

Quantification Vectorielle pour le Challenge Programming

Gilles Menez

Université de Nice – Sophia-Antipolis
Département d'Informatique
email : menez@unice.fr
www : www.i3s.unice.fr/~menez

19 novembre 2020: V 1.0

Vector Quantization : Why ?

La quantification vectorielle trouve ses racines dans la compression de l'information :

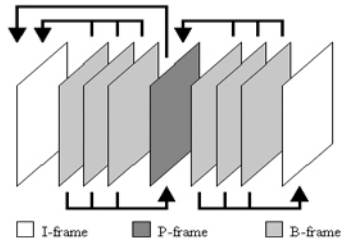
- Compression : Réduire la taille d'une information !

On distingue :

- la compression **sans perte**, qui permet de reconstituer exactement les données d'origine, de la compression en général,
- **avec pertes**, qui "abandonne" des données jugées non nécessaires, au profit de la diminution encore plus importante du débit ou de la taille des fichiers.

Vector Quantization : Image compression

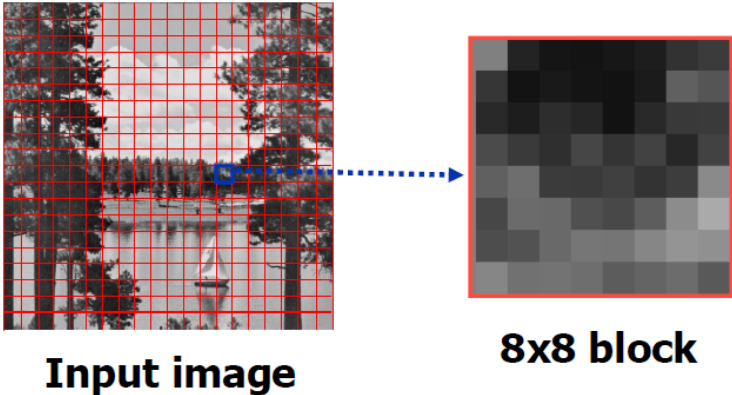
Le volume des données font qu'elle est souvent utilisé dans le contexte de l'image et des séquences d'images : les frames !



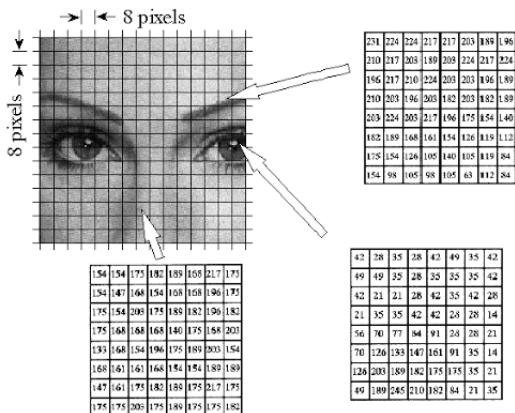
Il n'y a pas "que" le volume des données ... la nature redondante aussi.

Image : blocs, alphabets, vecteurs

Dans ce qui suit, mais aussi de façon générale, une image numérique est décomposée puis traitée en blocs d'une taille prédéfinie :



Les blocs de l'image peuvent être assimilés à des motifs ou un alphabet composant l'image.



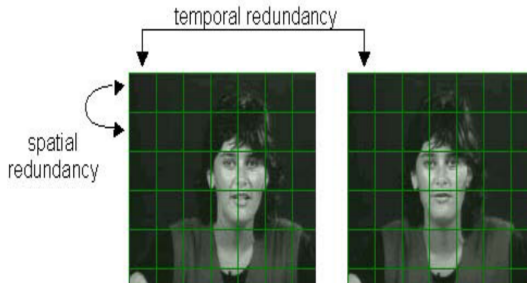
On peut aussi considérer que chaque bloc est un vecteur ...

➤ dans cet exemple dans \mathbb{R}^{64}

Video Compression : Redundancy

Un premier "réflexe" de compression est sans aucun doute celui de supprimer les "doublons" ... la plus évidente/simple des "redondances".

Sur cet exemple de présentatrice, la redondance se manifeste par une "répétition" en temps et en espace.

