Institut
Mines-Telecom

Introduction à la compression

Marco Cagnazzo

SIGMA201

Plan

Compression d'images

- Perception et représentation
- Transformées linéaires
- TCD
- JPEG

Compression audio

- L'audition
- Normes

Plan

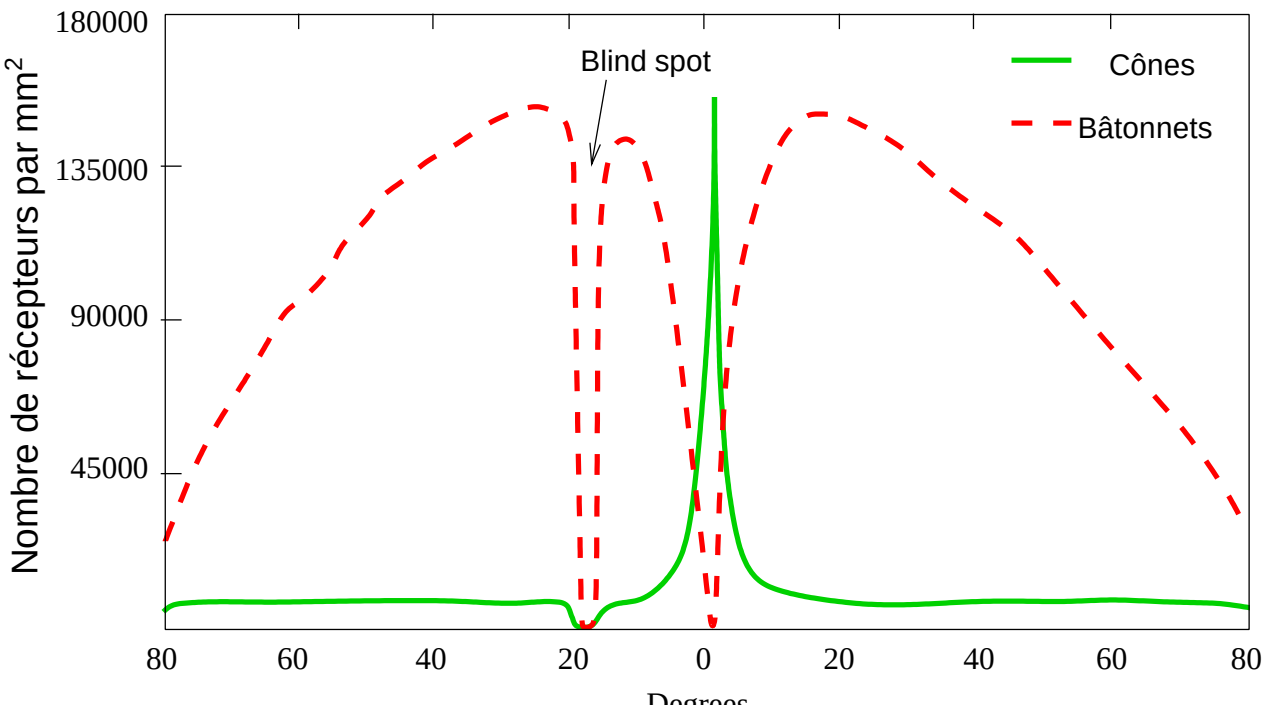
Compression d'images

- Perception et représentation
- Transformées linéaires
- TCD
- JPEG

Compression audio

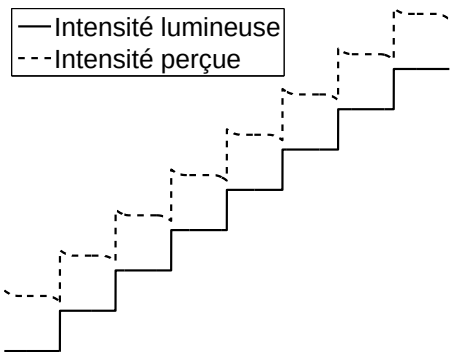
L'oeil

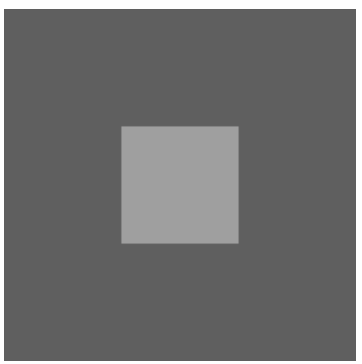
- Lumière transformée en impulsion nerveuse par les récepteurs (rétine)
  - **Cônes** (6÷7 millions, au centre de la rétine) : très sensibles aux couleurs, une bonne résolution, demandent beaucoup de lumière
  - **Bâtonnets** (75÷150 millions) : sensibles à l'intensité lumineuse, faible résolution, très sensibles à faible luminosité

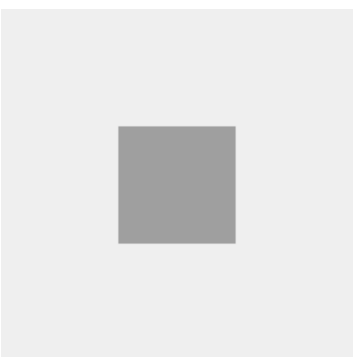


Perception de la lumière

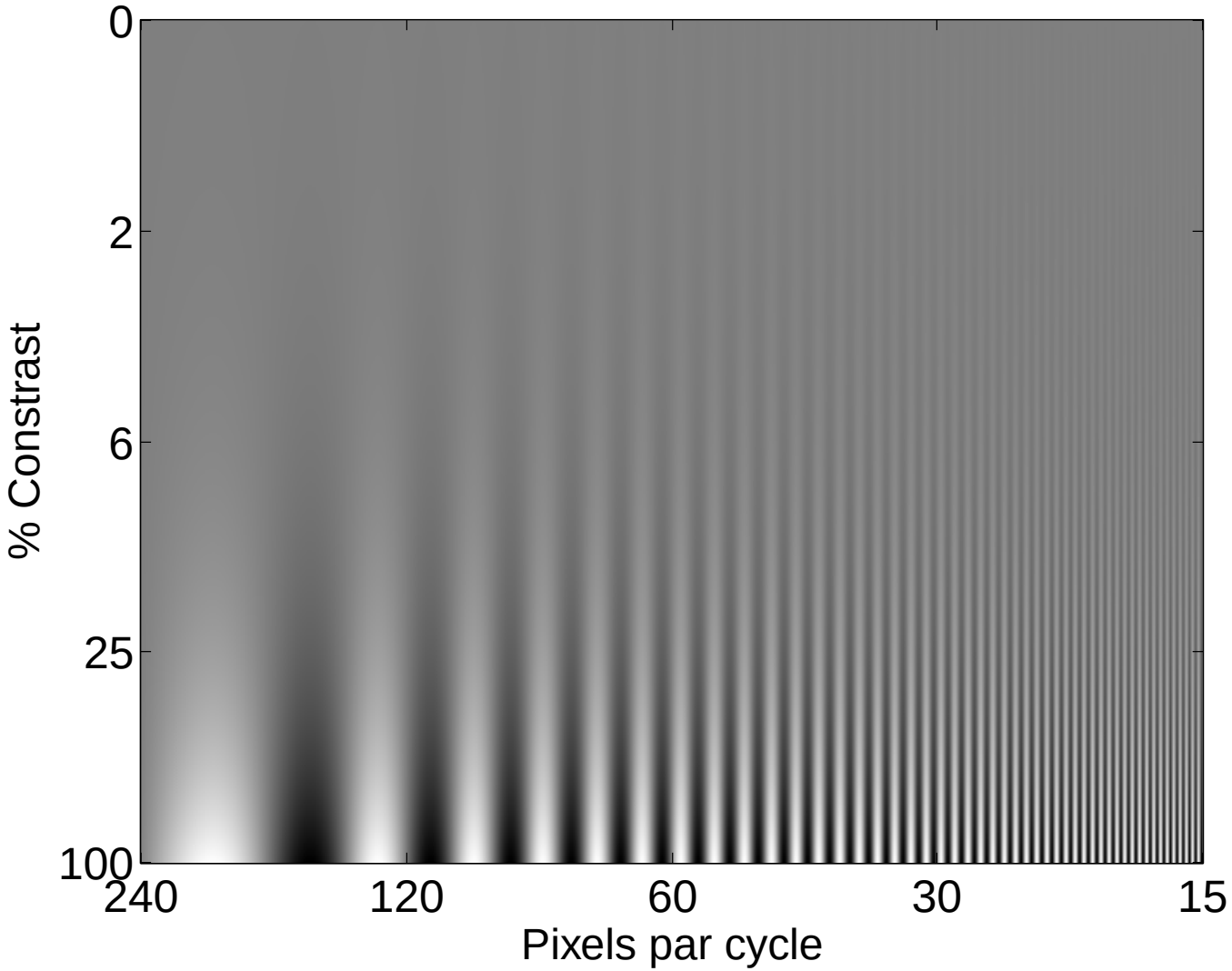
- Luminosité perçue : fonction logarithmique de l'intensité
- Dynamique des valeurs d'intensité : $\approx 10^{10}$ (100dB)
- Le système visuel ne peut pas opérer sur cette échelle simultanément
- Changements de la sensibilité globale, dynamique beaucoup plus limitée
- Luminosité perçue : ce n'est pas une simple fonction de l'intensité







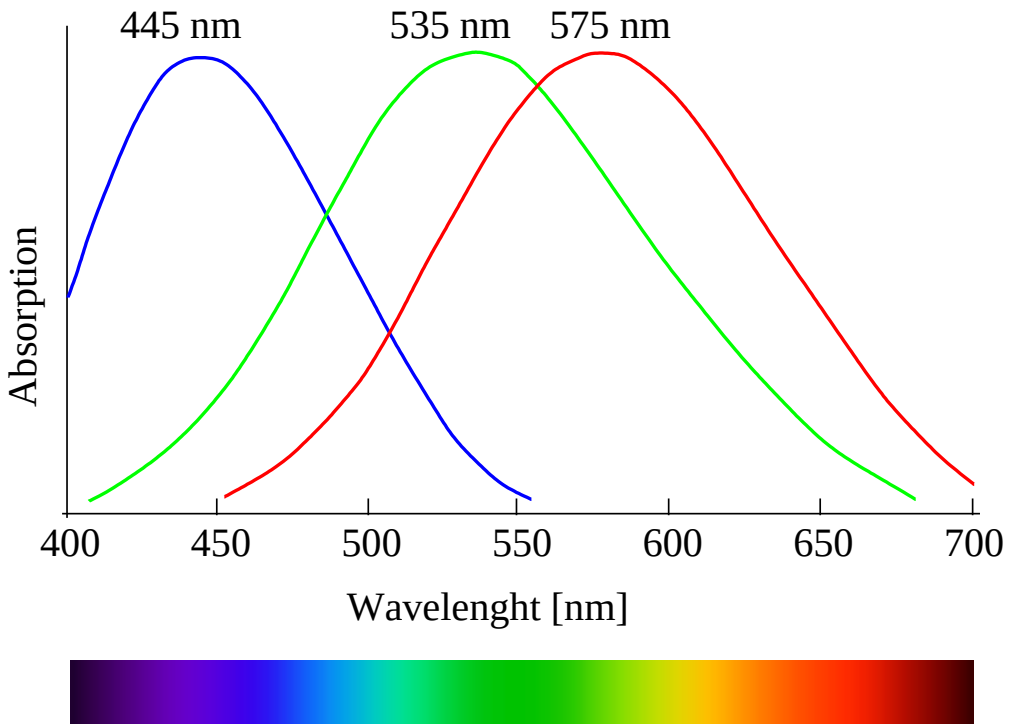
Sensibilité aux fréquences spatiales



- La sensibilité au contraste est l'habilité à discerner différents niveaux de luminosité
- Maximum à environ 2-5 cycles par degré

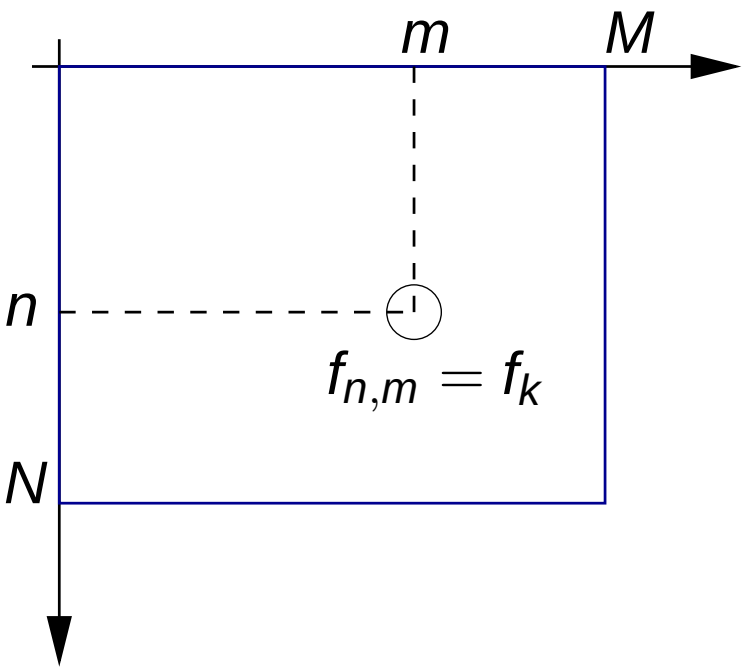
Perception de la couleur

- Spectre visible : 400÷700 nm
- Cônes sensibles à différents intervalles
  - 65% sensible au rouge
  - 33% sensible au vert
  - 2% sensible au bleu (mais très sensibles)
- Sensation de la couleur : correspond au *tristimulus*
- Couleur obtenue comme combinaison des *couleurs primaires*



Représentation des images numériques

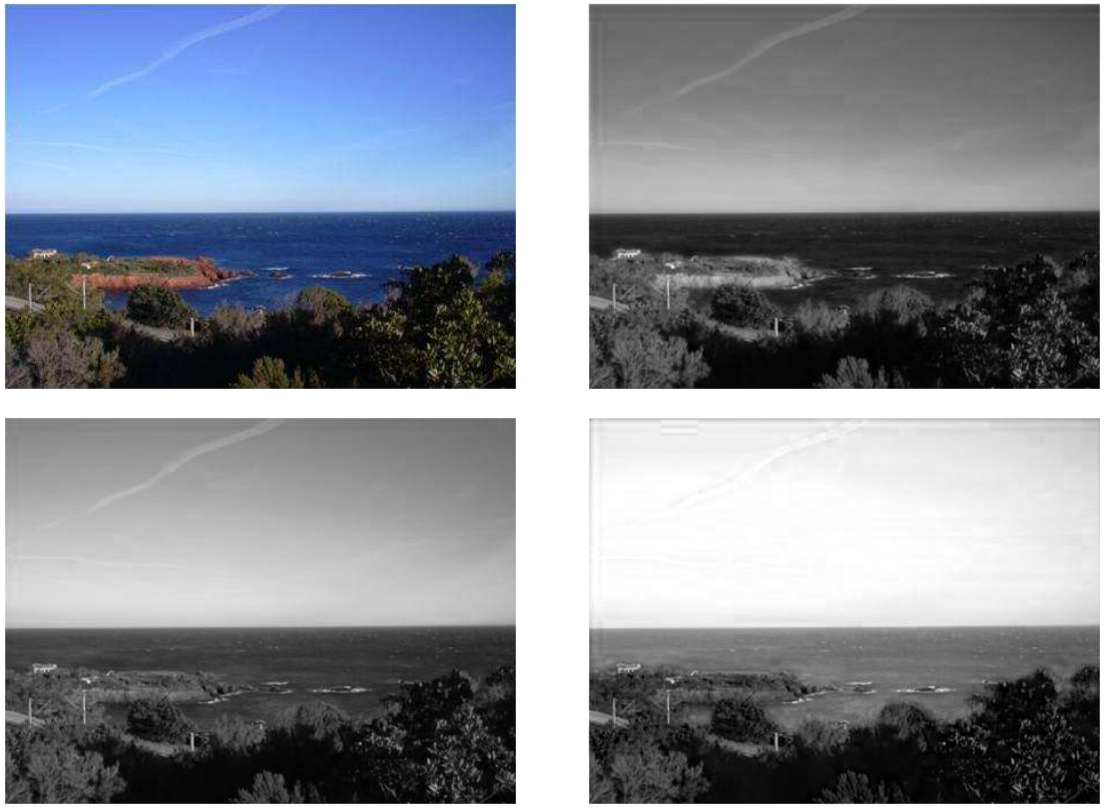
- Grille discrète, image $N \times M$ pixels
- A chaque pixel (m, n), on associe un ordre de traitement k
- Généralement, balayage ligne par ligne unilatéral : $k = (n-1)M + m$
- On notera indifféremment $f_{n,m}$ ou f_k



Représentation des images numériques

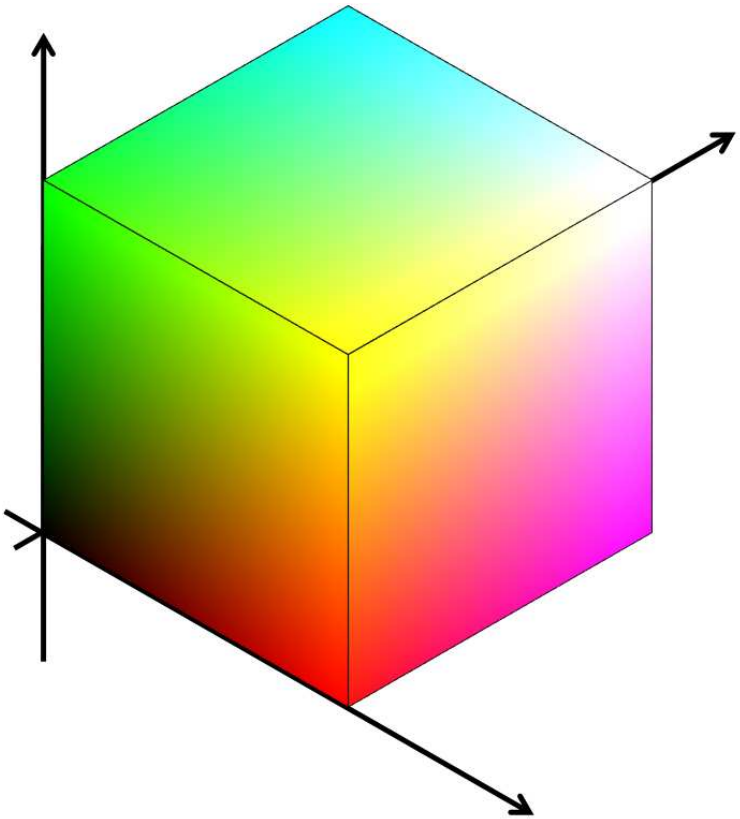
Images couleurs : Format RVB

Images en couleurs : trois composantes, chacune représentée comme une image en niveaux de gris.

