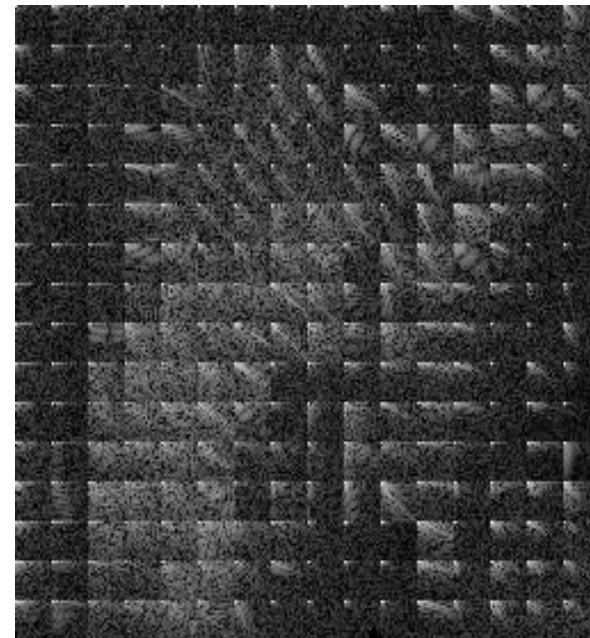
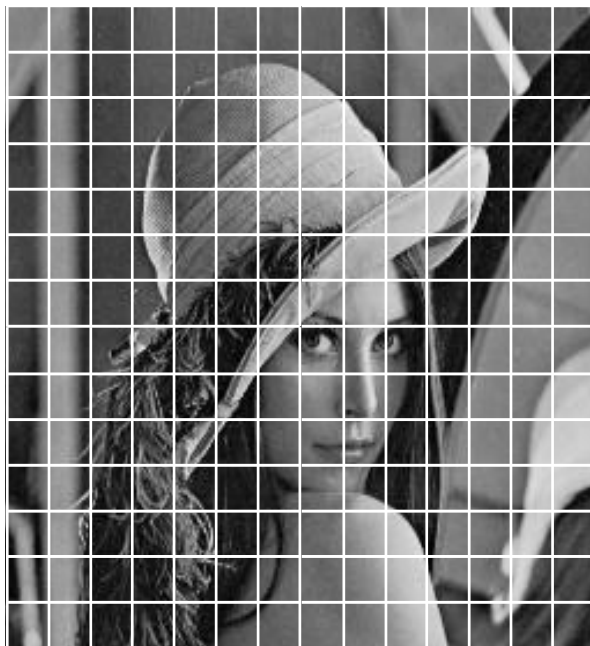


