathbf{D}_v manifeste ce genre de caractéristiques. Comme nous le démontrons aux figures 1.3(b) et 1.3(c).

Par notre contre-exemple, la figure 1.3, nous démontrons l'inefficacité de la détection d'erreurs par \mathbf{D} . La séquence *coastguard* est caractérisée par un déplacement horizontal de la caméra. Ce déplacement crée une très grande variation entre deux trames consécutives, comme illustré à la figure 1.3(b). L'erreur à détecter est identifiée à la figure 1.3(d). Il n'est pas simple d'extraire cette information de \mathbf{D} (figure 1.3(c)) sans connaître \mathbf{F} .

Bref, ce manque de précision, engendré par la variation *intertrame* présente dans le différentiel, force les auteurs à employer des systèmes complexes de votes ou de machines à vecteur de support. À l'opposé, le résultat de notre effort de recherche est une approche de détection simple qui oeuvre dans un contexte différent de \mathbf{D} , où la variation entre les trames est minimisée par la compensation de mouvement.

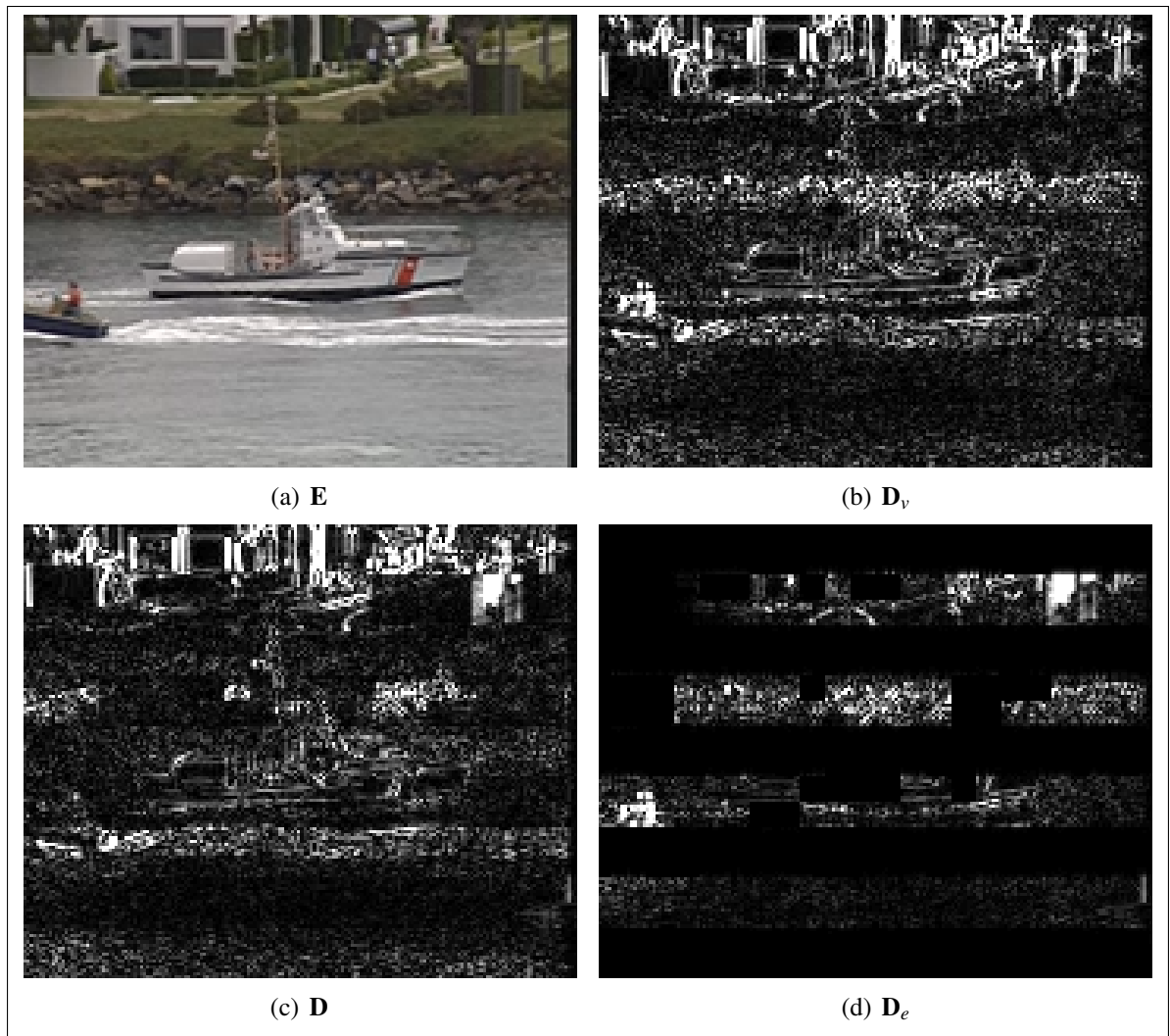


Figure 1.3 Contre-exemple de l'usage de **D** pour effectuer la détection d'erreurs.

1.2 Les effets de blocs compensés par le mouvement

Nous venons de présenter la variation entre les trames comme un obstacle majeur à la détection d'erreurs. En s'intéressant aux origines de cette variation, on constate qu'elle peut provenir d'une translation, d'une déformation, d'un changement d'intensité lumineuse ainsi que de l'insertion ou du retrait de contenu.

[LT]: Faire le lien avec le section sur les erreurs dans le vidéo.

En pratique, l'erreur de transmission agit comme un changement entre les trames équivalant soit à une déformation ou à une insertion de contenu. Ce qui fait en sorte que ces types de changements sont les plus difficiles à distinguer de l'erreur.

Dans le domaine de l'encodage vidéo numérique, la variation entre les trames est un concept qui a suscité une grande quantité d'efforts de recherche. Comme présentées dans notre section sur H.264, les approches de recherche et de compensation de mouvement sont raffinées, depuis maintenant plusieurs années, pour modéliser cette variation afin d'augmenter le taux de compression issue de leur encodage.