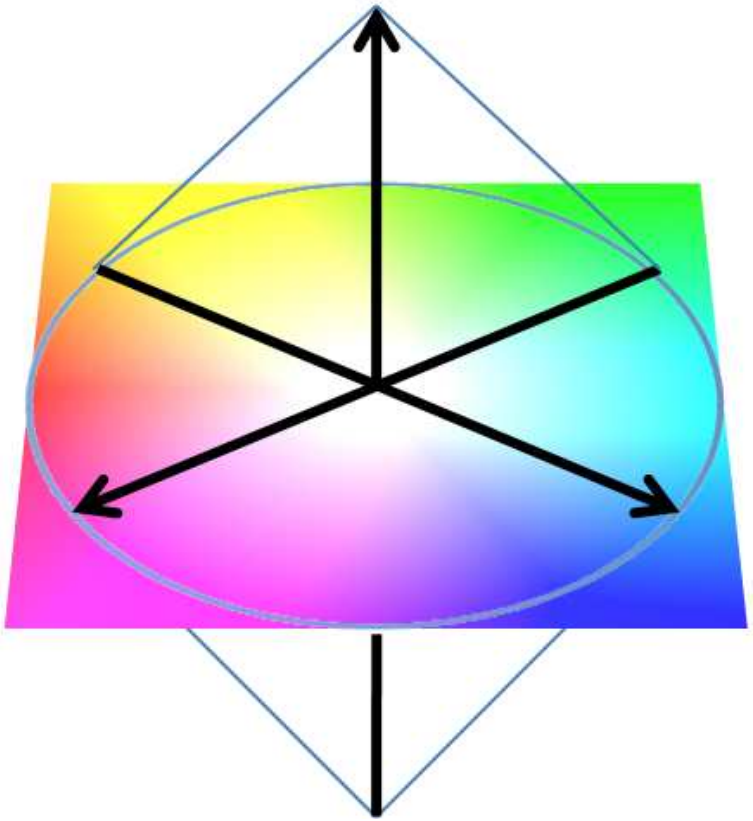


Représentation des images numériques

Espaces de couleurs



Espace RGB



Espace HSV

Représentation des images numériques

Images couleurs : Format YUV

Images en couleurs : une composante de luminance et deux de chrominance (sous-échantillonnées).




Représentation de la vidéo numérique

- Séquence d'images numériques
- On ajoute la dépendance du temps
- Trois composantes dans le cas de vidéo couleur
- Représentation RVB ou luminance/chrominance
- Sous-échantillonnage des composantes couleur

$$I : (n, m, T, c) \rightarrow x \in \left\{0, 1, \ldots, 2^b - 1\right\}$$

Pourquoi comprimer ?

Exemple 1 : Libraire de photos numériques

- Images à 5 Megapixel
- Trois composantes couleur
- Un octet par composant
- Occupation mémoire : 15 Mo par photo
- Publication sur le Web ?

Pourquoi comprimer ?

Exemple 2 : Télévision

- Système numérique : TV-SD
  1 composante de luminance 576×720
  2 composantes de chrominance 288×360
  quantification sur 8 bits
  25 images par seconde
  $R \approx 125$ Mbps
  $\Rightarrow$ bande de fréquence ?
- 2 heures de film > 100 Go
- TV-HD :
  - 1920×1080 pixels, 50 images par second :
    $R \approx 1250$ Mbps, 2h de film > 1To

Fondements de la compression

La redondance des images

- Redondance statistique des données
  - homogénéité des images
  - similitude entre images successives
- Redondance psychovisuel
  - sensibilité aux baisses fréquences
  - effets de masquage
  - autres limites du système visuel humain
- Un algorithme de compression (ou codage) doit exploiter au maximum la redondance des données

Fondements de la compression

Types d'algorithme

- Algorithmes sans perte (*lossless*)
  - Reconstruction parfaite
  - Basés sur la redondance statistique
  - Faible rapport de compression
- Algorithmes avec perte (*lossy*)
  - Image reconstruite $\neq$ image originale
  - Basés sur la quantification
  - Redondance psychovisuel : "visually lossless"
  - Rapport de compression élevé

Critères de performance

Débit

Rapport (taux) de compression

- $T = \frac{B_{\text{in}}}{B_{\text{out}}} = \frac{R_{\text{in}}}{R_{\text{out}}}$

Débit de codage

- Image : $R = \frac{B_{\text{out}}}{NM}$ [bpp]
- Vidéo, son : $R = \frac{B_{\text{out}}}{T}$ [bps]

Codage d'image sans perte : $T \leq 3$
Codage d'image avec perte : $T \approx 5 \rightarrow ?$
Codage vidéo avec perte : $T \approx 20 \rightarrow ?$

Critères de performance

Qualité et distorsion

Le seul débit n'est pas suffisant pour évaluer un algorithme avec pertes

Il faut déterminer la qualité ou la distorsion de l'image reconstruite

- Les **Critères objectifs** sont fonctions mathématiques de
  - $f_{n,m}$: image d'origine ; et
  - $\widetilde{f}_{n,m}$: image reconstruite après compression
- Les *critères objectifs non perceptuels* ne prennent pas en compte les caractéristiques du système visuel humain (SVH)
- Les *critères objectifs perceptuels* sont basés sur un modèle du système visuel humain (SVH)

Critères de performance

Critères objectifs non perceptuels

- Image d'erreur : $\mathcal{E}(f, \widetilde{f}) = f - \widetilde{f}$
- **Erreur quadratique moyenne** (MSE) $\mathcal{D}$:

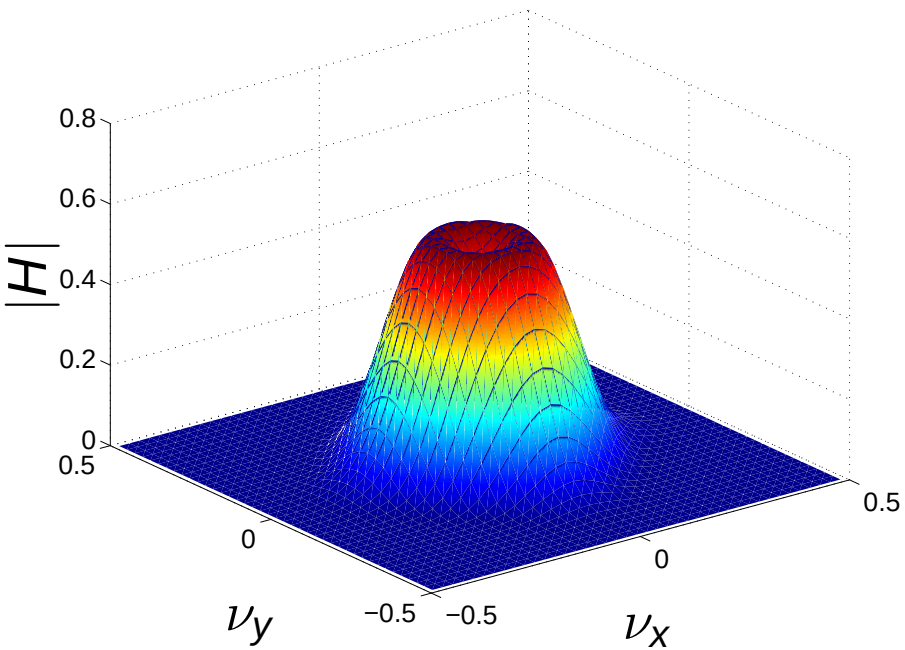
$$\mathcal{D}(f, \widetilde{f}) = \frac{1}{N\,M} \|\mathcal{E}\|^2 = \frac{1}{N\,M} \sum_{n=1}^{N} \sum_{m=1}^{M} \mathcal{E}_{n,m}^2$$

- **Rapport signal sur bruit crête** :
  $\mathrm{PSNR}(f, \widetilde{f}) = 10 \log_{10} \left(\frac{255^2}{\mathcal{D}(f, \widetilde{f})} \right)$
- Mesure simple, dérivable, liée à la norme $\mathcal{L}^2$

Critères de performance

Critères objectifs perceptuels

Weighted PSNR : Étant donnée une fonction de pondération fréquentielle (filtre linéaire) h :



$$\mathrm{WPSNR}(f,\widetilde{f}) = 10\,\log_{10}\left(\frac{255^2}{\mathcal{D}_{\mathrm{W}}(f,\widetilde{f})}\right) \qquad \text{où}$$

$$\mathcal{D}_{\mathrm{W}}(f,\widetilde{f}) = \frac{1}{N\,M}\,\|h * \mathcal{E}\|^2$$

Critères de performance

Critères objectifs perceptuels

Structural Similarity Index (SSIM Index) entre deux blocs x et y :

$$\mathrm{SSIM}(x,y) = [l(x,y)]^{\alpha} \cdot [c(x,y)]^{\beta} \cdot [s(x,y)]^{\gamma}$$

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \qquad \text{Luminance}$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \qquad \text{Contraste}$$

$$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \qquad \text{Structure}$$

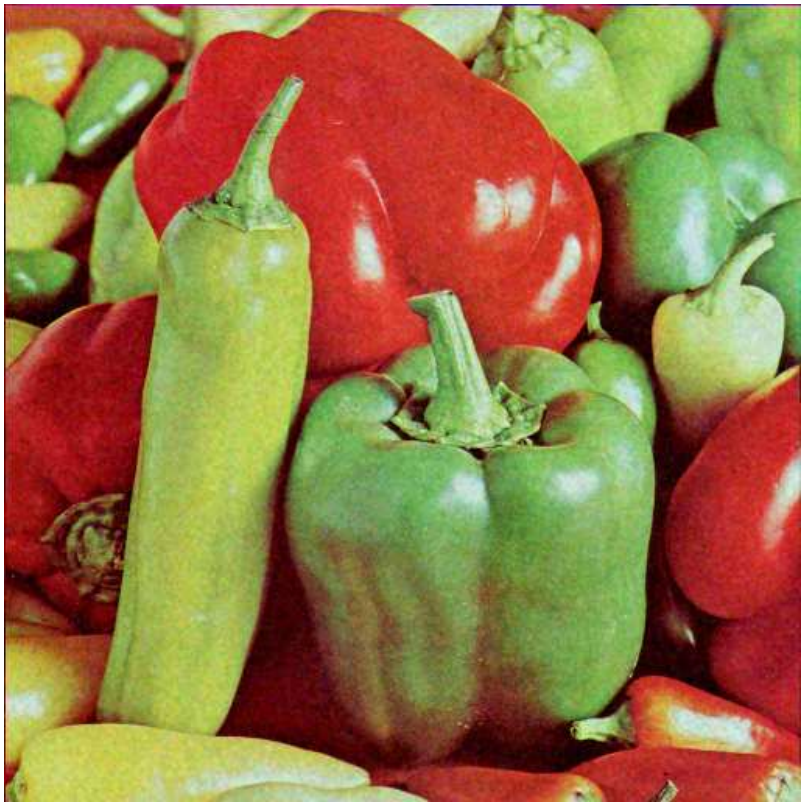
pour simplicité, $\alpha = \beta = \gamma = 1$, $C_3 = C_2/2$

$$\mathrm{SSIM} = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

Le SSIM entre deux images est la moyenne des SSIM des blocs

Perception des erreurs

Erreur distribuée, bruit blanc $\sigma = 4$



MSE: 16

SSIM: 0.906

Perception des erreurs

Bruit concentré sur 100 $\times$ 100 pixels

MSE: 16

SSIM: 0.972

Perception des erreurs

Bruit concentré sur les contours (estimation par filtre de Sobel)

MSE: 16

SSIM: 0.987

Perception des erreurs

Bruit sur les hautes fréquences spatiales

MSE: 16

SSIM: 0.882

Perception des erreurs

Sous-échantillonnage dans l'espace des couleurs

MSE: 15.69

SSIM: 0.846

Perception et qualité : bilan

- Modèles perceptuels nécessaires pour des bons performances de compression
- Système d'audition relativement bien compris, et exploité dans les codeurs audio
- Système de perception visuel encore loin d'être parfaitement compris
- Manque de mesures perceptuelles de qualité complètement fiables
- Tout de même, les meilleures performances de compression ne peuvent pas être atteintes si on tient pas en compte l'aspect psychovisuel

Critères de performance

Qualité et distorsion

- Les **Critères subjectifs** sont basés sur l'évaluation de la qualité des image faite par des humaines
  - Difficulté de créer un bon modèle du SVH
  - Analyse statistique des résultats
  - Évaluations longues, difficiles et coûteuses
- En conclusion, souvent on se limite à utiliser les critères objectifs non perceptuels :
  - Simplicité
  - Interprétation géométrique (norme euclidienne)
  - Optimisation analytique
  - Relation avec la qualité perçue ?

