



**POLYTECHNIQUE  
MONTRÉAL**

UNIVERSITÉ  
D'INGÉNIERIE

**INF8770 – Technologies multimédias**

**Automne 2024**

**Travail pratique #2 – Pipeline JPEG2000**

**Groupe 02 – Équipe A37**

**2221053 – Thomas Rouleau**

**2149244 – Justin Lefrançois**

**Soumis à : M. Yassine Achour**

**2024-10-28**

## Question 1 (/5)




**Discuter des effets positifs et négatifs du sous-échantillonnage 4:2:0 lors de la conversion RGB/YUV. Comparer avec au moins un autre sous-échantillonnage de votre choix. Pourquoi fait-on un changement de l'espace de couleur avant de faire un sous-échantillonnage ?**

Le sous-échantillonnage 4:2:0 présente plusieurs avantages et inconvénients lors de la conversion RGB/YUV. L'un des principaux avantages est la réduction de la taille des données : en réduisant la résolution des composantes chromatiques U et V, la quantité d'informations à stocker est diminuée, ce qui permet de compresser le fichier de manière significative. Par exemple, pour un sous-échantillonnage 4:2:0, la taille de l'image sous-échantillonnée réduit de moitié en hauteur puisque seulement 1 ligne sur deux des valeurs de U et V sont conservées. Cela peut donc être réalisé sans perte majeure puisque l'œil humain est moins sensible aux variations de couleurs qu'aux variations de luminosité (luminance). Cette réduction des composantes chromatiques permet donc une compression efficace des données sans altérer beaucoup la qualité visuelle perçue. Cependant, cette technique présente aussi des inconvénients, comme une dégradation de la qualité des couleurs, particulièrement dans les zones de transition avec des détails fins et des contours marqués. Cela peut produire des artefacts visibles, tels que des couleurs floues autour des bords ou des effets de "bavures" sur des images très colorées.

En comparaison, le sous-échantillonnage 4:2:2 réduit la résolution des composantes chromatiques U et V uniquement dans la direction horizontale. Cela permet de conserver une bonne qualité d'image, car la réduction est moins agressive que le 4:2:0, offrant une meilleure précision des couleurs tout en réduisant la taille des données. Cependant, le 4:2:2 génère des fichiers plus volumineux que le 4:2:0, mais plus petits que le 4:4:4, ce qui en fait un compromis intéressant pour les applications nécessitant une qualité supérieure à 4:2:0 sans atteindre la taille élevée des fichiers en 4:4:4.

Enfin, le changement de l'espace de couleur de RGB vers YUV est réalisé avant le sous-échantillonnage puisque comme expliqué plus haut, l'œil est plus sensible aux variations de luminosité (Y) qu'aux variations de chrominance (U et V). Le format de couleur RGB quant à lui, ne fait pas de distinction entre la luminosité et les couleurs mélangeant le tout parmi ses trois composantes rouge, bleu et vert. Il est donc impossible de séparer la couleur de la luminosité en format RGB et donc de sous-échantillonner dans le ratio voulu.

Voici un tableau comparant le résultat d'un sous-échantillonnage 4:2:0 par rapport à un 4:2:0 et un 4:4:4 (image originale) :

4:4:4	4:2:2	4:2:0
		
SSIM = 1,000	SSIM = 0,9769	SSIM = 0,9633

## Question 2 (/4)

**Expliquer l'usage de la DWT dans ce contexte. La DWT seule permet-elle de compresser ?**

La DWT est utilisée dans ce contexte pour analyser les différentes fréquences d'une image et les diviser en sous-bandes : LL, LH, HL, et HH. La sous-bande LL représente les basses fréquences, qui contiennent l'information la plus importante pour l'œil humain, tandis que les autres sous-bandes (LH, HL, HH) contiennent les hautes fréquences, c'est-à-dire les détails. Cette décomposition permet de conserver les informations cruciales dans la sous-bande LL, tout en permettant d'appliquer une compression plus forte sur les sous-bandes de haute fréquence. Cependant, la DWT seule ne permet pas de compresser les données ; elle ne fait que transformer l'image en différentes sous-bandes de fréquences. Pour obtenir une véritable compression, il est nécessaire de d'abord quantifier les coefficients de la DWT avec par exemple le quantificateur à zone morte, puis, de les encoder avec un algorithme de compression (ce qui a été omis dans ce laboratoire). En d'autres termes, la DWT prépare les données de l'image à la compression en les transformant et en les séparant en une forme plus facile à compresser.

### Question 3 (/5)

Expérimenter avec quelques quantificateurs différents et expliquer comment la qualité visuelle de l'image se dégrade-t-elle ? Observez-vous des artefacts avec une certaine structure ? Discuter.

- **Quantificateur léger** (**step\_size = 1**, **dead\_zone\_width = 1**) : L'image reconstruite présente peu de dégradations visuelles. Le SSIM de 0.9633 montre une forte similarité avec l'image originale. La qualité reste très élevée, et les différences sont difficilement perceptibles à l'œil nu.

Image Original



Image Reconstruite  
SSIM: 0.9633



- **Quantificateur modéré** (**step\_size = 10**, **dead\_zone\_width = 10**) : On observe une légère diminution de la qualité visuelle, avec un SSIM d'environ 80%. Des zones commencent à apparaître plus uniformes, particulièrement dans les parties de l'image où les variations étaient initialement plus faibles. Les contours deviennent légèrement moins nets et l'image semble plus sombre.

