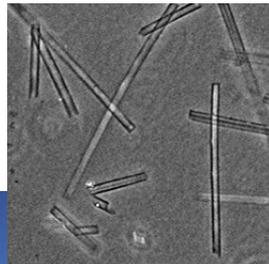


TS200

Compression des signaux

Marc Donias



Plan

- ▶ Débit
 - Définition
 - Calculs
- ▶ Histogramme
 - Définition
 - Calculs
- ▶ Compression
 - Entropie (définition et calculs)
 - Image fixe (standard JPEG)
 - Séquence vidéo (block matching)
- ▶ Analyse comparative de normes de compression d'image fixe (TP évalué)

Débit

▶ Définition

Quantité d'information transmise par unité de temps

$$D = \frac{Q}{T}$$

▶ Unités

- Bits par seconde (bit/s ou bps)
- Bauds (Bd) : nombre de symboles transmis par seconde
 - ✓ Usage souvent abusif : correct pour un signal à 2 valeurs possibles

Débits de données/calculs

Robot humanoïde articulé

Caméra stéréoscopique

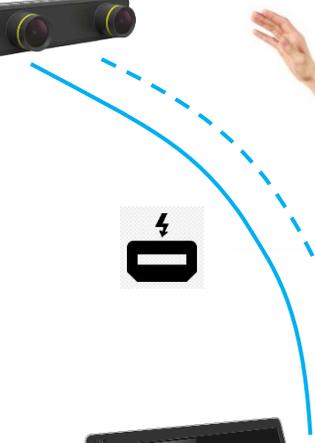
Casque VR

Images
Vidéos
Nuages de points
Modèles 3D
Instructions
Calculs

Stations
de calcul

Stockage SSD

Borne WiFi



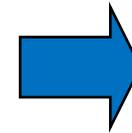
Lien(s) confus entre bits et octets

- ▶ Définition : 1 octet (o) = 8 bits (bit)

- ▶ Usage « abusif » historique



- 1 Ko = 1024 o (2^{10} octets)
- 1 Mo = 1024 Ko = 1024^2 o
- 1 Go = 1024 Mo = 1024^2 Ko = 1024^3 o



octets à
éviter si
possible
pour les
débits

- ▶ Usages « contradictoires » (capacité de stockage, ...)

- Disques durs : 1 Mo = 1 000 000 o, 1 Go = 1000 Mo, ...
- Disquettes 3"1/2
 - ✓ Simple face, double densité 360 Ko = 360×1024 o
 - ✓ Double face, haute densité 1,44 Mo = $1,44 \times 1000 \times 1024$



- ▶ Standard IEC 60027-2 (1999)

- 1024 o = 1 Kio (kibiocet), 1 048 576 o = 1 Mio (mébioctet), ...

Quelques débits (1 / 3)

TELECOMMUNICATIONS

▶ xG

- 3G UMTS (2000) 1,9 Mbit/s
- 5G (2020) 3 Gbit/s



▶ Filaire

- xDSL (1/2/2+) 512 kbit/s à 100 Mbit/s
- Fibre optique 100 Mbit/s à 8 Gbit/s



▶ Bluetooth

- 1.2 (2003) 720 kbit/s
- 5.4 (2023) 2 Mbit/s



SANS-FIL

▶ Wi-Fi (IEEE 802.11)

- b (Wi-Fi 1, 2007) 11 Mbit/s
- ax (Wi-Fi 6, 2021) 10 Gbit/s



Quelques débits (2 / 3)

INTERFACES EXTERNES

- ▶ USB (connecteurs Type-A, Type-B, Type-C)
 - 1.1 (2016) 12 Mbit/s
 - 3.2 Gen 2x2 (2017) 10 Gbit/s
- ▶ Thunderbolt (Mini Display Port/USB Type-C)
 - 1.0 (2007) 10 Gbit/s
 - 4.0 (2020) 40 Gbit/s



INTERFACES INTERNES

- ▶ SATA
 - I (2003) 1,5 Gbit/s
 - III (2009) 6 Gbit/s
- ▶ PCI Express (x1 à x16)
 - 1.0 (2003) 250 Mo/s à 4 Go/s
 - 6.0 (2022) 7,56 Go/s à 721 Go/s



MEMOIRES CPU/GPU

- ▶ SDRAM (bus 64 bits)
 - SDR PC66 (1993) 528 Mo/s
 - DDR5-8800 (2024) 70400 Mo/s
- ▶ GDRAM (bus 384 bits)
 - GDDR3 (2004) 6400 Mo/s
 - GDDR6 (2018) 512 Go/s



PERIPHERIQUES DE STOCKAGE

- ▶ HDD ~ 200 Mo/s
- ▶ Clé USB ~ 500 Mo/s
- ▶ SSD ~ 1000 Mo/s



Quelques débits (3 / 3)

MULTIMEDIA

▶ Audio/Data

- CD-Audio (1982)  172 ko/s
- CD-R (700 MB, 52x)
 - ✓ Audio 8,75 Mo/s
 - ✓ Data 7,62 Mo/s
- MP3 (1993)  128 à 384 kbit/s

▶ Vidéo/Data

- DVD-Video (1998, A/V)  9,83 Mbit/s
- DVD-R (4,7 Go, 22x)
 - ✓ Data 29,7 Mo/s
- Blu-ray (2006, 25/50 Go) 
 - ✓ SD (MPEG-2) 5 Mbit/s
 - ✓ HD (H.264) 18 Mbit/s
- Blu-ray Ultra-HD (2016, H.265)
 - ✓ 50 Go 82 Mbit/s
 - ✓ 66 Go 108 Mbit/s
 - ✓ 100 Go 128 Mbit/s

Facteur et taux (de compression)



$$Q_{finale} \leq Q_{initiale}$$

► Facteur de compression

$$F = \frac{Q_{initiale}}{Q_{finale}} \quad \text{Supérieur à 1, idéalement infini}$$