Critères de performance

Complexité, retard et robustesse

- La complexité d'un algorithme de codage peut être limitée par :
  - contraintes liées à l'application (temps réel)
  - limités du matériel (hardware)
  - coût économique
- Le retard est normalement mesuré au codeur
  - Lié à la complexité
  - Influencé par l'ordre de codage
- Robustesse: sensibilité de l'algorithme de compression/reconstruction à des petites altérations du code comprimé (erreurs de transmission)

Critères de performance

Bilan

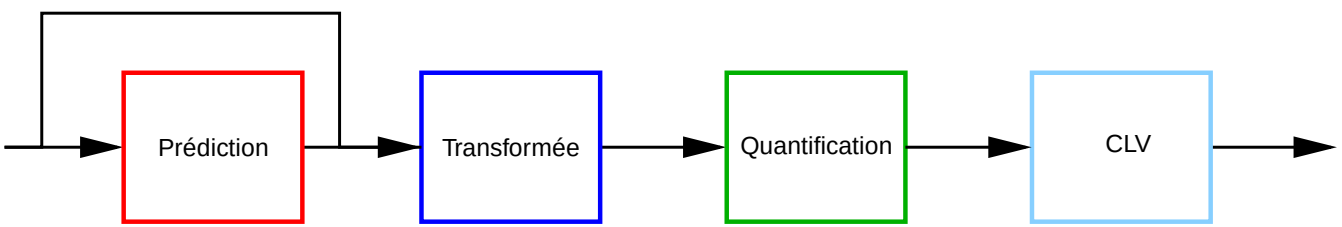
Besoins contradictoire :

$\Uparrow$ Qualité
$\Uparrow$ Robustesse

$\Downarrow$ Débit
$\Downarrow$ Complexité
$\Downarrow$ Retard

Outils fondamentaux pour la compression

- Transformée
  - Concentre l'information en peu de coefficients
- Prédiction
  - Méthode alternative (et parfois complémentaire) à la transformée pour concentrer l'information
- Quantification
  - Réduction du débit : représentation grossière des coefficients moins importants
- Codage sans pertes (codes à longueur variable)
  - Réduction de la redondance résiduelle



Quantification par blocs

- Bloc de variables aléatoires $\mathbf{X} = [X_1, X_2, \ldots, X_N]^T$
- Distorsion : $D = \frac{1}{N}\mathrm{E}\left[\|\mathbf{X} - \hat{\mathbf{X}}\|^2\right]$

$$D = \frac{1}{N}\mathrm{E}\left[\sum_{i=1}^{N} |X_i - \hat{X}_i|^2\right]$$

$$= \frac{1}{N}\sum_{i=1}^{N}\mathrm{E}\left[|X_i - \hat{X}_i|^2\right] = \frac{1}{N}\sum_{i=1}^{N} D_i$$

$$D_i = h_i \sigma_i^2 2^{-2R_i}$$

$$D = \frac{1}{N}\sum_{i=1}^{N} h_i \sigma_i^2 2^{-2R_i}$$

Quantification par blocs

- Problème : minimiser D sous contrainte

$$\min_{\mathbf{R}} D(\mathbf{R}) \text{ soumis à } \sum_{i=1}^{N} R_i = R_{\text{Tot}}$$

Quantification par blocs

- Problème : minimiser D sous contrainte

$$\min_{\mathbf{R}} D(\mathbf{R}) \text{ soumis à } \sum_{i=1}^{N} R_i = R_{\text{Tot}}$$

- Solution : Lagrange

$$J(\mathbf{R}, \lambda) = \sum_{i=1}^{N} h_i \sigma_i^2 2^{-2R_i} + \lambda \left(\sum_{i=1}^{N} R_i - R_{\text{Tot}} \right)$$

Quantification par blocs

- Calcul du gradient :

$$\frac{\partial J}{\partial R_i} = \frac{1}{N} h_i \sigma_i^2 (-2 \ln 2) 2^{-2R_i} + \lambda$$

$$\frac{\partial J}{\partial R_i}(\mathbf{R}^*) = 0$$

$$\lambda = \frac{1}{N} h_i \sigma_i^2 (2 \ln 2) 2^{-2R_i^*}$$

$$R_i^* = \frac{1}{2} \log_2(h_i \sigma_i^2) + \gamma \qquad \gamma = \frac{1}{2} \log_2 \frac{2 \ln 2}{N \lambda}$$

$$R_{\text{Tot}} = \sum_{i=1}^{N} R_i^* = \sum_{i=1}^{N} \frac{1}{2} \log_2(h_i \sigma_i^2) + N\gamma$$

$$\gamma = \frac{R_{\text{Tot}}}{N} - \frac{1}{2} \log_2 h_{\text{GM}} \sigma_{\text{GM}}^2$$

Quantification par blocs

$$R_i^* = \bar{R} + \frac{1}{2}\log_2 \frac{h_i \sigma_i^2}{h_{\mathrm{GM}} \sigma_{\mathrm{GM}}^2}$$

$$D_i^* = h_i \sigma_i^2 2^{-2\bar{R}} \frac{h_{\mathrm{GM}} \sigma_{\mathrm{GM}}^2}{h_i \sigma_i^2} = h_{\mathrm{GM}} \sigma_{\mathrm{GM}}^2 2^{-2\bar{R}}$$

$$D^* = h_{\mathrm{GM}} \sigma_{\mathrm{GM}}^2 2^{-2\bar{R}}$$

Expression de R_i^* : Formule de Huang et Schulteiss

Allocation des ressources

Algorithmes pratiques

La formule de Huang-Schulteiss

- Peut donner des valeurs négatifs
- Peut donner des valeurs fractionnaires

Donc on utilise des algorithmes sous-optimales

- Algorithme de Huang-Schulteiss modifié
- Algorithme *greedy*

Allocation des ressources

Algorithme de Huang-Schulteiss modifié

1. On calcule R_k^* avec Huang-Schulteiss ;
2. Si certaines R_k^* sont négatifs, on répète l'algorithme en enlevant les σ_k^2 concernée ; ces variables seront codées avec zéro bits ;
3. Le pas précédent est répété jusqu'à quand il n'y a plus de valeurs négatifs ;
4. Les valeurs trouvé sont arrondis à l'entier inférieur ;
5. Le débit résiduel est alloué aux coefficients avec l'erreur maximum.

Algorithme *greedy*

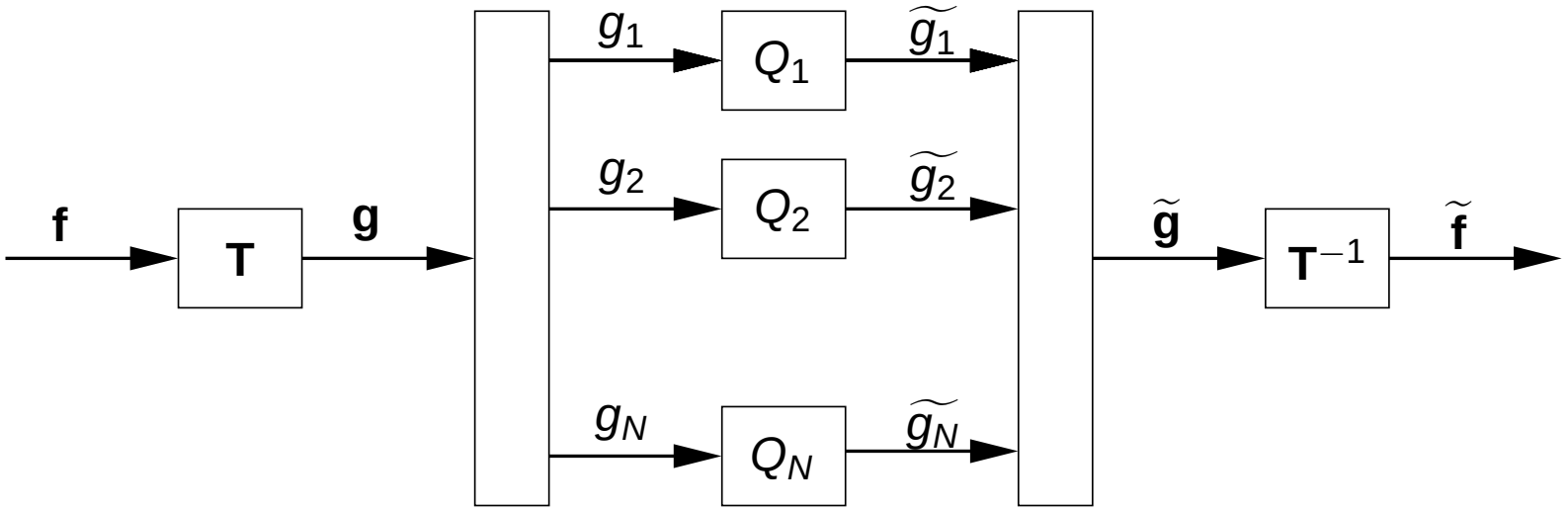
1. Initialisation
   - $R_k = 0 \quad \forall k \in \{0, 1, \ldots M-1\}$.
   - $D_k = \quad \sigma_k^2 \forall k \in \{0, 1, \ldots M-1\}$.
2. Tant que $\sum_k R_k \leq B$
   - $\ell = \arg\max_k D_k$
   - $b_\ell \leftarrow b_\ell + 1$
   - $D_\ell \leftarrow D_\ell / 4$

Principes

- Transformation linéaire : changement de base
- Représentation alternative de l'image
  - Mise en évidence des caractéristiques
  - Séparation des données entre importants et pas importants
  - Déterminer les informations importantes pour le SVH
- Réduire la corrélation
- Allocation des ressources

Transformée 1d

Paradigme du codage par transformée

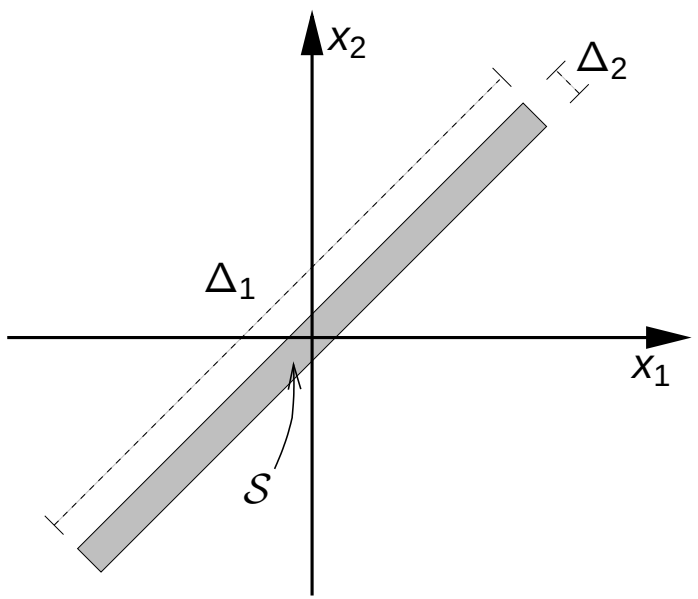


On passe du vecteur **f** à $\mathbf{g} = \mathbf{Tf}$: on veut un vecteur plus "facile" à quantifier : peu de coefficients "importants", beaucoup de coefficients "insignifiants"

Codage par transformée

Exemple

Couple de v.a. fortement corrélées



$$f_{X_1,X_2}(x_1,x_2) = \begin{cases} \frac{1}{\Delta_1 \Delta_2} & \text{si } (x_1,x_2) \in \mathcal{S} \\ 0 & \text{si } (x_1,x_2) \notin \mathcal{S} \end{cases}$$

$$\Delta_1 \gg \Delta_2$$

$$X_1 \sim X_2 \sim \mathcal{U}\left[-\frac{\Delta_1}{2\sqrt{2}}, \frac{\Delta_1}{2\sqrt{2}}\right]$$

Codage par transformée

Exemple

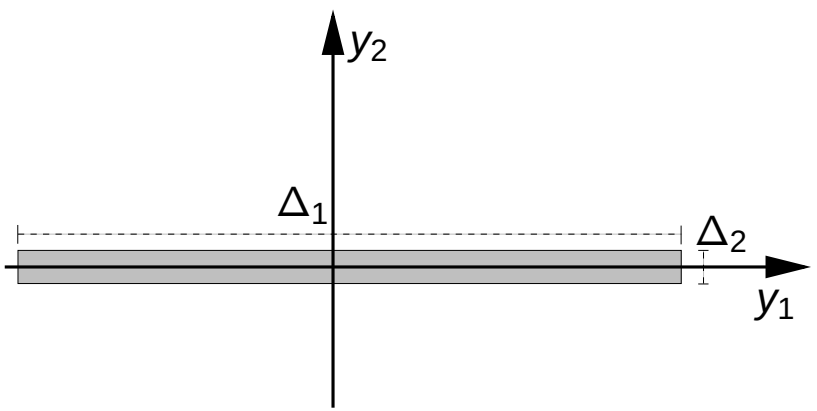
- Quantification de variables uniformes $\Rightarrow$ quantificateur uniforme
- $D_i(R_i) = \sigma_i^2 2^{-2R_i}$ pour chaque v.a.
- $D = \sum_i D_i$
- $\sigma_1^2 = \sigma_2^2 = \sigma^2 = \left(\frac{\Delta_1}{\sqrt{2}}\right)^2 \frac{1}{12} = \frac{\Delta_1^2}{24}$

bits	R	D_1	D_2	D
0	0	σ^2	σ^2	$2\sigma^2$
1	0.5	$\sigma^2/4$	σ^2	$\frac{5}{4}\sigma^2$
1	0.5	σ^2	$\sigma^2/4$	$\frac{5}{4}\sigma^2$
2	1	$\sigma^2/4$	$\sigma^2/4$	$\frac{1}{2}\sigma^2$
2	1	$\sigma^2/16$	σ^2	$\frac{17}{16}\sigma^2$
3	1.5	$\sigma^2/16$	$\sigma^2/4$	$\frac{5}{16}\sigma^2$
4	2	$\sigma^2/16$	$\sigma^2/16$	$\frac{1}{8}\sigma^2$

Codage par transformée

Transformée : rotation de 45 degrés

Après transformation : v.a. indépendantes



$$f_{Y_1,Y_2}(y_1,y_2) = \begin{cases} \frac{1}{\Delta_1 \Delta_2} & \text{si } (y_1,y_2) \in \mathcal{S} \\ 0 & \text{si } (y_1,y_2) \notin \mathcal{S} \end{cases}$$

$$Y_1 \sim \mathcal{U}\left[-\frac{\Delta_1}{2}, \frac{\Delta_1}{2}\right]$$

$$Y_2 \sim \mathcal{U}\left[-\frac{\Delta_2}{2}, \frac{\Delta_2}{2}\right]$$

Codage par transformée

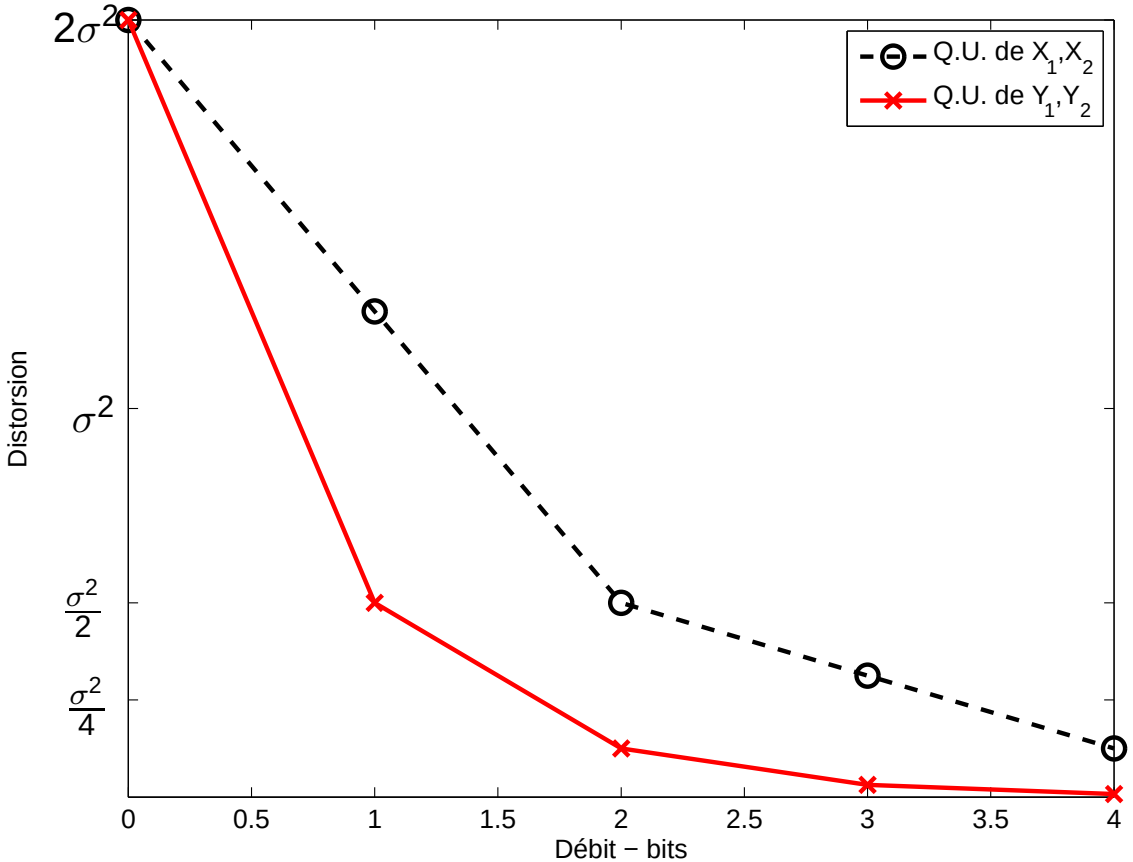
Exemple

- Quantification de Y_1 et Y_2 : courbe $D(R)$
- $\sigma_1^2 = \frac{\Delta_1^2}{12} = 2\sigma^2$
- $\sigma_2^2 = \frac{\Delta_2^2}{12} \ll \sigma^2$

bits	R	D_1	D_2	D
0	0	$2\sigma^2$	$\ll \sigma^2$	$2\sigma^2$
1	0.5	$\sigma^2/2$	$\ll \sigma^2$	$\frac{1}{2}\sigma^2$
2	1	$\sigma^2/8$	$\ll \sigma^2$	$\frac{1}{8}\sigma^2$
3	1.5	$\sigma^2/32$	$\ll \sigma^2$	$\frac{1}{32}\sigma^2$

