

Fête de la Science

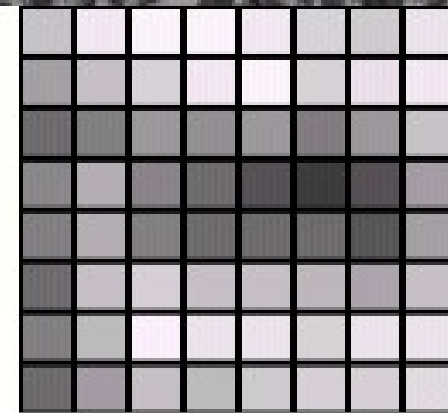
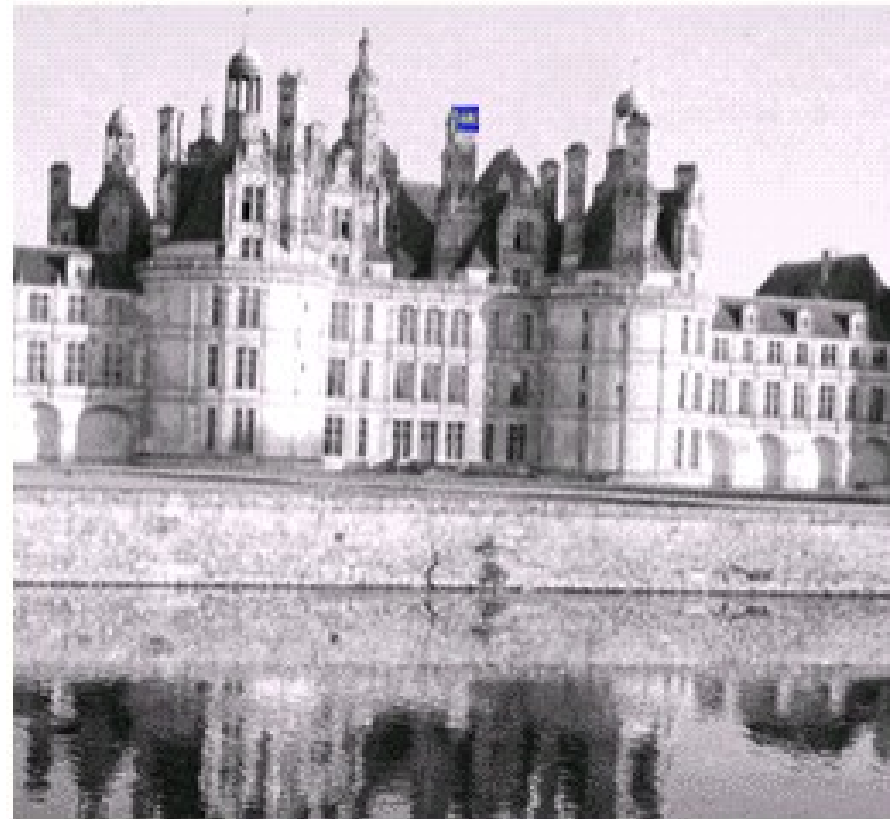
LA COMPRESSION DES IMAGES

Laboratoire de Mathématique et Physique
Théorique
C.N.R.S UMR 6083

Une image numérique

- Une image numérique est constituée de points élémentaires (pixels) ayant chacun une couleur codée généralement sur un ou trois octets (entier entre 0 et 255) représentant l'intensité de la couleur.
- Pour une image couleur, pour chaque pixel on représente l'intensité de chaque couleur Rouge, Vert et Bleu (RGB pour les anglo-saxons).
- Une image plein écran en couleurs représente $1024 \times 768 \times 3$ soit près de 2 360 000 octets.
- Une page A4 scannée à 600 points par pouce peut dépasser 20 Mega octets...
- Il faut donc compresser les images, c'est-à-dire réduire leur taille.

- Sur l'image ci-contre on a isolé un bloc de 8 pixels sur 8 pixels qui ont été agrandis.
- Ce bol est marqué en bleu et correspond au sommet la cheminée au centre de l'image.