► Taux de compression (exprimé en %)

$$T = 100 \times \frac{Q_{initiale} - Q_{finale}}{Q_{initiale}} = 100 \left(1 - \frac{1}{F} \right) \quad \text{Positif, idéalement égal à 100\%}$$

Exemples de calcul

▶ Temps de transmission (sans bits de contrôle)

- Image binaire 256×256 , à 2400 Bd
- Image 1024×1024 en 256 niveaux de gris, à 2400 Bd
- Image couleur RGB (8 bits/plan) 512×512 , à
 - ✓ 9600 Bd (réseau GSM)
 - ✓ 56 kBd (réseau V.90)
 - ✓ 10 Mbit/s (réseau Ethernet)

▶ Débit

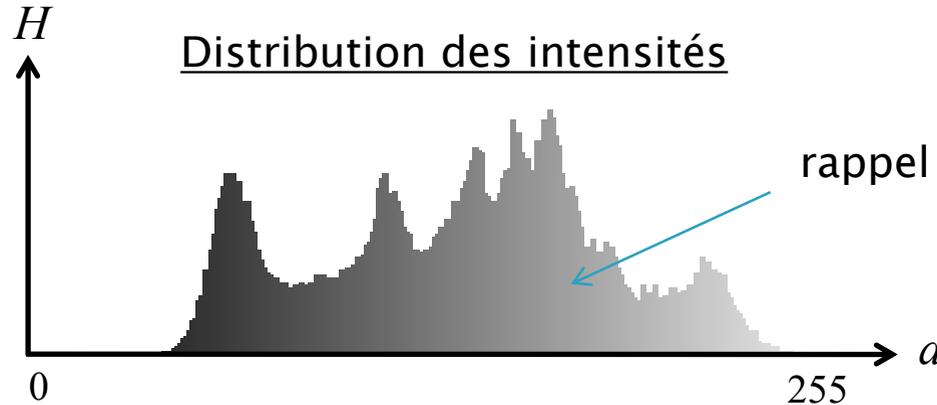
- Séquence d'images couleur RGB (8 bits/plan) 720×576 à 25 images par seconde avec un facteur de compression égal à 100

Histogramme (1 / 3)



Intensité scalaire

(nombre de pixels)



Distribution des intensités

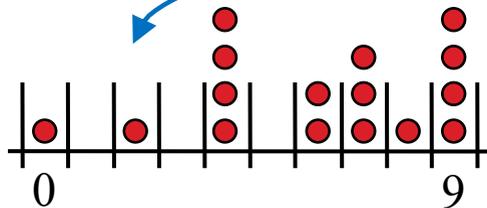
Définition usuelle

rappel de la palette

(valeurs d'intensité)

$$H(a) = \text{Card}(\{x \in I ; I(x) = a\}) \quad a = 0, \dots, 255$$

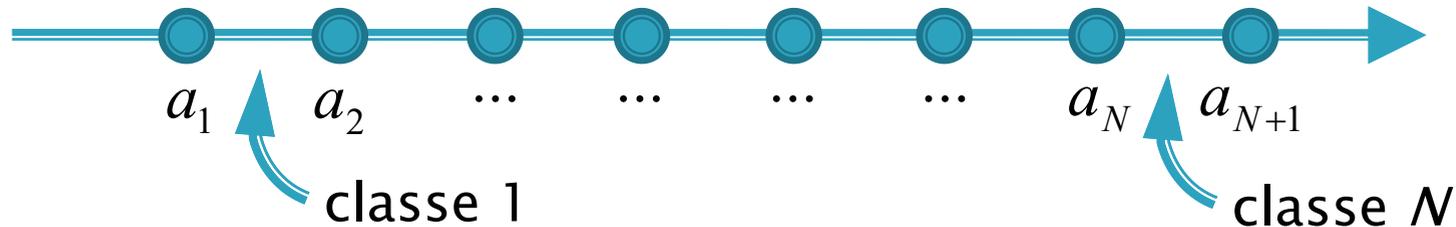
Mode opératoire



Histogramme (2 / 3)

- ▶ Classes (intervalles) d'intensité (index) $a \in [a_1, \dots, a_{N+1}]$

Définition formelle



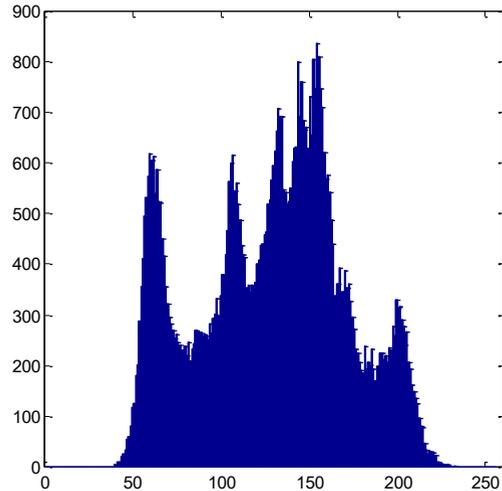
- ▶ Effectifs (distribution des intensités sur N intervalles ou classes)

$$H(i) = \begin{cases} \text{Card}(\{x \in I ; a_i \leq I(x) < a_{i+1}\}) & 1 \leq i < N \\ \text{Card}(\{x \in I ; a_N \leq I(x) \leq a_{N+1}\}) & i = N \end{cases}$$

~ densité de probabilité
(variables aléatoires continues)