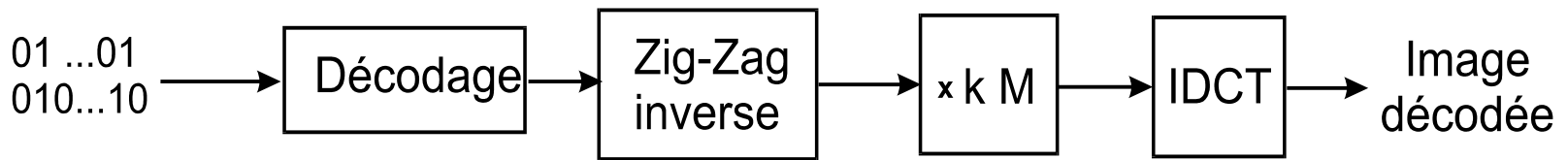
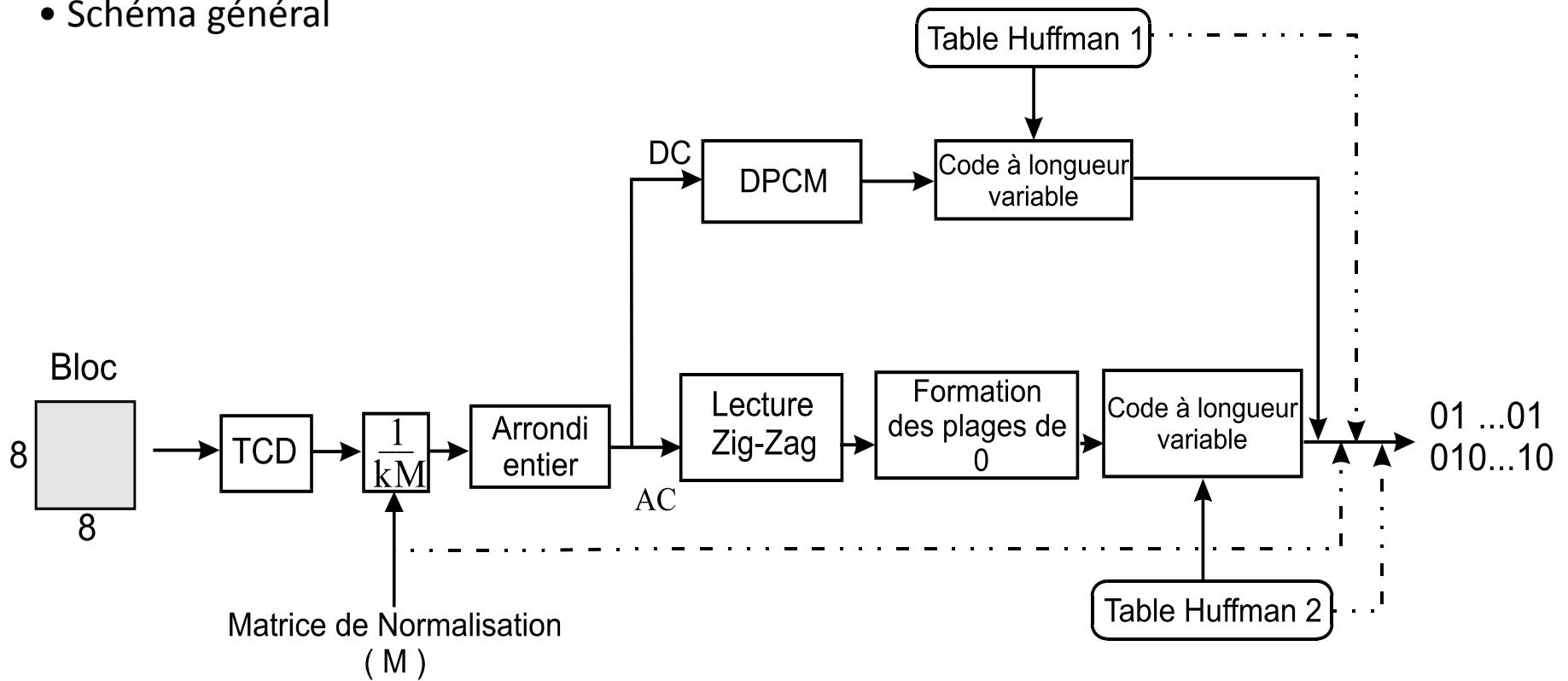
## ✓ Compression DCT bloc : JPEG (1989)

- DCT bloc 8x8
  - ⇒ homogénéité locale de l'image
  - ⇒ l'erreur de quantification est localisée au bloc



TCD bloc (16x16)