65551

Les valeurs correspondantes sont
écrites ci-dessous

188 225 237 237 227 205 199 222

157 186 205 225 237 205 218 225

104 125 149 143 151 123 149 191

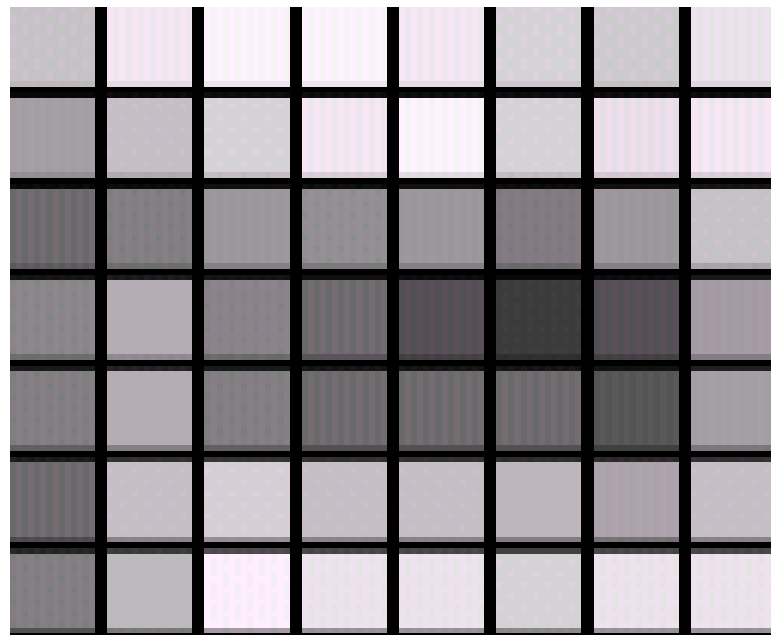
132 169 128 104 81 61 81 152

125 169 125 104 104 107 87 158

107 185 201 187 185 179 161 186

125 180 233 223 223 205 223 223

104 152 186 180 196 203 203 212



Principe de la compression

- On découpe l'image en blocs de 8x8 pixels.
- Chaque bloc est écrit à l'aide de blocs de base correspondants à des fonctions cosinus (transformée discrète en cosinus).
- Selon un seuil choisi on met à zéro les valeurs les moins significatives et on arrondit les autres valeurs (opération de seuillage).
- On réorganise les valeurs seuillées et on les écrit à la suite dans un fichier.
- On applique un algorithme de compression (sans perte) qui produit le fichier compressé.

Principe de la décompression

- On décompresse le fichier et on extrait chaque bloc.
- On fait l'opération inverse du seuillage.
- On recompose le bloc « original » à partir des blocs de base en cosinus.
- On affiche l'image complète.

La transformée en cosinus

- Pour les fonctions d'une variable : soit une fonction f telle que
- $f(0)=$ $f(1)=$
- $f(2)=$ $f(3)=$
-