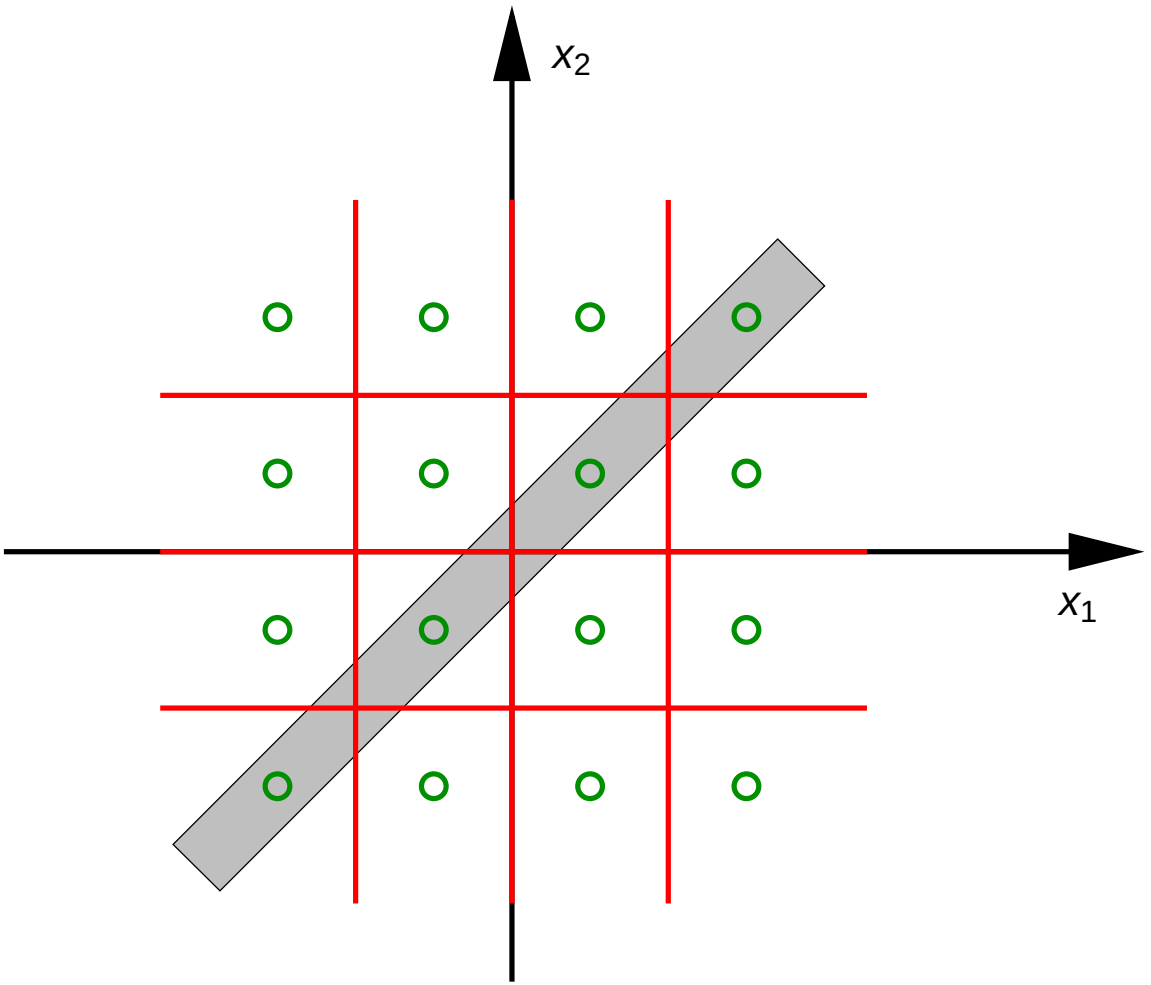
Codage par transformée

Performances RD de la quantification après transformée



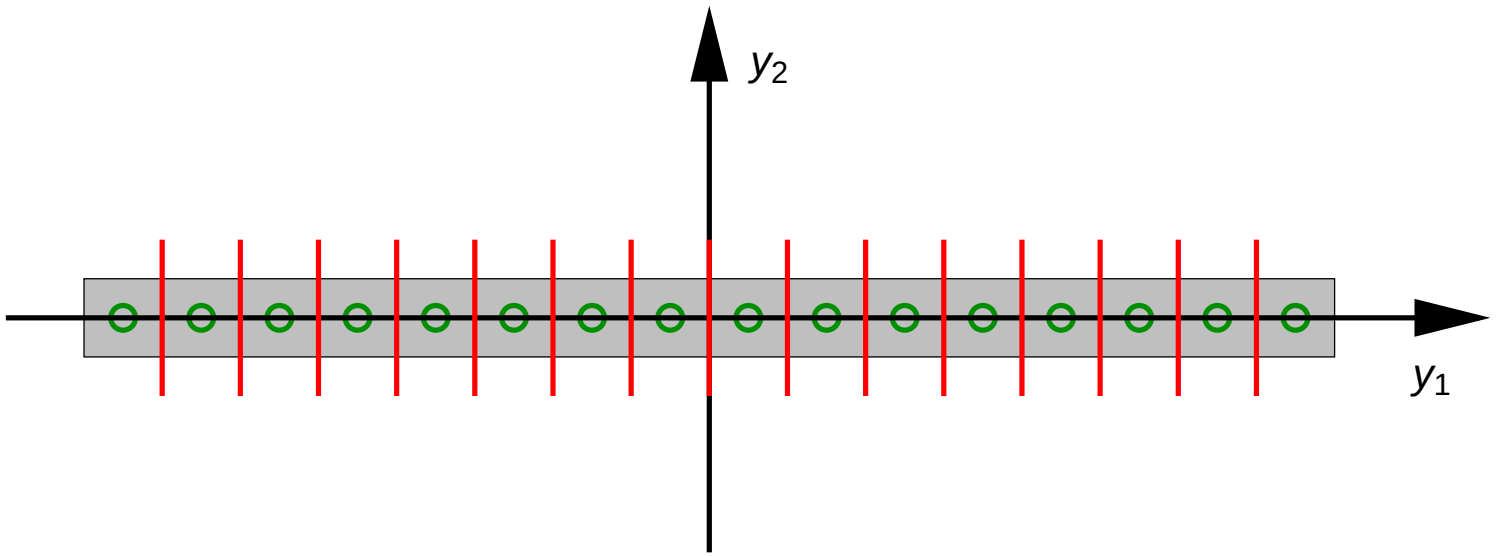
Codage par transformée

Les differentes approches : QS sans transformée



Codage par transformée

Les differentes approches : QS avec transformée



Gain de codage

- Soit **X** un vecteur aléatoire de N données d'entrée (son, image...)
- Hypothèse : les composantes de X sont i.d., p.ex. Gaussiennes $\mathcal{N}(0, \sigma_X^2)$
- Sans transformée, le mieux qu'on puisse faire est quantification et allocation optimale des ressources (PCM). La distorsion est :

$$D_{\mathrm{PCM}} = h_{\mathcal{N}} \sigma_X^2 2^{-2\bar{R}}$$

Gain de codage

On applique à $\mathbf{X}$ une transformée orthogonale A :

$$\mathbf{Y} = A\mathbf{X}$$

La distorsion calculée sur Y est égale à la distorsion qu'on obient une fois la transformée inverse appliquée

$$\frac{1}{N}\mathrm{E}\left[\|\mathbf{Y} - \widehat{\mathbf{Y}}\|^2\right] = \frac{1}{N}\mathrm{E}\left[\|A\mathbf{X} - A\widehat{\mathbf{X}}\|^2\right]$$
$$= \frac{1}{N}\mathrm{E}\left[(\mathbf{X} - \widehat{\mathbf{X}})^T A^T A(\mathbf{X} - \widehat{\mathbf{X}})\|^2\right] = \frac{1}{N}\mathrm{E}\left[\|\mathbf{X} - \widehat{\mathbf{X}}\|^2\right]$$

où $\widehat{\mathbf{X}} = A^T\mathbf{Y}$ est la version "décodée" de $\widehat{\mathbf{Y}}$

Cette distorsion depend de la transformée par le biais des variances et des facteurs de forme

Gain de codage

La distorsion minimale sur Y est

$$D_A = h_{Y,\mathrm{GM}} \sigma^2_{Y,\mathrm{GM}} 2^{-2\bar{R}}$$

Cette distorsion depend de la transformée par le biais des variances et des facteurs de forme
Si **X** est n vecteur gaussien, **Y** est aussi gaussien et

$$D_A = h_{\mathcal{N}} \sigma^2_{Y,\mathrm{GM}} 2^{-2\bar{R}}$$

Gain de codage

On observe que :

$$\sigma^2_{Y,\mathrm{AM}} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{E}[Y_i^2] = \frac{1}{N}\mathrm{E}[\|\mathbf{Y}\|^2] = \frac{1}{N}\mathrm{E}[\|\mathbf{X}\|^2]$$

$$= \frac{1}{N}\sum_{i=1}^{N} \mathrm{E}[X_i^2] = \sigma^2_{X,\mathrm{AM}} = \sigma^2_X$$

Une transformée orthogonale ne modifie pas la *moyenne arithmetique* des variances.

Gain de codage

- Le *gain de codage* d'une transformée $\mathcal{T}$ est défini comme le rapport entre la distorsion qu'on aurait sans transformée, et la distorsion qu'on peut atteindre avec la transformée :

$$G = \frac{D_{\mathrm{PCM}}}{D_A} = \frac{\sigma_X^2}{\sigma_{Y,\mathrm{GM}}^2} = \frac{\sigma_{Y,\mathrm{AM}}^2}{\sigma_{Y,\mathrm{GM}}^2}$$

- La transformée doit rendre les variances des composantes du vecteur Y les plus inégales possible
- La moyenne géométrique d'un ensemble de nombres positifs est toujours inférieur ou égal à la moyenne arithmétique

Transformations linéaires en 2D

Transformées basées sur la décomposition fréquentielle

- Asymptotiquement équivalentes ($N, M \to \infty$)

TFD

$$t(k,\ell,n,m) = \frac{1}{\sqrt{N\,M}} \exp\left[-\imath 2\pi\left(\frac{(n-1)(k-1)}{N} + \frac{(m-1)(\ell-1)}{M}\right)\right]$$

TCD

$$t(k,\ell,n,m) = \frac{c_k\,c_\ell}{\sqrt{N\,M}} \cos\left(\pi\frac{(2n-1)(k-1)}{2N}\right)\cos\left(\pi\frac{(2m-1)(\ell-1)}{2M}\right)$$

$$c_k = \begin{cases} 1 & \text{si } k = 1 \\ \sqrt{2} & \text{sinon} \end{cases}$$

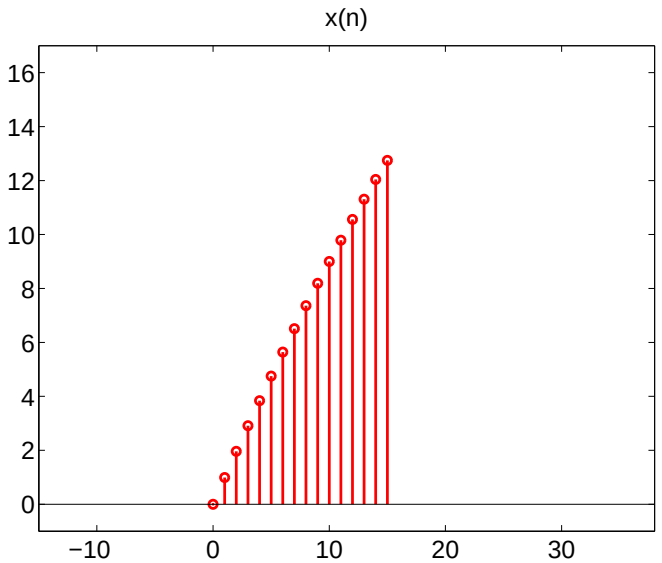
Transformations linéaires en 2D

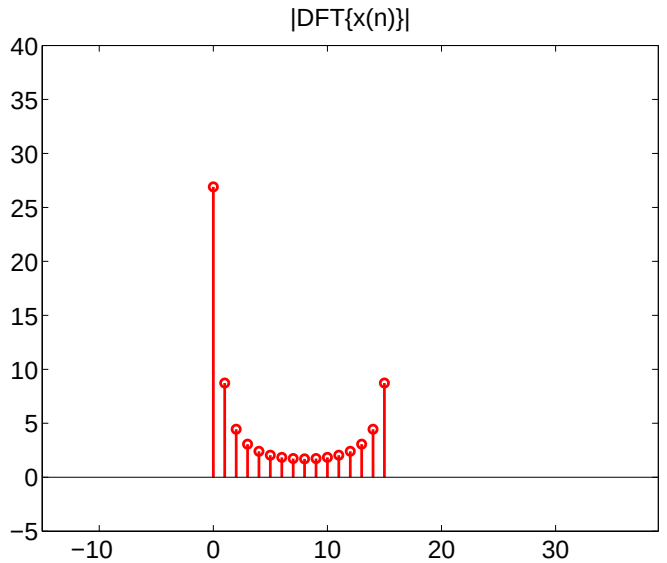
Transformées basées sur la décomposition fréquentielle

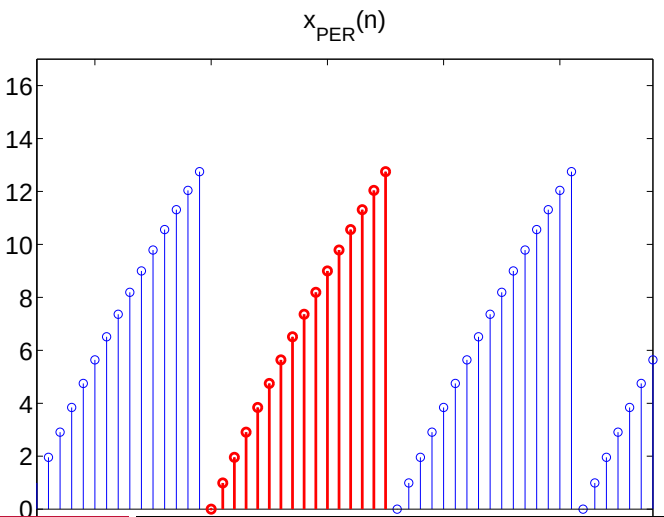
TCD et TFD

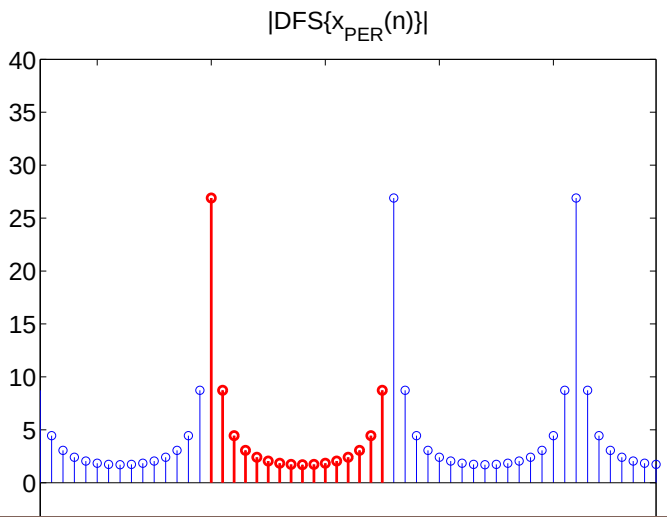
- caractéristiques communes
  - séparables : $t(k,\ell,n,m) = t_1(k,n)\,t_2(\ell,m)$
  - algorithmes de calcul rapides
  - interprétations fréquentielles
    k : indice de fréquence verticale
    ℓ : indice de fréquence horizontale
- spécificités de la TCD
  - réelle
  - meilleure concentration de l'information que la TFD
- existence d'autres transformations plus simples mais moins efficaces