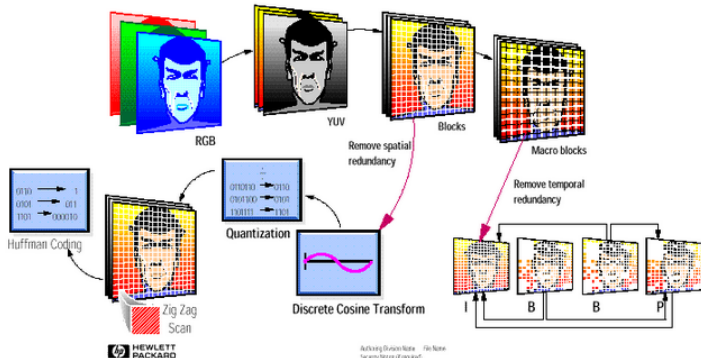


RMQ : La redondance est un concept bien plus complexe que la "duplication" :

- <https://faraday.emu.edu.tr/ee583/Lectures/EE%20583-Lecture09.pdf>
- [https://en.wikipedia.org/wiki/Redundancy_\(information_theory\)](https://en.wikipedia.org/wiki/Redundancy_(information_theory))

MPEG

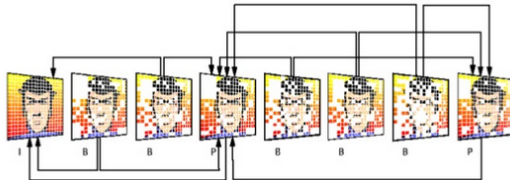
Si on considère un flux vidéo (frames) comme une "information", force est de constater que cette information peut être **extrêmement redondante** !



MPEG encoding process try to remove Spatial and Temporal redundancy

Inter-frame

Inter-frame compression exploits redundancy between "several successive frames" : temporal redundancy reduction !



- *I* frames are independently encoded
- *P* frames are based on previous *I*, *P* frames
 - Can send motion vector plus changes
- *B* frames are based on previous and following *I* and *P* frames
 - In case something is uncovered

L'image I est utilisée pour prédire la première image P puis à partir de ces deux images, le processus prédit les deux premières images B.

La seconde image P est prédite à partir de la première P-frame et les deux images B comprises entre ces deux images seront les suivantes.

Intra-frame

Intra-frame compression exploits redundancy in a single "frame" : spatial redundancy reduction !

JPEG est un exemple typique de "compression intra-frame".

- Elle utilise la Discrete Cosinus Transform ($X_k = \sum_{n=0}^{N-1} x_n \cos \left[\frac{\pi}{N} \left(n + \frac{1}{2} \right) k \right]$)

- F_{00} and F_{01} are large.
- For $u > 0$, they are almost 0.



122	121	126	122	117	119	119	116
120	121	127	121	116	114	120	122
113	115	124	114	119	113	117	120
112	109	116	110	112	113	114	116
103	103	111	114	112	106	110	121
100	102	110	105	111	107	111	106
86	93	95	92	91	87	81	78
47	53	57	58	64	72	62	59

Block with Horizontal edge



834	-5	-9	-10	-3	0	5	7
131	8	11	-7	1	4	0	2
-65	7	-5	6	-7	0	0	-5
47	5	-1	-2	2	3	-2	0
-30	-10	1	3	2	1	3	-5
10	13	-2	4	-3	0	1	1
-4	-10	-2	4	-3	-1	0	3
0	3	-2	-5	4	-2	0	-1

DCT of Horizontal edge

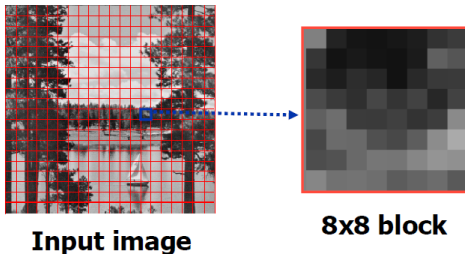
La DCT possède une excellente propriété de "regroupement" de l'énergie :

- l'information transformée est essentiellement portée par les coefficients basses fréquences.

