



UNIUNEA EUROPEANĂ



GUVERNUL ROMÂNIEI



Instrumente Structurale
2007-2013



Platformă de e-learning și curriculum e-content pentru învățământul superior tehnic

Transmisia datelor multimedia in rețele de calculatoare

40. Compresia prin cuantizarea vectoriala

Compresia prin cuantizarea vectoriala (*VQ=vector quantization*)

- Cuantizarea vectoriala este generalizarea cuantizarii scalare la cunatizarea unui vector
 - Saltul de la o dimensiune la mai multe dimensiuni atrage noi concepte, tehnici si aplicatii
 - In majoritatea cazurilor intrarea este sub forma numerica iar iesirea este o forma comprimata.

Compresia prin cuantizarea vectoriala

- In contrast cu cuantizarea scalara, VQ aplica cuantizarea nu la nivel de esantion ci la nivel de grup de esantioane pentru a forma un vector si apoi cuantizeaza acel vector
 - Deci, imaginea digitala este mai intai prelucrata pentru furnizarea unui set de vectori
 - Apoi, se genereaza un set al vectorilor reprezentativi, asa numitul codebook.
 - Compresia se obtine prin inlocuirea vectorului reprezentativ cu un index
 - Acest index este adresa unui tabel ce contine vectorii reprezentativi
 - Rezulta ca fiecare vector este codat cu doua piese de informatie distincte:
 - indexul
 - cuvantul de cod corespunzator
 - Modificarea dictionarului in timpul codarii si/sau a cuvintelor de cod din cadrul tabelului (vectorii reprezentativi) determina obtinerea unor variante adaptive ale cunatizarii vectoriale

Compresia prin cuantizarea vectoriala

- *Cuantizarea vectoriala* este o metoda de compresie cu pierdere de informatie, care se opreste asupra unei multimi de pixeli, in loc sa se uite la pixelii individuali
 - Deci, o multime de pixeli cu aceleasi proprietati statistice (valori apropiate ale stralucirii) sunt inlocuiti cu un singur element de imagine, in imaginea comprimata
 - Cuantizarea vectoriala se utilizeaza in multe aplicatii de compresie audio si video, in recunoasterea vorbirii

Compresia prin cuantizarea vectoriala

- O cale naturala de a aplica tehnica VQ la imagini este:
 - descompunerea imaginii numerice in blocuri dreptunghiulare/patrata de dimensiune fixata
 - apoi sa se utilizeze aceste blocuri ca sub forma de vectori
- Un cuantizor vectorial transforma spatiul vectorial R^k al vectorilor k -dimensionali intr-o multime finita de vectori $Y = \{ \mathbf{y}_j, j = 1, 2, \dots, N \}$
 - Fiecare vector \mathbf{y}_j se numeste *vector cod*
 - Multimea cuvintelor de cod se numeste *dictionar (codebook)*

Compresia prin cuantizarea vectoriala

- Fiecare cuvânt de cod are asociat un domeniu în spațiul k -dimensional, denumit domeniu Voronoi, după regula:

$$V_i = \left\{ \mathbf{x} \in R^k : \|\mathbf{x} - \mathbf{y}_i\| \leq \|\mathbf{x} - \mathbf{y}_j\|, \quad \forall j \neq i \right\}$$

- Domeniile (regiunile) Voronoi împart spațiul R^k astfel încât:

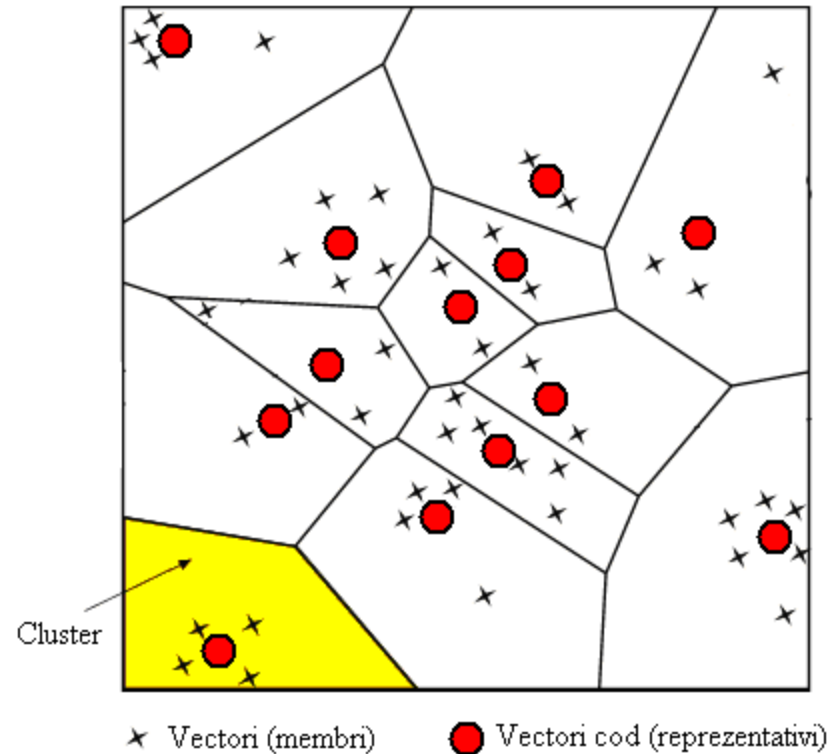
$$\bigcup_{i=1}^N V_i = R^k$$

și

$$\bigcap_{i=1}^N V_i = \emptyset, \quad \forall i \neq j$$

Exemplu

- Asociat cu fiecare regiune (sau cluster) se gaseste un cuvânt de cod
- Fiecare regiune are un singur cuvânt de cod
- Aceste regiuni sunt separate prin linii imaginare, trasate cu linie continua
- Pentru un vector de intrare, cuvântul de cod ce este ales este acela din care face parte vectorul de intrare.



Cuvinte de cod in spatiul bidimensional.
Vectorii de intrare sunt marcati cu „x”, cuvintele de cod sunt reprezentate prin cercuri, iar regiunile Voronoi sunt separate prin linii

Compresia prin cuantizarea vectoriala

- Cuvantul de cod reprezentativ este determinat de cea mai mica distanta Euclidiană de vectorul de intrare
- Distanta Euclidiană definită prin:

$$d(\mathbf{x}, \mathbf{y}_i) = \sqrt{\sum_{j=1}^k (x_j - y_{ij})^2}$$

Algoritm de cuantizare vectoriala

Begin

Operatii off-line:

#1: Stabileste dim. vectorului (domeniului) functie de eroarea acceptata

#2: Defineste setul de vectori de cod (imagini)

Operatii on-line:

#1: Prezinta imaginea de comprimat

#2: Imparte in blocuri (vectori) de aceeasi dimensiune cu vectorii de cod

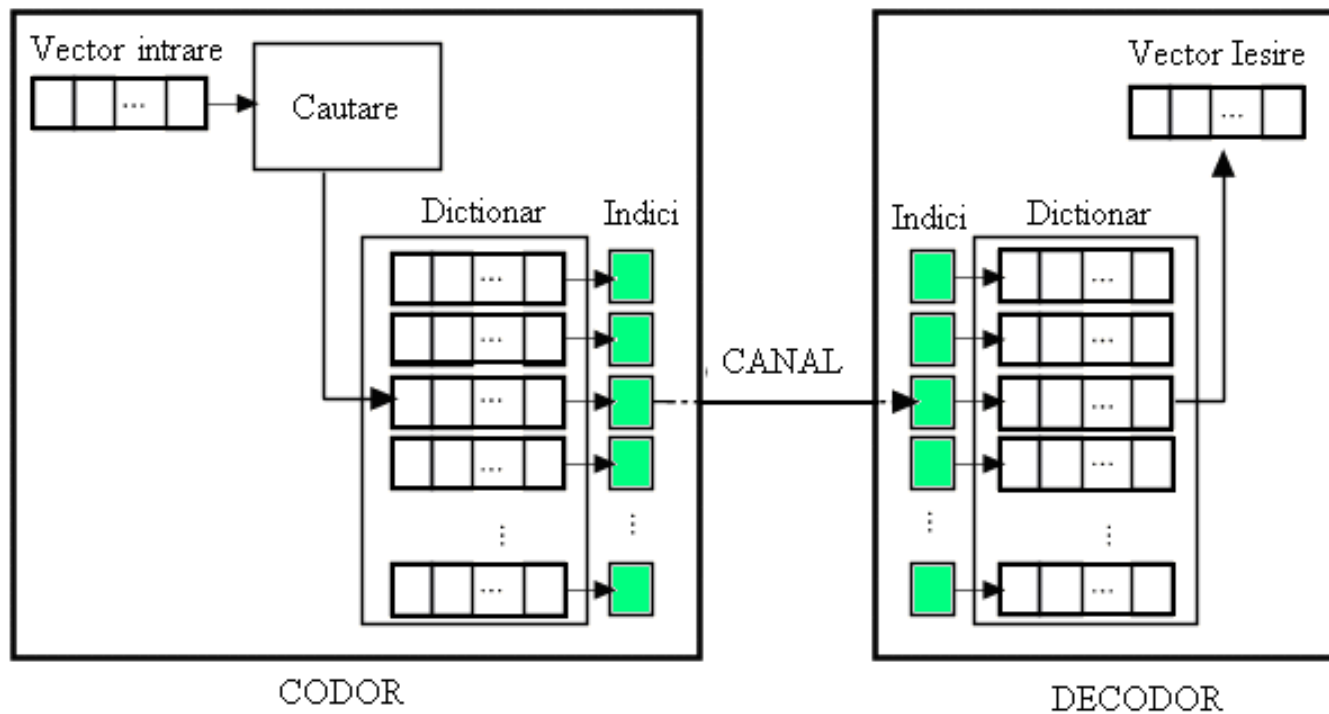
#3: PENTRU fiecare bloc al imaginii EXECUTA

#3.1: Cauta cel mai aproape vectorului de cod

#3.2: Codeaza si transmite indicele vectorului de cod.

End

Schema bloc



Schema bloc

- La codare se calculeaza indicele cuvintului de cod ce ofera distanta cea mai mica fata de vectorul de intrare
 - Criteriul folosit este cel de distorsiune minima, bazat pe distanta Euclidiana dintre vectorul de intrare si fiecare cuvint de cod memorat/cunoscut
 - Pe canal se trimite indexul acelui cuvint de cod
 - La de-compresie/decodare, se inlocuieste indexul cu cuvintul de cod corespunzator
- Operatia cea mai dificila este proiectarea cuvintelor de cod, deci de a stabili numarul de cuvinte de cod si modul de calcul al cuvintelor de cod pentru fiecare regiune

Schema bloc

- Desi, cuantizarea vectoriala ofera aceeasi rata a distorsiunii ca si cuantizarea scalara sau PCM, metoda nu are o raspandire foarte larga in aplicatiile comerciale
- Sunt doua motive:
 - primul se refera la timpul necesar generarii cuvintelor de cod
 - al doilea, se refera la viteza de cautare a acestora in multimea cuvintelor de cod
- Cea mai simpla metoda de cautare, aceea a cautarii totale, un vector de intrare este comparat cu toti vectorii cod (*full search method*)
- Numarul de operatii fiind foarte mare, metoda cautarii totale este costisitoare