- On peut l'écrire
exactement comme
somme des fonctions
ci-contre

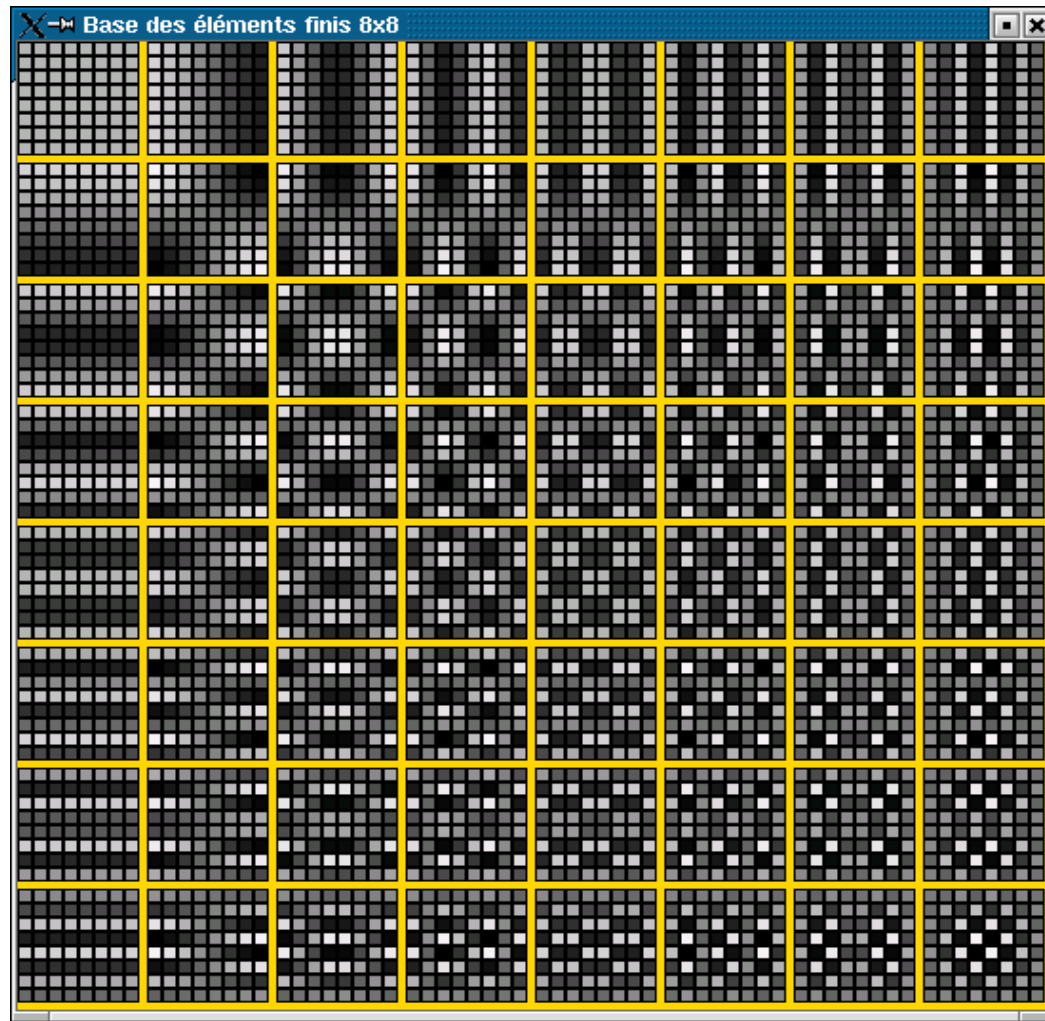
- Si on néglige les quatre dernières fonctions, on obtient ceci.

La transformée à 2 variables

- Pour des fonctions de deux variables on utilise des fonctions de la forme $\cos(x) \cdot \cos(y)$

