

TI : TP 12 : Compression d'image

Binôme : Benjamin Ruytoor et Aurore Allart

Date : 1^{er} avril 2013

Introduction

Dans ce TP, nous allons voir une méthode couramment utilisée pour compresser une image : la transformée en cosinus discrète (DCT).

1. Transformée en cosinus discrète (DCT)

1. Transformée 2D directe

Il est possible d'utiliser la DCT à une dimension (DCT1D) pour calculer la DCT à 2 dimensions (DCT2D). Il faut tout d'abord calculer pour chaque ligne de l'image la DCT1D, puis refaire appel à DCT1D sur le résultat récupéré du précédent calcul.

La méthode *forwardDCT* de la classe DCT1D prend en paramètre un tableau de signal qui représente les pixels d'une ligne de l'image initiale. Et elle retourne la fréquence (cycles/pixel) de chaque pixel de la ligne.

La méthode *inverseDCT* de la classe DCT1D prend en paramètre un tableau de fréquence et retourne le tableau de pixel calculé.

2. Validation sur un exemple

Voici le résultat obtenu en utilisant la transformée 2D directe :



Prefs	0	1	2	3	4	5	6
0	-415.38	-30.19	-61.20	27.24	56.12	-20.10	-2.39
1	4.47	-21.86	-60.76	10.25	13.15	-7.09	-8.54
2	-46.83	7.37	77.13	-24.56	-28.91	9.93	5.42
3	-48.53	12.07	34.10	-14.76	-10.24	6.30	1.83
4	12.12	-6.55	-13.20	-3.95	-1.88	1.75	-2.79
5	-7.73	2.91	2.38	-5.94	-2.38	0.94	4.30
6	-1.03	0.18	0.42	-2.42	-0.88	-3.02	4.12

Figure de gauche : le résultat en image.

Figure de droite : la valeur de certains pixels de l'image de gauche.

2. Utilisation de la DCT pour la compression JPEG

L'algorithme JPEG est basé sur la transformée 2D directe sur des blocs de 8x8. Cette algorithme a l'avantage d'avoir un ratio de compression élevé mais à l'inconvénient de perdre des données du à la quantification des coefficients de la DCT et des arrondis de nombre réels en entiers.

1. Traitement par blocs d'une image

En utilisant la DCT par bloc sur l'image de lena, on obtient une image en noir. On utilise ensuite la DCT inverse pour obtenir l'image de lena. On constate une légère différence sur l'image résultante au niveau des contours, du à la perte d'information entre la compression et la décompression.

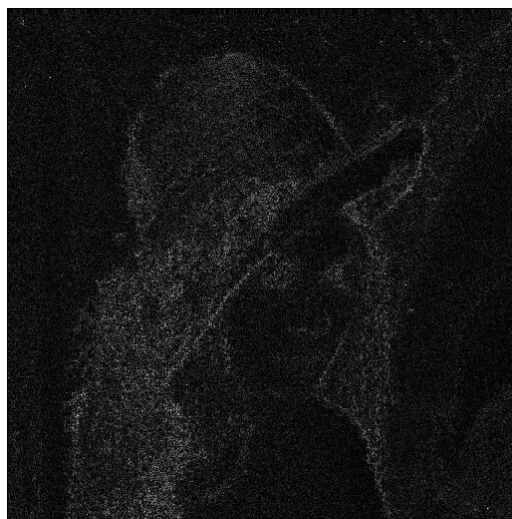
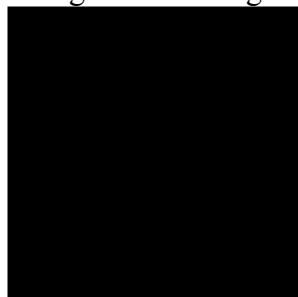


Figure de gauche : image de lena compressée.

Figure de droite : différence entre l'image initiale et l'image résultante.