[LT]: Faire le lien avec la formule de recherche de mouvement du chapitre sur H.264

En ce qui a trait à la détection de la détérioration visuelle, la recherche de vecteurs de mouvement peut servir à prendre en compte la variation entre les trames (\mathbf{D}_v). En contrepartie, les approches qui reposent sur \mathbf{D} insinuent que $\mathbf{D}_v = 0$.

Pour illustrer cette affirmation, nous recourons à la figure 1.4. Celle-ci montre la somme des pixels pour chaque macrobloc de la trame à la figure 1.3(a), pour \mathbf{D} (figure 1.4(a)) et le résiduel (figure 1.4(b)). Le résiduel est le différentiel absolu entre le bloc à évaluer et son meilleur candidat dans la trame précédente.

La première rangée de blocs de la figure 1.4(a) nous permet de constater que pour plusieurs blocs valides, comme le démontre la figure 1.3(d), le pointage obtenu est supérieur à ceux des blocs corrompus. Ceci implique qu'une approche qui exploite l'énergie des blocs pour effectuer sa détection effectuerait de fausses détections.

De plus, le bloc avec le pointage le plus élevé, dans la figure 1.4(b), est non seulement une erreur, mais l'erreur la plus dérangeante de la figure 1.3(a). L'écart de pointage, par rapport au prochain bloc, est considérable, soit plus de 2000. Notons que le second bloc au pointage le plus élevé est lui aussi erroné. Cependant, il est beaucoup moins dérangeant, ce qui justifie l'écart de 2000.

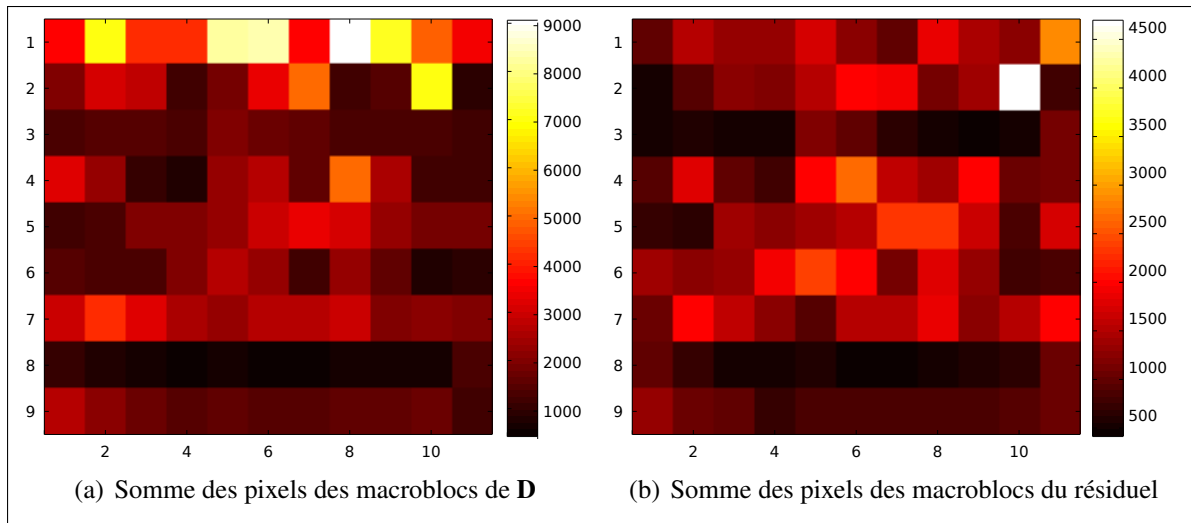


Figure 1.4 Valeurs des macroblochs issues de **D** (a) et du résiduel (b) mesurés sur la figure 1.3(a).

Quoiqu'elle améliore la détection d'erreurs, la compensation de mouvement n'est pas en mesure d'identifier les vecteurs de mouvement corrompus. Comme décrit au chapitre ??, ce type d'erreur insère, dans l'image erronée, du contenu fautif provenant du mauvais endroit dans la trame de référence. Pour ce genre d'erreur, la recherche de mouvement retrouve ce contenu fautif. Ce qui trompe l'algorithme de détection, et ce dernier n'est pas en mesure d'identifier l'erreur.

[LT]: Ajouter référence sur le chapitre des erreurs.

Cette limitation de la recherche de mouvement nous oblige à incorporer un élément important pour la détection d'erreurs : La notion de nuisance. Ce que nous cherchons à détecter n'est pas l'erreur absolue, mais bien l'erreur qui engendre une dégradation visuelle perceptible par le système visuel humain. Dans le chapitre sur l'effet des erreurs sur les séquences vidéo, nous avons conclu que, pour être dérangeantes, les erreurs doivent exhiber un ou plusieurs effets de blocs.

Les effets de blocs compensés par le mouvement (MCB), que nous proposons dans cette recherche, sont la combinaison de ces deux faits. Comme décrit à la figure 1.5, le MCB