- Schéma général



- Matrice de normalisation

⇒ allocation des bits aux coeffs avant quantification par arrondi

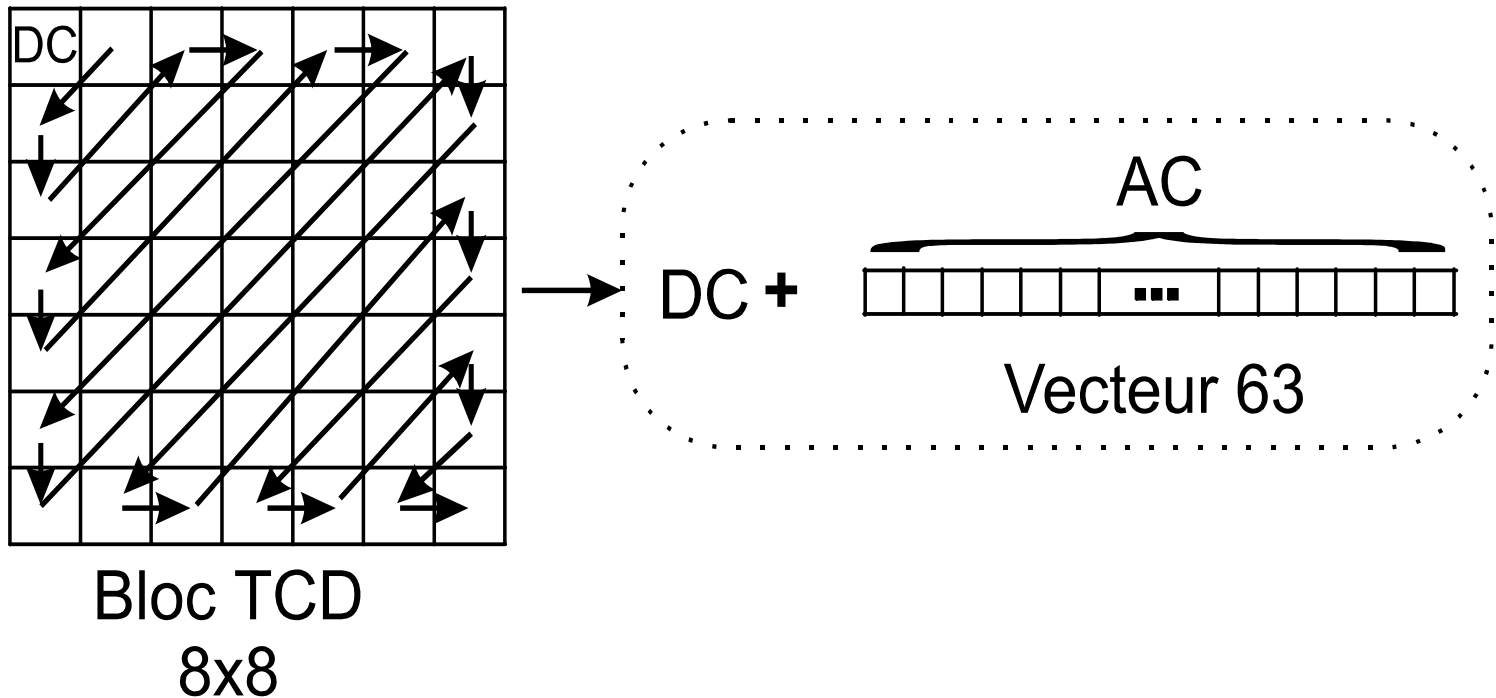
16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Matrice luminance

Matrice chrominance

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

- Lecture zig-zag
  - ⇒ prise en compte de la répartition spatiale de l'énergie pour faire apparaître de longues plages de coeffs nuls



- Codage du coeff DC
  - ⇒ DPCM d'ordre 1 + Huffman

- Codage des coeffs AC
  - ⇒ Codage hybride : runlength + ... + Huffman

- Huffman = Code (plage de 0 + catégorie)

$$162 \text{ codes} : 10_{\text{cat}} \times 16_{\text{lp}} + 2_{(\text{EOB}+16)}$$

Cat.	Intervalle des coefficients AC
1	-1 _ 1,
2	-3, .. , -2 _ 2, .. ,3
3	-7, .. , -4 _ 4, .. ,7
4	-15, .. , -8 _ 8, .. ,15
5	-31, .. , -16 _ 16, .. ,31
6	-63, .. , -32 _ 32, .. ,63
7	-127, .. , -64 _ 64, .. ,127
8	-255, .. , -128 _ 128, .. ,255
9	-511, .. , -256 _ 256, .. ,511
10	-1023, .. , -512 _ 512, .. ,1023

# AC $\Rightarrow$ | huffman | signe | k-1 bits |

- Exemple

$0^{-2} -1 0^2 -1 0^{46} \Rightarrow 111001\ 0\ 0 / 00\ 0 / 11011\ 0 / 1010$

- Extrait de la table d'Huffman des AC

Plage de Zéros	Catégorie	Code
0	1	00
0	2	01
0	3	100
0	4	1011
.	.	.
1	1	1100
1	2	111001
1	3	1111001
1	4	111110110
.	.	.
2	1	11011
2	2	11111000
.	.	.
3	1	111010
.	.	.
16		11111010
EOB		1010

