

# Introduction à la compression

Marco Cagnazzo

[cagnazzo@telecom-paristech.fr](mailto:cagnazzo@telecom-paristech.fr)

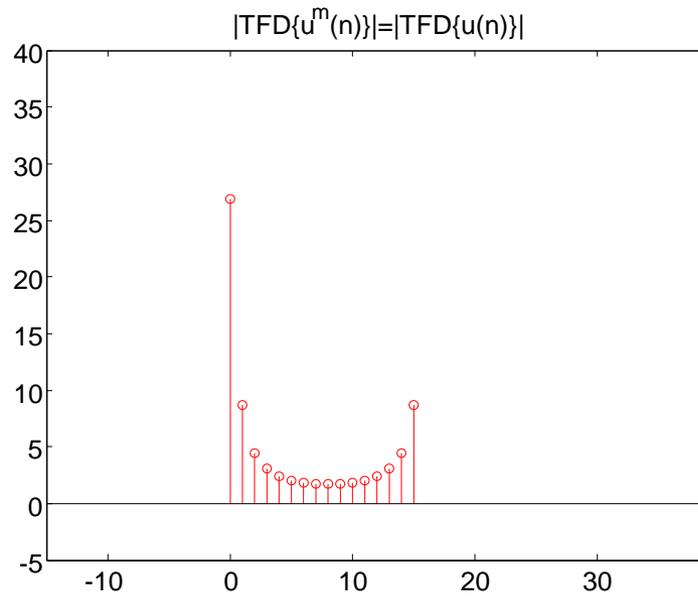
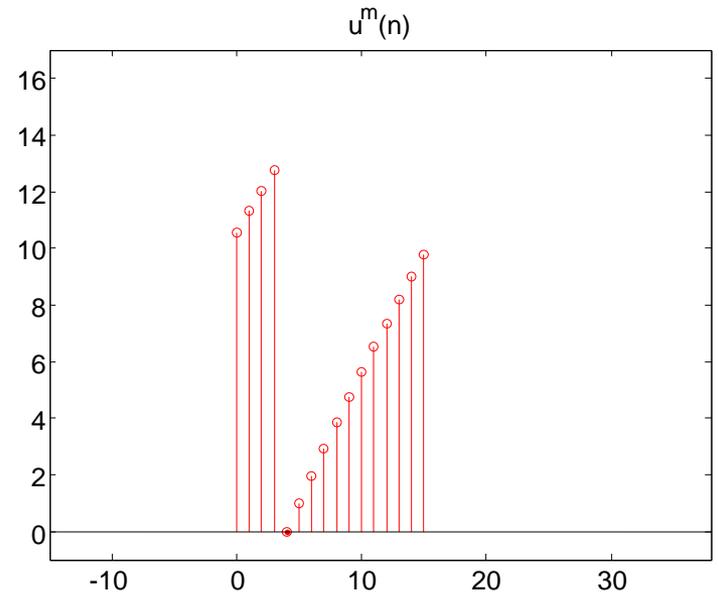
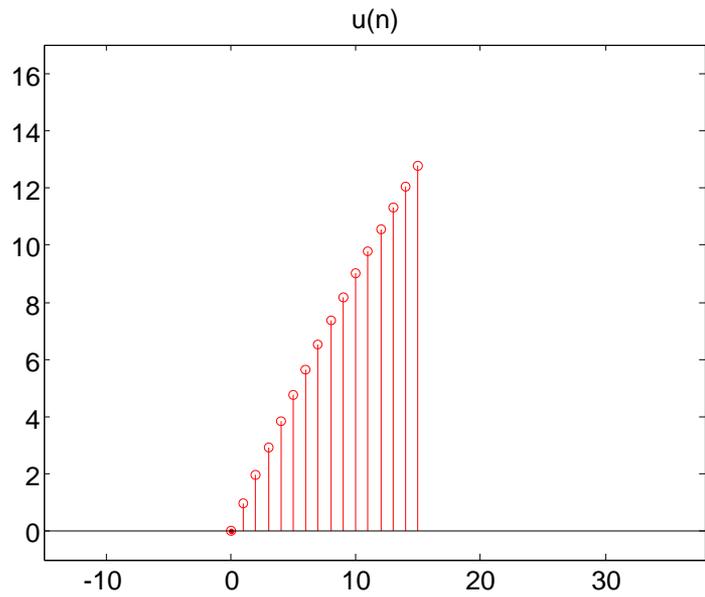
# Transformée et compression

- Signaux finis : la TFD sur  $N$  points est une base pour  $\mathbb{C}^N$
- Connaître les coefficients est équivalent à connaître le signal
- Comme l'on verra rigoureusement plus tard, représenter un signal par ses coefficients peut aider la compression
- Intuition :
  - Tous les  $N$  échantillons d'un signal ont la même importance
  - Il est possible que certains coefficients de sa transformée soient plus importants que d'autres

# Transformée et compression

- Pour un signal réel fini  $u$  la TFD double les coefficients (partie réelle et partie imaginaire)... mais symétrie
- Décroissance des coefficients : les derniers coefficients sont moins importants
- Décroissance *lente*

# Exemple



# La transformée cosinus discret

- Nouvelle transformée
- De  $\mathbb{R}^N$  en  $\mathbb{R}^N$
- Meilleure concentration de l'énergie  $\Leftrightarrow$  décroissance asymptotique rapide
- Lien avec la TFD  $\Rightarrow$  facile à implémenter et à interpréter

# La transformée cosinus discret

Soit  $u: n \in \{0, 1, \dots, N - 1\} \rightarrow \mathbb{R}$  un signal fini réel

Pour un tel signal, on définit sa *Transformée Cosinus Discret* (TCD ou DCT avec l'acronyme anglais), le signal fini  $\hat{u}^D$ :

$$\forall k \in \{0, 1, \dots, N - 1\},$$

$$\hat{u}_k^D = w_k \sum_{n=0}^{N-1} u_n \cos \left[ 2\pi \left( n + \frac{1}{2} \right) \frac{k}{2N} \right]$$

$$w_0 = \frac{1}{\sqrt{N}}, w_k = \sqrt{2}w_0$$

# La transformée cosinus discret

Lien avec la TFD

1. Exponentielle remplacée par un cosinus
2. Echantillonnage du cosinus en  $n + \frac{1}{2}$  au lieu de  $n$
3. Fréquence divisé par 2 :  $\frac{k}{2N}$
4. Coefficients  $w_k$  pour la normalisation

# La transformée cosinus discret

Étant donné le signal  $u$  de durée  $N$ ,  
sa TCD  $\hat{u}^D$  coïncide avec la TFD d'un signal  
 $x$  symétrisé et *décalé d'un demi-échantillon* :

$$\hat{u}_k^D = \left( \frac{W_k}{2} e^{-2i\pi \frac{1}{2} \cdot \frac{k}{2N}} \right) \hat{x}_k$$

Le décalage d'un demi-échantillon sert à rendre le signal symétrique et donc sa transformée réelle.

Il est traduit en fréquence, par la multiplication par un « onde de Fourier » de fréquence  $1/2$

# La transformée cosinus discret

Si

$$x_n = \begin{cases} u_n & \forall n \in \{0, 1, \dots, N-1\} \\ u_{2N-1-n} & \forall n \in \{N, \dots, 2N-1\} \end{cases}$$

alors

$$\hat{u}_k^D = \left( \frac{w_k}{2} e^{-2i\pi \frac{1}{2} \cdot \frac{k}{2N}} \right) \hat{x}_k$$

Pour  $k = 0$  la preuve est immédiate.