DFT vs DCT

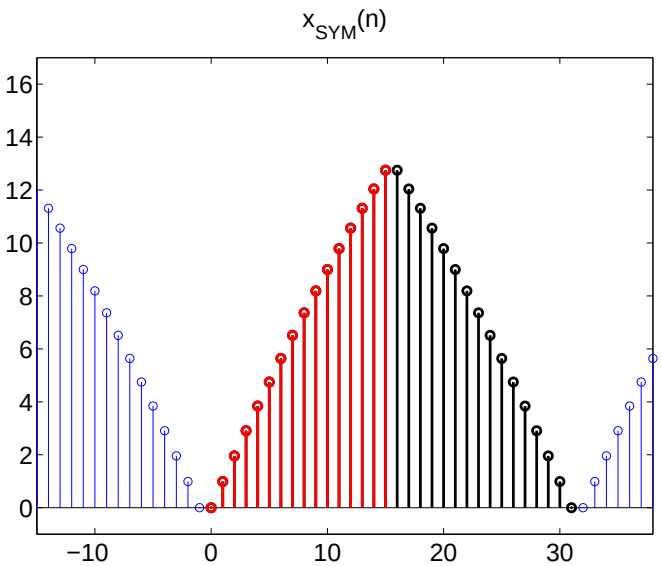

x(n)
16
14
12
10
8
6
4
2
0
−10
0
10
20
30

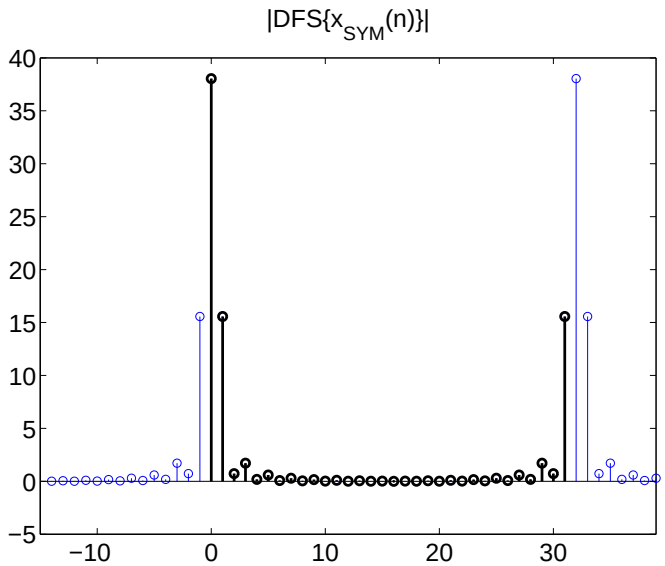

|DFT{x(n)}|
40
35
30
25
20
15
10
5
0
−5
−10
0
10
20
30

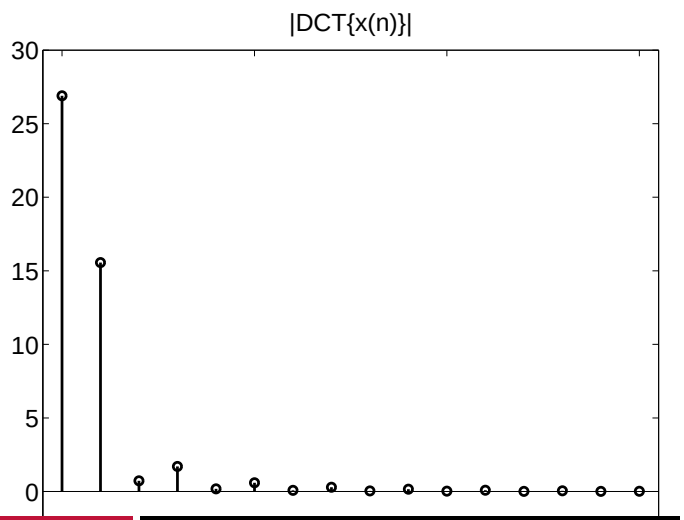

x_PER(n)
16
14
12
10
8
6
4
2
0

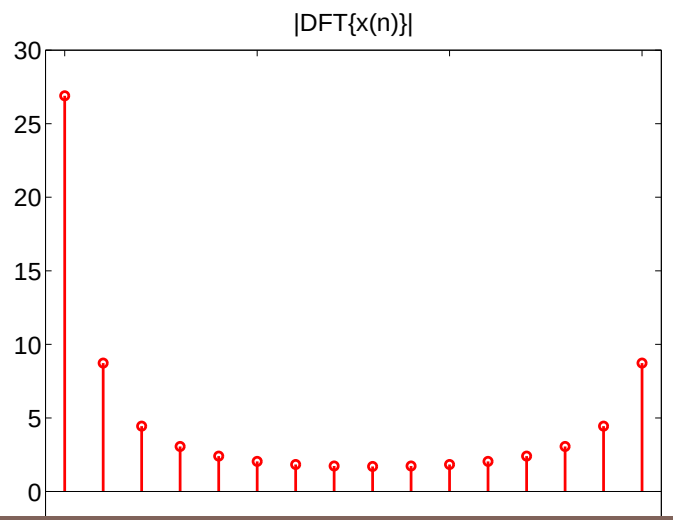

|DFS{x_PER(n)}|
40
35
30
25
20
15
10
5
0

DFT vs DCT


$x_{SYM}(n)$
16
14
12
10
8
6
4
2
0
−10
0
10
20
30


|DFS{$x_{SYM}(n)$}|
40
35
30
25
20
15
10
5
0
−5
−10
0
10
20
30

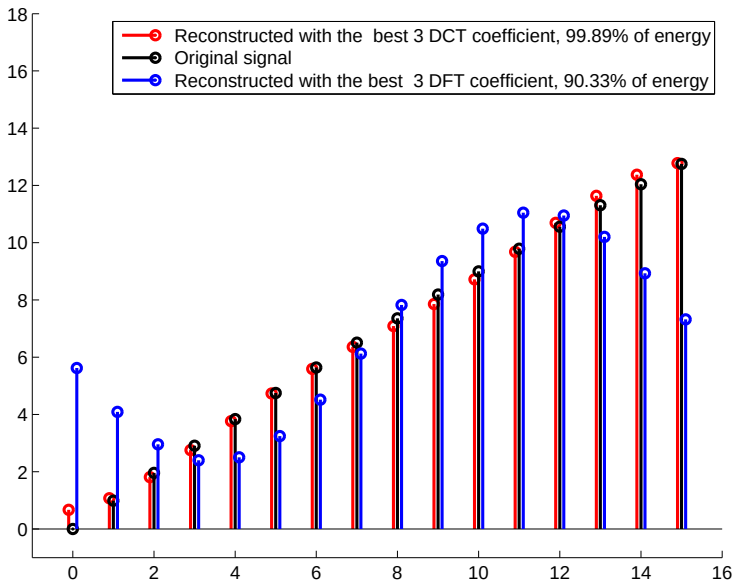

|DCT{x(n)}|
30
25
20
15
10
5
0


|DFT{x(n)}|
30
25
20
15
10
5
0

DFT vs DCT

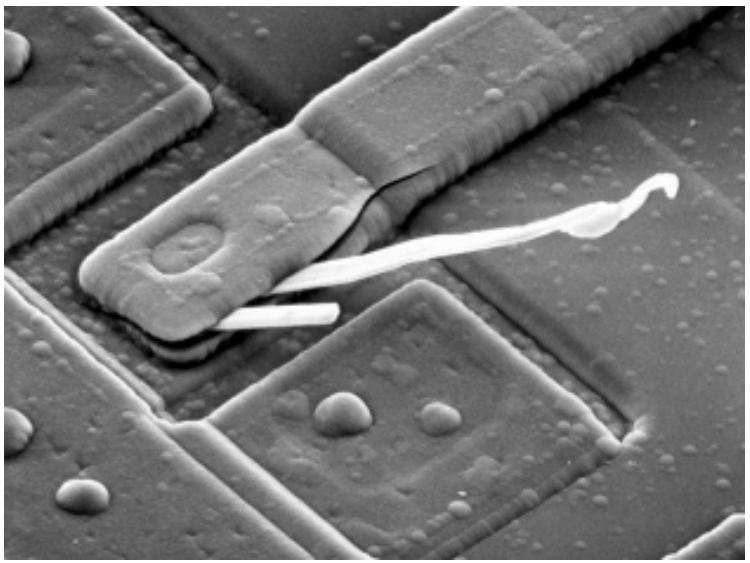
DCT	
Coeff	% Energy
1	74.61
2	24.98
3	0.05
4	0.30
5	0.00
6	0.04
7	0.00
8	0.01
9	0.00
10	0.00
11	0.00
12	0.00
13	0.00
14	0.00
15	0.00
16	0.00

DFT	
Coeff	% Energy
1	74.61
2	7.86
3	2.04
4	0.97
5	0.60
6	0.43
7	0.35
8	0.31
9	0.30
10	0.31
11	0.35
12	0.43
13	0.60
14	0.97
15	2.04
16	7.86

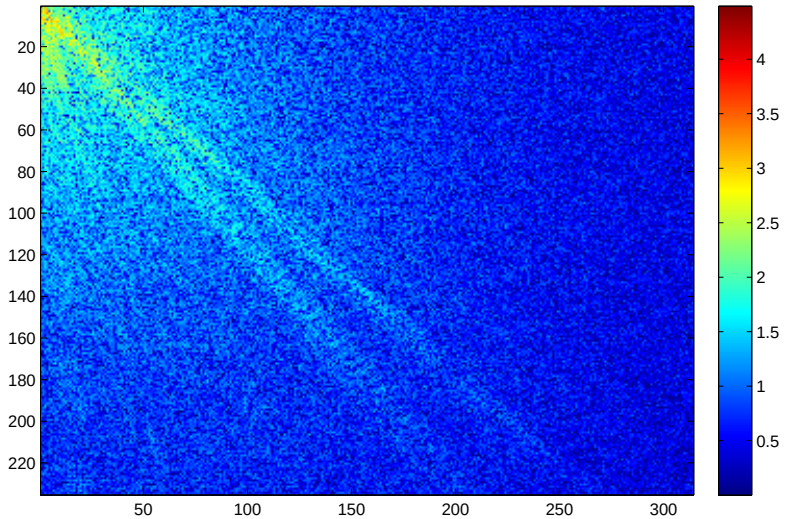


Exemples de TCD

Imagerie MEB (microscopie électronique à balayage)



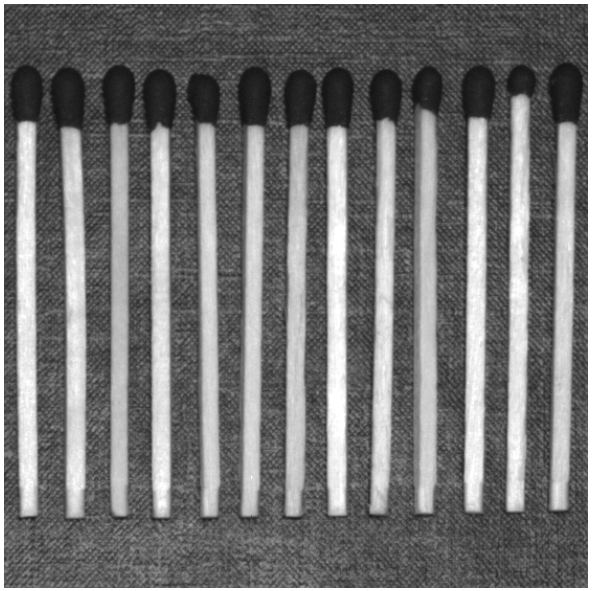
Image

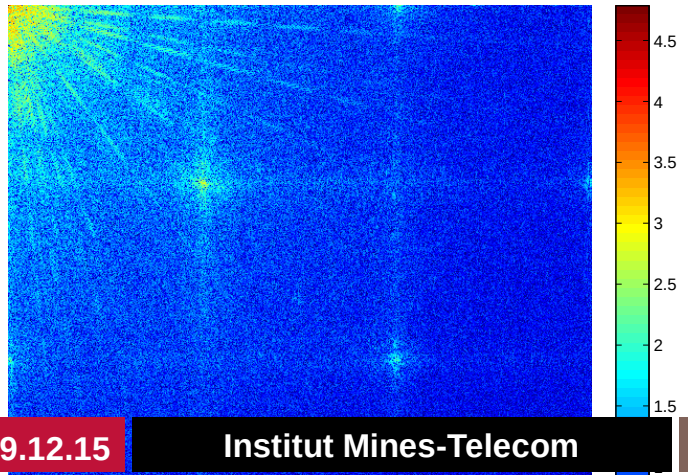


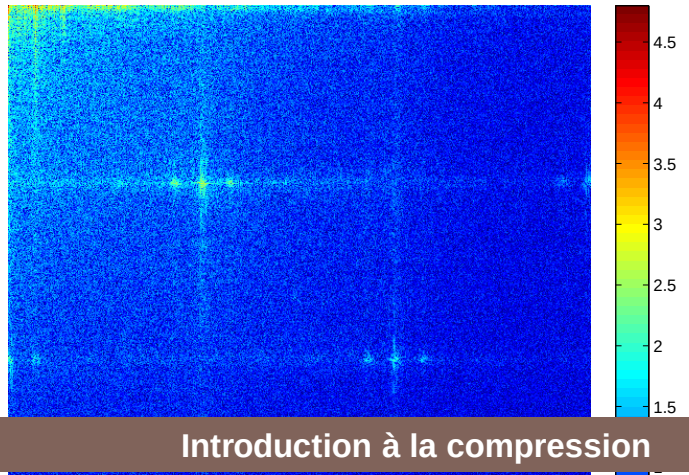
Logarithme des coefficients TCD

Exemples de TCD

Fréquences spatiales

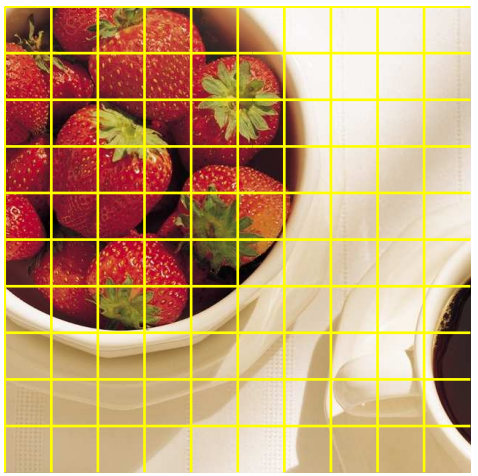


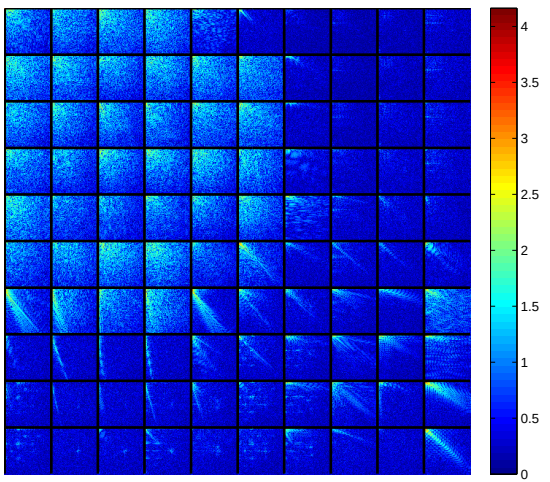




TCD par blocs

- image non stationnaire et de taille élevée
  $\Rightarrow$ découpage en $I \times J$ blocs rectangulaires $(\mathcal{B}_{i,j})_{0 \leq i < I, 0 \leq j < J}$, de taille $K \times L$ ($K = L = 8$)



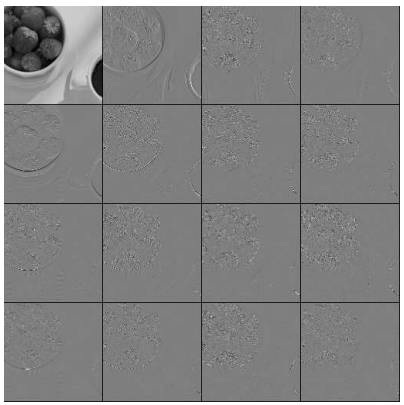


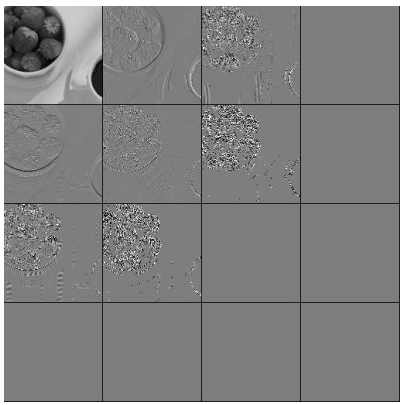
$$N = I\,K \qquad M = J\,L$$

Allocation des ressource pour les coefficients TCD

- Formule de Huang-Schulteiss
- Algorithme Greedy
- Allocation fixe (indépendante des données)
  - Solution très simple
  - Permet de prendre en compte les caractéristique psycho-visuelles

Allocation des ressource pour les coefficients TCD





sb	variance	lev	rate	dist
1	141.35	32	3.90	0.86
2	32.92	15	2.74	0.98
3	6.80	5	1.50	1.24
4	1.77	1	0.00	1.77
5	38.07	15	2.74	1.14
6	10.27	7	1.81	1.12
7	3.89	3	1.15	1.41
8	1.19	1	0.00	1.19
9	7.47	5	1.50	1.37
10	3.76	3	1.15	1.37
11	2.07	1	0.00	2.07
12	0.71	1	0.00	0.71
13	1.95	1	0.00	1.95
14	1.27	1	0.00	1.27
15	0.74	1	0.00	0.74
16	0.32	1	0.00	0.32

Norme de codage JPEG

- Norme de compression d'images basée sur la TCD
- Spécifiée en 1991, adoptée en 1992
- Normalise l'algorithme et le format de *décodage*
- On va parler de un codeur à niveaux de gris, qui produise un train binaire conforme

Norme JPEG : Schéma

- L'image est préalablement découpée en blocs 8×8
- On soustrait 128 aux valeurs de luminance
- Les blocs sont codés indépendamment

Norme JPEG : Découpage par blocs

Exemple de bloc 8x8

173	171	171	143	109	100	91	96
171	169	150	137	112	101	94	96
184	158	139	120	110	107	94	100
170	156	134	119	117	104	98	99
157	147	125	127	103	109	90	98
149	146	132	120	113	107	101	93
147	141	119	119	111	101	100	92
160	122	117	116	115	116	102	95