- Remarques

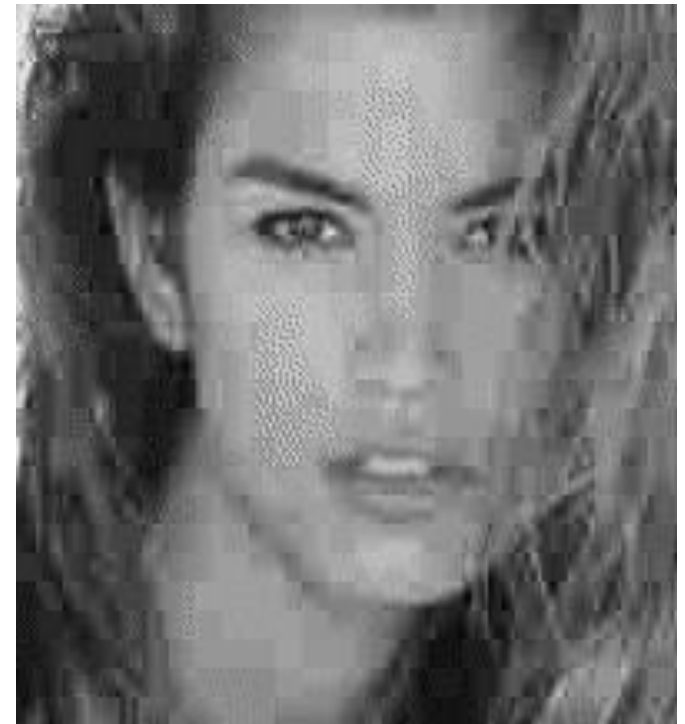
⇒ JPEG = méthode générale ☒ à adapter ...

☺ Très performant à taux faibles (#10)

☹ Effets de blocs à taux élevés

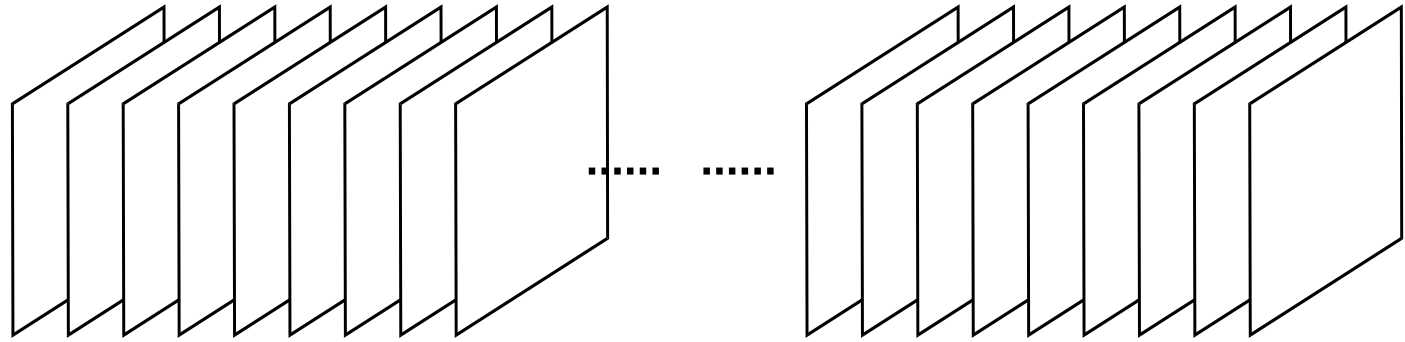


$T_c = 10 / RSB = 30.1 \text{ dB}$



$T_c = 20 / RSB = 28.7 \text{ dB}$

## d) Compression de séquences d'images



- ✓ Supprimer la **redondance spatiale** ou intra-image
  - >> Approches 2D
- ✓ Supprimer la **redondance temporelle** ou inter-image
  - >> Utiliser le déjà vu et le mouvement

# ✓ Les normes **MPEG**

## □ **H261** (1988)

- ↳ La base de la compression de séquences d'images
  - Block matching
  - DCT bloc + Run length + DPCM

## □ **MPEG 1** (1988-92)

- ↳ Vidéo + Audio / 1.5 Mbs ⇒ **CDI**

## □ **MPEG 2** (1990-94)

- ↳ 4-30 Mbs ⇒ **TV numérique** (Digital Video Broadcasting)