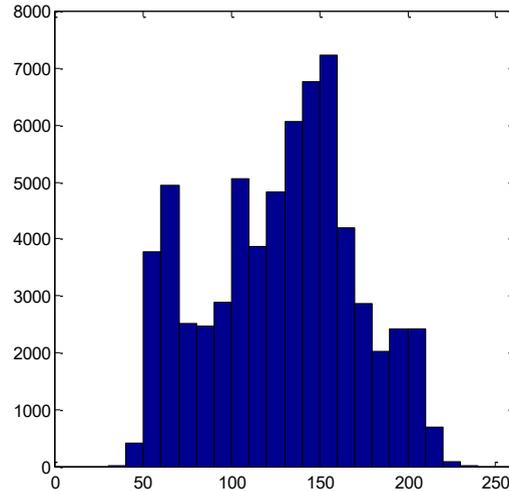
Histogramme (3 / 3)

Impact du nombre de classes



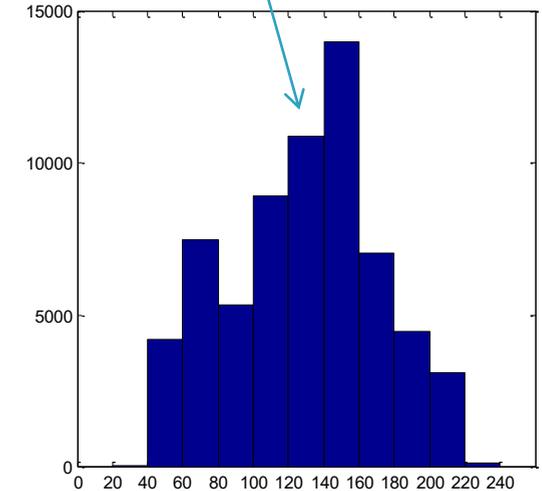
Histogramme de « 1 en 1 »

amplitude de plus en plus forte



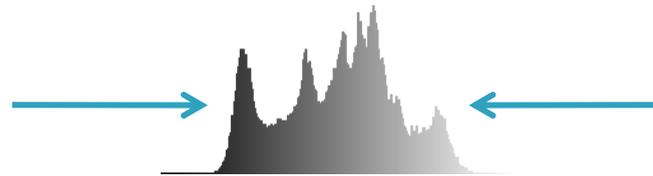
Histogramme de « 10 en 10 »

représentativité de plus en plus faible



Histogramme de « 20 en 20 »

Caractérisation statistique



même histogramme



Calculs d'histogramme

- ▶ Affichage d'une image ('lena.bmp')
- ▶ Calcul et affichage de l'histogramme défini par
 - le nombre de classes
 - les limites de classes
 - fonctions utiles
 - `numpy.histogram`
 - `matplotlib.pyplot.bar`
 - `matplotlib.pyplot.stairs`
 - `(matplotlib.pyplot.hist)`

Entropie (1 / 3)

► Notations

- Dictionnaire et alphabet (symboles) $D = \{C_i\}_{i=1:N}$
- Source (chaîne de symboles à (re)coder) $S = \{C_j\} \quad C_j \in D$

Exemple : AABCABBCBBCDCCCD

► Information ~ Innovation

- Degré de l'information inversement proportionnel à la probabilité d'apparition
- Degré fort (exemple : il neige à Tahiti)
- Degré faible voire nul (exemple : le soleil se lève à l'est)

Entropie (2 / 3)

Quantité (unitaire) d'information

$$Q(C_i) = -k \times \log(p_i)$$

$$k = \frac{1}{\log(K)}$$

nombre d'états différents du codage

$$p_i = \frac{n_i}{M}$$

nombre d'occurrences du symbole

nombre total de symboles de la source

Entropie

$$H(S) = E\{Q(C_i)\}$$

Entropie (3 / 3)

Codage binaire

$$k = \frac{1}{\log(2)} \longrightarrow Q(C_i) = -\log_2(p_i)$$

$$H(S) = - \sum_{i=1}^N p_i \log_2(p_i)$$

bits/symbole



$H_1 = 1$ bit/symbole



$H_2 = 7.56$ bits/symbole

Calculs d'entropie

- ▶ Ecrire une fonction qui mesure l'entropie d'une image A définie par des entiers non signés codés sur 8 bits et l'appliquer à
 - implant.bmp
 - lezard.bmp
- ▶ Ecrire une fonction qui calcule l'entropie de la différence « colonne à colonne » définie par

