

# Chapitre 1

## TRAITEMENT DE SIGNAL MULTIMEDIA

### Numérisation vidéo

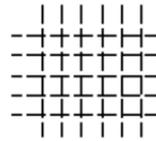
#### *Échantillonnage et quantification*

# Échantillonnage et Quantification

- Les systèmes de traitement numériques des images manipulent des données de type « discrètes »

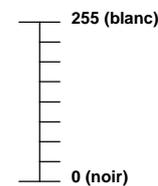
- **Échantillonnage**

- Échantillonnage des valeurs de l'image aux nœuds d'une grille rectangulaire disposée dans le plan de l'image
- Un pixel  $(m, n)$  (*picture element*) représente la valeur du niveau de gris de l'image monochrome au point de la grille indexé par les coordonnées entières  $(m, n)$



- **Quantification**

- Transformation d'une amplitude à valeurs continues en une amplitude qui ne peut prendre qu'un nombre fini de valeurs distinctes
- Pour une image à 256 niveaux de gris, chaque amplitude est codée sur un octet (8 bits)



Nous avons brièvement abordé au cours de la ressource précédente (Traitement d'images : Exemples d'applications), le fait que la numérisation d'une image suppose une double discrétisation :

- D'une part du domaine de définition de l'image : le domaine spatial 2-D pour les images fixes (respectivement le domaine spatial 2-D et le temps pour les images animées) : il s'agit de l'*échantillonnage*.
- D'autre part de l'amplitude du signal image : du signal de luminance dans le cas des images monochromes, des trois signaux « couleur » dans le cas des images couleurs (en représentation « Rouge, Vert, Bleu » : RVB ou « Luminance, Chrominance1, Chrominance2 » : Y, Cr, Cb) : il s'agit de la *quantification*.

On parlera donc d'images numériques (ou numérisées) seulement dans le cas où cette double discrétisation aura été effectuée : en espace et en amplitude. La représentation de base (appelée aussi canonique) d'une image correspondra donc à un tableau 2-D, dont chaque élément correspond à un pixel. Dans le cas d'une image monochrome, chaque pixel sera codé par exemple sur 8 bits et pourra donc ainsi prendre 256 valeurs différentes (effet de la quantification). Dans le cas d'une image couleur, le pixel aura trois composantes qui représenteront suivant le modèle choisi : les composantes (R, V, B) ou les composantes (Y, Cr, Cb), selon les cas.

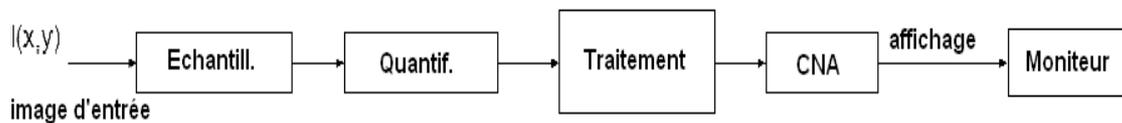
Remarque : si R, V, B prennent chacun leur valeur sur  $[0, 255]$ , et Y aussi, les deux composantes de chrominance Cr et Cb prennent leurs valeurs au départ sur  $[-128, +127]$ . En pratique, on les code en ajoutant un offset de 128, d'où une dynamique également de  $[0, 255]$ .

# Scénario typique d'un Traitement d'Image

## Rappel : Qu'est ce qu'une Image?

- Une fonction  $I(x,y)$  à deux coordonnées spatiales du plan
- Pour obtenir des données utilisables en traitement numérique :
  - *Échantillonnage (spatial) et Quantification des valeurs de luminance*
  - *Peut être effectué par des capteurs*  
*e.g. Charge-coupled Device (capteur CCD)*

## Scénario typique du traitement d'image



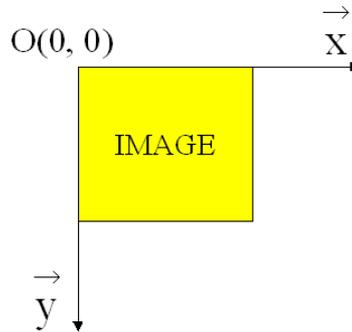
L'image ci dessus présente la chaîne complète utilisée pour effectuer un traitement d'image quelconque :

- L'image est représentée par une fonction de deux variables, ces variables correspondent aux coordonnées spatiales  $(x, y)$ .
- La première fonction est *Echantill* : on échantillonne le signal d'image (discrétisation des coordonnées spatiales).
- Le résultat est ensuite injecté dans la fonction *Quantif* : très souvent pour une image monochrome, on fixe 256 niveaux de luminance.
- L'image est maintenant numérisée, on peut donc appliquer le traitement d'image (*cf. Traitement*) que l'on souhaite : binarisation, augmentation du contraste, ...
- L'image numérique traitée est finalement envoyée vers un convertisseur Numérique/Analogique (CNA) pour obtenir un signal affichable sur un moniteur.

La chaîne fonctionnelle typique d'un traitement d'image nécessite donc un traitement pour numériser les images que l'on souhaite ensuite traiter. Nous avons vu que ce traitement correspond à un échantillonnage suivi d'une quantification. Ces deux notions ont pu être abordées en traitement du signal et en électronique. Cependant, nous allons voir que la conception et la réalisation de ces deux étapes en imagerie est sensiblement différente notamment parce que les images sont des signaux 2-D au niveau spatial et que leur représentation en image couleur nécessite de considérer trois numérisations différentes.