## □ **MPEG 4** (1996-99)

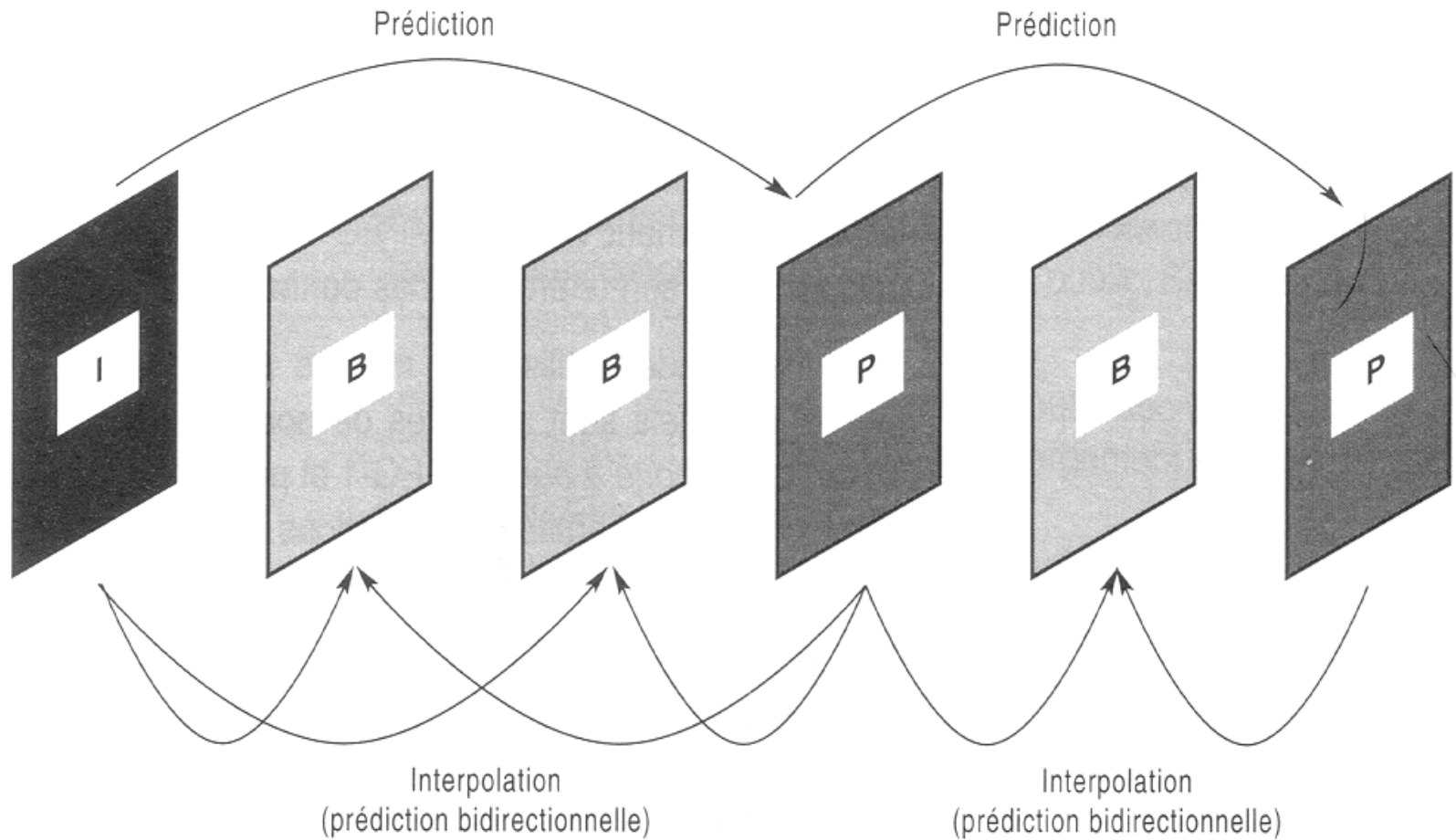
- ↳ L'approche **multimédia** interactif