Pour  $k \neq 0$  on a ce qu'il suit.

# La transformée cosinus discret

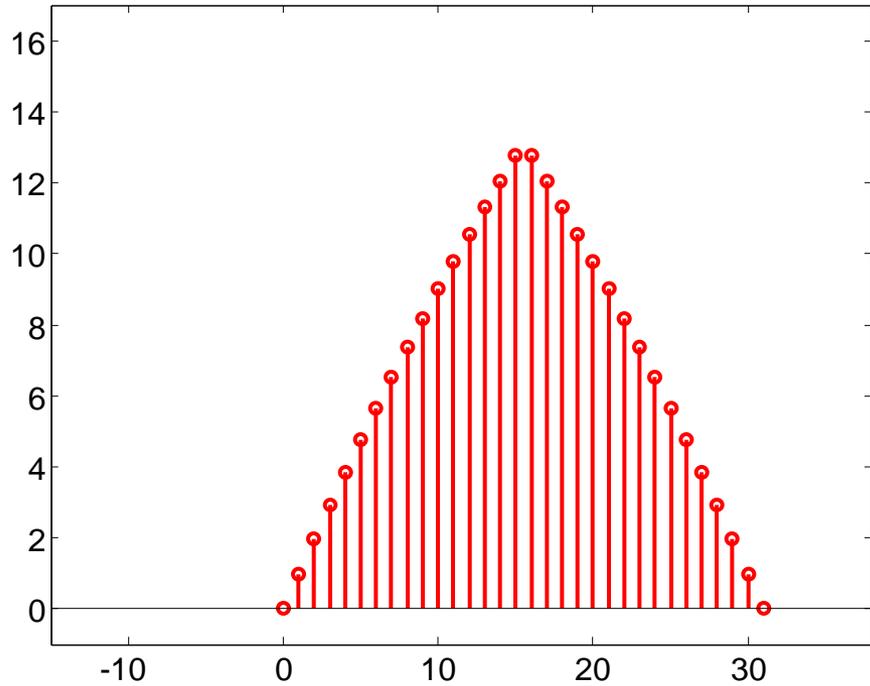
$$\begin{aligned}\hat{x}_k &= \sum_{n=0}^{2N-1} x_n e^{-2i\pi \frac{k}{2N} n} = \\ &= \sum_{n=0}^{N-1} u_n e^{-2i\pi \frac{k}{2N} n} + \sum_{n=N}^{2N-1} u_{2N-1-n} e^{-2i\pi \frac{k}{2N} n} \\ &= \sum_{n=0}^{N-1} u_n e^{-2i\pi \frac{k}{2N} n} + \sum_{m=0}^{N-1} u_m e^{-2i\pi \frac{k}{2N} (-n-1)}\end{aligned}$$

# La transformée cosinus discret

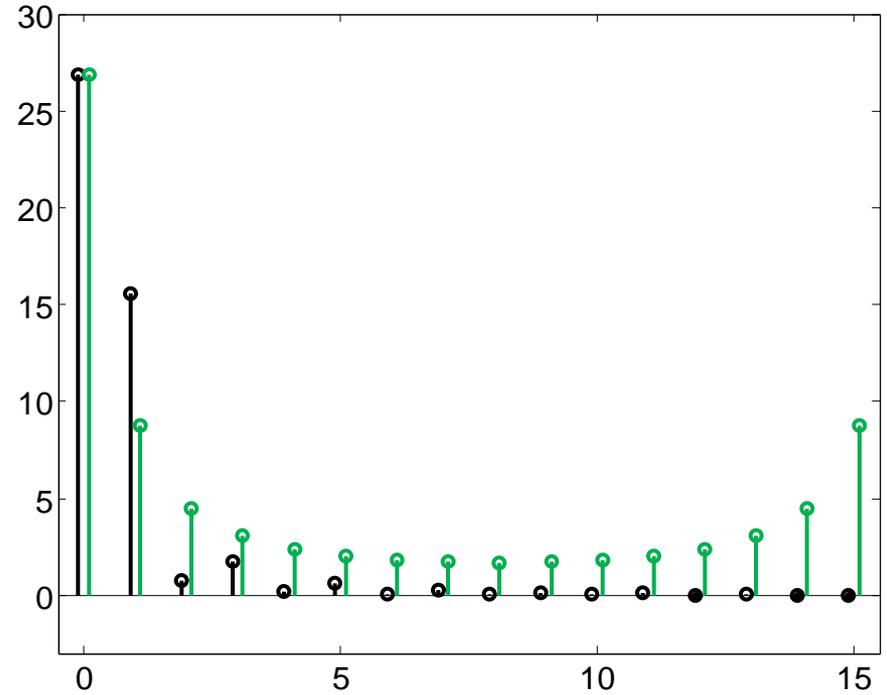
$$\begin{aligned}\hat{x}_k &= \sum_{n=0}^{N-1} u_n e^{-2i\pi \frac{k}{2N} n} + \sum_{m=0}^{N-1} u_m e^{-2i\pi \frac{k}{2N} (-n-1)} \\ &= \sum_{n=0}^{N-1} u_m \left[ e^{-2i\pi \frac{k}{2N} n} + e^{2i\pi \frac{k}{2N} (n+1)} \right] \\ &= \sum_{n=0}^{N-1} u_m e^{i\pi \frac{k}{2N}} \left[ e^{-2i\pi \frac{k}{2N} \left(n + \frac{1}{2}\right)} + e^{2i\pi \frac{k}{2N} \left(n + \frac{1}{2}\right)} \right] \\ &= 2e^{i\pi \frac{k}{2N}} \frac{1}{w_k} \hat{u}_k^D \Leftrightarrow \hat{u}_k^D = \left( \frac{w_k}{2} e^{-i\pi \frac{k}{2N}} \right) \hat{x}_k\end{aligned}$$

# Exemple

$x(n)$



TFD normalisée de  $x$  et de  $u$



# La transformée cosinus discret

- Les coefficients  $w_n$  assurent l'isométrie de la TCD :

$$\begin{aligned}\|u\|^2 &= \frac{1}{2} \|x\|^2 = \frac{1}{4N} \|\hat{x}\|^2 = \\ &= \frac{1}{4N} \left[ |\hat{x}_0|^2 + 2 \sum_{k=0}^{N-1} |\hat{x}_k| \right] = \frac{1}{4N} \left[ 4N |\hat{u}_0^D|^2 + 4N \sum_{k=0}^{N-1} |\hat{u}_k^D| \right] \\ &= \|\hat{u}^D\|\end{aligned}$$

# Décroissance asymptotique

- Décroissance asymptotique plus rapide : TFD d'un signal plus régulier

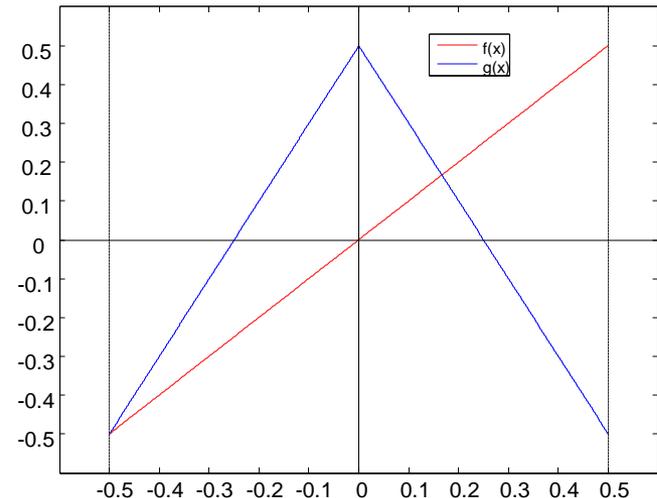
- La symétrisation élimine la discontinuité au bord