Quelques fonctions de base

Les blocs de base



La décomposition en cosinus

Les coefficients de la décomposition

336 -63 -34 -110 -5 -57 -6 0

28 41 22 6 32 3 -8 0

262 -75 -82 14 -12 30 0 17

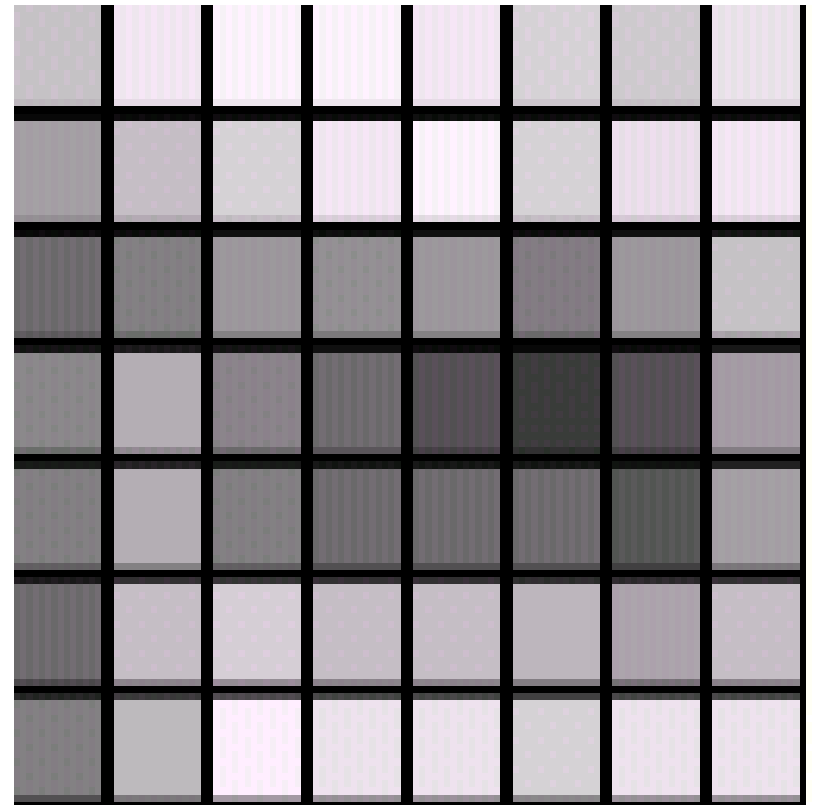
95 38 -25 0 -14 -12 2 -13

-94 59 58 4 2 -18 4 -19

6 26 6 -26 0 9 -2 6

-38 -10 -8 -4 -5 -4 8 -1

-17 -16 6 4 5 1 7 4



- Les valeurs les plus significatives sont en haut à gauche, les moins significatives en bas à droite.
- A ce stade l'image n'est pas dégradée : si on fait la transformation inverse on retrouve le bloc intact.
- Les valeurs en bas à droite sont généralement petites et peuvent être négligées.
- Les autres valeurs vont être arrondies.

336 -63 -34 -110 -5 -57 -6 0

28 41 22 6 32 3 -8 0

262 -75 -82 14 -12 30 0 17

95 38 -25 0 -14 -12 2 -13

-94 59 58 4 2 -18 4 -19

6 26 6 -26 0 9 -2 6

-38 -10 -8 -4 -5 -4 8 -1

-17 -16 6 4 5 1 7 4

La fonction de seuil

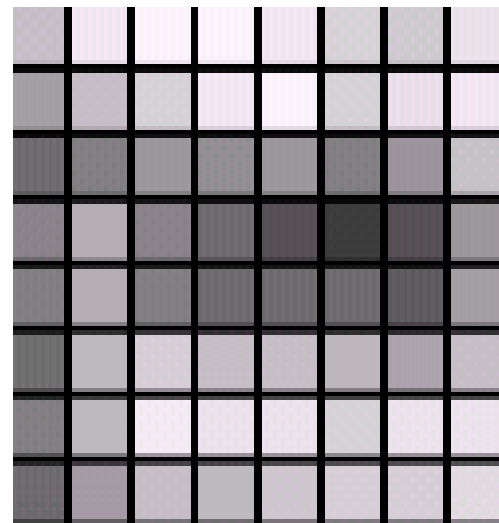
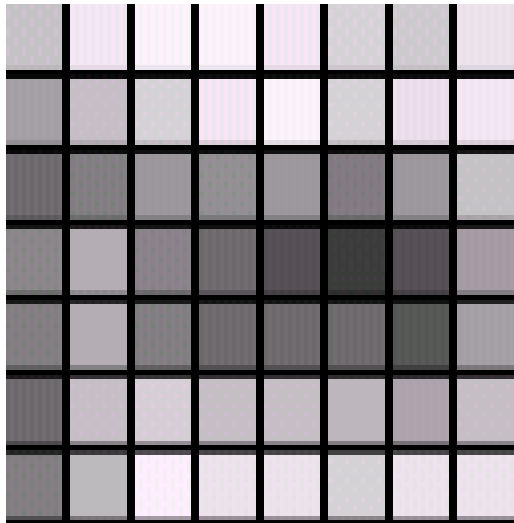
- On choisit un seuil s
- Pour la valeur N située sur la ligne i et la colonne j on calcule
- $F = 1 + ((i-1) + (j-1)) * s$
- On remplace N par le quotient (entier) de N par F .