Norme JPEG : Transformée

- La taille de la TCD est 8×8
- Petits blocs $\rightarrow$ signal stationnaire
- Grands blocs $\rightarrow$ exploit de la corrélation
- Taille choisie après des expériences
- Coefficients TCD : impact SVH

Norme JPEG : Transformée

Coefficients TCD du bloc considérée

985.3	186.2	34.1	11.6	7.3	1.6	4.9	-8.2
40.3	47.8	5.7	-26.0	-5.3	-3.5	4.0	-1.0
6.3	4.0	-9.3	-6.7	-1.2	8.1	3.4	4.1
-0.0	4.9	-13.3	-20.8	-10.4	-1.0	-4.5	-5.1
2.1	-1.3	-1.6	0.6	3.6	3.3	8.1	-1.7
1.3	3.7	2.4	-2.7	-2.2	-3.0	-4.1	7.8
5.1	0.4	3.1	4.8	-1.4	2.5	9.8	5.3
-5.6	1.6	4.4	0.1	3.3	2.3	4.3	-8.4

Norme JPEG : Transformée

Écart-type des coefficients TCD d'une image naturelle

396.64	100.99	49.26	31.15	19.74	14.57	8.76	7.33
100.23	55.78	37.40	24.77	16.44	11.70	8.44	6.25
49.42	36.39	28.01	20.40	14.64	10.46	7.64	5.88
30.82	24.05	19.73	15.47	11.99	8.88	6.83	5.45
21.09	16.79	14.79	11.54	9.19	7.30	5.90	4.68
15.32	11.91	10.31	8.71	7.15	5.78	4.61	3.91
11.22	8.58	7.66	6.78	5.69	4.64	3.82	3.24
8.21	6.65	5.93	5.52	4.45	3.75	3.15	2.80

Norme JPEG : Quantification

- Quantification uniforme à zone morte
- $\widetilde{c}_{i,j} = \left\lfloor \frac{c_{i,j}}{q_{i,j}} \right\rfloor$
- Le compromis débit distorsion est complètement géré par le tableau de quantification q
- Le standard ne spécifie pas q, qui doit être transmis
- Facteur de qualité Q

Norme JPEG : Quantification

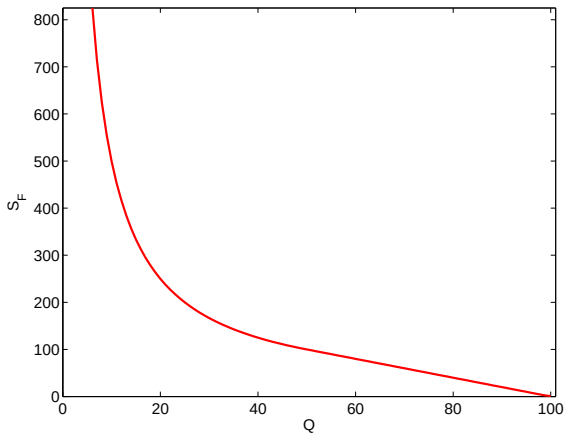
Exemple de table de quantification

$$q^* = \begin{array}{|c|c|c|c|c|c|c|c|} \hline 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ \hline 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ \hline 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ \hline 14 & 17 & 22 & 29 & 51 & 87 & 81 & 61 \\ \hline 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ \hline 24 & 35 & 55 & 64 & 81 & 104 & 111 & 90 \\ \hline 49 & 63 & 78 & 87 & 101 & 121 & 120 & 100 \\ \hline 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \\ \hline \end{array}$$

Norme JPEG : Facteur de qualité

- Outil **non** normatif
- "Facteur de qualité" Q variable entre 1 et 100
- Définit un facteur d'échelle S_F pour la matrice de quantification

$$S_F = \begin{cases} \frac{5000}{Q} & 1 \leq Q \leq 50 \\ 200 - 2Q & 50 < Q \leq 99 \\ 1 & Q = 100 \end{cases}$$



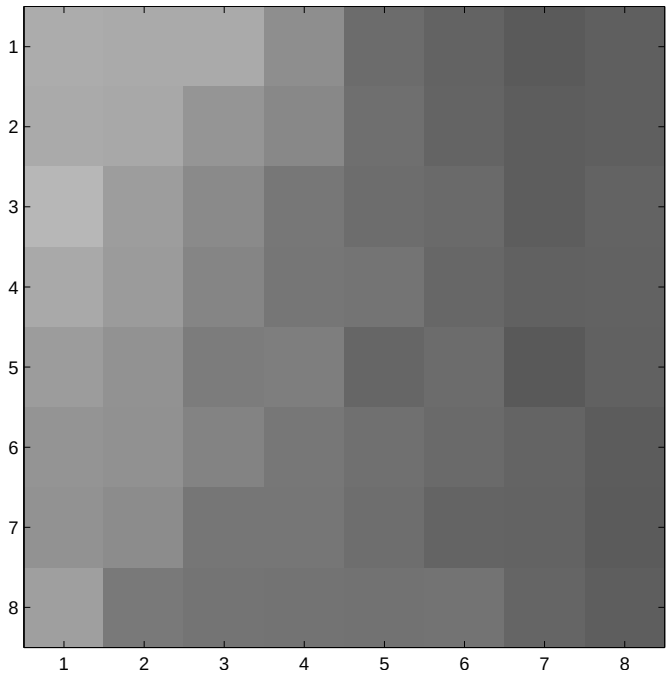
$$q \leftarrow \frac{S_F q^* - 50}{100}$$

Norme JPEG : Quantification

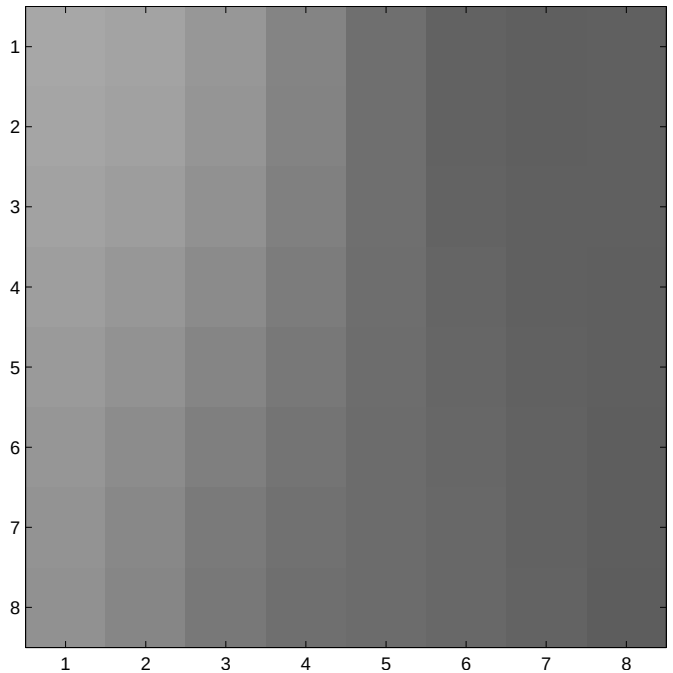
Coefficients quantifiés avec la table précédente

61	16	3	0	0	0	0	0
3	3	0	-1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Norme JPEG : Exemple

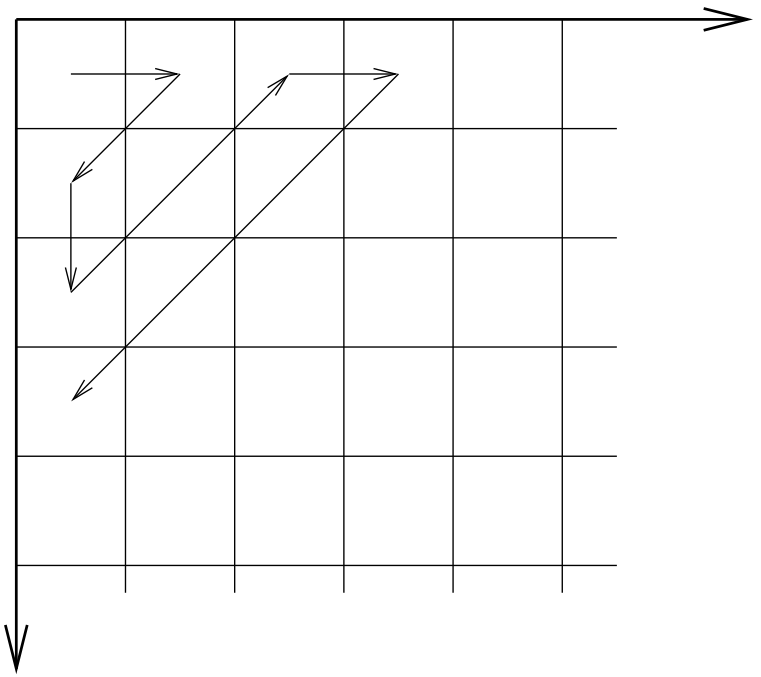


Original



$\mathrm{TCD} \rightarrow Q \rightarrow Q^{-1} \rightarrow \mathrm{TCD}^{-1}$

Norme JPEG : Codage sans perte



- Zig-zag scan : concentre les coefficients nuls à la fin du balayage

Norme JPEG : Codage sans perte

- Coefficient DC : codage prédictif + Huffman
- Coefficients AC : codage "run-lenght" + Huffman

coeff$\neq$ 0	n. de 0	coeff$\neq$ 0	n. de 0	...	EOB	...

Norme JPEG : Codage sans perte

Coefficients quantifiés et codés

61	16	3	0	0	0	0	0
3	3	0	-1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

61-dc_{k-1}	0,16	0,3	1,3	0,3	7,-1	EOB

Norme JPEG : Codage sans perte

Coefficients DC :

- représentés par la couple catégorie, amplitude.
- Il y a 12 catégories, nommées $0, \ldots, 11$, codée sur 4 bit
- La catégorie k contient 2^k valeurs : $\{\pm 2^{k-1}, \ldots, \pm 2^k - 1\}$; chaque valeur est codée sur k bits

Coefficients AC :

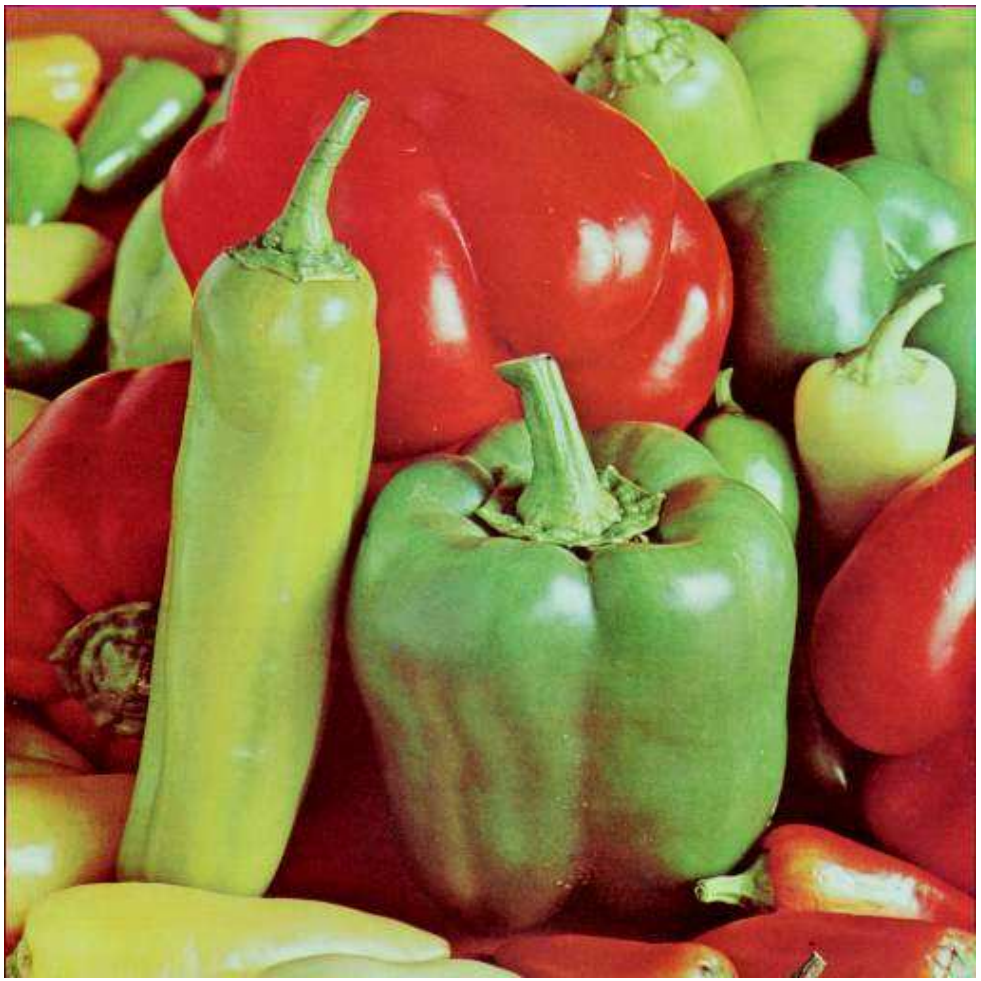
- représentés par run-length, catégorie, amplitude.
- Les catégories et les run-lengths sont codée sur 4 bit chacune : 8 bits pour le symbole (R,C)
  - Symbole spécial 1: (15,0) signifie "au moins 15 zéros avant le prochain coefficient non nul"
  - Symbole spécial 2: (0,0) signifie "fin du bloc"
- Comme dans le cas DC, la catégorie k contient 2^k valeurs, chacune codée sur k

Norme JPEG : Exemple de codage

Image Originale, 24 bpp

Norme JPEG : Exemple de codage

Débit 1.0198674.2 bpp PSNR 33.92 dB TC 23.532



Norme JPEG : Exemple de codage

Débit 0.7481384.2 bpp PSNR 33.45 dB TC 32.080

Norme JPEG : Exemple de codage

Débit 0.5017404.2 bpp PSNR 32.70 dB TC 47.834

Norme JPEG : Exemple de codage

Débit 0.3081364.2 bpp PSNR 31.31 dB TC 77.888

Norme JPEG : Exemple de codage

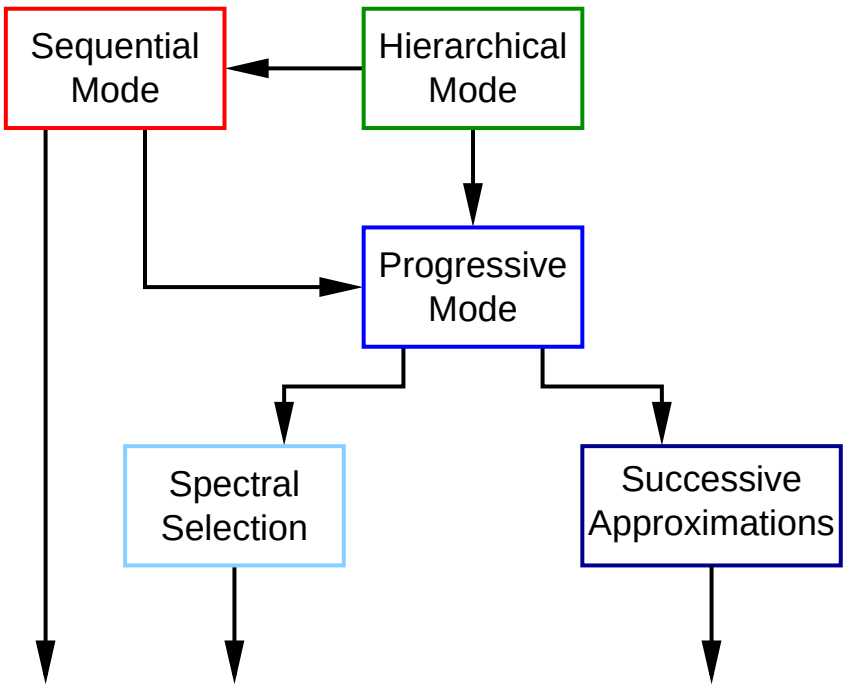
Débit 0.2069704.2 bpp PSNR 29.50 dB TC 115.959

Norme JPEG : Parties

- JPEG baseline (sequential DCT mode)
- JPEG progressif
- JPEG hiérarchique
- JPEG sequential lossless mode
- JPEG partie 3
  - Variable quantization
  - Tiling
- Standard JPEG-LS
- Motion JPEG

Modalités de JPEG

Les modalités de JPEG peuvent être utilisées conjointement



JPEG progressif

- Représentation progressive des images
- Application : Web, bases de données, ...
- Deux sous-modalités : *Spectral selection* et *Successive approximations*

JPEG progressif

Spectral selection

Exemple avec quatre couches de qualité :

1. Le coefficient DC de tout bloc
2. Le premiers trois coefficient AC de chaque bloc (ordre du zig-zag scan)
3. Le coefficient AC de 4 à 7 de chaque bloc
4. Le coefficient AC restants