- Élément de réflexion :

- Coefficients de Fourier de

$$f: x \in \left[-\frac{1}{2}, \frac{1}{2}\right] \rightarrow x \text{ et de}$$

$$g: x \in \left[-\frac{1}{2}, \frac{1}{2}\right] \rightarrow \frac{1}{2} - 2|x|$$



# Décroissance asymptotique

$$\int x e^{\alpha x} dx = \frac{e^{\alpha x}}{\alpha^2} (\alpha x - 1)$$

$$c_n = \int_{-1/2}^{1/2} x e^{2i\pi n x} dx = \frac{i(-1)^n}{2\pi n}$$

$$d_n = 2 \int_0^{1/2} \left( \frac{1}{2} - 2x \right) e^{2i\pi n x} dx = \begin{cases} 0 & \text{si } n \text{ pair} \\ \frac{2}{\pi^2 n^2} & \text{si } n \text{ impair} \end{cases}$$

# Bases TCD

Si on considère les vecteurs  $\mathbf{c}_k = [c_k(1) \dots c_k(N)]^T$  avec

$$c_k(n) = w_k \cos \left[ 2\pi \left( n + \frac{1}{2} \right) \frac{k}{2N} \right]$$

alors il est évident que *les coefficients de la transformée de  $u$*  sont le *produit ligne-colonne* de  $u$  et  $\mathbf{c}_k$ :  $\hat{u}(k) = \mathbf{c}_k^T u$

On peut aussi écrire :

$$\hat{\mathbf{u}}^D = \mathbf{C}u$$

où  $\mathbf{C}$  est la matrice dont les lignes sont les  $\mathbf{c}_k^T$

# Base locale TCD

Un signal de taille  $mN$  peut être décomposé en  $m$  signaux de taille  $N$

Sur chacun on peut effectuer une TCD *locale* de taille  $N$

On dit qu'on a fait une analyse par base locale TCD du signal

On sait que cela est équivalent à utiliser une base opportune de  $\mathbb{R}^{mN}$  (voir TD 4)



# TCD 2D

Une base TCD en 2D se construit en utilisant le produit tensoriel comme montré en TD4 :

$$\mathbf{g}_i^j(k, m) = \mathbf{c}_i(k)\mathbf{c}_j(m)$$

Pour calculer la décomposition d'une image sur cette base, il suffit d'appliquer la TCD 1-D sur les lignes, et en suite sur les colonnes du résultat

Si  $U \in \mathbb{R}^{N \times N}$  est une image, on peut calculer sa TCD-2D comme

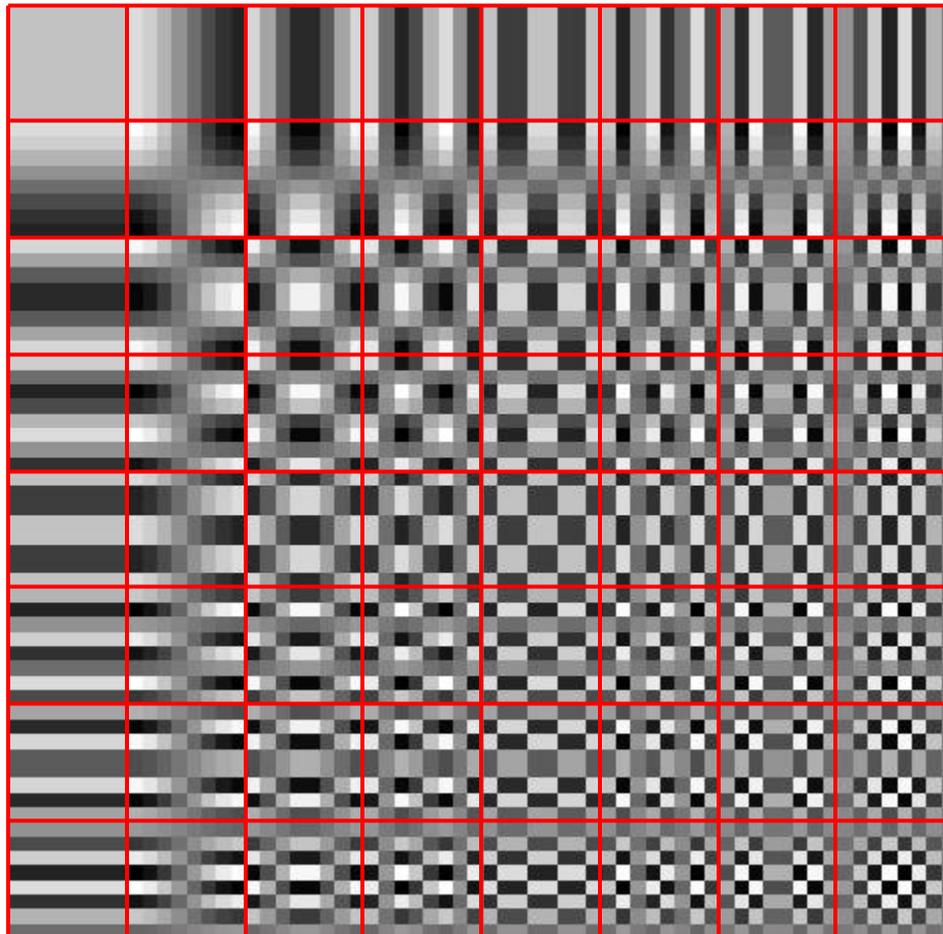
$$\hat{U} = CUC^T$$

# Base locale TCD-2D

- Pour une image de taille  $mN \times mN$  on peut utiliser la TCD 1-D de taille  $N$  (base locale)
  1. On décompose l'image en blocs  $N \times N$
  2. Sur chaque bloc on utilise la TCD 1-D de taille  $N$  sur les lignes et en suite sur les colonnes