2. Quantification

La quantification permet de réduire la quantité d'information de hautes fréquences, cela aura une conséquence sur les contours des objets de l'image. Pour l'œil humain la différence n'est pas visible au premier abord car il est moins sensible aux hautes fréquences. Malheureusement, la quantification est la principale cause de perte d'information et donc de réduction de la taille de l'image. Pour l'image 8*8 bits de wikipédia, on obtient :



Prefs	0	1	2	3	4	5	6	7
0	-26.000	-3.000	-6.000	2.000	2.000	-1.000	0.000	0.000
1	0.000	-2.000	-4.000	1.000	1.000	0.000	0.000	0.000
2	-3.000	1.000	5.000	-1.000	-1.000	0.000	0.000	0.000
3	-3.000	1.000	2.000	-1.000	0.000	0.000	0.000	0.000
4	1.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
5	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
6	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
7	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Figure de gauche : la quantification à 100 %.

Figure de droite : la valeur de chaque pixel.

On constate que l'image est noire car chaque pixel est inférieur ou égal à zéro du à la matrice de la quantification pour la luminance. Pour obtenir cette image, on divise chaque bloc de pixel 8*8, par la matrice de quantification.

3. Décompression

La décompression ne fait pas perdre des informations en soit, c'est l'utilisation de la quantification qui en fait perdre.

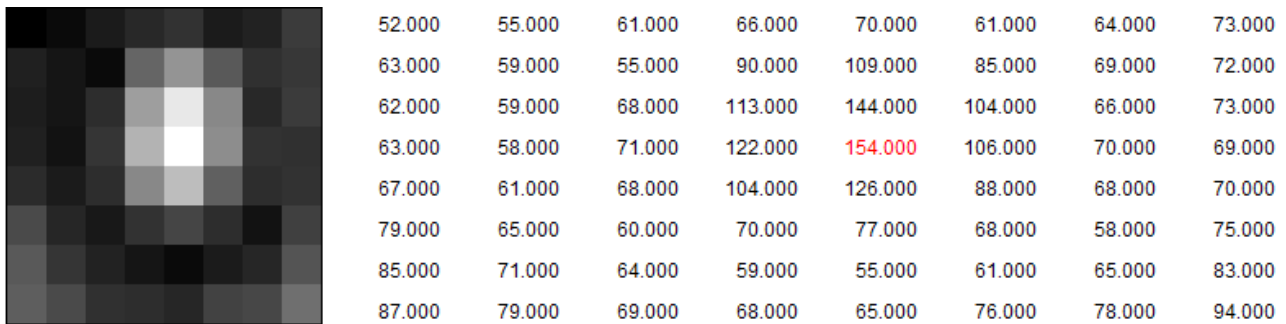


Figure de gauche : l'image de base.

Figure de droite : valeur des pixels de l'image de gauche.

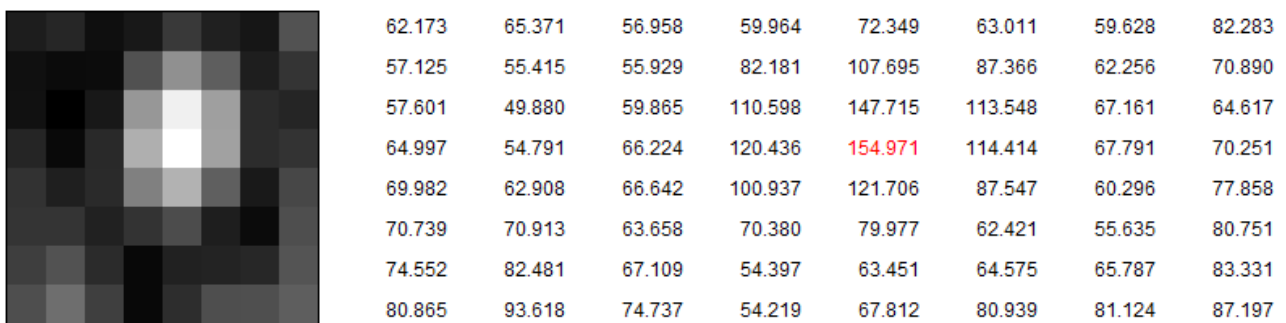


Figure de gauche : image de base après décompression.

