

Table des matières

1) Prendre une image couleur au format ppm.....	2
2) Dans l'espace RGB.....	3
a) Réduction spatiale (Compression).....	3
b) Ré-échantillonnage (Décompression).....	5
c) Calcul du PSNR entre l'image originale et l'image reconstruite.....	6
3) Dans l'espace YCrCb.....	7
a) Transformation de l'image de l'espace RGB vers l'espace YCrCb.....	7
b) Réduction spatiale (Compression).....	8
c) Ré-échantillonnage (Décompression).....	8
d) Transformation inverse de l'image afin de revenir dans l'espace RGB.....	8
e) Calculer le PSNR entre l'image originale et l'image reconstruite.....	8
4) Conclure et proposer une autre approche pour avoir un taux de compression de 2.....	8
5) Bonus.....	9

Index des figures

<u>Figure 1: Carcassonne, Image originale.....</u>	<u>2</u>
<u>Figure 2: Composante Rouge.....</u>	<u>4</u>
<u>Figure 3: Composante Verte.....</u>	<u>4</u>
<u>Figure 4: Composante Bleue.....</u>	<u>4</u>
<u>Figure 5: Image reconstruite.....</u>	<u>5</u>
<u>Figure 6: PSNR entre l'image originale et l'image reconstruite.....</u>	<u>6</u>
<u>Figure 7: Luminance Y.....</u>	<u>7</u>
<u>Figure 8: Chrominance Rouge Cr.....</u>	<u>7</u>
<u>Figure 9: Chrominance Bleue Cb.....</u>	<u>7</u>
<u>Figure 10: Erreur de redimensionnement de G et B.....</u>	<u>9</u>

1) Prendre une image couleur au format ppm

Voici l'image que j'ai choisie :

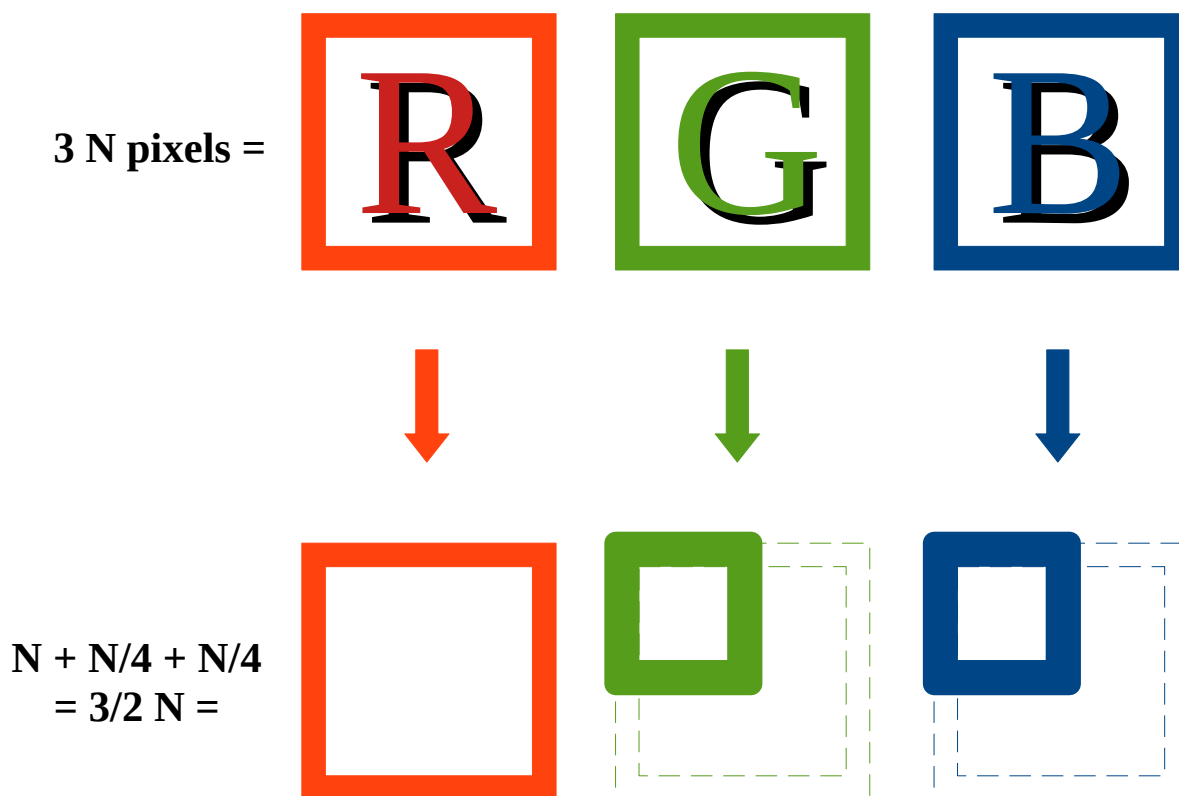


Figure 1: Carcassonne, Image originale

2) Dans l'espace RGB

a) Réduction spatiale (Compression)

J'ai décidé de prendre les composantes Verte et Bleue afin de réduire leur taille en divisant par 2 le nombre de lignes et le nombre de colonnes. Pour cela, on réduit spatialement ces 2 composantes en moyennant les valeurs des pixels voisins. Voici un schéma qui explique cette étape :



Compte rendu TP2 – Codage et Compression Multimédia – 11 Février 2024
LAPORTE Laëtitia

Voici les images des composantes obtenues :



Figure 2: Composante Rouge



Figure 3: Composante Verte



Figure 4: Composante Bleue

b) Ré-échantillonnage (Décompression)

Pour ré-échantillonner des 3 images pgm des composantes rouge, verte et bleue de l'image originale, il faut :

1. Transférer les données de l'image rouge compressée vers le canal rouge de l'image de sortie.
2. Transférer les données de l'image verte compressée vers le canal vert de l'image de sortie, en prenant en compte la réduction de moitié de la résolution.
3. Transférer les données de l'image bleue compressée vers le canal bleu de l'image de sortie, de la même manière que pour le canal vert.