## □ **MPEG 7** (1997-01)

- ↳ **Indexation** & recherche d'information

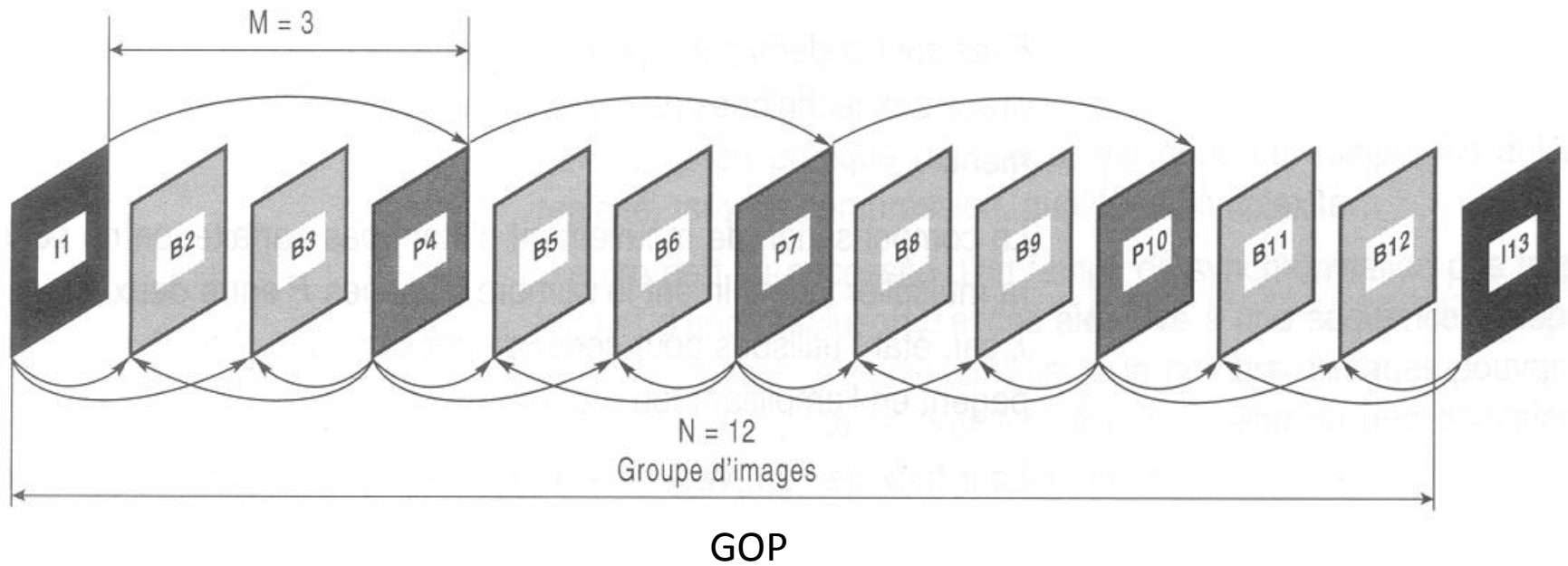
# ✓ Les bases de H261 à MPEG2

➤ 3 types d'images : 3 codages



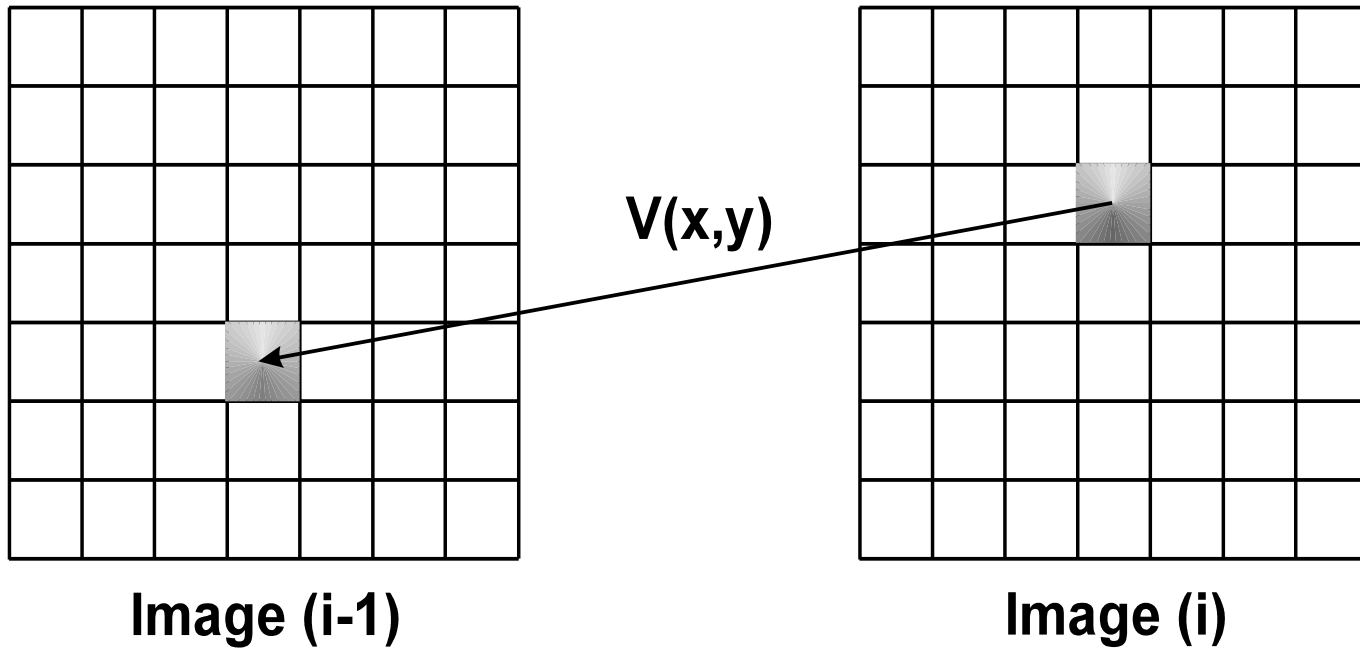
- Images I (intra)
  - Codées JPEG'
  - Point d'accès séquence (0.5s)
  - Tc faible
- Images P (Prédites)
  - Prédites à partir de I ou P
  - Codage DPCM des vecteurs mvt
  - Codage JPEG\* de l'erreur de prédiction
  - Tc élevé
  - Propagation de l'erreur
- Images B (Bidirectionnelles)
  - Interpolées à partir des I P
  - Tc le plus élevé