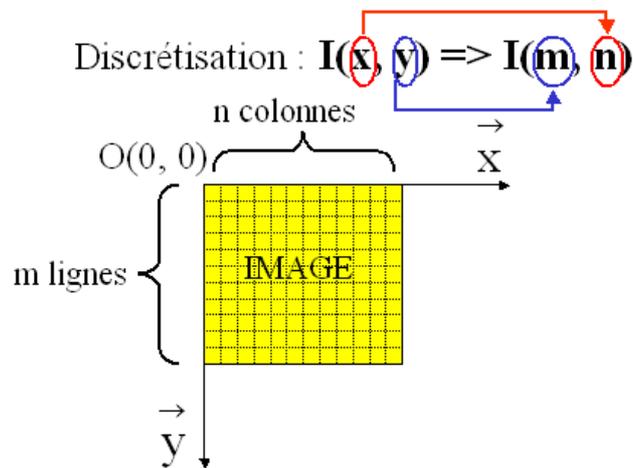
Deux remarques importantes :

1 – Le signal d'entrée  $I(x, y)$  est une image 2-D. En image, les axes (Ox) et (Oy) sont représentés et orientés comme sur la figure suivante :



**L'axe (Oy) est donc dirigé du haut vers le bas de l'image.**

2 – Une fois ces deux axes discrétisés, on obtient « m » lignes et « n » colonnes :

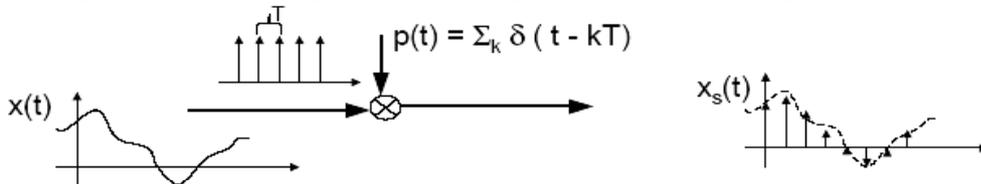


En traitement d'images, **on parle alors d'un signal d'image  $I(m, n)$**  et non pas  $I(n, m)$ . **L'ordre des coordonnées est donc modifié** puisque « m » correspond à la discrétisation selon (Oy) et « n » selon (Ox).

## Rappel : Échantillonnage 1-D

- **Domaine temporel**

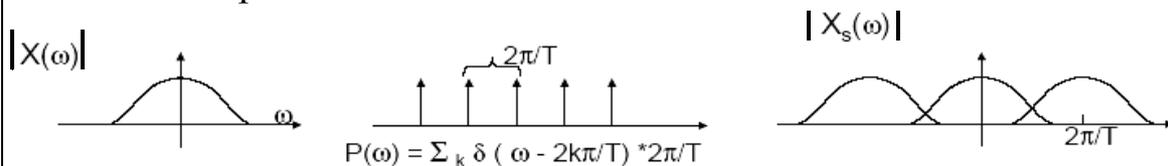
- Multiplication du signal continu par un train d'impulsion



- **Domaine fréquentiel**

- Dualité : échantillonner dans un domaine  $\Leftrightarrow$  rendre le signal périodique dans l'autre domaine

- la Transformée de Fourier d'un train d'impulsion est un train d'impulsion



Avant de pouvoir passer à l'échantillonnage d'un signal 2-D, il est nécessaire de faire quelques rappels sur l'échantillonnage des signaux 1-D. Le type d'échantillonnage le plus répandu en électronique et en traitement du signal est l'échantillonnage temporel. La figure ci-dessus présente les différentes conséquences engendrées par un tel traitement dans les domaines temporels et fréquentiels.

*Note :  $\omega = 2\pi f$ , avec  $\omega$  : pulsation en  $\text{rad.s}^{-1}$  et  $f$  : fréquence en Hz*

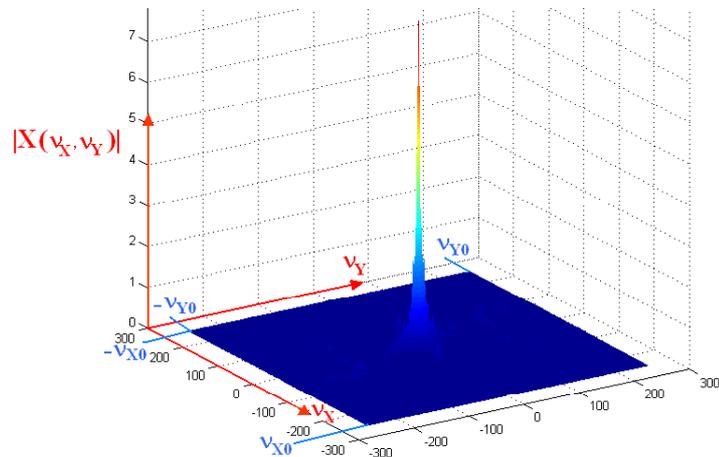
- Dans le domaine temporel : en partant d'un signal « x » dépendant du temps, il faut choisir une période d'échantillonnage « T ». Le signal échantillonné  $x_s(t)$  sera donc représenté par ses valeurs aux instants  $t = T$ ,  $t = 2T$ ,  $t = 3T$ , ... Pratiquement pour réaliser un tel échantillonnage, il faut réaliser le produit du signal analogique  $x(t)$  avec un train d'impulsion  $p(t)$ . Ce train d'impulsion correspond à la somme d'impulsion de Dirac aux instants  $kT$ ,  $k$  étant un entier naturel.
- Dans le domaine fréquentiel : il y'a une dualité entre le domaine temporel et le domaine fréquentiel. En effet, le fait d'échantillonner un signal dans l'un des domaines entraîne la périodisation de ce même signal dans l'autre domaine. Ici  $X(\omega)$  (resp.  $P(\omega)$ ) représente le spectre fréquentiel de  $x(t)$  (resp.  $p(t)$ ). L'échantillonnage de période  $T$  dans le domaine temporel a donc entraîné la périodisation en fréquence (période  $1/T$ ) du spectre dans le domaine fréquentiel (voir représentation de  $X_s(\omega)$ ).