# Exemple

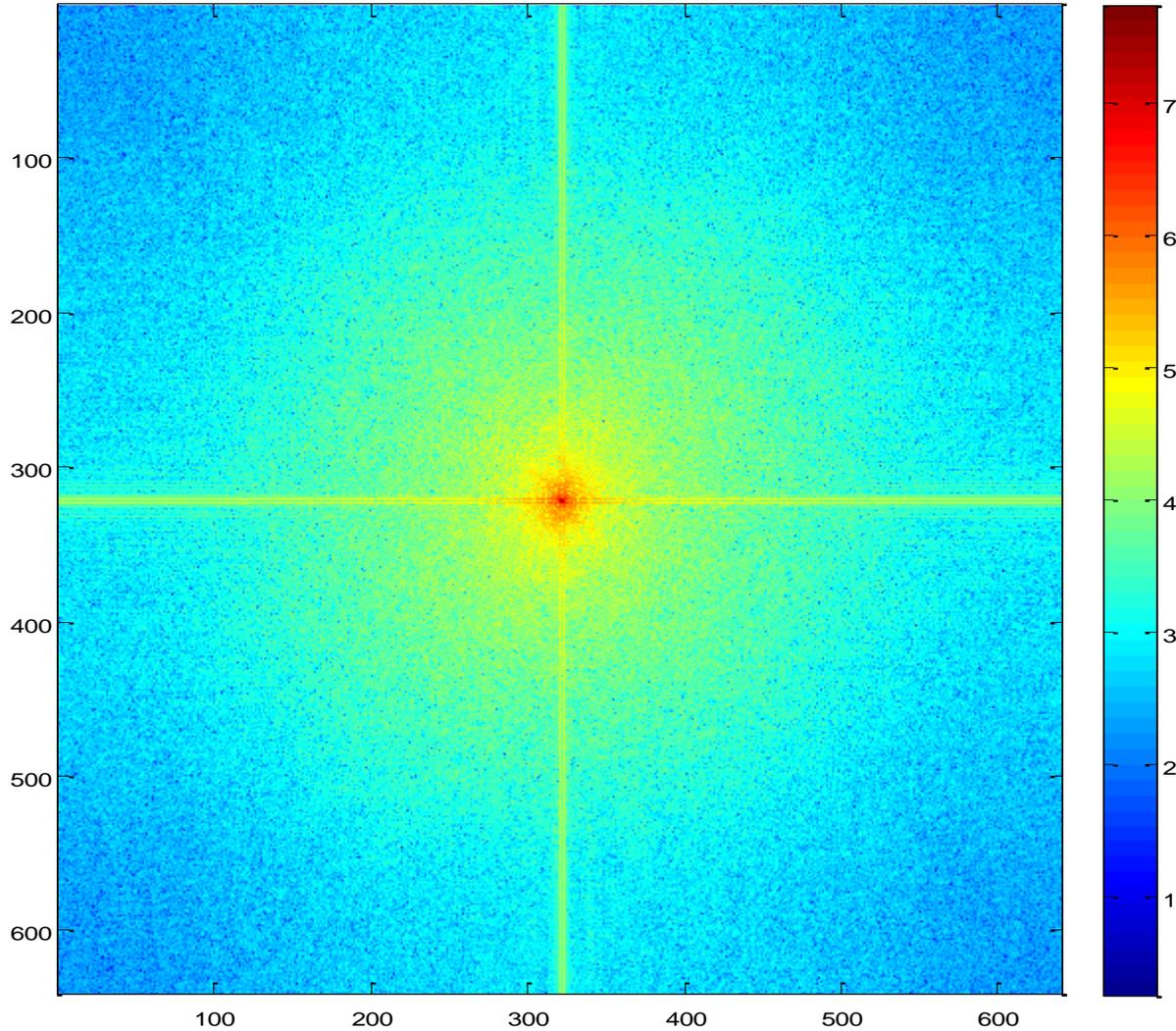
- Base locale TCD 8x8



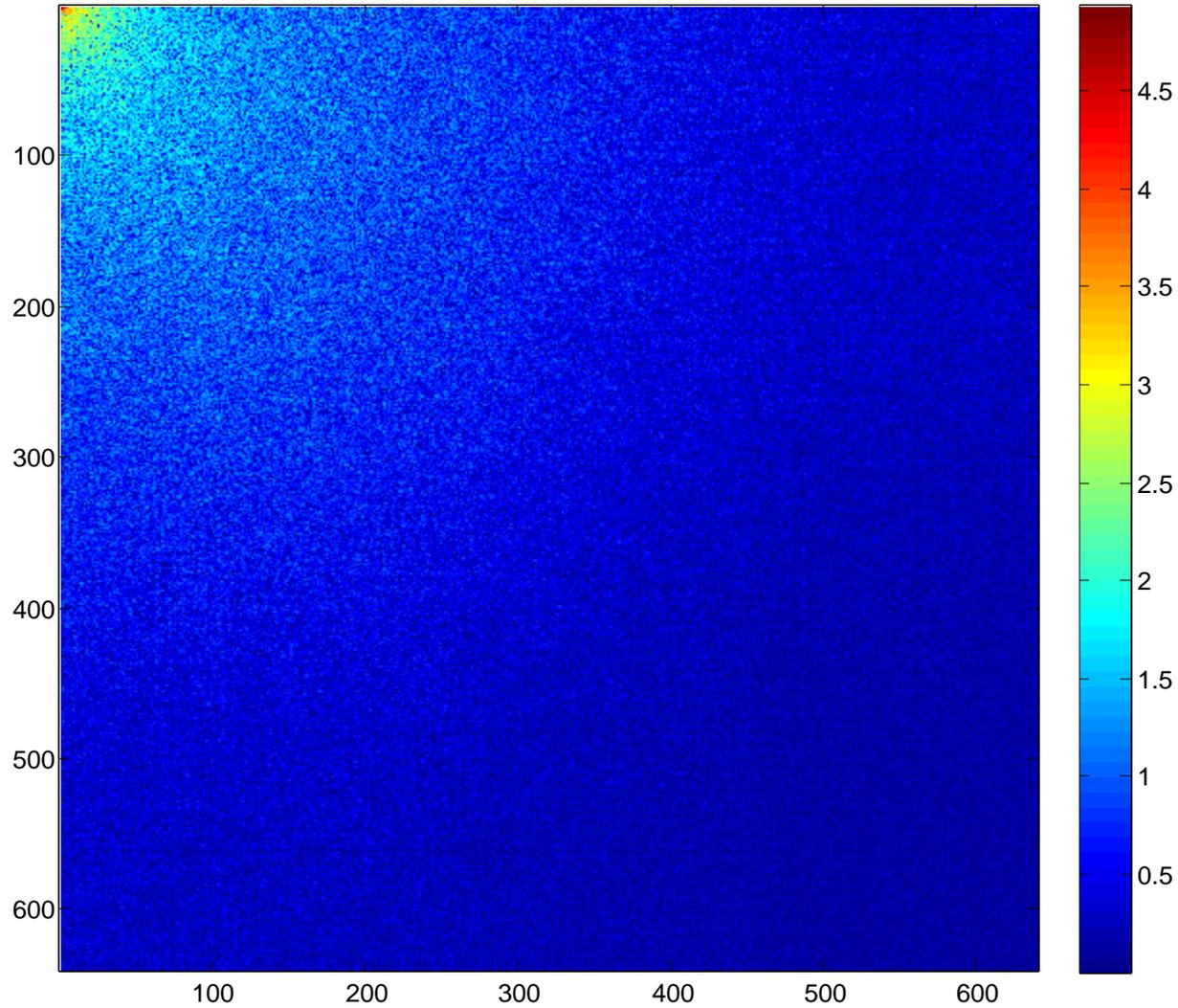
# Exemple



# TFD-2D



# TCD-2D



# TCD-2D locale

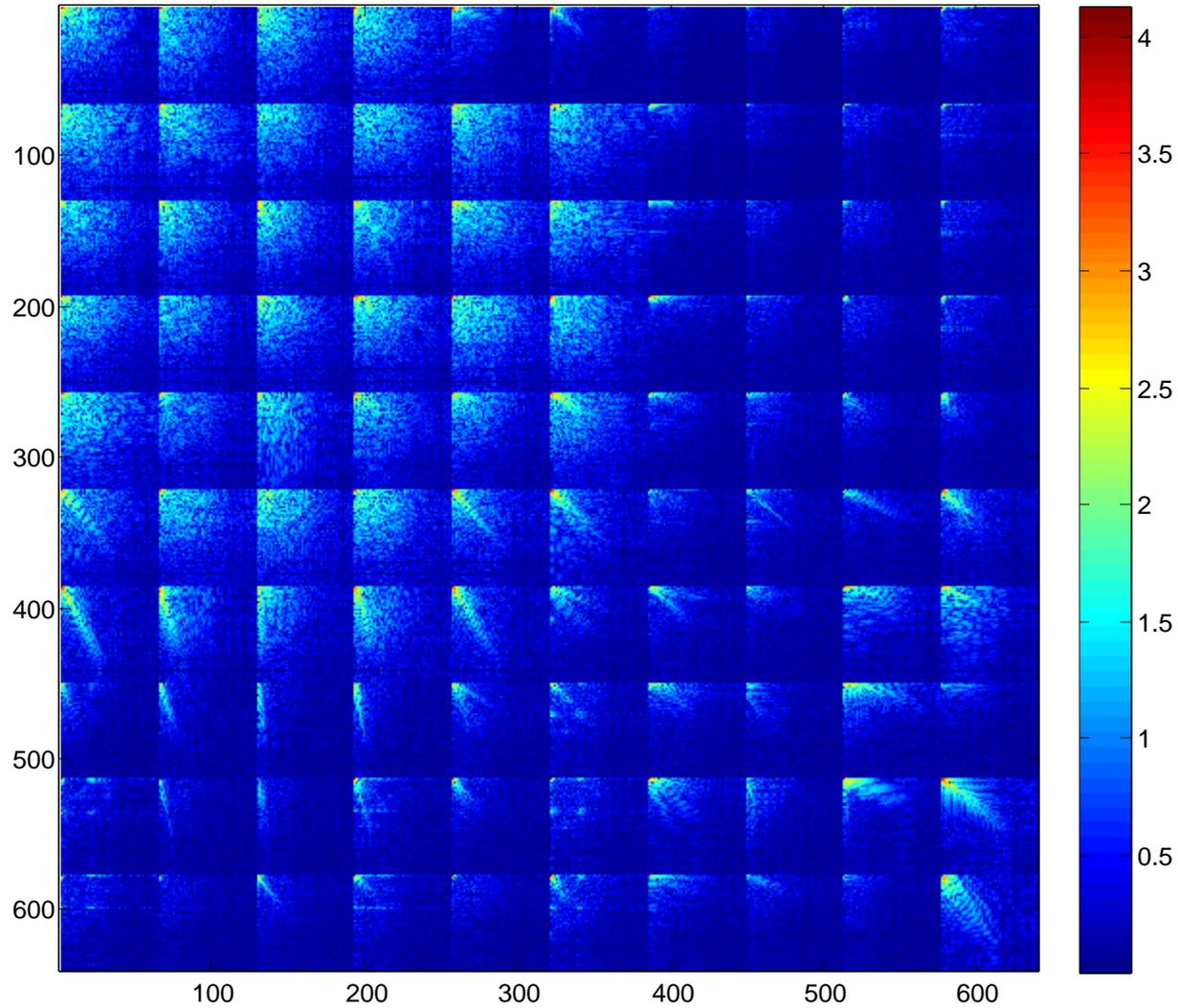


Image reconstruite avec 5 % des coeff TFD. MSE=31.39



Image reconstruite avec 5 % des coeff TCD. MSE=20.83



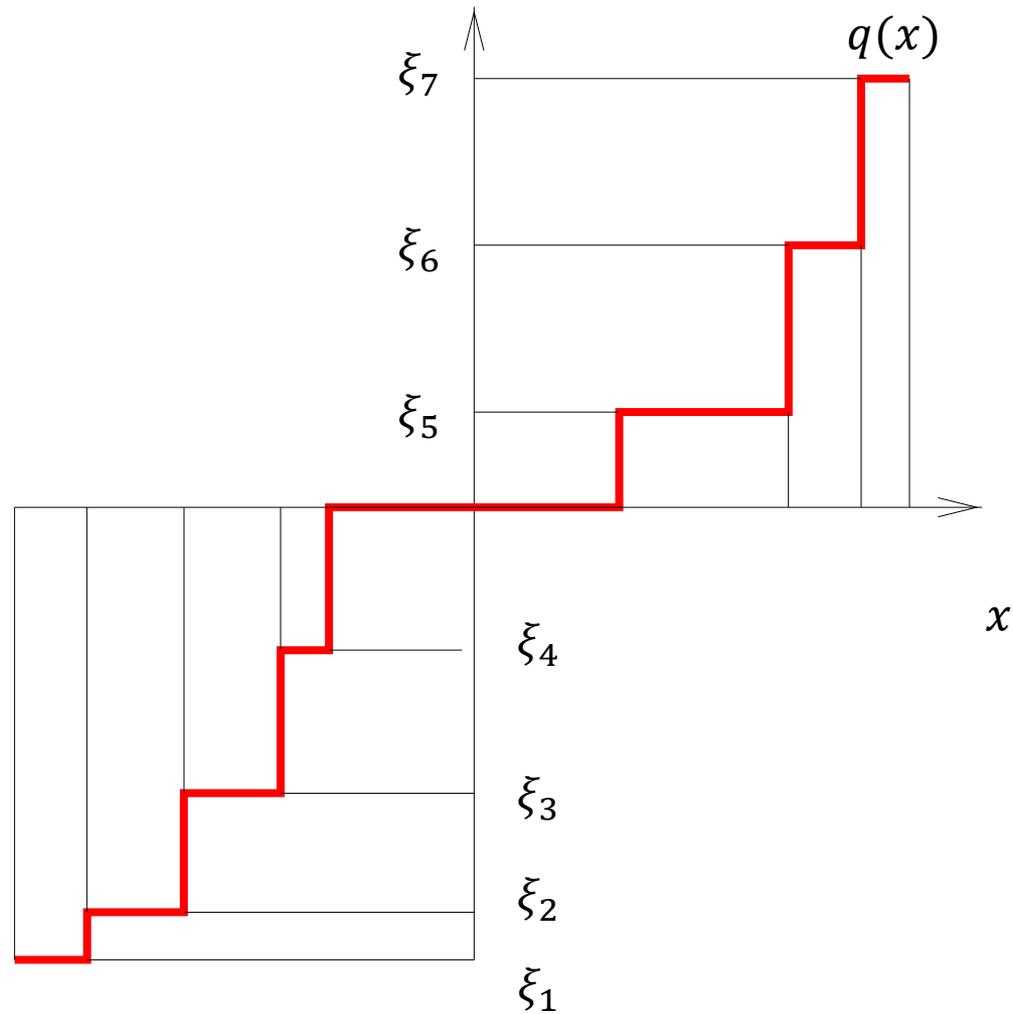
Image reconstruite avec 5% des coeff TCD 64x64. MSE= 9.02



# La quantification

- Définition
- Soit  $\mathcal{X} = (a, b)$  un intervalle réel et  $N$  un entier ( $N \geq 2$ )
- Soit  $\mathcal{C} = \{\xi_i\}_{i=1, \dots, N}$  un ensemble ordonné de  $N$  réel distincts
- Un quantificateur est une application :
$$q: x \in \mathcal{X} \rightarrow q(x) = \hat{x} \in \mathcal{C}$$
- Un quantificateur transforme un réel en un élément d'un ensemble discret

# La quantification



# Terminologie

$\mathcal{C}$  est appelé le dictionnaire

$\xi_i$  est l' $i$ -ème niveau (niveau de représentation)

$N$  est le nombre de niveaux

$C_i = \{x \in \mathcal{X} : q(x) = \xi_i\}$  est l' $i$ -ème cellule de quantification

Les cellules forment une partition de  $\mathcal{X}$

Quantificateur régulier :  $\forall i C_i = (t_i, t_{i+1})$  et  $\xi_i \in C_i$

# Terminologie

- Débit du quantificateur  $B$  : nombre de bit nécessaire pour représenter un niveau
- Pour simplicité, on considérera uniquement le cas  $N = 2^B$  :  $B$  bits sont suffisants pour représenter un niveaux
- Quantificateur uniforme :

$$C_i = [t_i, t_i + \Delta], \xi_i = t_i + \frac{\Delta}{2} \text{ avec}$$

$$\Delta = \frac{b-a}{N}, t_1 = a$$

# Mesure de distorsion

$$d[x, q(\cdot)] = |x - q(x)|^p$$

Souvent  $p = 2$ . Cela permet de voir la distorsion comme une distance euclidienne

A priori l'entrée est inconnue : on représente cela avec un modèle aléatoire.