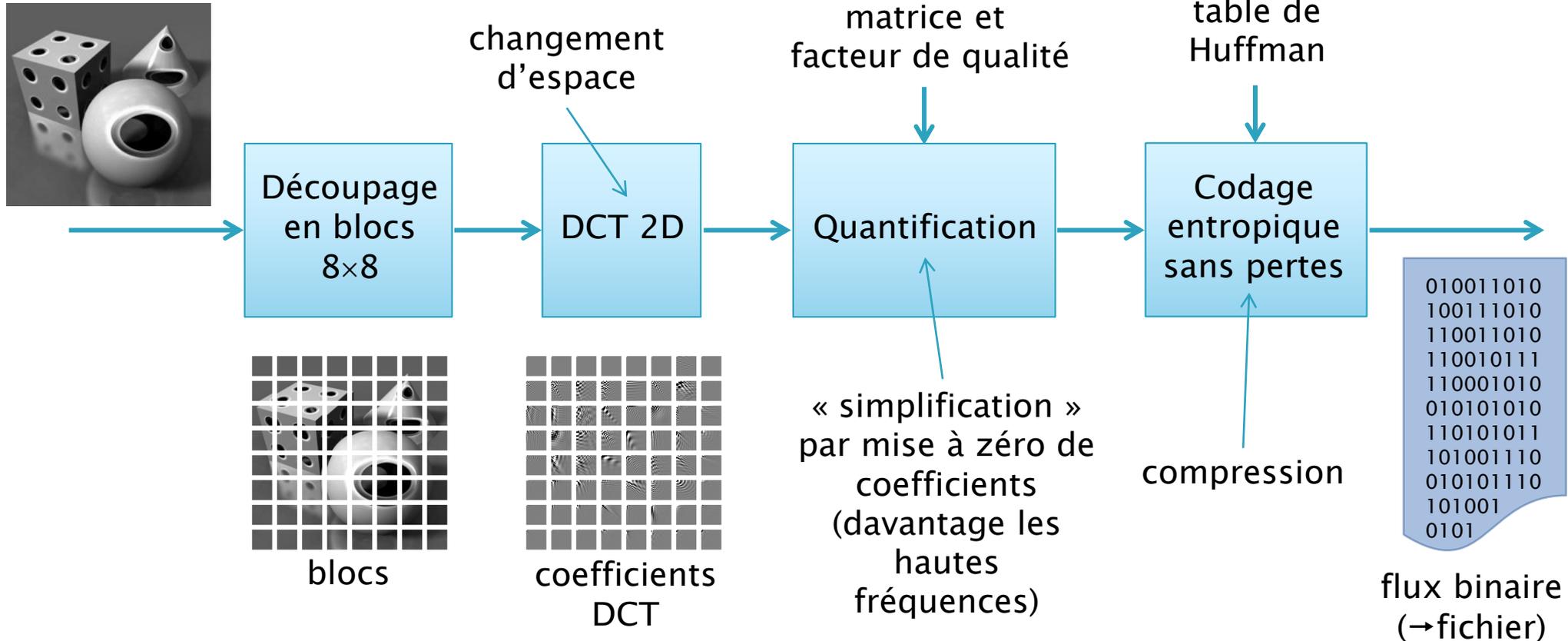
$$B(x, y) = A(x, y) - A(x + 1, y)$$

- ▶ Observer l'histogramme de B et expliquer les résultats obtenus

Compression Jpeg (1 / 3)

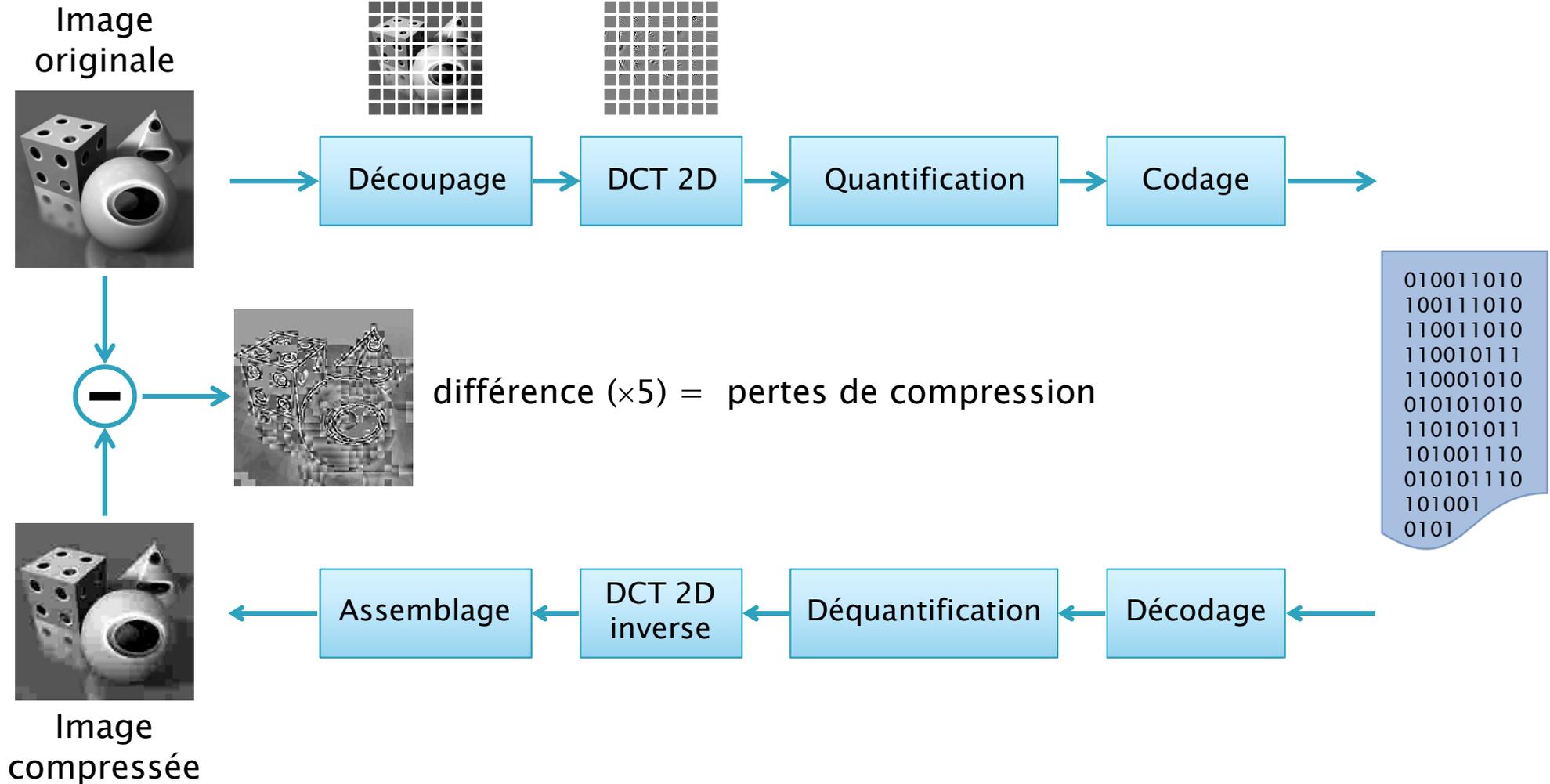
Principe

Codage du contenu fréquentiel « perceptible »

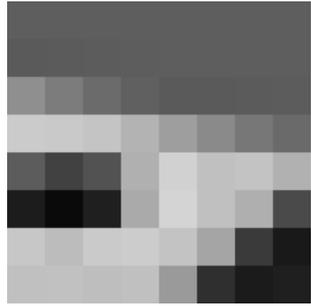


Compression Jpeg (2/3)