Cette dualité entre les deux domaines peut engendrer des effets de repliement dans le spectre du signal échantillonné, certaines contraintes sont donc à satisfaire.

## Extension à l'échantillonnage 2-D (1)

- Signal 2-D à bande limitée
  - sa transformée de Fourier est nulle en dehors de la région de fréquence spatiale :  $|v_X| < v_{X0}$  et  $|v_Y| < v_{Y0}$

Exemple : l'image *Barbara* et son spectre



Les problèmes posés par l'échantillonnage des signaux bidimensionnels sont a priori similaires à ceux des signaux monodimensionnels. Le critère de Nyquist s'applique de la même manière. Toutefois, une interprétation dans un espace à deux dimensions est nécessaire pour mieux comprendre les phénomènes liés à l'échantillonnage, comme les effets de la géométrie du motif d'échantillonnage.

La figure ci dessus représente un exemple de transformée de Fourier d'une fonction 2-D à bande limitée (image *Barbara*). On peut dans ce cas travailler avec un spectre borné. Le spectre s'étale sur un plan fréquentiel défini par deux vecteurs de base.

## Extension à l'échantillonnage 2-D (2)

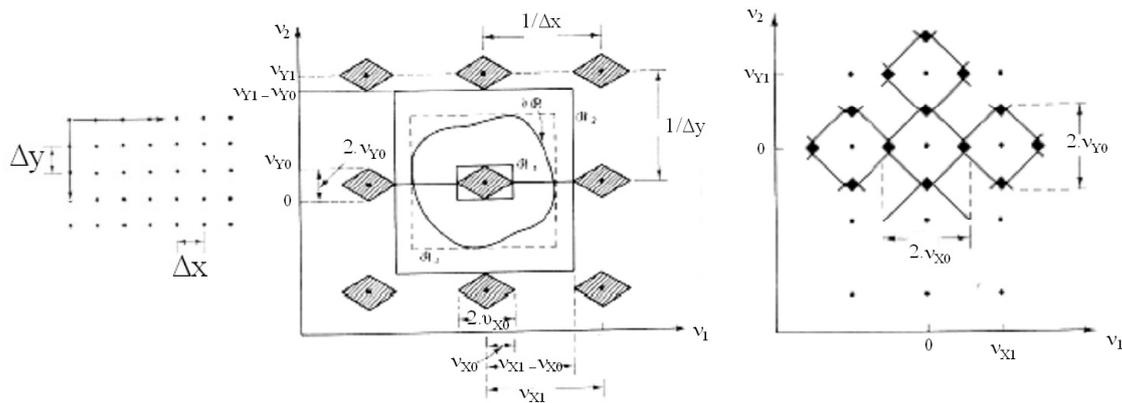
- **Train d'impulsion 2-D : fonction comb**

- $\text{comb}(x, y; \Delta x, \Delta y) = \sum_{m,n} \delta(x-n.\Delta x, y-m.\Delta y)$

- FT  $\Rightarrow \text{COMB}(v_x, v_y) = \text{comb}(v_x, v_y; 1/\Delta x, 1/\Delta y) / \Delta x.\Delta y$

- **Échantillonnage et Réplication**

- Critère de Nyquist ( $2.V_{x0}$  et  $2.V_{y0}$ ) et Aliasing



Les variations selon  $x$  ( $\Delta x$ ) et les variations selon  $y$  ( $\Delta y$ ) vont définir le plan dans lequel se trouve l'image 2-D. La structure d'échantillonnage est régulière (figure de gauche) donc l'arrangement des points échantillonnés (pixels) est régulier dans le plan  $(x, y)$ . La position du point de rang  $(m, n)$  est donc donnée par :  $n\Delta x + m\Delta y$ .