Codage Run-length suivi de Huffman, comme dans le Baseline
En effet la syntaxe JPEG permet de définir des couches arbitraires, avec la seule contrainte de avoir coefficients consécutifs en chaque couche
Qualité identique à JPEG baseline
Débit faiblement augmenté

JPEG progressif

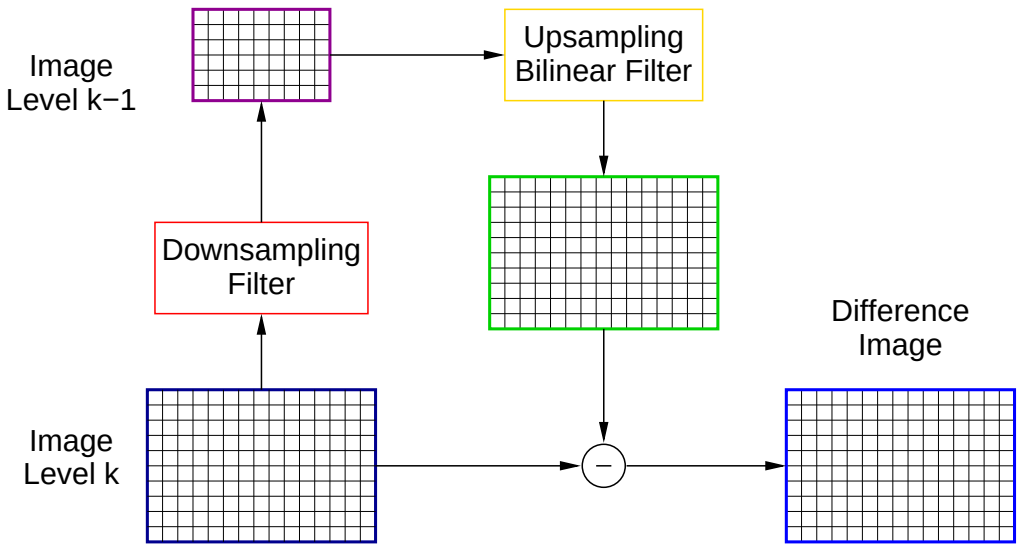
Successive approximation

- Première couche : coefficient DC de chaque bloc (comme en JPEG-SS)
- Couches successives : codage par bit-plane des coefficients
  - Matrice de 8×8 coefficients représentés sur 16 bits $\Rightarrow$ 16 Matrices de 8×8 bits
  - Codage entropique des plains de bit
- Complexité augmentée
- Performance débit-distorsion faiblement améliorée

Il est possible de coder en mode SA les groupes de coefficients définis en mode SS

JPEG hiérarchique

On définit une hiérarchie d'images à résolutions décroissantes (schéma pyramidale)



Chaque image peut être codée en JPEG baseline ou progressif.

Plan

Compression d'images

Compression audio
- L'audition
- Normes

Signal de parole

- Signal non-stationnaire, mais localement stationnaire (20 ms)
- Types de son : voisés (voyelles, consonnes sonores, vibrantes, nasales) non-voisés (consonnes sourdes), autre (transition entre phonèmes)
- Modèles de prédictions simples et efficaces :
  - Filtrage linéaire (autorégressif) d'une suite d'impulsions pour les sons voisés
  - Même filtre appliqué sur du bruit blanc pour les sons non-voisés
  - Ce système reproduit les caractéristiques du trait vocal

Signal de parole

Numérisation

- PCM (Pulse Code Modulation) / MIC (Modulation par Impulsions Codées)
  - Bande 200 Hz $\div$ 3400 Hz
  - Suffisante pour l'intelligibilité de la parole
  - Échantillonement a 8000 Hz : $F_s > 2F_{max}$
  - Représentation de chaque échantillon sur 8 bits
  - Cela donne 64 kbps
- Parole en bande élargie
  - Introduction 50-200 Hz : voix plus naturelle, amélioration de l'effet de présence
  - Extension 3.4-7 kHz : plus grande intelligibilité

Signal de musique

- Variations de puissance importantes (dynamique de 90dB)
- Signal localement stationnaire
- Pas de modèles simples

Normes de compression

Parole : Réseau téléphonique commuté

- G.711 (1972) PCM (pas de compression), 8 échantillons par ms codés sur 8 bits : 64 kbps
- G.721 (1984) Codage ADPCM (prédiction par filtre linéaire) : 32 kbps
- G.728 (1991) Codeur de type CELP (Code Excited Linear Predictive), avec faible délai de décodage : 16 kbps
- G.729 (1995) Codeur de type CELP, sans contrainte sur le retard : 8kbps
- G.723.1 (1995) Codeur à 6.3 kbps, pour visiophone

Normes de compression

Parole : Communication mobile

GSM 06.10 (1988) RPE-LTP : Regular Pulse Excitation Long Term Predictor. 13 (22.8) kbps

GSM 06.20 (1994) "Half-Rate". Débit de 5.6 (11.4) kbps

GSM 06.60 (1996) "ACELP". Débit de 12.2 (22.8) kbit/s

GSM 06.90 (1999) Codage source/canal à débit variable 4.75÷12.2 (11.4÷22.8)kbps (ACELP-AMR : Adaptive Multi Rate)

G.722 Codeur de parole en bande élargie, débit entre 24 et 56 kbps (AMR-WB)

- Orange et SFR utilisent ce codeur depuis octobre 2012 sur le réseau 3G+
- Utilisé aussi en UK, Espagne, Belgique

Normes de compression

Musique

Format CD : échantillonnage à 44.1 kHz, quantification à 16 bits par échantillon : 705 kbps en mono

MP3 : Il s'agit de la partie audio du standard MPEG-1. Trois couches, de qualité équivalente et de complexité croissante, à 192, 128 et 96 kbps

AAC : Partie audio de MPEG-2. Il est réputé le codeur plus performant à l'heure actuelle, avec une qualité "transparente" à 64 kbps

MPEG-4 : Représentation de sons d'origine quelconque (naturelle et synthétique), représentation des objet sonores.

Normes de compression

Musique

Dans MPEG-4 on a plusieurs codeurs :

- Harmonic Vector eXcitation Coding (HVXC), signal de parole en bande téléphonique, débits entre 2 et 4 kbps
- CELP, signal de parole en bande téléphonique ou élargie
- Pour le signal de musique, une nouvelle version du AAC, pas très différentes de celui-ci
- Un codeur sans pertes (compression ≈ 2)
- Un algorithme de synthèse de la parole, un langage pour engendrer de la musique, un langage pour la description d'une scène audio

Approche de codage *par objet* : une scène audio peut être décomposée en plusieurs objets audio, chacun codé avec le codeur le plus adapté

Évaluation de la qualité

- Les critères objectives (fonctions mathématiques comme l'EQM) ne sont pas satisfaisants : pour un même niveau de bruit le résultat peut être très différent (forme spectrale du bruit, masquage)
- Tests subjectifs :
  - Codeurs de parole : tests d'intelligibilité
  - Codeurs de musique : critère de "transparence". Méthode doublement aveugle à triple stimulus et référence dissimulée (Norme UIT-T BS.1116)
  - Codeurs de musique à qualité intermédiaire : autres tests subjectifs (Norme UIT-T BS.1534-1)
  - Quelques test objectif donne des résultats significatifs (Norme UIT-T BS.1387-1)

Audition

L'oreille

- Oreille externe (pavillon, conduit auditif)
- Oreille moyenne (chaîne ossiculaire, tympan)
- Oreille interne (cochlée: 3,5cm; membrane basilaire)

Oreille externe et oreille moyenne : filtre passe-bande (20Hz ÷ 20kHz)
La membrane basilaire est dense ment innervée

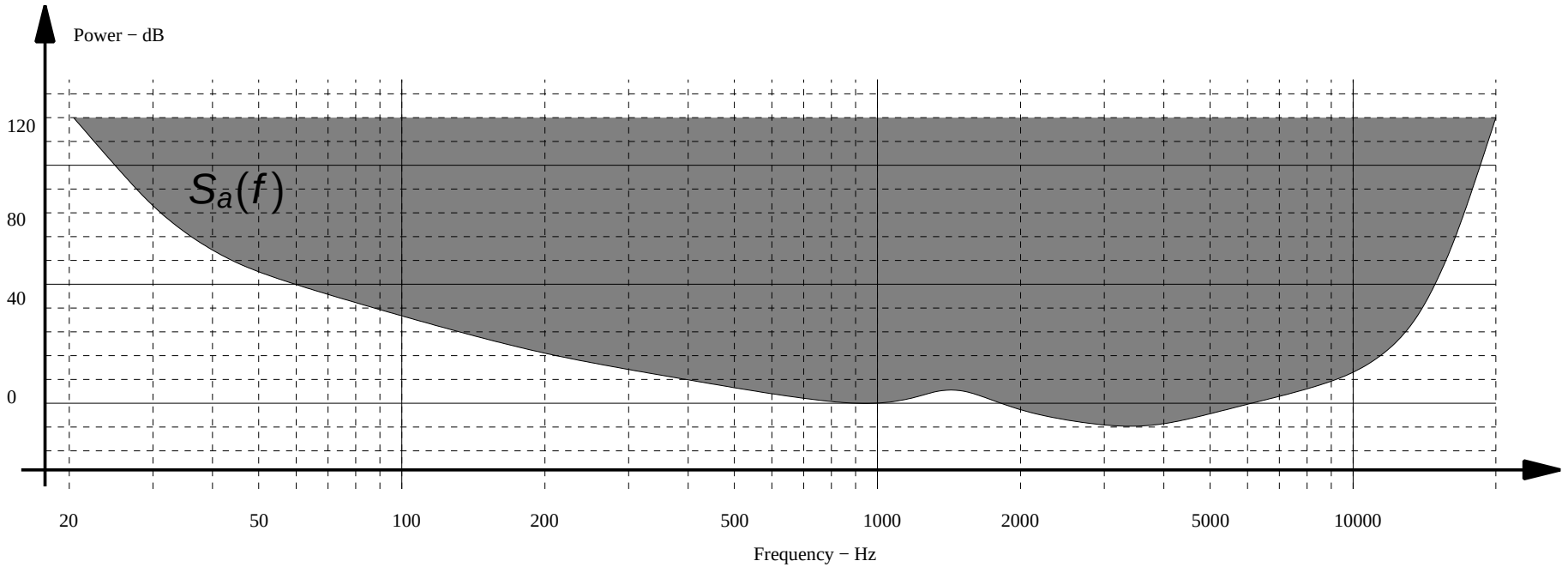
Perception d'un son pur

- Son pur : $x(t) = a\sin(2\pi f_1 t)$ sinusoïde de puissance $\sigma^2 = \frac{a^2}{2}$
- Ce son excite plusieurs fibres nerveuses (étalement de la puissance)
- Modèle : banc de M filtres
  - Le k filtre correspond à la k-ème fibre nerveuse
  - La réponse en fréquence du k-ième filtre est $H_k(f) = A_k(f)\exp^{j\phi_k(f)}$
  - La réponse à la sinusoïde à fréquence f_1 est :

$$y_k(t) = aA_k(f_1)\sin\left[2\pi f_1 t + \phi_k(f_1)\right]$$

- Le rapport entre les puissances est la fonction d'étalement : $S_E(k) = A_k^2(f_1)$

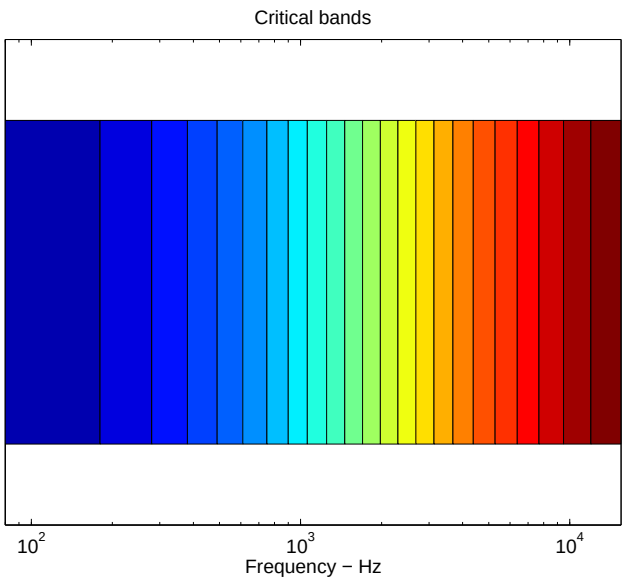
Seuil d'audition



- La gamme de fréquence audible est comprise entre 20Hz et 20kHz
- La puissance minimale nécessaire pour que le son soit audible est $S_a(f)$
- $S_a(f)$ varie avec la fréquence et a un minimum entre 1 et 4kHz (parole)

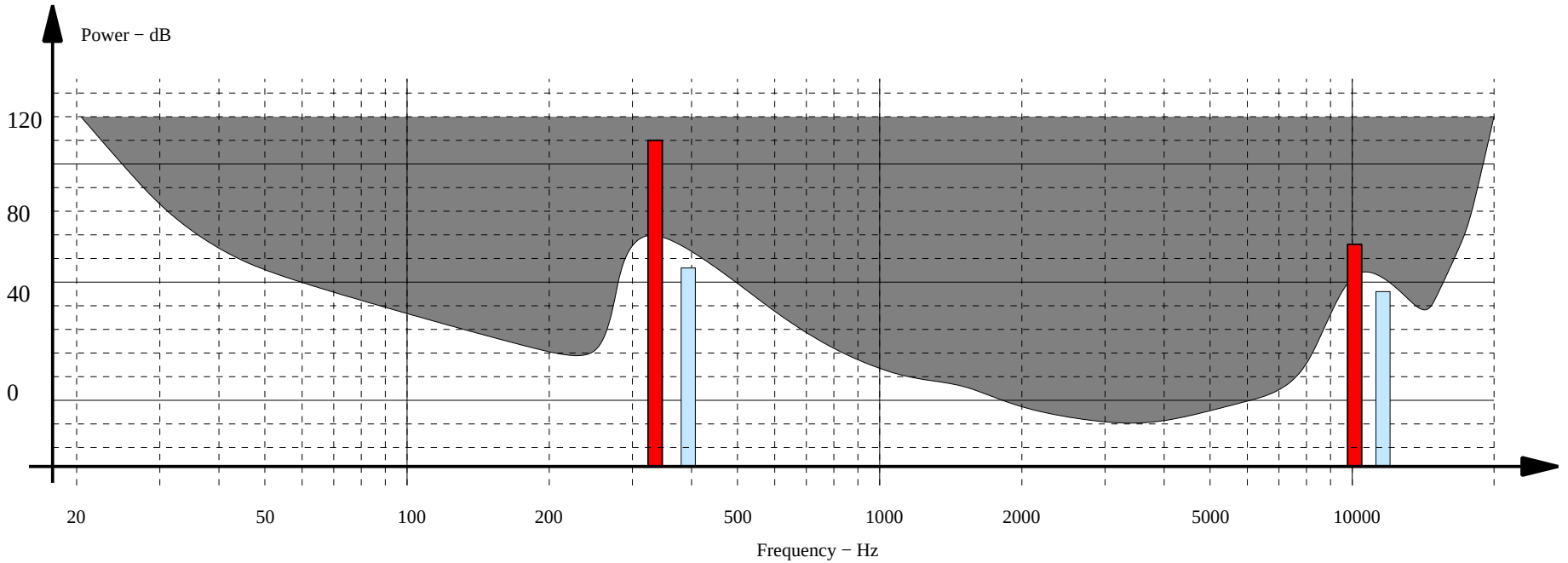
Bande critique (BC)

- Une sinusoïde de fréquence f_1 doit avoir puissance $\sigma_1^2 > S_a(f_1)$ pour être audible
- Pour N sinusoïdes de fréquence *proche à* f_1 il suffit que $\sum_i \sigma_i^2 > S_a(f_1)$
- Les sinusoïdes sont *proches* si sont dans la *bande critique*
- L'amplitude de la BC varie avec f_1



Courbes de masquage

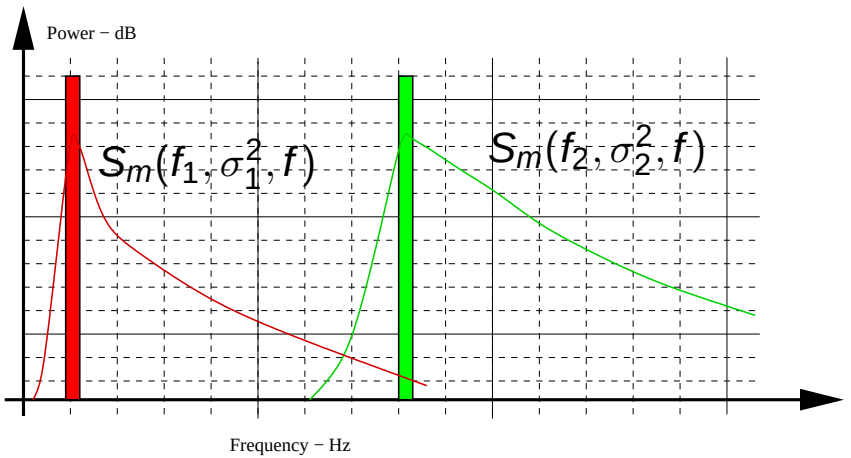
Masquage fréquentiel



- Le son masquante (rouge) réduit la sensibilité à un deuxième son
- On définit $S_m(f_0, \sigma^2, f)$ la puissance minimale pour un son pur a fréquence f pour ne pas être masque par un son pur à f_0 et de puissance σ^2, avec $\sigma^2 > S_a(f_0)$
- La même courbe est valable pour du bruit à bande étroite