Figure de droite : valeur de chaque pixel de l'image de gauche.

On remarque que les valeurs sont différentes du à la perte d'information de la quantification. Après calcul, on constate qu'il y a 6,8945 % d'erreur entre les 2 images.

3. Qualité et performance de la compression JPEG

1. Evaluation de la distorsion

Comme vu dans l'exercice précédent, on remarque une différence au niveau des pixels qui forment les contours des objets de l'image.

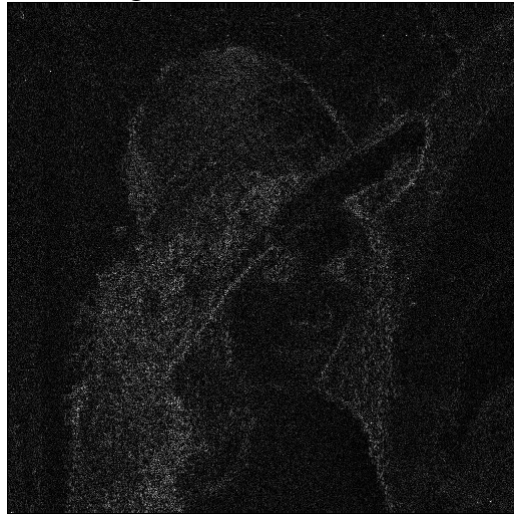


Figure 11 : image résultante de la différence entre l'image d'origine et l'image obtenue.

2. Influence de la quantification

Plus le facteur de qualité est petit et plus la pourcentage d'erreur est grand. Car il y a de plus en plus de groupe de pixel de même niveau de gris. On le voit très bien au niveau de l'épaule de l'ena dans les images suivantes.



Figure de gauche : image obtenue avec un facteur de qualité de 5 %.

Figure de droite : la différence entre l'image de gauche et l'image d'origine.

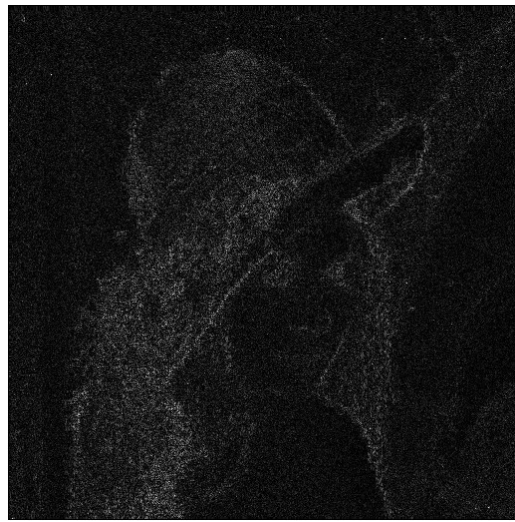


Figure de gauche : image obtenue avec un facteur de qualité à 50 %.

Figure de droite : différence entre l'image d'origine et l'image obtenue.

On ne remarque pas avec un facteur de qualité de 50 % que l'image est compressée.

3. Ratio de compression

En prenant comme exemple les facteurs de qualité précédents, on obtient pour un facteur de qualité de :

- 5 % : une taille d'image 30 fois moins que celle d'origine.
- 50 % : une taille d'image 5 fois moins que celle d'origine.

Plus on prend un facteur petit, et plus la taille est réduite.

Conclusion

La DCT permet de réduire la taille d'une image (ko) mais détruit beaucoup d'informations (apparition d'artefacts de compression). On peut cependant prendre un facteur de qualité qui évite d'avoir trop d'artefacts et en réduisant la taille du fichier par exemple, avec un facteur de 50 %.