Dans le domaine fréquentiel, les points sont définis par leurs coordonnées « fréquentielles »  $v_x$  et  $v_y$ . En prenant une fonction  $f(x, y)$ , à laquelle on associe le spectre d'amplitude  $F(v_x, v_y)$ , après échantillonnage on aura :

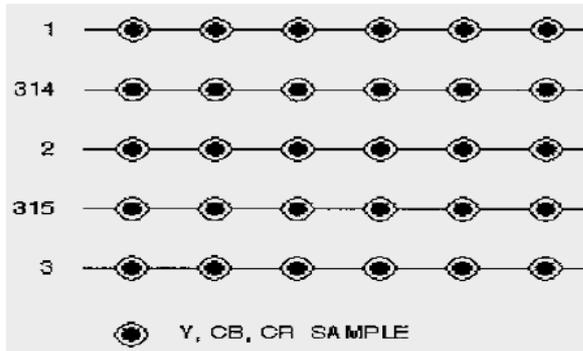
- $f_e(x, y) = f(x, y) \cdot \sum_m \sum_n \delta(x-n\Delta x, y-m\Delta y)$

- $F_e(v_x, v_y) = \left( \frac{1}{\Delta x.\Delta y} \right) \sum_m \sum_n F(v_x-n\Delta v_x, v_y-m\Delta v_y)$

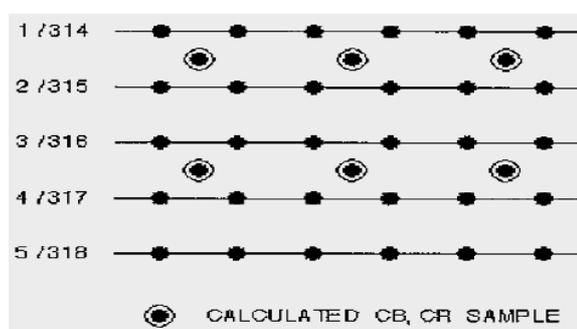
Pour un signal d'image à spectre borné, on peut alors retrouver l'image d'origine (non échantillonnée) à partir de l'image échantillonnée idéalement si les spectres du signal d'origine ne se chevauchent pas (figure centrale en opposition avec l'image de droite). Pour cela on utilise un interpolateur  $R$  de gain fréquentiel constant (fréquences spatiales), égal à  $\Delta x.\Delta y$  sur le domaine de  $F$  et nul à l'extérieur de ce domaine, tel que celui-ci n'ait aucun recouvrement avec les translats de  $F$ . On a alors un interpolateur linéaire bidimensionnel idéal : c'est le filtre d'interpolation spatiale de Nyquist.

# Structures d'échantillonnage en images et vidéos couleur

Structure 4 : 4 : 4



Structure 4 : 2 : 0



## Structure d'échantillonnage orthogonal

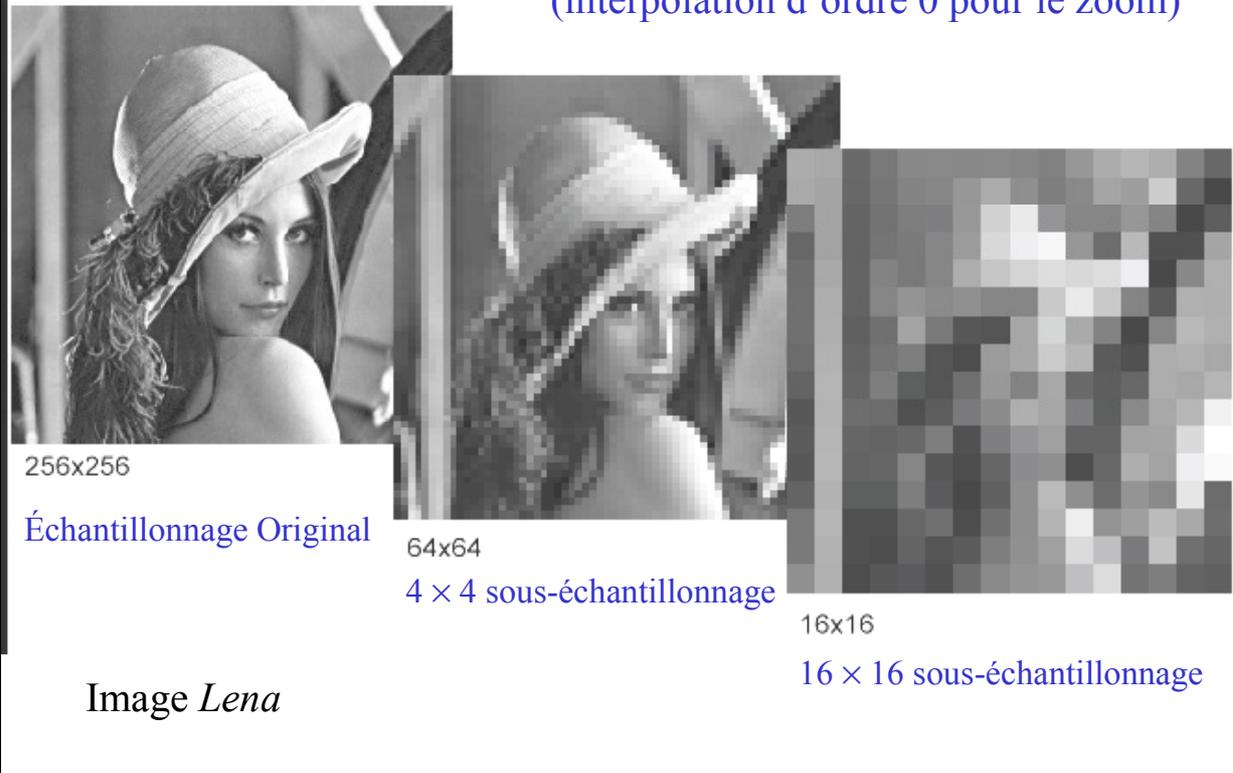
La figure ci-dessus présente deux structures usuelles d'échantillonnage orthogonal d'images couleurs :

- Une structure 4 : 4 : 4 : Pour chaque pixel, on dispose de la composante de luminance et des deux composantes de chrominance Cr et Cb. Il n'y a donc dans ce cas aucun sous-échantillonnage des composantes de chrominance.
- Une structure 4 : 2 : 0 : Pour chaque pixel, on dispose d'une composante de luminance (Y) et seulement des deux composantes de chrominance Cr et Cb pour un groupe de 2x2 pixels de luminance. La résolution de la chrominance est divisée d'un facteur 2 suivant les deux dimensions spatiales. Ce format (utilisé sur les DVD) réduit les résolutions verticale et horizontale.