Chaîne complète



Compression Jpeg (3 / 3)



Bloc 8×8

94	94	94	94	94	94	94	94
90	91	92	93	94	94	94	94
143	124	107	96	90	90	91	92
203	201	196	179	158	132	105	106
92	65	82	176	207	182	155	177
28	10	31	17	52	175	74	
199	188	202	204	195	165	58	25
192	193	190	192	154	47	27	31

spatial

B

Transformée en cosinus

Changement d'espace et « simplification »

coefficient DC

996,13	75,11	-123,06	50,57	39,88	13,67	-15,19	-1,40
-122,05	-84,56	121,19	-27,83	-31,88	-3,89	10,27	-0,08
-98,84	148,55	6,40	-42,46	-5,07	-19,66	10,27	4,80
-24,01	-218,45	-76,09	67,08	8,07	22,27	10,27	-8,17
70,38	39,24	29,75	-44,52	16,13	10,27	10,27	7,89
73,17	135,83	-0,77	2,11	-30,33	10,27	26,02	-5,54
-89,35	-69,77	-0,04	13,08	12,25	16,44	-26,65	3,62
10,98	-11,74	25,11	1,49	9,46	-8,00	14,95	-2,36

fréquentiel

F

Quantification

entiers faibles et plages de 0 → codage efficace

24	2	-4	1	0	0	0	0
-4	-2	3	0	0	0	0	0
-2	4	0	0	0	0	0	0
0	-5	-1	0	0	0	0	0
1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

fréquentiel

F_Q

partie entière

$$F_Q = E \left[\frac{F}{\alpha \times Q} \right]$$

fonction non-linéaire du facteur de qualité (1 à 99)

matrice (normalisée ou au choix) atténuant une gamme désirée de fréquences

Transformée (discrète) en cosinus

Transformée 1d

$$F(k) = W(k) \sum_{n=0}^{N-1} f(n) \cos\left(\pi k \frac{2n+1}{2N}\right)$$

Transformée directe

base d'exponentielles → base de cosinus

Rappel : transformée de Fourier

$$F(k) = \sum_{n=0}^{N-1} f(n) e^{-j2\pi k \frac{n}{N}}$$

- coefficients réels pour un signal réel
- transformation réversible

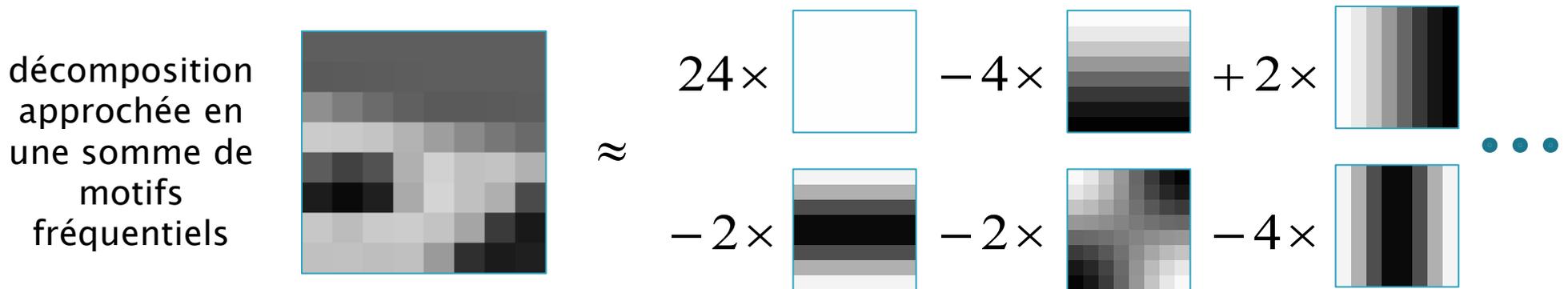
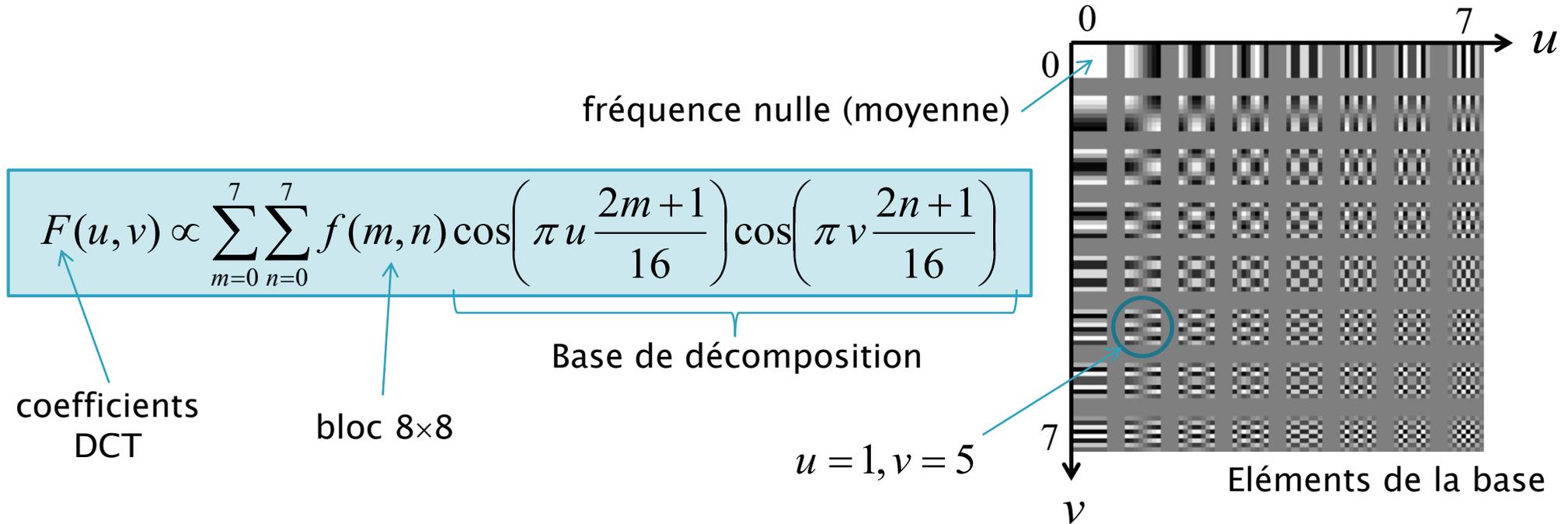
$$f(n) = \sum_{k=0}^{N-1} W(k) F(k) \cos\left(\pi k \frac{2n+1}{2N}\right)$$