- 2 paramètres de réglage
  - N : distance inter-I (#12)
  - M : distance inter-P (#3)



✓ Estimation du mouvement par **block matching**

- Blocs 16x16
- Compromis simplicité / efficacité
- Rapide : algorithme logarithmique



✓ Le codage des images P

1- Calcul des  $V_j$  entre

$$I(n) \text{ et } \hat{I}(n-1)$$

2- Synthèse de  $I_p(n)$  :

3- Calcul de l'erreur :  $E(n) = I_p(n) - \hat{I}(n-1) = I_p(n)$

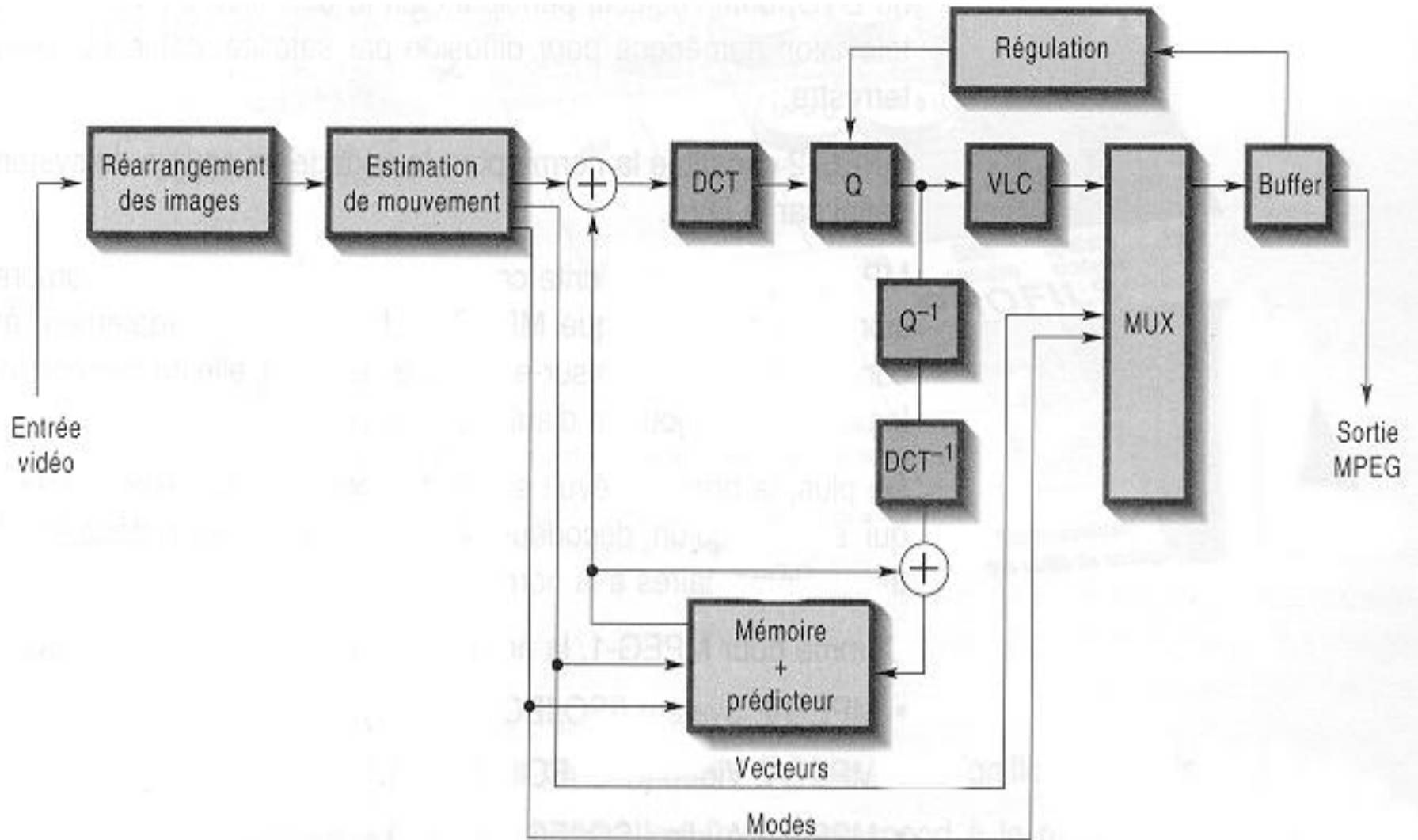
4- Codage JPEG\* de  $E(n)$

*4<sub>bis</sub> - Mémorisation de*

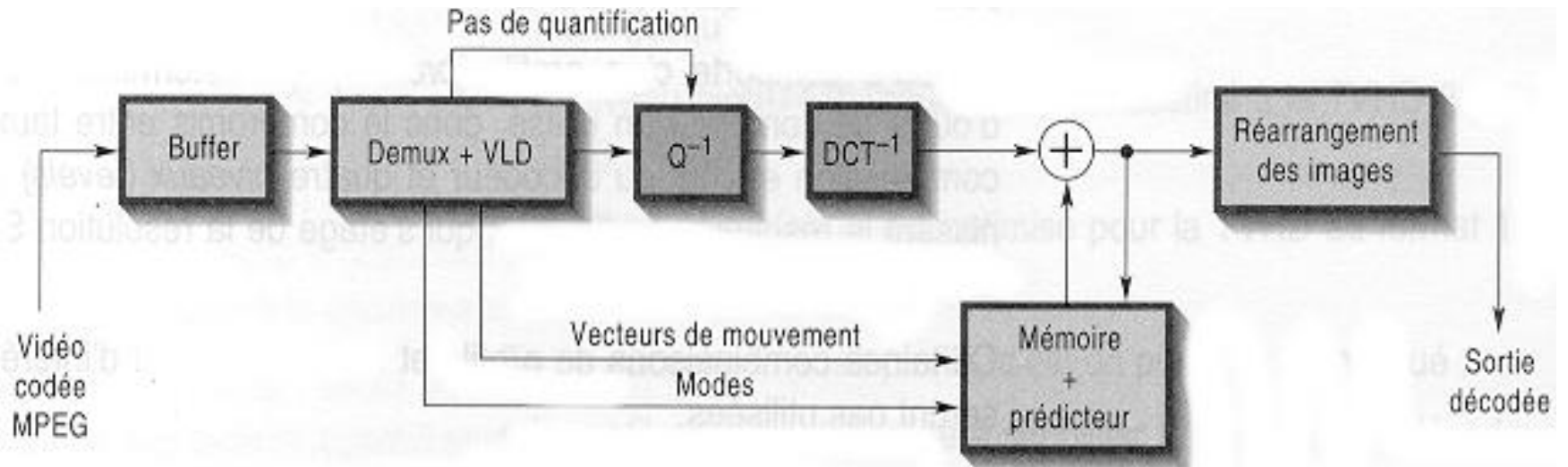
5- Codage DPCM des  $V_j$

$$\hat{I}(n)$$

✓ Codeur MPEG2



## ✓ Décodeur MPEG2



# ✓ Codage et TVnum

- Numérisation brute : 200 Mb/s
- DVB # DVD = MPEG2 MP@ML
  - 720 x 480/576 (30/25 Hz) avec IPB
  - 4 Mb/s (PAL/SECAM) à 9 Mb/s (studio)
  - Tc de 40 à 18