Pour ces structures d'échantillonnage, nous sommes passés au préalable au sous échantillonnage de la représentation RVB de l'image à la représentation Y, Cr, Cb. La réduction de la fréquence d'échantillonnage spatiale des deux composantes de chrominance est rendue possible du fait que, dans les images naturelles, les deux composantes ont une largeur de bande en fréquences spatiales, beaucoup plus réduite que celle de la luminance.

## Exemples d'échantillonnage d'image

(interpolation d'ordre 0 pour le zoom)



Sur cette figure, on a représenté l'image « *Lena* » échantillonnée avec deux structures d'échantillonnage différentes. L'image à gauche est l'image de référence (dimensions spatiales : 256×256 pixels). La seconde image est échantillonnée avec une cadence d'échantillonnage quatre fois moindre suivant chacune des deux dimensions spatiales. Elle est donc de dimension 64×64 pixels. Elle est cependant ramenée, pour des besoins de visualisation, à la même grandeur que l'image d'origine grâce à un zoom. Celui-ci est en fait une interpolation d'ordre 0 (chaque pixel est dupliqué 16 fois pour former à l'affichage sur l'écran un carré de 4×4 pixels identiques). Un effet de pixellisation apparaît clairement, mais il est encore possible de distinguer les différents éléments de l'image (le visage, le chapeau, ...). La troisième image (à droite) présente le résultat après un sous-échantillonnage de 16 par rapport à l'image de référence suivant les deux directions spatiales. Toujours pour visualiser à la même taille, on a utilisé un zoom d'un facteur 16 (interpolation d'ordre 0 engendrant pour chaque échantillon une reconstruction par 16×16 pixels de même amplitude). L'image est alors très fortement pixélisée et on ne distingue quasiment plus les objets présents dans la scène.

Après s'être intéressé à l'échantillonnage bidimensionnel d'un signal 2-D (dans le cas présent une image), il faut maintenant - pour obtenir l'image numérisée finale - déterminer comment procéder à la quantification de l'image échantillonnée.

# **Numérisation vidéo**

## *Quantification*

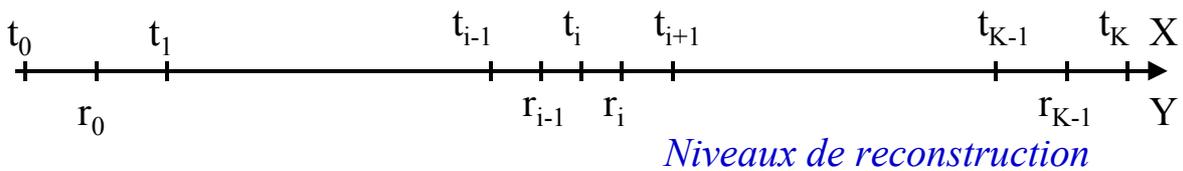
# Quantification

- **Quantification scalaire**



- Quantification scalaire :

*Seuils de décision*



- **Fonction de Quantification**

$$\begin{aligned} \text{pour } t_j \leq x < t_{j+1} & \quad \text{alors } y = r_j \quad \forall j = 0, \dots, K-1 \\ \text{et } t_0 = x_{\min} ; t_K = x_{\max} \end{aligned}$$

L'image est à présent échantillonnée dans le domaine spatial. Elle est composée d'un ensemble de pixels. Il convient maintenant de réaliser une quantification pour que chaque amplitude du signal d'image soit représentée par une valeur approchée. On choisit donc un ensemble de valeurs prédéfinies appelées niveaux de reconstruction, réparties sur la dynamique  $[x_{\min}, x_{\max}]$  du signal  $x$  à quantifier.

Sur la figure ci-dessus, ces niveaux de reconstruction sont définis par la suite  $(r_i)_{i \in [0, K-1]}$  constituée de  $K$  valeurs. Ces niveaux sont accompagnés de seuils de décision définis par la suite  $(t_i)_{i \in [0, K]}$  avec  $t_0 = x_{\min}$  et  $t_K = x_{\max}$ . Prenons par exemple le cas d'un signal d'amplitude continue «  $x$  » en entrée du quantificateur, le signal quantifié «  $y$  » en sortie sera défini par la condition:

$$\forall i \in [0..K-1], \text{ si } t_i \leq x < t_{i+1}, \text{ alors } y = r_i \text{ et } t_0 = x_{\min} ; t_K = x_{\max}$$

Il existe de multiples façons de quantifier. En effet, la manière de choisir les niveaux de reconstruction et les seuils de décision n'est pas forcément de type répartition uniforme sur  $[x_{\min}, x_{\max}]$ , et dépend fortement des caractéristiques du signal d'image à quantifier.