Exemplu

- Se considera imaginea „cameraman”, prezentata in figura de mai jos

Imaginea initiala



11944 vectori



9764 vectori

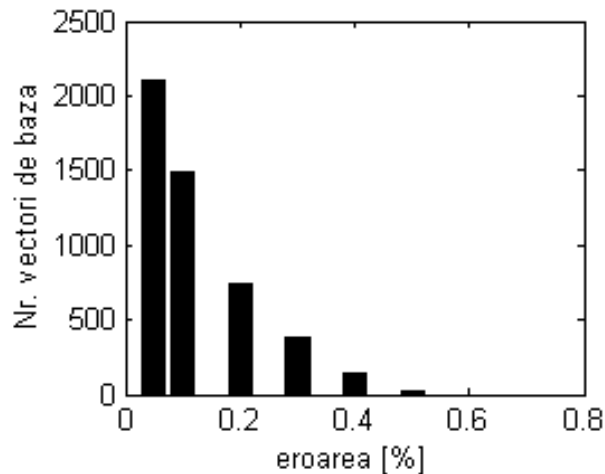


8200 vectori



Exemplu

- Prin impunerea unei erorii patratice medii mai mici decat un anumit procent din energia celulei de baza (ce defineste cuvintele de cod reprezentativi) se obtin rezultatele :
 - Dependenta numarului de vectori din dictionar functie de eroarea impusa



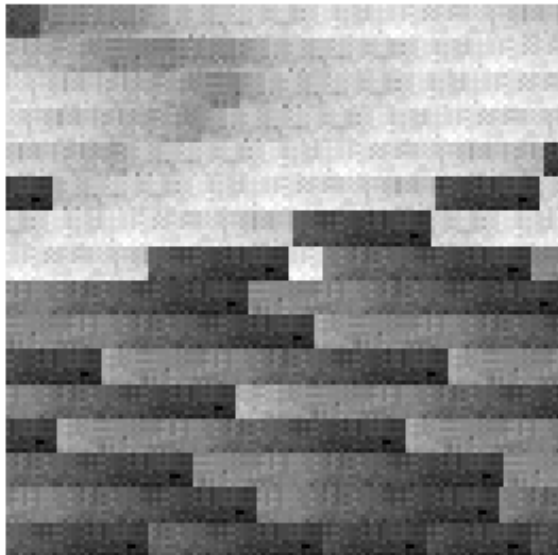
```
per_set = [ 0.0500 ... 0.5000];
```

```
Nr. vectori = [ 2096 ... 16];
```

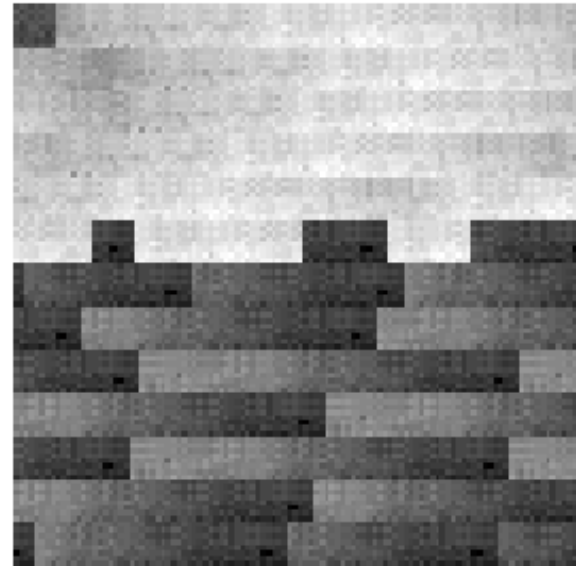
Exemplu

- Vectorii de cod pentru diferite valori ale distorsiunii patratice medii, ca procente din energia celulei de baza