Fonction de masquage fréquentiel

$S_m(f_0, \sigma^2, f)$

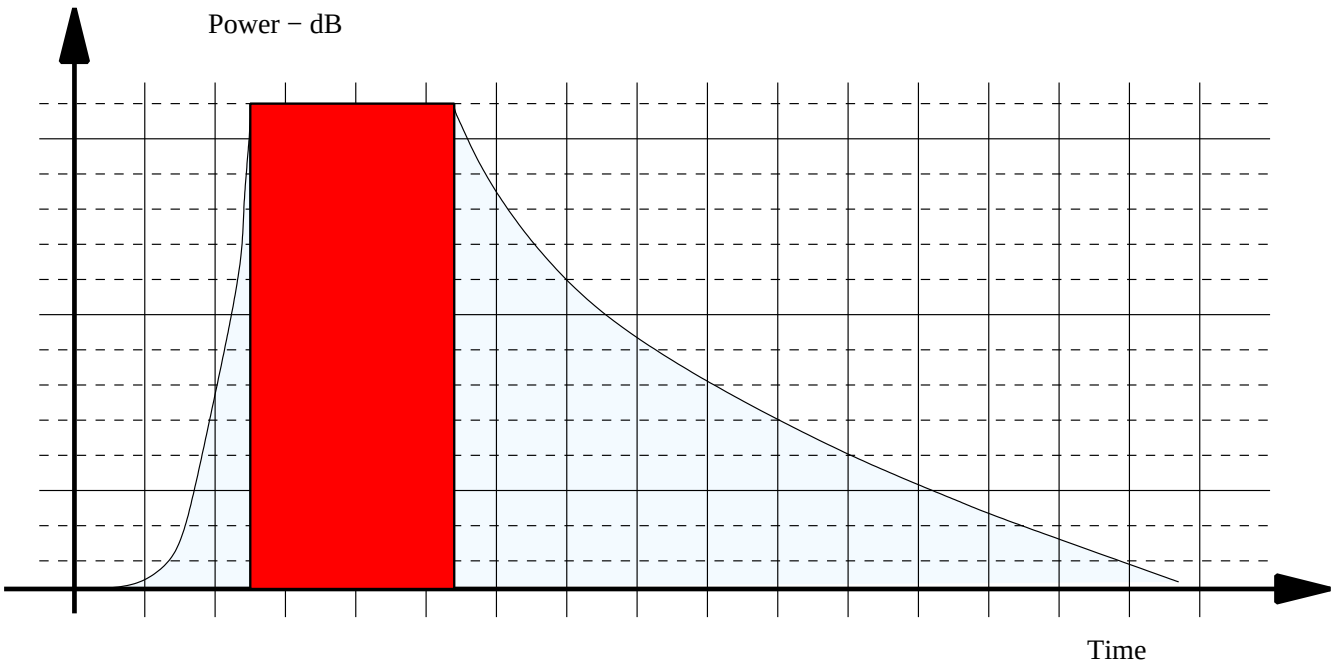


- Pour f_0 et σ^2 donnés, $S_m(f)$ a une allure triangulaire
- Le maximum est pour $f = f_0$
- Indice de masquage: $S_m(f, \sigma^2, f) - \sigma^2$

- On observe que $S_m(f, \sigma^2, f) < \sigma^2$ (le deuxième son ne doit pas forcement être plus puissant du premier)
- Le décroissance est moins rapide quand f_1 augmente
- La pente de décroissance est proportionnelle à la BC
- La pente vers le fréquence supérieures est fonction (décroissante) de σ^2

Courbes de masquage

Masquage temporel



- Pré-masquage : 2÷5 ms
- Post-masquage : 100÷200 ms

Applicabilité du modèle

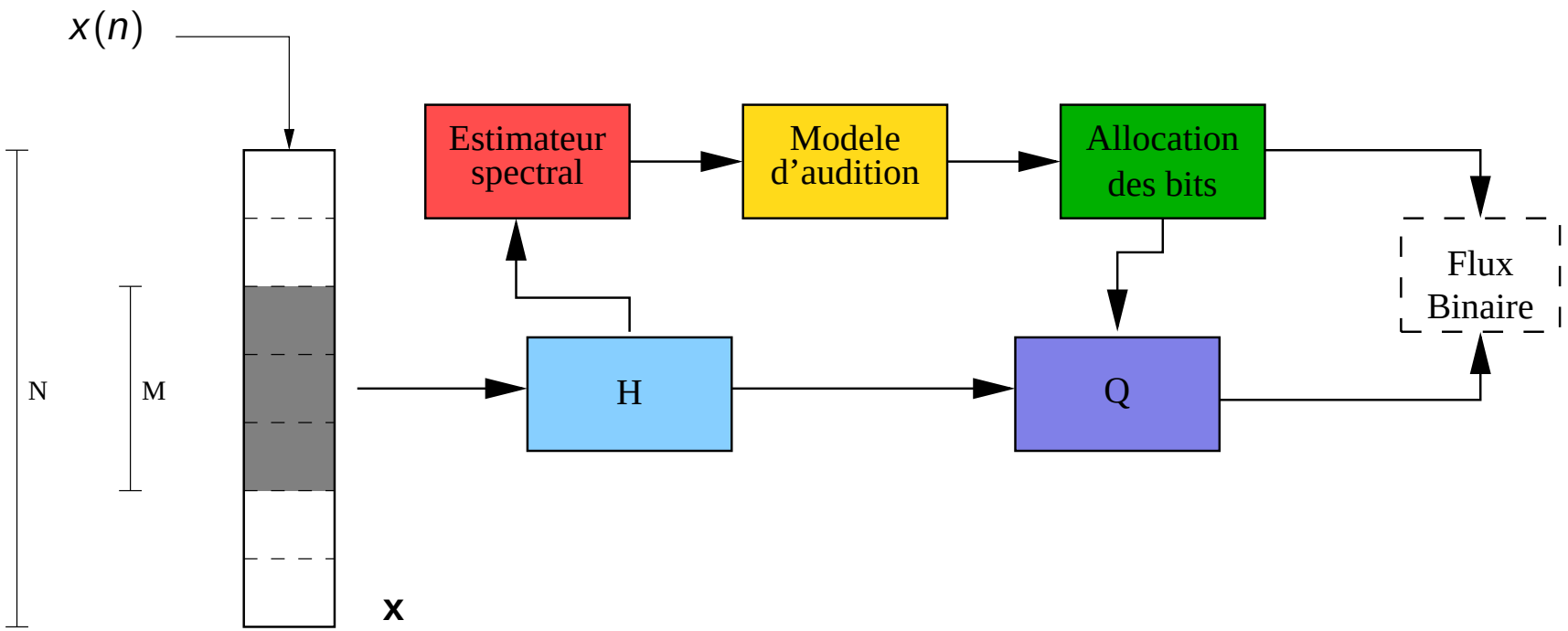
- Le modèle psychoacoustique permet de déterminer certains parties du signal non-audibles
- On permet au bruit de quantification de monter en puissance à condition de rester non-audible
- Tout de même, le modèle est loin d'être parfait :
  - Seul les sons pur ou à bande étroite sont considérés
  - On est capable de évaluer l'influence réciproque de pas plus que 3 sons à la fois
  - Les signaux réels sont composés de très nombreuses contributions : comment interagissent-elles ?
- En pratique, les paramètres des algorithmes de compression de son sont déterminés de façon expérimentale, après un grand nombre de tests

Codeurs perceptuels

- Codage par fenêtres à recouvrement
- Buffer de N échantillons ; M nouveaux entrent dans les buffer et sont codés
- Exemples : $M = 32$ et $N = 512$ (MP3) ; $M = 1024$ et $N = 2048$ (AAC)
- Trois modules qui sont actives pour chaque fenêtre d'analyse
  - Transformation temps-fréquence
  - Allocation de bits (sous le contrôle d'un modèle d'audition
  - Quantification (scalaire ou vectorielle)/codage sans pertes

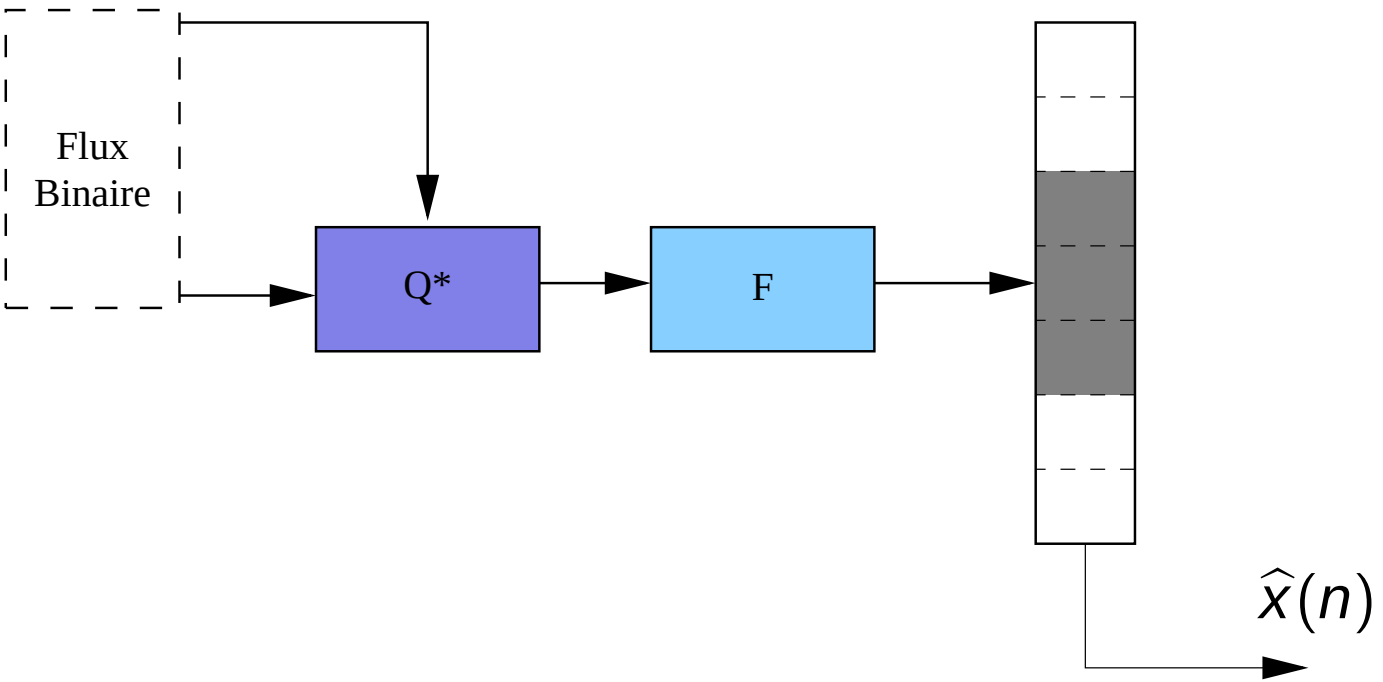
Codeurs perceptuels

Codeur



Codeurs perceptuels

Décodeur



Codeurs perceptuels

- Transformée temps-fréquence : analyse spectrale (M-DCT) pour chaque fenêtre temporelle
- Principe : faire en sorte que le bruit de quantification reste inaudible
- Basé sur un modèle d'audition

Codeur de musique MPEG-1/MP3

- Codeur de musique *transparent* : qualité subjective parfaite
- Trois "couches" de complexité, correspondants à des débits de plus en plus faibles
  - MP3 : codeur MPEG-1, 3-ème couche
- Trois fréquences d'échantillonnage sont possibles, mais fera référence au cas $f_e = 44.1\text{kHz}$

Codeur de musique MPEG-1/MP3

Transformée temps-fréquence

- Banque de 32 filtres
- Répartition uniforme des fréquences entre 0 et 22 kHz
  - À peu près 700 Hz par sous-bande
- Sous-échantillonnage critique et reconstruction quasi-parfaite (SNR$>$90dB en absence de quantification)
- Dans chaque sous-bande on regroupe 12 échantillon, en on les code conjointement
- Le vecteur de 12 coefficients de la sous-bande k est indiqué comme $\mathbf{y}_k$
- Chaque vecteur correspond à environ une dizaine de ms

Codeur de musique MPEG-1/MP3

Représentation des sous-bandes

- Normalisation des vecteurs de sous-bande :

$$\mathbf{y}_k = g_k \mathbf{a}_k$$

- g_k : facteur d'échelle, correspondants à la valeur absolue la plus élevée entre les composantes, et quantifiée sur 6 bit
- $\mathbf{a}_k$: vecteur normalisé (valeurs entre -1 et +1)
- Les échantillons de x de la fenêtre courante sont aussi utilisé pour calculer $\widehat{S}_X(f)$ et un seuil de masquage $\Phi(f)$ basé sur un modèle psychoacoustique

Codeur de musique MPEG-1/MP3

Allocation de débit et quantification

- Pour chaque sous-bande on connaît le rapport signal sur masque
- On alloue les bits disponible en donnant d'abord au sous-bandes avec le plus grand rapport signal sur masque et en suite aux autres (algorithme *greedy*)
- On choisit donc le nombre de bits utilisé pour coder la sous-bande k, pour tout k
- La sous-bande est codé avec un codeur scalaire uniforme
  - On choisit entre 16 quantificateurs (c'est-à-dire, débits) pour les premières 11 sous-bandes, 8 pour les 12 suivantes et 4 pour les 4 dernières. Les sous-bandes 27 à 32 ne sont jamais codées

Codeur de musique MPEG-2 AAC

- C'est aussi un codeur perceptuel
- Transformée M-DCT avec N=2048 et M=1024
- Le vecteur des coefficients de la transformé est

$$\mathbf{X} = [X(0)\, X(1)\, \ldots\, X(M-1)]$$

- Il est représenté par un couple de vecteurs :
  - Facteurs d'échelle :

$$\mathbf{g} = [g(0)\, g(1)\, \ldots\, g(M-1)]$$

  - Valeurs normalisées :

$$\mathbf{i} = \left[\frac{X(0)}{g(0)}\ \frac{X(1)}{g(1)}\ \cdots\ \frac{X(M-1)}{g(M-1)}\right]$$

Codeur de musique MPEG-2 AAC

Vecteur normalisé

- Les éléments du vecteur normalisé sont quantifiés sur 3 bits
- En suite ils sont groupé par demi-bandes critiques (51 groupes)
- Dans chaque groupe, on utilise un codeur de Huffman qui dépend de la valeur de la composante plus importante dans le groupe

Codeur de musique MPEG-2 AAC

Facteurs d'échelle

- Problème d'optimisation : déterminer **g** qui minimise l'erreur de reconstruction
  - Contrainte : débit maximale donné
- Dans ce cas on obtient des taux de compression faibles ($\approx$2)
- On comprime plus si on ajoute la contrainte perceptuelle : la puissance du bruit est inférieure au seuil de masquage

$$S_Q(F) < \Phi(f)$$

- Algorithme du gradient pour trouver les **g** optimaux ; ils sont groupés comme les **i** et codés en différentiel avec Huffman

Codeur MPEG-4 HE AAC

- High Efficiency Advanced Audio Coder
- État de l'art dans le codage de musique
- Développé par ISO, 3GPP, ETSI
- Basé sur MPEG-2 AAC, plus des outils :
  - Spectral Band Replication, pour élargir la largeur de bande représentée
  - Parametric Stereo, pour optimiser les cas de canaux multiples (du 2.0 au 5.1)

MPEG Unified Speech and Audio Codec

- ISO/IEC (JTC1/SC29/WG11) : Call for Proposals en 2007
- Codeur hybride : 2 modes distincts + un classifieur
  - Utiliser 3GPP AMR-WB pour la parole et MPEG-4 HE AAC pour la musique
- Difficultés :
  - assurer des transitions rapides et douces entre parole, musique et signaux mixtes
  - exploiter différents types de fenêtres, de différentes dimensions suivant le type des signaux

Conclusions

Codage de parole

- Modèle psychoacoustique
- Modèle de source et de filtre
- Quantification vectorielle

Codage de musique

- Modèle psychoacoustique
- Allocation du débit
- Optimisation avec contrainte en boucle fermée