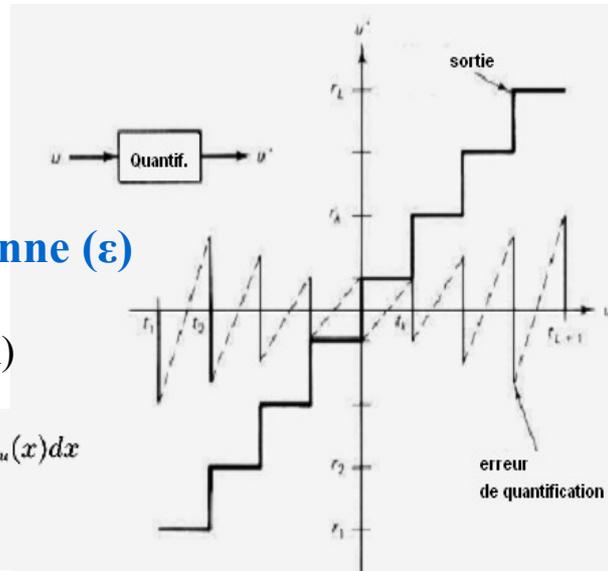
## Exemple de loi de quantification

- Multiple-to-one mapping
- Besoin de minimiser l'erreur

- **Erreur quadratique moyenne ( $\epsilon$ )**

- Cas d'une entrée « u » avec une densité de probabilité  $p_u(x)$

$$\begin{aligned} \epsilon &= E[(u - u')^2] = \int_{t_1}^{t_{L+1}} (x - u'(x))^2 p_u(x) dx \\ &= \sum_{i=1}^L \int_{t_i}^{t_{i+1}} (x - r_i)^2 p_u(x) dx \end{aligned}$$



Le graphique ci-dessus donne un exemple de loi de quantification. L'entrée du quantificateur correspond au signal « u », et la sortie au signal « u' ». Les seuils de décision sont donnés par la suite  $(t_i)_i$  et les niveaux de reconstruction sont donnés par la suite  $(r_i)_i$ . La quantification est un traitement qui engendre des pertes irréversibles contrairement à certaines formes de l'échantillonnage vu précédemment. Cependant, selon le paramétrage de la quantification, l'œil humain ne détectera pas forcément ces erreurs. Il faut donc essayer de les minimiser. Typiquement, on cherche à minimiser l'erreur quadratique moyenne. Par définition cette erreur quadratique  $\epsilon^2$  est donnée par la relation :

$$\epsilon^2 = E[(u - u')^2] = \int_{t_1}^{t_{L+1}} (x - u'(x))^2 \cdot p_u(x) \cdot dx$$

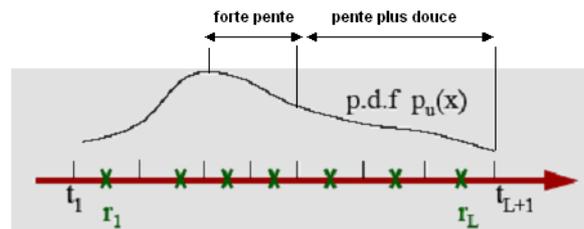
Où : E désigne l'espérance mathématique et  $p_u(x)$  la densité de probabilité de u.

En décomposant l'intégrale suivant les relation de Chasles, on obtient finalement :

$$\epsilon^2 = \sum_{i=1}^L \int_{t_i}^{t_{i+1}} (x - r_i)^2 \cdot p_u(x) \cdot dx$$

La quantification doit donc être choisie de façon à minimiser l'erreur quadratique moyenne, mais nous allons voir que d'autres critères liés aux caractéristiques de l'image entrent également en compte.

## Quel type de loi de quantification utiliser?



- Allouer plus de valeurs dans les domaines de valeurs les plus probables
- Minimiser l'erreur au sens des probabilités

- EQM minimale (MMSE en anglais) :

$$\begin{aligned}\mathcal{E} &= E[(u - u')^2] = \int_{t_1}^{t_{L+1}} (x - u'(x))^2 p_u(x) dx \\ &= \sum_{i=1}^L \int_{t_i}^{t_{i+1}} (x - r_i)^2 p_u(x) dx\end{aligned}$$

- Grande EQM  $\Rightarrow$  Fortes pertes
  - la minimalisation de l'EQM passe par l'utilisation d'outils mathématiques
- Problème à résoudre lors d'une optimisation
    - Quels  $\{t_k\}$  et  $\{r_k\}$  utiliser? Loi de quantification optimisée si :  
dérivées partielles de l'EQM égales à 0