Voici l'image de sortie obtenue :



Figure 5: Image reconstruite

c) Calcul du PSNR entre l'image originale et l'image reconstruite

Formules :

$$\text{PSNR} = 10 \log_{10} \left(\frac{3 * ndgMax^2}{EQMcouleur} \right)$$

$3 * ndgMax^2 \rightarrow$ Puissance du signal

$EQMcouleur \rightarrow$ Puissance du bruit

$$EQM = \frac{1}{MN} \sum (p(i) - p'(i))^2$$

$MN \rightarrow$ Nombre total de pixels dans l'image

$$EQMcouleur = EQM_R + EQM_G + EQM_B$$

Résultat :

Le PSNR obtenu entre l'image originale et l'image reconstruite est de **31,6281 décibels**.

```
laetitia@laetitia-GF75-Thin-10SER:~/Documents/MASTER/M1-S2/CCM/TP2$ ./PSNR carcassonne.ppm imageReconstruite.ppm
31.6281
```

Figure 6: PSNR entre l'image originale et l'image reconstruite

3) Dans l'espace YCrCb

a) Transformation de l'image de l'espace RGB vers l'espace YCrCb

Voici les formules nécessaires pour passer de l'espace RGB vers l'espace YcrCb :

Calcul de la luminance : $Y = 0,299 R + 0,587 G + 0,114 B$

Calcul de la chrominance rouge : $Cr = \frac{R - Y}{2 - 2 * 0,299 R} + 128$

Calcul de la chrominance bleue : $Cb = \frac{B - Y}{2 - 2 * 0,114 B} + 128$

Après cette transformation, on obtient les images Y, Cr et Cb :



Figure 7: Luminance Y

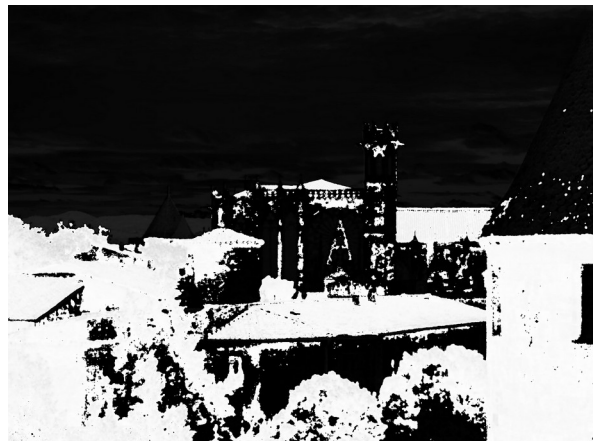


Figure 8: Chrominance Rouge Cr



Figure 9: Chrominance Bleue Cb

b) Réduction spatiale (Compression)

Je n'ai pas réussi à faire la compression d'une image dans l'espace YcrCb.

c) Ré-échantillonnage (Décompression)

Je n'ai pas réussi à faire la décompression d'une image dans l'espace YcrCb.

d) Transformation inverse de l'image afin de revenir dans l'espace RGB

J'ai tout de même réussi à faire la transformation inverse. Voici les formules nécessaires pour passer de l'espace YcrCb vers l'espace RGB :

Calcul de la composante Rouge : $R = Y + 1.402 * Cr$

Calcul de la composante Verte : $G = Y - 0.344 * Cb - 0.714 * Cr$

Calcul de la composante Bleue: $B = Y + 1.772 * Cb$

e) Calculer le PSNR entre l'image originale et l'image reconstruite

Je n'ai pas pu calculer le PSNR entre l'image originale et l'image reconstruite puisque je n'ai pas réussi l'étape b) et c).

4) Conclure et proposer une autre approche pour avoir un taux de compression de 2

En conclusion, j'aurai dû m'apercevoir que le PSNR calculé dans l'espace RGB est moins élevé que le PSNR calculé dans l'espace YcrCb. Car l'espace YCrCb représente une image avec trois composantes : la luminance (Y) et deux composantes de chrominance (Cr et Cb). Cette séparation permet de mieux refléter la perception humaine de la qualité de l'image, car elle accorde plus d'importance à la luminance qu'aux détails de chrominance. Ainsi, le PSNR calculé dans l'espace YCrCb est souvent plus élevé que celui calculé dans l'espace RGB, car il prend mieux en compte cette perception visuelle humaine.

Pour atteindre un taux de compression de 2 sans perdre trop de qualité, une autre approche pourrait consister à utiliser des techniques de compression plus avancées telle que la compression par transformation (comme la Transformée en Cosinus Discrète (DCT) pour convertir les blocs de l'image en un domaine où l'énergie est concentrée dans quelques coefficients).

5) Bonus

Voici une image erronée que j'ai eu lors de la décompression dans l'espace RGB :



Figure 10: Erreur de redimensionnement de G et B