Image Original



Image Reconstruite  
SSIM: 0.8010



- **Quantificateur fort ( $\text{step\_size} = 30$ ,  $\text{dead\_zone\_width} = 30$ )** : La qualité visuelle se dégrade de manière significative, comme en témoigne le SSIM de 0.6110. L'image perd beaucoup de détails fins, et des artefacts sous forme de zones lissées apparaissent. Les transitions sont moins marquées, créant une impression de perte de profondeur. De plus, la couleur disparaît significativement dans certaines zones, laissant l'image avec une tonalité plus désaturée, presque monochromatique. Cela affecte particulièrement les arrière-plans et les détails complexes, comme la végétation, qui deviennent indistincts.

Image Original



Image Reconstituée  
SSIM: 0.6110



- **Quantification asymétrique ( $\text{step\_size} = 10$ ,  $\text{dead\_zone\_width} = 30$ )** : Avec un SSIM de 0.7358, la dégradation est toujours aussi présente, surtout dans les zones où les détails chromatiques sont importants. Des artefacts tels que des bavures de couleur ou des uniformisations apparaissent, rendant certaines zones de l'image particulièrement floues ou simplifiées. Par rapport à la quantification modérée, on voit que la grande zone morte élimine aussitôt le vert de la végétation. On commence aussi à remarquer des effets inhabituels dans les nuages.

Image Original



Image Reconstituée  
SSIM: 0.7358





- **Quantification asymétrique inverse ( $\text{step\_size} = 30, \text{dead\_zone\_width} = 10$ ) :**  
Avec celle-ci, on peut voir que le pas de quantification affecte plus négativement l'image que la zone morte. Des artefacts particulièrement bizarres semblent apparaître dans certaines régions, notamment dans les nuages. Les détails de l'image sont perdus, ce qui affecte l'apparence des objets. La perte de couleur demeure également très prononcée, ce qui contribue à une impression d'aplatissement et de simplification des contrastes.



En général, plus les paramètres de quantification (le pas et la zone morte) augmentent, plus la dégradation est visible. Les artefacts prennent souvent la forme de lissages, de "taches" dans les zones de transition, ainsi que d'une perte de saturation des couleurs. On observe particulièrement que la quantification asymétrique montre que le pas de quantification a un impact négatif plus important que la zone morte, entraînant des pertes de détails, une décoloration marquée, et la simplification de certaines régions de l'image.

#### Question 4 (/3)

**Nommer un avantage de la DWT par rapport à la DCT et expliquer pourquoi.**

Un des principaux avantages de la Transformée en Ondelettes Discrète par rapport à la Transformée en Cosinus Discrète (DCT) est qu'elle n'entraîne pas de discontinuités de bloc. La DCT traite l'image par petits blocs de tailles fixes (par exemple,  $8 \times 8$ ), ce qui peut provoquer des artefacts visuels appelés artefacts de bloc ou "blocage", particulièrement visibles lorsque le taux de compression est élevé. En revanche, la DWT traite l'image de manière globale, sans la diviser en blocs, ce qui permet de préserver la continuité des contours et des détails. Cela se traduit par une meilleure qualité visuelle, surtout dans les zones avec des transitions douces, car la DWT conserve la continuité des informations à travers toute l'image, évitant ainsi les artefacts gênants souvent observés avec la DCT.

## Question 5 (/3)

**Quel est l'impact du niveau de récursion de la DWT sur la qualité visuelle pour un quantificateur donné ? Pourquoi ?**

L'impact du niveau de récursion de la DWT sur la qualité visuelle est directement lié à la réduction des détails dans l'image. Lorsque le niveau de récursion est faible (comme avec un niveau de 1 dans la première image), la sous-bande LL conserve suffisamment de détails pour maintenir une bonne qualité visuelle. On observe ici une image nette, où les contours et les détails de la voiture et de l'arrière-plan sont encore bien définis.

En revanche, avec un niveau de récursion plus élevé, comme le niveau 3 illustré dans la seconde image, on constate une perte significative de détails, rendant l'image floue et pixelisée. Cette dégradation visuelle s'explique par la suppression accrue des hautes fréquences à chaque niveau de récursion supplémentaire. Ces hautes fréquences sont essentielles pour conserver les détails fins. Ainsi, pour un quantificateur donné, l'augmentation des niveaux de récursion accentue la perte de qualité visuelle, comme on peut le constater dans la seconde image où les détails sont fortement atténués et où l'image semble devenir pixelisée.

Sous-bande LL après DWT - Y



Sous-bande LL après DWT - Y



## Code utilisé et provenance

Le code que nous avons utilisé pour réaliser nos expériences a été conçu par nous-même en suivant les exemples du professeur sur la DWT. Nous avons choisi de ne pas utiliser de bibliothèque python préconçue (à l'exception du SSIM) puisque l'algorithme de la DWT n'est pas compliqué à implémenter et le faire manuellement nous a permis de mieux comprendre son fonctionnement. Nous l'avons adapté en ajoutant la possibilité de choisir un niveau de récursion (une boucle for) ainsi qu'en ajoutant une méthode pour quantifier l'image avec un quantificateur à zone morte. Le reste des ajouts concerne principalement la réalisation du chemin inverse. Pour la conversion de l'espace de couleur RGB vers YUV, nous n'avons pas été capable d'utiliser la formule vue dans l'énoncé puisque celle-ci donnait des résultats inattendus à la fin en changeant totalement les couleurs de l'image reconstruite de manière erronée. Nous avons donc utilisé la formule qui multiplie chaque valeur RVB par un coefficient spécifique.