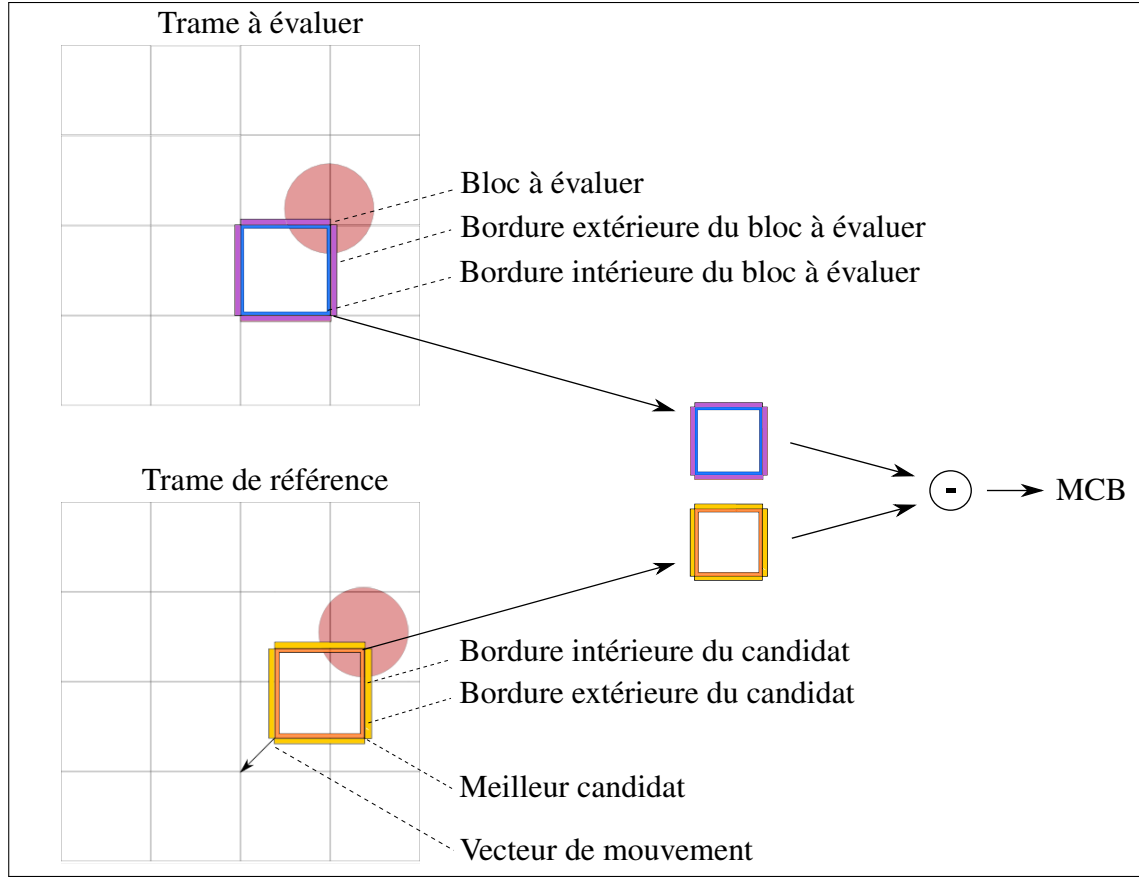


Figure 1.5 Visualisation des composantes liées au calcul du MCB.

recherche les discontinuités spatiales en bordure de blocs entre la trame à encoder et la trame de référence. Ne connaissant pas la trame de référence, nous utilisons la trame précédente.

Avant de procéder à l'explication pratique du fonctionnement du MCB (à la sous-section 1.2.2), nous présentons en détail, à la section 1.2.1, sa définition mathématique afin de bien comprendre comment le MCB est mesuré.

1.2.1 Définition théorique du MCB

Pour cette démonstration, et à des fins de simplicité, nous avons recours à des blocs carrés¹ de taille B . Nous définissons un bloc comme un ensemble de pixels de taille $B \times B$. La hauteur et la largeur d'une trame sont définies, respectivement, par H et W . Les intervalles $I_m = [0, \frac{W}{B} - 1]$

1. Il est aussi possible d'utiliser des blocs rectangulaires.

et $I_n = [0, \frac{H}{B} - 1]$ représentent les indices de blocs à l'intérieur d'une trame. Nous désignons p , comme la demi-hauteur de la fenêtre de recherche de mouvement.

Nous employons (u, v) , comme le vecteur optimal issu de la recherche de mouvement. Cette opération s'exprime par la formule suivante :

$$(\mathbf{U}_{m,n}, \mathbf{V}_{m,n}) = \arg \min_{(u,v) \in K} \text{SAD}(mB, nB, u, v, \mathbf{E}, \mathbf{P}), \quad (1.3)$$

où $K = [-p, p] \times [-p, p]$, $m \in I_m$, $n \in I_n$ et

$$\text{SAD}(x, y, u, v, \mathbf{E}, \mathbf{P}) = \sum_{q=0}^{B-1} \sum_{r=0}^{B-1} |\mathbf{E}_{x+q, y+r} - \mathbf{P}_{x+q+u, y+r+v}|. \quad (1.4)$$

Les composantes des vecteurs de mouvement optimaux (u, v) sont insérées à l'intérieur des matrices \mathbf{U} et \mathbf{V} , à la position correspondante à l'indice du bloc. Cette opération est effectuée sur l'ensemble des blocs qui composent la trame \mathbf{E} . Comme il a déjà été mentionné, nous effectuons la recherche de vecteurs de mouvement afin de tenir compte des variations issues de \mathbf{D}_v afin de les séparer de la mesure de détérioration visuelle. Ainsi, au lieu de faire une différence entre les blocs situés à la même position spatiale, notre métrique effectuera la différence sur les blocs correspondants d'une trame à l'autre, en considérant leur mouvement. De cette manière, nous réduisons l'effet du mouvement des objets entre les trames sur la mesure de différence.

Lors de nos expérimentations, il a été constaté que les gains réalisés par l'usage d'un niveau de précision au quart et au demi-pixel étaient minimes. Alors, pour abréger cette démonstration, ces notions ne seront pas présentées ici. Dans ce chapitre, nous utilisons une précision au pixel entier. Cependant, nous tenons à informer que des niveaux de précision au quart et au demi-pixel ont été considérées dans le cadre de nos recherches.

Pour établir l'appariement des blocs entre la trame erronée et la trame de référence, nous avons recours à la somme de la différence absolue (SAD) entre deux blocs (eq.1.4). Quoiqu'il n'en

résulte pas une identification exacte du mouvement, cette approche est peu complexe et permet l'identification du bloc candidat avec la plus faible variation par rapport au bloc désiré. Ainsi, lorsque, pour un bloc donné, le SAD minimum est élevé, on peut souvent conclure qu'une erreur s'est produite.