Donc il est important calcule la distorsion attendue :

$$D(X) = \mathbb{E}[|X - q(X)|^2]$$

# Condition du plus proche voisin

- Il est facile de vérifier que, si les niveaux sont assignés, il existe une façon optimale de choisir les cellules, selon la règle dite du plus proche voisin (*nearest neighbour*):

$$q(x) = \xi_i \Leftrightarrow \forall j \neq i, |x - \xi_i| \leq |x - \xi_j|$$

Ce qui est encore équivalent à choisir les seuils  $t_i$  comme point moyen entre deux niveaux consécutifs :

$$t_i = \frac{\xi_i + \xi_{i+1}}{2}$$

# Quantificateur de Voronoï

- Quand les cellules sont définies avec la règle du PPV, on parle de cellules de Voronoï
- Un quantificateur de Voronoï a comme cellules des cellules de Voronoï
- Le quantificateur uniforme est un quantificateur de Voronoï

# Quantificateur en haute résolution

- Hypothèses de haute résolution :

$$N \rightarrow \infty$$

$$\Delta_{\max} \rightarrow 0$$

$$\xi_i = \frac{t_i + t_{i+1}}{2}$$

$$\forall x \in C_i, f(x) \approx f(\xi_i)$$

Alors :

$$D(q) \xrightarrow{N} \frac{1}{12N^2} \int_a^b \frac{f(x)}{\zeta^2(x)} dx$$

# Formule de Bennett

$$N(x) = \frac{\# \text{ niveaux en } (x, x+dx)}{dx} \text{ densité de niveaux}$$

$$\Delta(x) = \frac{1}{N(x)} \text{ amplitude de la cellule de } x$$

$$\zeta(x) = \frac{N(x)}{N} = \frac{1}{N\Delta(x)} \text{ densité normalisée}$$

# Quantificateur optimale

- Inégalité de Hölder :

$$\text{Si } 1/a + 1/b = 1,$$

$$\int uv \, dx \leq \left[ \int u^a \right]^{\frac{1}{a}} \cdot \left[ \int v^b \right]^{\frac{1}{b}}$$

Avec égalité si et seulement si

$$u^a = k v^b$$

# Quantificateur optimale

- Si on applique l'inégalité de Hölder avec

$$u(x) = \left[ \frac{f_X(x)}{\zeta(x)} \right]^{\frac{1}{3}}, \quad v(x) = [\zeta(x)]^{\frac{2}{3}}, \quad a = 3, \quad b = \frac{3}{2}$$

On trouve :

$$D(q) \geq \frac{1}{12N^2} \left[ \int_a^b f_X^{1/3}(x) dx \right]^3$$

Avec égalité ssi :

$$\zeta(x) = \frac{f_X^{1/3}(x)}{\int_a^b f_X^{1/3}(t) dt}$$

# Quantificateur optimale

- Exemples

Pour une v.a. uniforme, le quantificateur optimale est uniforme, et

$$\begin{aligned} D^* &= \frac{1}{12N^2} \left[ \int_a^b \left[ \frac{1}{b-a} \right]^{\frac{1}{3}} dx \right]^3 = \frac{(b-a)^2}{12N^2} \\ &= 2^{-2B} \sigma_X^2 \end{aligned}$$

# Quantificateur optimale

- Soit  $U = \frac{X}{\sigma_X}$
- Si  $U \sim f_U$  alors  $X \sim \frac{1}{\sigma_X} f_U\left(\frac{x}{\sigma_X}\right)$

Calcul de la distorsion du q.o. de  $X$  en fonction de  $p_U$  et  $\sigma_X$

$$\int [f_X(x)]^{1/3} dx = \frac{1}{\sigma_X^{1/3}} \int \left[ f_U\left(\frac{x}{\sigma_X}\right) \right]^{1/3} dx = \sigma_X^{2/3} \int [f_U(t)]^{1/3} dt$$

$$\text{Avec } t = \frac{x}{\sigma_x}$$

Donc  $D(q) = 2^{-2B} \sigma_X^2 h_X$ , avec  $h_X = \frac{1}{12} \left[ \int f_U^{1/3}(t) dt \right]^3$  facteur de forme

V.a. uniforme :  $h_X = 1$ ; v.a. gaussienne :  $h_X = \frac{\sqrt{3}}{2} \pi \approx 2.72$

# Quantificateurs en bloc

- On considère la quantification conjointe d'un bloc de  $N$  échantillons d'un signal
- On dispose en total de  $B$  bits pour représenter le vecteur
- Comme répartir de la meilleure façon les bits de codage ? Problème de **l'allocation des ressources**

# Allocation optimale des ressources

- Formulation

Soit  $X$  un vecteur de  $n$  v.a. à valeurs en  $\mathcal{X}^n$

Supposons que chaque  $X_i$  soit quantifiée avec son quantificateur optimale, et avec  $N_i$  niveaux de quantification, correspondants à  $B_i$  bits

Quel est le meilleur vecteur d'allocation

$\mathbf{B} = [B_1, B_2, \dots, B_N]$  s.à  $\sum_i B_i = B$  ?

# Allocation optimale des ressources

$$\mathbf{B}^* = \operatorname{argmin}_{\mathbf{B}} D(\mathbf{B}) = \sum_i D_i(B_i)$$

Soumis à :

$$\sum_{i=1}^N B_i = B$$

~~$$\forall l \in \{1, \dots, N\}, B_l \geq 0$$~~

~~$$\forall i \in \{1, \dots, N\}, B_i \in \mathbb{N}_0$$~~

# Allocation optimale des ressources

$$J(\mathbf{B}, \lambda) = \sum_i D_i(B_i) + \lambda \left( \sum_{i=1}^N B_i - B \right)$$

Avec :

$$D_i(B_i) = \sigma_i^2 h_i 2^{-2B_i}$$

En imposant  $\frac{\partial J}{\partial B_i} = 0$  et  $\frac{\partial J}{\partial \lambda} = 0$ , on trouve

$$B_i^* = \bar{B} + \frac{1}{2} \log_2 \frac{\sigma_i^2 h_i}{\sigma_{\text{GM}}^2 h_{\text{GM}}}$$

# Allocation optimale des ressources

- La distorsion correspondante est :

$$D^* = Nh_{\text{GM}}\sigma_{\text{GM}}^2 2^{-2\bar{B}}$$

Pour v.a. gaussiennes,  $h_i$  ne varie pas

Comment peut-on réduire  $\sigma_{\text{GM}}^2$  ?

Avec une transformée !