Au seuil 4

67	-7	-2	-6	0	-2	0	0
3	3	1	0	1	0	0	0
20	-4	-3	0	0	0	0	0
5	1	-1	0	0	0	0	0
-4	2	2	0	0	0	0	0
0	0	0	0	0	0	0	0
-1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

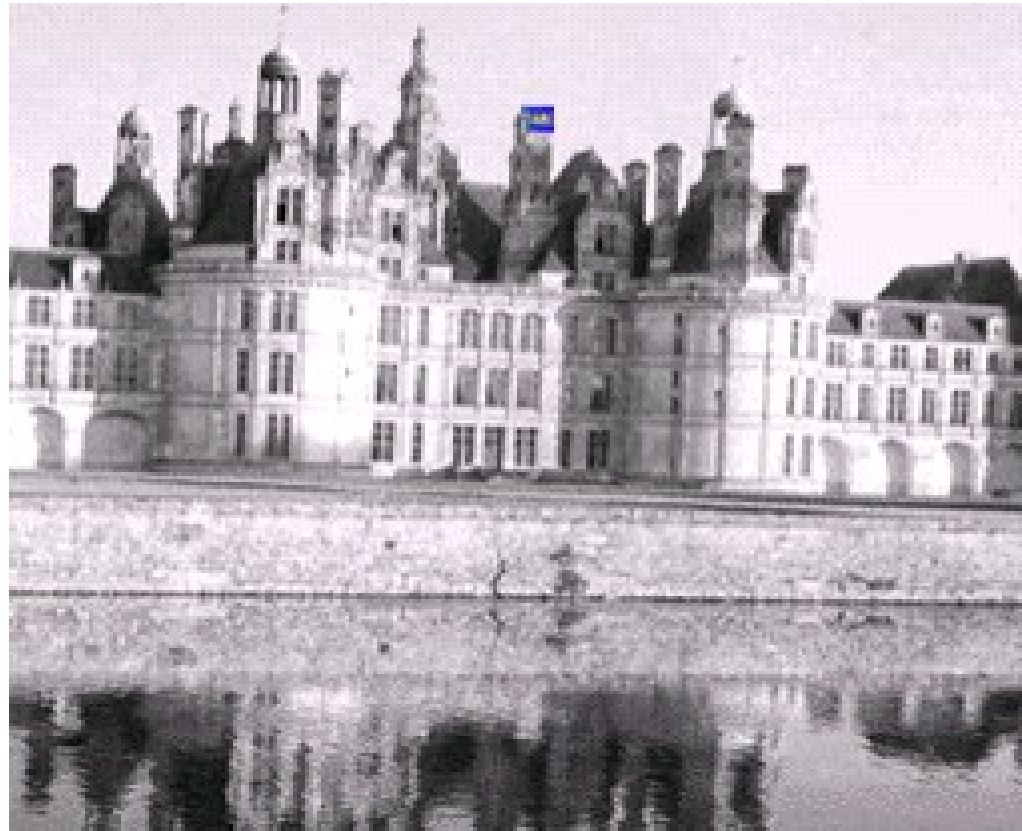
- Par principe les valeurs en bas à droite vont se retrouver à zéro, alors que les valeurs en haut à gauche vont être simplement divisées.
- Lorsqu'on procède à l'opération inverse, on multiplie la valeur située en (i,j) par F
- Les valeurs nulles restent nulles et les valeurs les plus significatives après ce double traitement seront simplement arrondies à des valeurs proches.

Le bloc avant et après (seuil 4)



L'image compressée (seuil 4)

11



L'image compressée (seuil 32)