Toutefois, l'usage du SAD minimum, tel que calculé en (1.3), n'est pas efficace comme approche de détection d'erreurs, lorsque le contenu erroné est présent dans la trame précédente \mathbf{P} (c.-à-d. lorsqu'un bloc possède un bloc similaire dans la trame précédente). Cette situation se présente souvent lorsque les vecteurs de mouvement sont corrompus. Dans cette situation, ces vecteurs pointent sur le mauvais contenu même si le SAD est faible, ce qui peut souvent entraîner des effets de blocs importants.

Comme illustré à la figure 1.5, c'est pour cette raison que nous mesurons plutôt les effets de blocs en bordure du bloc à évaluer, par rapport aux effets de blocs du candidat résultant, indiqué par le vecteur (u, v) , de la recherche de mouvement (eq.1.3). Nous nommons ainsi cette opération : mesure d'effets de bloc compensés par le mouvement. On y réfère aussi par son acronyme anglais MCB, et elle est définie par la formule :

$$\text{MCB}_d(\mathbf{E}, \mathbf{P}, m, n) = \sum_{l=0}^{B-1} |\mathbf{b}_{d,l}(\mathbf{E}, mB, nB) - \mathbf{b}_{d,l}(\mathbf{P}, mB + \mathbf{U}_{m,n}, nB + \mathbf{V}_{m,n})|, \quad (1.5)$$

où $m \in I_m$, $n \in I_n$. De plus, \mathbf{U} et \mathbf{V} proviennent de l'équation (1.3). Le MCB est appliqué à toutes les bordures d'un bloc ($d \in \mathcal{B}$). Nous référons aux bordures d'un bloc par l'ensemble suivant : $\mathcal{B} = \{\mathcal{N}, \mathcal{E}, \mathcal{S}, \mathcal{W}\}$ et nous mesurons les vecteurs d'effets de bloc $\mathbf{b}_{d,l}$ pour chacune d'elles, à l'aide de :

$$\mathbf{b}_{\mathcal{N},l}(\mathbf{F}, x, y) = \mathbf{F}_{x+l,y} - \mathbf{F}_{x+l,y-1}, \quad (1.6)$$

$$\mathbf{b}_{\mathcal{E},l}(\mathbf{F}, x, y) = \mathbf{F}_{x+B,y+l} - \mathbf{F}_{x+B-1,y+l}, \quad (1.7)$$

$$\mathbf{b}_{\mathcal{S},l}(\mathbf{F}, x, y) = \mathbf{F}_{x+l,y+B} - \mathbf{F}_{x+l,y+B-1}, \quad (1.8)$$

$$\mathbf{b}_{\mathcal{W},l}(\mathbf{F}, x, y) = \mathbf{F}_{x,y+l} - \mathbf{F}_{x-1,y+l}, \quad (1.9)$$

où $l \in [0, B - 1]$, et (x, y) sont les coordonnées en pixels des blocs.

Pour évaluer l'effet global des effets de blocs en bordure d'un bloc, nous utilisons la somme des effets de blocs compensés par le mouvement de ce dernier (SMCB). Cette mesure est obtenue, pour un bloc aux coordonnées de bloc (m, n) , à l'aide de la formule suivante :

$$\text{SMCB}(\mathbf{F}, \mathbf{P}, m, n) = \sum_{d \in \mathcal{B}} \text{MCB}_d(\mathbf{F}, \mathbf{P}, m, n). \quad (1.10)$$

Dans cette formule, on effectue la sommation des valeurs MCB pour chacune des bordures $d \in \mathcal{B}$ du bloc évalué. Ce qui est intéressant avec cette nouvelle approche, c'est que nous mesurons le niveau de discontinuité présent à la bordure du bloc, obtenue en soustrayant le bloc considéré du bloc qui lui correspond le mieux dans l'image précédente. S'il n'y a aucun bloc candidat correspondant bien au bloc considéré ou si le bloc correspond bien, mais qu'il provient de l'application d'un mauvais vecteur de mouvement et, par conséquent il est ainsi hors de son contexte initial, la mesure est élevée. Sinon, un objet s'est seulement déplacé et, par conséquent le contexte visuel est le même autour du bloc considéré et du bloc candidat, même s'il y a beaucoup de détails ou de bordures marquées autour de ces blocs, alors la mesure obtenue sera faible par l'effet soustractif du MCB.

Les figures 1.6, 1.7 et 1.8 montrent la mesure des effets de bloc compensés par le mouvement dans trois situations différentes choisies parmi celles rencontrées lors du décodage de séquence H.264. Pour chaque situation, les quatre bordures (nord, est, sud, ouest) extérieures du bloc à évaluer, encadré en rouge dans (a), sont comparées à celles du meilleur bloc candidat, de la trame précédente, encadré en rouge dans (b).

Dans la première situation, illustrée à la figure 1.6, la trame à évaluer ne contient pas d'erreur. Cependant, on y constate un changement important de l'emplacement du cercle rouge. Dans une telle situation, le différentiel ne serait pas efficace. Néanmoins, la translation est prise en compte, grâce à la recherche de mouvement, ce qui fait que les quatre bordures externes sont identiques.

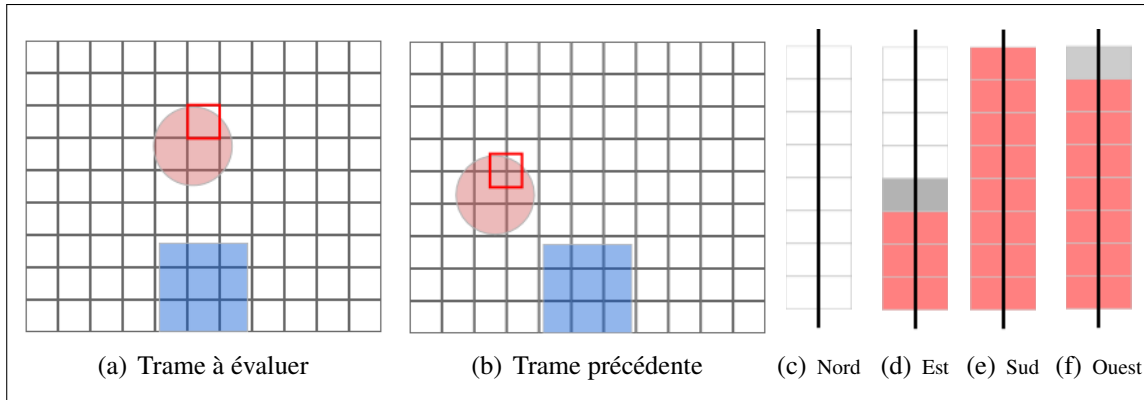


Figure 1.6 Exemple de la mesure des effets de bloc compensés par le mouvement, pour une trame normale.