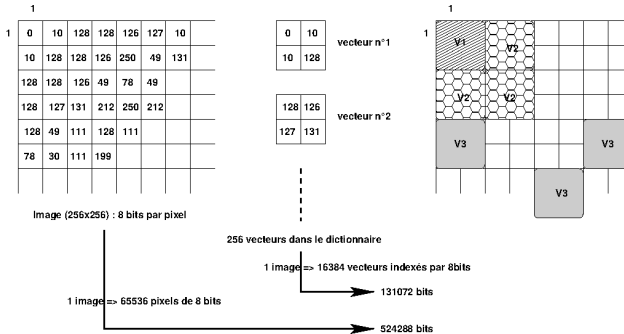
Et la quantification vectorielle ?

La quantification vectorielle est une technique de compression avec perte qui s'appliquera intra-frame.

Vous vous souvenez de l'histoire de l'alphabet/(les motifs/les vecteurs) ... une image étant un "mot" formé à partir de cet alphabet.



Ceci signifie qu'une image est représentée par un dictionnaire (codebook) de vecteurs (les codewords) et un ensemble d'index :



Par exemple, dans la cadre d'une transmission réseau, si le récepteur est en possession du dictionnaire, la transmission de l'image se résume à la transmission des index des vecteurs la composant.

- Bien entendu dans le cadre d'une séquence d'images, le dictionnaire peut servir à plusieurs frames (c'est de l'inter-frame) !

Compression avec Pertes

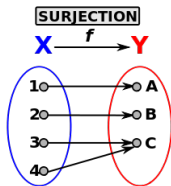
La quantification étant la fonction qui associe un vecteur (v) de l'image originelle un vecteur du dictionnaire (w_j : codeword j) :

$$\begin{aligned} Q : V &\rightarrow W \\ v &\rightarrow Q(v) = w_j \end{aligned}$$

Pour comprimer encore plus l'information, la QV va chercher à réduire la taille de l'alphabet et donc le nombre de symboles contenus dans le dictionnaire.

- L'idée étant de représenter les vecteurs "similaires" par un même codeword ... spatial redundancy removal.

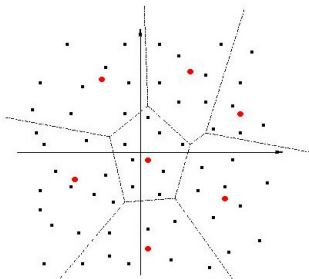
Dès lors, la quantification devient une surjection et donc il y a perte d'information !



Modélisation

Les vecteurs $\in \mathbb{R}^2$ de l'image sont représentés dans le plan :

- Les points rouges sont les codewords et les points noirs les vecteurs de l'image.



La "qualité" d'une QV vient de la minimisation des distances entre ces points.

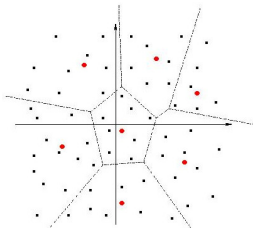
- ① Choisir les "bons" codewords ... qui représenteront au mieux les vecteurs de l'image (Algorithme LBG : hors du scope du challenge),
- ② Pour un dictionnaire donné, réaliser la meilleure quantification possible, celle qui minimise l'erreur commise (en moyenne?).

Challenge

Le "challenge" qui vous est proposé vient de là :

- Réaliser la meilleure quantification possible, celle qui minimise l'erreur commise (en moyenne ?).

Ceci signifie **pour chaque vecteur (v) trouver le codeword (w) le plus proche** au sens d'une distance par exemple euclidienne : $\sqrt{\sum_i (w_i - v_i)^2}$



Cette recherche ne peut pas être naïve car sinon elle est beaucoup trop lourde en temps de calcul ... on doit maintenir un rythme d'images et vous n'êtes pas en \mathbb{R}^2 !

- Spécification supplémentaire : si un vecteur v venait à être équidistant de plusieurs codeword w_i , on choisirait celui d'indice le plus petit.

Algorithmique de Challenge

Trouver une méthode pour minimiser le temps de quantification avec un dictionnaire donné.

On vous donne un point de départ avec quelques méthodes :

<http://docnum.univ-lorraine.fr/public/UPV-M/Theses/1992/Nyeck.Alain.SMZ928.pdf>

L'évaluation sera basée sur les critères :

- Résultats Sphere Engine
- Description de(s) l'algorithme(s) utilisé(s)
- Qualité du code
- Sans doute benchmarking sur des frames (4K \Rightarrow 8,3 Méga pixels en 16 :9) ... si je trouve une machine ?