2096 celule de baza distanta 0.05

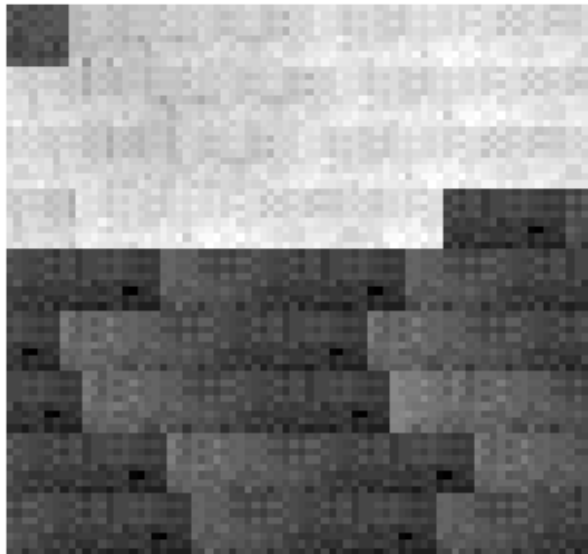


1488 celule de baza distanta 0.1

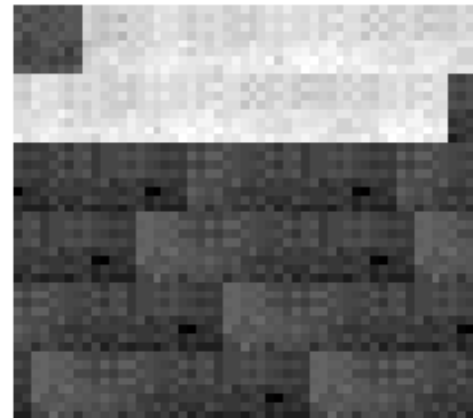


Exemplu

744 celule de baza distanta 0.2



376 celule de baza distanta 0.3



Exemplu

- Raportul de compresie este

$$RC = \frac{nc \cdot nl \cdot n_bit_per_pixel}{n_coef \cdot n_bit_per_index} = \frac{nc \cdot nl \cdot n_bit_per_pixel}{n_coef \cdot [\log_2(n_coef)]}$$

unde

nc este numarul de coloane

nl este numarul de linii

$n_bit_per_pixel$ este numarul de biti pentru reprezentarea intensitatii unui pixel

n_coef este numarul de coeficienti considerati in transformare

$n_bit_per_coef$ este numarul de simboluri binare pentru reprezentarea unui coeficient

Exemplu

- Se considera imaginea initiala „cameraman.tif” de dimensiune $256*256*8$ biti (imagine gri)
- Celula de baza are dimensiunea de $4x4$
- Se considerara un numar diferit de vectori de baza in cadrul dictionarului, de 11.944, 9.764 si 8200

Exemplu

$$RC = \frac{256 \cdot 256 \cdot 8}{11944 \cdot [\log_2(11944)]} = \frac{256 \cdot 256 \cdot 8}{11944 \cdot 14} = 3.65$$

$$RC = \frac{256 \cdot 256 \cdot 8}{9764 \cdot [\log_2(9764)]} = \frac{256 \cdot 256 \cdot 8}{9764 \cdot 14} = 4.03$$

$$RC = \frac{256 \cdot 256 \cdot 8}{8200 \cdot [\log_2(8200)]} = \frac{256 \cdot 256 \cdot 8}{8200 \cdot 14} = 4.56$$

Evolutia standardului MPEG

- MPEG-1 (1991) (ISP/IEC 11172)
 - debit de informatie pana la 1.5 Mbps
 - formatul de imagine tipic CIF (Common Interface Format)
 - frecventa cadrelor 24 ... 30 fps
 - Aplicatiile principale: stocarea informatiei video pentru multimedia (CD-ROM);
- MPEG-2 (1994) (ISP/IEC 13818)
 - Extensie pentru metodele cu intretesere, optimizat pentru rezolutia TV
 - Calitatea imaginii similara cu NTSC, PAL, SECAM la 4-8 Mbps
 - HDTV la 20 Mbps;
- MPEG-4 (1999) (ISP/IEC 14496)
 - Codare bazata pe obiecte

Descriere sumara a standardelor folosite in compresia video

Aplicatiile de baza ale compresiei video

| Aplicatia | Debit informatie | Standarde |
|--------------------------------------|---|-----------------------------|
| Televiziune numerica | 2 . . . 6 Mbps (10...20 Mbps for HD) | MPEG-2 |
| Video DVD | 6 . . . 8 Mbps | MPEG-2 |
| Video internet | 20 . . . 200 kbps | H.263, MPEG-4, H.26L/JVT |
| Video-conferinte, video-telefonie | 20 . . . 320 kbps | H.261, H.263 |
| Video sub 3G | 20 . . . 100 kbps | H.263, MPEG-4 |