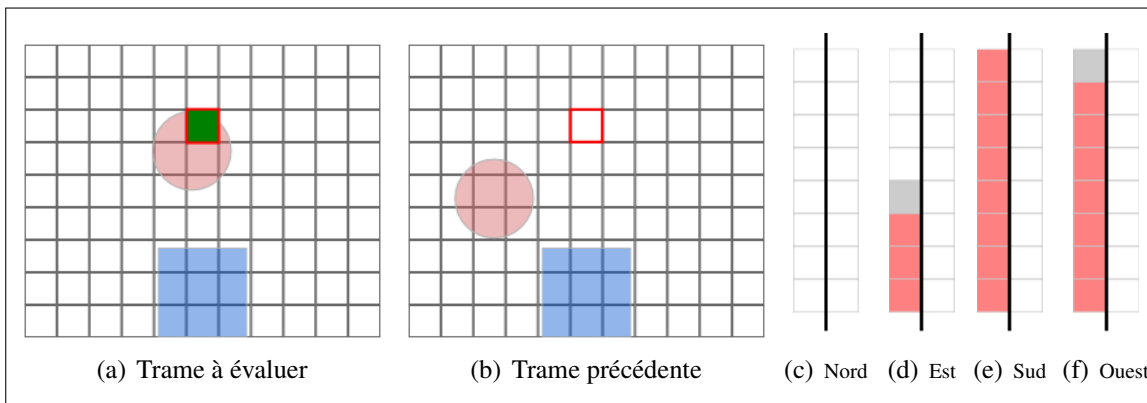


Figure 1.7 Exemple de la mesure des effets de bloc compensés par le mouvement, pour un macrobloc corrompu.

La seconde situation est celle du macrobloc corrompu, comme présenté à la figure 1.7. Dans un tel contexte, le macrobloc corrompu n'existe pas dans la trame précédente. Ceci fait en sorte que la recherche de mouvement ne réussira pas à trouver un bon candidat. Certaines implémentations d'algorithmes de recherche de vecteurs de mouvement, pour des SAD équivalents, vont favoriser les vecteurs de mouvement plus petits. Dans la situation présentée à la figure 1.7(a), le candidat présenté par la figure 1.7(b) est donc plausible. La comparaison des bordures compensées par le mouvement indique que trois bordures sont différentes ((d), (e), (f)), ce qui augmente le pointage du macrobloc de sorte à en permettre l'identification comme macrobloc erroné.

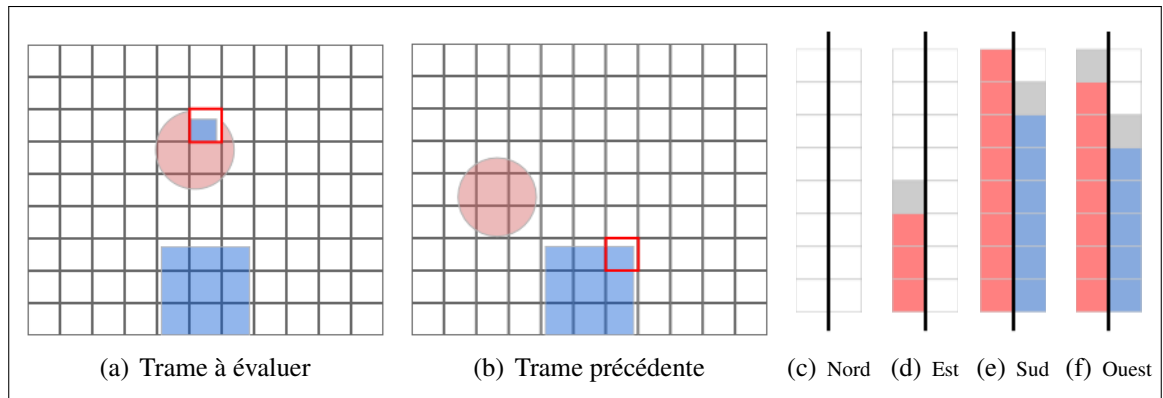


Figure 1.8 Exemple de la mesure des effets de bloc compensés par le mouvement, pour un vecteur de mouvement corrompu.

Le vecteur de mouvement corrompu est notre troisième et dernière situation. Celle-ci est illustrée à la figure 1.8. La particularité de cette situation est que le macrobloc erroné est présent dans la trame précédente et sera retrouvé par l'algorithme de recherche de vecteur de mouvement. Cependant, la comparaison des bordures extérieures démontre que les pixels externes ne s'agencent pas ((d), (e), (f)). Ceci fait en sorte que nous sommes en mesure d'identifier ce macrobloc erroné.

La détection d'effets de bloc par la mesure de pixels en bordure de blocs peut toutefois introduire de fausses détections pour les bordures de blocs avoisinant un bloc erroné. Comme illustré à la figure 1.9(a), dans une telle situation, la valeur du SMCB des blocs voisins de la détérioration visuelle est considérablement augmentée par la bordure limitrophe de l'erreur.

Pour résoudre ce problème, nous proposons le SDMCB, soit une version du SMCB où les bordures à valeurs élevées sont distribuées. La distribution de bordures est guidée par la valeur du SMCB. La valeur d'une bordure élevée est assignée au bloc avec le plus grand SMCB. La figure 1.9 montre le résultat de la distribution et l'algorithme 1.1 (p.15) démontre comment celle-ci est effectuée.