# Compression par quantification et transformée

- Une transformée o.n. permet de modifier  $\sigma_{GM}^2$
- La distorsion sur les coefficients de la transformée est égal à la distorsion dans le domaine original
- La transformée optimale est celle qui minimise la moyenne géométrique des variances : transformée de Karhunen-Loève

# Compression par quantification et transformée

$X$  vecteur de v.a. Gaussiennes centrées i.d. avec variance  $\sigma_{X_i}^2 = \sigma_X^2$  et  $h_i = h_{\mathcal{N}}$  pour tout  $i$

Donc la distorsion minimale, appelée  $D_{\text{PCM}}$  est

$$D_{\text{PCM}} = Nh_{\mathcal{N}}\sigma_X^2 2^{-2\bar{B}}$$

Si on transforme , on a  $Y = \mathbf{T}X$  encore un vecteur de v.a. Gaussiennes mais avec variances non identiques :

$$D_{\text{T}} = Nh_{\mathcal{N}}\sigma_{\text{GM},Y}^2 2^{-2\bar{B}}$$

# Compression par quantification et transformée

On définit le gain de codage :

$$CG = \frac{D_{PCM}}{D_T} = \frac{Nh_{\mathcal{N}}\sigma_X^2 2^{-2\bar{B}}}{Nh_{\mathcal{N}}\sigma_{GM,Y}^2 2^{-2\bar{B}}} = \frac{\sigma_X^2}{\sigma_{GM,Y}^2} = \frac{\sigma_{AM,Y}^2}{\sigma_{GM,Y}^2}$$

Donc une transformée doit minimiser la moyenne géométrique des variances

Par contre toute transformée o.n. ne modifie pas la moyenne arithmétique des variances

# Transformée optimale

- Soit  $X$  un vecteur de v.a. gaussiennes centrées avec matrice de covariance  $\Sigma$

$$\Sigma = \mathbb{E}[XX^T]$$

- Soit  $M$  une transformée orthogonale
- Soit  $Y = MX$  encore un vecteur de v.a. gaussiennes
- Quel est  $M$  qui minimise  $\sigma_{GM,Y}^2$  ?

# Transformée optimale

Si les composantes de  $X$  sont linéairement indépendantes,  $\Sigma$  est définie positive et admet la représentation :

$$\Sigma = U\Lambda U^T$$

$U = [u_1 | \dots | u_n]$  matrice des vecteurs propres o.n. et  $\Lambda = \text{Diag}(\lambda_1, \dots, \lambda_n)$  est la matrice diagonale des valeurs propres.

La transformée de Karhunen-Loève (TKL) est définie par :  $M = U^T$

# Optimalité de la TKL

- La démonstration est basé sur la propriété que le déterminant de la matrice d'autocorrélation d'un vecteur Gaussien centré à variances unitaires est toujours inférieur ou égal à un
- On peut aussi montrer que :

$\forall T$  transf. o.n., si  $Y = U^T X$  et  $Z = TX$ , alors

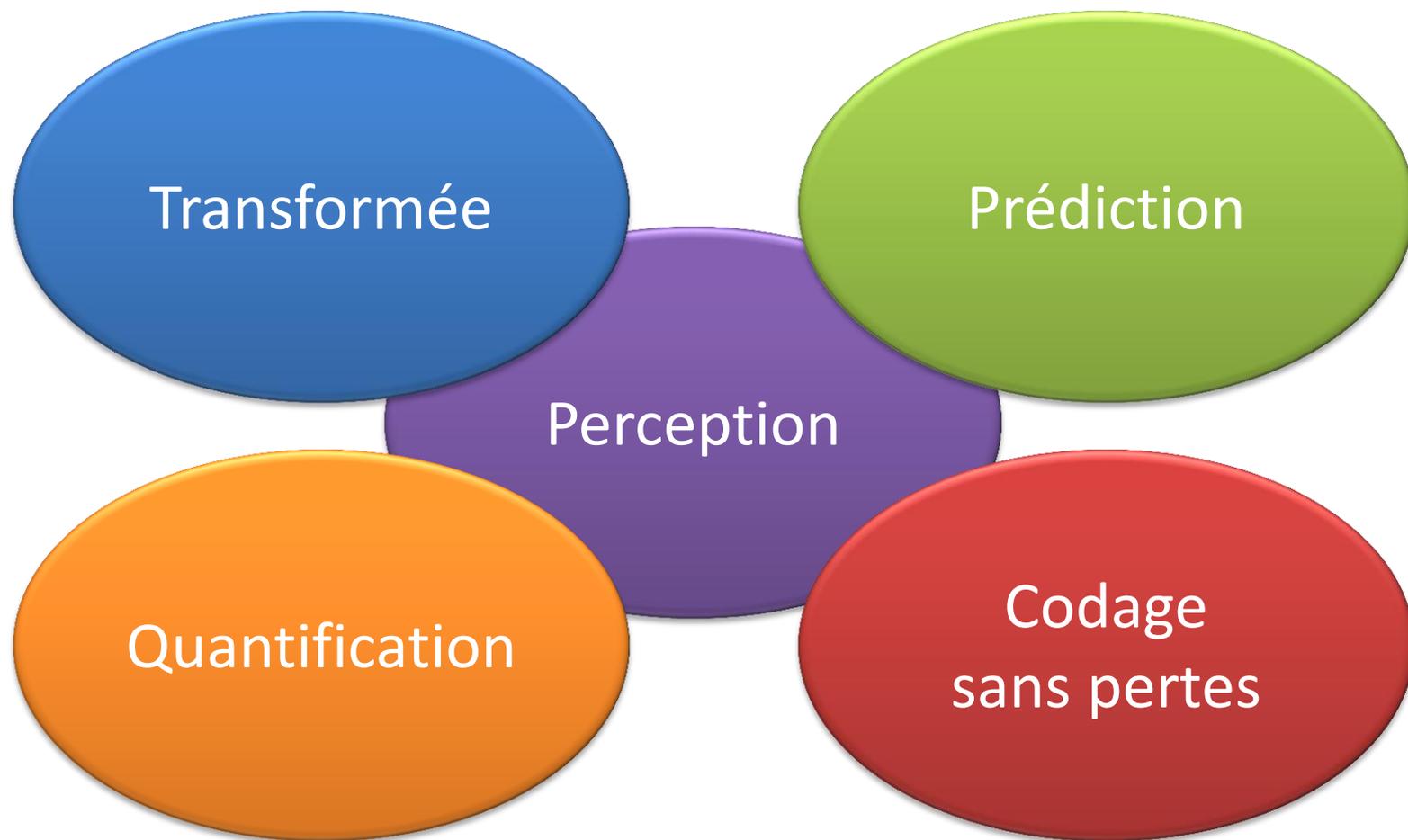
$$\sum_{i=1}^m \mathbb{E}[Y_i^2] \geq \sum_{i=1}^m \mathbb{E}[Z_i^2]$$

# Compression des signaux

# Principes de la compression

- Redondance des signaux
  - Statistique : intrinsèque au signal
  - Psychophysique : lié à l'utilisateur (système de vision, système d'audition)
- Compression sans et avec perte (quantification)
- Mesures de cout et qualité
  - Débit et distorsion
  - MSE : une bonne mesure de qualité ?
- Complexité, robustesse, retard

# Les ingrédients de la compression



# Les ingrédients de la compression

- Aujourd'hui on a le temps uniquement pour parler de transformées
  - Même pas toutes ! P.e., transformée en ondelettes, transformées entières, transformée hiérarchiques
- Les autres éléments sont également très important :
  - Prédiction spatiale (motifs, texture)
  - Prédiction temporelle (estimation du mouvement)
  - Codage sans pertes (zip, gzip, etc)
  - Qualité perçue
  - Quantification vectorielle, à zone morte ...
  - Compromis complexité/efficacité/robustesse/retard
- Aspects traités dans SI222 et SI350 !