La question posée ici est de savoir quels sont les critères à prendre en compte pour bien choisir une loi de quantification. La figure ci-dessus représente un quantificateur  $Q$  dont l'entrée est le signal «  $u$  », et la sortie le signal «  $u'$  ». Les seuils de décision sont donnés par la suite  $(t_i)$  et les niveaux de reconstruction sont donnés par la suite  $(r_i)$ . La fonction de densité de probabilité  $p_u(x)$  est également représentée. Dans cet exemple le quantificateur n'est plus linéaire : les seuils et les niveaux de reconstruction ne sont pas disposés régulièrement. En effet, en observant attentivement la densité de probabilité de «  $u$  », on s'aperçoit que la pente de la fonction est plus faible (ou plus forte) pour certaines valeurs de  $t_i$  que d'autres. Lorsque la pente est faible, on peut raisonnablement penser qu'il faut peu de niveaux de reconstruction pour représenter la plage de valeurs décrite. Inversement, quand la pente de la courbe devient plus raide, de nombreuses valeurs sont atteintes en peu de temps, et il faut donc choisir beaucoup plus de niveaux de reconstruction. D'autre part, nous avons vu que pour choisir la loi de quantification il fallait également minimiser l'erreur quadratique moyenne dont l'expression dépend directement des paramètres  $(t_i)$  et  $(r_i)$ . L'optimisation de la quantification passe donc par le choix des seuils de décision et des niveaux de reconstruction. Ces paramètres peuvent être directement déterminés en minimisant l'erreur quadratique moyenne (calcul de dérivées partielles, ...).

# Quantification Optimale (1)

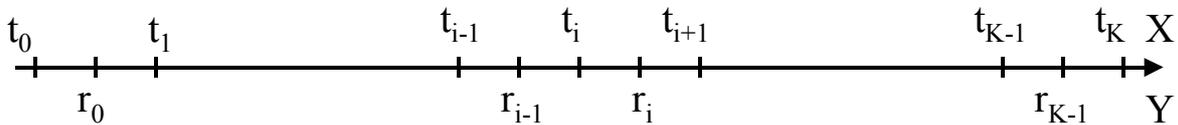
- Critère de quantification optimale

– Erreur Quadratique Moyenne (  $y = Q(x)$  )

$$\epsilon^2 = E \{ (X-Y)^2 \} = \int_{-\infty}^{+\infty} (x-y)^2 p_x(x) dx$$

$$\epsilon^2 = \sum_i \int_{t_i}^{t_{i+1}} (x-y_i)^2 p_x(x) dx$$

$$\text{minimisation de } \epsilon^2: \begin{cases} t_{i+1} = (r_i + r_{i+1}) / 2 \\ r_i = E \{ X / x \in [t_i, t_{i+1}] \} \end{cases}$$



- **Cas spécial** : fonction de densité de probabilité uniforme :  $p_x(x)$  est constant alors  $t_{i+1} = (r_i + r_{i+1}) / 2$  et  $r_i = (t_i + t_{i+1}) / 2$

Le critère de quantification optimale est lié à la minimalisation de l'erreur quadratique moyenne. L'entrée du quantificateur correspond au signal X, la sortie au signal Y. Les niveaux de reconstruction sont donnés ici par  $(r_i)$ , et les seuils de décision par  $(t_i)$ . Nous allons donc chercher à déterminer les relations qui découlent de ce critère d'optimisation et de minimalisation:

Pour minimiser  $\epsilon^2$ , on calcule ses dérivées partielles :

- Par rapport à  $t_i$  :  $\frac{\partial \epsilon^2}{\partial t_i} = p_x(t_i) \cdot (t_i - r_{i-1})^2 - p_x(t_i) \cdot (t_i - r_i)^2$
- Par rapport à  $r_i$  :  $\frac{\partial \epsilon^2}{\partial r_i} = -2 \cdot \int_{t_i}^{t_{i+1}} p_x(x) \cdot (x - r_i) \cdot dx$

En annulant la première dérivée partielle, on obtient la relation :  $t_{i+1} = \frac{r_i + r_{i+1}}{2}$

En annulant la seconde dérivée partielle, on obtient :

$$\begin{aligned} \int_{t_i}^{t_{i+1}} p_x(x) \cdot (x - r_i) \cdot dx = 0 &\Leftrightarrow \int_{t_i}^{t_{i+1}} p_x(x) \cdot x \cdot dx = \int_{t_i}^{t_{i+1}} p_x(x) \cdot r_i \cdot dx \\ &\Leftrightarrow E\{X; x \in [t_i, t_{i+1}]\} = r_i = \int_{t_i}^{t_{i+1}} p_x(x) \cdot dx = r_i \end{aligned}$$

Ainsi, la minimisation de l'erreur quadratique moyenne (quantification optimale) implique que les seuils de décision  $d_i$  et les niveaux de reconstruction  $q_i$  soient donnés par les

$$\text{relations : } \begin{cases} d_{i+1} = \frac{q_i + q_{i+1}}{2} \\ q_i = E\{X; x \in [d_i, d_{i+1}]\} \end{cases}$$

Ce sont les deux relations obtenues par Max. Ces dernières montrent que le quantificateur optimal d'un signal à loi de probabilité non-uniforme n'est pas un quantificateur linéaire. Dans le cas particulier d'un signal d'entrée dont la densité de probabilité est constante sur

$[x_{\min}, x_{\max}]$ , on a :  $p_X(x) = \frac{1}{(x_{\max} - x_{\min})}$ , et on obtient alors les relations:

$$\begin{cases} r_i = \frac{t_i + t_{i+1}}{2} \\ t_{i+1} = \frac{r_i + r_{i+1}}{2} \end{cases}$$

Le quantificateur optimal est, dans ce cas, linéaire :  $\Delta t_i = \frac{x_{\max} - x_{\min}}{K}$ .