Cet algorithme de distribution de bordures, montré à l'algorithme 1.1 (p.15), est appliqué, en ordre décroissant de valeur de SMCB, sur tous les blocs dont la valeur de SMCB est supérieure à un seuil T_b . Le recours au seuil est justifié, car si la valeur du SMCB est trop basse, la

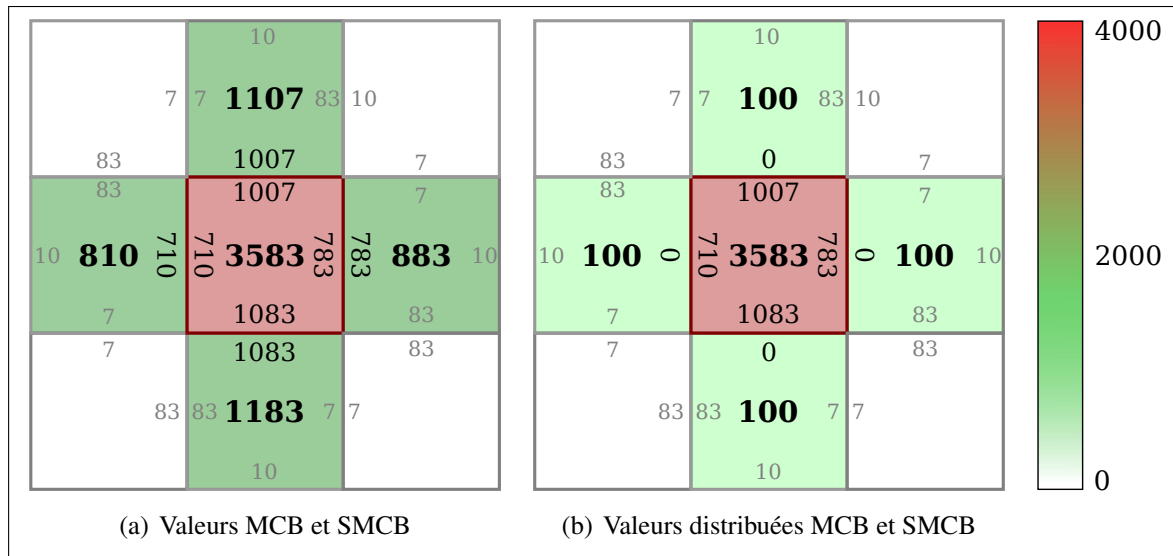


Figure 1.9 Valeurs MCB des bordures et valeurs SMCB (au centre) des blocs avant (a) et après (b) la distribution de bordures.

distribution n'est pas précise et ne sert à rien. Le but de la distribution est d'éviter la propagation des effets de bloc importants aux blocs voisins. Le seuil T_b établit la valeur à partir de laquelle un bloc est considéré comme ayant des effets de bloc importants.

Algorithme 1.1 Distribution de bordures

```

%Bordure supérieure
si  $\text{SMCB}(\mathbf{F}, \mathbf{P}, m, n) < \text{SMCB}(\mathbf{F}, \mathbf{P}, m, n - 1)$  alors
     $\text{MCB}_{\mathcal{N}}(\mathbf{F}, \mathbf{P}, m, n) = 0$ 
sinon
     $\text{MCB}_{\mathcal{S}}(\mathbf{F}, \mathbf{P}, m, n - 1) = 0$ 
fin si

%Bordure droite
si  $\text{SMCB}(\mathbf{F}, \mathbf{P}, m, n) < \text{SMCB}(\mathbf{F}, \mathbf{P}, m + 1, n)$  alors
     $\text{MCB}_{\mathcal{E}}(\mathbf{F}, \mathbf{P}, m, n) = 0$ 
sinon
     $\text{MCB}_{\mathcal{W}}(\mathbf{F}, \mathbf{P}, m + 1, n) = 0$ 
fin si

%Bordure inférieure
si  $\text{SMCB}(\mathbf{F}, \mathbf{P}, m, n) < \text{SMCB}(\mathbf{F}, \mathbf{P}, m, n + 1)$  alors
     $\text{MCB}_{\mathcal{S}}(\mathbf{F}, \mathbf{P}, m, n) = 0$ 
sinon
     $\text{MCB}_{\mathcal{N}}(\mathbf{F}, \mathbf{P}, m, n + 1) = 0$ 
fin si

%Bordure gauche
si  $\text{SMCB}(\mathbf{F}, \mathbf{P}, m, n) < \text{SMCB}(\mathbf{F}, \mathbf{P}, m - 1, n)$  alors
     $\text{MCB}_{\mathcal{W}}(\mathbf{F}, \mathbf{P}, m, n) = 0$ 
sinon
     $\text{MCB}_{\mathcal{E}}(\mathbf{F}, \mathbf{P}, m - 1, n) = 0$ 
fin si
  
```

1.2.2 Explication de l'approche MCB

L'approche MCB comporte deux composantes clés pour l'identification d'erreurs. La première est spatiale, l'analyse de bordures identifie les discontinuités entre les pixels en bordure de blocs. Toutefois, ces discontinuités pourraient appartenir à l'image. La seconde composante est temporelle et, dans ce cas, recherche ces discontinuités dans la trame précédente.

L'insertion ou la déformation de contenu causant un effet de bloc important est le seul cas problématique. Il est rare et l'effet de bloc doit être considérablement important pour être

considéré comme erroné. Nous présentons à la section 1.3 notre approche de détection d'erreurs à l'aide des valeurs de MCB.