Transformée inverse

Transformée 2d

$$F(u, v) = W(u, v) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) \cos\left(\pi u \frac{2m+1}{2M}\right) \cos\left(\pi v \frac{2n+1}{2N}\right)$$

Interprétation fréquentielle



Quantification

$$F_Q = E \left[\frac{F}{\alpha \times Q} \right]$$

$$\alpha = \begin{cases} \frac{50}{q} & \text{si } 1 \leq q \leq 50 \\ \frac{100-q}{50} & \text{si } 50 \leq q \leq 99 \end{cases}$$

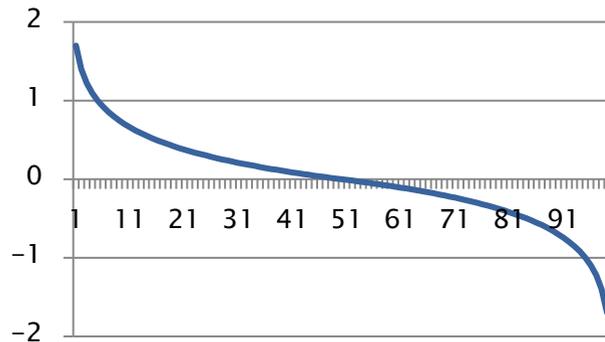
$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Luminance (exemple)

$$Q = \begin{bmatrix} 17 & 18 & 24 & 47 & 99 & 99 & 99 & 99 \\ 18 & 21 & 26 & 66 & 99 & 99 & 99 & 99 \\ 24 & 26 & 56 & 99 & 99 & 99 & 99 & 99 \\ 47 & 66 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \end{bmatrix}$$

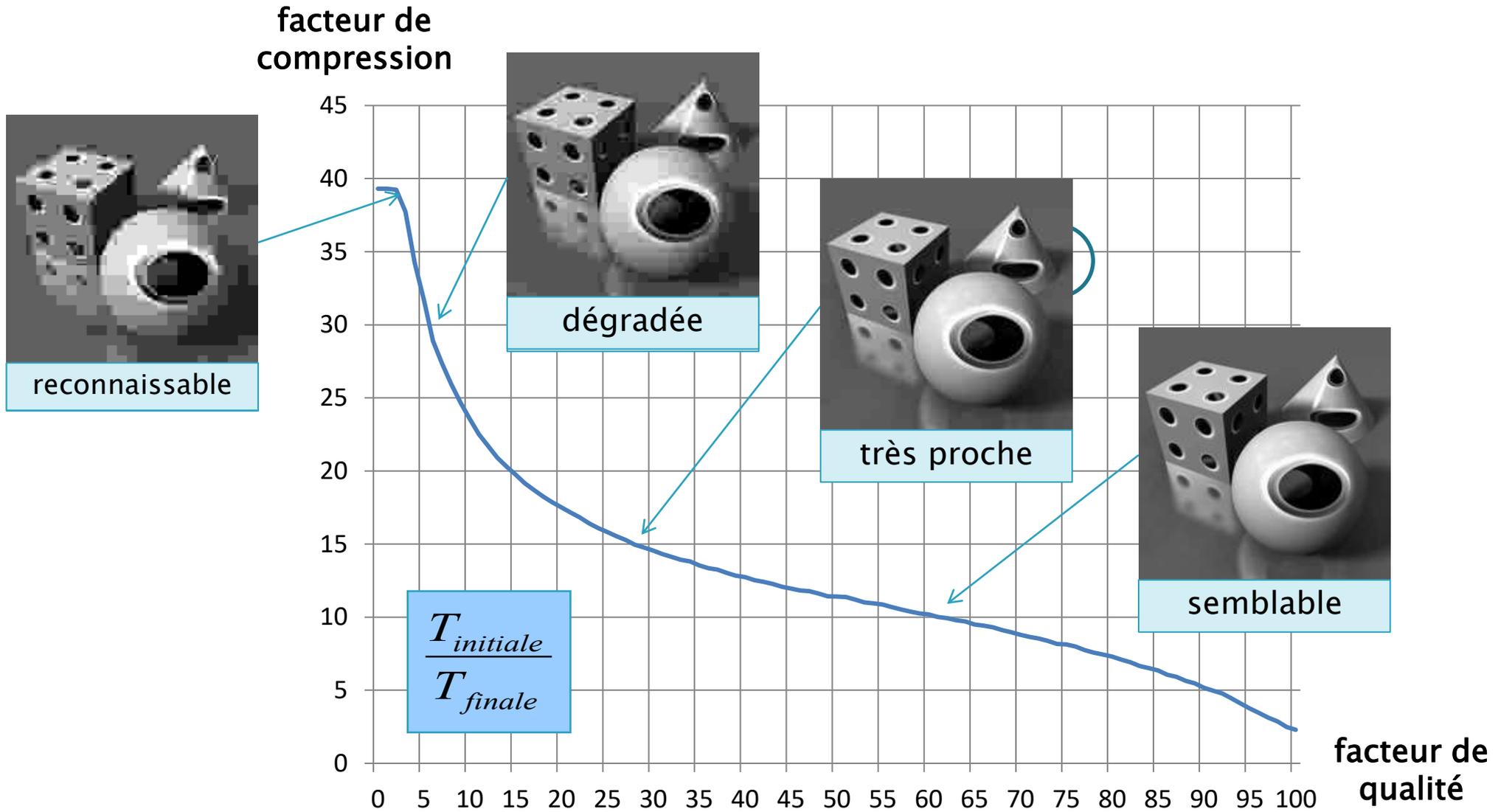
Chrominance (exemple)