Pour un signal normalisé sur  $[0, 1]$ , l'erreur quadratique minimisée vaut :  $\epsilon^2 = \frac{1}{12.K^2}$ .

Nous avons vu que pour optimiser le quantificateur, il fallait, avec ce critère, minimiser l'erreur quadratique moyenne de quantification. Naturellement, d'autres critères peuvent être utilisés, en particulier en prenant en compte des aspects visuels.

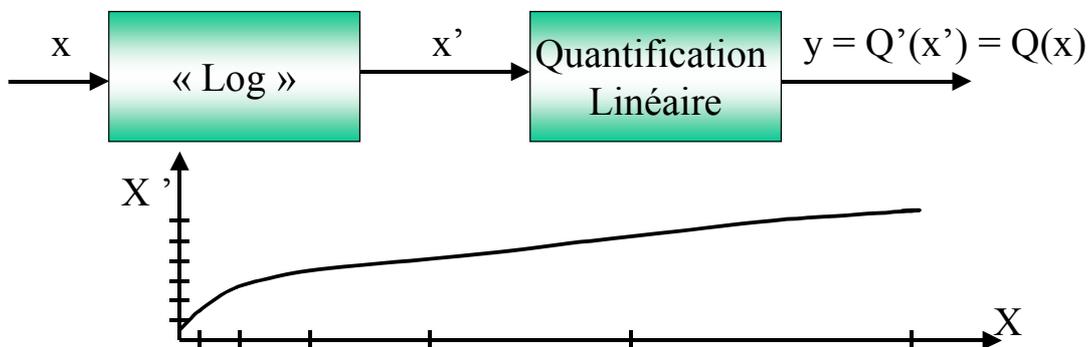
## Quantification Optimale (2)

- **Quantification à entropie maximale** :  $H(Y)$  maximum si  $Y$  est à loi de probabilité constante

$$\Rightarrow p_i = \int_{t_i}^{t_{i+1}} p_x(x) dx = \text{constant} \quad \forall i = 1, \dots, K$$

- **Quantification Psycho-visuelle des images**

- Perception de petits stimulus sur l'arrière plan :  $\Delta L / L = \text{cst}$
- $\text{Cst} \cong 0.01$  dans la gamme des luminances affichées sur écran TV (loi de Weber)



Les caractéristiques du système visuel humain peuvent être utilisées pour concevoir la loi de quantification. Pour une dégradation de forme donnée et d'amplitude  $\Delta L$  fixée, le seuil de visibilité  $\Delta L / L$  est à peu près constant ( $L$  étant la luminance). Dans la gamme des luminances TV, il varie légèrement quand  $L$  passe du noir au blanc. Pour une quantification au seuil de visibilité, on a donc  $r_{i+1} - r_i = \Delta L$ . Il est alors possible d'utiliser une quantification uniforme sur un signal compressé par une non-linéarité « f » (voir schéma bloc ci-dessus). Il faudra bien évidemment effectuer l'opération inverse à la sortie du quantificateur avec une fonction d'expansion.

Cette fonction de compression non linéaire est définie par  $f(x) = \int_{L_0}^x \Delta L(x) dx \approx \lambda \cdot \text{Log}(x / L_0)$ .

En télévision, une compression « f » est déjà utilisée : c'est la correction gamma donnée par  $f(x) = x^{\frac{1}{\gamma}}$  qui compense la non-linéarité des écrans TV de type CRT. On applique ensuite une quantification linéaire sur 8 bits. En télévision couleur, on quantifie usuellement les signaux de luminance et de chrominance séparément. En effet, pour à la perception, l'espace  $(Y, Cr, Cb)$  est plus uniforme que l'espace  $(R, V, B)$ .

On voit donc par les applications du type télévision que malgré les propriétés intéressantes du quantificateur optimal, il peut parfois être plus approprié d'utiliser une fonction compression ponctuelle non-linéaire puis de réaliser une quantification linéaire (même si la loi de probabilité du signal n'est pas uniforme). L'ensemble « fonction de compression – quantification – fonction d'expansion » sera choisi de façon à approximer la loi de quantification optimale.

## Exemples de quantifications d'une image



Cette figure montre des exemples d'une quantification effectuée sur l'image *Lena* :

- Pour l'image de gauche : la quantification est suivie d'un codage binaire naturel avec 8 bits par pixel. Il y'a donc  $2^8 = 256$  niveaux de reconstruction pour la représentation des amplitudes de chaque pixel. C'est le cas typique d'une image monochrome (en niveaux de gris uniquement).
- Pour l'image au centre : la quantification est effectuée avec un codage sur 4 bits par pixel et donc  $2^4 = 16$  niveaux de reconstruction. Les contours sont bien rendus mais les textures moins bien rendues dans certains cas. Ce sont les zones à variation spatiales lentes du signal qui sont les plus dégradées par l'apparition de faux contours (dégradations sur les joues et sur l'épaule).
- Pour l'image de droite : la quantification est effectuée avec un codage sur 2 bits par pixel et donc  $2^2 = 4$  niveaux de reconstruction. Les dégradations observées sur l'image précédente sont encore nettement plus accentuées.

Nous avons présentés les diverses étapes dans la numérisation d'une image : la double discrétisation par échantillonnage spatial puis par quantification. Les images sont maintenant numériques et prêtes à être traitées avec des méthodes appropriées, dépendantes des applications.