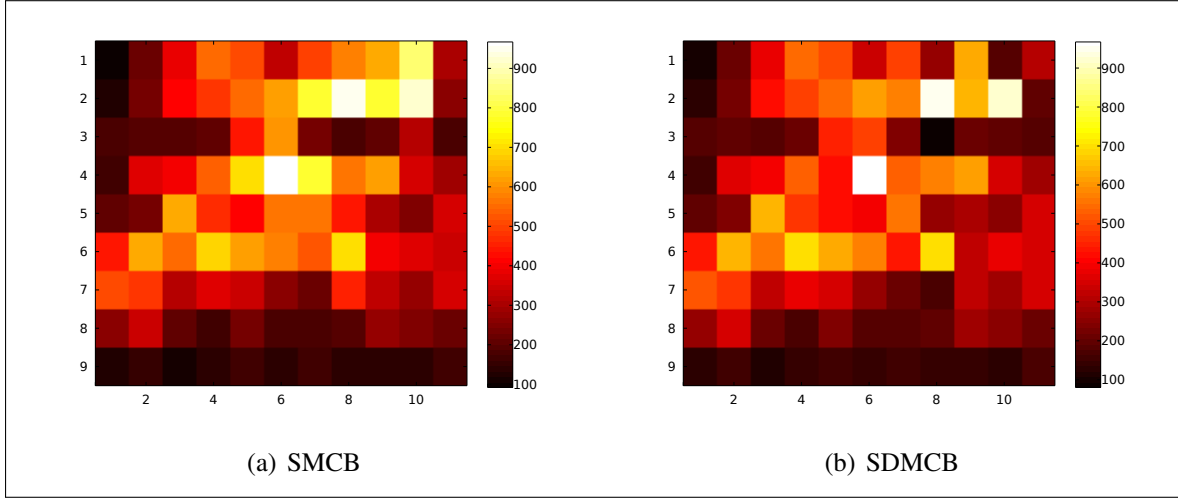


Figure 1.10 Approches de détection d'erreurs basées sur le MCB, mesurées par rapport à la figure 1.3(a). Pour l'approche SDMCB, un seuil $T_b = 900$ a été utilisé.

Nous présentons, à la figure 1.10, les résultats obtenus de l'utilisation des approches SMCB et SDMCB sur la trame erronée de la figure 1.3(a). On y constate en (b) une réduction importante des valeurs des blocs, non erronés, voisins aux macroblocs erronés (c.-à-d. $SMCB > 800$). Rappelons que la propagation d'erreur due à l'usage des bordures extérieures cause cette augmentation des valeurs des macroblocs non erronés.

En ce qui a trait à la détection d'erreur, nous observons que le SDMCB détecte l'erreur la plus dérangeante de la figure 1.3(a). De plus, en regardant l'erreur identifiée à la figure 1.3(d), nous constatons que les valeurs les plus élevées de SDMCB (c.-à-d. $SDMCB > 800$) sont assignées à des macroblocs qui ont subi une détérioration visuelle. Les valeurs basses de SDMCB (c.-à-d. $SDMCB < 200$) sont assigné à des blocs qui ne contiennent pas d'erreurs. Certes, une zone d'incertitude existe entre 200 et 800, cependant, la détérioration visuelle contenue par les macroblocs de cette plage de valeurs n'est pas importante, et les gains de leur dissimulation sont minimes.

À première vue, la comparaison du SDMCB par rapport aux autres approches comme le résiduel et le différentiel \mathbf{D} n'est pas évidente. Cependant, en regardant la zone d'incertitude de \mathbf{D} , soit d'environ 1000 à 9000 (voir figure 1.4(a)), on constate qu'il n'est pas possible de distinguer la détérioration visuelle la plus dérangeante \mathbf{D}_e à la variation entre les trames \mathbf{D}_v .

1.3 Les approches sélectives de dissimulation de la détérioration visuelle.

Dans cette section, nous sortons du contexte théorique de la détection d'erreurs et nous offrons une solution pratique de l'application du MCB. Celle-ci a pour but d'améliorer la qualité visuelle des dissimulations effectuées par le décodeur de référence H.264. Dans le cadre de nos recherches, nous avons intégré notre solution au décodeur inclus dans le logiciel de référence H.264/AVC JM, mais il est aussi concevable de le voir comme un module d'extension (*plugiciel*) destiné à d'autres décodeurs.

L'approche MCB assigne une valeur numérique à un bloc évalué en fonction de l'écart de son effet de bloc par rapport au bloc qui lui ressemble le plus dans la trame précédente. Nous affirmons, résultats à l'appui², que cette approche permet d'identifier les détériorations visuelles importantes dans une trame corrompue. La quantification de la notion d'une dégradation visuelle importante a été un problème colossal que nous avons eu à surmonter, lors de notre implémentation pratique.

Une solution simple est l'usage d'un seuil. Cette valeur numérique définit un point après lequel un bloc avec un certain SDMCB est considéré comme erroné. Ce type d'approche, employée par Superiori et al. (2007) ainsi que Ikuno (2007), ne tient pas compte du caractère variant du contenu d'une séquence vidéo.

L'effet de bloc est relatif à l'image dans laquelle il est perçu. Ceci implique que l'usage d'un seuil fixe ne sera pas en mesure d'identifier les erreurs dans des séquences où les caractéristiques spatiotemporelles divergent.

2. Voir le chapitre Résultats et analyse.

De cette observation est née l'approche sélective. Comme son nom l'identique, cette approche repose sur la notion d'un choix. Ce changement de paradigme provient du constat suivant : dans un contexte pratique, la dissimulation est inutile si l'erreur ne peut être dissimulée. Il est préférable de se concentrer sur le contenu que nous sommes en mesure de dissimuler.

En se basant sur le fait que nous cherchons à effectuer une dissimulation, nous prenons d'un côté une dissimulation traditionnelle, telle la trame calquée ou le calque des vecteurs de mouvement et nous la comparons à la trame à évaluer. Par la suite, nous choisissons la plus petite valeur SDMCB. Que ce soit pour une trame, comme à la section 1.3.1, ou pour un macrobloc, section 1.3.2.