$\log(\alpha)$



q
(facteur
de
qualité)

Performances

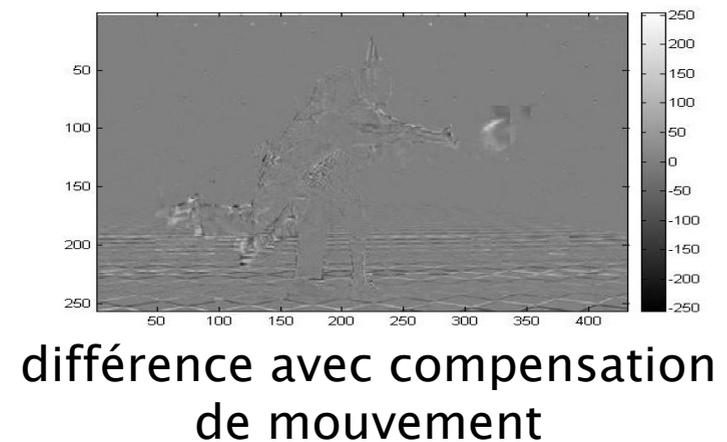
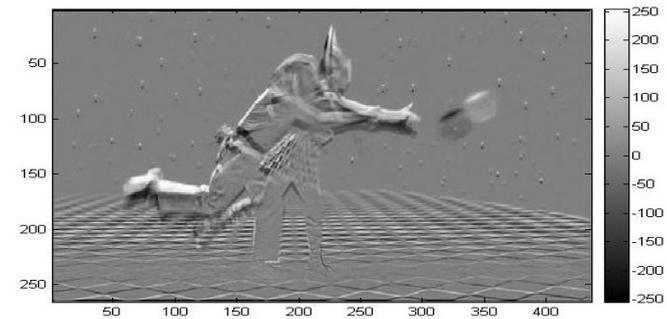
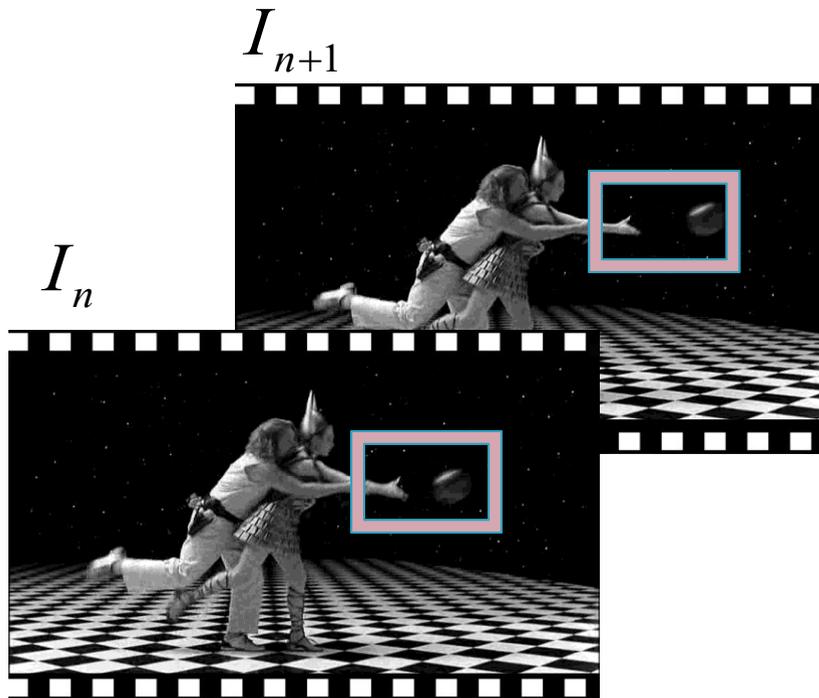


Simulation Jpeg

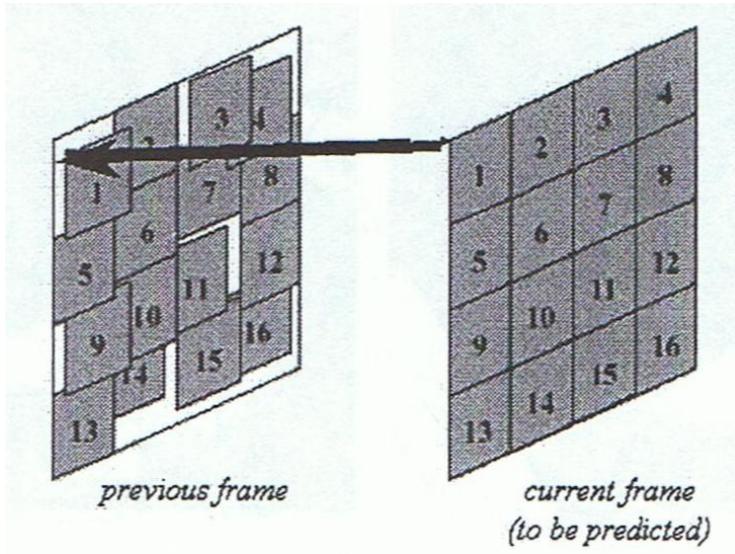
- ▶ Simulation pour une image entière
 - Charger et visualiser une image scalaire (exemple : 'cameraman.tif')
 - Appliquer la chaîne complète (ensemble des blocs 8×8 juxtaposés)
 - ✓ Appliquer la DCT2D (`scipy.fftpack.dct`)
 - ✓ Appliquer la quantification avec une matrice et un facteur de qualité à choisir et une troncature (`numpy.trunc`) des valeurs obtenues
 - Réaliser la reconstruction
 - ✓ Déquantification avec une séquence de calculs inverses
 - ✓ Application de la DCT2D inverse (`scipy.fftpack.idct`)
 - Visualiser les images intermédiaires des différentes étapes (valeurs signées et prise en compte de l'importance de la valeur nulle)
 - ✓ Avant la quantification
 - ✓ Après la quantification
 - Visualiser l'image reconstruite et la comparer à l'image initiale