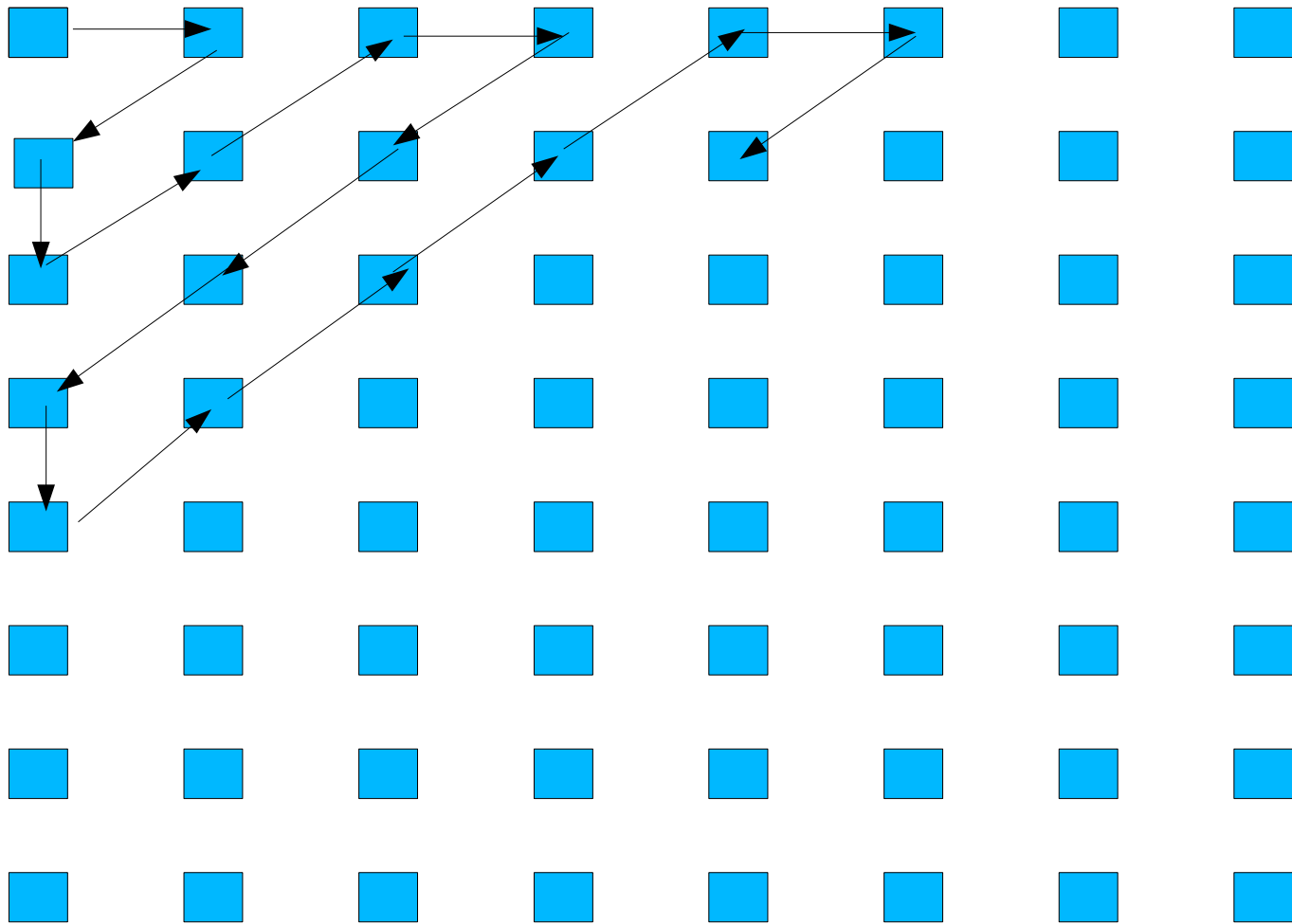
11



L'écriture du fichier

- Il y a encore une astuce dans l'écriture d'un bloc seuillé : on écrit les termes en zig-zag afin de grouper d'abord les termes les plus significatifs et de grouper beaucoup de termes nuls ensuite.
- Cette façon de procéder, ainsi que le petit nombre de valeurs différentes rencontrées garantit que l'algorithme de compression sera beaucoup plus performant que si on l'applique au fichier original.

L'écriture du bloc



Quelques résultats

- Taille fichier originel : 65536 octets
- Compression simple (seuil 0) : 52761 octets
- Compression JPEG seuil 4 : 10800 octets, on a divisé la taille par 6.
- Compression JPEG seuil 32 : 2087 octets, on a divisé la taille par 31 !

Pourquoi est-ce efficace ?

- Parce que l'on opère sur des blocs de petite taille où de grandes variations de l'image sont peu probables.
- Parce que la méthode de seuillage détruit des détails peu perceptibles : l'oeil est peu sensible aux variations de luminosité des pixels.
- Parce que l'organisation des données augmente l'efficacité des programmes de compression.

Points faibles

- Largement inadapté aux images comportant du texte.
- Un effet « mosaïque » désagréable aux forts taux de compression (seuils élevés).
- Impossibilité de relier le taux de compression avec le seuil de traitement.

FIN

- Présentation entièrement réalisée avec des logiciels libres
- Copyright Jean-René LICOIS