# Définitions

Soit :

$x$  un vecteur de  $\mathbb{R}^N$

$\{\alpha_i\}_{i \in \{1, \dots, M\}}$  une collection de  $M$  vecteurs de  $\mathbb{R}^N$

$\{n_j\}_{j \in \{0, \dots, m-1\}}$  une collection de  $m$  indices entre  
1 et  $M$

$\{a_j\}_{j \in \{0, \dots, m-1\}}$  une collection de  $m$  réels

# Définitions

- On définit une approximation de  $x$  dans la collection d'atomes  $\alpha_i$  le vecteur  $\tilde{x}$  :

$$\tilde{x} = \sum_{i=0}^{m-1} a_j \alpha_{n_j}$$

Le taux de compression est  $\tau_c = \frac{m}{N}$

L'erreur relative de compression :  $\text{err}_c = \frac{\|x - \tilde{x}\|}{\|x\|}$

# Problèmes

- Trouver une bonne famille  $\alpha$  pour coder les signaux naturels
- Étant donnée la famille  $\alpha$  trouver les coefficients  $a_j$ 
  - Cela implique le choix des indices  $j$ , donc des atomes à utiliser pour représenter le signal

# Exemple (simple)

$$x = (1, -1, 1, -1)$$

$\alpha$  base canonique de  $\mathbb{R}^4$

On utilise seulement deux vecteurs de la base canonique, le premier et le dernier: donc

$$m = 2, n_0 = 1, n_1 = 4$$

On code le signal avec coefficients  $a_0 = 1/3$  et

$$a_1 = -1$$

# Exemple, suite

Résultat :  $\tilde{x} = (1/3, 0, 0, -1)$

Erreur quadratique :  $\frac{\frac{4}{9}+1+1+0}{4} = 0.6\bar{1}$

Amélioration : avec des meilleurs coefficients

$$a_0 = 1$$

$$\tilde{x} = (1, 0, 0, -1)$$

Erreur quadratique :  $\frac{0+1+1+0}{4} = 0.5$

Pour cette famille et avec  $m = 2$  on peut pas faire mieux

# Exemple, suite

Nouvelle famille :  $\alpha$  sont les vecteurs de la TFD d'ordre 4

$$\alpha_k = \left(1, e^{2i\pi\frac{k}{4}}, e^{4i\pi\frac{k}{4}}, e^{6i\pi\frac{k}{4}}\right)$$

Il suffit de prendre :  $m = 1, n_0 = 2$  pour que

$$\tilde{x} = (1, -1, 1, -1) = x$$

Donc erreur = 0

Conclusion : importance du choix des coefficients, mais aussi importance du choix de la famille

# Choix de la base

- Problème très proche de ce qu'on a vu en TD 4
- Nous avons  $n$  vecteurs  $V_i$  de  $\mathbb{R}^N$
- Nous voulons les approximer en utilisant une famille  $\{v_j\}$
- On fixe  $m$ , le nombre d'éléments de la famille à utiliser
- On doit minimiser la distance des  $V_i$  de leur représentation sur la famille
- En TD on a vu le cas  $m=1$

# Choix de la base

Formalisation du problème

Trouver les coefficients  $\beta_j^i$  et les vecteurs  $v_j$  qui minimisent

$$E = \sum_{i=1}^n \left\| V_i - \sum_{j=1}^m \beta_j^i v_j \right\|^2$$

Quand les  $v_j$  sont orthonormés, le meilleur choix pour les coefficients est le produit scalaire

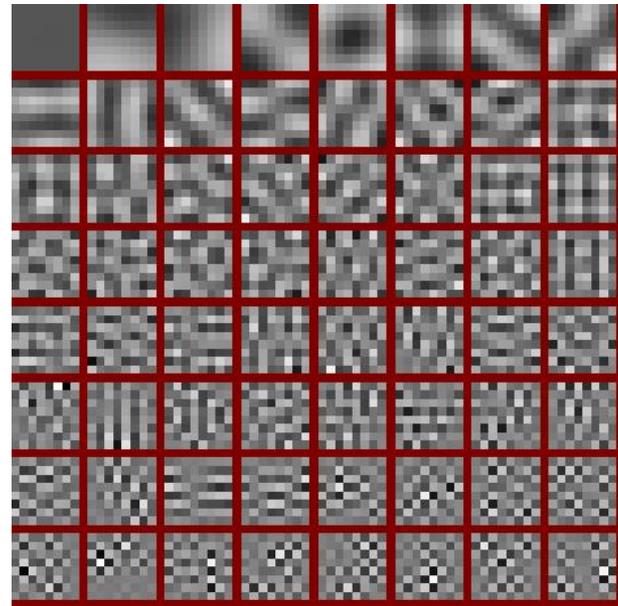
On doit donc minimiser

$$E = \sum_{i=1}^n \left\| V_i - \sum_{j=1}^m \langle V_i | v_j \rangle v_j \right\|^2$$

Problème très proche à celui du TD

# Choix de la base

- On procède comme dans le TD et on trouve que la meilleure base est celle des vecteurs propres o.n. de la matrice  $A = \sum_i V_i V_i^T$



# Compression linéaire

- On choisit une base o.n.  $\alpha_n$  ordonnée
- Pour un taux  $\tau = \frac{m}{N}$  on approxime le signal avec :

$$\tilde{x} = \sum_{i=1}^m \langle x | \alpha_i \rangle \alpha_i$$

On parle de compression linéaire car, une fois fixé  $\tau$ ,  $\tilde{x}$  dépend linéairement de  $x$

L'ordre des vecteurs est déterminé par une étude statistique des signaux

# Compression non-linéaire

- Au lieu de prendre toujours les mêmes atomes de la famille  $\alpha$  on prend ceux qui ont le maximum produit scalaire avec  $x$
- On parle aussi de compression adaptative
- Aucune compression de taille  $m$  peut avoir une plus petite erreur de la compression adaptative

# Localisation et plan T/F

- Base locale : compromis entre localisation spatiale et en fréquence
- Le plus petit est le support (p.e. petits blocs dans la DCT-2D), la meilleure est la résolution spatiale, mais on perd en résolution en fréquence
- Pour les images il faudrait une bonne résolution en fréquence aux basses fréquences (caractérisation des motifs, patterns) et une bonne résolution spatiale aux hautes (position des contours)
  - Transformée en ondelettes

# Matching Pursuit

- Permet de trouver les coefficients quand la famille  $\alpha$  n'est pas une base
- Algorithme itératif

# Normes de compression

- JPEG
  - DCT 8x8, quantification uniforme mais avec pas différent pour chaque coefficient, zig-zag scan, codage run-length et codage de Huffman
- JPEG2000
  - Transformée en ondelettes, allocation des ressources entre blocs de coefficients, codage sans pertes avancé
- MP3
  - Analyse du signal audio, quantification des coefficients DCT avec modèle psycho-acoustique

# Ce qu'il manque...

- Quantification vectorielle
- Compression sans pertes
- Prédiction spatiale et temporelle
- Compression perceptuelle
- Les ondelettes
- Les normes
- La vidéo et l'estimation du mouvement
- La vidéo 3D et l'estimation de disparité ...

Ça continue en SI222 et SI350 !