Compression Vidéo

Séquence et mouvement

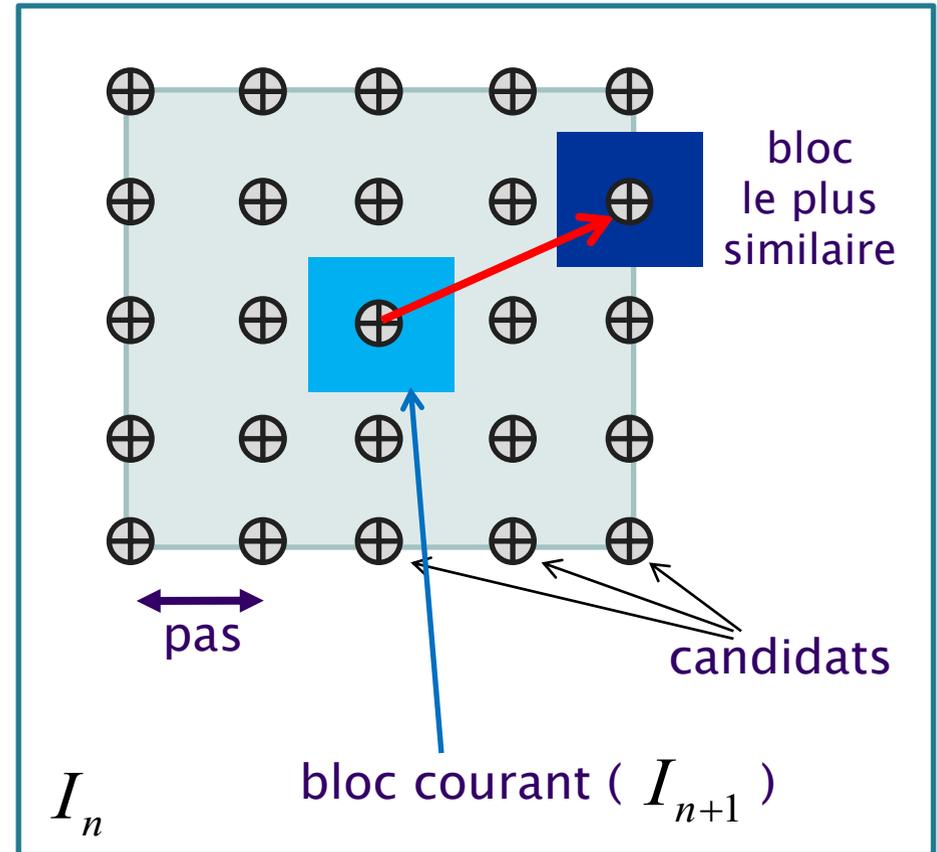


Block Matching (1 / 2)



I_n

I_{n+1}



Block Matching (2/2)

▶ Paramètres

- Découpe en blocs
 - ✓ 4×4 , 8×8 , 16×16 , ...
- Fenêtre de recherche de candidats
- Algorithme de recherche de candidats
 - ✓ Exhaustif
 - ✓ Diamant
 - ✓ ...
- Critère de similarité
 - ✓ SAD
 - ✓ SSE
 - ✓ ...

$$\sum_{\substack{i,j \\ (u,v) \in \Omega}} |I_{n+1}(i, j) - I_n(i + u, j + v)|$$

Compensation de mouvement (1 / 3)

- ▶ Lire et afficher deux images consécutives I_1 et I_2 de la séquence « *football* »
- ▶ Ecrire une fonction qui estime le mouvement « backward » de « I_2 vers I_1 » par la méthode « Block Matching »
 - Découpage en blocs de 8×8 pixels
 - Recherche exhaustive sur des fenêtres de taille 15×15
 - Critère SAD
- ▶ Afficher le mouvement
 - Images (composantes horizontale et verticale, amplitude, direction)
 - Vecteurs (flèches) en surimpression (`matplotlib.pyplot.quiver`) de l'image traitée

Compensation de mouvement (2 / 3)

- ▶ Lire trois images consécutives I_1 , I_2 et I_3
- ▶ Calculer et afficher les différences simples $I_2 - I_1$ et $I_2 - I_3$
- ▶ Calculer les mouvements « backward » de « I_2 vers I_1 » et « forward » de « I_2 vers I_3 »
- ▶ Calculer et afficher les images
 - I_1^C obtenue après compensation de mouvement « backward »
 - I_3^C obtenue après compensation de mouvement « forward »
- ▶ Calculer et afficher les différences « compensées »
 $I_2 - I_1^C$, $I_2 - I_3^C$ et $I_2 - (I_1^C + I_3^C) / 2$

Compensation de mouvement (3 / 3)

- ▶ Comparer les entropies et les énergies (écart-type de l'intensité) de
 - I_1, I_2, I_3
 - $I_2 - I_1, I_2 - I_3$
 - $I_2 - I_1^C, I_2 - I_3^C$
 - $I_2 - (I_1^C + I_3^C) / 2$
- ▶ Calculer et afficher les entropies et les énergies sur tous les triplets d'images successives de la séquence « *football* »