Ces approches se comportent comme des seuils dynamiques capables de s'adapter non seulement au contenu, mais aussi à l'aptitude de dissimulation du macrobloc évalué. Les approches sélectives effectuent une dissimulation d'un macrobloc, seulement lorsque du contenu de qualité visuelle supérieure (jugée par le SDMCB) est disponible.

Une considération pratique importante est le temps de calcul requis par notre algorithme. Ce temps est restreint, vu les contraintes temps réel ainsi que les limitations matérielles des appareils mobiles. Ce qui est dispendieux en terme de temps de calcul pour le MCB est la recherche des vecteurs de mouvement. Cependant, les récentes avancées dans ce domaine, telles PMVFAST (Tourapis et al., 2001) et UMHexagonS (Cai et al., 2009) réduisent de façon substantielle le temps de calcul requis pour trouver les vecteurs de mouvement. De plus, notons que dans ce contexte, le décodeur est informé par les en-têtes RTP des paquets corrompus et mesure le SDMCB uniquement pour les trames qui en résultent.

1.3.1 L'approche sélective au niveau de la trame

Notre première approche sélective repose sur le choix d'une trame. Lorsque des trames sont corrompues, le décodeur peut recourir à plusieurs trames pour la dissimulation : la tranche calquée, la tranche issue du calque des vecteurs de mouvement, la trame endommagée.

Le pointage SDM CB d'une trame est obtenu par la sommation des valeurs du SDM CB pour chacun de ses macroblocs. Cette sommation est décrite par la formule :

$$c_c = \sum_{m=0}^{\frac{W}{B}-1} \sum_{n=0}^{\frac{H}{B}-1} \text{SDM CB}(\mathbf{F}', \mathbf{P}, m, n), \quad (1.11)$$

où \mathbf{F}' est une trame dissimilée par le décodeur, soit par tranche calquée ou par la tranche issue du calque des vecteurs de mouvement, etc. On mesure une seconde sommation des valeurs SDM CB pour chaque bloc d'une trame. Cette fois, nous utilisons la trame endommagée \mathbf{E} :

$$c_e = \sum_{m=0}^{\frac{W}{B}-1} \sum_{n=0}^{\frac{H}{B}-1} \text{SDM CB}(\mathbf{E}, \mathbf{P}, m, n), \quad (1.12)$$

Par la suite, la trame avec le plus petit pointage est choisie comme dissimulation \mathbf{F}^* :

$$\mathbf{F}^* = \begin{cases} \mathbf{E}, & \text{si } c_e < c_c \\ \mathbf{F}', & \text{sinon} \end{cases}. \quad (1.13)$$

1.3.2 L'approche sélective au niveau du macrobloc

Notre deuxième approche offre un niveau de granularité supérieur en effectuant la sélection pour chaque macrobloc. Le but de cette approche est de tirer profit du gain mutuel de différentes approches en fonction des caractéristiques spatiotemporelles des régions d'images.

Dans certaines conditions, la tranche calquée est optimale, dans d'autres c'est la trame erronée ou bien le calque des vecteurs de mouvement. De cette approche résulte une trame de dissimulation formée d'un conglomérat de macroblocs optimaux (jugés selon le SDM CB) issus de différents macroblocs candidats pour la dissimulation.

La dissimulation résultant de la sélection entre les blocs aux indices (m, n) provenant de la trame corrompue \mathbf{E} et la trame issue de la dissimulation par calquage de tranches \mathbf{F}' est obtenue

à l'aide de

$$\mathbf{F}_{x,y}^* = \begin{cases} \mathbf{E}_{x,y}, & \text{si } \text{SDMCB}(\mathbf{E}, \mathbf{P}, m, n) < \text{SDMCB}(\mathbf{F}', \mathbf{P}, m, n) \\ \mathbf{F}'_{x,y}, & \text{sinon} \end{cases}, \quad (1.14)$$

pour tout $x \in mB + [0, B - 1]$, tout $y \in nB + [0, B - 1]$, et pour tout bloc de l'image (c.-à-d. $m \in I_m = [0, \frac{W}{B} - 1]$, et $n \in I_n = [0, \frac{H}{B} - 1]$).

Dans ce chapitre, nous avons présenté deux variantes d'un nouveau type d'algorithme sélectif apte à la dissimulation d'erreurs. Ces variantes agissent à deux différents échelons de la hiérarchie de la vidéo numérique, soit ceux des macroblochs et des trames. Pour guider la sélection, nous proposons un algorithme de détection d'erreurs fondé sur une nouvelle notion : les effets de bloc compensés par le mouvement. Cette dernière prend en compte une partie des changements *intertrame*. Ceux-ci, comme nous avons démontré, entravent à la détection des algorithmes existants.

Le chapitre subséquent porte sur l'analyse des résultats expérimentaux obtenus à l'aide de notre banc d'essai, conçu pour valider les notions présentées dans ce chapitre.

BIBLIOGRAPHIE

- Cai, C., Huanqiang Zeng, et Sanjit K. Mitra, Septembre 2009. Fast motion estimation for H.264. *Image Commun.*, 24 :630–636. ISSN 0923-5965. doi : 10.1016/j.image.2009.02.012. URL <http://portal.acm.org/citation.cfm?id=1595898.1596124>.
- Cortes, C. et Vladimir Vapnik, 1995. Support-vector networks. *Machine Learning*, 20 : 273–297. ISSN 0885-6125. URL <http://dx.doi.org/10.1007/BF00994018>. 10.1007/BF00994018.
- Farrugia, R. A. et Carl James 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.
- Ikuno, J. C., Decembre 2007. Performance of an Error Detection Mechanism for Damaged H.264/AVC Sequences. Master's thesis.
- Superiori, L., Olivia Nemethova, et Markus 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.
- Tourapis, A. M., Oscar C. Au, et Ming L. Liou, 2001. Predictive Motion Vector Field Adaptive Search Technique (PMVFAST) - Enhancing Block Based Motion Estimation. *Proceedings of Visual Communications and Image Processing 2001 (VCIP'01)*. URL <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.17.9457>.